



HAL
open science

Structured representation learning and multimodal perception, towards more autonomous sensorimotor systems

Mathieu Lefort

► To cite this version:

Mathieu Lefort. Structured representation learning and multimodal perception, towards more autonomous sensorimotor systems. Computer Science [cs]. Université lyon 1, 2024. <tel-04858357>

HAL Id: tel-04858357

<https://theses.hal.science/tel-04858357v1>

Submitted on 29 Dec 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License

HABILITATION À DIRIGER DES RECHERCHES

présentée devant
l'Université Claude Bernard LYON I
Spécialité : Informatique

Structured representation learning and multimodal perception, towards more autonomous sensorimotor systems

par
Mathieu Lefort

Soutenue publiquement le 13 décembre 2024

Composition du jury

<i>Rapporteurs :</i>	Alexandre Pitti	Full Professor at Université Cergy-Pontoise
	Nicolas Rougier	Research Director at Inria
	Rufin VanRullen	Research Director at CNRS
<i>Examineurs :</i>	David Filliat	Full Professor at ENSTA ParisTech
	Élisa Fromont	Full Professor at Université de Rennes
	Alain Mille	Emeritus Professor at Université Lyon 1

Acknowledgements

First of all, I would like to thank all the members of the jury (David Filliat, Éliisa Fromont, Alain Mille, Alexandre Pitti, Nicolas Rougier and Rufin VanRullen) for their valuable remarks and comments on my work, and for the intensive but interesting questions/discussion session during the defence. The work presented in this manuscript was mainly done by people I supervised, so I also would like to thank all the undergraduate students (Clément Cottet, Nathan Gauthier, Yu-Guan Hsieh, Adrien Techer and Christophe Techer), the master students Yoann Ait-Itchou, Quentin Bouchon, Valentin Chaffraix, Nicolas Condomitti, Pierre Couy, Valentin Cuzin-Rambaud, Brice Denoun, Étienne Desbois, Tom Dusséaux, Lucas Fournier, Florent Gattoni, Aleksandra Loginova, Sofia Mabrouk, Louis Manhès, Nawel Medjkoune, Alisa Rieger, Léo Schneider, Mohamed Massamba Sene, Étienne Vareille, Amaury Velasco, José Villamar), the PhD students 'Anaëlle Badier, Nadir Bendoukha, Axel Bessy, Ruiqi Dai, Alexandre Devillers, Simon Forest, Alexandre Galdeano, Julien Lefebvre, Victor Lequay, Nathan Salazar, Pierre-Elliott Thiboud), the post-docs (Nicolas Jacquelin, Simon Pageaud), the engineers (Laurianne Charrier, Pierre Ecarlat, Dorian Goepf, Paul Montcuquet) with whom I had the pleasure of working in the past (or in the near future). As the research process can only be carried out as a collaborative work standing on the shoulders of giants, my thanks also go to the current members of my team (Samir Aknine, Frédéric Armetta, Farah Ben Slama, Yasmine Bouamra, Rémy Chaput, Timon Deschamps, Véronique Deslandres, Laury Diadhiou-Coulon, Olivier Georgeon, Laëticia Matignon, Maya Medjad, Maxime Morge, Brooke Stephenson, Dorian Tonnis, Bruno Yun) and all the co-authors and colleagues of the projects I participated (Xavier Alameda-Pineda, Benoît Alcaraz, Arthur Aubret, Marie Avillac, Anthony Baccuet, Flavien Balbo, Heike Baldauf-Quiliatre, Aurélien Belle, Yann Boniface, Olivier Boissier, Rémy Cazabet, Alan Chauvin, Julien Clavel, Amélie Cordier, Emmanuel Dellandrea, Nicolas Duchateau, Stefan Duffner, Alexander Gepperth, Alix Gonnot, Bernard Girau, Mathieu Guillermin, Erwan Guillou, Nathalie Guyader, Thomas Hecht, Hamid Ladjal, Marie Lefevre, Antoine Louchard, Alexandre Meyer, Yukie Nagai, Bastien Nollet, Julien Perrier-Camby, Mickaël Sdika, Audrey Serna, Céline Teulière, Lucien Tisserand, Jochen Triesch, and the members of Ubiant, Hoomano, Nomad Education, Previa Medical, Atos and Neovision). Among them, I would like to address a special and deep thank you to two of them: Jean-Charles Quinton, who has been guiding me in the sensorimotor theories since his post-doc and my PhD and with whom I still have the pleasure and the chance to collaborate and discuss a lot until today, and Salima Hassas, the former head of the team until her death in 2023, who guided and helped me a lot as a young associate professor. Finally, I would like to thank all the other members of the lab/university that I have not explicitly mentioned (especially the pedagogical and popularisation teams and the BIATSS people) that I interact with during my work, the funding agencies that support my research, and the AI tools that helped me correct typos and English sentence structure.

On a personal note, I would really like to thank Leïla Dada for tolerating my professional agenda, for supporting me as a person every day, and for all the profound discussions that we can have.

Contents

1	Introduction	1
1.1	A brief history of AI	1
1.2	Neural networks	2
1.3	Machine learning	4
1.4	Representation learning	7
1.5	Autonomous agent	9
1.6	Summary of my research work	11
1.6.1	PhD and post doc	11
1.6.2	PhD supervisions	12
2	Self-supervised visual representation learning	16
2.1	Introduction	16
2.2	Learning visual representations	17
2.2.1	Preliminary works	17
2.2.2	Equivariant representation learning	21
2.3	Structure of learned representations	25
2.3.1	Context and objectives	25
2.3.2	Method and results	26
2.4	Conclusion and perspectives	28
3	Multimodal perception	30
3.1	Introduction	30
3.2	Decision making algorithms: from psychophysics to robotics	32
3.2.1	Context and objectives	32
3.2.2	Comparative framework and results	32
3.3	Modeling of audio-visual perception in humans	35
3.3.1	Context and objectives	35
3.3.2	Model and results	36
3.4	Multimodal perception with automatic weighting	39
3.4.1	Context and objectives	39
3.4.2	Models and results	40

3.5	Conclusion and perspectives	44
4	Incremental and active learning	46
4.1	Introduction	46
4.2	Active learning	47
4.2.1	Biased datasets with spurious correlation	47
4.2.2	Algorithmic learning with a multi-task approach	49
4.3	Unsupervised class-incremental learning	51
4.3.1	Context and objectives	51
4.3.2	Model and results	52
4.4	Conclusion and perspectives	53
5	Conclusion and perspectives	55
5.1	Conclusion	55
5.1.1	Summary	55
5.1.2	Integrated view	56
5.2	Future works	58
5.2.1	Positioning	58
5.2.2	Research axes	59
5.3	Impacts of my research	65
6	Appendices	69
6.1	The Impact of Action in Visual Representation Learning	70
6.2	EquiMod: An Equivariance Module to Improve Visual Instance Discrimination	76
6.3	Which Structural Patterns Emerging from Instance Discrimination Benefit Linear Evaluation?	90
6.4	An interdisciplinary view on behavioral properties in decision-making algorithms	98
6.5	A dynamic neural field model of multimodal merging: application to the ventriloquist effect	150
6.6	Combining manifold learning and neural field dynamics for multimodal fusion	193
6.7	Suréchantillonnage Actif pour Modérer l'Apprentissage de Biais Visuels . . .	201
6.8	Algorithmic learning a next step for AI. An application to arithmetic operations	207
6.9	Novelty detection for unsupervised continual learning in image sequences .	221
6.10	Curriculum Vitae	229
6.10.1	Informations générales	229
6.10.2	Enseignement	230
6.10.3	Recherche	230
6.10.4	Médiation	235

Publications	236
Bibliography	242

1 Introduction

1.1 A brief history of AI

Artificial Intelligence (AI) is currently one of the most active areas in academic research and in industry. This domain offers a multitude of facets as it encompasses engineering aspects (to design relevant robotic parts or piece of software *e.g.*) but of course also scientific ones, with a methodology that depends on the study objectives. It has always been at the confluence of diverse research domains including computer science, mathematics, robotics, cognitive science, neuroscience, psychology, philosophy, etc. This interdisciplinarity and transdisciplinarity can also be found in my research works that are or have been related to computational neuroscience, psychophysics, developmental robotics and machine learning to name the more important ones. Throughout history, AI carried a lot of hope, fear and disappointment, leading to the alternation of multiple winters and springs, until the new spring boosted by deep learning in which we are today. But what are we really talking about?

The term 'artificial intelligence', first proposed at the Darmouth conference in 1956, is at least ill-defined and can be somehow considered an oxymoron, given that the only intelligent creatures known today are biological ones. If we consider each term separately, the term 'artificial' refers to something that has been created by humans (from Wiktionary). However, we, as a species, have a significant impact on nature (including on other animal and vegetal species through genetic selection *e.g.*), even at a geological scale as emphasised by the term Anthropocene, which designates our current era¹. Thus, precisely establishing a clear boundary between the natural and the artificial is a challenging if not an impossible task. On the other hand, despite significant progress in neuroscience, psychology and cognitive science, among others, the fundamental essence of intelligence in biological systems remains unknown regarding its nature, components or underlying processes. Moreover, there have been two major trends in AI regarding this quest of intelligence by trying to look for either human-like behaviours/principles or rational ones [181]. Without intending to delve more deeply into these debates here, I will rely on a broad and computer science-oriented definition of AI proposed by Minsky as "the construction of computer programs that engage in tasks that are currently more satisfactorily performed by human beings because they require high-level mental processes such as perceptual learning, memory organization and critical reasoning"², where I am mostly interested in computer programs with perceptual learning and memory organisation capabilities.

¹Formally the era has not been yet validated, especially as there is an ongoing debate concerning the precise starting date.

²It is noteworthy that this sentence can be interpreted as indicating that, once computer programs are able to perform a task as well as or better than humans, they are no longer considered to be part of artificial intelligence. In this sense, the definition of AI is always drifting on the frontier of our knowledge.

Looking back at the history of AI, we can find multiple eras and schools of thought. At the beginning, intelligence was considered to be more related to the ability to solve high-level cognitive tasks, which at that time seemed to be most difficult or most valuable ones as those mastered by humans but (may be) not by other animals. This resulted in significant progresses in logics and rule-based systems in the 1960s, knowledge-based systems in the 1970s, etc., with the emblematic example of Deep Blue beating Garry Kasparov at chess in 1997. In the 1980s, however, it quickly became apparent that these tasks were in fact quite easy to solve formally³, at least in terms of defining a methodology for their resolution, contrary to the ones that require sensorimotor skills to interact with the real world. This is known as the Moravec paradox [158] which is related to the symbol grounding problem, a fundamental limitation of cognitivism, as illustrated by the famous thinking experiment of the Chinese room, which discussed the difference between the ability to manipulate symbols and the ability to understand them [188]. In response, much research at the time demonstrated that simple forms of cognition could emerge without the need for symbols, such as the Braitenberg's vehicles [77] which exhibit avoidance behaviours based on purely reactive systems, or the subsumption architecture, which combines various reactive behaviours to generate more complex ones, as proposed by Brooks [79]. This has been formalised in several theories, such as the one of enaction [202], which emphasises the role of the body-environment interaction in shaping the cognition, or the concept of embodiment, which comes from cognitive psychology, focusing more on the role of the body in the process [166]. My research works are inspired by these theories and I will detail in the conclusive chapter 5 how my future works can be grounded more in these frameworks.

1.2 Neural networks

To also overcome some of the limitations of cognitivism, connectionism, a theory coming from the fields of psychology and modelling, proposes that cognitive processes emerge from the interaction of relatively simple connected units. One of the most well-known examples is that of neural networks, a field that encompasses a huge variety of models, ranging from spiking ones, *e.g.* with the Hodgkin-Huxley model [119], to rate-coding ones, with the artificial neuron proposed by McCulloch and Pitts⁴ [154] (see figure 1.1). I employ here the terminology coming from computational neuroscience, the field in which I did my PhD (see section 1.6.1). Spiking neurons compute and exchange continuous values (membrane potential or spike) at continuous time (in practice discretisation is often obtained with event programming), which can be seen as analogous to the functioning of biological neurons. On the contrary, rate-coding neurons compute and exchange continuous or discrete values at discrete (synchronised) times. In order to interpret them through the spectrum of biological neurons, we can consider that the exchanged values correspond to the number of spikes during the given discrete time period, *i.e.* the spiking rate, hence their name. This rate coding is one of the form of information encoding in the brain, although there are

³Although implementing them may lead to solutions that are intractable in a reasonable time frame, which also contributes to the decline of this kind of approach.

⁴This is one of the earliest works in AI (retrospectively attributed to AI since the field did not yet formally exist at the time) that attempted to establish a link between logic (interpreting the neuron's binary output as a proposition being either true or false) and simple brain physiology. Thus, interdisciplinarity, brain inspiration and neural networks that drive my research today were already present at (before) the beginning of AI.

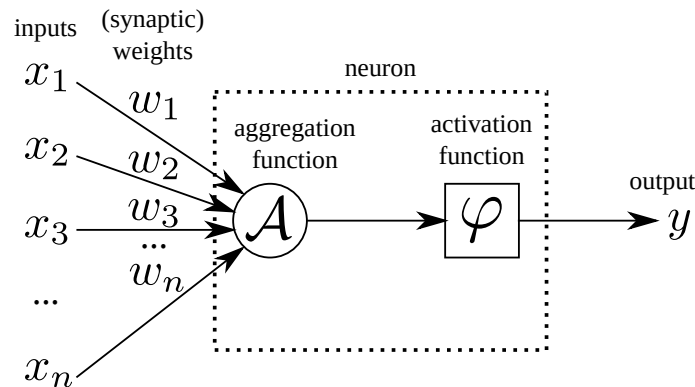


Figure 1.1: Artificial neuron model. The neuron receives a vector $\mathbf{x} = \{x_i\}_i$ of input values through a vector of weights $\mathbf{w} = \{w_i\}_i$, computes a simple mathematical function (usually aggregating the values through a weighted sum, that passes then through a non-linear activation function φ) that gives the neuron’s output $y = \varphi(\mathbf{w} \cdot \mathbf{x})$.

other forms of coding, such as temporal ones which can also be efficient [102]. Nowadays, in artificial intelligence, which is based more on mathematics and computer science than on neuroscience, the term ‘artificial neural networks’ mainly (or even exclusively) designates the use of the artificial neuron model of McCullochs and Pitts, and is sometimes even restricted to deep learning architectures. In this manuscript I will use the term ‘neural network’ (NN) to refer to any artificial model composed of basic and similar interconnected units that perform simple computations, and the term ‘artificial neural network’ (ANN) when the NN is composed of McCullochs and Pitts neurons.

The history of the field of NN is closely related to the question of learning (formally, ANN is a subfield of machine learning). D. Hebb was the first to propose a rule for (biological) neurons to update their weights, which consists of increasing (respectively decreasing) the strength of the connection between neurons that are (respectively are not) co-activated. This can be summarised as “cells that fire together wire together” [115]. Applied to the McCullochs and Pitts model⁵, it can be expressed as $\Delta \mathbf{w} = \eta \mathbf{x} y$ where $\Delta \mathbf{w}$ is the variation of the weights vector, \mathbf{x} is the input vector (or pre-synaptic activity in neuroscience), y is the output value (or post-synaptic activity) and η is the learning rate. A more general version of Hebbian learning, corresponding to $\Delta \mathbf{w} = \eta f(\mathbf{x})g(y)$ where f and g are any arbitrary function, encompasses almost all existing learning rules for ANN. Since then, multiple learning rules have been proposed often mixing aspects of neuroscience and machine learning. For instance, the Bienenstock Cooper and Munro rule [73], which I used during my PhD (see section 1.6.1), comes from neuroscience and was proposed to model the emergence of selectivity in the visual cortex. But it also corresponds to feature learning associated with the third and fourth orders of the probability distribution of the data. The self-organising map model proposed by Kohonen [133], which I used during my post doc (see section 1.6.1), is at the frontier of neuroscience and machine learning, as it models the self-organisation of sensory areas in the cortex, but is also an extension of the k-mean algorithm⁶. Rosenblatt proposed the perceptron model [179] as a fundamental principle

⁵After D. Hebb formalised his learning principle, similar mechanisms were observed in biological neurons. This is modelled by Spike Time Dependent Plasticity, in which the notion of co-activation is replaced by the temporal order of spikes.

⁶More precisely, the ANN equivalent of online k-mean is the neural gas [152], whose one derivative, the

of learning in the brain. However, consisting of a single layer of neurons, it is limited to linear regression/classification, which was illustrated by the well-known example of the XOR problem [156]⁷. The addition of more layers, a.k.a. multilayer perceptron, that have some resemblance to the architecture of the cerebellum, opened the way to obtain universal approximators. However, a learning mechanism is required to compute the gradient in all layers and neurons with a reasonable degree of precision and which can be automated to any number of layers and neurons, and to any differentiable neuron. This is the now well-known backpropagation algorithm [180]. It allowed the proposal of convolutional neural networks [143], whose convolutional neurons are inspired by the computational and organisational principles observed in the neurons of the visual cortex. At that time, however, these models were not particularly efficient compared to other approaches, and it took approximately 25 years for deep learning, formally defined as models with at least three layers (two hidden plus the output layers) and trained with backpropagation, to take centre stage. This was initially due to the introduction of new training techniques [117] but also to the availability of large datasets and computational resources. Although it has some roots in neuroscience, deep learning is now predominantly studied from a mathematical and computer science perspectives, especially as part of machine learning which I will introduce in the next section 1.3.

Most of my research works, and all those that I will detail in this manuscript, are based on artificial neural networks. During my PhD I used them because of their relative proximity to brain processing and learning, but also for their dynamical, robustness and emergent properties provided by decentralised computation. In chapter 3, I will present a model, with dynamic and decentralised perception properties, that helps to build bridges between neuroscience, psychophysics and AI considerations. In the chapters 2 and 4, I will use various deep learning architectures and improve their ability to learn relevant features. Thus I am interested in ANN due to their state-of-the-art learning performance, their properties related to connectionism, and their relationship with neuroscience and cognitive science, fields that I have used and will use as interdisciplinary sources of inspiration and collaboration in my studies (see the perspectives section 5.2).

1.3 Machine learning

As defined by its name, the field of machine learning is interested in the question of how to make systems that learn. Much of the progress of the last few decades has been due to the definition of benchmarks, which were conceived as representative of some typologies of problems, on which the various algorithms can compete. Modern machine learning can thus theoretically be conceived as the science of benchmarks [113]. The drawback is that the objectives and features learned are focused on the content of the datasets⁸, which cannot be representative of all tasks⁹, under all conditions, especially if we expect some degree of autonomy from the system. There are indeed works on few shot learning or transfer learning for instance to study the generalisation properties, but this is still on

growing neural gas, will be used in section 3.4.

⁷This is one of the reasons for the AI winter that occurred in the 1970s, as much hope was based on ANN at that time already.

⁸Although, some generalisation can be expected as recent research show that the ranking obtained on one dataset, transfers to others, at least for visual classification [182].

⁹All tasks is not to be read in the mathematical sense, as the no free lunch theorem states it to be impossible, but in the practical sense of all interesting tasks for the system that is currently proposed.

datasets, thus following precise protocols. Consequently, the benchmarks have favoured a scientific methodology in machine learning but at the cost of a slight drift from the original goal of using learning to obtain human-like intelligence, as illustrated by Turing, one of the pioneers of AI, who suggested in 1950 that “instead of trying to produce a program to simulate the adult mind, why not rather try to produce one which simulates the child’s? If this were then subjected to an appropriate course of education one would obtain the adult brain.” While most of my research works are part of machine learning and thus rely on benchmarks, I am also interested in giving systems more autonomy (see section 1.5), and one way to do this may be to take some inspiration back from cognitive science, as I will propose in the perspectives section 5.2.

I will hereafter introduce the main types of learning, that can be considered as different types of benchmarking protocols, in machine learning. Obviously, multiple types of learning can be combined, for instance by using unsupervised learning as pre-training before supervised fine tuning, which is classical in image classification, or reinforcement learning smoothing as for example in ChatGPT.

Supervised learning In supervised learning each data in the dataset is labelled, meaning that we have pairs $(\mathbf{x}_i, y_i)_i$ with $\mathbf{x}_i \in \mathbb{R}^n$ an input vector (when having tabular data, formally $\mathbf{x}_i \in \mathbb{N}^n$) and y_i the expected output¹⁰. Thus, the goal can be expressed as learning the function f that relates the inputs to the outputs: $y_i = f(\mathbf{x}_i)$. If y_i is discrete, the problem is called classification, otherwise if y_i is continuous, it is called regression. The usual way to tackle this kind of problem is to define a loss function (which measures how far the output of the model is from the target) that is optimised, usually in combination with some regularisation techniques to avoid overfitting and favour the parsimony principle. Supervised learning is the most intuitive and has been the most widely used in ANN renewal. However, it is limited by the constraint of having to label each data individually, which can contain errors and is very time consuming, but also impractical when trying to have autonomous systems. Consequently, my works do not specifically focus on supervised learning. Nevertheless, in the section 4.2 I will present contributions to active learning, i.e. choosing the right labelled examples in the dataset, applied to a classification problem with biased data, or to improve the multiplication learning with a multi-task approach.

Unsupervised learning Here, data are not labelled, and the dataset is reduced to $(\mathbf{x}_i)_i$. Since we cannot rely on an expected output, the objective is much harder to define and several different kind of approaches have been proposed.

One way can be to group similar inputs into clusters to obtain a clustering. The objective can then be for instance to maximise intra-cluster similarity while minimising inter-cluster similarity, which can be formalised as a cost function to be optimised. Some methods, such as the self-organising map [133] can also perform clustering without optimising an energy function at the global level, relying instead on local optimisation of each neuron [94]. I will present in section 4.3 a contribution to clustering, more precisely to class-incremental learning, which means that each class is presented successively and can also reappear in the inputs afterwards.

Another way is to try to define some kind of supervision signal from the data themselves, to define a pretext task. For this reason these approaches are often called self-supervised

¹⁰When considering multi-label classification or multi-dimensional regression, the expected output is a vector, but these cases are often split into several one-dimensional problems, while sharing features.

learning. One approach is to learn the density function of the input distribution, which can be formalised as a regression problem. Other approaches perturb the input and try to reconstruct the original in some way, so they can be called generative methods. This is the case for instance with denoising auto-encoders¹¹ where the original input has to be generated from a noisy version of it, or with diffusion models, where the input has to be reconstructed from a very noisy version of it¹². From modified inputs, some works also derive a classification task, where the aim is to find a disturbed version of an input from another noisy version of it, corresponding to visual discrimination tasks. Finally, when relying on multimodal inputs (*e.g.* multispectral data, language translation, etc.), it is easy to use one signal to supervise the others¹³. This is a very active field that proposes a huge variety of approaches, as it allows the use of large unlabelled datasets, and in machine learning size matters to achieve good performance [68]. Most of my research work concerns unsupervised machine learning and I will present in chapter 2 our contributions to self-supervised learning of visual inputs.

Reinforcement learning The framework of reinforcement learning is an agent interacting with its environment. More precisely, the agent has a set of actions that it can perform. This will induce a change in the state of the environment (via a transition function), which in return will give some observation and a reward (most of the time not really informative, as most of the reward quantity is given only when the task is completed) to the agent¹⁴. The goal of the agent is then to maximise its cumulative reward (with a discount factor) over its policy, *i.e.* the choice of action it has made. The focus and challenge of reinforcement learning is therefore this temporal decision making and the distribution of the reward obtained over the individual actions constituting the sequence. There are two main ways of approaching this problem.

In model-based approaches, the system tries to learn the transition model of the environment, *i.e.* a world model. This is very challenging, especially if the environment is complex or has a large dimension. The main advantage is that when this learning is done, the agent can be very adaptive, as the policy can easily be adapted to a change in the reward function, *i.e.* the task to be accomplished, for instance by using methods from operations research.

In model-free approaches, the system learns directly what action to perform in each situation. This is a simpler problem, that is why most of the works that has been applied to real-world problems has used this technique initially, but only to solve a specific task.

While my general research interest in autonomous agents (see section 1.5 for more details) is close to the reinforcement learning framework, I have not used it in my research yet¹⁵

¹¹Technically, with classical auto-encoders, the input is intact and the same as the output, but it is the encoding process that is noisy in some way.

¹²Formally the aim is to learn the inverse process of the one which gradually transforms the input into quite random noise by adding noise little by little.

¹³This type of work may also be qualified as supervised learning, depending on whether the modalities are considered as inputs (for the global system) or outputs (for supervising another modality).

¹⁴This can be formalised within the framework of Markov Decision Process (when the “observation” is directly the state of the environment) or of Partially Observable Markov Decision Process (when the state is only partially observed). In this framework, the Markov hypothesis is made, meaning that everything at time t is determined only by what happened at time $t - 1$. In practice this is usually not the case, but people can either neglect the influence of time before $t - 1$, or include it in the state, meaning that time $t - 1$ represents everything (relevant) that happened up to $t - 1$.

¹⁵Except for my master’s thesis on distributed solving of stochastic games, an extension of Markov decision process to multi-agent systems [56] which I will not detail in this manuscript.

as my primary interest is in unsupervised representation learning (see section 1.4) but I will include it in my future works as a framework for studying action decisions to improve learning and perception (see the perspectives section 5.2).

The trend since the renewal of deep learning, which represents now the overwhelming part of machine learning, has been to go with less and less supervision. The first progress were in supervised learning (*e.g.* for image recognition), then reinforcement learning (*e.g.* for go playing) then unsupervised learning, especially self-supervised approaches (*e.g.* for natural language processing). By unlocking the use of large datasets from the Internet, a lot of research has focused on defining foundation models in several domains. These models offer general representation that can serve as a basis for solving multiple tasks within a domain. Some works also try to extend this generality across domains [177]. Since the beginning of my research I have been interested in generic unsupervised learning, but rather than focusing on large models, I am studying how to structure representations learning (see section 1.4) to help their generalisation and use in the context of autonomous systems (see section 1.5).

1.4 Representation learning

Representation learning is a subfield of machine learning that is interested in learning low-dimensional embeddings of the inputs. While much past research has been dedicated to the study of hand-crafted features with good properties, such as Scale-Invariant Feature Transform in computer vision for instance, to be used with relatively simple machine learning techniques, since the renewal of deep learning the representations are automatically extracted from a representative dataset of the problem. Thus, the whole field of deep learning can be considered to be included in representation learning. But “[w]hat is a good representation? Many answers are possible, and this remains a question to be further explored in future research. [...] [I]n general, a good representation is one that makes further learning tasks easy.” [71] Thus, this is a large domain with a non-exhaustive list of objectives whose some cannot be clearly quantified. Let us analyse the learning of representations along two axes: whether they are local or global, and whether they are organised or not.

Local vs global Some representations are local, i.e. they are activated only for a small (often contiguous) part of the input space. The extension to the whole space is then obtained by accumulating of these local representations. Some models learn a set of prototypes, each one corresponding to a vector in the input space, which are distributed over the input space. These prototypes can be used for clustering, where each cluster is the set of inputs closer to one prototype than the others, or for regression by considering any function of the distance between the input and the prototype, as I studied during my post doc [16]. Other models rely on multi-dimensional Gaussians¹⁶, which have a support depending on their covariance matrices. They can then be combined, as in the Gaussian Mixture Models (GMM), that we used in section 4.3, to form the basis of linear regression, as in Radial Basis Function Networks, or to define the support of local linear models, as in Locally Weighted Projection Regression. The main advantage of these local methods

¹⁶Any local function can be used, but because of its simplicity and good mathematical properties, Gaussians are mainly used.

is their robustness to errors in the data, including non-stationary setups, as an outlier will only affect a limited part of the model and not the whole one. Combined with their relative simplicity, which helps to deal with limited-size datasets, these approaches have been and to some extent still are popular in robotics. The drawback is that they do not scale well as the number of local models needed to split a space grows exponentially with the dimension of that space.

On the other hand¹⁷, global methods rely on representations that extend over the whole space, and are called features in this case. The most popular methods currently are deep neural networks, which originally used sigmoid functions, now mainly Rectifier Linear Units, as activation functions. The main advantage of these features is that they generalize very well in high dimensional space.

Overall, most (regression) algorithms can be viewed as a combination of linear and non-linear functions. Under this unified view proposed in [195], models differ in the choice of the non-linear functions, which parameters are learned and with which learning method. Thus, the local/global aspect of the representation is related to the choice of the non-linear function which can have a finite or infinite support. It should be noted that the distinction between local and global representations can sometimes be presented through the spectrum of the activity distribution. Indeed, to identify a (local) region in the space, only one local representation is needed, whereas multiple features are required. However, multiple local representations can be used simultaneously, *e.g.* with the k-Nearest Neighbour (k-NN) algorithm, and it is common to use a sparsity regularisation term when learning features to limit the number of features to be used simultaneously, which makes this distinction on the distribution fuzzy.

Organisation The learned representations, independently of being local or global, can or cannot contain some kind of organisation, often reflecting some underlying structure of the inputs. This is somehow related to manifold learning that hypothesises that the set of data in the input space lies on a manifold [83]. Mathematically, a manifold is a topological space (of some dimension n) that is locally Euclidean. In machine learning, it is used in a looser sense, as the manifold dimension can vary locally (formally it is a set of sub-manifolds) and data are not strictly on the manifold, but close to it.

At one end of the spectrum, are algorithms such as Principal Component Analysis, where the learned representations are simple (linear), but the data distribution is preserved at best (in terms of the second order of the distribution). At the other end, there is for example MultiLayer Perceptron (MLP), where the learned features are highly non-linear and dedicated to the task but do not try to enforce any kind of structure within the layers. There are a lot of models in between. Some are more dedicated to data visualisation, trying to have some kind of global structure, such as the Self-Organising Maps (SOM), which projects the input space onto a topologically organised predefined grid of prototypes, or the t-distributed Stochastic Neighbour Embedding (t-SNE), which projects the input space non-linearly onto a low-dimensional feature space, where the distance matrix tries to match the one in the input space. Among the models mostly dedicated to feature learning, some use a local structure such as Variational Auto-Encoders, which, by considering each representation as a Gaussian distribution, favour inputs that share similar features to be embedded closely, or search for a global structure, for instance Wasserstein

¹⁷There are also methods that are in between, such as decision trees, as the decision boundaries are not local, but do not extend infinitely in all directions (except the first one).

auto-encoders, where the distribution of the representation should follow an *a priori* distribution. Sometimes the structure can also emerge without being induced in any way, as it was the case with word embedding where we can find some kind of linear semantic algebra in the feature space [155].

Let us go back to the question of what makes a good representation. In supervised learning, the representations are dedicated to solving a labelled task, so their quality can be directly related to the performance, and the structure of the features is usually not really a concern. In unsupervised learning, however, the evaluation criteria of the representations must be determined. This is often related to a downstream task such as generation, classification, few shot or transfer learning to name a few. Due to that diversity, the expected properties that the representations should exhibit during their unsupervised training is still an open question. The term representation learning is often used to designate this more specific area of research, as the object of study is (the properties of) the representation. It is this terminology that I will use in this manuscript. In this case, obtaining some kind of structure in the representation may be a desirable property. It may help to generalise better (*e.g.* with the analogy in word embedding), to have better adaptability of the systems (especially if we want to combine different models together that may have some requirements on the structure of the data), to have better explainability (if the structure can be understood or used by a human), or to use less computational resources (especially when considering embedded systems where the number and size of the connection lines are a bottleneck¹⁸). In a more global view, these properties can therefore improve the autonomy of systems (see next section 1.5).

Most of my works that I will present in this manuscript and my research project (see section 5.2) are studying the impact of the structure of the representations on the properties and performances of the system. This can be through a (complementary) objective for unsupervised representation learning to improve performance (see chapter 3), to achieve multisensory fusion and perception (see chapter 3) or to detect and recognise classes in a stream (see chapter 4).

1.5 Autonomous agent

Most of my research work can be related to some extent to the global framework of an agent interacting with its environment (see figure 1.2). It is usually used in robotics to illustrate the three main components of a robot: sensors, a cognitive system and actuators. However, my interests are not in physically building such agents, although I have regular interactions with the robotics community and have participated in two projects on human-robot interaction. Such a scheme is also often used in reinforcement learning to illustrate the dynamics of interaction. However, I have not studied the sequential choice of actions, except for a contribution during my post doc [17], yet I will use reinforcement learning with internal reward in my future research (see the perspectives section 5.2). Therefore, this framework needs to be considered in a broader sense, where the following elements are important:

- The agent refers to any system that can be delimited in some way from its environment. In terms of more theoretical considerations, we can define an agent as an

¹⁸This kind of constraint is a hypothesis to explain self-organisation in the brain, as the connections pattern of neurons is driven by molecules that necessarily diffuse locally.

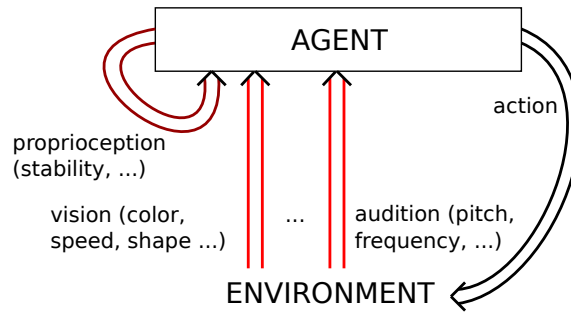


Figure 1.2: The general framework of my research work is an autonomous agent that receives a flow of data through various modalities and interacts with its environment (scheme taken from my PhD defence). I am particularly interested in the perceptual process and how the agent can learn to make sense of its environment.

auto-poetic system that maintains its constitutive autonomy [202]. However, the proper boundary of the agent may be smooth for example regarding the extension of the peri-personal space when using tools [151] or considering phenomena as stigmergy. I will not go into these subjects in more detail as they are beyond the scope of my current research. My focus is more on the autonomy of the agent. This is not to be read in a philosophical sense, i.e. I am not referring to any form of agentivity, subjectivity or proto-consciousness, but from a computer science perspective. This is related to different keywords depending on the field of research, such as lifelong, open-ended or developmental learning in robotics or unsupervised, continual, stream and meta-learning in machine learning. Especially, I am interested in the ability of the agent to make sense of the data coming from its sensors on its own through unsupervised learning (see chapter 2)

- The environment is everything that is not the considered agent and that can evolve according to its own rules or as a consequence of the agent's actions. Thus, it can also include other agents (artificial or human) when considering multi-agent systems, which has been the case in some of my works *e.g.* [13].
- The agent gets observations from various modalities. They can come from internal sensors (as for the proprioception) or external ones (such as vision or audition). I rather prefer not to use the term 'sense', unless there is no ambiguity, as it may implicitly refer to the five senses in human beings¹⁹. I also distinguish sensation, as the raw data arriving in the agent, from perception, as the result of the internal processing of sensations, as defended in my PhD [55]. Thus, perception can be related to the assimilation process in the constructivist approach [167]. Moreover, various kinds of information can come from a single sensor, such as colour, speed or shape from a camera. In my research, therefore, I use the term modality in a broad sense that can refer to any consistent subset of a data flow coming from a sensor²⁰. It can also refer to information related to the action performed, as in my

¹⁹By the way, human beings have sensors for more than their five senses and what really defines a sense in human subjectivity is not yet clearly identified. It may be related to the structure of the information received and the way it changes when humans move [161].

²⁰This is aligned *e.g.* with the Cambridge Dictionary's definition of modality as "a particular way of doing or experiencing something".

post doc [21]. I am particularly interested in the process of merging various kind of sensations to obtain a unified multimodal perception (see chapter 3).

- The agent can act on the environment. Without an explicit task, this can be to get more information from the environment to desambiguate the current perception (active perception) or to explore it to improve its world model (active learning). These active processes can also be closely intertwined with the learning of representation (and thus the perceptual process) as we discussed in [30] within an information-theoretic framework. Moreover, the action will influence the distribution of data received by the agent raising questions such as incremental learning (see chapter 4). I will further explore these links between action, learning and perception in relation to the sensorimotor theories in my future works described in the perspectives section 5.2.

1.6 Summary of my research work

My research focuses mainly on how an autonomous system can make sense of the multimodal data flow it receives, possibly by controlling it with actions, targeting at most genericity and general principles. I am particularly interested in unsupervised representation learning, studying what information should be learned, how to organise them and how to use them. In this section, I will first summarise the research work I did during my PhD in the Cortex team at Loria and my post doc in the Flowers team at Ensta Paristech (see section 1.6.1). Then, I will present the PhD thesis that I co-supervised as an associate professor in the SyCoSMA²¹ team at LIRIS, starting with those that are at the edge of my research interests (see section 1.6.2.1), then those that I have chosen to detail in this manuscript (see section 1.6.2.2) to illustrate the three main facets of my research interests: self-supervised learning (chapter 2), multimodal perception (chapter 3), and active and incremental learning (chapter 4).

1.6.1 PhD and post doc

PhD thesis The main focus of my PhD thesis was to take inspiration from the brain processing and learning to propose a model of multimodal learning and perception. In particular, I considered the cortex at a mesoscopic scale, i.e. at the level of neuronal populations, to propose the Self-Organising Maps for Multimodal Association (SOMMA) architecture [23] (see figure 1.3). The model is composed of generic maps, which are an analogue of cortical areas, composed of layered units inspired by cortical columns observed in the cortex. Each unit learns to discriminate a feature of its unimodal input flow via its sensitive layer. This is achieved using BCM [73], a cortically inspired learning rule that I have modified to be sensitive to some feedback signal [47] that can change over time [25] with some guarantees on its dynamical properties. Each unit, via its cortical layer, integrates information from other maps and thus modalities, whose information are forced to be aligned [24, 45] by modulating the learning rule, a mechanism I proposed for learning multimodal features. Each unit, via its perceptual layer, participates in a decentralised competition, using Dynamic Neural Fields (a brain-inspired model that I will detail in chapter 3), whose spatial consistency of activity is used as a feedback signal

²¹which stands for Cognitive Systems and Multi-Agent Systems, and was originally named SMA when I was recruited even though the cognitive systems research axis already existed.

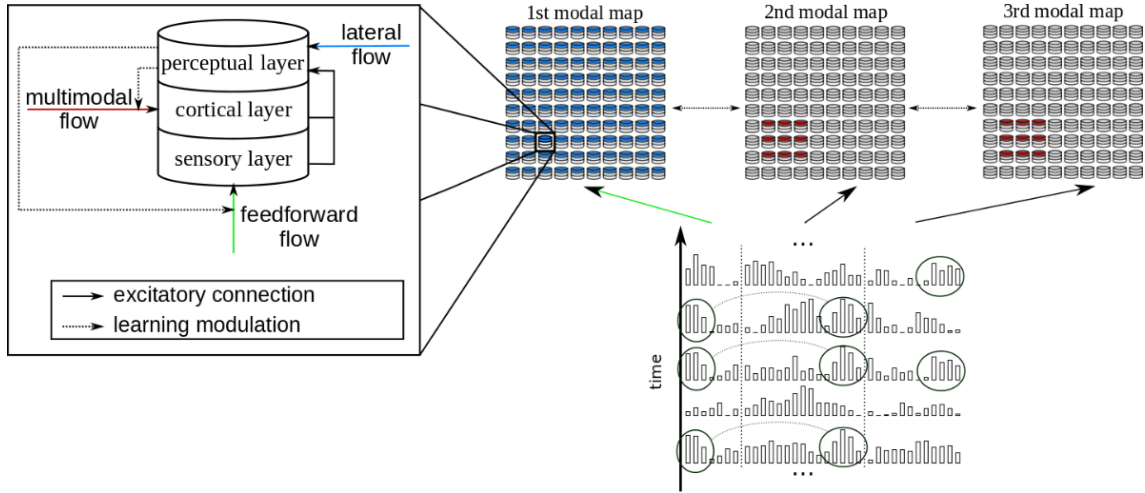


Figure 1.3: In SOMMA, each modality is processed by a dedicated map, composed of generic layered learning units. Overall the model learns multimodal features within self-organisations aligned between modalities, which are the support of multimodal perception.

so that to obtain the self-organisation of the map, i.e. that nearby units will be selective to close features [26, 46]. It also provides multimodal perception, by dynamically merging relevant features based on their spatial localisation, whose qualitative properties are similar to those of humans [55].

Post doc During my post doc, I studied a similar architecture as in my PhD, except that the maps used the classical model of Kohonen self-organising maps [133] and that the multimodal relation was obtained by learning to project from one map to another and vice versa instead of imposing the alignment between the self-organisations (see figure 1.4). My main focus of interest was in the predictability module, which measures the ability of one representation (in this case a prototype) to predict those in the other modalities. This quantity is used to drive the learning of representations by favouring those with high predictability scores. This general principle can be used to improve the representation in the case of supervised learning [20, 21], using the target as one of the modalities, and to be more robust to noisy labels [18], *e.g.* when having only part of the dataset that can be predicted or when having multiple camera inputs [22]. The discrete derivative of the predictability measure can also serve as a drive for active learning [17], with a mechanism similar to artificial curiosity to guide the system towards areas where there is high learning progress, while learning to map the input space.

1.6.2 PhD supervisions

1.6.2.1 Those at the edge of my research project

Multi-agent systems for load shedding I co-supervised with Salima Hassas, professor in the SyCoSMA team, the PhD of Victor Lequay between 2016 and 2019. This thesis was done in collaboration with the company Ubiant, specialised in smart home systems, in particular for electrical energy management. The subject was decentralised load shedding (i.e. the ability of consumers to reduce their electricity consumption globally to meet a certain load for a period of time defined in advance by the producers) within a smart grid

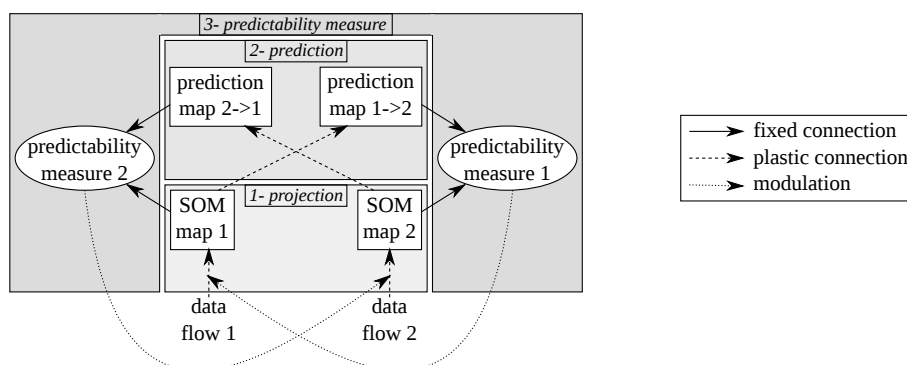


Figure 1.4: The PROjection PREDiction architecture. Each modality is projected onto a self-organising map, while projection modules learn the multimodal mapping. The predictability measure favours the learning of features that can predict the other modality.

(i.e. local electrical networks mixing producers and consumers). The proposed model is based on a multi-agent system, using gossip methods that allow to obtain overall indicators such as mean or max over the population in a decentralised way while preserving the data privacy of each agent. We proposed that each agent engages a reduction of its consumption depending on its maximum capacity, which can be known from Ubiant’s smart home systems, and its reliability, which is computed taking into account some indicators such as its ability to participate in previous events. Then, during the load shedding, each agent detects, by accumulating evidence, the difference between the expected load and the current one, and, if necessary, dynamically adjusts its consumption proportional to its participation with respect to the global one [13, 37, 44].

Recommanders system for adaptive learning I co-supervised from 2020 to 2023, with Nathalie Guin and Marie Lefevre of the LIRIS, the PhD of Anaëlle Badier, in collaboration with Nomad Education, a company that proposes a mobile application for para-scholar education. In this context, we proposed a recommender system of quizzes based on several components [3]. First, the content is filtered based on the student’s level, which is automatically evaluated by statistical methods based on their previous results to other quizzes. We then choose to propose contents that correspond to predefined strategies than can be of remediation, continuation or deepening relying on a tree that links the contents through their prerequisite and expected notions. Secondly, the contents are ranked according to different indicators: the pedagogical relevance based on the notions shared with the current quiz, the history in order not to recommend a quiz too soon and the novelty to stimulate the student’s learning. This system was tested in real conditions of use with thousands of users and we show that it significantly improves the time passed on the application and the number of quizz realised [4, 35, 39]. All this work was done using an iterative framework to progressively adapt the system to test new hypotheses raised by the continual analysis of the student’s behaviours logged within the application.

Human-robot interaction As they are both in the context of human-robot interaction, I will regroup here the work done during two projects. First, the work done by Laurianne Charrier during her master’s thesis and Alexandre Galdeano during his PhD thesis from 2017 to 2020, both of which I supervised with Amélie Cordier and Salima Hassas from the

SyCoSMA team, within the Behaviors.ai ANR labcom project with Hoomano, a company that developed software for social robots. Secondly, the work of Lucien Tisserand (post doc) in the PepperMint labex project, in collaboration with Heike Baldauf-Quilliatre from ICAR and Salima Hassas and Frédéric Armetta from the SyCoSMA team. The Behaviors.ai project proposed to use developmental learning to adapt the behaviour of social robots to the user [33]. We mainly worked on how robot’s empathy can be perceived by humans [32] and proposed an evaluation protocol to quantify this aspect [11]. In the PepperMint project, which is still ongoing, we have focused more on the verbal interaction between human and robot, in collaboration with conversation analysts to study more specifically the emergence of the interaction sequence [1, 29].

Explainability with medical data Since 2022, I am co-supervising, with Michaël Sdika and Nicolas Duchateau from the Creatis laboratory, in collaboration with Previa Medical, a company proposing an IT solution for the prevention of life-threatening emergencies in hospitals, the PhD thesis of Pierre-Elliott Thiboud. The aim is to propose a deep learning solution for the detection of sepsis, from tabular data, that can provide some level of explanation for the classification, which is a key issue in this field. We are currently evaluating derivatives of state-of-the-art classification algorithms in order to extend them, especially by imposing some constraints within the architecture and the loss function in order to gain more insight and control over the features used for detection.

1.6.2.2 Those at the core of my research project

In the following of the manuscript, I will present several contributions to the three main axes of my research work:

- **Self-supervised visual representation learning** (chapter 2). This work is part of the Master’s theses of Nawel Medjkoune and Valentin Chaffraix, which I co-supervised with Frédéric Armetta and Stefan Duffner from LIRIS, and of the Master thesis then PhD thesis of Alexandre Devillers, which I supervised²² from 2021 to 2024. We were interested in learning visual representation in an autonomous way, thus relying on self-supervised methods. More specifically we studied whether low-level descriptors are sufficient to automatically extract semantically related examples from videos to learn interesting representations (section 2.2.1.1) and whether action can help to learn and structure the representation from visual glimpses (section 2.2.1.2). Based on these initial works, we focused on the properties related to the information that the representations should contain in order to improve the performances of the state-of-the-art models 2.2.2. Finally, the section 2.3 consists in a systematic study of the structure of representations that are effectively learned by discriminative approaches.
- **Multimodal perception** (chapter 3). This axis is related to my AMPLIFIER regional project, with Salima Hassas, Marie Avillac from the CRNL laboratory, Alan Chauvin, Nathalie Guyader and Jean-Charles Quinton from the Univ. Grenoble Alpes. This was an interdisciplinary project, bringing together people from psychophysics, statistics and computer science, on the question of how an agent (biological or artificial) can decide which piece of information, coming from various modalities, to merge and how to weight them. The work I will detail was done by Simon Forest

²²Administratively speaking it was a co-supervision with Salima Hassas, until her death in 2023.

during his PhD, which I co-supervised with Salima Hassas and Jean-Charles Quinton between 2018 and 2022. The problem of multimodal merging can be seen as a decision problem between the different cues coming from the various modalities. In section 3.2 I will present a framework for analysis the links between different models of decision-making used in neuroscience, psychophysics and robotics. I will then show in section 3.3 how one of these models, coming from neuroscience, can model psychophysical data of humans performing a pointing task with audio-visual stimuli, using underlying topologies for fusion. Finally, I will illustrate in section 3.4 how this model can be coupled with manifold learning for an artificial agent to weight modalities depending on the density of representations, in a way qualitatively similar to humans.

- **Incremental and active learning** (chapter 4). Here I will explore some temporal aspects of learning that an interacting agent would experiment, either by controlling the inputs via active learning mechanisms, or by dealing with non-stationary datasets. In section 4.2.1 I will show how active learning can improve classification performance, especially by selecting relevant examples in the context of a biased dataset, a work done during Alexandre Devillers' PhD, while in section 4.2.2 I will illustrate how it can help learning arithmetic operations with a deep learning model, a work done by Anthony Baccuet during his Master's thesis supervised by Frédéric Armetta. Finally, in section 4.3, I will present some work when the distribution of the data stream is not stationary, especially in the context of unsupervised class-incremental learning, where we have studied how to detect the arrival of new classes, or previous ones, in order to progressively update and structure the representations. This was the research question of Ruiqi Dai's PhD thesis from 2018 to 2022, which I co-supervised with Stefan Duffner, Frédéric Armetta, from the LIRIS laboratory, and Mathieu Guillermin from the university of UCLy.

2 Self-supervised visual representation learning

2.1 Introduction

In this chapter, we are interested in unsupervised learning of representations from visual inputs, for object classification, which can be a relevant capability for an autonomous agent to perceive its environment. There are many other subfields in computer vision such as object localisation, tracking, etc., with their own specificities and goals, but they often require content recognition at some point. Historically, feature engineering was the usual way to build image representations. Although they were quite generic with good mathematical properties, they often lacked the ability to achieve very high classification performance on challenging datasets or environments. Then, deep learning, by automating this representation learning process, provides a big leap in performance initially with supervised approaches. The drawback was that the obtained features were more dedicated to the dataset they were trained on, yet transfer learning or fine tuning can be applied. Recent works have focused on Self-Supervised Learning (SSL), i.e. determining a supervising signal from data themselves thanks to the definition of a pretext task. It has the advantage that the learned representations are less biased toward a specific goal and thus tends to be more generic. Moreover, as it does not require human labelling, the models can use plentiful of raw data, which is particularly useful for domains lacking annotations, which can increase the final performance. There are two main approaches in SSL using deep learning applied to images [126].

- Generative approaches propose to reconstruct part or all of the input. For example, it can consist in masking a part of the input and predicting it from the rest, or in finding back the colouring of the image. When the input is also used as the target, with the auto-encoder models, the function to be learned is the identity one which is a somewhat ill-posed problem. To prevent this, multiple tricks have been proposed, such as noising the input or the process in some way, or constraining the computation of the model by introducing a bottleneck. One of the main difficulties of these methods is the definition of a proper loss function that correctly quantifies the quality of the reconstruction. The Generative Adversarial Network approach [108] and its derivatives tackle this problem by delegating this evaluation to a network that learns to discriminate between real and fake examples. Overall, these kind of approaches are efficient for generating new content and are widely used in generative AI.
- Discriminative approaches use a classification task as a pretext. This can be, for instance, determining the relative positions of two patches from the same image,

or ordering the patches in a jigsaw puzzle created from the image. For efficient representation learning in image classification, the best methods today are instance discrimination ones where each image in a batch is slightly modified twice (*e.g.* with blurring, colour change, etc.) to create related pairs whose representations by the network has to be similar. Current research questions mainly concern the creation of the pairs, especially as it depends on the domain, and the kind of (in)sensitivity of the representations to the image augmentations used to create the pairs.

In the absence of an explicit objective, the main challenges in self-supervised learning of visual representation are then to determine the right type and definition of the pretext task and the relevant inputs to the model, as this will induce the properties and structure of the representations. Especially, we want them to be useful for the downstream tasks, usually object classification, while not being too specific in order to preserve some genericity to be easily transferable to the different tasks and datasets that may require different types of features at multiple scales. In this chapter, I will summarise some of our contributions to this line of research. First, we will discuss how an autonomous agent can select relevant related examples for a visual stream (section 2.2.1.1). Then, in section 2.2.1.2, we will study how the action can shape the learned representations in line with sensorimotor theories [161]. Based on these results, we proposed a module, relying on an equivariant structure, that can be added to the state-of-the-art visual discrimination instance models to improve classification performance (section 2.2.2). Finally, section 2.3 is dedicated to a study of the properties of the representations that emerge from instance discrimination methods, especially focusing on the evolution of their structure along the path from the pretext task to the downstream one.

2.2 Learning visual representations

2.2.1 Preliminary works

2.2.1.1 Object representation learning from video

Context While visual instance discrimination methods have been around for a long time (see *e.g.* the siamese neural network [78]), they became an active area of research in 2020 with the paper by Chen and colleagues [86]. Our work, done in 2017, focused on how to create pairs of related examples, at a time when this question was more open. This was applied to the detection and recognition of objects in a video stream. While there exists multiple models for this problem *e.g.* [76, 142] these methods are designed and pre-trained for specific object categories in a dedicated environment and are not able to accommodate new object categories during operation. Other approaches, *e.g.* [148], consist in continuously updating the learned visual representations using recognised objects from videos. However, their method incorporates a considerable amount of prior knowledge by using a Convolutional Neural Network (CNN), pre-trained on a large labelled dataset. On the contrary, we focused on the context of an autonomous agent, and thus we want to rely only on purely unsupervised learning, without any supervised pre-training whatsoever. Thus, we proposed an approach to unsupervised object learning in videos by enforcing a similar representation for objects that are considered related, using saliency-based detection combined with spatio-temporal continuity.

Model and results Our proposed approach is illustrated in the figure 2.1. From the video stream, we detect areas of interest that differ from their surroundings in colour, intensity, texture, orientation, depth, and other simple features using saliency-based detection inspired by the human visual system [124] to define proto-objects. We then rely on the spatio-temporal coherence principle, i.e. that detections that are close in space and time are likely to correspond to the same object, in order to provide a weakly supervised signal. We therefore create tubelets of patches, each representing one potential object. Based on these tubelets, positive and negative pairs of patches are created by associating respectively two patches of similar objects, i.e. within a tubelet, and two patches of dissimilar objects, i.e. between different tubelets. Finally, we used a siamese neural network to learn the representations. This model brings similar data points closer together and pulls apart dissimilar data points in feature space by minimising a contrastive loss function with a margin.

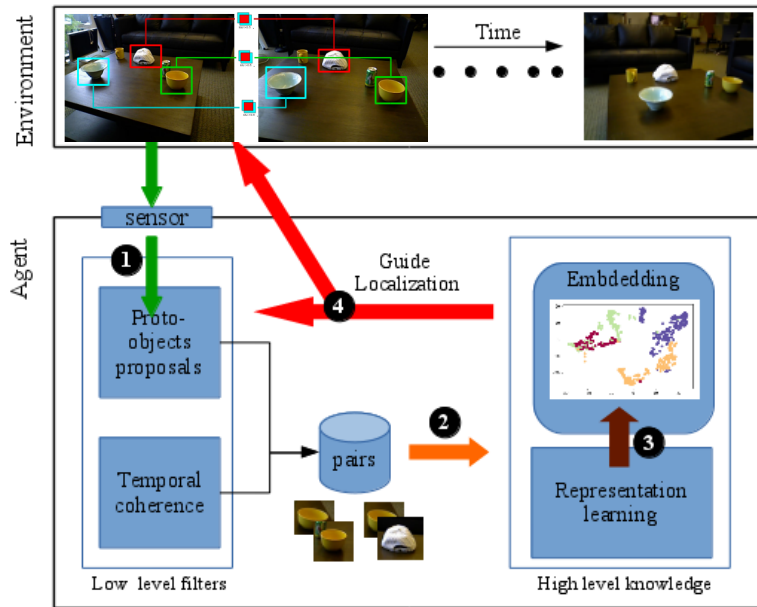


Figure 2.1: Overview of the proposed approach. The agent captures the visual field frame by frame and processes it to discover proto-objects (arrow 1). Using spatio-temporal coherence, pairs of similar and dissimilar objects are created and fed to the siamese network (arrow 2) which learns an internal visual representation (arrow 3). This representation may be used to guide object localisation in next steps (arrow 4) in a bootstrap process. Note that arrow 4 has never been properly studied and is an open perspective of this work.

We evaluate the proposed approach on the RGB-D Scenes Dataset Version 2 [140]. The visualisation of the embeddings indicates that our model is able to cluster between the different objects seen. This was to some extent confirmed by a comparison with a similar CNN trained in a supervised way, where our model performed almost similarly with 2 objects, but performance dropped as the number of objects increased. This was a simple preliminary network, nevertheless it confirmed the potential of instance discrimination networks, which we will use in section 2.2.2 with state-of-the-art models. We also planned to improve it by using a bootstrapping mechanism that would use the learned representation

to improve the proto-object detection that feeds the network, thus improving its learning and so on. Unfortunately this has not been tested yet, but a similar idea appears to be efficient in a recent work with more state-of-the-art architectures [203].

This work was part of Nawel Medjkoune’s master’s thesis and led to a publication in a workshop [49].

2.2.1.2 Impact of action in the representations

Context and objectives Sensorimotor theories are based on substantial evidence from neuroscience, developmental psychology and cognitive science, among others. The main claim is that actions, and more especially the sensory changes induced by motor actions, play a key role in learning a predictive model of the world and in perceiving it [161] (I will say more about this in the perspectives section 5.2). On the contrary, even if some self-supervised approaches are action-related, the action itself is usually not used in the model. To introduce action into image processing, we consider a sequence of glimpses that provide only a sub-part of the image at a time. Such models that process only glimpses of images were originally introduced for computational advantage, but they also open up the possibility of making models that actively perceive the world by choosing where to look. This idea of processing glimpses of an image has been applied to classification in two ways: either by dedicating a neural network to each glimpse w.r.t. its temporal index [173], or by letting a recurrent network learn to perform saccades in a reinforcement learning environment [157]. In this work we have focused on studying the influence of action consideration in the learning of visual representations in deep neural network models, using a simple model to facilitate analysis. More specifically, we quantify two independent factors: 1- whether or not the action is used during the learning of visual features, and 2- whether or not the action is integrated into the representation of the current image.

Models and results We propose simple models based on three elements. First, a convolutional Variational Auto-Encoder (VAE) [131] provides a representation of the current glimpse. Second, a Long Short-Term Memory (LSTM) neural network [118] integrates the representation of the current observation with past ones to construct a global representation of the observed image. Third, the recoder, a neural network that we introduced, generates a latent embedding of the next glimpse with respect to the upcoming action and the representation from the LSTM. This embedding is used to reconstruct the next visual input as a predictive generative task. With these elements we proposed 4 versions of the model that differ in two aspects.

To study the importance of using action within the representation, we compare two variations of the architecture: the PreLSTM one (see figure 2.2a), which integrates the actions before the LSTM, thus forcing the representation to be a mixture of sensory and motor information, and the PostLSTM one (see figure 2.2b), where the action is concatenated after the LSTM, thus the representation is purely visual.

In order to study the importance of action while learning representation we compare two variants of the training method: a classical end-to-end learning, where action can influence the learning of representations, and a two-step learning procedure, where the VAE is first

pretrained²³ so that the action is not included in the glimpse representations, then the LSTM (either with the PreLSTM or the PostLSTM version) is trained with the weights of the VAE frozen.

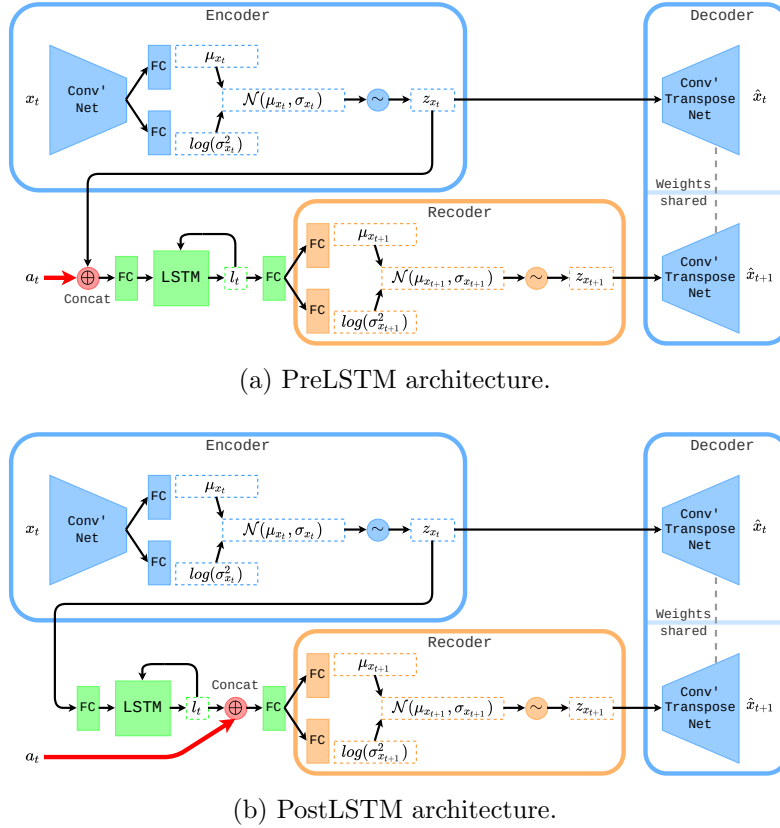


Figure 2.2: The input x_t passes through the encoder, transforming x_t in its latent representation z_{x_t} . Then z_{x_t} passes through the decoder, giving \hat{x}_t the reconstructed input produced by the VAE. On the other branch, for the PreLSTM architecture, the action a_t (the position of the next observed glimpse) is concatenated with z_{x_t} and is then fed to the LSTM, which outputs l_t the global representation of the image. For the PostLSTM one, z_{x_t} is fed directly to the LSTM and the action a_t is concatenated with l_t . From this representation (either l_t alone or concatenated with a_t) the recoder computes $z_{x_{t+1}}$ and finally, by passing through the decoder, shared with the one of the VAE, the constructed prediction \hat{x}_{t+1} of the next glimpse x_{t+1} .

Performance was evaluated on a classification task using the learned representation (see figure 2.3). First, we observe that in all cases the models that do not use the action during the learning of the VAE’s encoder (-Sep suffix) perform worse than their counterpart that uses the action (no suffix). Secondly, we can see that all models that integrate the action in the LSTM perform better than their counterpart that integrate the action after the LSTM, except for the 28×28 MNIST dataset, which is the simplest. Overall, these results confirm the importance of integrating the action in the representation structure and learning. Based on these promising results, in the next section 2.2.2, we will integrate

²³To have a fair comparison, the prediction task, that requires action, is replaced by a second reconstruction of the current glimpse with an identity recoder.

similar ideas into the instance discrimination architecture, instead of a generative one as here.

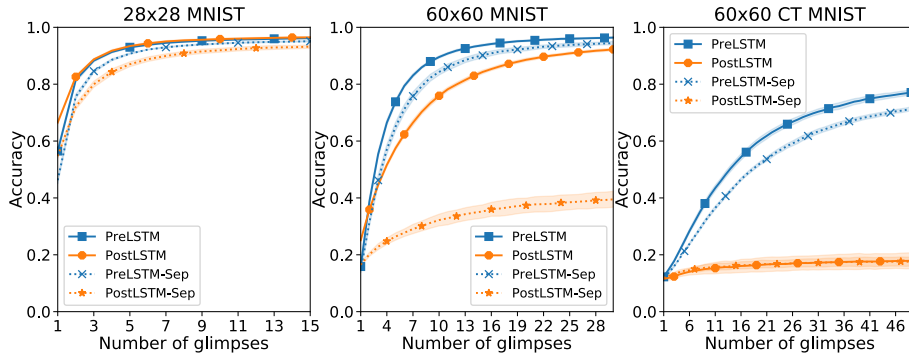


Figure 2.3: Classification accuracy as a function of the number of glimpses received. The MNIST digits dataset [144] consists of 28×28 pixel images containing centred white handwritten digits on a black background. The 60×60 MNIST are the same images but rescaled to 60×60 pixels so that the patches are no longer digit fragments but strokes and curves. The 60×60 CT MNIST [157] are 60×60 images with black background with a 28×28 MNIST digit randomly placed on them, and with four 8×8 clutters (extracted from other MNIST digits) randomly added to them, so that clutters and digit positions are totally unpredictable if never seen.

This work started during Valentin Chaffraix’s master’s thesis and was finished during Alexandre Devillers’ one. It led to a publication in ICDL [6], which can be found in the appendix 6.1 .

2.2.2 Equivariant representation learning

2.2.2.1 Context and objectives

Our previous research work has shown that the siamese learning architecture can be a good basis for learning visual representation (see section 2.2.1.1). In the meantime, SSL for visual representation has made a lot of progress and the models are progressively closing the gap with the supervised baseline [69, 86, 111, 213]. Although these SSL methods are diverse (see [149] for a review), they are essentially siamese networks performing an instance discrimination task. Their underlying mechanism is to maximise the similarity between the embeddings of related synthetic inputs (a.k.a. views), created by data augmentations (*e.g.* cropping, colour jitter, etc.), that share the same concepts, while using various tricks to avoid a collapse to a constant solution [122, 125]. This induces the latent space to learn an invariance to the transformations used. Thus, a lot of effort has been put into experimentally searching for the right set of augmentations, focusing mainly on achieving the highest object recognition performance on the ImageNet dataset.

However, for a given downstream task the representations benefit from invariance to some augmentations while variance to others is preferable [89]. Indeed, there is a trade-off in the choice of augmentations: they need to significantly modify the images to avoid simple solution shortcut learning (*e.g.* relying only on colour histograms), while keeping augmentation-related information in the representations to perform the downstream tasks

(*e.g.* the need of colour for bird or flower classification). Consequently, recent works have explored various ways of incorporating sensitivity to augmentations while maintaining an invariance objective in parallel, *e.g.* by imposing a sensitivity to rotations that is not used for invariance [89] or by learning multiple latent spaces, each one being invariant to all but one transformation [209]. In line with our previous work showing that integrating action into the representations with generative methods seems to improve performance (see section 2.2.1.2), we proposed the same kind of idea but introducing augmentation with this instance discrimination approach. Indeed, the augmentations applied to the image can, to some extent, be considered as the result of applying an action to the environment [138]. More specifically, we proposed a module that learns another embedding space with an equivariance to the augmentations, whose structure is automatically learned, in addition to the classical invariance task.

2.2.2.2 Model and results

While the notion of augmentation insensitivity is related to invariance in the literature, sensitivity can be conceived in various ways. In this work we proposed to use the mathematical concept of equivariance as a way to implement sensitivity and structure the latent space (see figure 2.4). Formally, let \mathcal{T} be the distribution of possible transformations applied to the image \mathbf{x} to create the views, and f denotes a projection from the input space to a latent space, usually a CNN plus an MLP for image representation learning. This latent space is said to be invariant²⁴ to \mathcal{T} if $\forall \mathbf{x}, \forall t \in \mathcal{T}, f(t(\mathbf{x})) = f(\mathbf{x})$. On the contrary, the latent space is said to be equivariant²⁵ if $\forall \mathbf{x}, \forall t \in \mathcal{T}, \exists u_t f(t(\mathbf{x})) = u_t(f(\mathbf{x}))$ where u_t is a transformation in the latent space parameterised by the transformation t . In this work, we aim at non-trivial equivariance (*i.e.* different from invariance), where u_t actually produces a displacement in the latent space (*i.e.*, that $u_t \neq Id$) to somehow encode information related to the augmentations in the representation.

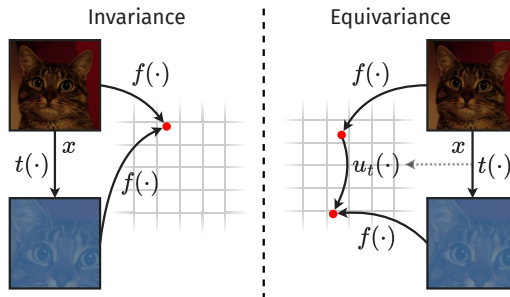


Figure 2.4: Comparison between the properties of invariance and equivariance. With invariance, a transformation of the input has no effect on its embedding, whereas with equivariance it induces a transformation that depends on the transformation applied (and, in our work, on the input).

²⁴As we are interested in the properties of the representation space, some works (*e.g.* [69, 86, 111, 213]) prefer to rely on the following formula $\forall \mathbf{x}, \forall t \in \mathcal{T}, \forall t' \in \mathcal{T} f(t(\mathbf{x})) = f(t'(\mathbf{x}))$. Note that if the identity function is part of \mathcal{T} , which is the case in recent approaches, then the two definitions are equivalent.

²⁵Note that the order of the quantifiers in the formula is $\forall \mathbf{x}, \exists u_t$ and not $\exists u_t, \forall \mathbf{x}$ which would have imposed more constraints on the latent space. This could be a very interesting property for generalisation, but we had to relax some constraints in the network to achieve good performance. However, putting more structure in the representation is definitely a future research axis to look for better robustness and generalisation (see the perspectives section 5.2).

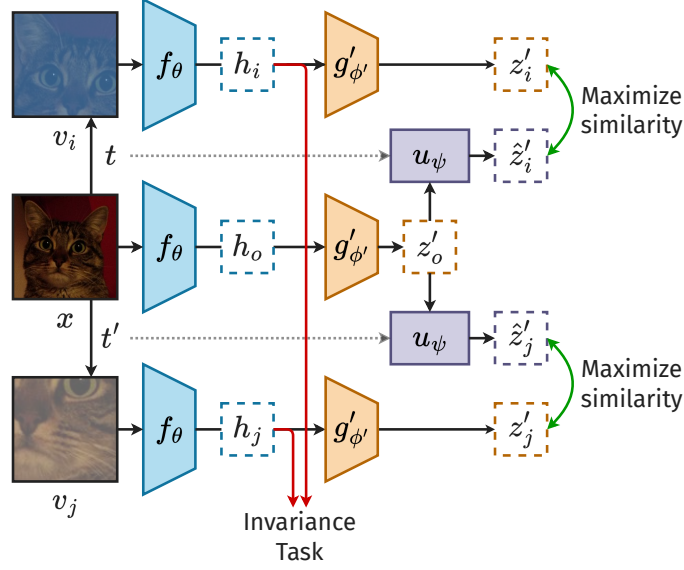


Figure 2.5: The model learns similar embeddings for an augmented view (z'_i) and the prediction (\hat{z}'_i) of the displacement in the embedding space caused by this augmentation, i.e. a space where the projected representations are equivariant, while there is classically another space where the projected representations are invariant (red arrows).

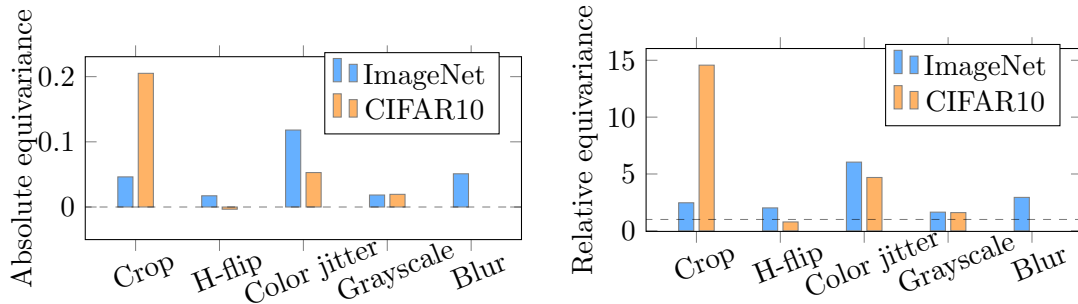
Based on these definitions we proposed EquiMod, which is an additional generic module that learns a projection of the representation where the embeddings are equivariant to the augmentations, to capture some augmentation-related information originally suppressed by state-of-the-art methods that learn an invariance task (see figure 2.5). More technically, let t and t' denote two augmentations sampled from the augmentation distribution \mathcal{T} . For a given input image \mathbf{x} , two views are defined as $\mathbf{v}_i := t(\mathbf{x})$ and $\mathbf{v}_j := t'(\mathbf{x})$. We note f_θ an encoder parameterised by θ which produces representations from images (in practice a CNN) and g_ϕ a projection head parameterised by ϕ (in practice an MLP) which projects the representations into an embedding space. The representations are defined as $\mathbf{h}_i := f_\theta(\mathbf{v}_i)$ and as $\mathbf{h}_j := f_\theta(\mathbf{v}_j)$, and the embeddings as $\mathbf{z}_i := g_\phi(\mathbf{h}_i)$ and as $\mathbf{z}_j := g_\phi(\mathbf{h}_j)$. Then, state-of-the-art models learn to maximise the similarity between \mathbf{z}_i and \mathbf{z}_j , while using diverse tricks to maintain a high entropy for the embeddings, thus avoiding collapsing to constant representations for all inputs. To extend these previous works, we introduce a second latent space to learn our equivariance task. Thus, we define a second projection head $g'_{\phi'}$ parameterised by ϕ' and we note $\mathbf{z}'_i := g'_{\phi'}(\mathbf{h}_i)$ and $\mathbf{z}'_j := g'_{\phi'}(\mathbf{h}_j)$, the embeddings of the views \mathbf{v}_i and \mathbf{v}_j , respectively, in this latent space. Moreover, for the given unaugmented image \mathbf{x} we note its representation $\mathbf{h}_o := f_\theta(\mathbf{x})$, which is used to create the embedding $\mathbf{z}'_o := g'_{\phi'}(\mathbf{h}_o)$. We define u_ψ a projection parameterised by the learnable parameters ψ , which will be referred to later as the equivariance predictor (in practice an MLP). The goal of this predictor is to produce $\hat{\mathbf{z}}'_i$ from a given \mathbf{z}'_o and t (respectively $\hat{\mathbf{z}}'_j$ for \mathbf{z}'_o and t'). To satisfy to the equivariance equation, we should have $\hat{\mathbf{z}}'_i = \mathbf{z}'_i$ (respectively $\hat{\mathbf{z}}'_j = \mathbf{z}'_j$). In practice, this is approximated by considering $(\mathbf{z}'_i, \hat{\mathbf{z}}'_i)$ as a positive pair (respectively $(\mathbf{z}'_j, \hat{\mathbf{z}}'_j)$) and all others, except $(\mathbf{z}'_i, \mathbf{z}'_j)$, as negative ones to learn a contrastive loss, more precisely the Normalised Temperature-scaled cross entropy [86]. This equivariance loss was weighted with some hyperparameters and added to the invariance loss of the chosen baseline model.

Method	ImageNet		CIFAR10	
	Top-1	Top-5	Top-1	Top-5
SimCLR [86]	69.3	89.0	-	-
Barlow Twins [213]	73.2	91.0	-	-
VICReg [69]	73.2	91.1	-	-
BYOL [111]	74.3	91.6	-	-
SimCLR*	71.57	90.48	90.96	99.73
SimCLR* + EquiMod	72.30	90.84	92.79	99.78
BYOL* (100 epochs)	62.09	84.01	-	-
BYOL* + EquiMod (100 epochs)	65.55	86.74	-	-
BYOL* (300 epochs)	71.34	90.35	-	-
BYOL* + EquiMod (300 epochs)	72.03	90.77	-	-
BYOL* (1000 epochs)	74.03	91.51	90.44	99.62
BYOL* + EquiMod (1000 epochs)	73.22	91.26	91.57	99.71
Barlow Twins*	-	-	86.94	99.61
Barlow Twins* + EquiMod	-	-	88.87	99.71

Table 2.1: **Linear Evaluation**; top-1 and top-5 accuracies (in %) under linear evaluation on ImageNet and CIFAR10 (symbols * denotes our re-implementations, and ‡ denotes only 100 epochs training).

We tested our method on the ImageNet [91] and CIFAR10 [135] datasets. As our model is an additional module, we used it as a complement to SimCLR, BYOL and Barlow Twins, 3 state-of-the-art invariance methods with quite different ideas, in order to test the genericity of our module. The results of the supervised linear evaluation of the representation are presented in Table 2.1. In all but one baselines and datasets tested, EquiMod improves the performance of the baselines used, which supports the efficiency and the genericity of our approach. Moreover, we observed in practice that the model effectively learns a non-trivial equivariant structure where the magnitude of the displacement in the latent space depends on the type of augmentation (see figures 2.6a and 2.6b). All this confirms our idea that adding information from the action in the representation, here via an equivariance task to the applied augmentation, helps to extract more relevant information and to improve performance.

This work was part of the PhD thesis of Alexandre Devillers. It led to a publication in ICLR [5], which can be found in the appendix 6.2 . As we had to reimplement state-of-the-art SSL models, as the support of our architecture, we also submitted two articles (consisting of the open source codes and a description of the reimplementations) to the journal ReScience, which is dedicated to scientific reproducibility.



(a) Absolute equivariance measure (computed as $\text{sim}(z'_i, \hat{z}'_i) - \text{sim}(z'_i, z'_o)$) for each augmentation. (b) Relative equivariance measure (computed as $\frac{1 - \text{sim}(z'_i, z'_o)}{1 - \text{sim}(z'_i, \hat{z}'_i)}$) for each augmentation.

2.3 Structure of learned representations

2.3.1 Context and objectives

Despite the success of the instance discrimination methods, that we introduced and improved in the previous section 2.2.2, the underlying mechanisms that enable these models to produce representations that are highly effective for downstream classification tasks remain poorly understood. The learning mechanism tends to favour similar (respectively dissimilar) representations for all the variations of one instance (respectively for different instances), which should discourage the model from learning shared embeddings across instances belonging to the same class. This apparent contradiction has been explored in some theoretical and practical approaches.

Theoretically, the projection head may benefit to the learning by allowing the representations to retain more information related to the augmentations, rather than forcing them to specialise solely on the invariance task [210]. More globally, [207] suggests that aggressive data augmentations create similarity overlap between samples from different instances. This helps the model to gradually cluster these intra-class samples together, effectively climbing the ladder of chaos. However, it has been shown that this theory alone may not fully explain the underlying mechanisms [184].

Beyond theoretical explanations, recent studies have empirically investigated some of the structural properties of the latent spaces learned by SSL methods. For instance, [214] observed that SSL representations tend to organise images such that nearest neighbours in the latent space are often not of the same class, in contrast to supervised learning. They also find that in SSL, similar representations are more closely related in pixel space. Similarly, [110] find that while both SSL and supervised learning models improve in performance as the layers deepen, the representations become increasingly dissimilar. Notably, the similarity between self-supervised and supervised representations collapses after the projection head in SSL models, highlighting the distinct nature of the features learned through self-supervision.

In this work, we have taken an experimental approach to empirically analyse the structural components that emerge in the latent spaces of different instance discrimination-based methods. Our objective is to identify both shared and distinct structural patterns across various SSL approaches, and to determine which of these patterns contribute most effectively to high classification accuracy in downstream tasks. Ultimately, our findings could inform the design of new SSL methods that explicitly optimise the emergence of

beneficial structural patterns, leading to more generalisable representations.

2.3.2 Method and results

We employed a diverse set of structural descriptors, that capture different facets of the latent space organisation, to analyse and compare them at multiple scales, across SimCLR [86] and BYOL [111] models, and at various network depths (the backbone representation h and the projection head output z). More precisely, we identified the structural characteristics that consistently correlate with high classification accuracy on the ImageNet validation dataset [91].

We found no structure (correlated with the accuracy) in the false negative and false positive samples. This suggests that misclassified samples, as identified during the linear evaluation step, may already be outliers in the learned representation space, thus questioning a possible limitation in the SSL methods or in the nature of these samples. There is also no correlation with metrics such as PCA, mutual information, and activation distribution statistics. This could be explained if classes do not need the same number of dimensions to be well represented while SSL methods have more than enough dimensionality in their latent spaces to allow some variation in the number of dimensions used. More intriguingly, cluster-related metrics within each class are not correlated with performance. These structures may be useful for more fine-grained labels than those used in classification, which is an open perspective.

On the contrary, we observed that classes benefit from being orthogonal to each other, as the closer the maximum and median similarity values between classes are to zero, the better the classification performance (see figure 2.7). Notably, this relationship diminishes when nearest neighbours are masked, suggesting that confusion may occur for classes that are close in the latent space. Moreover, the more collapsed a class is, the higher the accuracy tends to be (see figure 2.8). Taken together, these observations suggest that the representations learned by instance discrimination favour high intra-class similarity and high inter-class dissimilarity. These properties mirror those optimized by instance discrimination, namely alignment and uniformity [206], but they manifest at the class level rather than at the instance level, despite the absence of class information in this SSL context, which is quite surprising. Remarkably, this structural pattern also emerges in the output of the projection head, indicating that the projection head alone is not essential to explain the emergence of this pattern.

Descriptors based on the correlation dimension, community detection and connected components further confirm the previous results. The correlation dimension, which reflects the diversity and intrinsic dimensionality of the latent space, supports the idea that denser, less hierarchical structures are more favourable for classification (see figure 2.9). Interestingly, the representations produced by the backbone are richer and more complex than those of the projection head, supporting the hypothesis that the projection head can simplify the representation space for the invariance pretext task. Similarly, the modularity score from community detection, which measures the degree to which the latent space can be divided into distinct communities, is also consistent with these findings. Lower scores, indicating less modular and more homogeneous latent spaces, are associated with higher accuracy. This suggests that a more uniform latent space, where data points are less fragmented into distinct clusters, is advantageous for linear evaluation. The connected component analysis further supports this, showing that lower thresholds are needed to achieve a minimum of 50% connected components in higher performing classes. This implies that

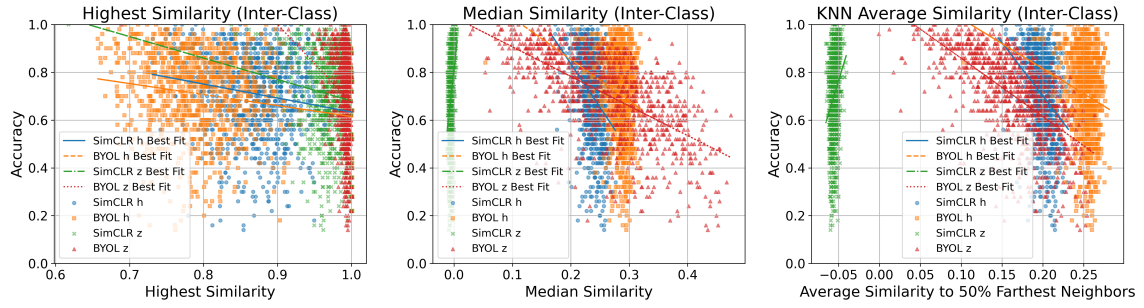


Figure 2.7: Left (respectively Centre / Right): The plot shows the relationship between the highest similarity between interclass samples (respectively the median similarity between interclass samples / the average similarity to the 50% most distant neighbours) and the accuracy. The different colours represent the two models and two network depths tested. Plain lines represent the best linear fits of the measures.

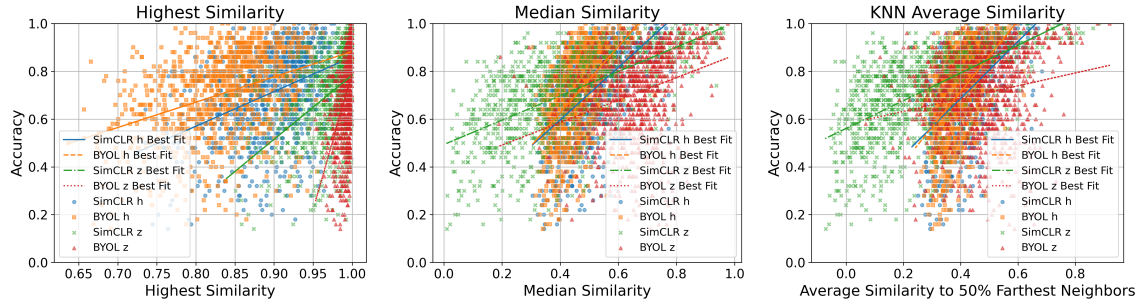


Figure 2.8: Left (respectively Centre / Right): The plot shows the relationship between the highest similarity between neighbours within the same class (respectively the median similarity within a class / the average similarity to the 50% most distant neighbours within a class) and the accuracy.

a dense structure, where most data points remain connected even as the similarity threshold changes, is beneficial for classification accuracy. Fewer, and therefore larger, isolated islands that persist as the threshold moves indicate a more robust, interconnected latent space that supports better classification results.

Finally, while structural differences between the methods are sometimes apparent at the output layer of the projection head, understandably given their distinct learning objectives, the structures at the output of the backbone remain similar. Moreover, apart from differences in Median Similarity and KNN Average Distance between classes (see figure 2.7), likely due to SimCLR’s use of explicit negative pairs, the projection head structures for both methods are still more similar to each other than to the backbone representations. This suggests that even fundamentally different instance discrimination methods, such as SimCLR and BYOL, tend to learn similar structure of representations to some extent.

This work was done in the PhD thesis of Alexandre Devillers and can be found in its current version in appendix 6.3 .

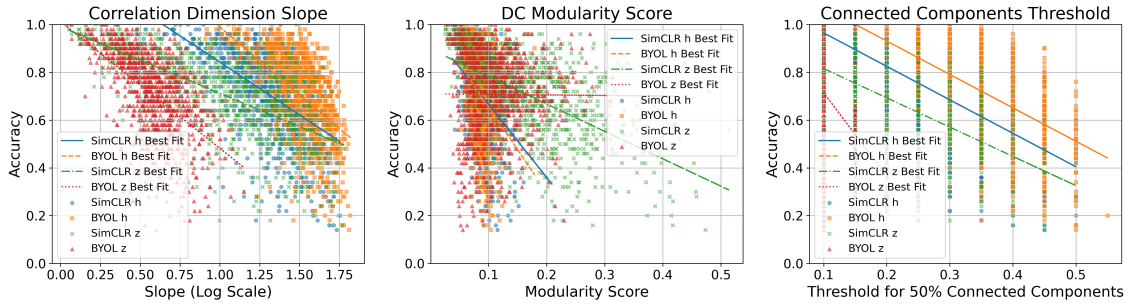


Figure 2.9: Left (respectively Centre / Right): The plot shows the relationship between the slope of the correlation dimension (on a logarithmic scale) (respectively the modularity score from community detection / the threshold at which 50% of connected components are formed in the latent space) and the accuracy.

2.4 Conclusion and perspectives

In this chapter, I presented contributions related to representation learning with self-supervised methods. In section 2.2.1.1 I showed that simple salience methods can provide relevant pairs of data for siamese methods to learn object representations from video. In another aspect, we studied more specifically the importance of using action in the representation and during learning with an input stream of visual glimpses from images (see section 2.2.1.2). In this context, we observed that action can improve the learning and structure of the representation. By combining these two ideas formalised in an equivariance framework, we proposed in section 2.2.2 a module that can be added to any visual instance discrimination model, the current state-of-the-art in SSL architectures. By forcing the learning of representations that can be projected onto two distinct embeddings, one classically invariant to augmentations, and another equivariant, with a function learned by a dedicated network, this module improves the classification performances obtained. Although we still have to evaluate its generalisation capabilities with transfer learning. As illustrated in section 2.3, the structures learned by these methods are not yet well understood. We have shown that, although the projection head plays a role, it does not alone explain the properties that emerge from the representation. Moreover, we observed that the representations of images from the same class tend to be very close, while being orthogonal to those of the other classes, which is well suited for linear classification, but may be limited for other tasks, especially those requiring finer details, which should be further tested.

These works open many perspectives for a deeper understanding of the latent space structure and its improvement, especially to obtain more robust, generic and transferable representations. It could be interesting to formalise the required properties of the representations to be transferable to various tasks, which may include multi-scale structures, for instance to focus on shape or texture, and projections to different spaces, for instance to be able to consider colour when needed (*e.g.* to recognise my white coffee cup in my office) and discard when needed (*e.g.* to find any coffee cup during a conference break). This may require attention mechanisms, which I will evoke in chapter 3, or some kind of prompting or information retrieval that has been shown to be effective in large language models. In the future, I will focus on the deeper study of the integration of action in representation in line with sensorimotor theories [161], that will be discussed in more details

in the perspectives section 5.2.

More globally, when adopting an embodied vision of cognition, what makes a good representation is not limited to classification performance. One of the goals is to help the agent make sense of the surrounding environment, which I will consider from the perspective of multimodal perception in the next chapter 3. Moreover, the agent must be able to deal with a changing environment and to interact with it, which I will study with active and incremental learning in chapter 4.

3 Multimodal perception

3.1 Introduction

In the previous chapter 2, I presented some of my works on learning visual representations, where the interest was based on their ability to correctly classify objects from a given dataset. For an autonomous agent, however, one of the aims of representations may be to help make sense of the world through a perceptual process. This can be formulated as a question of projection, from the external world, through a sensor, to structured internal representations that are the support of the emergence of meaningful perceptions for the agent, related to its (own) current objectives. This is not a one-way pipeline as (inter)action can also play a major role in perception as suggested by sensorimotor theories [161], and as representation learning and perception can be considered as two dynamically intertwined processes, as emphasised by the constructivist approach [167]. Moreover, attentional processes play a fundamental role in filtering relevant information for perception. This can be bottom-up attention (a mechanism we used in section 2.2.1.1 for extracting proto-objects from video), *e.g.* when our gaze is attracted by movement at the periphery of our field of view, or top-down one, *e.g.* when we do not pay attention to (an actor dressed as) a gorilla in the middle of the screen because we are carefully counting basketball passes, a mechanism called attentional blindness [112]. Although I will not go so far, in this chapter, I will present some work on the emergence of integrated perception from spatially organised representations.

In biological agents, but also in robotics, perception is fundamentally multimodal, as the world is observed through multiple sensors that provide various types of information about the environment. This raises the question of the different frames of reference and their alignment between modalities [168], which I considered to some extent in my PhD and post doc (see section 1.6.1), but will not address here. The main focus will be on the combination of information, as sensory cues can be redundant (*e.g.* the shape of my coffee cup can be seen and touched), complementary (*e.g.* I can hear the sound of my coffee cup when I put it on my desk and touch its shape at the same time) or even unrelated (*e.g.* the sound I hear in my headphones should not be associated with my coffee cup I see in front of me). Moreover, this can depend on the specific context, *e.g.* if I touch my spacebar it may or may not start the music in my headphones, depending on the active window. All this raises the question of which modality to merge. The other question is how to merge them, especially which ones are the most relevant. There can be intrinsic preferences for a type of sensor depending on the property to be measured, *e.g.* recognising the shape is usually easier by looking than by touching. However, this relevance is context dependent (*e.g.* I will mainly rely on touch to find my cup on my desk when it gets dark at night), but also task dependent (*e.g.* if I am looking where my cat is sleeping I will look for its shape and colour rather than its snoring, whereas if I am trying to find him/her as soon

as possible I would rather call him/her and listen to any sound).

These questions have been studied extensively in humans with psychophysical experiments²⁶. Regarding the stimuli to merge, the spatial and temporal dimensions have mainly been studied, especially in audio-visual tasks. The main findings of these studies are that humans merge information based on their spatial and temporal congruencies [191], with some margin, and that these congruencies are based on correlations that calibrate the senses [171]. As for the fusion, experiments show that the integration is based on the weighting of relevant stimuli according to their reliability [93]. Therefore these processes can be seen through the spectrum of mixing cues in a statistically optimal way. Thus, most of works, both in modeling but and in machine learning, are based on Bayesian approaches.

Regarding multimodal learning, three main approaches are used in machine learning [63] (with some variations depending on the type of learning considered). In early fusion, multimodal inputs are considered as usual inputs, and the model has to learn the correlations that exist within modalities (the classical objective) but also between modalities. This can be challenging as each modality can have its own feature structure which can be hard to make compatible altogether. On the contrary, late fusion proposes to process each modality with a dedicated architecture and to mix them at the end, for example with a voting system. In this way, the fusion takes place within a common semantic space, but finer grain correlations may be lost. In between, intermediate fusion proposes to mix information during the processing flow, trying to get the best of the two other alternatives. In all cases, the models focus on learning the information shared by all modalities, possibly conditioned by the task in supervised learning. Thus, they are more interested in the question of what to merge rather than how to merge. Indeed, the weighting of modalities can be seen as a side effect that helps to accomplish the task rather than the actual focus of the study.

In this chapter, I will present the work done during my AMPLIFIER project, in collaboration with psychophysicists, statisticians and computer scientists. The main focus was on the question of the weighting of audio-visual information in perception, and the influence of active perception (via saccades) in this process.²⁷ We have chosen to look at this question from the point of view of a spatial decision process. This is consistent with the kind of task that we have considered, which consists of localising a sound while a visual stimulus is presented at a different location. But this may be generalisable if we consider this in relation with a representation learning task whose aim would be to project the external world onto a decision space²⁸, where similar choices are closely projected. A parallel can be drawn with brain structure, where features in sensory areas are spatially self-organised, which may be an important feature of cognition [80]. More specifically, we will rely on Dynamic Neural Fields (DNF), which I will describe in section 3.3, which is a neural network model that provides a decentralised competition and fusion mechanism relying on a topological substrate where information is spatially localised. It can also be

²⁶There is also a lot of literature in neuroscience on multimodal integration at the neuronal level, mostly on animals for ethical reasons. I will not go into details here, as I am more focused on the behavioural aspect, yet with some relationship to the neuronal substrate. An interested reader can refer to [82] for more details.

²⁷In this project we proposed a new psychophysical protocol of the ventriloquist effect, especially including the active perception part. While the participants data have been recorded, the analyses are not yet complete, and the modelling has barely begun. Thus, this psychophysical part and the active perception aspect cannot unfortunately be included in this manuscript.

²⁸We can also imagine that this space may be multidimensional to represent multiple objectives.

considered as a dynamical system that has been proposed as a theoretical model for the dynamics of cognition [187]. In section 3.2 I will present how this model and its properties can be located on a continuous panorama of decision-making algorithms in psychophysics and robotics. In section 3.3, we propose a modelling of audio-visual fusion in humans using DNF, which is more grounded in the neural substrate than classical Bayesian approaches. Finally, in section 3.4 we investigate how this model can be extended to irregular topologies for application with autonomous agents.

3.2 Decision making algorithms: from psychophysics to robotics

3.2.1 Context and objectives

In this work, we are interested in decision-making, a process that ranges all the way from the smallest steps of perception, like picking a visual stimulus to gaze at, to more complicated procedures, like solving a puzzle. It has been studied extensively in various fields of the humanities, from psychophysics and neuroscience to social sciences and economics, but also in engineering science, and robotics in particular. Out of these very different fields of research, similar behaviours are studied, albeit using different setups and different models. Psychology and neuroscience have put a lot of focus on how and what decisions are made by studying perceptual experiments that can provide insights into some of the inner mechanisms of decision-making. In robotics, on the contrary, decision-making is most often viewed from the perspectives of computer vision (categorisation, tracking, etc.), machine learning (classification, reinforcement learning, etc.) or swarm intelligence. In these cases, the focus is more on what decisions are made rather than on the how it is taken, yet the field of explainable AI is pushing in this direction. Nevertheless many parallels can be drawn, either by design or not, between biologically-motivated and engineering-driven models [165]. Thus, we proposed to compare a representative sample of existing learning-free decision-making algorithms from different domains (notably neuroscience, psychophysics, and robotics) in a domain-agnostic perspective within a unified framework²⁹ that we proposed to build bridges between different classes of models. We also set up simple but representative scenarios to illustrate and compare the qualitative properties of each algorithm.

3.2.2 Comparative framework and results

Decision-making algorithms can be divided into three main families: dynamic accumulators, probabilistic/Bayesian models and logic-based models.

Psychophysics and neuroscience have proposed many models of decision-making, most notably using accumulators, where evidence is gradually accumulated over time until a given threshold is reached. The most common of these is the drift-diffusion model (DDM), in which two³⁰ opposite thresholds are set [175], while the leaky competing accumulator (LCA) also accounts for information decay over time [74]. Applications of accumulator models are not exclusive to the humanities but have also been used for decision-making in

²⁹Note that this framework was co-designed with an open source software.

³⁰The DDM can be easily extended to more choices by using multiple units, each representing a possible choice, that are put in competition until one prevails.

robotics, most notably in the form of Dynamic Neural Fields (DNF)³¹, population-coded accumulator models running on a topological map [187], which we will use for multimodal merging in the contributions presented in the next sections 3.3 and 3.4.

Probabilistic models³² constitute another cross-disciplinary category. For instance, many models based on maximum likelihood estimation (MLE), a simple Bayesian inference, have been used for data fusion to reflect computations observed in psychophysics [93], and Kalman filters (KF) [129] are widely used in robotics.

Engineering science has also its own methods, for instance Fuzzy Logic (FL), which describes operations made on fuzzy sets in which truth values are no longer binary but instead compared to membership functions expressing possibility values, between 0 and 1 [212]. By fuzzifying sensory inputs and combining their membership functions, it is possible to create fuzzy commands that can be exploited in computer vision, data fusion or robotics.

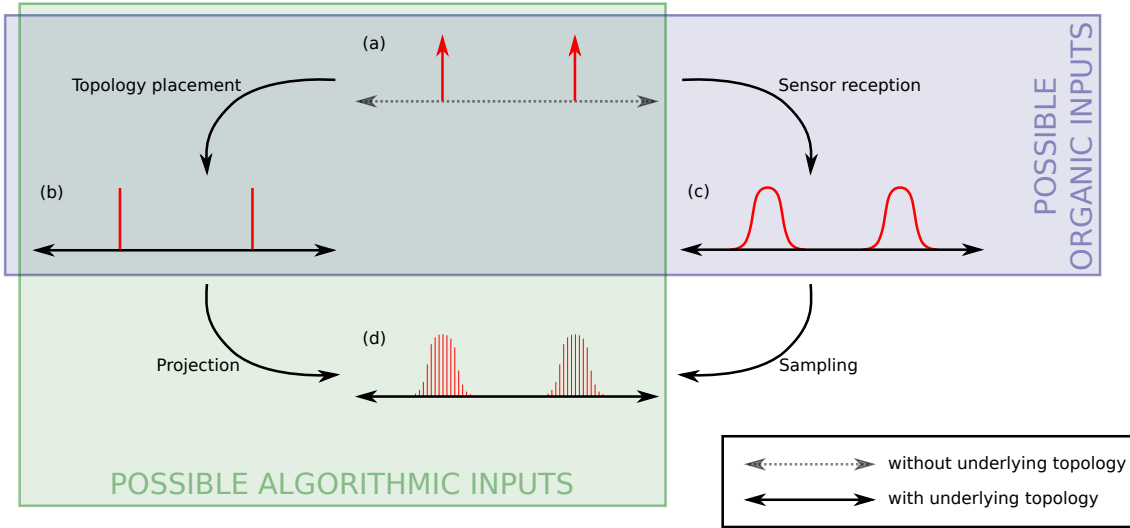


Figure 3.1: Depending on the level at which activities are considered in the perception-decision process, they can be viewed in an organic system as: (a) sparse without an underlying topology (i.e. a forced choice between the available options), (b) sparse in an underlying topology (meaning there exists a distance metric that allows interpolations), or (c) continuous (*e.g.* because of the spread of the signal in its environment, the blur added by sensors with limited resolution, or the overlapping receptive fields of the units receiving that activity). Artificial systems and computational models may require activities to be discretised (i.e. sampled in signal processing sense of the term) from continuous stimuli (d). The latter can also be obtained from (b) using a discrete kernel projection *e.g.*, a Gaussian.

In our review, we considered decision-making models in a broad sense as filters of noise and unwanted components that take activities as input and give one activity as output that can be used by another system: motors, other decision modules, or the same

³¹We will test two different sets of parameters, referred to as DNF₁ and DNF₂ in the results, to illustrate the various possible dynamics of this model.

³²In this category we can also find active inference, a general framework for explaining decision making as free energy minimisation, which has been used in either neuroscience or robotics [98] but was not included in the comparison.

model in cases of recursion. When the models do not compute a single output value, we have provided one additional aggregator (a Winner Takes All (WTA) or a Weighted Sum (WS)) outside the model to produce such a result. Regarding the inputs, we assume that a minimum amount of projections has already been made, for instance by the sensors or pre-processing. We thus categorised the kind of inputs that the model (biological or artificial) can receive (see figure 3.1). Within these considerations³³, we formalised a framework using specific notations in order to emphasise the various shared characteristics of the decision-making algorithms: topology-based interaction between processing units, output aggregation, recursion, etc. (see figure 3.2). We described the chosen representative models of learning-free decision-making algorithms within this framework and proposed simple scenarios to illustrate their properties on the spatial aspect (the robustness to distractors, the interpolation or selection between multiple options, or the preference for single strong stimuli rather than weak but multiple close ones) and on the temporal aspect (the reaction time, the robustness to temporal obstruction or the tracking speed) (see table 3.1).

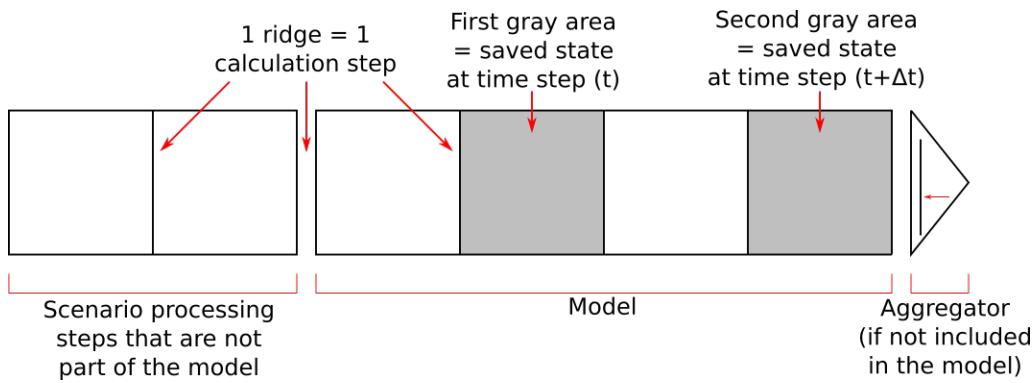


Figure 3.2: Visual framework used to unify the models. Models with recurrent states are shown unfolded, i.e. the process to go from time step t to time step $t + \Delta t$ is visible. The recurrence can be pictured by furling the pattern so that the grey areas touch each other. The aggregator part is shown attached to the model when the latter produces a readable direction directly, and detached if it has been added retrospectively. The arrow in the aggregator indicates where the final decision is read.

First works on formalism date back to discussions with Jean-Charles Quinton during my post doc. The work was (almost) finalised during the Jose Villamar's master's thesis and Simon Forest's PhD thesis, which we both supervised with Jean-Charles Quinton. This work will soon be submitted to a journal and the current version can be found in the appendix 6.4. Please note that the article is an unfinished version, although close to a final one, so some minor content is missing and it may still contain some errors and typos.

³³In practice, we have made the assumption that there exists an underlying topology in which the input can be placed, so we have not considered the case (a) on figure 3.1. This is not a risky assumption and it allows to consider operations such as interpolations and barycenters, which are otherwise not properly defined.

Model	Able to select one stimulus and ignore the rest									
	Able to interpolate between stimuli		Able to privilege a group of stimuli			Robust to temporary obstruction		Controllable speed of switch between stimuli		
	Able to track a target with little delay					Able to smooth trajectories				
	Suitable for sparse stimuli					Number of parameters				
WTA	Y	N	N	N	N	(NR)	N	Y	0	
FL	Y	Y	N	N	N	(NR)	N	Y	1	
WS	N	Y	(D)	N	N	(NR)	N	N	0	
KF	N	Y	(D)	N	(I)	Y	Y	N	3	
FFI	Y	N	N	Y	N	(NR)	N	Y	2	
NLCA	Y	N	N	Y	Y	(NR)	N	Y	3	
DNF ₁	Y	N	N	Y	Y	N	Y	N	7	
DNF ₂	Y	Y	Y	N	N	Y	Y	N	7	

Table 3.1: Summary of model properties. Y stands for Yes, N for No, (NR) for Not Relevant, (D) for depends on weight, (I) for starts instantly but can be slowed down. Note that a higher number of parameters provides more flexibility and more properties, but may be limited theoretically (because of the parsimony criteria) or practically (because each (hyper-)parameter has to be tuned and there is rarely a dedicated learning method).

3.3 Modeling of audio-visual perception in humans

3.3.1 Context and objectives

Dynamic Neural Fields (DNF) originated as a mathematical model of neural dynamics [61] and has been used to model neural activity at a mesoscopic scale, i.e. population or neurons [187]. Thus, it allows bridging the gap between the microscopic level of neural processes, to better understand adaptive functions found in living systems, and behavioural data, opening the way to building artificial systems able to reproduce them. As seen in the previous section 3.2, DNFs are a versatile decision-making algorithm. Depending on their parametrisation, they can achieve, for instance, selection or interpolation between multiple conflicting signals [197], robust selective attention in presence of noise and distractors [96], working or long-term memory of stimuli [183], and thus can be applied to visual attention [96] or (visuo)motor control [183]. However, the literature is scarcer when it comes to using DNF for multimodal fusion especially for modelling psychophysical phenomena.

In this work, we propose to apply DNF to model the ventriloquist³⁴ effect, in which human participants exposed to spatially incongruent visual and auditory stimuli will perceive the position of one stimulus shifted towards the other, depending on which modality has the highest relative precision. More specifically, we will draw on psychophysical data reported in the seminal work of [58], whose experimental paradigm and protocol can be easily replicated in silico. For each bimodal trial, participants were exposed to a

³⁴The effect takes its name from ventriloquist shows, in which the audience has the illusion that a puppet is speaking, while the sound is actually produced by the ventriloquist holding it.

sequence of two presentations of audio-visual stimuli (conflicting and non-conflicting in random order) and had to report which of them was perceived to be more leftward. In the non-conflict presentation, auditory information (1.5 ms sound click) and visual information (15 ms low-contrast Gaussian blob of controlled width, with standard deviation $\sigma_V \in \{2^\circ, 16^\circ, 32^\circ\}$) were perfectly aligned with each other, but their eccentricity relative to the centre of the participant’s field of view was manipulated from -20° to $+20^\circ$. In the conflict presentation, the stimuli were still aligned on the azimuthal axis, but an horizontal spatial discrepancy was introduced between the two, with the visual stimulus moving of $\Delta \in \{-5^\circ, -2.5^\circ, 0^\circ, 2.5^\circ, 5^\circ\}$ (from left to right) and the auditory stimulus moving of $-\Delta$. The experimental mean and variance over the population for each condition are reported in figure 3.5. We compared the performance of our model with these empirical data and with optimal Bayesian integration, the golden standard in multisensory integration [93]. Moreover, our model will be grounded in some considerations from neuroscience and would allow for trial-by-trial modelling, whose individual fit to the data is left for future work.

3.3.2 Model and results

Our model (see figure 3.3 for an overview) is based on the deep Superior Colliculus³⁵ (SC), a subcortical structure that receives projections from different modalities onto a series of multimodal neural maps and is well known for its multisensory integration [130]. The neural activity within the SC is computed with a DNF that models its evolution over time at each point of a topological space \mathbf{X} ³⁶. The mean field potential U at position $\mathbf{x} \in \mathbf{X}$ and time t is described by the following stochastic integro-differential equation (in practice the equation was simulated using an Euler scheme):

$$\tau \frac{\partial U}{\partial t}(\mathbf{x}, t) = -U(\mathbf{x}, t) + I(\mathbf{x}, t) + \int_{\mathbf{x}' \in \mathbf{X}} W(\|\mathbf{x} - \mathbf{x}'\|) f(U(\mathbf{x}', t)) d\mathbf{x}' + \varepsilon \quad (3.1)$$

where τ is the time constant that determines the response timescale of the entire field, I is the input stimulation over the field and f is a non-linear activation function, here a ReLU function to approximate the mean firing rate of the neurons. The last term ε is sampled from a normal distribution $\mathcal{N}(0, \sigma_N)$ and represents noise that can be interpreted either on a neurological level (a sum of numerous variations of activity induced by external neurons) or on a psychophysical level (*e.g.* perceptual noise) [187]. Finally, the kernel approximating lateral interactions within the continuous population of neurons is defined by:

$$W(\Delta\mathbf{x}) = \lambda_+ \exp\left(-\frac{\Delta\mathbf{x}^2}{2\sigma_+^2}\right) - \lambda_- \exp\left(-\frac{\Delta\mathbf{x}^2}{2\sigma_-^2}\right)$$

with $\lambda_+ > \lambda_-$ and $\sigma_+ < \sigma_-$, resulting in local excitation and more diffuse inhibition.

Based on neurophysiological findings, we decomposed the input I defined at each point of the DNF as the sum of a visual input I_V and an auditory input I_A (see figure 3.4 (a) and (b)). More precisely, the visual stimuli (a Gaussian blob) are projected from the retina onto our SC with a logpolar transformation [162] (as an ablation study, we also tested without this specific projection, hence you will find DNF+log and DNF+id in the results

³⁵Our architecture does not target to be an exact model of the multisensory pathways in the brain, which are much more diverse and complex than what we have considered.

³⁶In practice we used a 1-dimensional DNF because the inputs lie in a 1-dimensional axis and to make simulation and interpretation easier than a 2-dimensional map, which would have been closer to neuronal organisation.

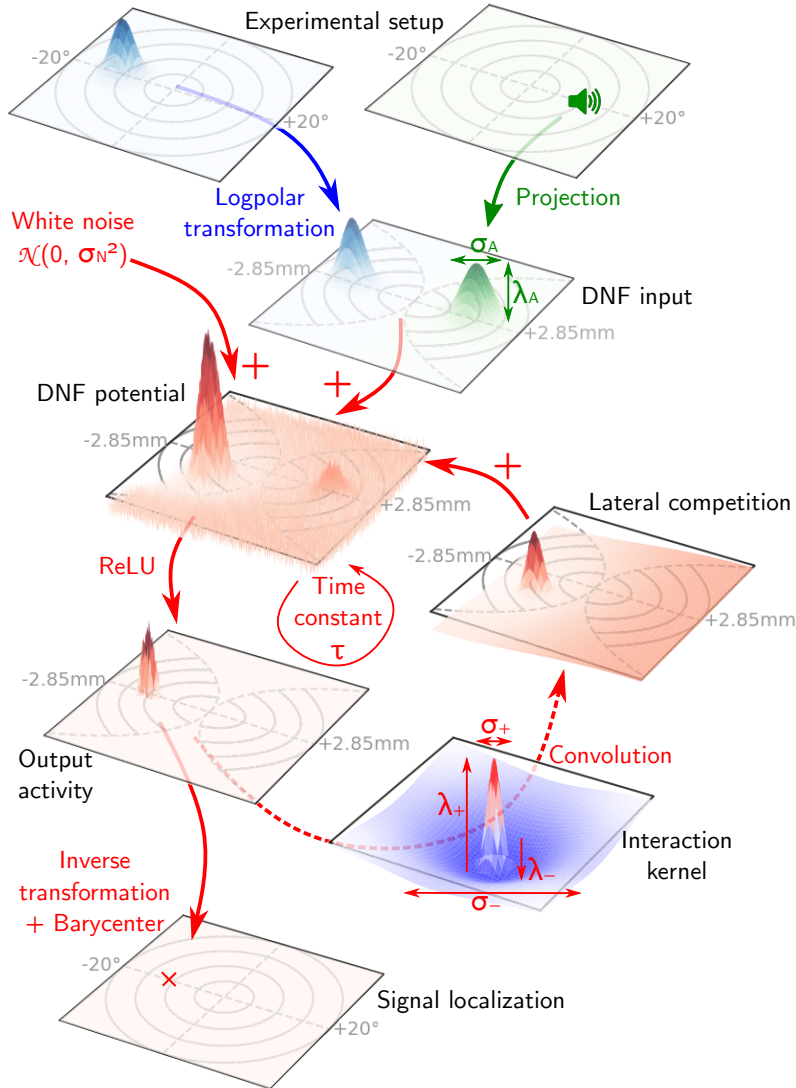


Figure 3.3: Our DNF model of audiovisual merging DNF. Each rectangle represents a map, either in retinal space (shown with concentric circles) or SC (hourglass shape, obtained by performing a log-polar transformation on the visual map). The blue (respectively green) arrow and text refer to visual (respectively auditory) pre-processing. Steps and parameters from the model, other than pre-processing, are shown in red.

figure 3.5). To our knowledge, there is no mathematical formulation of the projection of auditory inputs onto the SC, so we simply aligned the audio stimuli (also a Gaussian blob) with their spatially congruent visual counterparts to avoid introducing additional model parameters. The output of the model is the barycentre of the field output $f(U)$ (see figure 3.4 (c)). However, given the noise and non-linearities of equation 3.1, we relied on the Monte Carlo method to sample the localisation distribution under each condition through repeated simulations and estimated an empirical Gaussian distribution (see figure 3.4 (d)).

In the end, our model has eight free (hyper)parameters (six from the DNF equation and two for the inputs). For each parameter we extracted an interval in which suitable behaviour was possible and simply relied on an iterative and partial grid search approach

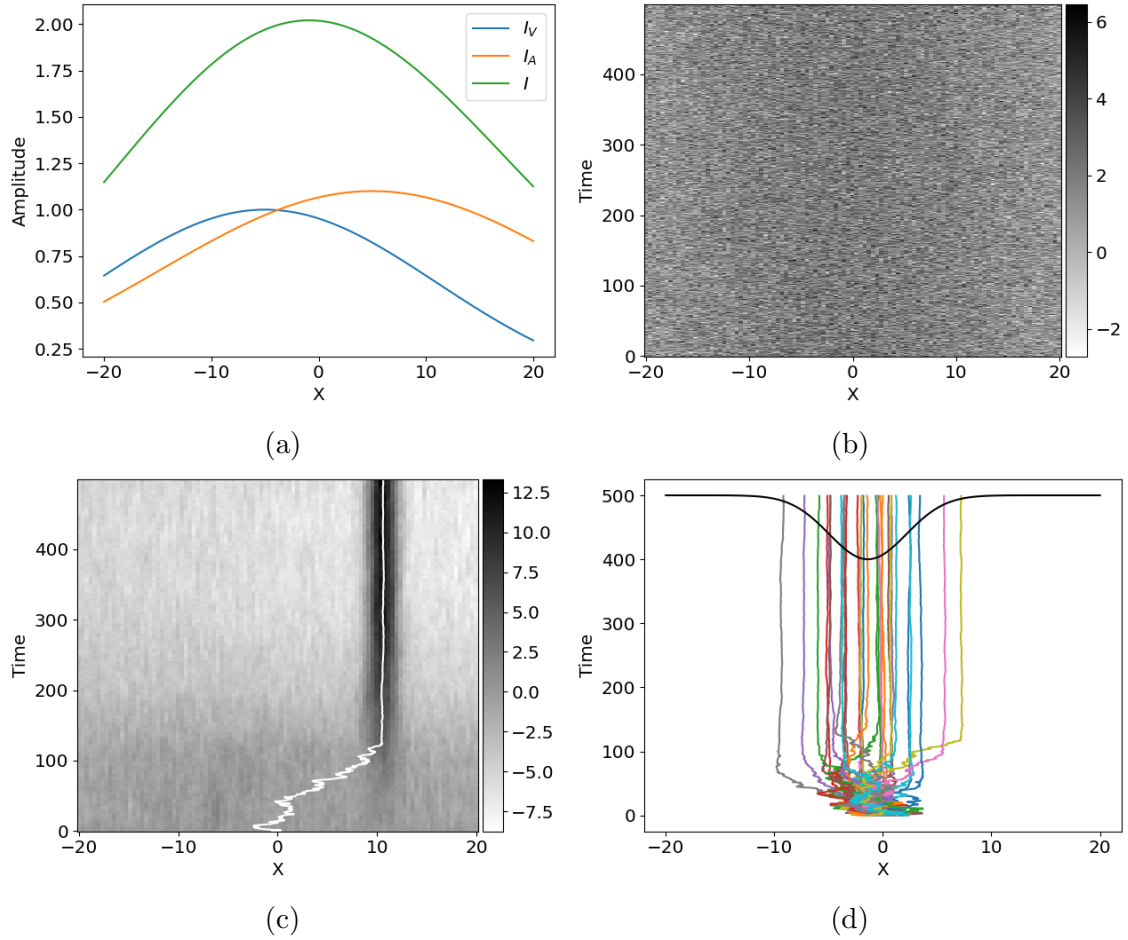


Figure 3.4: (a) Inputs $\Delta = -5^\circ$ and $\sigma_V = 16^\circ$. (b) Inputs summed with noise on the neural field (x) over time. (c) Field potential U during one single run. The white line shows the evolution of the barycentre of the field output $f(U)$. (d) Barycentres of the DNF output for the 30 runs of the model. The black line shows the approximate Gaussian distribution obtained with the mean and SD of the last 30 positions.

to fit, at best, the mean and standard deviation of the empirical data. The results in figure 3.5 show that we are competitive compared to the Bayesian modelling using maximum a posteriori estimates of localisation distributions, which remains the dominant paradigm for multisensory integration [93]. Moreover, unlike the Bayesian model, which uses unimodal performance to predict bimodal behaviour and relies on the hypothesis that the psychometric functions of visual and auditory stimuli are Gaussian cumulative distribution functions, our model was fitted directly to the bimodal scenarios without prior knowledge of the unimodal variances. We also verified that the DNF behaviour is robust to hyper-parametrisation (not shown here) and found that some non-linear combinations of hyper-parameters seem to give similar results. This opens the way to integrate complementary mechanisms into the architecture, such as saccades, to model active perception, while having some parametrisation margin to fit the empirical data at best.

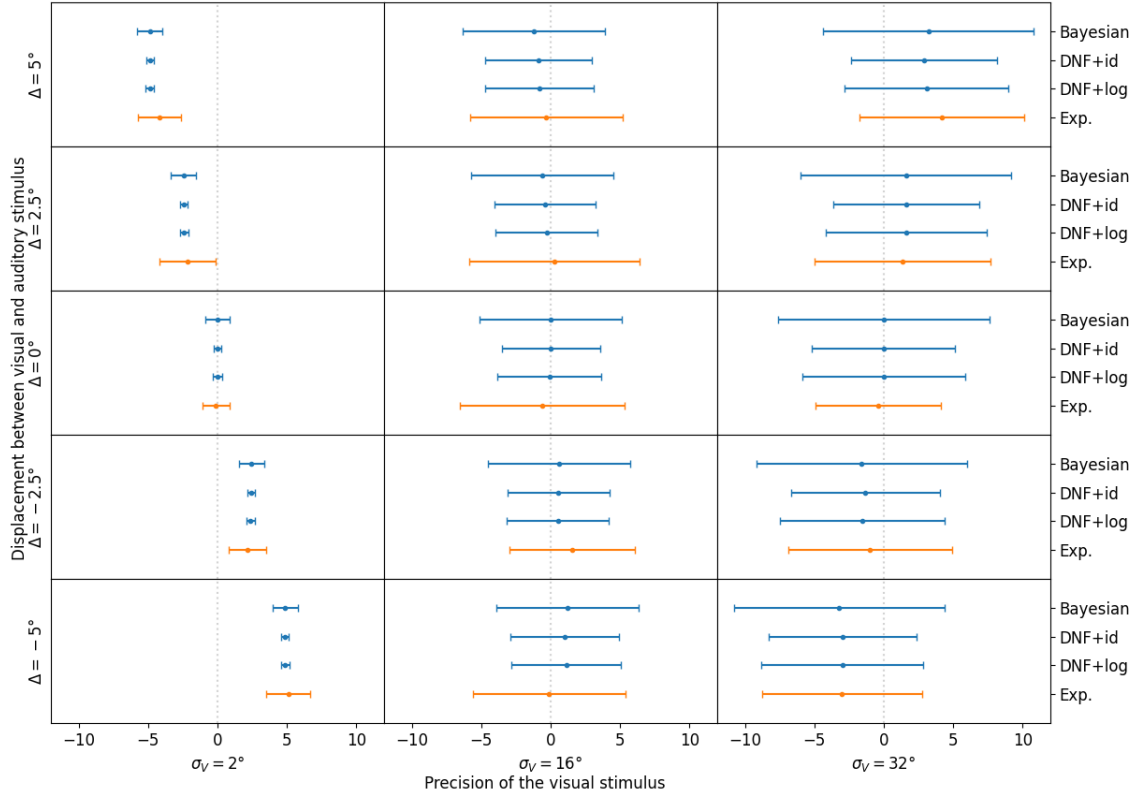


Figure 3.5: Experimental results of the bimodal presentation (orange intervals for empirical data) and corresponding model outputs (in blue). For each error bar, the centre dot represents the mean localisation, and the half-amplitude is the standard deviation.

This work was part of Simon Forest’s PhD thesis and led to an article in the journal *Neural Computation* [2], which can be found in the appendix 6.5 .

3.4 Multimodal perception with automatic weighting

3.4.1 Context and objectives

As we have shown in the previous section 3.3, humans are able to combine incongruent stimuli in a statistically optimal way, without the need for any conditioning (i.e., learning³⁷) phase. This is why Bayesian modelling achieves good performance, yet it does not explain how this can be achieved in the brain³⁸. Our hypothesis is that the density of sensors, or more generally the density of the representations, may somehow encode the relative weight of a modality in the fusion. This is what we expected to observe in the previous section by comparing an identity topology with a log-polar one, which has a

³⁷There is obviously the learning of relevant multisensory correlated features at some point, but there is no explicit learning dedicated to this task.

³⁸Nevertheless, there are articles in the literature defending the Bayesian brain hypothesis, i.e. that the brain performs Bayesian computation. However, at a computational level this is often intractable, *e.g.* with the active inference framework [98], and at a neural level it is often based on the hypothesis that the neural code uses Poisson coding [90], which is still the subject of much debate.

higher density in the centre rather than in the periphery (which in humans is related to the density of cone cells, which is higher in the foveal region). Unfortunately, there was no statistical difference between the two, but this may be due to the experimental setup, where the visual stimuli are not very eccentric and thus in a range where the difference between the two projections may be marginal³⁹.

Nevertheless we wanted to test whether such a mechanism of coupling spatial competition (with DNF) on topologies with different densities can provide to an artificial system with the ability to automatically weight the modalities in a consistent perception⁴⁰. Moreover, projecting the inputs onto low-dimensional topologies may help to overcome the limitation of DNF to scale to high-dimensional space [27]. This raises the question of the DNFs' ability to adapt to irregular manifolds while maintaining its decentralized decision-making properties. Indeed, DNFs are based on stochastic integro-differential equations, which are dynamical systems that can be sensitive to changes in their structure. Especially, the properties emerge mainly from the interaction induced by the kernel, which is symmetric for each unit and isotropic in the field when used on a regular lattice. Therefore, the vast majority of works using DNFs assume that the dynamics take place on a completely regular topology and no previous work, to our knowledge, has applied DNF to significantly unregular topologies.

3.4.2 Models and results

In these works, we first learn independent manifolds of the sensory space in each modality. For this purpose, we use the Growing Neural Gas model [99], which learns prototypes that are connected based on their co-activation, providing in the end a topology with a graph-like structure. We then created one multimodal graph containing all nodes and edges of each modality by connecting neurons of the different modalities that are co-activated by multimodal stimuli. This multimodal graph is then used as a support for the DNF to produce multimodal perception. The DNF uses the classical equation 3.1 except that the distance function between two units is changed to be the minimum distance in number of edges between two nodes in the multimodal graph (instead of the Euclidean distance as before). Note that this allows the computation to be completely modality-agnostic and to mix neurons that do not share a common coordinate system.

In our model, each neuron is tied to a specific modality, so its external inputs will be modality specific (although the rest of the DNF operations will not be). To ensure that the total amount of external stimulation is independent of the local resolution of a modality, we will rank all neurons of a modality by their Euclidean distance to the stimulus, and stimulate them in descending order of rank to divide the amount of activation between them. The output will be the barycentre of the activity $f(U)$ of the field. As there is no shared space where the positions of the GNG nodes can be averaged, we will rely on the input data to interpolate a corresponding position in the input spaces for each neuron to compute the results.

³⁹This is why we tested more eccentric stimuli in our experimental setup. The modelling has not yet been completed, so we cannot yet confirm or refute our hypothesis yet. However, preliminary results from data analysis in the active condition, where the participants are allowed to make a saccade (so the visual stimulus should be better perceived as then located in the foveal region), tend to indicate that the weight of the visual stimulus is indeed higher in this case.

⁴⁰Note that in this case it is the topology that will help the DNF to obtain new properties, whereas during my PhD thesis (see section 1.6.1) it was the DNF that provided the topological structure to learn the multimodal features.

Logpolar 2D vision and regular 2D audio. Related to the previous modelling in section 3.3, we tested a log-polar visual sensory system with an uniform audio one. More precisely, we take the coordinates of a visual stimulus in a regular 2D visual hemifield, and displace them following the log-polar transformation. The new 2D coordinates are used as inputs to the visual GNG. The audio is simply modelled as an uniform 2D space with the same range as vision. The learned bimodal graph is shown in figure 3.6. We are then interested in what a DNF would select when confronted to conflicting bimodal stimulus. We place two conflicting stimuli A and B at a common azimuth x , and elevations -5° and 5° respectively. Both stimuli can be seen and heard, but A (respectively B) is 20% more auditory (respectively visually) salient than B (respectively A). The results (figure 3.7) show two trends. First, we can see that B (visually stronger) is indeed selected more often than A at lower azimuths. Then A is preferred for higher azimuths. This illustrates that the DNF relies more on vision near the fovea, based on the relatively higher density of the GNG in this region. Secondly, we can see that the probability of A and B being merged increases with the azimuth, as the global density decreases, meaning that the two stimuli may be perceived as too blurred to be distinguishable. This phenomenon can be controlled by the size of the excitatory kernel which determines the number of neurons that are considered to encode similar inputs.

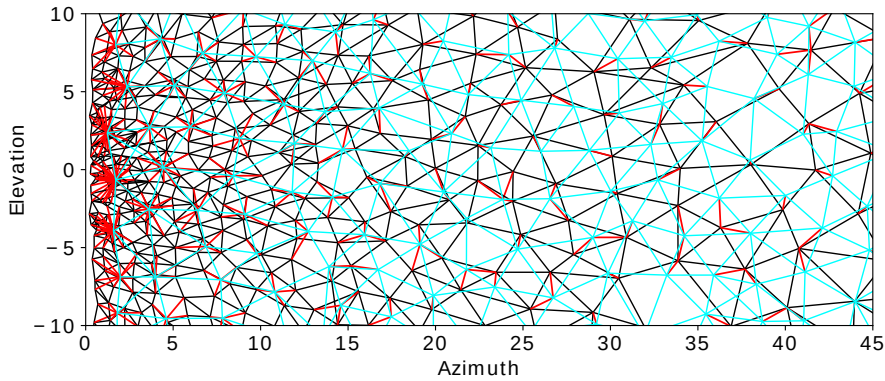


Figure 3.6: Sample representation of a bimodal graph with log-polar visio and uniform audio. Edges are coloured according to the modalities of the neurons they connect. Visual-visual: black. Auditory-auditory: cyan. Visual-auditory: red.

Regular 2D vision and HTRF 100D audio We will show that the qualitative properties of our model remain unchanged even with more complex sensory spaces. The auditory space is based on HRTF, a function that associates spectral features (caused by interferences on the signal by the head and pinnae) to source orientations that can be used in robotics [62]. Based on the data provided by [59], we computed 100-dimensional HRTF inputs corresponding to an external stimulus position in 2D. For vision, we used a uniform 2D space that had a smaller range than the auditory space. Like in the previous scenario, we tested the DNF with two stimuli A and B (see results on figure 3.8). This time they are separated both horizontally and vertically. Stimulus A has congruent auditory and visual components, while B is not audible but visually more salient by 1%.

In the visual-only graph, B largely prevails as expected, as B is more visibly salient. However, it is worth noting that the 1% difference between them matters even considering

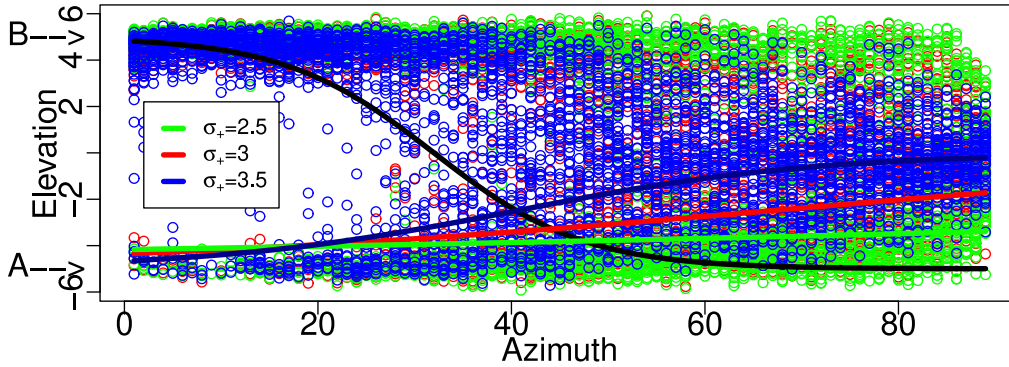


Figure 3.7: Statistical model of modality priority change (logistic regression in black) and stimulus merging (coloured logistic regressions). One point represents the barycentre of the output of one of the three differently parameterised DNFs (green: $\sigma_+ = 2.5$, red: $\sigma_+ = 3$, blue: $\sigma_+ = 3.5$), on one of the 50 randomised GNG, with two bimodal stimuli A and B at azimuth x and elevations $\pm 5^\circ$.

that the topology is not entirely regular. In the audio-only graph, A is trivially selected, but we can see some loss of precision in elevation: the barycentre is found 7° higher than the actual stimulus. This is very consistent with the GNG obtained (see figure 3.9), where the horizontal density is higher than the vertical one. In the bimodal graph, as expected, the audiovisual congruent stimulus A is selected over the visual-only stimulus B. But, the precision is improved, as the barycentre is also closer to the actual stimulus position than in the audio-only case, meaning that the better perception in the visual elevation had a positive impact.

Logpolar 2D vision and 100D HRTF audio projected onto 2D In order to test the limits of our model, we combined and extended the topology irregularities of the two previous cases. The vision is a log-polar 2D space, as in the first case. The audio is still a 100D HRTF, but instead of learning a GNG directly, we learned a Sliced Wasserstein Auto-Encoder [134] (SWAE), a representation learning algorithm whose 2D latent space is organised indirectly by minimising the Wasserstein distance⁴¹ between the distribution of the input representations and a normal one. To be compatible with our model, we then learn a GNG from the learned representations. We can observe that despite the regularisation of the SWAE the topology is much less organised than previously (see figure 3.10). The bimodal graph is learned as before.

To evaluate the ability of our model to deal with such irregular topologies, we measured the difference in localisation between the perceived stimulus and the real one (see figure 3.11). For this purpose, we presented a stimulus (purely visual or purely auditory or bimodal) at various regularly spaced azimuths (from -75 to 75) and elevations (from -30 to 30). In terms of visual precision, it is higher in the central region, which is quite logical as there are more units there. The opposite is true for audition. Although there is no specific reason for this regarding the inputs, the central region is the one that the SWAE struggled the most to represent (see figure 3.10), which ultimately affects performance. Bimodal performance lies between auditory and visual ones. As the DNF merges the perception of the two modalities, this is somewhat logical, even if we could have expected to

⁴¹More precisely, this is approximated by a sliced Wasserstein, hence the name of the model.

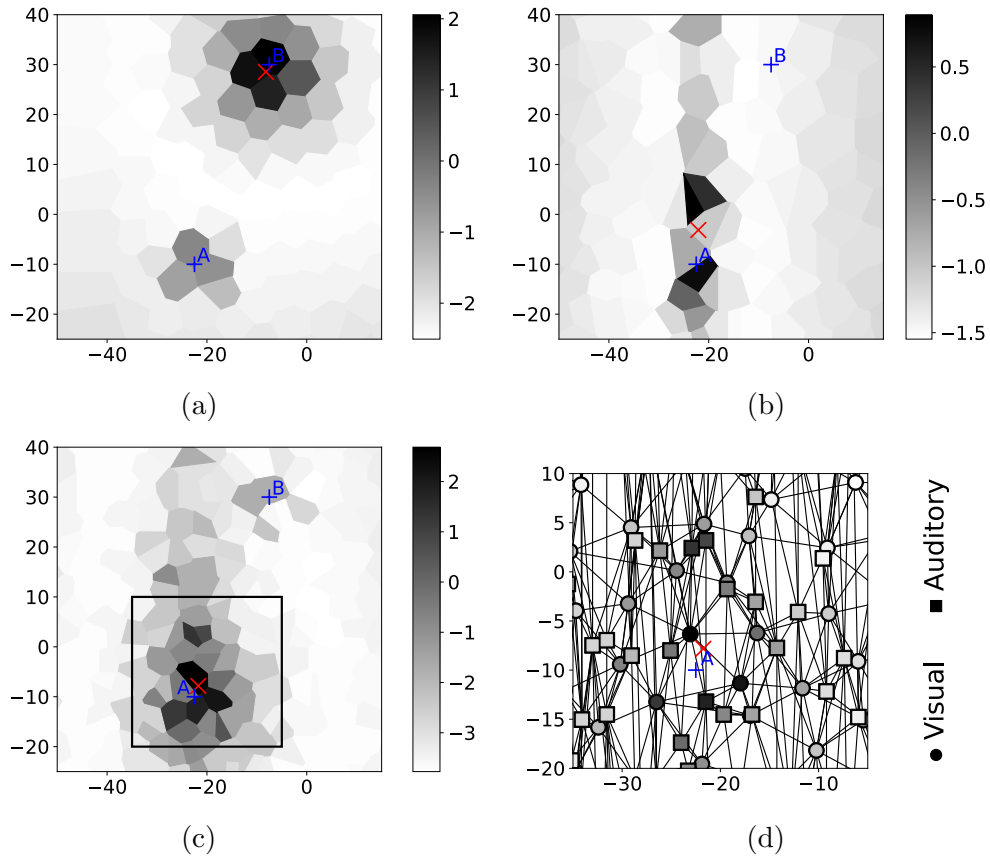


Figure 3.8: Shades of grey reflect the neuronal potential U at each node that is represented by a Voronoi cell in (a), (b) and (c). Red crosses indicate the barycentre of output activation $f(U)$ in the reconstructed 2D projection. (a) Visual-only neural gas with two stimuli located at A and B, where B is slightly more salient. (b) Auditory-only neural gas with only one input at A. (c) Bimodal neural gas with input as the sum of those used for (a) and (b). (d) Zoom on (c) around A, where all nodes and edges are shown.

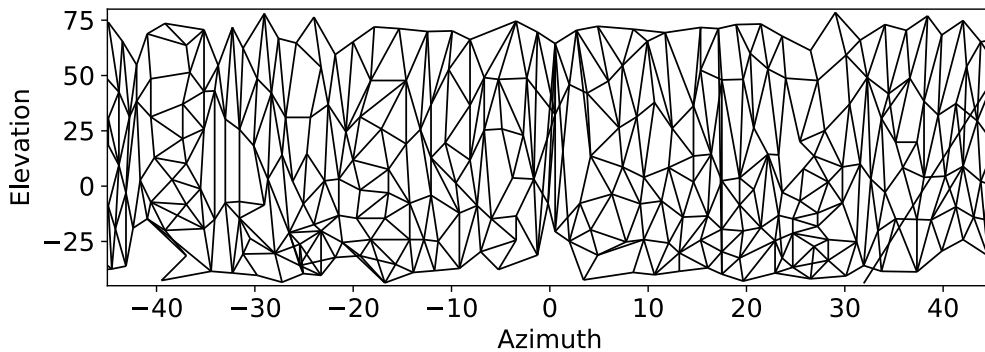


Figure 3.9: Sample of the auditory graph obtained from HRTF data. Note that the x -axis and y -axis have different scales.

be closer to the best performance (either the visual in the centre, or the auditory in the

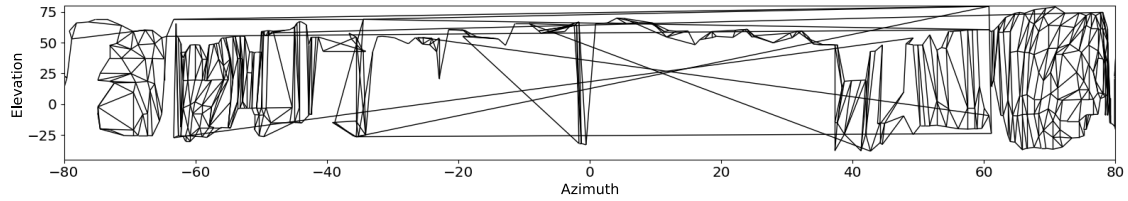


Figure 3.10: Sample of the auditory graph obtained from HRTF data when the GNG is trained from a SWAE with a 2D latent space. Note that the two axes have different scales.

extreme periphery). We probably face here the current limitations of our model with such an irregular (auditory) topology.

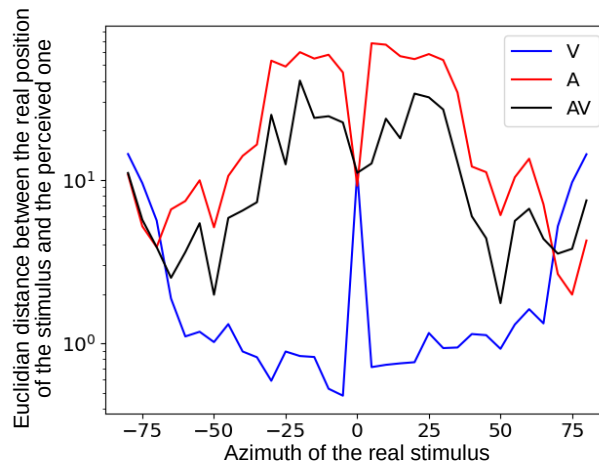


Figure 3.11: Euclidean distance (with a logarithmic scale) between the real and perceived position of the stimuli with visual (V), auditory (A) or bimodal (AV) topology. For each azimuth the distance is averaged over the 13 regularly spaced elevations.

This work was part of Simon Forest’s PhD thesis and led to two publications, one in IJCNN [7], which can be found in appendix 6.6, and one in CNAI [41].

3.5 Conclusion and perspectives

In this chapter, I have presented our work on the question of how to obtain relevant multimodal perception, especially regarding the automatic weighting of modalities, from structured, potentially learned, representations. We have chosen to consider this as a decision making process that has to be coupled with spatially organised options that can have different densities and thus reliability. More precisely we rely on DNFs, which are a neural network model providing distributed and dynamic fusion and competition properties. In section 3.2 we reviewed various decision-making algorithms, from psychophysics to robotics, to position the DNF within this panorama and to illustrate its versatile properties. We then used the DNF to model data from a psychophysical task of audio-visual merging in section 3.3. It allows to bridge the gap between some neuronal substrates

and the behavioural property of optimal combination of stimuli observed in humans. It thus provides an actual implementation of this mechanism whereas the classical Bayesian model is limited to a description at the population level. Finally in section 3.4 we adapted the DNF computation to irregular topologies. Thus, multimodal perception, tested on simulated but somehow realistic inputs, automatically relies on the modality that has the denser topology and consequently the higher precision.

Obviously there are multiple perspectives on these works.

Regarding the modelling part, the direct following is to test our model on the (spatial and temporal) data that we obtained in the AMPLIFIER project in our psychophysical experiment with more spatially distributed stimuli and with an active condition. This last aspect will be particularly interesting to model as it will involve areas with multiple visual densities (first the periphery then the centre after the saccade) with a dynamic setting of temporal accumulation of cues that could fully exploit the dynamic properties of the DNF. It would also be an opportunity to explore the combination of multisensory fusion with action selection within an active perception framework, possibly combined with dedicated multisensory attention mechanisms that are not yet fully understood in humans. Testing other psychophysical setups (other senses or different tasks, such as categorisation instead of localisation) and including other elements in the modelling to better match with the current neuroscientific knowledge of neuronal pathways would also be interesting research axes.

Much work remains to be done on combining learned topologies with DNF especially to validate this mechanism with more realistic inputs and complex topologies. This should require an adaptation of the representation learning and/or of the computation of the DNF. This is part of my research project, which I will detail in the section 5.2.

4 Incremental and active learning

4.1 Introduction

We have seen in the previous chapter 2 how to learn (visual) structured representations and in chapter 3 how to obtain multimodal perception as a competitive mechanism relying on spatially organised representations. If we step back to the context of autonomous agents (figure 1.2 in the introductory chapter), the next aspect to consider in the sensorimotor loop is the action to be taken. This is a less studied part of my research, yet I will develop it in the perspectives section 5.2. We have already considered action to some extent in the previous chapters. Indeed, we have shown that it can be useful to learn relevant representations, in relation with the sensorimotor theories, and that a decision-making algorithm can be the support for multimodal merging, while active perception may play a role in the process. This chapter will still focus on representation learning, but where action, through data modification, is one of the central research questions. This modification can be ambivalent, as by changing the distribution of the data the independent and identically distributed hypothesis on which most machine learning methods rely is then invalid and can thus provoke catastrophic forgetting [97], but at the same time the model can focus on more relevant inputs via active learning mechanisms to improve its performance.

Active learning is a vast area of research that study what action strategy a system can adopt to improve its absolute performance or reduce the time it takes to achieve it [178]. This can encompass, for instance, requesting of a label from an oracle for unlabelled data or selecting the next data to process. These two questions can also be intertwined when the system interacts with an environment, as the receipt of new data is often associated with some kind of label (*e.g.* in robotics when learning to move an arm, the sampled data is the motor action and the corresponding perceived position) or reward in reinforcement learning. Depending on the domain, the implementations and objectives of the active strategy may differ. For example, in supervised learning a classical goal is to reduce the uncertainty about the error made [178], while in reinforcement learning it is widely used as a pretext task to favour exploration or skill discovery [64], which is also the case in robotics where intrinsic motivation guides the robot towards learnable areas [163]. While part of my post doctoral work focused on coupling the intrinsic motivation mechanism with representation learning to improve performance and exploration time (see section 1.6.1), the works I will present here in section 4.2 use off-the-shelf active learning mechanisms to avoid learning spurious correlations, which can significantly affect classification performance (section 4.2.1) or to improve algorithmic learning, which is a hard problem for current deep learning architectures (section 4.2.2).

When considering autonomous agents, we want them to exhibit lifelong learning [146], i.e. the ability to adapt to unknown contexts, environments, tasks, etc., without requiring much human feedback or substantially forgetting previous knowledge. While desirable, this

is a rather broad, ambitious and somewhat fuzzy goal. We will focus here on continual learning, which still encompasses a wide variety of objectives, depending on what aspects of the dataset change and what information is provided during training and testing [204]. In increasing order of difficulty, the content of the batch can change (instance-incremental), which can also be called online learning. The domain of the input can also change over time (domain-incremental), which is somehow a problem of domain adaptation during training. Then the task (defining the labels) can also evolve. The identity of the task can be given during training and testing (task-incremental), only during training (class-incremental), or never given (task-free continual). Finally, data can arrive in a stream (online continual) and tasks may overlap (blurred boundary continual). Moreover, continual pre-training learning focuses more on transfer to downstream tasks. Overall, one of the main underlying research question is the stability-plasticity dilemma, as the model is expected to retain some performance on previous tasks/data while being able to adapt to changes. This can also be seen as a problem of generalisation across data, domains and tasks.

Within this panorama, we proposed a contribution to task-incremental learning⁴² with some aspects of online continual learning, which is one of the most difficult contexts. Moreover, as we were interested in what an autonomous agent may experience while evolving in unknown environments, this was done in an unsupervised setting, which is rarely studied in the literature. More specifically, examples from each class are presented in a sequential order and can then reappear. For this reason, we call this unsupervised class-incremental learning⁴³ problem. More precisely, the main research question we studied was how to structure the representation space to support the detection of new or reappearing classes (section 4.3).

4.2 Active learning

4.2.1 Biased datasets with spurious correlation

Context and objectives In supervised learning (which also includes SSL approaches with discriminative objectives that I presented in chapter 2), machine learning methods, and especially deep learning, tend to converge towards simple correlations between input and output. When these are not expected⁴⁴ (*e.g.* the model will rely on the blue background to recognise a bird) they are considered as spurious correlations and the models use them as shortcut learning [103]. In this work, we are interested in supervised learning in the context of biased image datasets, which is to be understood in the sense of (artificially) included spurious correlations within the dataset, and not in terms of any gender or skin colour bias, which are serious societal issues, that are addressed with more formal (in a

⁴²In our unsupervised context, the distinction between task-incremental, class-incremental and task-free continual somehow fades away.

⁴³This does not perfectly match the nomenclature of [204], that I reported here, where class-incremental would consider that labels are provided during training. However, these definitions were proposed after our work, and this name is aligned with the terminology used in [201], where they defined various kinds of tasks, with class-incremental as the one where the task is not given and has to be retrieved from the data, which is our use case.

⁴⁴From the perspective of an autonomous agent, any correlation that can perform the task can be considered a good and beneficial solution. The disappointment from the designer’s point of view may come either because the representations are not aligned with his/her (projected) ones, or because they are not relevant to solve other tasks that may not have been adequately incorporated in the data and/or the loss function.

mathematical and/or moral sense) approaches that define what a fair decision would be. In this case, the performance of classical deep learning methods may experience a huge drop [145] so that dedicated approaches have to be proposed.

As for continual learning (see section 4.3), models use three levers (data, regularisation/optimisation of the loss function, architecture), yet often mixed, to tackle this problem. Some models try to correct the data to at best remove the bias, which has to be characterised beforehand, directly at the pixel level [205] or using the image representation [145]. When knowing the kind of bias present in the data, multiple models add to their architecture a naive model that learns this bias. It can then be used to modify the distribution of the data to favour those that maximise the uncertainty in this naive classifier [147], or to act as a repeller regularisation term during the training of the real model [66, 88]. In the latter case, the naive classifier can be replaced by a dedicated loss that learns the bias [215]. If each image is labelled as biased or not, the loss can also be modified so that images with similar biases can be forced to be represented differently [198]. Finally, if nothing is known about the bias, a way is to rely on an indirect cue, such as the degree of confidence of the classification. For instance, RUBi [81] uses this indicator to retro-propagate a low gradient for examples with a high degree of confidence, which may correspond to the biased ones, as they are easier to classify. This last case is the one we were interested in, as it provides less information to the system, and we proposed a similar approach, but using an active learning mechanism to update the data distribution rather than changing the learning process.

Model and results Our Image Representation Avoiding Naive (ImRAN) learning protocol is rather simple (see algorithm 1). After each epoch, we evaluate the error of each example averaged over its augmented versions (line 5), to determine, proportionally to a predetermined maximum value and with a sliding average mechanism, the number of occurrences it will be present in the dataset for the next epochs (line 6). To avoid overfitting, the data are augmented with some Gaussian noise to add some variation (line 7). The underlying idea is that, when the model learns the bias, unbiased data will have a high error rate, so that increasing their number of occurrences will penalise the model more, and thus forcing it to learn the non-spurious correlations.

Algorithm 1 Pseudo code of ImRAN learning

\mathcal{D}_i : training dataset (tuples (input, label)) at epoch i , with \mathcal{D}_0 the initial one
 $n \in \mathbb{N}$: number of training epochs
 $K \in \mathbb{N}$: maximum number of duplications
 $C \in (0; 1]$: exponential average hyperparameter
 $\sigma \in (0; +\infty)$: the standard deviation of the Gaussian noise

- 1: **for** $i = 0, i < n, i++$ **do**
- 2: **learn**(\mathcal{D}_i)
- 3: $\mathcal{D}_{i+1} \leftarrow \emptyset$
- 4: **for all** $d \in \mathcal{D}_0$ **do**
- 5: $A \leftarrow \text{AllAugmentedExamples}(d, \mathcal{D}_i)$
- 6: $nbCopies \leftarrow \lceil (AvgErrorRate(A) \times (K - 1) + 1) \times C + |A| \times (1 - C) \rceil$
- 7: $\mathcal{D}_{i+1} \leftarrow \mathcal{D}_{i+1} \cup \{d + \mathcal{N}(0, \sigma)\}_{j \in [1, nbCopies]}$
- 8: **end for**
- 9: **end for**

We tested a classical CNN trained with or without ImRAN and compared it to state-of-the-art models on the Biased MNIST dataset [66]. The dataset is similar to MNIST except that the background is coloured with each of the 10 colours associated with a digit with some ρ probability (see figure 4.1). The results (see table 4.1) show that our method improves the vanilla CNN in all cases and is better than the other methods when the dataset is highly biased. We initially wondered whether the active learning mechanism may also increase the proportion of hard to impossible to learn examples, a phenomenon I observed during my post doc on another use case, which would have hindered performance. This do not appear to be the case, at least on this dataset, and we have to validate this result on harder ones.



Figure 4.1: Examples from the Biased MNIST dataset. Each digit is associated with a unique background colour with probability ρ and a random colour from other digits with probability $1 - \rho$, creating a spurious correlation (the colour) to predict the digit. When $\rho = 0.1$ the dataset is unbiased as the background colour is no longer correlated with the content.

ρ	Vanilla	LearnedMixIn [88]	RUBi [81]	ReBias [66]	ImRAN
0.999	10.4	12.1	13.7	22.7	44.2 ± 2.8
0.997	33.4	50.2	43.0	64.2	74.8 ± 2.2
0.995	72.1	78.2	90.4	76.0	82.7 ± 1.3
0.990	89.1	88.3	93.6	88.1	90.5 ± 0.5
0.100	99.2	54.6	99.3	99.3	99.0 ± 0.1

Table 4.1: Accuracy of various methods on the Biased MNIST dataset as a function of the fraction of spurious correlation.

This work was part of Alexandre Devillers’ PhD thesis and led to a publication at the CAp national conference [40], which can be found in the appendix 6.7 (in French) .

4.2.2 Algorithmic learning with a multi-task approach

Context and objectives Algorithmic learning may be a way to provide more adaptability in solving any task that can be expressed as a Turing machine. In theory, recurrent neural networks are already Turing complete [189], but they struggle with trainability problem, i.e. the ability to learn complex problems effectively and efficiently end-to-end. Dedicated architectures such as the Neural Turing Machine (NTM) [109], which adds specific mechanisms such as a memory and differentiable ways to read and write it, have been proposed, but also appear to be difficult to train in practice.

In this work, we have chosen to focus on arithmetic operations, and more specifically, the multi-digit multiplication of two decimal numbers as an illustrative problem. Yet being simpler than learning any algorithmic task, it also exhibits the problem of inferring long-term dependencies due to carry propagation, which affects the performance of deep learning models [185]. Multi-purpose large language models are able to handle some

mathematical operations, but this is only one of the evaluation tasks and the performance is highly correlated with the frequency of the terms in the pre-training corpus [176]. In the literature, some works have proposed dedicated architectures for arithmetic operations, such as the neural GPU [128], which shares similar ideas with NTM. Sometimes dedicated modules were considered such as in [85] that learns the hierarchical combination of single-digit pre-trained tasks using reinforcement learning, or in [200] that proposed Neural Arithmetic Logic Units that use specific activation functions such as log, exp, etc. to more easily extrapolate learning to large domains. Other works also considered representing the problem in an efficient way, for instance using a prefix notation [141].

These approaches often fail with multi-digit decimal multiplication (or do not explicitly target and evaluate it). Moreover, we were more interested in studying the training procedure itself. In a previous article [10] we proposed to decompose the arithmetic processing flow into successive 1-digit operations, so that the model computes and learns iteratively by feeding back its previous output, which can be considered as a kind of teacher forcing with external recurrence (since the network was not an RNN but an MLP). Here we proposed to learn the task end-to-end by combining a kind of multi-task learning (without an additional layer dedicated to each task) with an active learning strategy between these tasks.

Model and results In this work we considered the decimal multi-digit multiplication of two operands with at most n digits. It is composed of $n + 1$ sub-tasks: n single-digit multiplications and 1 final addition of the partial multiplications (see figure 4.2). We used a Seq2Seq model [196] (see figure 4.3), which is learned classically with teacher forcing from the inputs/output pairs of the current task (see figure 4.2b) chosen with an active learning mechanism similar to the one in section 4.2.1 to favour the learning of harder sub-tasks. More precisely, the probability of choosing a sub-task (among the set of all tasks) is a moving average, with a uniform probability between the sub-tasks by default, updated with the relative proportion of errors of each task.

	0023	(1)			
×	0048	(2)	Task	Inputs	Output
	0012	(3)	st1	(1//2)	(3//4)
	0184	(4)	st2	(1//2) (3//4)	(5//6)
	0010	(5)	st3	(1//2) (3//4) (5//6)	(7//8)
+	0920	(6)	e2e	(1//2) (4 empty lines)	(7//8)
	0110	(7)			
	1104	(8)			

(a) Lines (1) and (2) are the two operands. Lines (3) and (4) (respectively (5) and (6)) represent the carries and the result of the 1-digit multiplication of 8 (respectively 4) by 23. Lines (7) and (8) are the carries and the result of the addition of lines (4) and (6), i.e. the result of the global multiplication.

(b) Sub-tasks and end-to-end task associated with the multiplication of two 2-digit operands. Lines are read and written two at a time (“//”). For the end-to-end task, we added empty lines to make it distinguishable from sub-task 1 and to add some computational power to the network, via the recurrence.

Figure 4.2: Example of a multiplication of two 2-digit operands and its associated tasks.

We can see from the results (table 4.2) that our training procedure improves the performance of the network, especially for the 4-digit operands setting, where the carry propagation is the longer. Moreover, we can even increase the performance by fine-tuning the

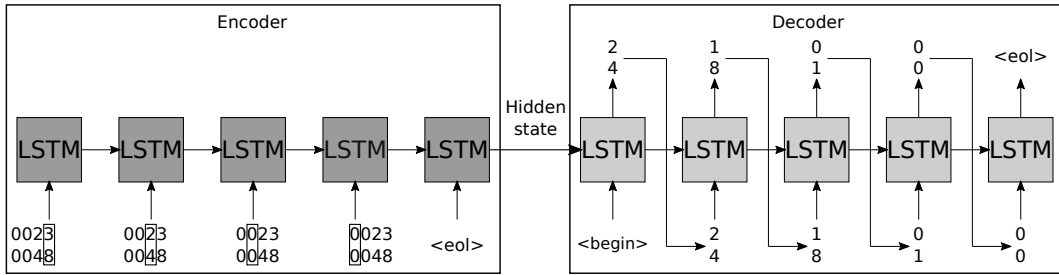


Figure 4.3: Seq2Seq architecture. The encoder receives successively from right to left the two digits (here boxed) of two lines (here the operands 23 and 48). From the encoder embedding, the decoder recurrently produces the digits, from right to left, of the next two lines (here the carry and the result of the first intermediate multiplication 8×23). In practice, each digit is encoded as a one-hot vector.

model on the end-to-end task only. Interestingly, this fine-tuning only works when the end-to-end task was also included in the set of tasks learned during pre-training. So, this seems to indicate that our multi-task active learning somehow favours the transfer from the sub-tasks to the end-to-end task. We also validated the robustness of these results to the number of empty lines given as complementary inputs for the end-to-end task (not shown here).

	3-digit operands	4-digit operands
trained on end-to-end only (Vanilla)	4.05% \pm 1.72%	92.42% \pm 3.64%
trained on sub-tasks & end-to-end	3.33% \pm 1.32%	23.34% \pm 14.69%
fine tuned + pre-trained on sub-tasks & end-to-end	–	6.68% \pm 4.31%
fine tuned + pre-trained on sub-tasks	–	73.31% \pm 11.10%

Table 4.2: Error rate on the end-to-end multiplication task over 10000 operations.

This work was part of Anthony Baccuet’s master thesis and led to a publication in the AIC workshop [28], which can be found in the appendix 6.8 .

4.3 Unsupervised class-incremental learning

4.3.1 Context and objectives

In this work, we were interested in continual learning, a domain where several approaches have been proposed, which can be grouped into different categories that are not mutually exclusive and can be combined [204]. First, some are based on mitigating the problem of data distribution changes. This can be obtained with replay-based mechanisms that complement the content of the batch with examples of previous tasks, either recorded or generated [174]. These approaches may be difficult to scale to large numbers of tasks as the amount of data to be recorded or generated increases, that is why some models propose to impose a fixed memory usage [60]. Representation-based approaches use SSL as a pretext or pre-training task, which indirectly mitigates the problems of distribution changes as

the SSL loss does not depend on the task [121]. Second, other models try to limit the catastrophic forgetting that results from classical learning methods. This can be achieved by some regularisation, typically limiting the changes in the weights [132], especially those that were most involved in the previous tasks. Optimisation-based approaches propose to modify the learning rule to include some (mathematical) constraints, for instance orthogonality [84] over the gradient landscape. Finally, some approaches consider changing the architecture, either by distributing the available resources among the different tasks [150] or by adding some free parameters [211] each time a new task arrives, which may also face some computational limitations when the number of tasks is high.

Most of works have been on supervised learning, where the main research question is how to learn relevant features that can evolve over time without forgetting previous ones. When considering an unsupervised setting, as we did, the other questions that arise are the ability to detect novelty, which often rely on the output scores of the classifier [105], but also to identify previously seen categories, which is somehow a (internal) classification problem. In the unsupervised continual learning literature we were particularly interested in the Continual Unsupervised Representation Learning (CURL) architecture [174], which is based on a Variational Auto-Encoder (VAE) architecture (as we used in section 2.2.1.2), except that the latent space is an incremental Gaussian Mixture Model (GMM) and some generative replay is used to mitigate forgetting. Our work studied how to couple the structured representation learned by this latter architecture with statistical methods applied to its latent space to detect new categories and recognise previous ones in order to better construct the GMM.

4.3.2 Model and results

The CURL architecture [174] uses an unsupervised ELBO loss for training, while the addition of new components is triggered when the log-likelihood of the input estimated with the loss is below a certain threshold. We proposed to replace this mechanism with some statistical tests. More precisely, we used a Hotelling t^2 test [120] to compare the empirical (assumed Gaussian) distribution of the representations of the inputs (computed on the batch) with that of each of the existing components (with stored mean and covariance). Based on the results of these tests, we either affect the batch to the best-matching component (and update its mean and covariance with a moving average) or create a component one if all tests are below some threshold⁴⁵ (with mean and covariance initialised with those of the batch) and train the model with a supervised ELBO loss⁴⁶. Thus, we tried to obtain one component per class, for an automatic discovery of the number of classes, whereas CURL’s initial aim was more to have the best modelling of the distribution in the latent space, thus allowing more components in the GMM. Moreover, we studied a preliminary version based on a Page-Hinckley test, which accumulates evidence of drift in the ELBO loss to detect the presence of new components, but without determining whether it is a category seen in the past or not. We proposed another alternative version of our model that combined the Page-Hinckley test to detect a new class, and used the Hotelling test to determine the correct component when necessary.

⁴⁵In practice, as in CURL, we used a buffering mechanism to store a certain number of batches to have enough representativeness before creating the component.

⁴⁶In their article, the authors of CURL have already proposed a supervised version of their model, for comparison purpose. This is the one we are based on, but with a label that is self-supervised by our statistical tests.

model	AMI	ARI	homogeneity	components	accuracy (batch)	accuracy (instance)
CURL [174]	0.6 ± 0.06	0.42 ± 0.09	0.66 ± 0.05	20.0 ± 2.2	0.92 ± 0.02	0.72 ± 0.04
SOINN ⁴⁷ [100]	0.54 ± 0.02	0.16 ± 0.01	0.82 ± 0.04	126.7 ± 9.4	1.0 ± 0.0	0.86 ± 0.04
STAM ⁴⁸ [192]	0.81 ± 0.01	0.73 ± 0.01	0.81 ± 0.01	11	0.89 ± 0.0	0.84 ± 0.01
Our HT	0.78 ± 0.01	0.77 ± 0.02	0.78 ± 0.01	11.0 ± 1.4	1.0 ± 0.0	0.89 ± 0.01
Our PH+HT	0.77 ± 0.01	0.75 ± 0.02	0.77 ± 0.01	10.0	1.0	0.88 ± 0.01

Table 4.3: Results on the MNIST dataset (averaged over 3 runs). We used metrics related to clustering (as our training is unsupervised) and accuracy by labelling our components afterwards with a majority vote in each component. Accuracy by batch is either a voting mechanism or the result of our statistical test for our models.

model	AMI	ARI	homogeneity	components	accuracy (batch)	accuracy (instance)
CURL [174]	0.47 ± 0.01	0.19 ± 0.02	0.62 ± 0.00	55 ± 5.0	0.9 ± 0.00	0.63 ± 0.01
SOINN [100]	0.45 ± 0.00	0.15 ± 0.01	0.63 ± 0.02	74.0 ± 8.5	0.9 ± 0.02	0.67 ± 0.01
STAM [192]	0.66 ± 0.01	0.48 ± 0.02	0.62 ± 0.01	9	0.74 ± 0.04	0.66 ± 0.02
Our HT	0.56 ± 0.01	0.39 ± 0.01	0.53 ± 0.00	21.3 ± 1.3	0.73 ± 0.05	0.61 ± 0.01
Our PH+HT	0.55 ± 0.02	0.39 ± 0.01	0.53 ± 0.02	11.7 ± 1.9	0.79 ± 0.09	0.62 ± 0.02

Table 4.4: Results on the Fashion-MNIST dataset (averaged over 3 runs).

We tested our model on the MNIST [144] and Fashion-MNIST [208] datasets. For both, during learning, we first presented batches of each of the 10 classes in random order (to test the ability to detect new classes), then presented the same 10 classes again in the same order (to test the ability to recognise previous classes). The results are shown in tables 4.3 and 4.4. On MNIST, both our models achieved competitive performances against the other models, especially when considering the automatic discovery of the number of classes. On Fashion-MNIST, the best accuracies were obtained by models that had too many components with respect to the actual number of classes. Our model, which combines the two tests, detected a number of classes close to the real one, while being competitive with STAM on the accuracy metrics. A direct perspective would be to extend the evaluation to other more challenging datasets, while mixing new and previously seen classes during learning.

This work was part of Ruiqi Dai’s PhD thesis and led to two publications, one in IC-TAI [8] and one in ICONIP [9], which can be found in the appendix 6.9 .

4.4 Conclusion and perspectives

In this chapter I presented some of my works on the temporal dimension of learning, which is essential when considering an agent interacting with its environment. More precisely I focused on the temporal evolution of the dataset, which is either controlled by active learning or has to be managed when facing incremental learning.

I showed that the use of active learning, by focusing on the examples that are not well mastered, can improve performance when facing a biased dataset to mitigate the learning of spurious correlations (section 4.2.1). The model is currently quite simple and can therefore be greatly improved, either in the way it detects interesting examples (*e.g.* based on the representations), or in the way it generates new examples (*e.g.* using generative techniques or augmentations). However, we did not really push in these directions, as some of these ideas were explored in [57], which was published shortly after ours.

Active learning can also help the transfer from sub-tasks to the end-to-end task of multi-digit multiplication (section 4.2.2). Future works will focus on understanding more precisely this dynamic of transfer between tasks during training, as the combination of tasks is a key element for algorithmic learning. Another perspective is to apply such a learning mechanism to other algorithmic procedures to study if it is applicable and generalisable to different operations, possibly with transformer architectures. Moreover, another interesting research question would be to automatically decompose the task into relevant sub-tasks, through trial and error and via learning by demonstration.

In section 4.3 we proposed the combination of simple statistical tests with a representation learning algorithm to detect new or reappearing categories in the context of unsupervised class-incremental learning, which is barely studied in the literature. Much work needs to be done to improve the overall performance, especially to deal more accurately with components drift, to be able to disentangle close classes, and to try to obtain these properties with a fixed computational footprint. In relation to my other research interests, the use of instance discrimination SSL methods to better structure the latent space (see chapter 2) or to use active learning strategies may also be interesting perspectives.

More globally, I would also like to study the coupling of (inter)action with representation to improve and structure learning, but also perception. This will partly be explored within the framework of sensorimotor theories (see the perspectives section 5.2 for more details).

5 Conclusion and perspectives

5.1 Conclusion

5.1.1 Summary

In this manuscript I presented my main contributions of the last 10 years as an associate professor in the SyCoSMA team. My research axes are in between cognitive science/neuroscience and artificial intelligence, with a clear focus on the latter. While my initial research work during my PhD was in computational neuroscience, bringing a computer scientist's view to some computational and learning mechanisms of cortical structures, I then gradually drifted towards machine learning, also related to computer vision. I still keep some links with neuroscience and cognitive science, and a stronger one with psychophysics, but I use them mainly as a source of inspiration to gather more evidence on brain processing to transfer them to artificial systems. My research topics are thus interdisciplinary, which is facilitated and supported by my (re)current collaborations with colleagues from other fields.

In chapter 2 I presented some works on representation learning of images using self-supervised learning (SSL) methods. First, I showed preliminary works on an approach to automatically construct semantically related pairs of proto-objects from a simple saliency-based mechanism for learning visual representations, and on a study dedicated to the role of action in the structure and learning of relevant representations when receiving visual glimpses from an image. Second, based on these ideas, we proposed a new module relying on equivariance, i.e. a transformation of the embedding that can be predicted from the augmentation applied to the image, that can be plugged into any state-of-the-art SSL architecture. By increasing the amount of information from the input that has to be preserved in the representations, as they have to be projected onto two spaces, one invariant and the other equivariant to the augmentation, it improves the classification performances and opens the way to more general and transferable features. Finally, we have systematically studied the structures of representation that emerge from instance discrimination methods. We empirically observed that they are organised in a way that favours discrimination between classes, with high intra-class similarities and low inter-class ones, while trained on a loss of instance discrimination. While this mechanism is not fully understood, these observations may help to improve SSL methods and open some research questions on the transferability of this structure to various downstream tasks.

In chapter 3, I presented a model for multisensory fusion with a decentralised competition/fusion mechanism based on dynamic neural fields (DNF), which relies on topologically organised representations of the sensory spaces. We first proposed an integrated framework to highlight the similarities of DNF with other decision-making algorithms from psychophysics to robotics, and illustrated its versatile spatio-temporal properties on

a benchmark of various scenarios. We then used it to model the audio-visual fusion of human participants in a psychophysical experiment of auditory localisation. They achieve similar results to classical state-of-the-art modelling based on Bayesian inference, while being more grounded in neuronal substrate. It also opens the way to modelling behaviour at the level of the participant rather than the one of the population, and could be made compatible with active perception mechanisms. Moreover, coupled with topology learning of the sensory space, we validated that the properties of the model are qualitatively preserved until the topology becomes too distorted. More interestingly, we showed that the importance of a signal within the fusion can rely on the relative density of the learned topology, which can be related to the density of the sensors. This mechanism has yet to be validated on our empirical psychophysical data. Nevertheless, it could be used to autonomously and online evaluate of the relevance of a sensor for multimodal perception in artificial agents.

Chapter 4 was dedicated to the temporal component of learning. In the first contributions, we proposed an active learning mechanism that increases the frequency of poorly classified examples/tasks in the training dataset for the next epochs. Although this mechanism is quite simple, it improves the classification performance when the dataset is highly biased by helping the model to avoid learning of spurious correlations. A similar mechanism, coupled with some kind of multi-task learning, improves the performance of the end-to-end learning of multiplication. Interestingly, it somehow induces a transfer from the learning of sub-operations to that of global multiplication. The second contribution is in the context of unsupervised class-incremental learning. We proposed to augment a variational auto-encoder architecture, which uses a mixture of Gaussians to model each component of the distribution, with statistical tests to detect new or recognise previously encountered components within the input stream. This allows to automatically discover the right number of presented categories while maintaining reasonable classification performance.

5.1.2 Integrated view

All these works can be regarded as various contributions to the improvement of machine learning methods, mainly related to unsupervised learning of structured representations. However, here I want to replace all these pieces of work altogether as a coherent puzzle within the broader framework of an autonomous agent interacting with an environment (presented in the introductory section 1.5), which is really driving my current and future researchs.

At a more applied level, to build such an agent, if we rely on the robotics community, we need sensors, actuators, and a cognitive system (to process signals, to take decision, to reason)⁴⁹. As a computer scientist, I will focus on this last part, leaving aside the hardware (which drives many other research questions in automatic control, mecatronics, robotics design, etc.), which includes (but is not limited to) the following elements and their associated research questions:

- a perceptual system. Chapter 2 targets to learn better visual representations that could be used to improve scene understanding. Moreover, chapter 3 focuses on

⁴⁹This is a slightly different way of grouping the various skills than the one proposed by Minsky in his definition of AI (reported in the introductory chapter 1) “perceptual learning, memory organization and critical reasoning”.

improving multisensory perception by considering the relevance of each signal within the fusion. Finally, chapter 4 considers incremental learning, which is the kind of issue that appears when considering a continuous interaction with an environment.

- a decision-making system. Multisensory perception, presented in chapter 3, is considered as a dynamic decision process between multiple incongruent choices. Chapter 4 also considers decision making with the aim of selecting the examples that would improve the learning of the representation, or when to create a new category.
- a cognitive/reasoning system. Although not directly addressed in my work yet, chapter 2 proposes to learn richer and more general representations, which can be a good building block for making sense of the surrounding environment. Moreover, by integrating augmentation into the representation, this can be made compatible with the learning of a world model, allowing predictive and anticipatory behaviours.

On a more fundamental level, actions play a crucial role in cognitive processes, as stated for example in the enactivist position mentioned in the introductory chapter 1 or in the sensorimotor contingencies theory [161]. Although this aspect is not (yet) deeply integrated into my research work, my contributions can also be analysed through this prism.

In chapter 2, we studied the question of integrating action into the representation with a simple architecture and use case that gave results in favour of this integration. Moreover, the augmentations used to construct pairs of related inputs in visual discrimination SSL methods can be, to some extent, conceived as the result of an agent's action on the environment [138]. We have thus also shown that using the action in the representation, via the equivariance module, helps to achieve better performance.

In the AMPLIFIER project, whose current results are presented in chapter 3, we aim to study how active perception influences multisensory fusion, which is an ongoing work. Considering the psychophysical experiment, we build a new protocol for the ventriloquist effect that has two conditions. The first is passive, where the participant has a fixed gaze, and the second is active, where he/she is allowed to make a visual saccade. The research hypothesis we want to test, in line with the work presented here, is that after the saccade the sensory precision of the visual input will improve (as the stimulus will be located closer to the fovea, where the density of sensors is higher) and thus its weight in the fusion will also increase. Preliminary statistical analyses of the data seem to confirm this hypothesis. On the modelling side, we want to incorporate a new part that will decide where and when to saccade, for the model to fit the empirical data on fusion localisation but also on saccadic position and timing. This module will directly use dynamic neural field activity, as it has already been shown in the literature that such a system can model the various dynamics of saccadic movements [170], although not in a multisensory context.

In chapter 4, I presented an active learning mechanism that can be interpreted as choosing an action (picking the right sample from the dataset) to improve the learning of representations. This kind of mechanism can also be applied when there is an environment to explore, as it was the case during my post doc [17]. In this case, the better action may not be to pick examples with a high prediction error, but rather those with a high decrease in prediction error. Indeed as some part of the environment can be unpredictable, a high error can reflect either that the model has not yet learned properly or that there is nothing to learn here. Such a mechanism is often called intrinsic motivation, a concept related to developmental psychology [164].

To summarise, on an applied level, all the works presented could be coupled as pieces required for an autonomous agent, yet a lot is missing, opening new research questions beyond the practical integration of these various models. On a theoretical level, my research is aligned with sensorimotor theories of cognition, which raises the question of the potential benefits of more deeply integrating their principles into models of multisensory learning and perception of the environment. All of this is developed in more detail in the perspectives section 5.2.

5.2 Future works

5.2.1 Positioning

Recent advances in AI have been made possible thanks to the increase in the size of datasets (either with larger labelled datasets, or with architectures and training techniques that can exploit the vast amount of unlabelled data available on the Internet) and the increase in computing power, which, combined with dedicated deep learning development frameworks, allows large models to be run in a reasonable amount of time. While AI models achieve superhuman performance in many well-defined domains (Go, Atari games, traffic sign recognition, text translation, etc.), they are clearly very far away from human or most animal abilities in terms of adaptability, robustness, common sense, theory of mind, etc., not to mention fine motor control skills.

Some researchers are deeply convinced that we have to continue to push in this direction towards larger datasets and larger models to unlock (some of) these missing properties. This has been the case to some extent in natural language processing, where the large language models (LLMs) achieve high levels of reasoning and understanding.⁵⁰ However, although the larger models are still about 3 orders of magnitude smaller than the human brain, they are larger than the brains of many animals and have access to an amount of data (on the specific task it was trained on) that is certainly much larger than what humans have access to.⁵¹ Moreover, relying on datasets or artificial environments (with the reality gap problem well known in robotics) calls into question the theoretical possibility of obtaining representative data for any task in any context, especially if we want our model to have some degree of autonomy in defining its own goals, or to be able to access some intrinsic properties of objects, such as weight or softness, that require manipulation [160]. Thus, I argue that there is (at least) one key element missing to really push forward the performance of artificial intelligence, at least if we target to obtain it with computational and data resources on the same scale as those used by animal and humans. As stated in [114], there is “a fundamental misalignment between human and typical AI representations: while the former are grounded in rich sensorimotor experience, the latter are typically passive and limited to a few modalities such as vision and text”.

Based on my research interests and academic career, I propose to try to overcome this limitation by studying how we can draw inspiration from the literature on how humans and animals make sense of the world. When studying how human babies learn, we can observe that they explore their environment through actions that shape the multimodal experience of their embodied cognition [193]. More generally, actions have a fundamental

⁵⁰I confess that I would not have bet on such success “only” by pushing statistical learning on textual datasets, and that it would have required another ingredient in the way language processing is conceived.

⁵¹This was the same for Do as AlphaGo zero was trained on 4.9 million games [190], which should have taken more than 500 years for a human playing 13 hours a day with a game duration of 30 minutes.

place in the learning of representation. One of the most well known illustrations of this key role is the kitten carousel [116]. In this experiment, two kittens experience the same visual stream by being attached to either side of a carousel in a simple environment composed of alternating vertical white and black lines. However, one kitten controls the displacement of the carousel axis by walking, while the other goes passively with the flow. The kittens were placed in this environment for 3 hours a day for 8 weeks. At the end, the passive kittens failed simple visual tests, while the active ones passed them as normal.

In psychology, the fundamental intertwining of action and perception dates back to William James' ideomotor theory (where actions are represented by their perceptible effects), Piaget's theory of sensorimotor development [167] (where skills are progressively constructed through interacting with the environment), as well as Gibson's theory of affordance [106]. In the latter, an object is defined not by a set of perceptual properties (as green leaves and a brown trunk for a tree *e.g.*), but by the potential ways of interacting with it that are elicited in the agent (sitting down to read and have a shadow for a human, resting or getting food for a bird, hiding for a cat, etc.). These ideas have been extended in both the interactivist [72] and active inference [98] frameworks. In my research project, I am more interested into the SensoriMotor Contingencies Theory (SMCT) [161], which combines coherent pieces of evidence from neuroscience and psychology into a unified framework with some implementable statements. The key claims are about:

- regularities, with Sensory Motor Contingencies (SMCs) defined as “the structure of the rules governing the sensory changes produced by various motor actions” [161]
- active perception as the “organism's exploration of the environment that is mediated by knowledge of SMCs” [159].

This may sound very close to classical concepts of pattern/feature learning, active learning/perception and the reinforcement learning framework. However, a radical conception of SMCT is a paradigm shift, as instead of looking for the right representation to act, we have to look for the right action to perceive within a dynamic interaction with the environment. To try to make the difference clear, let me rephrase an example given by K. O'Regan, who proposed SMCT. In traditional AI, the concept of a straight line is learned as the co-activation of the visual sensors aligned with this line, whereas in SMCT it is defined as the invariance of the change in all visual sensors due to a movement in a direction following this line. Some of these concepts have already been successfully applied to some extent in the field of machine learning. For example, learning sensorimotor correlations improves object recognition and manipulation in robotics [75]. Moreover, models more strongly inspired by SMCT can learn complex concepts such as containment [114] or space [139].

5.2.2 Research axes

Although my future research may not strictly adhere to the SMCT paradigm, I really want to study how considering action as a core element can influence the way to learn and improve multimodal representation learning and perception. I will thereafter outline the research questions that I would like to study over the next five years. They are structured around the three axes that I have presented through this manuscript, but they are obviously interconnected and each axis will feed and be fed by the results of the others. I am very fortunate that almost all of them are already funded, thanks to the support

of university of Lyon 1 (with the acceptance of a SENS project, a local call to support the emergence of new research talents during the first 10 years of their career, for a PhD student), of École Centrale Lyon and the InfoMaths doctoral school of the university of Lyon (for a doctoral contract), of the Auvergne-Rhône-Alpes region (with the acceptance of a R&D BOOSTER project, which supports collaborative projects between companies and universities to develop new services with a high TRL, for 2×12 months of post doc) and of the ANR (with the acceptance of a PRC project⁵², that I am leading, with partners from the university of Lyon 1, Clermont-Ferrand and Grenoble, see figure 5.1 for an overview). I would like here to thank them all for their confidence in my research projects, and all my (future) collaborators for bringing their expertise in their respective research fields to these projects.

5.2.2.1 Representation learning

In line with the research trend to add some kind of sensitivity to augmentation within visual instance discrimination SSL methods, I presented in chapter 2 our own equivariance module to improve classification performance, which will be the basis for these perspectives.

How to obtain more adaptive representations. In the R&D BOOSTER project, we will study how equivariant self-supervised learning can help to learn more general representations in the context of industrial data. These datasets have the particularity of containing classes that are quite different from those in classical machine learning, focusing on very specific and application-dependent content. This will require to validate and improve the transfer learning capabilities of the SSL models. Moreover, the number of labelled data is small, so it will also be necessary to achieve good performances with few shot learning. In additionn, more structured representation learning also opens the way to obtain relevant data visualisation to support assisted labelling for massive data, a tool already proposed by the industrial partner of this project. Thus, we will study how equivariant methods can contribute to the structure of the visualisation to improve this annotation process.

The following research questions are more related to the question of how to consider and integrate the action more deeply in the representation learning process.

How to learn from a stream of inputs. Instead of having static inputs that can be modified by any augmentation, as in our previous work, we will consider an agent interacting with a simulated environment containing various objects. Thus, the model will receive a stream of inputs from which we will have to decide how to make the pairs. This could be seen as a challenge, but in practice many articles in the literature show that we can rely on temporal proximity as a guide, possibly combined with saliency-based mechanisms as we did in section 2.2.1.1. Indeed, it is an efficient way to define a pretext task for learning representations of objects being manipulated [65] or for reinforcement learning [104]. Thus, a simulated environment provide an opportunity to study richer sources of modification of the input, such as access to the multiple views of an object, which is not really possible from a single image. Much work remains to be done to determine the right environment, set of actions and models to learn more generic and robust representations. This will be

⁵²projet.liris.cnrs.fr/mesmrise/index.html

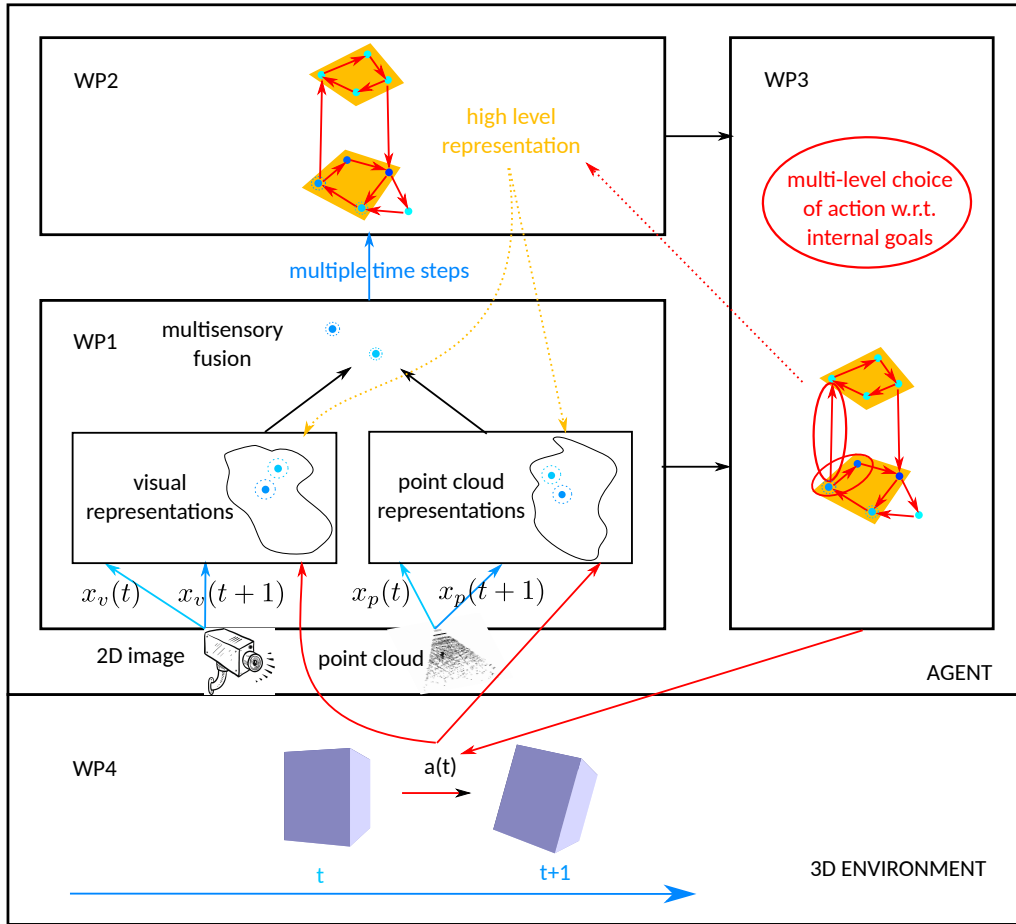


Figure 5.1: Overall structure of my ANR PRC project MeSMRise (Multimodal deep SensoriMotor Representation learning). We want to study whether using action as a unifying key point in learning will guide towards more generic self-supervised architectures and representations. Moreover, having more human-like perceptual and learning mechanisms should help to generalise better and be more robust to various environments. By interacting with the environment to have access to all its dynamics and properties, we target to make the agent more adaptive by finding the relevant information at the right time. WP1 (one PhD student) considers learning unimodal and multimodal representation and perception based on architectures that integrate action as the core of their representation, in line with SMCT.

WP2 (one PhD student) focuses on learning higher level representations based on the dynamics of interaction with the environment. Moreover, these representations will also support learning in WP1.

From multi-level and multimodal representations, WP3 (24 months post doctoral position) will define a hierarchical policy for exploring the environment to improve learning and perception in WP1&2 .

WP4 (12 months engineer) defines the shared evaluation environment, consisting of a simulated world filled with objects that the agent can manipulate.

explored in work package (WP) 1 of my ANR project. It will also allow to interactively test some hypotheses for improving representation or perception, as described later in the section 5.2.2.3 of the perspectives.

How to encode action in the representations. Action can be introduced into the representations in various ways, *e.g.* by structuring a visual servoing space with metric learning [95]. So far, we have done this with an indirect drive as an additional loss function. I would like to study how to favor more strictly this prediction objective and whether it can help to learn better features. Our preliminary results using only the equivariance loss with EquiMod (see section 2.2.2) seem to indicate that it will not improve the classification performance. However, we can hope that it will help to generalise better and to tend towards a world model. There are also theoretical insights suggesting that learning the prediction within the latent space can give better properties, especially regarding the complexity and dimensionality of the problem, as defended by Y. LeCun with his Joint-Embedding Predictive Architecture [101]. We will also study how to encode this prediction. While not exhaustive, this could be achieved classically as the embedding of the predicted next input, as the combination of basis functions as a form of efficient coding that is observed in the brain [70], or as the difference to apply to the current input, which is related to the predictive coding also observed in the brain [123]. The latter would be more in line with the first principle of SMCT. The difference between these coding schemes will be investigated in the SENS project.

How to learn more abstract representations. The stream of inputs should provide successive pieces of information about the same object present in the environment. It would then be relevant to be able to combine these various views into a high-level representation. From a fundamental point of view, we can rely on the definition of an object as a stable network of predictive sensorimotor interactions rather than a set of perceptual properties [194]. In this framework, a cup will not be represented as a cylinder with a handle, but as a structure encoding that the agent will see a cylinder and when rotating it, it will then see a cylinder with a handle, and vice versa. This is consistent with the principles of SMCT, where it is the ability to perform actions and to predict (the structure of) their consequences on the inputs that defines the perceptual concepts. From a practical point of view, this combination of learning spatial features with their temporal evolution induced by the action may help to cluster the inputs to better detect new or reappearing objects, for the emergence of more relevant concepts in an unsupervised way.

This clustering can also help to create pairs of relevant inputs at lower levels to support the learning of spatial representations. This will form a loop: the spatial representations will support the spatio-temporal ones, which in return will help to define better spatial representations. Fundamentally, this will raise the research question of detecting meaningful boundaries in sensorimotor flow, which is related to the bootstrap problem [137]. All this will be studied in WP2 of my ANR project.

5.2.2.2 Multimodal learning and perception

I want to study how to structure a multimodal space in order to obtain general and robust representations that allow the automatic weighting of each modality signal in the perception of an autonomous agent. Especially I want to study the following research questions, which will be explored in WP1 of my ANR project.

How to make the multimodal space and the decision-making process of fusion compatible.

In chapter 3 I presented how to adapt dynamical neural fields (DNF), a decentralised and cortico-inspired decision-making model, to multisensory perception with irregular topologies. However, this was still a simple topology learning method with prototypes, and I want to explore whether this kind of model can be coupled with deep representation learning approaches. Theoretically, DNF is an integro-differential equation which is a complex mathematical object, whose formal solutions are undetermined in a space with more than one dimension. More technically, DNF have been studied mainly with low-dimensional topologies, yet there exist some optimisation technics for regular high-dimensional spaces [169]. Moreover, it defines a dynamical system that requires a fine shaping of its phase diagram in order to obtain and guarantee, to some extent, interesting properties. More precisely, the behaviour of the DNF is tightly related to the interaction between the lateral kernel and the input topology. We can then try to adapt the shape of the kernel, but this is a difficult task as stated in the literature [27]. For these reasons, the main focus will be on trying to obtain structured representation spaces, or learned low-dimensional projections of them, possibly at multiple levels, that can fit with the DNF. We could also propose alternative fusion mechanisms, as DNF is part of a large family of decision-making processes, as non-exhaustively presented in section 3.2, that would conduct to less rich dynamics but could ensure the expected properties in a more reliable and flexible way.

How the action can structure multimodal learning. An agent interacting with an environment may not always receive relevant information from all modalities. Therefore, I want to study model architectures that have representations at the level of each modality simultaneously with the multimodal one. To this end, I have already proposed some architectural principles during my PhD with the alignment of unimodal spaces [23], or during my post doc [22] with the learning of unimodal spaces predictable by the others. Moreover, these unimodal and multimodal representations may be more generic as [92] recently shows that by projecting each unimodal space in a global workspace that allows to predict back the unimodal embedding, the learned representations transfer better than the features learned by CLIP [172], a classical contrastive architecture that is limited to the information shared by text and image modalities. In this context, I want to study how action can be integrated into such an architecture. Action can have different effects on various modalities. For example, when considering an object seen via 2D images and 3D point clouds (which is the use case for my ANR project), a rotation will affect both inputs, while a change in illumination will only affect the 2D image. This has to be taken into account in the way the learned multimodal representations are structured to allow these one-to-many projections. Moreover, relying on SMCT, learning the relationship between the changes induced by the action in the inputs of the various modalities would lead to a different structure than the usual statistical linking of the co-occurrences in the modalities, which could exhibit better properties. This perspective will be closely related to the one on the integration of action in representation presented in section 5.2.2.1.

5.2.2.3 Active learning and perception

The choice to perform a specific action can be aimed at obtaining more information from the environment to improve perceptual performance (active perception), or at orienting the agent towards some objective, such as learning a better model (active learning), a spectrum that is integrated in the active inference framework [98]. Without trying to fit

formally into this framework, this can be a support for the balance between active learning and active perception, the two axes that will be studied in WP3 of my ANR project, within a reinforcement learning framework.

Active perception. It allows dimensionality reduction (by processing only relevant information at the sensorimotor level) and alleviates framing problems (*e.g.* by fixating on an object of interest) [67]. This mechanism can be used to obtain more information from the environment to help desambiguate between different concepts. For example, relying on the high-level representation of objects presented in section 5.2.2.1, we can choose to rotate the perceived previously unseen cylinder to test whether a handle will then be seen to distinguish between a glass and a cup. This will also raise the question of keeping this accumulation of evidences in the agent’s perception over time. We also want to study how active perception can be combined with multimodal inputs. One objective would be to determine which modality is the more relevant for desambiguating a perception (which can in return change the weights in the multimodal fusion) or to fulfill a specific task, and then choose the action that will get the most new information from that modality.

Active learning. In the absence of an explicit external task, intrinsic motivation can provide a drive towards regions of interest, *e.g.* those that lead to learning progress [164]. In particular, we want to explore the dynamics between this kind of exploration incentive and representation learning. Here again, we will face a bootstrap problem, as the representation will help determine what action to perform next, which in return will provide new examples to learn relevant features from. This biases the received inputs by disturbing the probability distribution of the examples, which can lead to catastrophic forgetting [97]⁵³. While we will not explicitly consider this question in the ANR project, relying on classical techniques such as replay buffers, I also would like to explore this question of incremental learning as well. More specifically the combination of self-supervised learning and incremental learning has not been studied extensively [153], but the genericity of SSL may help to learn more robust representations. Moreover, I want to study whether the structure of the representation can help to detect outliers, which may be the cue for the arrival of a new class.

5.2.2.4 Datasets and models

Since recent advances in machine learning are partly due to the increase in the number of architectural parameters and the size of datasets (the use of self-supervised learning is especially related to the use of large unlabelled datasets), these two aspects may have an impact on, or at least define a framework for, the perspectives outlined above. Here are some considerations on these aspects.

Datasets If interacting with an environment is indeed a key to more generic, adaptive and robust representations, a lot of effort from the research community will have to be dedicated to creating rich and optimised interacting environments to allow a large number of interactions in a reasonable amount of time. In the longer term, the community may also face the reality gap well known to roboticists, which will raise a lot of research questions to try to tighten it. In the meantime, in WP4 of my ANR project, we will rely on and

⁵³Note that we can also use active learning to debias the inputs, as we proposed in chapter 4

adapt existing platforms dedicated to reinforcement learning environments (which face similar questions to ours) and datasets of objects. Another option will be to perform pre-training with datasets containing action, before fine-tuning and adapting the model in the interaction environment. Such datasets can be recorded offline in the simulator, raising the question of which action policy to use. But we may also rely on videos. It has been recently shown that learning from 1 hour of video can be quite as efficient as learning from the 1 million images in ImageNet [203]. While this requires automatic detection of the objects in the image (as an image in video is much less structured than one from ImageNet), it was obtained with the model itself while learning. Although this model does not take action into account, another recent work [186] learned a world model by inferring an embedding of the displacement from the successive images and then linking this embedding to a real action from demonstrations. However, this method seems to suffer from the problem of discriminating between self-induced and external movements (which were not so much present in their videos), which is a fundamental problem that can be more easily addressed by actually interacting with an environment and testing actions under different conditions.

Related to learning representations from videos, I will also co-supervise a PhD student on learning representations of skeleton animations to be able to generate new ones while applying some style to them. In this field, the datasets are quite small, because the recording of motion capture data is expensive or is done by private companies in the video game or in the animation film industries. To try to overcome this issue, we will rely on methods based on machine learning that can extract the skeleton from videos of moving people to obtain large datasets and apply self-supervised learning methods to them.

Models So far in my research, I have favoured the use of models that are as simple as possible. This can be illustrated by the choice of GNG, a simple topology learning method, to study the ability of a DNF to adapt to unregular topologies (see section 3.4). It helps to better understand and identify the properties of the whole model. Moreover, the default use of a large model as a transformer may raise some epistemological considerations. Indeed, the choice of the model should come after the formalisation of the scientific question, otherwise the model becomes (part of) the empirical object of study [199]. Nevertheless, identifying the properties of such large models is also an interesting field, and I want to explore it as well, *e.g.* how transformer can learn representation from videos, as presented above.

Using simpler models will also help to limit the computational resources required, which has an environmental footprint. In this respect, in the R&D BOOSTER project, we will study how reducing the size of the model, *e.g.* through quantization techniques or by pruning some connections or layers, affects the quality of the learned representations. As the main focus will be on the genericity of the representations and their ability to transfer to various industrial datasets, we can hope that even if the accuracy sometimes drops, the overall performance over all measures and datasets will not be seriously affected while drastically limiting the number of parameters to be learned.

5.3 Impacts of my research

As the university of Lyon 1 requires 2 pages of this manuscript to be in French, this last section will thus be in French. Here is a brief introductory summary and disclaimer for non-

French speaking readers. The research process is about applying scientific methodology to raise questions and research hypotheses, and to evaluate them correctly. But it is not isolated from the rest of the world, as it can have an impact on academic people (students, colleagues, scientific community, etc.), but also on society (especially as AI is currently being deployed massively) and its underlying structuring, which is related to a technocentric ideology, and more globally on the environment, as IT technologies have a significant impact on the overall carbon footprint of humanity, with AI taking an increasing share, even if a precise attribution remains difficult [127]. I do not claim to propose any new or scientifically relevant ideas or research questions in these areas, as I in no way pretend to be a sociologist or philosopher (though I hope to have more opportunities to work with colleagues from these disciplines in the future). But since I will (hopefully) soon be allowed to conduct my own research, and am already co-supervising PhD/master students and leading research projects, I consider that I need to attest my ability to master and transmit the scientific method and content of my discipline (and I hope that what you have read so far of my manuscript has convinced you of this) but also to apply a critical thinking (and ethical considerations) about my research works. This section will then be a sort of impact and benefits section, as it is now classically required when responding to a project proposal, extended to my whole research and also including some personal thoughts and wonders as a computer scientist associate professor in AI about his work within his discipline and research domain.

En épistémologie, la définition d'une science peut s'appuyer sur le principe de réfutabilité [107]. Une théorie doit ainsi définir des prédictions qui peuvent alors être vérifiées et éventuellement invalidées, ce qui la rend fausse le cas échéant. Une théorie est donc vraie dans le sens où elle est la plus crédible jusqu'à preuve du contraire ou potentiellement à défaut d'autres théories alternatives lorsque certains indices semblent indiquer une faillibilité de la théorie courante comme par exemple avec les orbites de Mercure pour la théorie de la gravitation de Newton. Les principes de reproductibilité et de répétabilité sont ainsi essentiels au processus d'évaluation des théories. L'application de cette méthodologie, pouvant être réalisée par n'importe qui et sur n'importe quel aspect réfutable, peut alors se concevoir comme porteuse d'une certaine forme d'objectivité. Ainsi mes travaux n'auraient pas d'impact en soi, ce serait leur application éventuelle qui en aurait. Cependant, mon application même de la méthodologie scientifique pour la construction de connaissance, n'est pas un processus complètement désintéressé sur de nombreux aspects.

Tout d'abord, j'ai eu un certain nombre d'étudiants et de projets en collaboration avec des entreprises. Bien que cela présente un intérêt scientifique en permettant de confronter les modèles à des cas d'usage métier et à des applicatifs parfois plus réalistes et complexes, je ne peux négliger le fait qu'ils ont une visée applicative directe, qui est au cœur même du sujet de recherche. Ensuite, même dans le cadre de recherches plus théoriques ou fondamentales, cadre dans lequel s'inscrivent plutôt mes travaux, le fait de conceptualiser et de rendre disponible une connaissance ne peut être entièrement détaché de son utilisation puisque c'est le premier point qui permet et autorise le second. Ce genre de débat a émaillé l'histoire des sciences, que ce soit pour la bombe atomique ou pour le génie génétique par exemple. De plus, comme énoncé dans le chapitre 1 d'introduction, l'intelligence artificielle, mais aussi l'informatique en tant que science au sens large, possède une certaine dualité pratique/théorique dans le sens où l'objet d'étude est un artefact et que sa création porte donc une certaine forme d'intentionnalité. Ce dernier point peut toutefois être atténué dans le cas de l'utilisation de l'informatique comme un outil de modélisation pour l'étude d'un phénomène naturel, comme par exemple la modélisation

de comportements humains comme illustré dans le chapitre 3. Cependant, dans tous les cas, les idées de recherche ne peuvent se concevoir qu'en relation avec les connaissances et les courants de pensée actuels et antérieurs, au sein de paradigmes scientifiques [136]. Mes recherches, en particulier mes perspectives, s'inscrivent d'ailleurs clairement dans celui des théories sensori-motrices. Enfin, d'un point de vue pratique, les thématiques de recherche sont en partie pilotées avec une volonté politique à travers les agences de financement, ce qui favorise et oriente l'étude de certains sujets. La recherche, a minima telle qu'elle est pratiquée, ne peut donc être neutre et ses acteur.rice.s sont donc légitimes, même si pas forcément les mieux formé.e.s, pour se poser la question de l'impact de leur recherche.

L'intelligence artificielle est une discipline particulièrement porteuse d'enjeux, comme en témoignent les financements publics mais aussi privés accordés à ce domaine. Il y a bien sûr des questionnements existentialistes, parfois reliés à la notion de singularité, qui peuvent se retrouver au sein de la communauté de l'intelligence artificielle générale. C'est ce genre de sujet dont s'est emparé Y. Bengio depuis quelques années, et bien que tou.te.s les chercheur.euse.s ne partagent pas forcément son opinion, cela a au moins le mérite d'alimenter le débat et de le faire entrer dans un cadre scientifique. Sans forcément être aussi prospectif, l'IA permet l'automatisation du traitement de l'information pour la résolution de tâches et de problèmes de plus en plus complexes, ce qui a déjà un certain nombre de répercussions dans la société. Cette automatisation a entre autres pour but de remplacer l'humain dans certaines tâches, ce qui peut viser par exemple à l'amélioration de la sécurité ou du confort, ou à la réduction du coût économique. Cela touche de manière globale à notre rapport au travail, mais également au rôle que l'emploi joue dans la conception et la structuration, entre autre économique, de notre société. Ce traitement automatique de l'information questionne également l'accès aux données (avec le concept de données ouvertes ou l'introduction du RGPD en Europe pour le respect de la vie privée), mais aussi notre accès et notre utilisation de l'information en tant qu'êtres humains. Il y a par exemple les mécanismes de désinformation de plus en plus poussés, qui peuvent être utilisés comme moyen de pression jusqu'à un niveau géopolitique. Mais les algorithmes d'apprentissage automatique sont également sujets aux biais de genre, de couleur de peau, etc. et l'utilisation de ces méthodes risque de les accentuer ou du moins de les faire perdurer. Cela a en revanche le mérite de mettre en lumière ces discriminations à l'œuvre dans nos sociétés, puisque les apprentissages sont en grande partie le reflet des corrélations qui se trouvent dans les bases de données, possiblement d'autant plus exacerbées par le fait que les états, structures et personnes qui développent et promeuvent l'IA ne sont pas représentatives de la diversité de l'humanité et du monde. Enfin, l'apprentissage des modèles de réseaux de neurones profonds est particulièrement gourmand en puissance de calcul, ce qui amène à un impact environnemental croissant. Dans le cas d'outils largement déployés comme ChatGPT par exemple, c'est cependant la phase d'inférence, à cause du nombre de requêtes, qui a l'impact le plus important (estimé à 25 fois celui de l'entraînement par an) et qui dépend fortement du mix énergétique du pays où les requêtes sont traitées [87]. L'IA soulève ainsi de nombreux questionnements économiques, sociologiques, éthiques, anthropologiques, environnementaux, etc.

En pratique, l'impact de mes recherches reste, pour le moment, limité, que ce soit de par mes thématiques de recherche ou par le nombre restreint de personnes concernées. Cependant, en tant que membre d'une communauté de recherche, je participe à l'activité et à la visibilité de l'IA en tant que discipline, ce qui m'incite, voire m'oblige, à porter un regard critique sur mes activités. Je cherche donc à essayer de mettre en place un certain nombre d'éléments dans ma pratique quotidienne de la recherche pour essayer de

restreindre leurs potentielles conséquences négatives. Tout d’abord, j’essaie d’aborder, de sensibiliser et de débattre de ces questionnements avec les étudiant.e.s que j’encadre dans le cadre de leur formation par et pour la recherche. Plus largement, je participe très régulièrement à des événements de médiation scientifique pour expliquer mes recherches, mais plus largement pour informer le public ⁵⁴ sur les principes de fonctionnement de l’apprentissage machine ainsi que les questionnements que cela soulève. Plus largement, j’essaie de cadrer mes recherches, que ce soit par le choix des thématiques, de mes collaborations, ou encore du type de financement recherché. Cela est grandement facilité par ma position privilégiée de (enseignant) chercheur en IA, qui dispose donc d’un large panel d’offres disponibles vu les crédits de recherche actuellement alloués à cette thématique. En ce sens, j’ai participé à un projet de recherche en apprentissage de comportements éthiques grâce auquel j’ai co-écrit un papier avec un collègue philosophe sur les questionnements éthiques de l’IA [36] et avec lequel, entre autres, un article est en préparation sur un cadre d’analyse et de discussion des injections éthiques dans les systèmes socio-techniques. De plus, dans le cadre du projet R&D BOOSTER dont je parlais dans les perspectives (voir section 5.2), nous allons étudier comment réduire la taille des modèles d’apprentissage sans trop affecter la performance générale des modèles, ce qui pourrait être un premier pas pour moi dans l’IA frugale. À plus long terme, j’envisage donc d’essayer d’intégrer plus régulièrement les enjeux sociétaux au sein de mes recherches, par exemple dans le cadre des sciences de l’environnement.

⁵⁴Cela fait d’ailleurs parti des obligations statutaires des enseignants chercheurs. “Ils contribuent au dialogue entre sciences et sociétés, notamment par la diffusion de la culture et de l’information scientifique et technique” (Article 3 du décret n°84-431 du 6 juin 1984 fixant les dispositions statutaires communes applicables aux enseignants-chercheurs et portant statut particulier du corps des professeurs des universités et du corps des maîtres de conférences.)

6 Appendices

The Impact of Action in Visual Representation Learning

Alexandre Devillers

Univ Lyon, Université Lyon 1
LIRIS, UMR5205

Lyon, France

alexandre.devillers@liris.cnrs.fr

Valentin Chaffraix

Univ Lyon, INSA Lyon
LIRIS, UMR5205

Lyon, France

valentin.chaffraix@gmx.com

Frédéric Armetta

Univ Lyon, Université Lyon 1
LIRIS, UMR5205

Lyon, France

frederic.armetta@liris.cnrs.fr

Stefan Duffner

Univ Lyon, INSA Lyon
LIRIS, UMR5205

Lyon, France

stefan.duffner@liris.cnrs.fr

Mathieu Lefort

Univ Lyon, Université Lyon 1

LIRIS, UMR5205

Lyon, France

mathieu.lefort@liris.cnrs.fr

Abstract—Sensori-motor theories, inspired by work in neuroscience, psychology and cognitive science, claim that actions, through learning and mastering of a predictive model, are a key element in the perception of the environment. On the computational side, in the domains of representation learning and reinforcement learning, models are increasingly using self-supervised pretext tasks, such as predictive or contrastive ones, in order to increase the performance on their main task. These pretext tasks are action-related even if the action itself is usually not used in the model. In this paper, we propose to study the influence of considering action in the learning of visual representations in deep neural network models, an aspect which is often underestimated w.r.t. sensori-motor theories. More precisely, we quantify two independent factors: 1- whether or not to use the action during the learning of visual characteristics, and 2- whether or not to integrate the action in the representations of the current images. Other aspects will be kept as simple and comparable as possible, that is why we will not consider any specific action policies and combine simple architectures (VAE and LSTM), while using datasets derived from MNIST. In this context, our results show that explicitly including action in the learning process and in the representations improves the performance of the model, which opens interesting perspectives to improve state-of-the-art models of representation learning.

Index Terms—Sensori-motor theory, Representation learning, Predictive learning, Deep learning

I. INTRODUCTION

Sensori-motor theories are based on substantial evidence in neuroscience, developmental psychology and cognitive science. The main claim is that actions, and more especially the sensory changes induced by motor actions, play a key role in learning a predictive model of the world and in perceiving it [23]. For example, a kitten that cannot walk, i.e. it only passively receives a visual flow, will learn defective visual representations [14]. The role of action is also emphasised in the notion of affordance in psychology [9], where an object is not defined by a set

This work was performed using HPC resources from GENCI-IDRIS and a GPU donated by the NVIDIA Corporation. We gratefully acknowledge this support. This work was financed by the Auvergne Rhône-Alpes (AURA) region, within the Ethics.AI project (Pack Ambition Recherche). The authors would like to thank the AURA region and their partners in this project.

of properties but by its elicited interactions for the agent. According to the sensori-motor contingencies theory, acting may even play a role in some form of consciousness [25]. These concepts are also related to the theories of enactivism and embodiment that states that the body, as the structure to interact with the world, is required for an intelligent behavior to arise [8]. While contributing to the learning of representations, the actions could also be aimed at perceiving relevant regions of the environment, which would make perception an active process. This way the actions would be required to accumulate evidence of the current state of the world as unified in the free energy principle e.g. [7]. With regard to vision, this is for example the role of saccades which allow to get successive glimpses over a visual scene [6].

Since some years, deep learning achieves state-of-the-art performance in multiple domains such as visual recognition, natural language processing, game playing etc. [20]. Initially, these data-driven approaches were mainly supervised, e.g. by using a Convolutional Neural Network (CNN) to classify objects in images [13]. Contrary to human beings that perform saccades to perceive a scene, most CNN models have a translation invariance property that allows them to process the whole image at once. Then, deep architectures have been adapted to the reinforcement learning framework [27]. Here, actions are considered through sequential decision making, but are not explicitly included neither in the perception nor in the building of representations. More recently, self-supervised approaches have emerged. They propose to use a pretext task during learning, usually making close the representations of inputs considered similar, to improve performance of a predefined task or in the context of unsupervised learning. In computer vision, some of the similar inputs generation processes can be interpreted as the resulting from movements [18]. In reinforcement learning, temporal prediction of consequences of action is often used as a pretext task.

Thus, sensori-motor theories and the recent and promising

trend of including action-related pretext tasks in deep learning seem to point towards a benefit of action in representation learning and perception, at different degrees. Yet the precise quantification of the impact of action in representation learning is still barely known. In this article, we propose to open this research question by studying two independent factors: 1- whether or not to use action in the learning of visual features and 2- whether or not to use action in the computation of the current image representation. To keep the study tractable, we restrain ourselves to simple deep architectures as illustrative examples. Moreover, to put apart the question of the action decision process, which would introduce a retro-action loop during learning, the model will perform random actions.

Section II introduces existing works related to representation learning considering actions. In section III, we derive from our research objectives the different neural network architectures and loss functions used in our study. The protocol and hyper-parameters used and the obtained results are presented in section IV. Finally, we draw our conclusions and expose various perspectives for future works (section V).

II. RELATED WORK

Multiple works in robotics considered action while learning predictive models of the environment for achieving a variety of tasks such as object manipulation, recognition or grasping. The benefits of such interactive perception are mainly to get access to some objects characteristics requiring manipulation as weights for example and to enrich and structure the regularities in the inputs (see [5] for an in-depth survey). Considering explicitly sensori-motor contingencies can even push these properties a step further. Arranging sensori-motor schemes hierarchically leads to the learning of the complex concept of container, that could be reused across environments [12]. Sensory representation learning can be shaped by action, through the notion of compensating movements, i.e. that some displacements in the sensory inputs can be reversed via motor actions. A deep architecture, designed with this principle, is able to learn the underlying spatial structure of the input [19].

In computer vision, recent visual representation learning methods rely on either contrastive or predictive pretext tasks. In contrastive ones, models usually learn to embed multiple views of an image into similar representations [4]. The generation process used to obtain these views can be related to some form of action [18], such as *cropping* which can be linked to head movement and eye saccades. However, these methods include the actions neither to build nor to learn the representations. Such tasks can also rely on predicting the motion that led from the actual view to the future one [2], in this case the action can be seen as a supervision signal, however it is not directly integrated in the representations. For predictive tasks, they generally aim at predicting future inputs based on historical ones, as in [24] where a contrastive predictive task has been successfully applied to vision, audio, natural language processing, and reinforcement learning. Such tasks are well suited for environments with a temporal aspect, as in

the context of reinforcement learning where the prediction of future observations from historical observations and actions has shown to learn good representations [11].

While most computer vision models have been focused on treating full images at once, only few works consider processing sub-parts of images. Such models that only process glimpses of images were initially introduced for computational advantage, but also open the possibility of making models actively perceiving the world by choosing where to look. This idea of processing glimpses of an image has been applied to classification two ways: either by dedicating a neural network to each glimpse w.r.t. its temporal index [26], or by letting a recurrent network learn to perform saccades in a reinforcement learning environment [22]. Later, these models have been enhanced to perform multiple object recognition, as in [3] where the model learns to classify objects from left to right by moving a virtual glimpse sensor over the image, or in [1] where the model classifies objects sequentially while determining an affine transformation to produce the next glimpse to locate the next object. Moreover, [10] also used glimpses for image generation both to "read" and "write" images, iteratively generating the result with small patches while showing strong representational and generational capacities.

III. STUDIED MODELS

A. Overview

1) *Problem statement*: In this article we consider a system that receives visual saccades to perceive its environment. At each step, the system only takes as input a sub-part of the image and the action to come. The action defines the 2D position of the center of the next visual input. In order to decorrelate the action policy from the learned representations, it is the same for all models and consists in a random sampling from a uniform distribution. We note x_t the observed glimpse (i.e. image sub-part) at time t , a_t the next action performed, i.e. the position of the next observed glimpse x_{t+1} .

2) *General overview of the model*: The task the model has to perform mixes the prediction of the future visual input for a given action and the reconstruction of the current visual input. We note \hat{x}_t (resp. \hat{x}_{t+1}) the reconstruction by the model of the glimpse x_t (resp. x_{t+1}). All the model variations that we study are relying on the same modules, each one addressing a specific point of the combined task:

- The first is a convolutional Variational Auto-Encoder (VAE) [17], which reduces the dimensionality of the current glimpse by projecting it in a latent space and then reconstructing it.
- The second is a Long Short-Term Memory (LSTM) neural network [16]. As the system only gets partial glimpses of the environment, it needs to integrate the current observation with past ones to construct a global representation of the observed image. Its output is what we consider as the representation of the current image.
- The third, which we call the recoder, is a neural network we introduce, to generate a latent embedding of the next

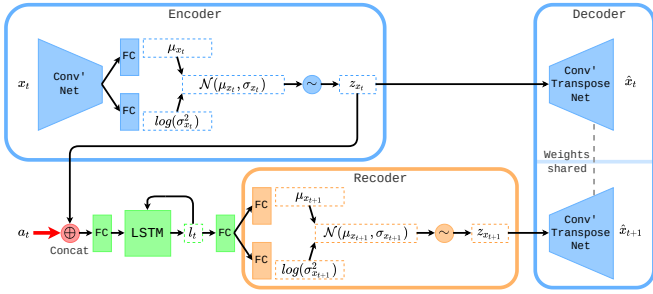


Fig. 1. PreLSTM — The input x_t passes through the encoder, transforming x_t in its latent representation z_{x_t} . Then z_{x_t} passes through the decoder, giving \hat{x}_t the reconstructed input produced by the VAE. On the other side, the action a_t is concatenated with z_{x_t} and is then fed to the LSTM, which outputs l_t the global representation of the image. From this representation the recoder computes $z_{x_{t+1}}$ and finally, by passing through the decoder, the constructed prediction \hat{x}_{t+1} of the next glimpse x_{t+1} .

glimpse w.r.t. the action to come and the representation from the LSTM. This embedding is used to reconstruct the next visual input. From a technical perspective, the functioning of the recoder is similar to a VAE’s encoder by generating a distribution from which the recoded latent embedding is sampled. As it is the case for the VAE, this distribution is regularized.

In the next two sections, we will describe the 4 models compared in this article, that vary over two axes:

- 1) whether or not to use the action in the LSTM representations, i.e. to make the representations sensori-motor (Sec. III-B),
- 2) whether or not to use the action during the learning of the visual characteristics by the VAE’s encoder, i.e. making the learning of the visual characteristics partly sensori-motor (Sec. III-C).

B. Influence of action in the representations

1) *With action:* The PreLSTM architecture, illustrated in Fig. 1, integrates the actions before the LSTM. While combining observed glimpses, by providing the action the LSTM will construct sensori-motor representations. Indeed, the content of the action is forced to pass through the LSTM in order to get used by the recoder, forcing the representation to be a mix of sensory and motor information. Note that to ensure that the dimensions of the VAE’s latent space and the LSTM’s output are constant across all model variations architectures, in addition to keep similar computational capacity between both architectures, one Fully-Connected (FC) layer is placed before and after the LSTM.

2) *Without action:* The PostLSTM architecture, see Fig. 2, is similar to the PreLSTM one except that it concatenates the action after the LSTM. This makes the latent representation l_t purely visual, as the action is no more directly used to construct the representation. Note that the recoder still has access to the information of the performed action and the representation as in PreLSTM.

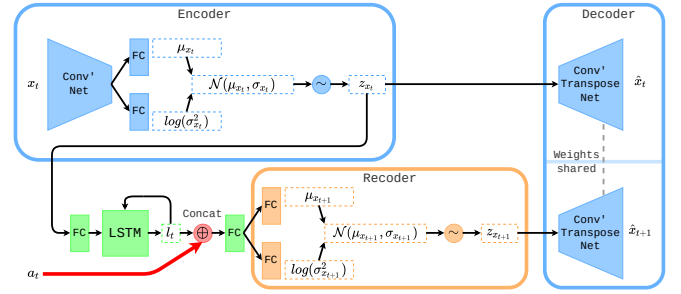


Fig. 2. PostLSTM — The whole network works the same way as in PreLSTM except that z_{x_t} is directly fed to the LSTM, and that the action a_t is concatenated with l_t before passing through the recoder.

C. Influence of action in the learning

1) *With action:* The first method jointly optimizes the parameters of the whole neural network during training *end-to-end*. The overall loss of the model \mathcal{L}_{tot} (Eq. 3) is composed of the loss of the VAE \mathcal{L}_{vae} (Eq. 1), consisting in the reconstruction of the current glimpse, and the loss of the recoder \mathcal{L}_{rec} (Eq. 2) which represents the prediction of the future glimpse. While minimizing this loss, information from the action is backpropagated through the whole architecture including the visual features learned by the VAE.

$$\mathcal{L}_{vae} = \|x_t - \hat{x}_t\|^2 + \beta_{vae} D_{KL}[\mathcal{N}(\mu_{x_t}, \sigma_{x_t}) || \mathcal{N}(0, 1)] \quad (1)$$

$$\mathcal{L}_{rec} = \|x_{t+1} - \hat{x}_{t+1}\|^2 + \beta_{rec} D_{KL}[\mathcal{N}(\mu_{x_{t+1}}, \sigma_{x_{t+1}}) || \mathcal{N}(0, 1)] \quad (2)$$

$$\mathcal{L}_{tot} = \mathcal{L}_{vae} + \mathcal{L}_{rec} \quad (3)$$

\mathcal{L}_{vae} is the loss of a Beta-VAE [15]. The first term is the Mean Squared Error (MSE) between the input glimpse x_t and its reconstruction \hat{x}_t . The second term, used as a regularization weighted by β_{vae} , is the KL-Divergence between the distribution created by the VAE, $\mathcal{N}(\mu_{x_t}, \sigma_{x_t})$ and the standard normal distribution, $\mathcal{N}(0, 1)$. \mathcal{L}_{rec} is derived from \mathcal{L}_{vae} , where the MSE is between the next glimpse x_{t+1} and its recoded reconstruction \hat{x}_{t+1} , while the regularized distribution is the one created by the recoder $\mathcal{N}(\mu_{x_{t+1}}, \sigma_{x_{t+1}})$ and is weighted by β_{rec} .

2) *Without action:* In order to analyze if the action has an impact on the extracted visual features, we propose a *separated* two-step learning procedure.

In the first step, the VAE is trained without actions so that the learned features are purely visual. To have a fair comparison, the prediction task, that requires actions, is replaced by a second reconstruction of the current glimpse. For this purpose, we use a temporary architecture which instead of having a *classic* recoder, has an identity recoder (see Fig. 3), that recodes the current perceived glimpse from the LSTM. The loss $\mathcal{L}_{pretrain}$ (Eq. 5), used for this first step, is composed of \mathcal{L}_{vae} (Eq. 1) the loss of the Beta-VAE, but also of \mathcal{L}_{recId} (Eq. 4) the loss of the identity recoder.

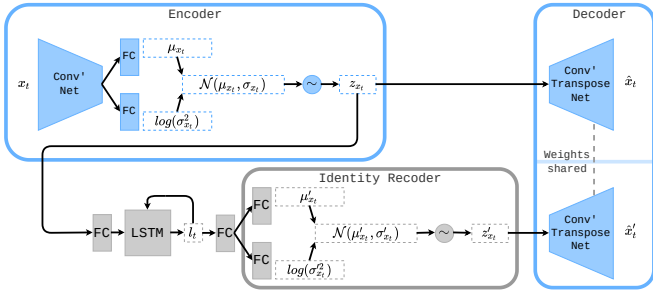


Fig. 3. Temporary architecture used in the first step (VAE training) of the two-step separated training, at the end only the weights of the encoder and decoder are kept and frozen and the identity recoder, is replaced by the one of PreLSTM or PostLSTM architectures.

$$\mathcal{L}_{recId} = \|x_t - \hat{x}_t'\|^2 + \beta_{rec} D_{KL}[\mathcal{N}(\mu'_{x_t}, \sigma'_{x_t}) \| \mathcal{N}(0, 1)] \quad (4)$$

$$\mathcal{L}_{pretrain} = \mathcal{L}_{vae} + \mathcal{L}_{recId} \quad (5)$$

In \mathcal{L}_{recId} , the MSE is between the input glimpse x_t and its recoded reconstruction \hat{x}_t' , and the distribution of the identity recoder $\mathcal{N}(\mu'_{x_t}, \sigma'_{x_t})$ is regularized.

In the second step, we replace the LSTM/identity recoder by a normal predictive one (i.e. either the PreLSTM or the PostLSTM architecture) while freezing the VAE's weights to train only the LSTM/recoder using the loss \mathcal{L}_{rec} (Eq. 2).

IV. EXPERIMENTS

A. Datasets

1) 28×28 MNIST: The MNIST digits dataset [21] is composed of 28×28 pixels images that contain centered white hand-written digits on a black background. We used a number of 15 glimpses per image for this dataset.

2) 60×60 MNIST: To make the digits unrecognizable at the first glimpse, we use images of MNIST resized to 60×60 pixels. This way, patches are no more digit fragments, but strokes and curves. For this dataset, we used a number of 30 glimpses per image as images are bigger.

3) 60×60 Cluttered Translated MNIST: In this dataset [22] (named 60×60 CT MNIST hereafter), images are 60×60 black background with a 28×28 MNIST digit randomly placed on them, and where four 8×8 clutters (extracted from other MNIST digits) are also randomly added on them. This dataset is the hardest since clutters and digit positions are totally unpredictable if never seen. This stochasticity makes the predictive task way harder. We used a number of 50 glimpses per image as the task is harder.

Note that for every dataset we split the train set in 5-folds, leading to 48k (resp. 12k) images for the training (resp. validation) and we used the test set with 10k samples.

B. Evaluation

To evaluate and compare the representations learned by the different models, we trained a classifier taking the LSTM output as input and measured the respective classification accuracy

of the digits, averaged over 10 executions. We used a MLP composed of two hidden layers of 32 neurons each (one with Dropout $p = 0.5$), and the ReLU activation function. The training was done a posteriori, thus the weights of the rest of the model were frozen. This classifier was trained on all representations produced by the LSTM from the successive glimpses. Thus, we can study how the performance evolves when new glimpses are integrated in the model.

We also tracked the loss of the different models on the predictive and reconstruction tasks, and complete our quantitative evaluation with a more qualitative one based on t-SNE projections of the LSTM representations after the last glimpse.

C. Implementation details

1) *Glimpses and actions*: Glimpses are patches of the observed image extracted using a cropping window with a fixed size of 14×14 pixels. The position of this window is determined by the performed actions, and cannot be out of the image. Actions are 2D vectors encoding the continuous absolute position of the center of this cropping window, and they are uniformly sampled from the action space.

2) *Models hyperparameters*: The CNN used for the encoder (see section III) is composed of three 2D convolution layers and a FC layer, with ReLU as activation function. Convolution layers have respectively 8, 16, and 32 output channels, and kernels of size 3, 3, and 5. The output of the last convolution is flattened, and passed through the FC whose output dimension is 128. The dimension of the latent space z is 16, therefore the output size of FC generating μ and σ is also 16 for both the encoder and the recoder. The decoder is composed of a 16 to 128 FC followed by a mirror version of the encoder's CNN where input and output sizes are swapped, order is reversed and convolution layers are transposed ones. The LSTM has an input and hidden size of 128. Therefore, in the PreLSTM the FC before the LSTM has an input size of 18 (16+2) and an output size of 128, while the input size is 16 in PostLSTM (see section III-B1). Finally, the FC after the LSTM in PreLSTM has an input size of 128 and output size of 128, while the input size is 130 (128+2) in PostLSTM.

We used the Adam optimizer with a learning rate of 0.001 for both the self-supervised and the classification tasks, and have chosen $\beta_{vae} = \beta_{rec} = 0.5$ as it showed better performances. Models are trained for 200 epochs (200 epochs for the VAE then again 200 epochs for the LSTM/recoder, in the case of a separated training), while the a posteriori classification task is trained with 75 epochs for all models. We used a batch size of 128 in all configurations.

D. Results

The classification performance on the 3 datasets for the various models with increasing number of perceived glimpses is presented in Fig. 4. This metric allows us to compare the representations learned by the different models on the presence of semantic information through the ease of separation.

Firstly, we observe that the models not using the action during the learning of the VAE's encoder (-Sep suffix) perform

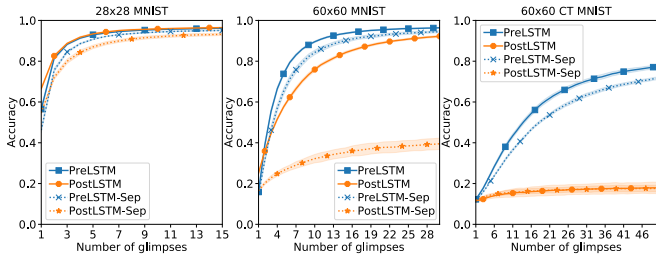


Fig. 4. Classification accuracy versus number of received glimpses, for the various models on the 3 datasets.

worse than their counterpart using the action (no suffix). The presence of this trend for both architectures and for all datasets shows that considering the action in the learning of visual characteristics seems to be beneficial for the extraction of meaningful features.

Secondly, we can see that all models integrating the action in the LSTM (PreLSTM architectures) perform better than their equivalent ones integrating the action after the LSTM (PostLSTM architectures), except for the 28×28 MNIST dataset. In this last case, they are similar when both performing an end-to-end training (no suffix), which may be due to the simplicity of the dataset. This difference of performance between Pre and Post architectures tends to show that considering the action in the representations, i.e. in the LSTM, helps to build better representations of the environment. Moreover, as this trend accentuates as the dataset becomes harder, the presence of the action in the representations seems to be more important for complex tasks. However, this difference of performance may also be explained by the fact that in the PreLSTM architectures the LSTM can use the action as an additional information to integrate the glimpses using their position in a global internal picture. Yet this may not be enough to explain all the differences as we observe the strongest difference on the 60×60 CT MNIST dataset where the position is less important as digits are small enough to get mostly captured by one glimpse. Note that all the observed trends are clearer after a certain amount of glimpses. This can be explained by the fact that the models need to temporally integrate the glimpses in order to build the representations. As all models start with an empty representation their few first representations may have similar results, but the more and the better they integrate the glimpses the better the representations would be.

The evolution during the training of the reconstruction error for the predictive task on the validation set is shown in Fig. 5. The results show the same trends as the ones on the accuracy, confirming the findings about the importance of including the action both in the learning of visual features and in the representations. However, we note that having better reconstruction loss does not necessarily imply better learned representations. For instance, the PostLSTM model on the 28×28 MNIST dataset has a higher error compared to the PreLSTM, while both have similar results on the classification task.

Finally, Fig. 6 shows the t-SNE projection of the represen-

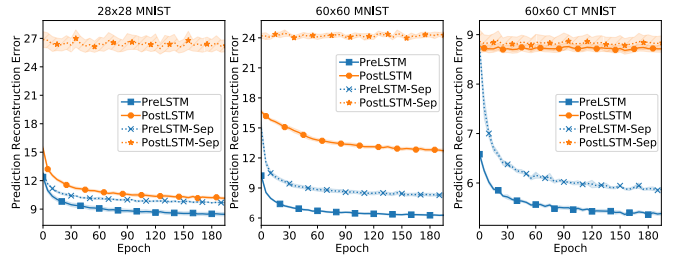


Fig. 5. Evolution of the reconstruction error for the predictive task in validation.

tations for all models and for all datasets. We consider that the representations are better if the clusters are separable, i.e. with few outliers and with some space between them, and if they are expressive, i.e. clusters are spread and detailed. For each architecture, clusters of the end-to-end trained model (no suffix) are clearer and have less outliers than the ones trained in two steps (-Sep suffix). This shows that visual features learned with the action led to easier separable representations. We also observe that the models using the action to build the representations (PreLSTM and PreLSTM-Sep) are always able to cluster the representations with a varying quality depending on the dataset, where PostLSTM and PostLSTM-Sep models produce mixed representations for the hardest datasets. These results are in line with those found previously.

V. CONCLUSION AND PERSPECTIVES

In this article, we studied the impact of action in visual representation learning in deep networks. Our questioning is raised by recent deep learning methods, which are increasingly using pretext tasks based on transformations that are action-related. Yet, these methods are not considering these *actions* to build their representations while sensori-motor theories, based on substantial evidence in many fields, claim that action is essential to perception. For this purpose, we studied and crossed two independent factors: 1- whether or not to use the action during the learning of visual features, and 2- whether or not to integrate the action in the building of image representations. By comparing these four configurations, we show that models including action during the learning of visual characteristics always perform better than their counterpart. We also observe that variations integrating the action directly in the representations tends to perform better, a trend that is more prominent for harder datasets.

These results are in line with sensori-motor theories and open perspectives to improve state-of-the-art representation learning methods by integrating the action both in the representations and during the learning. An other interesting perspective could be to study the influence of the action policy, in active learning and active perception contexts, on the learned representations. In the future, we want to extend the test-bed we elaborated for the study and make use of these first promising results to explore if it transfer to state-of-the-art representation learning methods and for more general problems studied by community (robotic, open world environments, etc.).

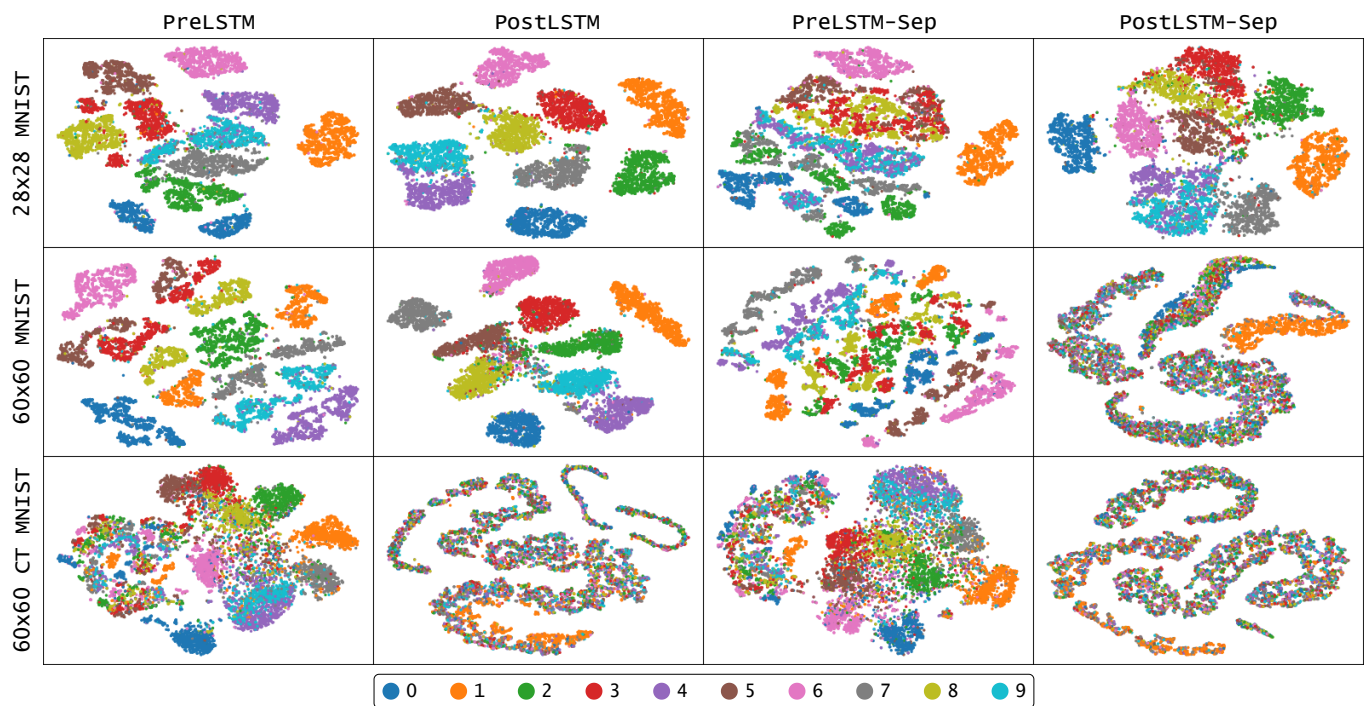


Fig. 6. t-SNE of the latent LSTM encoding for all models and datasets.

REFERENCES

- [1] Ablavatski, A., Lu, S., Cai, J.: Enriched deep recurrent visual attention model for multiple object recognition. In: IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 971–978 (2017)
- [2] Agrawal, P., Carreira, J., Malik, J.: Learning to see by moving. In: Proceedings of the IEEE international conference on computer vision. pp. 37–45 (2015)
- [3] Ba, J., Mnih, V., Kavukcuoglu, K.: Multiple object recognition with visual attention. arXiv preprint arXiv:1412.7755 (2014)
- [4] Bardes, A., Ponce, J., LeCun, Y.: Vicreg: Variance-invariance-covariance regularization for self-supervised learning. arXiv preprint arXiv:2105.04906 (2021)
- [5] Bohg, J., Hausman, K., Sankaran, B., Brock, O., Kragic, D., Schaal, S., Sukhatme, G.S.: Interactive perception: Leveraging action in perception and perception in action. IEEE Transactions on Robotics **33**(6), 1273–1291 (2017)
- [6] Friston, K., Adams, R., Perrinet, L., Breakspear, M.: Perceptions as hypotheses: saccades as experiments. *Frontiers in psychology* **3**, 151 (2012)
- [7] Friston, K., Mattout, J., Kilner, J.: Action understanding and active inference. *Biological cybernetics* **104**(1), 137–160 (2011)
- [8] Froese, T., Ziemke, T.: Enactive artificial intelligence: Investigating the systemic organization of life and mind. *Artificial Intelligence* **173**(3-4), 466–500 (2009)
- [9] Gibson, J.J., Carmichael, L.: The senses considered as perceptual systems, vol. 2. Houghton Mifflin Boston (1966)
- [10] Gregor, K., Danihelka, I., Graves, A., Rezende, D., Wierstra, D.: Draw: A recurrent neural network for image generation. In: International Conference on Machine Learning. pp. 1462–1471. PMLR (2015)
- [11] Guo, Z.D., Pires, B.A., Piot, B., Grill, J.B., Althé, F., Munos, R., Azar, M.G.: Bootstrap latent-predictive representations for multitask reinforcement learning. In: International Conference on Machine Learning. pp. 3875–3886. PMLR (2020)
- [12] Hay, N., Stark, M., Schlegel, A., Wendelken, C., Park, D., Purdy, E., Silver, T., Phoenix, D.S., George, D.: Behavior is everything: Towards representing concepts with sensorimotor contingencies. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018)
- [13] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
- [14] Held, R., Hein, A.: Movement-produced stimulation in the development of visually guided behavior. *Journal of comparative and physiological psychology* **56**(5) (1963)
- [15] Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework (2016)
- [16] Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural computation* **9**(8), 1735–1780 (1997)
- [17] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
- [18] Laflaquière, A.: A sensorimotor perspective on contrastive multiview visual representation learning. *IEEE Transactions on Cognitive and Developmental Systems* (2021)
- [19] Laflaquière, A., Garcia Ortiz, M.: Unsupervised emergence of spatial structure from sensorimotor prediction. arXiv e-prints pp. arXiv-1810 (2018)
- [20] LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *nature* **521**(7553), 436–444 (2015)
- [21] LeCun, Y., Cortes, C., Burges, C.: Mnist handwritten digit database (2010)
- [22] Mnih, V., Heess, N., Graves, A., Kavukcuoglu, K.: Recurrent models of visual attention. arXiv preprint arXiv:1406.6247 (2014)
- [23] Mossio, M., Taraborelli, D.: Action-dependent perceptual invariants: From ecological to sensorimotor approaches. *Consciousness and cognition* **17**(4) (2008)
- [24] Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
- [25] O’Regan, J.K., Noë, A.: A sensorimotor account of vision and visual consciousness. *Behavioral and brain sciences* **24**(5), 939 (2001)
- [26] Ranzato, M.: On learning where to look. arXiv preprint arXiv:1405.5488 (2014)
- [27] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al.: Mastering the game of go without human knowledge. *nature* **550**(7676), 354–359 (2017)

EQUIMOD: AN EQUIVARIANCE MODULE TO IMPROVE VISUAL INSTANCE DISCRIMINATION

Alexandre Devillers & Mathieu Lefort

Univ Lyon, UCBL, CNRS, INSA Lyon

LIRIS, UMR5205, F-69622

Villeurbanne, France

{alexandre.devillers, mathieu.lefort}@liris.cnrs.fr

ABSTRACT

Recent self-supervised visual representation methods are closing the gap with supervised learning performance. Most of these successful methods rely on maximizing the similarity between embeddings of related synthetic inputs created through data augmentations. This can be seen as a task that encourages embeddings to leave out factors modified by these augmentations, i.e. to be invariant to them. However, this only considers one side of the trade-off in the choice of the augmentations: they need to strongly modify the images to avoid simple solution shortcut learning (e.g. using only color histograms), but on the other hand, augmentations-related information may be lacking in the representations for some downstream tasks (e.g. literature shows that color is important for bird and flower classification). Few recent works proposed to mitigate this problem of using only an invariance task by exploring some form of equivariance to augmentations. This has been performed by learning additional embeddings space(s), where some augmentation(s) cause embeddings to differ, yet in a non-controlled way. In this work, we introduce *EquiMod* a generic equivariance module that structures the learned latent space, in the sense that our module learns to predict the displacement in the embedding space caused by the augmentations. We show that applying that module to state-of-the-art invariance models, such as BYOL and SimCLR, increases the performances on the usual CIFAR10 and ImageNet datasets. Moreover, while our model could collapse to a trivial equivariance, i.e. invariance, we observe that it instead automatically learns to keep some augmentations-related information beneficial to the representations.

Source code is available at <https://github.com/ADevillers/EquiMod>

1 INTRODUCTION

Using relevant and general representation is central for achieving good performances on downstream tasks, for instance when learning object recognition from high-dimensional data like images. Historically, feature engineering was the usual way of building representations, but we can currently rely on deep learning solutions to automate and improve this process of representation learning. Still, it is challenging as it requires learning a structured latent space while controlling the precise amount of features to put in representations: too little information will lead to not interesting representations, yet too many non-pertinent features will make it harder for the model to generalize. Recent works have focused on Self-Supervised Learning (SSL), i.e. determining a supervisory signal from the data with a pretext task. It has the advantages of not biasing the learned representation toward a downstream goal, as well as not requiring human labeling, allowing the use of plentiful raw data, especially for domains lacking annotations. In addition, deep representation learning encourages network reuse via transfer learning, allowing for better data efficiency and lowering the computational cost of training for downstream tasks compared to the usual end-to-end fashion.

The performances of recent instance discrimination approaches in SSL of visual representation are progressively closing the gap with the supervised baseline (Caron et al., 2020; Chen et al., 2020a;b;

Chen & He, 2021; Bardes et al., 2021; Grill et al., 2020; He et al., 2020; Misra & Maaten, 2020; Zbontar et al., 2021). They are mainly siamese networks performing an instance discrimination task. Still, they have various distinctions that make them different from each other (see Liu (2021) for a review and Szegedy et al. (2013) for a unification of existing works). Their underlying mechanism is to maximize the similarity between the embedding of related synthetic inputs, a.k.a. views, created through data augmentations that share the same concepts while using various tricks to avoid a collapse towards a constant solution (Jing et al., 2021; Hua et al., 2021). This induces that the latent space learns an invariance to the transformations used, which causes representations to lack augmentations-related information.

Even if these models are self-supervised, they rely on human expert knowledge to find these relevant invariances. For instance, as most downstream tasks in computer vision require object recognition, existing augmentations do not degrade the categories of objects in images. More precisely, the choice of the transformations was driven by some form of supervision, as it was done by experimentally searching for the set of augmentations giving the highest object recognition performance on the ImageNet dataset (Chen et al., 2020a). For instance, it has been found that color jitter is the most efficient augmentation on ImageNet. One possible explanation is that color histograms are an easy-to-learn shortcut solution (Geirhos et al., 2020), which is not removed by cropping augmentations (Chen et al., 2020a). Indeed, as there are many objects in the categories of ImageNet, and as an object category does not change when its color does, the loss of color information is worth removing the shortcut. Still, it has been shown that color is an essential feature for some downstream tasks (Xiao et al., 2020).

Thus, for a given downstream task, we can separate augmentations into two groups: the ones for which the representations benefit from insensitivity (or invariance) and the ones for which sensitivity (or variance) is beneficial (Dangovski et al., 2021). Indeed, there is a trade-off in the choice of the augmentations: they require to modify significantly the images to avoid simple solution shortcut learning (e.g. relying just on color histograms), yet some downstream tasks may need augmentations-related information in the representations. Theoretically, this trade-off limits the generalization of such representation learning methods relying on invariance. Recently, some works have explored different ways of including sensitivity to augmentations and successfully improved augmentations-invariant SSL methods on object classification by using tasks forcing sensitivity while keeping an invariance objective in parallel. Dangovski et al. (2021) impose a sensitivity to rotations, an augmentation that is not beneficial for the invariance task, while we focus in this paper on sensitivity to transformations used for invariance. Xiao et al. (2020) proposes to learn as many tasks as there are augmentations by learning multiple latent spaces, each one being invariant to all but one transformation, however, it does not control the way augmentations-related information is conserved. One can see this as an implicit way of learning variance to each possible augmentation. Contrary to these works that do not control the way augmentations-related information is conserved, here we propose to explore sensitivity by introducing an equivariance module that structures its latent space by learning to predict the displacement in the embedding space caused by augmentations in the pixel space.

The contributions of this article are the following:

- We introduce a generic equivariance module *EquiMod* to mitigate the invariance to augmentations in recent methods of visual instance discrimination;
- We show that using *EquiMod* with state-of-the-art invariance models, such as BYOL and SimCLR, boosts the classification performances on CIFAR10 and ImageNet datasets;
- We study the robustness of *EquiMod* to architectural variations of its sub-components;
- We observe that our model automatically learns a specific level of equivariance for each augmentation.

Sec. 2 will present our *EquiMod* module as well as the implementation details while in Sec. 3 we will describe the experimental setup used to study our model and present the results obtained. The Sec. 4 will position our work w.r.t. related work. Finally, in Sec. 5 we will discuss our current results and possible future works.

2 EQUIMOD

2.1 NOTIONS OF INVARIANCE AND EQUIVARIANCE

As in Dangovski et al. (2021), we relate the notions of augmentations sensitivity and insensitivity to the mathematical concepts of invariance and equivariance. Let \mathcal{T} be a distribution of possible transformations, and f denotes a projection from the input space to a latent space. That latent space is said to be invariant to \mathcal{T} if for any given input \mathbf{x} the Eq. 1 is respected.

$$\forall t \in \mathcal{T} \quad f(t(\mathbf{x})) = f(\mathbf{x}) \quad (1)$$

Misra & Maaten (2020) used this definition of invariance to design a pretext task for representation learning. This formulation reflects that the embedding of a non-augmented input sample \mathbf{x} will not change if the input is transformed by any of the transformations in \mathcal{T} . However, more recent works (Bardet et al., 2021; Chen et al., 2020a; Chen & He, 2021; Grill et al., 2020; Zbontar et al., 2021) focused on another formulation of invariance defined by the following Eq. 2.

$$\forall t \in \mathcal{T}, \forall t' \in \mathcal{T} \quad f(t(\mathbf{x})) = f(t'(\mathbf{x})) \quad (2)$$

With this definition, the embedding produced by an augmented sample \mathbf{x} is independent of the transformation used. Still, note that Eq. 1 implies Eq. 2, and that if the identity function is part of \mathcal{T} , which is the case with recent approaches, then both definitions are indeed equivalent.

While insensitivity to augmentation is reflected by invariance, sensitivity can be obtained by achieving variance, i.e. replacing the equality by inequality in Eq. 1 or Eq. 2. Yet, this is not an interesting property, as any injective function will satisfy this constraint. In this paper, we propose to use equivariance as a way to achieve variance to augmentations for structuring our latent space. Eq. 3 gives the definition of equivariance used in the following work.

$$\forall t \in \mathcal{T}, \exists u_t \quad f(t(\mathbf{x})) = u_t(f(\mathbf{x})) \quad (3)$$

With u_t being a transformation in the latent space parameterized by the transformation t , it can be seen as the counterpart of the transformation t but in the embedding space. With this definition, the embeddings from different augmentations will be different and thus encode somehow information related to the augmentations. Yet, if u_t is always the identity then this definition of equivariance becomes the same as invariance Eq.1. Indeed, one can see invariance as a trivial specific case of equivariance. In the following, we only target non-trivial equivariance where u_t produces some displacement in the latent space. See Fig. 1 for a visual comparison of invariance and equivariance.

2.2 METHOD

EquiMod is a generic equivariance module that acts as a complement to existing visual instance discrimination methods performing invariance (Bardet et al., 2021; Chen et al., 2020a; Chen & He, 2021; Grill et al., 2020; Zbontar et al., 2021). The objective of this module is to capture some augmentations-related information originally suppressed by the learned invariance to improve the learned representation. The main idea relies on equivariance, in the sense that our module learns to predict the displacement in the embedding space caused by the augmentations. This way, by having non-null displacement, we ensure embeddings contain augmentations-related information. We first introduce a formalization for these existing methods (see Bardet et al. (2021) for an in-depth explanation of this unification), before introducing how our approach adds on top.

Let t and t' denote two augmentations sampled from the augmentations distribution \mathcal{T} . For the given input image \mathbf{x} , two views are defined as $\mathbf{v}_i := t(\mathbf{x})$ and $\mathbf{v}_j := t'(\mathbf{x})$. Thus, for N original images, this results in a batch of $2N$ views, where the first N elements correspond to a first view (\mathbf{v}_i) for each of the images, and the last N elements correspond to a second view (\mathbf{v}_j) for each of the images. Following previous works, we note f_θ an encoder parameterized by θ producing representations from images, and g_ϕ a projection head parameterized by ϕ , which projects representations in an embedding space. This way, the representations are defined as $\mathbf{h}_i := f_\theta(\mathbf{v}_i)$ as well as $\mathbf{h}_j := f_\theta(\mathbf{v}_j)$,

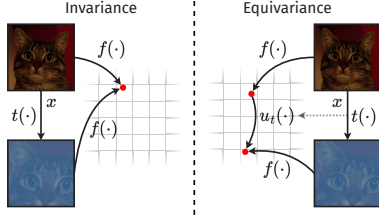


Figure 1: On the left, invariance described by Eq. 1, on the right, equivariance considered in this paper and described by Eq. 3.

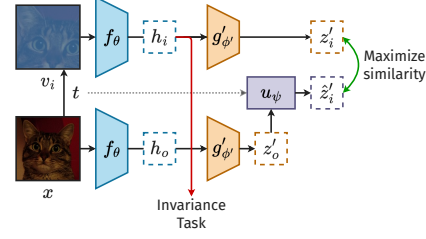


Figure 2: The model learns similar embeddings for an augmented view (z'_i) and the prediction of the displacement in the embedding space caused by that augmentation (\hat{z}'_i), t is a learned representation of the parameters of the transformation, see Sec. 2 for notation details.

and the embeddings as $z_i := g_\phi(\mathbf{h}_i)$ as well as $z_j := g_\phi(\mathbf{h}_j)$. Then, the model learns to maximize the similarity between z_i and z_j , while using diverse tricks to maintain a high entropy for the embeddings, preventing collapse to constant representations.

To extend those preceding works, we introduce a second latent space to learn our equivariance task. For this purpose, we first define a second projection head $g'_{\phi'}$ parameterized by ϕ' whose objective is to project representations in our latent space. Using this projection head we note $z'_i := g'_{\phi'}(\mathbf{h}_i)$ and $z'_j := g'_{\phi'}(\mathbf{h}_j)$, the embeddings of the views v_i and v_j in this latent space we introduce. Moreover, the way we define equivariance in Eq 3 requires us to produce the embedding of the non-augmented image x , thus we note the representation $\mathbf{h}_o := f_\theta(x)$, which is used to create the embedding $z'_o := g'_{\phi'}(\mathbf{h}_o)$ for the given image x . Next, as mentioned in Sec.2.1, to learn an equivariant latent space one needs to determine a transformation u_t for any given t , this can be done either by fixing it or by learning it. In this work, we learn the transformation u_t . To this end, we define u_ψ a projection parameterized by the learnable parameters ψ , referenced later as the equivariance predictor (implementation details about how t is encoded and influences u_ψ are given below in Sec. 2.3). The goal of this predictor is to produce \hat{z}'_i from a given z'_o and t (resp. \hat{z}'_j for z'_o and t'). One can see \hat{z}'_i as an alternative way to obtain z'_i using the equivariance property defined by Eq. 3. Instead of computing the embedding of the augmented view $v_i := t(x)$, we apply t via u_ψ on the embedding z'_o of the original image x .

Therefore, to match this equivariance principle, we need to train $g'_{\phi'}$ and u_ψ so that applying the transformation via a predictor in the latent space (\hat{z}'_i) is similar to applying the transformation in the input space and then computing the embedding (z'_i). For this purpose, we denote (z'_i, \hat{z}'_i) as positive pair (resp. (z'_j, \hat{z}'_j)), and design our equivariance task so that our model learns to maximize the similarity between the positive pairs. Yet, one issue with this formulation is that it allows collapsed solutions, e.g. every z' being a constant. To avoid such simple solutions, we consider negative pairs (as in Chen et al. (2020a); He et al. (2020)) to repulse embedding from other embedding coming from views of different images. We use the Normalized Temperature-scaled cross entropy (NT-Xent) loss to learn from these positive and negative pairs, thus defining our equivariance loss for the positive pair of the invariance loss (i, j) as Eq. 4:

$$\ell_{i,j}^{EquiMod} = -\log \frac{\exp(\text{sim}(z'_i, \hat{z}'_i)/\tau')}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i \wedge k \neq j]} \exp(\text{sim}(z'_i, z'_k)/\tau')} \quad (4)$$

where τ' is a temperature parameter, $\text{sim}(\mathbf{a}, \mathbf{b})$ is the cosine similarity defined as $\mathbf{a}^\top \mathbf{b} / (\|\mathbf{a}\| \|\mathbf{b}\|)$, and $\mathbb{1}_{[k \neq i \wedge k \neq j]}$ is the indicator function evaluated to 1 (0 otherwise) when $k \neq i$ and $k \neq j$.

This way, we exclude from negative pairs the views of the same image, related to index i and j , that are considered as positive pairs in the invariance methods. While we could consider these pairs as negative and still be following 3, we found that not using them as negative nor as positive leads to slightly better results. One hypothesis is that repulsing views that can be very close in the pixel

space (e.g. if the sampled augmentations modify weakly the original image) could induce training instability. One can notice that $g'_{\phi'}$ and u_{ψ} are learned simultaneously, thus they can influence each other during the training phase. We finally define the total loss of the model as:

$$\mathcal{L} = \mathcal{L}_{Invariance} + \lambda \mathcal{L}_{EquiMod}$$

with $\mathcal{L}_{EquiMod}$ being the loss Eq. 4 applied to all pairs, both (i, j) and (j, i) , of a batch and $\mathcal{L}_{Invariance}$ being the loss of the invariance baseline. λ is a hyperparameter that ponders the equivariant term of the loss.

2.3 IMPLEMENTATION DETAILS

We tested our module as a complement of 3 different baselines. The first one is SimCLR (Chen et al., 2020a) as it represents a contrastive approach to instance discrimination and performs well on CIFAR. The second one is BYOL (Grill et al., 2020), which offers a different kind of architecture (as it is a bootstrapping approach rather than a contrastive one) while having the highest top-1 accuracy with linear evaluation on ImageNet using a ResNet50 backbone in a self-supervised fashion. We also tested Barlow Twins (Zbontar et al., 2021) as it is not exactly a contrastive approach nor a bootstrapping one to illustrate the generality of our approach, yet limited to CIFAR10 due to computational limitation. Here are the details of each part of the architecture, including the baseline ones and our equivariance module:

- *Encoder*: we follow existing works and use a convolutional neural network for the encoder f_{θ} , more specifically deep residual architectures from He et al. (2016).
- *Invariance projection head*: for the projection head g_{ϕ} (and potential predictor as in BYOL Grill et al. (2020)), we used the same experimental setups as the original papers, except for SimCLR where we used a 3 layers projection head as in Chen & He (2021).
- *Equivariance projection head*: the setup of our projection head $g'_{\phi'}$ is a 3 layers Multi-Layer Perceptron (MLP), where each Fully-Connected (FC) layer is followed by a Batch Normalization (BN) and a ReLU activation, except the last layer which is only followed by a BN and no ReLU. Hidden layers have 2048 neurons each.
- *Equivariant predictor*: the predictor u_{ψ} is a FC followed by a BN. Its input is the concatenation of a representation of t and the input embedding z'_{ϕ} . More precisely t is encoded by a numerical learned representation of the parameters that fully define it. More precisely, we reduce the augmentation to a vector composed of binary values related to the use of transformations (for transformations applied with a certain probability) and numerical values corresponding to some parameters (of the parameterized transformations). This vector is projected in a 128d latent space with a perceptron learned jointly with the rest of the model, see Sec.A.1 for details and examples of this encoding. This way, the input dimension of the predictor is the dimension of the latent space plus the dimension of the encoding of augmentations, while the output dimension is the same as the latent space.

3 EXPERIMENTS

3.1 EXPERIMENTAL SETTINGS

In our experimentations, we tested our method on ImageNet (IN) (Deng et al., 2009) and CIFAR10 (Krizhevsky et al., 2009). As mentioned before, we have used our module as a complement to SimCLR, BYOL, and Barlow Twins, 3 state-of-the-art invariance methods with quite different ideas, to test the genericity of our module. For these methods, we used the same experimental setup as the original papers. As in previous works, while training on ImageNet we used a ResNet50 without the last FC, but while training on CIFAR10 we used the CIFAR variant of ResNet18 (He et al., 2016). For all our experimentations we used the LARS You et al. (2017) optimizer, yet, biases and BN parameters were excluded from both weight decay and LARS adaptation as in Grill et al. (2020)). Finally, we have fixed λ to 1 as it led to the best and more stable results.

3.1.1 SIMCLR

The model is trained for 800 epochs with 10 warm-up epochs and a cosine decay learning rate schedule. We have used a batch size of 4096 for ImageNet and 512 for CIFAR10, while using an initial learning rate of 2.4 for ImageNet (where we use 4.8 for SimCLR without EquiMod, as in the original paper) and 4.0 for CIFAR10. For the optimizer, we fix the momentum to 0.9 and the weight decay to $1e^{-6}$. Both the invariant and equivariant latent space dimensions have been set to 128. Finally, we use $\tau' = 0.2$ for our loss, but $\tau = 0.2$ on ImageNet $\tau = 0.5$ with CIFAR10 for the loss of SimCLR (we refer the reader to the original paper for more information about the loss of SimCLR (Chen et al., 2020a)).

3.1.2 BYOL

The model learned for 1000 epochs¹ (800 on CIFAR10) with 10 warm-up epochs and a cosine decay learning rate schedule. The batch size used is 4096 for ImageNet and 512 for CIFAR10. We have been using an initial learning rate of 4.8 for ImageNet (where we use 3.2 for BYOL without EquiMod, as in the original paper) while using 2.0 for CIFAR10. Momentum of the optimizer is set to 0.9 and weight decay to $1.5e^{-6}$ on ImageNet, but $1e^{-6}$ on CIFAR10. The invariant space has 256 dimensions while we keep our equivariant latent space to 128. Last, we use $\tau' = 0.2$ for our loss, and $\tau_{\text{base}} = 0.996$ for the momentum encoder of BYOL with a cosine schedule as in the original paper (once again, we refer the reader to the paper for more details (Grill et al., 2020)).

3.1.3 BARLOW TWINS

We tested our method with Barlow Twins only on CIFAR10 with the following setup: 800 epochs with 10 warm-up epochs and a cosine decay learning rate schedule, a batch size of 512, an initial learning rate of 1.2, a momentum of 0.9 and weight decay of $1.5e^{-6}$. Both the invariant and equivariant latent space has 128 dimensions, while we use $\tau' = 0.2$ for our loss and $\lambda_{\text{Barlow Twins}} = 0.005$ for the loss of Barlow Twins (as in the original paper (Grill et al., 2020)).

3.2 RESULTS

3.2.1 LINEAR EVALUATION

After training on either ImageNet or CIFAR10, we evaluate the quality of the learned representation with the linear evaluation which is usual in the literature. To this end, we train a linear classifier on top of the frozen representation, using the Stochastic Gradient Descent (SGD) for 90 epochs, which is sufficient for convergence, with a batch size of 256, a Nesterov momentum of 0.9, no weight decay, an initial learning rate of 0.2 and a cosine decay learning rate schedule.

Results of this linear evaluation are presented in Table 1, while some additional results are present in supplementary material Sec. A.3. Across all baselines and datasets tested, EquiMod increases the performances of all the baselines used, except BYOL while trained on 1000 epochs. Still, it is worth noting that under 100 and 300 epochs training (Sec. A.3), EquiMod improves the performances of BYOL. Overall, this supports the genericity of our approach, and moreover, confirms our idea that adding an equivariance task helps to extract more pertinent information than just an invariance task and improves representations. On CIFAR10, we achieve the second-best performance after E-SSL, yet, contrary to us, they tested their model on an improved hyperparameter setting of SimCLR.

3.2.2 EQUIVARIANCE MEASUREMENT

The way our model is formulated could lead to the learning of invariance rather than equivariance. Indeed, learning an invariant latent space as well as the function identity for u_{ψ} is an admissible solution. Therefore, to verify that our model is really learning equivariance, we define two metrics of equivariance Eq.5 and Eq.6. The first one evaluates the absolute displacement toward z'_i caused by the predictor u_{ψ} . One can see this as how much applying the augmentation t to z'_o in the latent space via u_{ψ} makes the resulting embedding \hat{z}'_i more similar to z'_i . This way, if our model is learning invariance, we should observe an absolute displacement of 0, as u_{ψ} would be the identity. On the

¹We also performed 100 and 300 epochs training, see Sec. A.3.

Method	ImageNet		CIFAR10	
	Top-1	Top-5	Top-1	Top-5
PIRL (Misra & Maaten, 2020)	63.6	-	-	-
E-SimCLR (Dangovski et al., 2021)	68.3 \ddagger	-	94.1	-
E-SimSiam (Dangovski et al., 2021)	68.6 \ddagger	-	94.2	-
SimCLR (Chen et al., 2020a)	69.3	89.0	-	-
SimSiam (Chen & He, 2021)	71.3	-	-	-
SwAV (w/o multi-crop) (Caron et al., 2020)	71.8	-	-	-
Barlow Twins (Zbontar et al., 2021)	73.2	91.0	-	-
VICReg (Bardes et al., 2021)	73.2	91.1	-	-
BYOL (Grill et al., 2020)	74.3	91.6	-	-
SimCLR*	71.57	90.48	90.96	99.73
SimCLR* + EquiMod	72.30	90.84	92.79	99.78
BYOL*	74.03	91.51	90.44	99.62
BYOL* + EquiMod	73.22	91.26	91.57	99.71
Barlow Twins*	-	-	86.94	99.61
Barlow Twins* + EquiMod	-	-	88.87	99.71

Table 1: **Linear Evaluation**; top-1 and top-5 accuracies (in %) under linear evaluation on ImageNet and CIFAR10 (symbols * denote our re-implementations, and \ddagger denote only 100 epochs training).

contrary, if it is learning equivariance, we should observe a positive value, meaning that the u_ψ plays its role in predicting the displacement in the embedding space caused by the augmentations. A negative displacement means a displacement in the opposite direction, in other words, it means that the predictor performs worse than the identity function. Furthermore, a small displacement does not mean poor equivariance, for instance, if z'_o is very similar to z'_i , the room for displacement is already very small. This is why we also introduce the second metric, which evaluates the relative displacement toward z'_i caused by u_ψ . It reflects by which factor applying the augmentation t to z'_o in the latent space via u_ψ makes the resulting embedding \hat{z}'_i less dissimilar to z'_i . Thus, if the model is learning invariance, we should see no reduction nor augmentation of the dissimilarity, thus the factor should remain at 1 while a model achieving equivariance would exhibit a positive factor.

$$\text{sim}(z'_i, \hat{z}'_i) - \text{sim}(z'_i, z'_o) \quad (5) \quad \frac{1 - \text{sim}(z'_i, z'_o)}{1 - \text{sim}(z'_i, \hat{z}'_i)} \quad (6)$$

Fig. 3 shows the absolute equivariance measured for each augmentation. Note that this is performed on a model already trained with the usual augmentation policy containing all the augmentations. If an augmentation induces a large displacement, it means the embedding is highly sensitive to the given augmentation. What we can see from Fig. 4, is that regardless of the dataset used, the model achieves poor sensitivity to horizontal flip and grayscale. However, on ImageNet, we observe a high sensitivity to color jitter as well as medium sensitivity to crop and gaussian blur. On CIFAR10 we observe a strong sensitivity to crop and a medium sensitivity to color jitter. Therefore, we can conclude that our model truly learns an equivariance structure, and that the learned equivariance is more sensitive to some augmentation such as crop or color jitter.

3.2.3 INFLUENCE OF THE ARCHITECTURES

We study how architectural variations can influence our model. More precisely, we explore the impact of the architecture of the $g'_{\phi'}$, u_ψ as well as the learned projection of t mentioned in Sec. A.1. To this end, we train models for each architectural variation on CIFAR10, and report the top-1 accuracies under linear evaluation, the results are reported in Table 2. What we observe in Table 2a, is that the projection head of the equivariant latent space benefits from having more layers, yet this effect seems to plateau at some point. These results are in line with existing works (Chen et al., 2020a). While testing various architectures for the equivariant predictor Table 2b, we note only small performance variations, indicating that u_ψ is robust to architectural changes. Finally, looking at

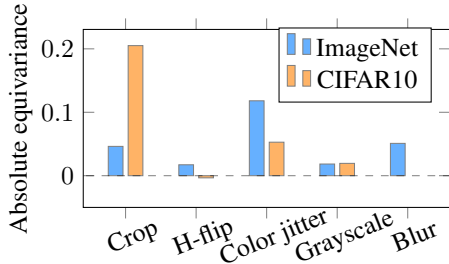


Figure 3: Absolute equivariance measure for each augmentation (the dashed line represents invariance).

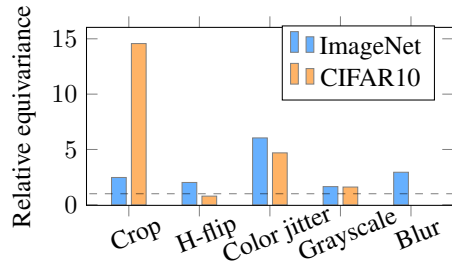


Figure 4: Relative equivariance measure for each augmentation (the dashed line represents invariance).

Table 2c, we observe that removing the projection of t only leads to a small drop in performance. On the contrary, complex architectures (two last lines) lead to a bigger drop in accuracy. Furthermore, while testing different output dimensions (lines 2 to 4), we note that using the same dimension for the output and for the equivariant latent space led to the highest results. Some more analysis on hyperparameter variations of our model, such as λ , batch size, or τ' can be found in Sec. A.2.

Layers in $g'_{\phi'}$	Top-1	Layers in u_{ψ}	Top-1	Layers in the projection of t	Top-1
None	88.46	1 †	92.79	None	92.50
1	91.58	2 (H: 16-d)	92.67	1 (O: 16-d)	92.57
2	92.58	2 (H: 128-d)	92.59	1 (O: 128-d) †	92.79
3 †	92.79	2 (H: 2048-d)	92.70	1 (O: 2048-d)	92.50
				2 (H: 16-d; O: 128-d)	92.47
				2 (H: 128-d; O: 128-d)	92.13
				2 (H: 2048-d; O: 128-d)	92.05

(a) **Equivariance projection head** (b) **Equivariant predictor** (c) **Augmentation projector**

Table 2: Top-1 accuracies (in %) under linear evaluation on CIFAR10 for some architectural variations of our module. H stands for hidden layer, O for output layer, † denotes default setup.

4 RELATED WORK

Most of the recent successful methods of SSL of visual representation learn a latent space where embeddings of augmentations from the same image are learned to be similar. Yet, such instance discrimination tasks admit simple constant solutions. To avoid such collapse, recent methods implement diverse tricks to maintain a high entropy for the embeddings. Grill et al. (2020) rely on a momentum encoder as well as an architectural asymmetry, Chen & He (2021) depend on a stop gradient operation, Zbontar et al. (2021) rely on a redundancy reduction loss term, Bardes et al. (2021) rely on a variance term as well as a covariance term in the loss, Chen et al. (2020a) use negative pairs repulsing sample in a batch. In this work, our task also admits collapse solutions, thus we make use of the same negative pairs as in Chen et al. (2020a) to avoid such collapse. The most recent methods are creating pairs of augmentations to maximize the similarity between those pairs. However, our addition does not rely on pairs of augmentations, and only needs a source image and an augmentation. This is similar to Misra & Maaten (2020) which requires to have a source image and an augmentation, however, they use these pairs to learn an invariance task while we use them to learn an equivariance task.

Our approach is part of a line of recent works, which try to perform additional tasks of sensitivity to augmentation while learning an invariance task. This is the case of E-SSL (Dangovski et al., 2021), which simultaneously learns to predict rotations applied to the input image while learning an invariance pretext task. This way, their model learns to be sensitive to the rotation transformation, usually

not used for invariance. Where this can be considered as a form of equivariance (a rotation in input space produces a predictable displacement in the prediction space) this is far from the equivariance we explore in this paper. Indeed, E-SSL sensitivity task can be seen as learning an instance-invariant pretext task, where for any given input, the output represents only the augmentation (rotation) used. Here, we explore equivariance sensitivity both to images and to augmentations. Moreover, we only consider sensitivity to the augmentations used for invariance. In LooC (Xiao et al., 2020), authors propose to use as many different projection heads as there are augmentations and learn each of these projection heads to be invariant to all but one augmentation. This way the projection heads can implicitly learn to be sensitive to an augmentation. Still, they do not control how this sensitivity occurs, where we explicitly define an equivariance structure for the augmentations-related information. Note that a work has tried to tackle the trade-off from the other side, by trying to reduce the shortcut learning occurring, instead of adding sensitivity to augmentations. Robinson et al. (2021) shows that shortcut learning occurring in invariant SSL is partly due to the formulation of the loss function and proposes a method to reduce shortcut learning in contrastive learning.

Some other works have also successfully used equivariance with representation learning. For instance, Jayaraman & Grauman (2015) uses the same definition of equivariance as us and successfully learns an equivariant latent space tied to ego-motion. Still, their objective is to learn embodied representations as well as using the learned equivariant space, in comparison we only use equivariance as a pretext task to learn representations. Moreover, we do not learn equivariance on the representations, but rather on a non-linear projection of the representations. Lenc & Vedaldi (2015) learns an equivariant predictor on top of representations to measure their equivariance, however, to learn that equivariance, they require the use of strong regularizations.

5 CONCLUSION AND PERSPECTIVES

Recent successful methods for self-supervised visual representation rely on learning a pretext task of invariance to augmentations. This encourages the learned embeddings to discard information related to transformations. However, this does not fully consider the underlying dilemma that occurs in the choice of the augmentations: strong modifications of images are required to remove some possible shortcut solutions, while information manipulated by the augmentation could be useful to some downstream tasks. In this paper, we have introduced EquiMod, a generic equivariance module that can complement existing invariance approaches. The goal of our module is to let the network learn an appropriate form of sensitivity to augmentations. It is done through equivariance via a module that predicts the displacement in the embedding space caused by the augmentations. Our method is part of a research trend that performs sensitivity to augmentations. Nonetheless, compared to other existing works, we perform sensitivity to augmentations also used for invariance, therefore reducing the trade-off, while defining a structure in our latent space via equivariance.

Testing EquiMod across multiple invariance baseline models and datasets almost always showed improvement under linear evaluation. It indicates that our model can capture more pertinent information than with just an invariance task. In addition, we observed a strong robustness of our model under architectural variations, which is a non-negligible advantage as training such methods is computationally expensive, and so does the hyperparameters exploration. When exploring the sensitivity to the various augmentations, we noticed that the latent space effectively learns to be equivariant to almost all augmentations, showing that it captures most of the augmentations-related information.

For future work, we plan on testing our module on more baseline models or even as a standalone. As EquiMod almost always improved the results in our tests, it suggests that EquiMod could improve performances on many more baselines and datasets. Then, since E-SSL adds sensitivity to rotation, yet still does not consider sensitivity to augmentations used for invariance, it would be interesting to study if combining EquiMod and E-SSL can improve even further the performances. Another research axis is to perform an in-depth study of the generalization and robustness capacity of our model. To this end, we want to explore its capacity for transferability (fine-tuning) and few-shot learning, both on usual object recognition datasets, but also on more challenging datasets containing flowers and birds as in Xiao et al. (2020). Since the trade-off theoretically limits the generalization on the learned representation, and since we reduce the effect of the trade-off, we hope that EquiMod may show advanced generalization and robustness properties. On a distant horizon, the equivariant structure learned by our latent space may open some interesting perspectives related to world model.

ACKNOWLEDGMENTS

This work was performed using HPC resources from GENCI-IDRIS (Grant 2021-AD011013160 and 2022-A0131013831) and GPUs donated by the NVIDIA Corporation. We gratefully acknowledge this support.

REFERENCES

- Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.
- Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.
- Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020b.
- Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.
- Rumen Dangovski, Li Jing, Charlotte Loh, Seungwook Han, Akash Srivastava, Brian Cheung, Pulkit Agrawal, and Marin Soljačić. Equivariant Contrastive Learning. *arXiv preprint arXiv:2111.00899*, 2021.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.
- Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. Publisher: Nature Publishing Group.
- Jean-Bastien Grill, Florian Strub, Florent Alché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, and Mohammad Gheshlaghi Azar. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.
- Tianyu Hua, Wenxiao Wang, Zihui Xue, Sucheng Ren, Yue Wang, and Hang Zhao. On feature decorrelation in self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9598–9608, 2021.
- Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1413–1421, 2015.
- Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *arXiv preprint arXiv:2110.09348*, 2021.
- Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

- Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 991–999, 2015.
- Ran Liu. Understand and Improve Contrastive Learning Methods for Visual Representation: A Review. *arXiv preprint arXiv:2106.03259*, 2021.
- Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.
- Joshua Robinson, Li Sun, Ke Yu, Kayhan Batmanghelich, Stefanie Jegelka, and Suvrit Sra. Can contrastive learning avoid shortcut solutions? *Advances in neural information processing systems*, 34:4974–4986, 2021.
- Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.
- Tete Xiao, Xiaolong Wang, Alexei A. Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. *arXiv preprint arXiv:2008.05659*, 2020.
- Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.
- Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

A APPENDIX

A.1 ENCODING OF THE AUGMENTATIONS

We use the classical augmentations of the literature, which depend on the dataset and model used, applied in the given order:

- Resized Crop: crop a subregion of the image;
- Horizontal flip: flip the image with a given probability;
- Color jitter: jitter the image on different aspects with a random order (brightness, saturation, contrast, and hue) and with a given probability;
- Gray-scale: gray-scale the image with a given probability;
- Gaussian blur (not used with CIFAR10 except in BYOL): blur the image using a sampled σ and with a given probability;
- Solarize (applied only with BYOL): solarize the image with a given probability.

We refer the reader to the original papers (Chen et al., 2020a; Grill et al., 2020; Zbontar et al., 2021) to know how the different methods parameterize these augmentations (e.g. values of the probability, or intervals of values sampled, as factors in color jitter).

To encode these augmentations, we represent them by a numerical vector where some of the components are binary values related to the use of augmentations (for those applied with some probability) and some others are numerical values corresponding to some parameters (of the parameterized transformations). We only consider the corresponding augmentations w.r.t. the tested dataset and model. For each of these considered augmentations except crop, we define an element valued at 1 when the augmentation is performed and valued at 0 otherwise (since each augmentation, but the crop, is applied with a given probability it may be applied or not). Then, some augmentations require additional elements. To this end, we define elements to represent these parameters using the following direct ways (note that when a parametrized augmentation is not applied due to its probability of application, its numerical components are set to some predefined default values):

- *Resized Crop* (4 elements): x and y coordinates of the top-left pixel of the crop as well as width and height of the crop.

- *Color Jitter* (8 elements): the jitter factors for brightness, saturation, contrast, and hue (1, 1, 1, 0 is the default encoding if color jitter is not applied), as well as their order of application. More precisely, to encode the order of modification, we use the following mapping {0 : brightness, 1 : contrast, 2 : saturation, 3 : hue}. For instance an encoding with "1, 3, 2, 0" would mean that contrast jitter is first applied, then hue, contrast, and finally brightness (0, 1, 2, 3 is the default encoding if color jitter is not applied).
- *Gaussian Blur* (1 element): the value of sigma used (0 if blur is not applied).

At this point, we have a numerical vector that represents which augmentations are applied or not and what are their parameters if any, see the following Sec. A.1.1 and Sec. A.1.2 for some examples. We then normalize this vector component-wise using experimental mean and standard deviation computed over many examples, and we use a perceptron to project the constructed vector into a 128d latent space. This perceptron is learned jointly with the rest of the model.

A.1.1 EXAMPLE 1

Here is an example of one transformation applied during the learning of BYOL on ImageNet.

Let's consider the randomly generated transformation composed of the following augmentations:

- Crop at coordinates $x, y=(12, 9)$ with width, height of (120, 96);
- Probabilistic horizontal flip not triggered;
- Probabilistic color jitter triggered with factors and order of: hue -0.09, contrast 1, saturation 0.84, brightness 1.13;
- Probabilistic gray-scale triggered;
- Probabilistic blur not triggered;
- Probabilistic solarize not triggered;

According to A.1, for the binary part representing the performed augmentations we have [0, 1, 1, 0, 0] for [No H-Flip, Yes Color jitter, Yes Gray-scale, No Blur, No Solarize] (one per augmentation, except crop which is always performed), and for the parameterized transformations : [12, 9, 120, 96, 1.13, 1, 0.84, -0.09, 3, 1, 2, 0, 0] for [Crop X, Crop Y, Crop Width, Crop Height, Brightness Factor, Contrast Factor, Saturation Factor, Hue Factor, Index of the First Color Modification Applied, Index of the Second Color Modification Applied, Index of the Third Color Modification Applied, Index of the Fourth Color Modification Applied, Default value for sigma (as blur is not triggered)]

Finally, this gives us the 18d vector [0, 1, 1, 0, 0, 12, 9, 120, 96, 1.13, 1, 0.84, -0.09, 3, 1, 2, 0, 0], which is then normalized and given to a perceptron to project it to a 128d vector.

A.1.2 EXAMPLE 2

And here is another example this time with SimCLR on CIFAR10 (which uses a different augmentation policy, thus solarization and blur are not considered).

Let's consider the randomly generated transformation composed of the following augmentations:

- Crop at coordinates $x, y=(1, 2)$ with width,height of (24, 27);
- Probabilistic horizontal flip triggered;
- Probabilistic color jitter not triggered;
- Probabilistic gray-scale not triggered;

For the binary part representing the performed augmentations, we have [1, 0, 0] for [Yes H-Flip, No Color jitter, No Gray-scale]. And for the parametrized transformations : [1, 2, 24, 27, 1, 1, 1, 0, 0, 1, 2, 3] for [Crop X, Crop Y, Crop Width, Crop Height, Brightness Factor (Default), Contrast Factor (Default), Saturation Factor (Default), Hue Factor (Default), Index of the First Color Modification (Default), Index of the Second Color Modification (Default), Index of the Third Color Modification (Default), Index of the Fourth Color Modification (Default)]. Note the default values for all the parameters of the color jitter which is not triggered.

This gives us the 15d vector [1, 0, 0, 1, 2, 24, 27, 1, 1, 1, 0, 0, 1, 2, 3], which is then normalized and given to a perceptron to project it to a 128d vector.

A.2 INFLUENCE OF HYPERPARAMETERS (λ , τ' AND BATCHSIZE)

In this section, similarly to Sec.3.2.3, we study how variations of minor hyperparameters can influence our model. To that purpose, we train models on CIFAR10 for each hyperparameter modification and present the top-1 accuracy under linear evaluation.

We first inspect the influence of the λ , the weighting factor between our equivariance loss and the invariance baseline loss. One can see Table 3a that when λ is small (< 1) there is a drop in performance. As λ can be seen as weighting the importance between the equivariance and the invariance terms of the loss, this confirms that our model learns better features when our equivariance addition is considered with at least the same importance as the invariance task. On the opposite, interestingly, where λ is set to high values such as 5 or 10, we do not observe a clear modification of the performance. This tends to indicate that there is no degradation of the representation when the equivariance is prioritized.

Then we study the temperature hyperparameter of the NT-Xent loss that we use to learn equivariance. Similarly to what is reported in Chen et al. (2020a), we find Table 3b that the optimal values to be around 0.2 and 0.5.

Finally, we explore the impact of the batch size on the learned representations. This hyperparameter directly determines the number of negative pairs, therefore it highly influences the learning dynamic. We observe Table 3c a decrease in performance where the batch size is too small (≤ 256) or too big (≥ 1024). Once again, these findings are in line with the literature (Chen et al., 2020a).

λ Factor	Top-1	Temperature τ'	Top-1	Batch size	Top-1
0	90.96	0.05	92.13	64	92.23
0.1	92.07	0.1	92.13	128	92.24
0.2	92.31	0.2 †	92.79	256	92.38
0.5	92.37	0.5	92.31	512 †	92.79
1 †	92.79	1	92.14	1024	92.23
2	92.33				
5	92.81				
10	92.66				

(a) **Weighting factor between equivariance and invariance losses**

(b) **Temperature of the NT-Xent used in our equivariance loss**

(c) **Batch size**

Table 3: Top-1 accuracies (in %) under linear evaluation on CIFAR10 for some hyperparameter variations of our module. † denotes default setup.

A.3 ADDITIONAL RESULTS

The Table 4 shows the impact of the number of training epochs on the results of the linear evaluation of BYOL with and without EquiMod.

Method	ImageNet		CIFAR10	
	Top-1	Top-5	Top-1	Top-5
BYOL* (100 epochs)	62.09	84.01	-	-
BYOL* + EquiMod (100 epochs)	65.55	86.74	-	-
BYOL* (300 epochs)	71.34	90.35	-	-
BYOL* + EquiMod (300 epochs)	72.03	90.77	-	-
BYOL* (1000 epochs)	74.03	91.51	90.44	99.62
BYOL* + EquiMod (1000 epochs)	73.22	91.26	91.57	99.71

Table 4: **Linear Evaluation**; top-1 and top-5 accuracies (in %) under linear evaluation on ImageNet and CIFAR10 (symbols * denote our re-implementations).

Which Structural Patterns Emerging from Instance Discrimination Benefit Linear Evaluation?

Anonymous submission

Abstract

Instance discrimination tasks have propelled self-supervised learning of representations to approach the performance of supervised baselines. These methods typically involve aligning augmentations of the same instance while ensuring, through various techniques, that the overall distribution of representations does not collapse. Despite some hypotheses suggesting why optimization for instance discrimination yields linearly separable representations that facilitate classification, the precise underlying mechanisms remain elusive. In this work, we shift focus from formally explaining the emergence of linear structures to empirically studying the shared structural components across different self-supervised learning methods and identifying which elements contribute to high classification accuracy under linear evaluation. We employ a comprehensive set of structural descriptors to quantitatively describe the learned representations, analyzing these descriptors at various local and global scales, different models, and depths. We hope that this work will pave the way for designing methods that explicitly optimize for the emergence of beneficial structural patterns in learned representations.

Introduction

In recent years, Self-Supervised Learning (SSL) has emerged as a powerful approach for learning visual representations without the need for labeled data (Caron et al. 2020; Chen et al. 2020a,b; Chen and He 2021; Bardes, Ponce, and LeCun 2021; Grill et al. 2020; He et al. 2020; Misra and Maaten 2020; Zbontar et al. 2021). A key advantage of SSL is its ability to learn directly from raw data, making it highly versatile and applicable across a wide range of tasks and domains. It is also data and compute efficient due to its strong transferability (Zhao et al. 2021). Unlike supervised learning, which requires extensive labeled datasets and may bias the representation toward specific tasks (Geirhos et al. 2020), SSL aims at leveraging the inherent structure in the data, enabling the model to learn more generalizable features.

Self-supervised methods, particularly those based on instance discrimination (Caron et al. 2020; Chen et al. 2020a,b; Chen and He 2021; Bardes, Ponce, and LeCun 2021; Grill et al. 2020; He et al. 2020; Misra and Maaten 2020; Zbontar et al. 2021), have achieved remarkable success, reaching performance levels comparable to supervised

learning on downstream tasks such as image classification and object detection. These methods typically involve creating multiple augmented views of the same data instance and encouraging the model to learn similar representations for these views while maintaining diversity across different instances.

Despite the success of instance discrimination methods, the underlying mechanisms that enable these models to produce representations highly effective for downstream tasks remain poorly understood. Theoretically, such pretext tasks are expected to learn a structure that is not necessarily conducive to tasks like classification. Invariance in instance discrimination methods is learned at the instance level rather than the class level, as class labels are unknown in an SSL context. This means that each instance is treated as its own category. To avoid collapse, the model is often regularized to make representations of different instances as distinct, or even orthogonal, as possible. This approach theoretically hinders generalization, as it discourages the model from learning shared embeddings across instances that belong to the same class.

However, in practice, the representations learned through instance discrimination often generalize well to downstream tasks, particularly in classification, where such structure should theoretically be a disadvantage. Moreover, when the loss is applied through a projection head (a non-linear transformation of the representation), making the representations less explicitly structured by the loss, performance tends to improve. Note that these properties are consistently observed across different instance discrimination methods. This apparent contradiction has led to a deeper exploration of these phenomena, prompting several theoretical works that propose hypotheses, including the role of the projection head in shaping the learned representations (Chen et al. 2020a; Xue et al. 2024) and the concept of the chaos ladder to explain how classification could emerge from instance discrimination (Wang et al. 2022).

In this work, we take an experimental approach to empirically analyze the structural components that emerge in the latent spaces of different instance discrimination-based methods. Our objective is to identify both shared and distinct structural patterns across various SSL approaches and determine which of these patterns most effectively contribute to high classification accuracy in downstream tasks. To achieve

this, we employ a diverse set of structural descriptors that capture different facets of the latent space’s organization. By systematically analyzing and comparing these descriptors at multiple scales, across different models, and at varying network depths, we aim to provide a practical understanding of what emerging structures contribute to the success of SSL. Ultimately, our findings could inform the design of new SSL methods that explicitly optimize for the emergence of beneficial structural patterns, leading to even more powerful and generalizable representations.

Related Work

Instance discrimination has become an important part of self-supervised learning (SSL) due to its effectiveness in learning robust visual representations without labeled data. Key methods in this area (Caron et al. 2020; Chen et al. 2020a,b; Chen and He 2021; Bardes, Ponce, and LeCun 2021; Grill et al. 2020; He et al. 2020; Misra and Maaten 2020; Zbontar et al. 2021) share the common goal of aligning augmented views of the same instance while maintaining a diverse and high-entropy representation space to prevent collapse.

SimCLR (Chen et al. 2020a) employs a contrastive learning framework, utilizing negative pairs within a batch to push apart embeddings from different instances while pulling together embeddings of augmented views of the same instance. SimSiam (Chen and He 2021) builds on this by removing the need for negative pairs altogether, relying on a stop-gradient operation and a simple siamese network architecture to prevent collapse and maintain meaningful representations. BYOL (Grill et al. 2020), similarly to SimSiam, eliminates the need for negative pairs by introducing a momentum encoder, which, along with an asymmetric architecture, helps maintain the diversity of representations. PIRL (Misra and Maaten 2020) introduces a pretext task that learns invariant representations by contrasting original images with augmented ones, ensuring that the model captures both high-level semantics and low-level details. Barlow Twins (Zbontar et al. 2021) focuses on redundancy reduction across embeddings by minimizing the redundancy in the output features, while VICReg (Bardes, Ponce, and LeCun 2021) adds constraints on variance and covariance to ensure the learned representations do not collapse and remain useful for downstream tasks. Despite the differences in their approaches, all these methods achieve competitive performance on downstream tasks, such as classification, suggesting that they effectively capture essential features of the data. However, the precise mechanisms by which these instance-level discriminations lead to representations that generalize well to class-level tasks remain an open question.

The apparent paradox of how instance discrimination leads to strong classification performance has been the focus of several theoretical investigations. One hypothesis emphasizes the role of the projection head (Chen et al. 2020a; Xue et al. 2024), which is applied to the representations before the loss is computed. The projection head’s benefit may lie in its ability to allow the representations to retain more information related to the augmentations, rather than forcing them to specialize solely toward the invariance task.

Another theoretical approach involves the concept of the “ladder of chaos” (Wang et al. 2022), which provides a novel explanation for how contrastive learning transitions from instance-level discrimination to effective class-level generalization. This theory suggests that aggressive data augmentations in contrastive learning create “augmentation overlap”, where intra-class samples become more similar due to the augmentations. This overlap introduces “chaos” among intra-class samples. The contrastive loss, which aligns augmented views of the same instance, helps the model gradually cluster these intra-class samples together, effectively “climbing the ladder of chaos”. However, it has been shown that this theory alone may not fully explain the underlying mechanisms (Saunshi et al. 2022).

Beyond theoretical explanations, recent studies have empirically investigated some of the structural properties of the latent spaces learned by SSL methods, shedding light on how these properties impact downstream performance. For instance, Purushwalkam and Gupta (2020) demonstrate that SSL methods often unexpectedly learn invariances that are heavily influenced by dataset biases, such as those present in ImageNet, and primarily focus on certain invariances like occlusion when transferred to different domains.

In another study, Zhang, Lu, and Xuan (2024) observed that SSL representations tend to organize images such that nearest neighbors in the latent space are often not of the same class, unlike in supervised learning. They also find that in SSL, similar representations are more closely related in pixel space. Similarly, Grigg et al. (2021) explored how representations learned by SSL and supervised learning (SL) diverge across network layers. Their findings suggest that while both SSL and SL models improve in performance as layers deepen, the representations become increasingly dissimilar. Notably, the similarity between SSL and SL representations collapses after the projection head in SSL models, highlighting the distinct nature of the features learned through self-supervision.

Finally, Cole et al. (2022) examined key factors that influence the effectiveness of contrastive learning, such as the quantity and quality of data, the pretraining domain, and the granularity of tasks. Their study reveals that SSL methods are sensitive to the domain of the pretraining data and the resolution of images, and they struggle with fine-grained tasks compared to supervised baselines.

While these studies offer important insights into the structural properties and generalization capabilities of SSL methods, they often focus on individual approaches or specific aspects of the latent space, sometimes tailoring their findings to particular model designs (Zhang, Lu, and Xuan 2024). This focus leaves gaps in our understanding of the broader structural patterns that emerge across different SSL techniques. Addressing this gap, our work takes a more holistic approach, systematically analyzing a wide array of structural descriptors across multiple SSL models and network layers. By identifying the structural characteristics that consistently correlate with high classification accuracy, our study provides a unified approach to better understand the structural factors that contribute to the success of instance discrimination methods.

Method

In this work, we study the impact of latent space structure on downstream task performance and identify the specific structural patterns that contribute to this performance. Extracting and defining structural patterns from a latent space remains an open challenge. For this work, we have started by defining the properties of the latent space that could be of interest based on the literature, followed by determining appropriate descriptors that can verify and quantify the presence of these properties. These descriptors serve as tools to analyze the structure of the latent space within the selected subsets of data points. To keep the study comprehensible and facilitate the interpretation of the resulting structural patterns, we focus on usual and widely used structural descriptors, each of which can be summarized by a few scalar values with easily understandable meanings.

We then compute these descriptors for subsets of data points to correlate them with the performance metrics of the corresponding subset of samples. We naturally define these subsets as the various classes within the dataset, as this provides a clear basis for comparison. However, it is important to note that the methods are learned in a self-supervised manner, meaning that the structural patterns emerging within these subsets arise without any explicit class labels or supervision. This approach is driven by the hypothesis that local structural variations may exist and could be linked to the model’s performance on data points within these local areas. For example, samples from a particular class might exhibit a structure that differs from other classes, leading to varying accuracy levels. Such structural variations are plausible given the instance discrimination objective, which has no inherent reason to produce a homogeneous space.

Structure Descriptors

We refer to “structural descriptors” as metrics that quantify specific characteristics of the latent space, providing insight into the underlying structure learned by the model. These descriptors are chosen based on their demonstrated ability in the literature to capture meaningful aspects of the data’s distribution, organization, and relationships.

Since the structures within the latent space are not explicitly optimized during training, they emerge naturally and unpredictably. This unpredictability necessitates exploring a wide variety of descriptors to capture different facets of the latent space’s organization. By analyzing the following diverse set of descriptors, we aim to uncover the most comprehensive patterns that may correlate with and contribute to the model’s success in downstream tasks.

Principal Component Analysis We apply Principal Component Analysis (PCA) to the latent space to quantify its structure. Key metrics include the number of dimensions required to capture 10%, 50%, 90%, 95%, and 99% of the variance, providing insight into the compactness of the latent space. We also compute the effective dimensionality, which reflects the overall spread of variance, and the ratio of variance captured by the first to the last principal component, highlighting the dominance of leading components.

Additionally, the Gini coefficient is calculated to assess the inequality in variance distribution across components. These metrics collectively characterize the dimensionality within the latent space.

Statistics on the Activation Distribution We compute various statistics on the activation distributions within the latent space to gain insights into the utilization and spread of neuron activations. Specifically, we calculate the average, maximum, minimum, and median values for the mean, standard deviation, variance, skewness, and kurtosis of the activations. These metrics help us understand the distribution of activations across neurons, revealing how balanced or skewed the activations are, and how much the model relies on specific neurons or dimensions within the latent space.

Covariance We assess the covariance matrix to understand the linear relationships between neurons within the latent space. Specifically, we examine the trace (total variance), determinant (overall spread), and the average off-diagonal values, which indicate the average linear correlation between different dimensions. These metrics help determine how effectively the latent space utilizes its dimensions and whether there are redundancies.

Mutual Information We calculate mutual information between neuron pairs to measure the dependence between different dimensions of the latent space. This includes computing the redundancy index, which summarizes the overall degree of information overlap across the dimensions. Additionally, we analyze the average, maximum, and variance of mutual information values, along with their skewness and kurtosis, to understand the distribution of dependency across neuron pairs. These metrics provide insights into how much information is shared between neurons and whether certain dimensions carry redundant information.

Similarity Metrics We assess the relationships between data points within the latent space by computing various cosine similarity metrics, including the average, median, highest, and lowest similarities among data points, as well as the similarity of points to the average representation. These metrics help evaluate the compactness, cohesion, and distinctiveness of the representations, indicating how effectively similar data points are grouped and how distinct different representations are from one another.

Nearest Neighbors We assess the local density and proximity of data points within the latent space by analyzing their nearest neighbors. This involves calculating the average cosine distance to the k-nearest neighbors and evaluating metrics such as median distance, variance, and the slope of the distance distribution in both logarithmic and non-logarithmic scales. Additionally, we track the number of neighbors within specific distance thresholds and compute average average distances for particular percentiles of neighbors, offering a detailed view of the separation and cohesion within the latent space.

Fractal and Correlation Dimension We assess the complexity and dimensionality of the latent space using correlation and fractal dimensions (Grassberger and Procaccia

1983a,b). The correlation dimension estimates the intrinsic dimensionality by examining how the number of point pairs within a given cosine distance scales as the distance increases, providing insight into whether data points are uniformly distributed or reside on a lower-dimensional manifold. The fractal dimension evaluates how the latent space fills as we zoom in at different scales by comparing the latent space to a set of uniform points on a hypersphere. Both metrics are summarized using the slope of the log-log plot and the average proximity of points, offering a detailed view of the latent space’s self-similarity and scaling properties.

Clustering Analysis We analyze the latent space structure by grouping data points into distinct clusters using spherical k-means clustering, focusing on the first 50 principal components. The optimal number of clusters is determined via the elbow method. Key metrics include the optimal number of clusters, median cluster size, variance in cluster sizes, and average cosine similarity to cluster centroids. These metrics provide insight into the compactness, separation, and overall organization of the latent space, helping to identify how well the representations form distinct groups.

Connected Components We analyze the connected components within the latent space by evaluating the subgraphs formed in a binarized similarity matrix (Tarjan 1972). By varying the similarity threshold, we identify distinct connected regions and examine metrics such as the threshold required to reach specific percentages of connected components (e.g., 10%, 50%, 90%, 95%, 99%), as well as the number, median size, standard deviation, and variance of these components. This analysis reveals the level of fragmentation or connectedness within the latent space, providing insight into how data points cluster and whether the space contains isolated or densely connected regions.

Community Detection We perform community detection within the latent space by treating the similarity matrix as a weighted graph, where edges represent the strength of connections between data points. Using the Louvain method (Blondel et al. 2008), we identify clusters, or communities, and analyze key metrics including the number of communities, median community size, standard deviation, variance of community sizes, and the modularity score. This analysis uncovers the hierarchical and modular structure of the latent space, providing insights into how data points form complex, interrelated groups.

Instance Discrimination Baselines

We selected SimCLR (Chen et al. 2020a) and BYOL (Grill et al. 2020) as the baseline models for our analysis, as they represent two distinct approaches to self-supervised learning, both centered on instance discrimination. SimCLR is foundational in contrastive learning, focusing on distinguishing between positive and negative pairs to generate meaningful representations. BYOL, in contrast, eliminates the need for negative samples by using a momentum encoder in one of the siamese branches, solely aligning augmented views to learn representations, and has demonstrated high accuracy in downstream tasks. Thanks to their differing

mechanisms, the shared structural patterns of these models may reveal shared and distinct trends inherent to instance discrimination, allowing us to identify the core structures consistently leveraged across varied SSL paradigms.

In our experiments, we utilize a ResNet-50 backbone combined with a two-layer projection head, following the original implementations of each method. The models are trained on the ImageNet dataset. To gain insights into the evolution of the latent space structure, we analyze the learned representations at multiple stages of the network. Specifically, we examine the outputs from both the backbone and the projection head. This multi-layer analysis provides a comprehensive view of how the structure of the latent space develops as the data flows through different layers of the model. The representations are then evaluated through linear classification on frozen features extracted from these layers, following the standard procedures outlined in the original implementations. Note that, unlike the original approaches, we also compute the accuracy of the projection head output to examine how its structure relates to its classification ability.

Dataset and Subsets Selection

ImageNet Validation Set For studying the representations, we utilize the ImageNet validation set for several reasons. Firstly, using the validation set ensures that the representations are not learned by heart, which can be a risk with the training set, especially in self-supervised learning where overfitting can obscure the true performance of the learned representations. Secondly, the validation set provides a more realistic evaluation of how well the learned representations generalize to unseen data, which is crucial for assessing the true utility of the latent structures. Additionally, the validation set is significantly smaller than the training set, making it more manageable for computationally expensive structural descriptors. This smaller size allows for thorough and efficient analysis across multiple layers and descriptors without compromising the breadth of the study. By using the validation set, we can focus our resources on extracting meaningful insights while still covering a wide range of structural properties within the latent space.

Sample Subsets Explored To gain deeper insights into the latent space, we analyze specific subsets of data points derived from the classes within the dataset. These subsets include all samples belonging to a class (intra-class), those predicted as belonging to that class, as well as true positives, false positives, and false negatives. Additionally, we explore the interaction between different subsets, i.e. the relationship between samples from a class and samples that do not belong to that class (inter-class). This approach enables us to examine how the structural properties of these subparts of the latent space correlate with model performance.

To account for potential biases in descriptors due to varying set sizes (such as true positives, false positives, false negatives, etc.), we applied a threshold-based approach. For each threshold value, we only considered classes where the set size met or exceeded the threshold. In cases where the set size exceeded the threshold, we randomly subsampled

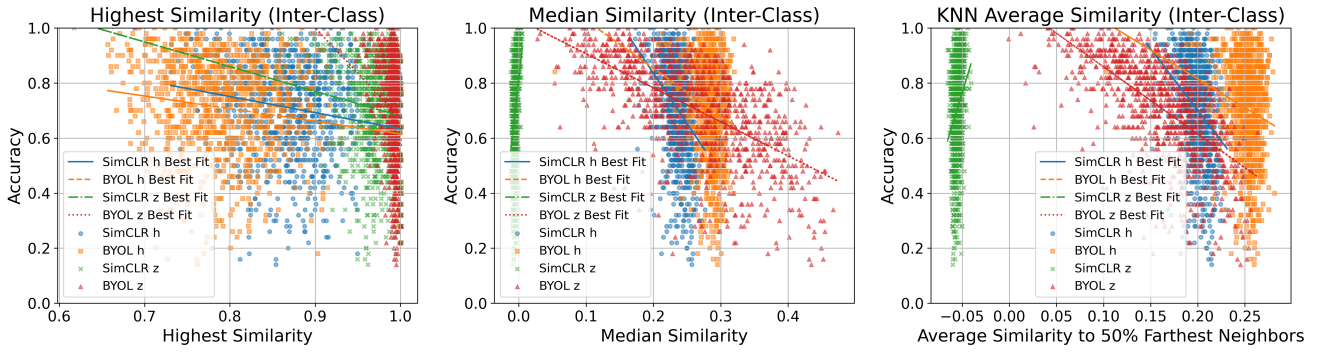


Figure 1: Legends with h refer to measurements done on the output of the backbone, while z refers to the output of the projection head; the best-fit line corresponds to a linear regression of the points. **Left:** The plot shows the relationship between the highest similarity among inter-class samples and classification accuracy. Higher accuracy is generally associated with lower maximum similarity, suggesting that classes that are more distinct from each other in the latent space tend to perform better in classification tasks. **Center:** This subplot illustrates the correlation between median similarity among inter-class samples and accuracy. Lower median similarity indicates better class separability, which in turn enhances classification performance. **Right:** The plot depicts how the average similarity to the 50% farthest neighbors affects accuracy. Close-to-zero similarity indicate better separation between classes, leading to improved classification outcomes.

to standardize the number of samples across classes. This was crucial, for instance, in preventing biases where a larger number of true positive samples, which typically correlate with higher accuracy, could artificially inflate metrics like cluster size. By standardizing set sizes, we aimed to ensure that the structural patterns observed were not merely artifacts of differing sample quantities.

Results

In this section, we present the findings from our extensive analysis of the latent space structures across multiple layers (i.e., backbone and projection head outputs), as well as two different self-supervised methods, SimCLR and BYOL. Given the large number of structural descriptors and layers studied, our amount of quantitative values is vast. Therefore, we focus on highlighting the most significant, relevant, and surprising patterns that emerge from the data.

The first surprising result is the total lack of a relationship between structure and performance within the sets corresponding to false negative and false positive samples. Across all the descriptors tested, these sets consistently showed non-significant correlations with their class’s accuracy, unlike true positives. This suggests that misclassified samples, as identified during the linear evaluation step, may already be outliers in the learned representation space. However, these outliers do not appear to belong to any specific outlier structure that could be regularized or identified. Instead, they seem to exist without fitting into any recognizable pattern.

The implications of this observation are twofold. First, it challenges the assumption that false negatives and false positives can be addressed simply by refining decision boundaries or applying more sophisticated regularization techniques. Since these samples lack any inherent structure that correlates with accuracy, traditional methods relying on

identifying and correcting specific patterns may prove ineffective. Second, this raises a critical question about the nature of the representations themselves: if the learned representations are unable to capture the subtleties that distinguish these outliers, it might indicate a fundamental limitation in the self-supervised learning methods or in the nature of these samples. This suggests that the models might primarily focus on optimizing for the majority of the data while neglecting edge cases that do not conform to the dominant structures.

Subsequently, the differences between sets comprising positive samples, predicted positives, and true positives are negligible (with true positives being very slightly cleaner visually in terms of correlation). This is likely due to the fact that these sets differ only in the inclusion or exclusion of false negatives and false positives, which do not significantly impact the structural characteristics of the other samples. As a result, the following analysis will focus on sets composed of samples from the same class, as this ensures consistency in sample size and facilitates a more straightforward comparison.

Continuing to prune results that do not correlate to linear evaluation performance, we find that metrics like PCA, Mutual Information, and Statistics on the Activation Distribution also show no correlation. This might be because these SSL methods offer more than enough dimensionality in their latent spaces to learn comprehensive representations, allowing for a significant margin of error in the number of dimensions. This could also be explained if classes do not need the same number of dimensions, and that this is not related to the performance. Although previous works have studied total and dimensional collapse as potential issues, at the levels observed in the SSL methods we analyzed, these collapses do not appear to significantly affect performance.

Continuing in the same vein, cluster-related metrics also

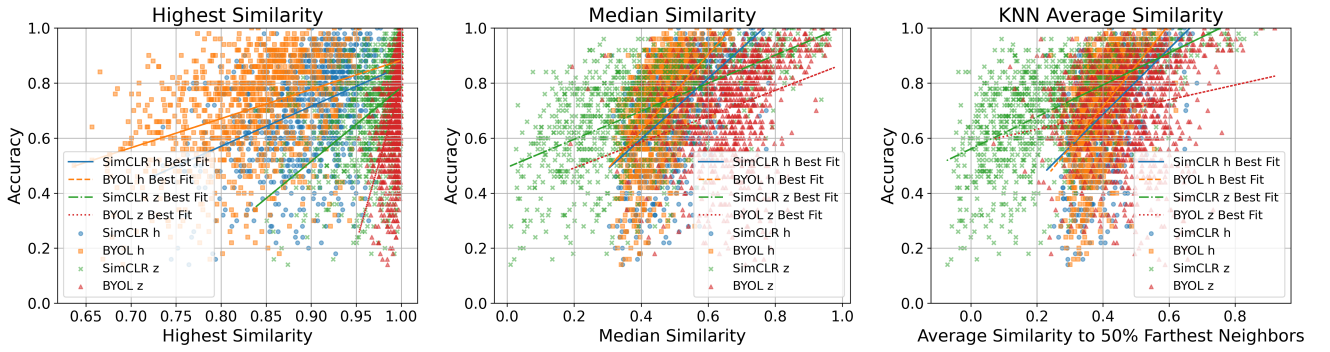


Figure 2: Legends with h indicate measurements on the backbone output, while z indicates the projection head output; the best-fit line represents a linear regression of the plotted data points. **Left:** The plot shows the relationship between the highest similarity among neighbors within the same class and classification accuracy. Higher accuracy is generally associated with higher similarity, indicating tighter class clusters. **Center:** This subplot illustrates the correlation between median similarity within a class and accuracy. Higher median similarity enhances classification performance. **Right:** The plot depicts how the average similarity to the 50% farthest neighbors within a class affects accuracy. High similarity suggests that the class samples are more tightly packed, which leads to better classification outcomes.

do not exhibit a relationship with linear evaluation performance. Given that these metrics, when applied to sets of representations belonging to the same class, may reflect the variety and expressivity within the class, it is understandable that they do not influence linear evaluation results. However, we suspect that they may have an impact on tasks requiring more fine-grained labels than those used in classification. As this falls outside the scope of the current study, we leave this exploration for future work.

Focusing on the descriptors that exhibit correlations with classification performance, we first observe that, for sets involving inter-class relationships, the descriptors suggest that classes benefit from being orthogonal to one another. This is illustrated in Figure 1, where the closer the highest and median similarity values are to zero, the better the classification performance. Notably, this relationship diminishes when nearest neighbors are masked, as shown in Figure 1, where only the 50% farthest points are considered, indicating that confusions may occur for classes close in the latent space.

While this result may seem expected, given the accuracy, since classes that are more independent from each other should be easier to separate with a linear classifier, it may not be not necessarily a desirable property for expressive representations. One might expect that classes within the same meta-class, such as dog breeds, would exhibit some degree of similarity to reflect their belonging to the same broader category. Moreover, following the same reasoning, when examining how samples from within the same class relate to each other, as shown in Figure 2. The more collapsed a class is, the higher the accuracy tends to be. Once again, this might initially seem expected, however, this reduction in the space occupied by a class could hinder its ability to encode diverse properties, which could be an area for future study. Together, these observations suggest that the representations learned through instance discrimination

favor high intra-class similarity and high inter-class dissimilarity, a structure well-suited for linear classification, the typical evaluation metric. Nevertheless, this might raise concerns for tasks like transfer learning, where more flexible and adaptive representations are needed. Further studies are required to explore the right balance between linear separability and the expressiveness of representations, a question not fully addressed in this work.

These two properties, though expected given the final performance of these approaches, are quite surprising considering that the loss function is solely performing an instance discrimination task. This structure is sub-optimal for what one would expect in an ideal instance discrimination latent space. Remarkably, this structural pattern also emerges in the output of the projection head, indicating that the projection head alone is not essential in explaining the emergence of this pattern. Furthermore, these properties mirror those optimized by instance discrimination, namely alignment and uniformity (Wang and Isola 2020), but they manifest at the class level rather than the instance level, despite the absence of class information in this SSL context.

These findings also highlight a key observation, which, to the best of our knowledge, has never been reported before: the correlation between accuracy and structure is measured at the class level. This means that our results do not simply suggest that clustering data points is beneficial, but rather that the more effectively a class can exhibit these properties, the better the classification performance. This holds true across different layers and models, as tested with both SimCLR and BYOL. This raises an important perspective: why does the model succeed in learning those for certain classes more effectively than others, despite using the same architecture and loss function? One possible explanation is that some classes may be inherently more challenging, either because they are less distinct from other classes or because they have intrinsic characteristics that make them harder to

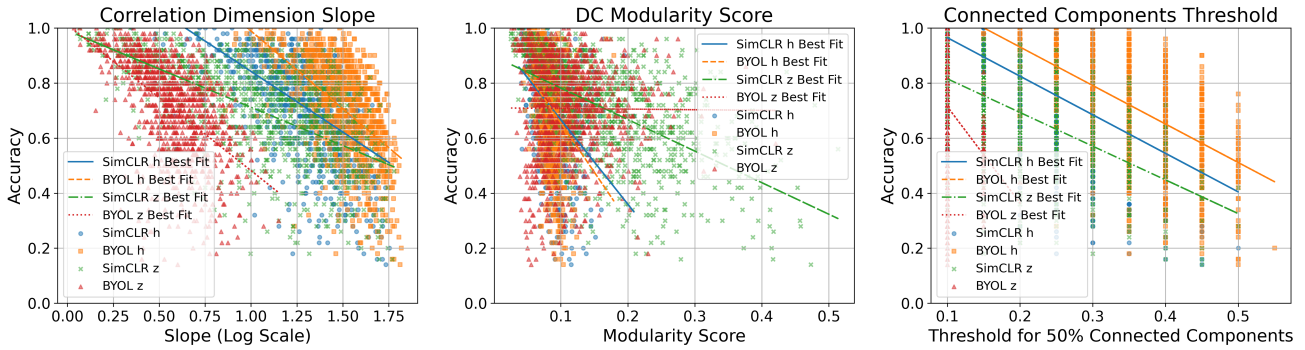


Figure 3: Legends with h denote measurements from the backbone output, while z denotes the projection head output; the best-fit line is derived from a linear regression of the points shown. **Left:** This plot demonstrates the relationship between the slope of the correlation dimension (on a logarithmic scale) and classification accuracy. Low values indicate denser and more compact representations, which correlate with higher accuracy. **Center:** Here, the modularity score from community detection is plotted against accuracy. Lower modularity scores, which reflect a more homogenous latent space with less fragmentation, are associated with better classification performance. **Right:** The plot shows the threshold at which 50% of connected components are formed in the latent space. Lower thresholds indicate that fewer, larger clusters exist, and this compactness correlates with higher classification accuracy.

represent effectively in the latent space with the chosen augmentations and loss function.

Descriptors relying on correlation dimension, community detection, and connected components further confirm the preceding results. The correlation dimension, which reflects the diversity and intrinsic dimensionality of the latent space, supports the idea that denser, less hierarchical structures are more favorable for classification (Figure 3). Interestingly, the representations produced by the backbone are richer and more complex than those of the projection head, which aligns with existing work and the hypothesis that the projection head ultimately simplifies the representation space for the invariance pretext task. Similarly, the modularity score from community detection, which measures the degree to which the latent space can be divided into distinct communities, also aligns with these findings. Lower scores, indicating less modular and more homogeneous latent spaces, are associated with higher accuracy. This suggests that for linear evaluation, a more uniform latent space, where data points are less fragmented into distinct clusters, is advantageous. The connected components analysis further supports this, showing that lower thresholds are needed to achieve a minimum of 50% connected components in higher-performing classes. This implies that a dense structure, where most data points remain connected even as the similarity threshold changes, is beneficial for classification accuracy. Fewer, thus bigger, isolated islands that persist as the threshold moves indicate a more robust, interconnected latent space that supports better classification outcomes.

Finally, while structural differences between the methods can sometime be apparent at the projection head’s output layer, understandably so, given their distinct learning objectives, the structures at the output of the backbone remain similar. Moreover, aside from differences in Median Similarity and KNN Average Distance in inter-class (Figure 1),

likely due to SimCLR’s use of explicit negative pairs, the projection head structures for both methods are still more similar to each other than to the backbone representations. This suggests that even fundamentally different instance discrimination methods, such as SimCLR and BYOL, tend to learn similar structure of representations to some extent.

Conclusion

In this study, we investigated the structural patterns emerging from instance discrimination-based self-supervised learning methods and their correlation with classification performance. Our findings reveal that despite the different objectives of methods like SimCLR and BYOL, the latent space structures they produce share significant similarities, particularly at the backbone level. These structures, characterized by high intra-class similarity and inter-class dissimilarity, are well-suited for linear evaluation but may pose challenges for tasks requiring more flexible representations.

Interestingly, our results suggest that the effectiveness of a class’s representation in a model is linked to its ability to exhibit these structural properties, prompting further exploration into why some classes achieve better representation than others under the same architecture and loss function. This could be due to intrinsic class characteristics or their relationship with other classes.

Ultimately, to apply these insights in designing new SSL methods, one might consider strategies like pseudo-labeling or defining subsets of data based on unsupervised groups to encourage the emergence of beneficial structural patterns. Such approaches could optimize the balance between linear separability and representational flexibility, potentially enhancing the performance of self-supervised models across a wider range of tasks.

References

- Bardes, A.; Ponce, J.; and LeCun, Y. 2021. Vi-creg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*.
- Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10): P10008.
- Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; and Joulin, A. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*.
- Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020a. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.
- Chen, T.; Kornblith, S.; Swersky, K.; Norouzi, M.; and Hinton, G. 2020b. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*.
- Chen, X.; and He, K. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15750–15758.
- Cole, E.; Yang, X.; Wilber, K.; Mac Aodha, O.; and Belongie, S. 2022. When does contrastive visual representation learning work? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14755–14764.
- Geirhos, R.; Jacobsen, J.-H.; Michaelis, C.; Zemel, R.; Brendel, W.; Bethge, M.; and Wichmann, F. A. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11): 665–673. Publisher: Nature Publishing Group.
- Grassberger, P.; and Procaccia, I. 1983a. Characterization of strange attractors. *Physical review letters*, 50(5): 346.
- Grassberger, P.; and Procaccia, I. 1983b. Measuring the strangeness of strange attractors. *Physica D: nonlinear phenomena*, 9(1-2): 189–208.
- Grigg, T. G.; Busbridge, D.; Ramapuram, J.; and Webb, R. 2021. Do Self-Supervised and Supervised Methods Learn Similar Visual Representations? *ArXiv:2110.00528* [cs, stat].
- Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P. H.; Buchatskaya, E.; Doersch, C.; Pires, B. A.; Guo, Z. D.; and Azar, M. G. 2020. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*.
- He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9729–9738.
- Misra, I.; and Maaten, L. v. d. 2020. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6707–6717.
- Purushwalkam, S.; and Gupta, A. 2020. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *Advances in Neural Information Processing Systems*, 33: 3407–3418.
- Saunshi, N.; Ash, J.; Goel, S.; Misra, D.; Zhang, C.; Arora, S.; Kakade, S.; and Krishnamurthy, A. 2022. Understanding contrastive learning requires incorporating inductive biases. In *International Conference on Machine Learning*, 19250–19286. PMLR.
- Tarjan, R. 1972. Depth-first search and linear graph algorithms. *SIAM journal on computing*, 1(2): 146–160.
- Wang, T.; and Isola, P. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, 9929–9939. PMLR.
- Wang, Y.; Zhang, Q.; Wang, Y.; Yang, J.; and Lin, Z. 2022. Chaos is a Ladder: A New Theoretical Understanding of Contrastive Learning via Augmentation Overlap. *ArXiv:2203.13457* [cs, stat].
- Xue, Y.; Gan, E.; Ni, J.; Joshi, S.; and Mirzasoleiman, B. 2024. Investigating the Benefits of Projection Head for Representation Learning. *ArXiv:2403.11391* [cs].
- Zbontar, J.; Jing, L.; Misra, I.; LeCun, Y.; and Deny, S. 2021. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*.
- Zhang, Y.; Lu, Y.; and Xuan, Q. 2024. How Does Contrastive Learning Organize Images? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 497–506.
- Zhao, N.; Wu, Z.; Lau, R. W. H.; and Lin, S. 2021. What makes instance discrimination good for transfer learning? *ArXiv:2006.06606* [cs].

An interdisciplinary view on behavioral properties in decision-making algorithms

Simon Forest^{1,2*}, Jose Villamar^{1,2,3}, Flora Gautheron^{1,4},
Léo Pio-Lopez⁵, Jean-Charles Quinton^{1†}, Mathieu Lefort^{2†}

^{1*}Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, UMR 5224,
F-38000, Grenoble, France.

^{2*}Univ. Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR 5205, F-69622,
Villeurbanne, France.

³Department, Organization, Street, City, 610101, State, Country.

⁴Department, Organization, Street, City, 610101, State, Country.

⁵Department, Organization, Street, City, 610101, State, Country.

*Corresponding author(s). E-mail(s): forest.phd@proton.me;

Contributing authors: j.villamar@fz-juelich.de;

flora.gautheron@univ-grenoble-alpes.fr; leo.lopez@tufts.edu;

quintonj@univ-grenoble-alpes.fr; mathieu.lefort@univ-lyon1.fr;

†These authors contributed equally to this work.

Abstract

Please provide an abstract of 150 to 250 words. The abstract should not contain any undefined abbreviations or unspecified references.

Keywords: keyword1, Keyword2, Keyword3, Keyword4

1 Introduction

Decision-making, the art of taking in information and drawing from it an action to undertake, encompasses many aspects of human behaviors. It ranges all the way from the smallest steps of perception, like picking a visual stimulus to gaze at, to more complicated procedures, like solving a puzzle. As such, it has been studied extensively in various fields of humanities, from psychophysics and neuroscience, to social science and economics. Meanwhile, decision-making has found additional interest in engineering science, and robotics in particular. All manners of complex objectives are given to robots, including solving a puzzle as well. This kind of tasks may include finding out where the correct piece is, how to reach, grasp, and manipulate it, and how to navigate in the room. And, for each of these steps, information is taken in from not only the agent and the object it interacts with, but also its environment, other robots (multi-agent systems), humans present in the scene, and more. So, decision-making in artificial systems ranges from a perceptual level too. Eventually, it can be argued that out of very different research fields, similar behaviors are being studied, albeit using different setups and different models. And it actually turns out that a lot of parallels can be drawn, either by design or not, between biologically-motivated and engineering-driven models (Pezzulo et al, 2014; Escobar et al, 2022).

First, psychology and neuroscience have put a lot of focus on how and what decisions are made (Gold and Shadlen, 2001, 2007; Lepora and Gurney, 2012). Perceptual experiments can offer insights into some inner mechanisms of decision-making. Take the Stroop effect for example, which designates the cognitive interference that takes place when incongruent information is presented (Stroop, 1935; Scarpina and Tagini, 2017), for instance the ink color of a printed word (e.g., yellow ink) which may differ from the meaning of the word itself (e.g., the word “blue”). One possible way of testing this effect (figure 1) is to present to human subjects, on a computer screen, color names printed using different font colors, and ask them to indicate the color of the

stimulus by clicking a button (Incera et al, 2013). This test can be seen as a purely binary decision-making task, where the choice between two options is imposed. But the process can also be studied as continuous, relying on the mouse-tracking technique to collect human behavioral data. In this experiment, the cursor can be seen to be temporarily attracted to the wrong answer (in our figure, the word content “blue”). This puts the task, unchanged, in a new empirical paradigm, where the study relies on the measurable action to reflect on the continuous internal process. As there are different ways of studying the problem in space and time, decision-making models may need to adapt to these changes. With more or less fidelity to the neural mechanisms of the human brain, psychophysical models can focus on explaining either the statistical outcomes of the decision (e.g., economy models on prospect theory, Kahneman and Tversky, 2013, or multinomial models, Conrey et al, 2005), the delay between the stimulus presentation and the choice (Ratcliff and McKoon, 2008), or even the entire mouse trajectory (Falandays et al, 2021).

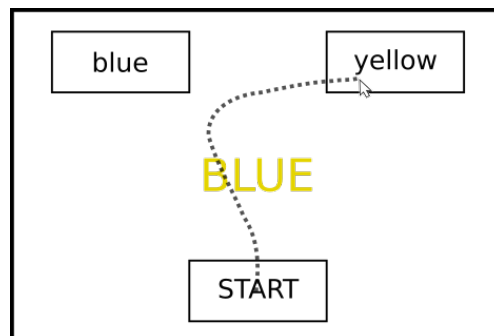


Figure 1 Schematic depiction of a setup used to demonstrate a type of color–semantics Stroop effect (Incera et al, 2013). A colored word appears in the center of the screen when a human subject clicks “start”, after which they are asked to click on the button indicating the stimulus color. Cursor trajectory is shown in dashes.

The same variety can be found in behavioral models embedded in artificial systems (e.g., for action selection or control). In figure 2, a robot is tasked with navigating towards a target in a cluttered environment. Here, the decision is about which direction to follow in a continuous 2D space. One can also restrict the decision space to 1D

by taking mainly the heading direction into account (Bicho and Schöner, 1997), and assuming, e.g., that the robot moves forward at a constant speed. In any case, the robot trajectory can be seen similarly to the mouse trajectory in the previous example, swapping the “yellow” button for another attractor and the “blue” distractor for an obstacle repeller. The robot decision can also be made binary if one allows only two actions: “turn left” and “turn right”.

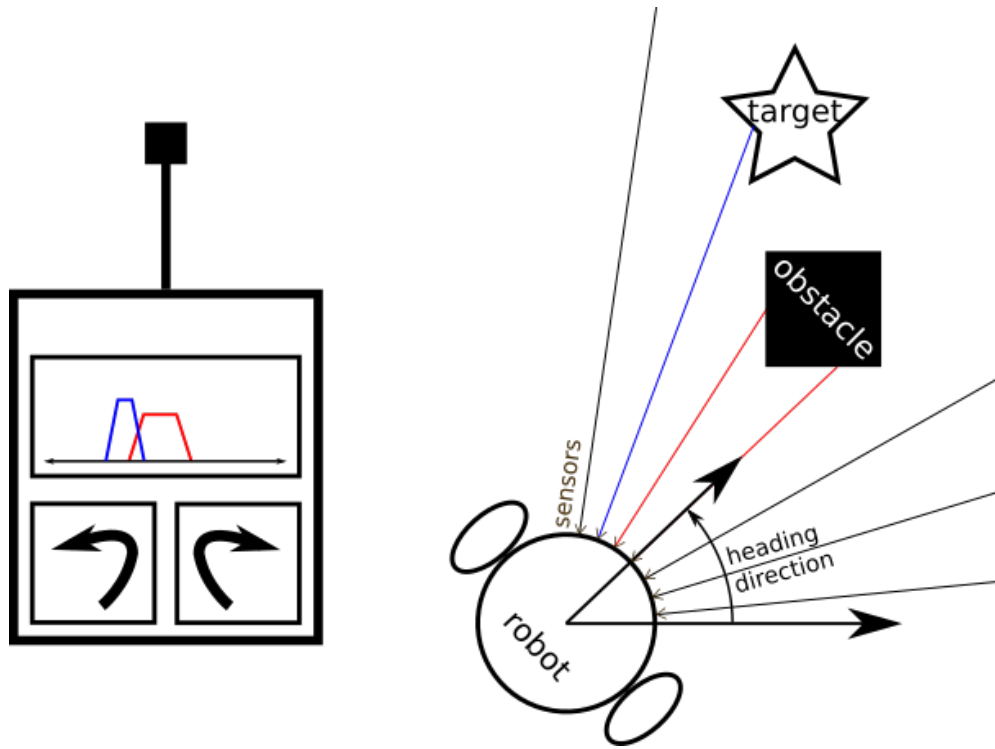


Figure 2 Depiction of a robot required to reach a target and avoid an obstacle (inspired by Bicho et al, 2000). The orientation command can be specified as either continuous (a given heading direction in degrees) or binary (“turn left” or “turn right”).

Obviously, the sensors and actuators involved in the decisions and actions are very different between a robot and a human body, but we believe that the decision by itself could be put in a frame of reference common to both biological and artificial systems. There is however one big difference in how decision-making is approached

across the research communities. In robotics for instance, decision-making is most often seen through the perspectives of artificial intelligence: computer vision (categorization, tracking), machine learning (classification, behavior prediction, reinforcement learning), or swarm intelligence. In these cases, the focus can be put solely on which decisions are made, the “how” can be cast aside, and learning is often used to circumvent the need for an explainable process in the construction of a decision. Today, tasks like our robot example find a mainline solution in deep reinforcement learning ([Mnih et al, 2015](#); [Arulkumaran et al, 2017](#)). In short, a model is trained by running simulations and setting rewards depending on the outcome, and behaviors are obtained by tuning a neural network made of thousands of neurons so that the reward expectations are maximized. For this robot, this would mean learning whether to turn left or right, and when to move forward, for any set of inputs received by the sensors, as long as the generalization of the neural network allows it. More and more systematically, learning is being put forward as the go-to bridge from a problem to its solution, but little to no importance is given to the capabilities of the system itself, i.e., that the design of the action model by itself allows to produce the adequate response, and that relevant constraints have been integrated in the system ([Brooks, 1991](#)). On the contrary, devices such as Braitenberg vehicles ([Braitenberg, 1986](#)), where actuators are directly activated by sensors for their simplest versions, show that even the simplest systems can exhibit many interesting properties, as long as we focus on how the system works.

The properties in question take many shapes, and, of course, do not always gather the same interest from one field to another. Research in artificial intelligence and robotics has stirred up many challenges. Data integration may include interpolating knowledge from incomplete or ambiguous inputs, generalizing from random samples, and merging from different sources signals that are not always congruent. Knowledge

and decisions may have to be processed in a dynamic setting, thus adding a requirement of stability: the ability to focus on targets while able to react to sudden and important stimuli, and showing robustness to noise and to unwanted distractors. Decisions may then take the form of a sequence of events, as hypothesized by reinforcement learning for instance (Kaelbling et al, 1996). From this point, some new dynamic and spatiotemporal properties can be expected, as systems that are meant to interact with their environment may generate sensorimotor behaviors in which decisions influence actions and reciprocally (Lepora and Pezzulo, 2015).

The common point in these properties is that they are all at least partially found in human behaviors. We are looking at somewhat cognitive properties. Thus, an increasing share of researchers in engineering science have turned their attention to biological inspirations, as showcased by the emergence of so-called cognitive architectures, that are frequently laid as an inevitable groundwork to artificial general intelligence (Goertzel, 2014). However, these advanced architectures most often rely on learning as a support of cognition, disregarding some instantly-available properties of decision models that have been developed in either human or engineering science. And yet, every field of research has its own ways of modeling task resolutions in a naive context, using only the stimuli and/or the system reactions as input. Sometimes, these simplest components are taken for granted or not questioned. But these different models, stemming from different practices from different disciplines, may actually fit together in a common symbolization.

1.1 Main paradigms

This section is not meant to deliver an exhaustive review of all decision-making architectures, but rather to present a quick overview and representative sample of classical approaches used in different disciplines, pointing either to field-specific review articles, or to typical examples of applications.

Regarding cognition, psychophysics and neuroscience have proposed many models of decision-making, most notably using accumulators (Gold and Shadlen, 2007). In these models, which were originally designed for two-alternative force-choice tasks, evidence is gradually accumulated over time until a given threshold is reached (Vickers, 1970). The most common of these is the drift-diffusion model (DDM), in which two opposite thresholds are set, representing two possible decisions (Ratcliff and McKoon, 2008). The DDM can be linked to many alternative models in which two (or more) units, each representing a possible choice, are put in competition until one prevails (Bogacz et al, 2006). The most notable of these would be the leaky competing accumulator (LCA), which also accounts for information decay over time (Usher and McClelland, 2001; Bogacz et al, 2007).

Applications of accumulator models are not exclusive to humanities. They have been used for decision-making in robotics, most notably in the form of dynamic neural fields (DNF), population-coded accumulator models running on a topological map (Schöner et al, 2015). The decision is then read from a weighted sum or argmax of the model output. Originally meant to simulate the interaction of cortical columns in neural maps (Amari, 1977), DNF have found applications as much in neuroscience (Wijeakumar et al, 2017; Buss and Spencer, 2018) as in robotics (Sandamirskaya, 2014; Tekülve et al, 2019; Grieben et al, 2020).

Probabilistic models constitute another category that crosses disciplines. For instance, many models based on maximum-likelihood estimation (MLE) and Bayesian inference have been used for data fusion (Castanedo, 2013). MLE in particular reflects computations observed in psychophysics (Ernst and Banks, 2002). Among other implementations, one has to cite active inference, that has been used in either neuroscience (Friston et al, 2013) or robotics (Pio-Lopez et al, 2016); and Kalman filters (KF) (Kalman, 1960), which are widely used in robotics (Chen, 2011). Behind this

paradigm, one also finds learning algorithms like variational autoencoders (Kingma and Welling, 2019) using Kullback–Leibler divergence minimization (Doki et al, 2015).

Algorithms containing a training phase are not in the focus of this review, but as far as decision-making paradigms go, one can hardly miss out on classification and regression algorithms. Decision trees, support vector machines (Somvanshi et al, 2016), deep neural networks (LeCun et al, 2015) and clustering algorithms such as self-organizing maps (Kohonen, 2012) can be used to learn relationships between data and a potential decision (often made legible under the form of a “best-matching unit”). For regression, we can cite uses of Gaussian processes (Rasmussen, 2004), Gaussian mixture models (Plataniotis and Hatzinakos, 2000) and locally-weighted projection regression (Vijayakumar and Schaal, 2000) in robotics (Khansari-Zadeh and Billard, 2011) for example. See Sigaud et al (2011) for a survey and unifying framework on these methods.

All models mentioned so far claim some amount of biological inspiration or plausibility. But engineering science also has some fully self-made methods, for instance fuzzy logic (FL). It describes operations made on fuzzy sets, where truth values are no longer binary but instead compared to membership functions expressing possibility values, between 0 and 1 (Zadeh, 1965; Bellman and Zadeh, 1970; Dubois et al, 2004; Dubois and Perny, 2016). By fuzzifying sensory inputs and combining their membership functions, one can create fuzzy commands, that can be exploited in computer vision (Krishnapuram and Keller, 1992; Sobrevilla and Montseny, 2003), data fusion (Russo and Ramponi, 1994), or robotics (Wakileh and Gill, 1988; Bajrami et al, 2015; Qureshi et al, 2018).

While these various models were designed with different approaches and objectives in mind, either to explain observed behaviors, or to achieve a given task, they are by no means incompatible with one another. In fact, it can be argued that some

different algorithms may model the same behavior, but at a different level of abstraction. To give an example, [Bitzer et al \(2014\)](#) and [Gepperth and Lefort \(2016\)](#) argue that DDM and DNF respectively provide a plausible implementation of Bayesian inference. Alternatively, some of the basic models we have cited may be combined in order to achieve more complex behaviors. There have been many instances of hybrid architectures in artificial intelligence ([Sun et al, 1999](#)), mimicking different parts of a brain working together to produce new, smoother, and richer behaviors. [Goertzel et al \(2011\)](#) call this a cognitive synergy. Their reasoning is that combining model parts allows to overcome their individual limitations and exploit fully the capabilities of each, with different stages of learning, states of memory, and steps of decision-making. [Goertzel \(2014\)](#) cites hybrid models as a thorough, yet “inelegant” way towards modeling human-like intelligence.

1.2 Positioning

There have been little attempts at unifying these decision-making algorithms in a field-agnostic formal setting. Our argument is that decision-making can be described with generic terms and criteria, no matter what field of research it is studied in. To start with a broader view, let us decompose our definition of decision-making, which is described in two parts: take in information, and pick out an action.

Information taken in

One expects an unambiguous response to a set of stimuli that can be diverse, conflicting, and sometimes extremely dense. In the robotic navigation example, image processing may acquire a mountain of evidence: position and nature of objects, their relative proximity, and perhaps all sorts of visual indications such as warning signs or movement detection. All of this has to be integrated into a single decision, namely, what direction the robot should follow.

A first challenge is in the merging of modalities. A robot may need to combine the sight and sound of an object to better locate it. In the Stroop test, an observer receives both semantic and visual information about two colors. Multimodal merging depends on the task in progress (e.g., to select the font color and ignore the text content), but also on stimulus saliency and reliability. On the psychophysics side, the effect of reliability on merging has been studied extensively (Ernst and Banks, 2002; Alais and Burr, 2004; Calvert et al, 2004). This is not an effect we delve into in this review, but the question of adapting psychophysical models of fusion into artificial systems has been raised elsewhere (Forest et al, 2022b). Another issue is that the different modalities may be experienced in different spaces, e.g., a robot with a camera and two microphones will have a good visual resolution anywhere in line of sight, and a poor auditory resolution outside its azimuthal axis, so, before any decision can be made, either some signals have to be projected into the sensory space of another, or they all have to be projected into a common space (Forest et al, 2022a). We do not include this step in our methodology. Instead, we treat decision-making at a level in which a minimal amount of projections has already been made, so that potential multimodal stimuli can be put in a common ground.

Indeed, regardless of the system under study, available information never consists in raw stimuli. At the very least, sensors transform the signals. In vision, cameras are limited by their resolution and encode visual stimuli over a few channels (e.g., colors), possibly with some compression underway; human eyes are limited by the sensor distribution on the retina, and turn stimuli into electric discharges sent to the brain. Then the information is processed through different operations (e.g., convolutional neural networks for computer vision, or neural pathways in the human brain) until it becomes actionable. For this reason, we consider stimuli as sets of input *activities*. In this review, *activity* is the generic term we use for the material manipulated by a processing *unit* during a decision-making process. We regroup under this term the

values given by output neurons in an artificial neural network, the membrane potential of neurons in a cortical map, and the evidence accumulated in diffusion decision models. All models have some activity going in (the output of the preprocessing system, if any, or the stimuli themselves) and some activity going out (some form of decision).

This activity can take many forms. In the Stroop test example, the decision space appears to be categorical. There are two choices, “blue” and “yellow”. Sometimes, activities may be sparse (figure 3a), meaning any and all decisions must be made to one of the stimuli, and no compromise can be made. This makes sense in some situations, for instance, in the implementation of the Stroop test of figure 1, clicking in-between the two buttons, or answering a color other than blue or yellow would not be acceptable. Having a sparse decision space does not exclude the possibility that some part of the decision process involves an underlying continuous topological space, as illustrated by the mouse trajectory in figure 1. On the contrary, this is quite beneficial for an all-inclusive implementation, since operations such as interpolations and barycenters would not be viable in a sparse setting. For this review, we make the assumption that there exists an underlying topology (in our example, either the cursor position, or the color spectrum) in which the sensations of the stimuli could be placed (figure 3b). While not systematically necessary, this is not a risky assumption. In human brains, most low level stimuli can be encoded into peaks of membrane potential on cortical maps. At least in this topology, some interpolation is possible. In artificial systems, the encoding is more or less free. Then, once the topology is present, we can reconsider the sparsity of the stimuli. Between the spread of the signal in its environment, the blur added by sensors with limited resolution, and the possibly overlapping receptive fields of the units that receive this activity, the sensations can be viewed as continuous blobs (figure 3c). While this is realist in the eye of a perceptory system, this is hardly processable in a computational model. Yet some models do

require working on spatially-spread activities. In that case, one can add some spatial discretization as a step of input preprocessing (figure 3d). This kind of algorithmic input can also be artificially projected from the sparse topology-grounded stimuli using some kind of kernel function, e.g., a Gaussian.

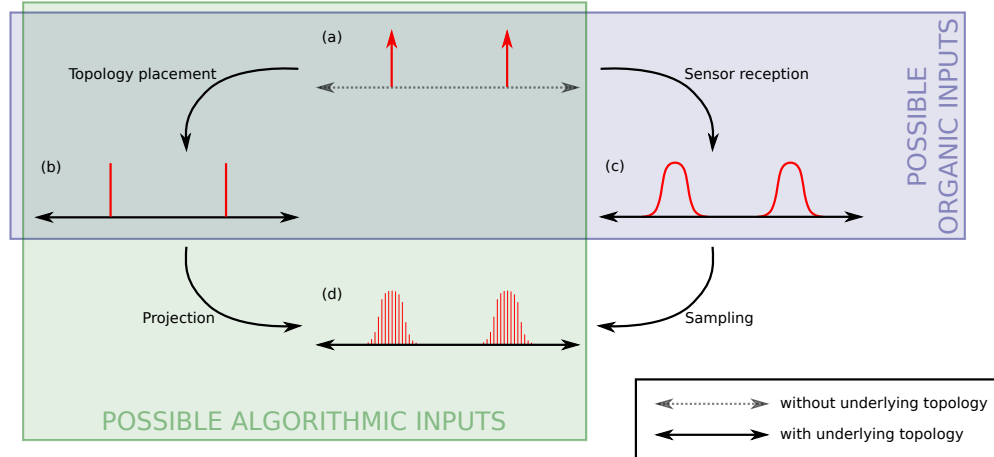


Figure 3 Possible ways of integrating stimuli. Depending on the level at which activities are viewed in the perception–decision process, in an organic system, they can be seen as: (a) sparse without an underlying topology (no distance measure available), (b) sparse in an underlying topology (meaning there exists a distance metric enabling interpolations), or (c) continuous. Artificial systems and computational models may require activities to be discretized (or “sampled”, in the signal-processing meaning of the term) from continuous stimuli (d). The last setup can also be simulated from (b) using a kernel projection akin to receptive fields.

Action picked out

The output of a decision-making model is meant to be used by another system: either motors or other decision modules. It will sometimes be of use to the same model in cases of recursion. Indeed, some stimuli can be time-dependent, and some algorithms make use of previous internal states to generate the next output (Kalman, 1960; Amari, 1977). Internal states can take multiple forms, from priors to membrane potentials, and are not always easy to interpretate, the most obvious example being the activation in hidden layers of recurrent neural networks (Ming et al, 2017).

In the end, models act as filters, as they take activities as input and give activities as output, filtering out noise and unwanted components. The type and dimensionality of output activities may be the same as input, or not. In-between, the way these values are processed is specific to each algorithm. We define a *unit* as the basic element that will either process one activity value, or aggregate multiple activities into one. For example, a model working on two sparse stimuli will be made of two units that receive two input activity values. Sometimes, each unit processes its own activity independently from the others. Sometimes, some interaction occurs between units.

In any case, at some point, it is necessary to compute a single output value, representing an intelligible answer, e.g., a motor command. Not all models include systematic solutions to aggregate unit outputs into a single value. In that case, we provide additional computation steps outside the model to create such a result, i.e., aggregators.

1.3 Objectives

Our goal is to bring together some elementary solutions from different domains (most notably, neuroscience, psychophysics, and robotics) in order to compare them in a field-agnostic point of view. This also means that we do not claim to regroup all variants and composites of models used in any given field. We make the choice to highlight the readily-available properties of a representative sample of existing models, and we propose a framework that allows to unify them. As we mentioned earlier in this introduction, we restrict ourselves to models with intrinsic behavioral properties, not learned properties. We also refrain from presenting all derivatives of popular architectures in this framework, and mostly stick to the most essential implementations. Note that this framework has been co-designed with a software architecture which is openly accessible and easily allows the addition of more extended models.

In this article, we want to pick a representative sample of existing learning-free decision-making algorithms. We find that they are mostly divided in three families: logic-based models, probabilistic/Bayesian models, and dynamic accumulators. Our selection is made of both simple and advanced models from each category, going from bare aggregators such as winner-takes-all (WTA) and weighted sum (WS), to more complex methods such as FL, DNF and KF. We formalize them using common notations in order to emphasize their different characteristics: topology-based interaction between processing units, output aggregation, recursion

We set up toy examples to display the qualitative properties of each algorithm. Our focus is mostly on the decision, although we can also measure numerical values of model activity, a quantification of model internal state. Our purpose here is not to tune or train models to fit complex behaviors, we stick to the emergent properties of standard and isolated models.

In the next section, we describe the models selected for the formal unification and comparison, as well as the scenarios they are tested on and the way their outputs are read. Section 3 gives the results and comparisons of all the models on all the scenarios. We conclude and add perspectives in section 4.

2 Methods

In this section, we start by describing the experimental setup in subsection 2.1. Then we explain how models are evaluated in subsection 2.2. Finally, we present the core of all models in subsection 2.3. In the end, each simulation is made of a combination of up to three parts: scenario generation, model processing, and aggregation (sometimes included in the model). See figure 4 for a visual representation of this formalization process.

2.1 Scenarios

Time scale

Our simulations take place in discrete time. Scenarios are defined over a finite series of timesteps, the step Δt being constant throughout the simulations. While the value of Δt can have visible effects on the behaviors of models with temporal integration, we use a sufficiently low step timestep in order not to hinder the performance of any model. This choice is consistent with real-life applications of decision-making algorithm, with the perception of artificial agents being limited to a certain amount of frames per second, as well as psychological modeling, with neurons having a finite fire rate.

Working space

We make the hypothesis that all scenarios can be expressed in a topological space X . For computational purposes, we assume that X can be discretized into a regular set $\{x_1, x_2, \dots, x_n\}$. A scenario is composed of one or more stimuli, the sensation of which can be decomposed into a set of input activities. An input activity k is characterized by an amplitude a_k and position x_{i_k} .

As illustrated in figure 3, computational models may differ in the way they take these inputs. Some are composed of sparse units operating for each existing activity : $\{x_{i_1}, x_{i_2}, \dots\}$. They will take input type (b). While it is always present in this particular implementation, the underlying topology X might not be relevant to some models. In that case, input type (b) can be seen as a placeholder for input type (a). The difference will be shown in the model formalisation, but neither in the implementation nor the graphical results. Some other models are made of n units filling the decision space: $\{x_1, x_2, \dots, x_n\}$. (This is compatible with the previous notation, with $i_1 = 1, i_2 = 2, \dots, i_n = n$.) They can take either input type (b) or (d), although the latter is more appropriate for some models expecting a continuum in the activity.

Noise treatment

The following models have very different relations to noise. Perception and control result of multiple, interwoven processes, and models integrate this bundle of mechanisms, and the inevitable stochasticity that it comes with, with different levels of abstraction (see recap in table 2). Some will consider that noise is part of the decision process, and treat it like a supplementary parameter. Some assert that noise is statistically estimable from the inputs, and that estimation is part of the results. Some do not process noise unless it is added manually to the inputs. To put all models on an equal measure, the scenarios we use are all deterministic. The different approaches to integrating noise have been discussed elsewhere, and especially in (Forest et al, 2022b), where it plays a crucial role.

Simulation plan

We set up eight non-stochastic scenarios that determine the inputs to give to all models. They are presented in table 1. Each stimulus plotted is shown as a thick bar. For non-spatialized models, only the amplitude of the bar is taken into account. For some other models, the bar will be replaced by a Gaussian.

The scenarios were picked to show the various spatiotemporal properties of the models, so they include cases where, depending on tasks, interpolation between signals is likely, and cases where it is not, as well as dynamic settings to evaluate attentional properties and reactivity.

2.2 Aggregators

Depending on the model, two kinds of outputs can be read, sometimes both:

1. A positional decision \bar{x} , possibly accompanied by an activation value \bar{y} . \bar{y} can sometime be related to an estimation of the certainty of the decision.

2. A set of activity values y_k for all stimulated x_{i_k} . The i_k designate the indices of all units in the model.

In order to make a decision, we want to extract a singular value $\bar{x} \in X$ after the model processing, in all cases. If some interpolation is made, \bar{x} can fall outside the set $\{x_1, \dots, x_n\}$. When a model does not include a way of reading the decision directly, we need to add an aggregator to compute the decision localization from the model activity. It takes the following form:

$$\bar{x}(t) = \sum_k w_k(t) x_{i_k} \quad (1)$$

This is a weighted sum of all evaluated positions. The weights w_k depend on the activities y_k , and can be configured in mainly two ways:

- Plain barycenter:

$$w_k(t) = \frac{y_k(t)}{\sum_j y_j(t)} \quad (2)$$

i.e., all units contribute proportionally to their activity.

- Mean of maxima (or argmax): Let $S(t) = \operatorname{argmax}_k(y_k)$.

$$w_k(t) = \begin{cases} 1/|S(t)| & \text{if } k \in S(t) \\ 0 & \text{if not} \end{cases} \quad (3)$$

where $|S(t)|$ denotes the size of set $S(t)$. In short, this is a barycenter of all units of maximum activity. Very often, there is a unique maximally-activated unit, in which case this aggregator is essentially an argmax.

2.3 Models

2.3.1 Representation convention

As one of the objectives is to propose a unified frame of analysis of the models, they will be depicted using a common formalism, captioned in figure 4. The entire evaluation process is split into two or three parts, the model being separated from the scenario generation, and its aggregator if one is necessary. Some varying properties of the models can be seen in the following depictions:

- The topology on which the decision takes place is shown as a black line (e.g., figure 6). For models that do not require knowledge of the topology, the line is dotted (e.g., figure 5).
 - Some models are iterative. We show the intermediate steps from a state at time t to a state at time $t + \Delta t$, but the time loop is not explicitly represented (the state at $t + \Delta t$ replaces the one at t , then links to the one at $t + 2\Delta t$, etc.). Instead, when a previous state is used recursively, it is highlighted in gray in our representation.
 - Some models contain some amount of interaction (i.e. the potential at position x_i depends on the potential at position $x_{j \neq i}$). In our depiction, this always results in a vertical step (models with two rows , e.g. figures 6, 10 and following).
 - TODO: add \bar{y}
- + make one item per property from table 2 (cf. comm. JC)

2.3.2 Logic-based models

Noise integration. Models in this family take any information as a truth value. Noise in inputs would be taken as is and not filtered in any way. Most notably, these models would be rendered totally useless in noisy competition tasks: for example, add a bit of white noise to a scenario made of two very similar stimuli (scenarios A and B in

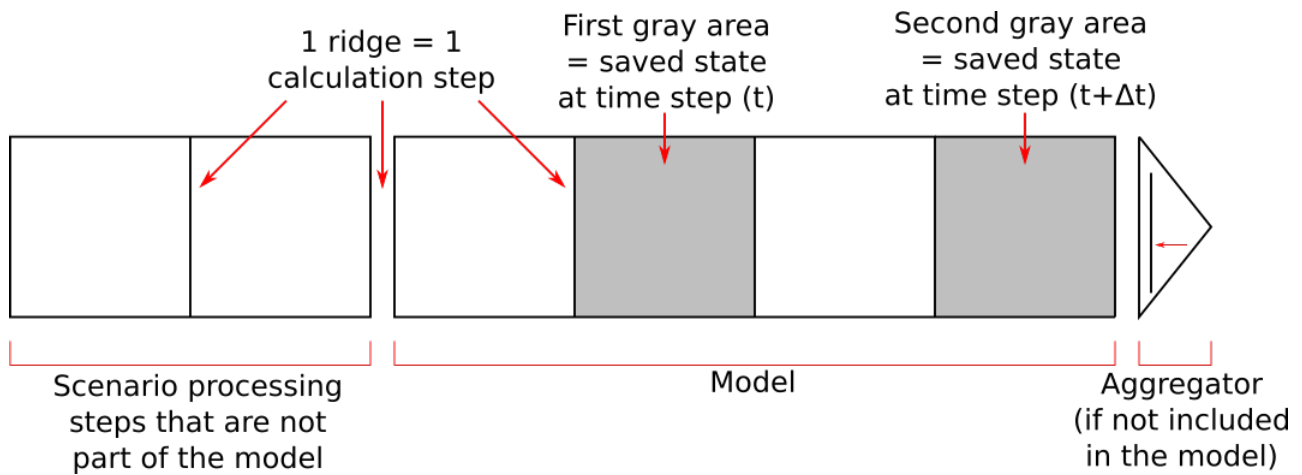


Figure 4 Legend for the schematics of the models. Models with recurrent states are shown unfolded, i.e. the process to go from time step t to time step $t + \Delta t$ is visible. The recurrence can be pictured by furling the pattern so that the gray areas touch each other. The aggregator part is shown attached to the model when the latter produces a readable direction directly, and detached if it has been added retrospectively. The arrow in the aggregator indicates where the final decision \bar{x} is read.

table 1), and the output will start switching back-and-forth randomly between the two stimuli.

Winner-takes-all (WTA)

This is the simplest model of all. The decision $\bar{x}(t)$ is made at the position of the stimulus of highest intensity. It amounts to applying the mean of maxima aggregator directly on the input (figure 5). See equation (3).

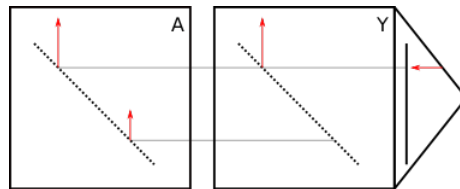


Figure 5 Main steps of a WTA model. Explanations in text and figure 4.

For WTA to fit in model formalization, we arbitrarily define its activity $\bar{y}(t)$ at position $\bar{x}(t)$ as:

$$\bar{y}(t) = \max_k(a_k(t)) \quad (4)$$

Fuzzy logic (FL)

As shown in figure 6, this model functions in two steps. First (top middle square), the inputs are fuzzified using a truncated triangular distribution (Dubois et al, 2004), so that they can express a possibility value between 0 and 1, everywhere in the topological space. Second (bottom middle square), they are accumulated using a minimum:

$$\begin{cases} y_k(t) = \min_j \left(\max \left(1 - a_j(t), P(x_{i_k}, x_{i_j}(t)) \right) \right) \\ P(x, x') = 1 - \alpha|x - x'| \end{cases} \quad (5)$$

where α specifies the slope around the stimuli. α determines how likely stimuli are to interact. With a high α , they are unlikely to mix and the model acts closer to a WTA. With a low α , a midpoint is easily reached and the model acts closer to a WS.

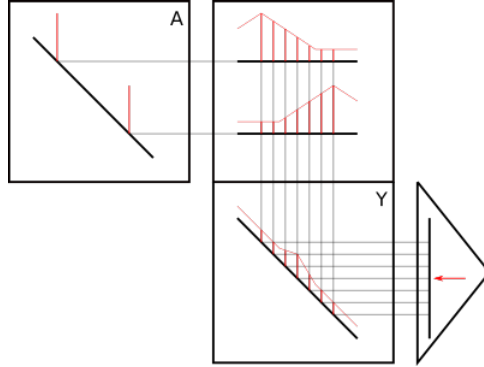


Figure 6 Main steps of a FL model

The decision \bar{x} is then found by using a mean of maxima.

2.3.3 Distribution-based models

This class of models operates on interpolations of inputs. Consequently, it is necessary for the inputs to be placed in a topology, as there is no telling that a barycenter of categories \bar{x}_i makes sense.

Noise integration. Noise is one side of the estimation. It is not that each presentation contains a certain amount of noise, but instead that each presentation and/or sensation is assumed to vary following a probability distribution that is asserted by the model. In our implementation, inputs are assumed to aggregate into a Gaussian.

Weighted Sum (WS)

This model consists of a plain barycenter of the inputs (figure 7). See equation (2).

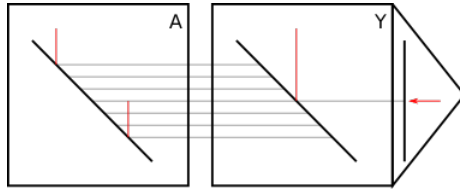


Figure 7 Main steps of a WS model

For WS to fit in model formalization, we arbitrarily define its output activity $\bar{y}(t)$ at position $\bar{x}(t)$ as:

$$\bar{y}(t) = \sum_k a_k(t) \quad (6)$$

Maximum-likelihood estimation (MLE)

This is the main paradigm used in multisensory integration (Ernst and Banks, 2002; Rohde et al, 2016). Given stimuli drawn in Gaussian distributions of estimated position x_{i_k} and variance σ_k^2 , MLE models the decision as a Gaussian distribution of

mean m and variance s^2 given by:

$$\begin{cases} m = \frac{\sum_k \frac{1/\sigma_k^2}{\sum_j 1/\sigma_j^2} x_{i_k}}{\sum_j 1/\sigma_j^2} \\ s^2 = \frac{1}{\sum_j 1/\sigma_j^2} \end{cases} \quad (7)$$

Our implementation is not directly compatible with this paradigm. Our models are meant to receive individual trials, while MLE operates on a distribution of trials. In particular, we do not have statistical variances σ_i in the sensations. Oppositely, MLE does not take into account stimulus intensities represented by a_i . So, for readers interested in what MLE would give in our scenario, we can simulate it using a variable transform $a_k = 1/\sigma_k^2$, making the amplitude a measure of stimulus reliability. Equation (7) then becomes:

$$\begin{cases} m = \frac{\sum_k a_k x_{i_k}}{\sum_k a_k} \\ 1/s^2 = \sum_k a_k \end{cases} \quad (8)$$

which is exactly the same as our implementation of WS, with $\bar{x}(t) = m$ and $\bar{y}(t) = 1/s^2$ (figure 8).

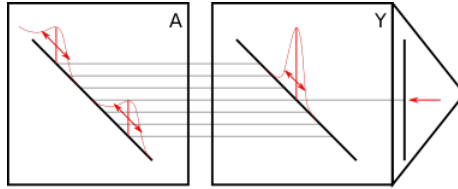


Figure 8 Main steps of a MLE model

Kalman filter (KF)

This model acts as a time-related MLE: instead of interpolating between two inputs at the same time (using their respective variance to determine their weight), it interpolates between a new aggregated input at time $t + \Delta t$ and its older interpolation at time t (figure 9). This time, we use the spatial variance, estimated from the entire input array at each time step. Consequently, an unambiguous presentation has less variance (so more weight) than a presentation with two or more stimuli. Also, it is necessary here to assume continuous sensations, as a sparse input made of a single Dirac would have zero variance, rendering the model quickly irrelevant.

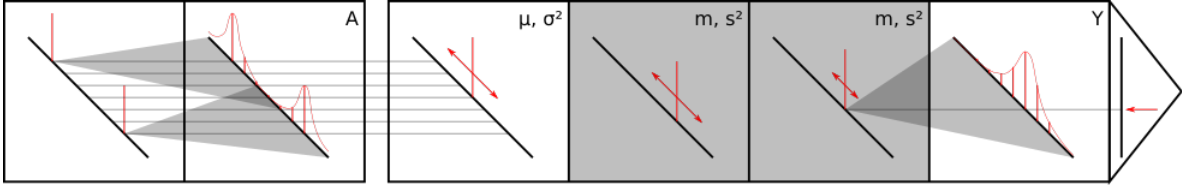


Figure 9 Main steps of a KF

The activity takes the form of a Gaussian of mean m and variance s^2 . We compute the mean μ and variance σ^2 of the input in order to update the activity:

$$\begin{cases} m(t + \Delta t) = K(t + \Delta t) \mu(t + \Delta t) + (1 - K(t + \Delta t)) m(t) \\ s^2(t + \Delta t) = (1 - K(t + \Delta t)) p(t) \end{cases} \quad (9)$$

with K the Kalman gain, defined as:

$$K(t + \Delta t) = \frac{p(t)}{p(t) + \sigma^2(t + \Delta t)} \quad (10)$$

and p the extrapolated estimate uncertainty:

$$p(t) = s^2(t) + q \quad (11)$$

where q is a parameter representing the process noise. A high q means that internal variance remains high and the model always gives a strong weight to new inputs. A very low q implies that new inputs are ignored.

Initial values $m_0 = m(t = 0)$ and $s_0^2 = s^2(t = 0)$ may have an influence on the behavior of this model. In particular, a low s_0^2 will give a strong influence of the prior position m_0 over the incoming inputs. To make this prior knowledge negligible, we set a high initial variance $s_0^2 = 1$ (and $m_0 = 0$ to stay as neutral as possible).

In any case, the model output activity can be represented as a Gaussian of mean $m(t)$ and standard deviation $s(t)$:

$$y_k(t) = \exp\left(-\frac{(x_{i_k}(t) - m(t))^2}{2s^2(t)}\right) \quad (12)$$

However, the KF does not need an aggregator, as its decision can directly be read as the predicted mean:

$$\bar{x}(t) = m(t) \quad (13)$$

2.3.4 Accumulators based on sparse inputs

Inspired from neuroscience, accumulators are a whole family of models consisting of units that accumulate evidence over time (Bogacz et al, 2006; Roxin, 2019). This section describes the main accumulator models that can be used for two (or more) alternative choice tasks, which do not necessarily take place within a given topology. Each processing unit represents a possible decision, its potential (internal activity)

starts at zero and increases gradually as evidence in favor of the decision is brought. The relations between the different models is synthesized in figure 15.

Noise integration. These models see noise as a part of the decision process. Either added to the sensory inputs as an outcome of background stimulations and sensor imperfections, or embedded as an inevitable side-effect of microscopic neural mechanisms, noise favors bifurcation when a dynamic system is stuck in an unstable equilibrium. For instance, given two distant competitors of similar intensity, a small amount of noise is sufficient to ensure that one is selected over the other. Temporal integration is complementary to the stochasticity, as it permits keeping a random decision stable, contrarily to WTA and FL. For this reason, it is very common to add a supplementary parameter to the implementation of accumulators, which determines the amount of (often white) noise added to all units. This is very different to models such as KF, for which adding white noise to the inputs would cause very little change to the results. Noise integration is at the heart of (Forest et al, 2022b), in which this distinction is discussed further.

Topology. As depicted in figure 15, DNF is a special kind of accumulator model that relies on and exploits a topology, making it comparable to a spatially-continuous diffusion model (Ratcliff, 2018). This makes a big enough difference that it has a separate subsection. On the contrary, models presented in this subsection are meant to process sparse inputs, that may lie in a topology (e.g., left/right) or not (e.g., blue/red). Consequently, the argmax aggregator is the only one that is always suitable for these models. Given the system dynamics, there should be no ambiguity anyway.

Drift-diffusion model (DDM) and race model (RM)

The DDM is the seminal accumulator model, and a baseline on which other models are based (Bogacz et al, 2006). Given a stimulus of intensity a_k , the model accumulates an activity y_k (“evidence” in the DDM literature) over time (Ratcliff and McKoon, 2008):

$$\tau \frac{\Delta y_k}{\Delta t} = a_k \quad (14)$$

This is equivalent to:

$$y_k(t + \Delta t) = y_k(t) + \frac{\Delta t}{\tau} a_k(t + \Delta t) \quad (15)$$

although we will keep the first, lighter writing style for all the following models, as it is easier to read.

Our implementation is actually made of several DDM units in parallel. So when multiple (traditionally two) stimuli are put in competition, one way to make a decision is to run one DDM per stimulus and pick the first to have its activity reach a given threshold. This algorithm is called a “Race model” (RM) (Bogacz et al, 2006). In our case, for comparison purposes, we will instead add the argmax aggregator at all times (figure 11).

Feed-forward inhibition (FFI)

This model (figure 10) is designed to put several stimuli in competition. Each accumulator unit is positively stimulated by one input activity and inhibited by all others:

$$\tau \frac{\Delta y_k}{\Delta t} = a_k - w_- \sum_{j \neq k} a_j \quad (16)$$

The actual implemented equation is found from (16) the same way equation (14) is found from (15).

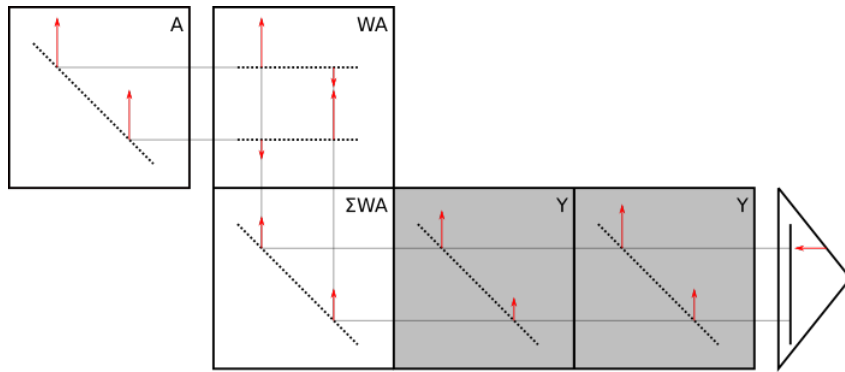


Figure 10 Main steps of a FFI

Ohrstein-Uhlenbeck model (OUM)

This (figure 11) is an upgrade of the DDM with the addition of a leakage term $\lambda > 0$:

$$\tau \frac{\Delta y_k}{\Delta t} = a_k - \lambda y_k \quad (17)$$

It allows the accumulator activity to converge when the stimulus amplitude stagnates, contrarily to the previous two models, in which activity may diverge to infinity. All the models that follow include this stabilization term.

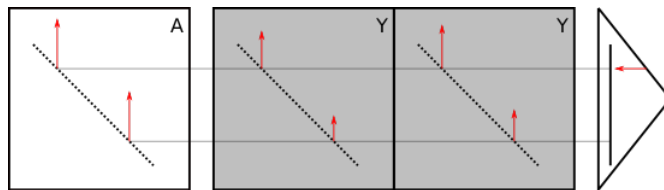


Figure 11 Main steps of a RM or OUM. The difference between the two is that given constant inputs a_k , RM activity will increase indefinitely, whereas OUM activity should converge to a_k/λ due to the leakage term.

Leaky competing accumulator (LCA)

The novelty of this model (figure 12) is that the activities are put in competition and inhibit each other (Usher and McClelland, 2001). Also, a term of self-excitation is added:

$$\tau \frac{\Delta y_k}{\Delta t} = a_k - \lambda y_k + w_+ y_k - w_- \sum_{j \neq k} y_j \quad (18)$$

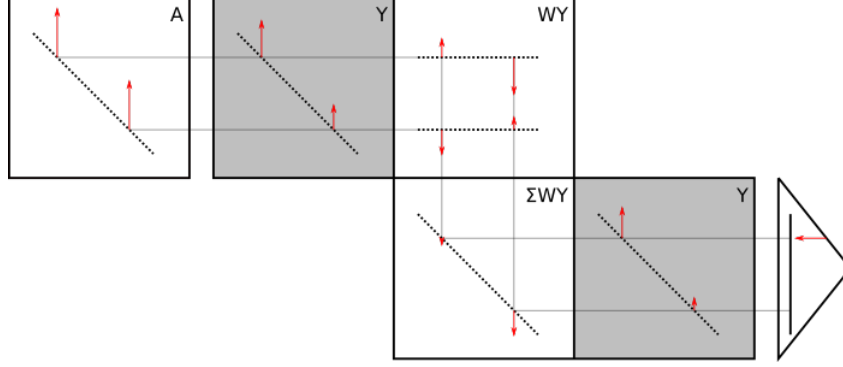


Figure 12 Main steps of a LCA

Nonlinear LCA (NLCA)

Now (figure 13), we differentiate the model activity from its output. The output is obtained by putting the potential through an activation function (Bogacz et al, 2007):

$$\begin{cases} \tau \frac{\Delta u_k}{\Delta t} = a_k - \lambda u_k + w_+ u_k - w_- \sum_{j \neq k} y_j \\ y_k = f(u_k) \end{cases} \quad (19)$$

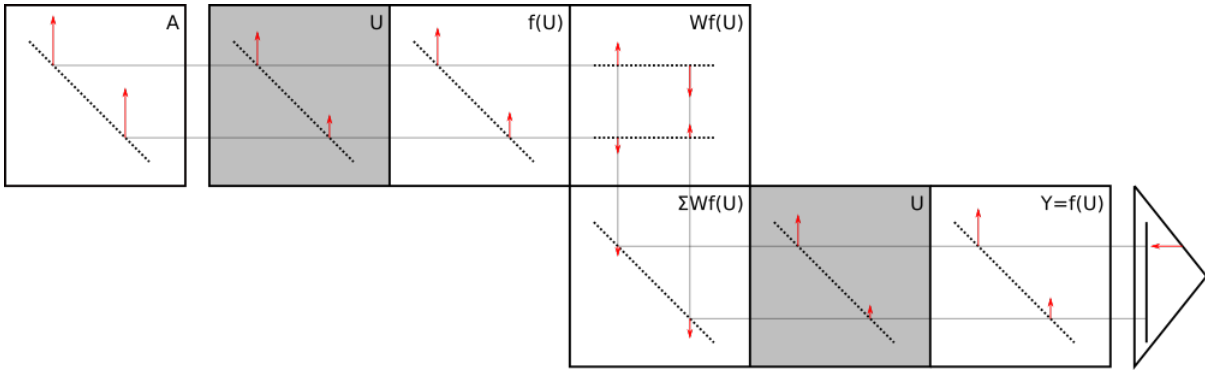


Figure 13 Main steps of a NLCA

Pooled inhibition model (PIM)

Contrarily, to LCA, in the PIM (figure 14), inhibition is shared (Wang, 2002). A new accumulator is added that gets stimulated by the others and inhibits them all:

$$\begin{cases} \tau \frac{\Delta y_k}{\Delta t} = a_k - \lambda y_k + w_+ y_k - w_- y_I \\ \tau \frac{\Delta y_I}{\Delta t} = -\lambda_I y_I + w_I \sum_j y_j \end{cases} \quad (20)$$

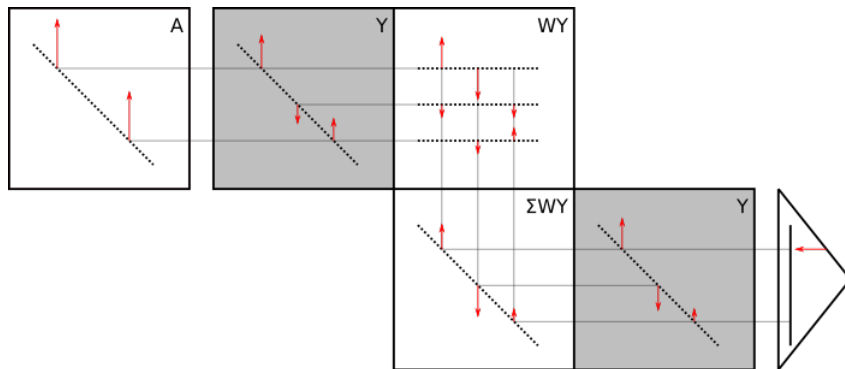


Figure 14 Main steps of a PIM

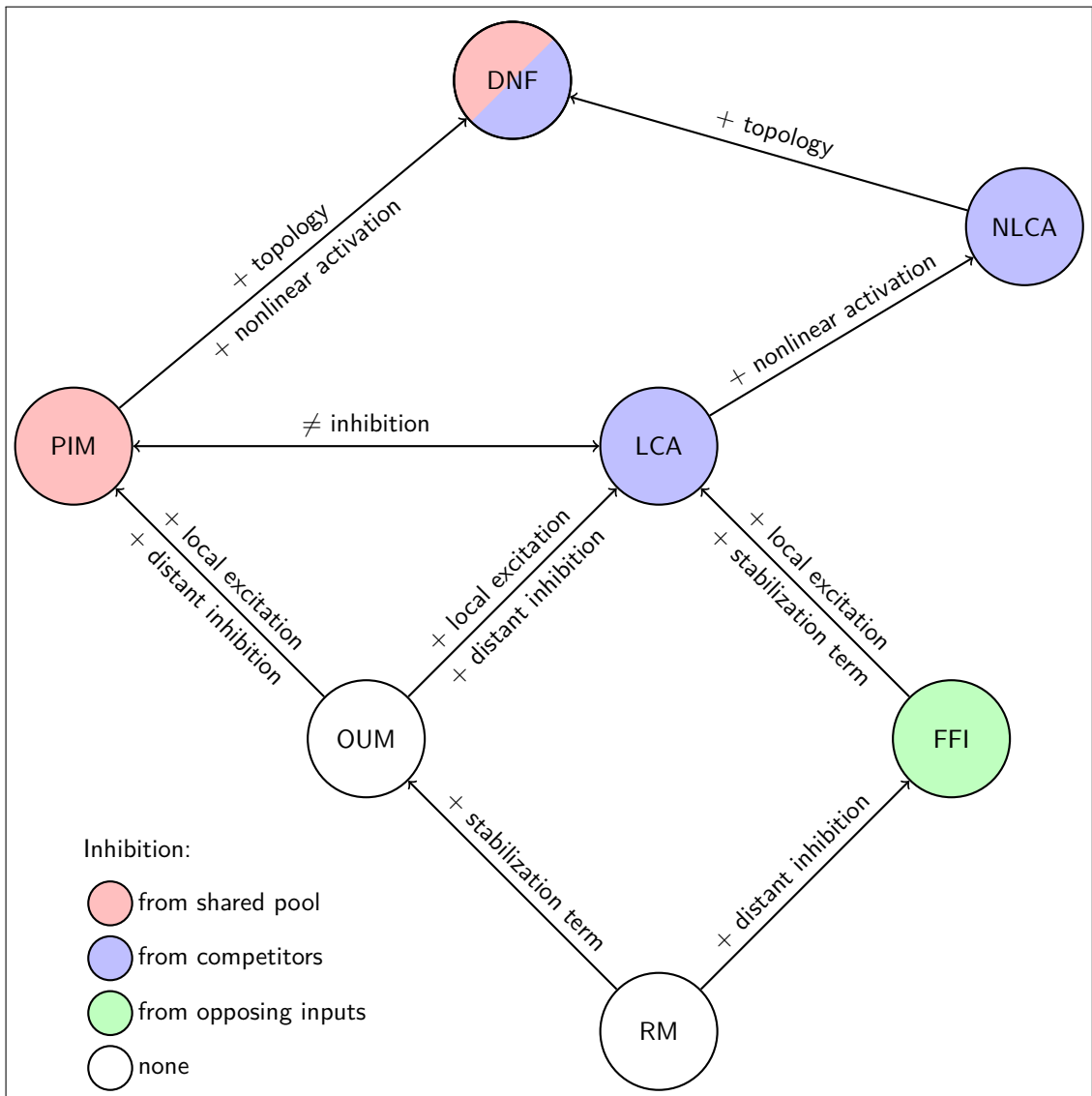


Figure 15 Relations between accumulator models. Adapted from (Bogacz et al, 2006) with added DNF.

2.3.5 Dynamic neural fields (DNF)

DNF describe the evolution of mean field potential over a continuous domain such as the average membrane potential of neurons on a mesoscopic scale (Trappenberg et al,

2001; Wilimzig et al, 2006). They can be used to bridge the gap between microscopic-scale neural processes and macroscopic behavioral data (Fix et al, 2011; Taouali et al, 2015).

DNF originated as a mathematical model of neural dynamics (Wilson and Cowan, 1973; Amari, 1977). While the first descriptions of membrane potential in neural maps date back to the 1950s, (Wilson and Cowan, 1973) were among the first to propose an algorithmic implementation of it. (Amari, 1977) expanded their work by describing in details the behaviors that could emerge from this model of neural dynamics. There are three main categories of behaviors: a monostable field where all excitation eventually dies out; a monostable field where activity increases indefinitely; a bistable field where two different states can be reached depending on the successive presentations received as inputs. To summarize, in a bistable field, once a stimulus is selected, switching focus becomes much harder. This property has made DNF a popular computational model in the study of attention mechanisms (Rougier and Vitay, 2006; Babaie-Janvier and Robinson, 2019). Extensive analytical studies on the emerging properties of DNF have been made by Gregor Schöner, John Spencer and their teams, and are now condensed in a book (Schöner et al, 2015).

From a computational aspect, DNF (figure 16) can be seen as an extension of NLCA to a regularly discretized continuous domain, where each unit acts as an accumulator:

$$\begin{cases} \tau \frac{\Delta u_k}{\Delta t} = a_k - \lambda u_k + w_+ \sum_j \exp^{-\frac{\|x_{i_k} - x_{i_j}\|^2}{\sigma_+^2}} y_j - w_- \sum_j \exp^{-\frac{\|x_{i_k} - x_{i_j}\|^2}{\sigma_-^2}} y_j + h \\ y_k = f(u_k) \end{cases} \quad (21)$$

The amount of interaction in the model is determined by parameters w_+ , w_- , σ_+ and σ_- . DNF are updated by convoluting their output with a kernel made of a difference of Gaussians, shaped like a mexican hat: strong close-range excitation

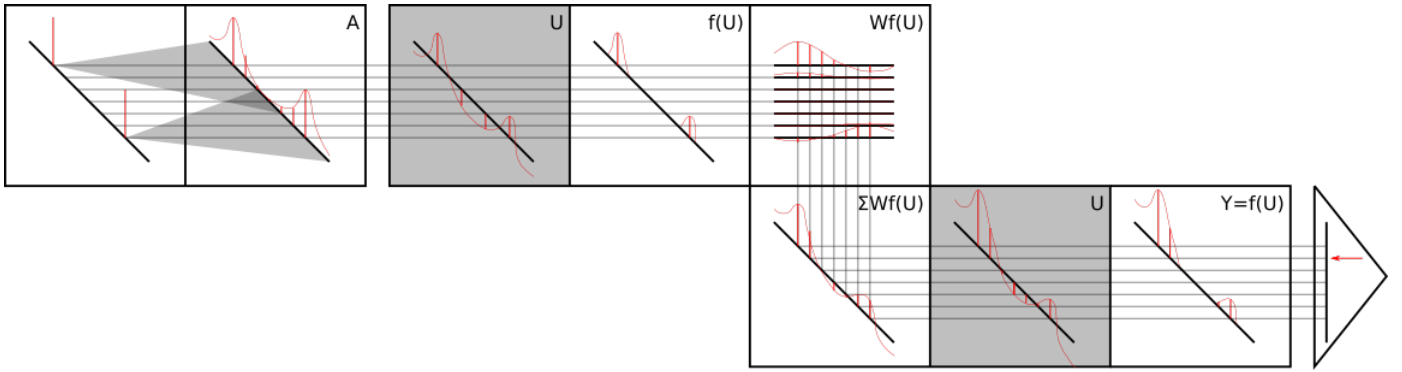


Figure 16 Main steps of a DNF

and moderate long-range inhibition ($w_+ > w_- > 0$, $\sigma_+ < \sigma_-$) (Amari, 1977). As a result, close-by units enhance each other while distant ones go in competition, until a stereotypical peak of activity (sometimes called a bubble) emerges.

The resting level $h \leq 0$ is a parameter that does not appear in all accumulator models. It serves to create an initial resting state with negative potential, so that activity is only produced once strong enough stimuli are received. That parameter can be adjusted easily to filter out low noise or weak stimuli, but it is not always necessary. We maintain it to 0 in this implementation.

In parallel, one common variant of DNF is to integrate all output activity into a global inhibition term (as if $\sigma_- \rightarrow +\infty$). This ensures that competition between stimuli encompasses the entire field (instead of a more or less wide neighbourhood). In that case, DNF can also be seen as a generalization of PIM (see recap figure 15).

As the DNF activity converges into (usually one) bubble, a decision can be interpreted from either a WS or WTA of the outputs y_k . The only situation where these aggregators can yield different results is in cases where more than one bubble reach a stable state, and that can be easily prevented with a high enough σ_- .

2.3.6 Comparison

A comparison of the main design properties of the models is given in table 2. We can already see that depending on the task (stimuli topologically correlated, sparsity of inputs, time relation), some models are more suitable than others. But these models can also be classified according to the level of abstraction at which they compute activity. Figure 17 shows how some of these models fit on Marr’s hierarchy (Marr, 1982). For example, models based on Bayesian theory are at the level of computational theory, making assumptions on the distribution of inputs and outputs, and explaining the processing with a theoretical paradigm, with little focus on how the computation is made. Models such as FL, DNF, FFI and NLCA are on the representation–algorithm level, where the operations are explained but the outcome is measured after the fact, and not theorized beforehand. Zooming in on the latter three models, we can look at the units that constitute them and that can be likened to sets of DDM or OUM. These can be placed at the hardware level, as they simulate the physical operations that implement decision-making, mimicking actual neurons or cortical columns.

To discern even more the most complex models (FL, KF and DNF), we can differentiate them by the kernel with which inputs are confronted: triangular for FL, Gaussian for KF, a difference of Gaussians for DNF. This competition also does not occur at the same time for all models. For both FL and DNF, inputs are divided, matched to the kernel then put in competition with each other. But for KF, inputs are matched to a Gaussian, all together, then aggregated. For both KF and DNF, the state of the model is gradually updated over time. But for FL, the state is reset at each time step, i.e. there is no memory trace.

2.3.7 Implementation

The properties given in last section have repercussions on the way models integrate inputs in space and over time. In order to make a broad overview of the achievable

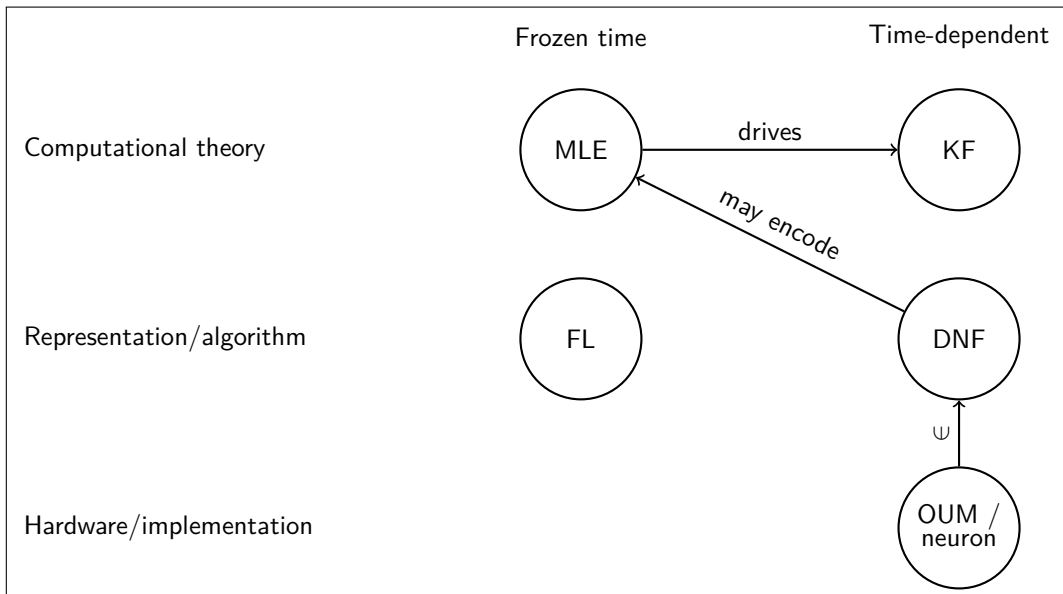


Figure 17 Positioning of models in Marr’s hierarchy. FFI, LCA, NLCA and PI can be put in the place of DNF, and DDM of OUM, following the relationships described in figure 15 . Models in the first column operate independently of time: at each time step, an output is given as if time was frozen and inner operations had fully converged. Models in the second column are iterative and may behave differently depending on simulation time step.

properties, we select a varied sample of all models presented. We pick two model instances from each subsection: WTA and FL, WS and KF, FFI (the simplest model with interaction) and NLCA (the most complete accumulator model outside DNF), and DNF. Since DNF can produce very different behaviors depending on its parametrization, we use two very different set of parameters to give a glimpse of the range of properties available. Initialization is made to zero potential, also setting $h = 0$. We take ReLU as the activation function, and WS as the aggregator. The parameters used for all models are listed in table 3. Parameter values were selected through expert knowledge and preliminary tests.

Here is a quick breakdown of the way parameter values are picked. For FL, we take a α value in-between WTA-like behavior (too low) and WS-like behavior (too high). For KF, q has to be low enough for internal states to matter, but not too low,

otherwise new inputs are eventually ignored. For DNF (and FFI/NLCA likewise), previous research on parameter effects has been presented in (Forest et al, 2022b).

3 Results

The evolution over time of activities and decisions of the 8 models in the 8 scenarios (from figure 1) is given in figure 18. To describe a result, we use the acronym of the model followed by the scenario letter in superscript, e.g., WTA^E designates the output of WTA in scenario E. Before we detail the main takeaways, here are some explanations to help understand the figure:

- WTA returns a single position with an activity equal to the maximum intensity. The activity is plotted with a thick line for visualization. When the input is completely empty (beginning of WTA^E), the center of the field is returned by default. Same goes for WS^E , FFI^E and $NLCA^E$. When no decision is rendered by a model, the default answer is plotted in blue.
- As a reminder, FL returns the intersection (minimum) of truncated triangles. In FL^A , the triangles overlap slightly in the middle of the field. In FL^B , they do not overlap, all that remain are the truncatures: either $1 - 1 = 0$ (fuzzification of the left stimulus) or $1 - 0.99 = 0.01$ (right stimulus). This results in a 0.01 plateau centered on the left stimulus (the only place where it is not truncated to 0). This is why a decision can be made even if no activity is visible.
- KF activity is a Gaussian, where the variance is updated depending on the variance of all inputs projected in a base of Gaussians. The less variance in the input, the thinner the output.
- In $NLCA^{A/B}$ and $DNF^{A/B}$, we can distinguish two phases. First, peaks appear at the position of each stimulus, of apparent equal activity. After a certain delay, the slight superiority of one stimulus (or group of stimuli) allows one peak to grow stronger, self-excite more than it is inhibited by the others, and inhibit the others

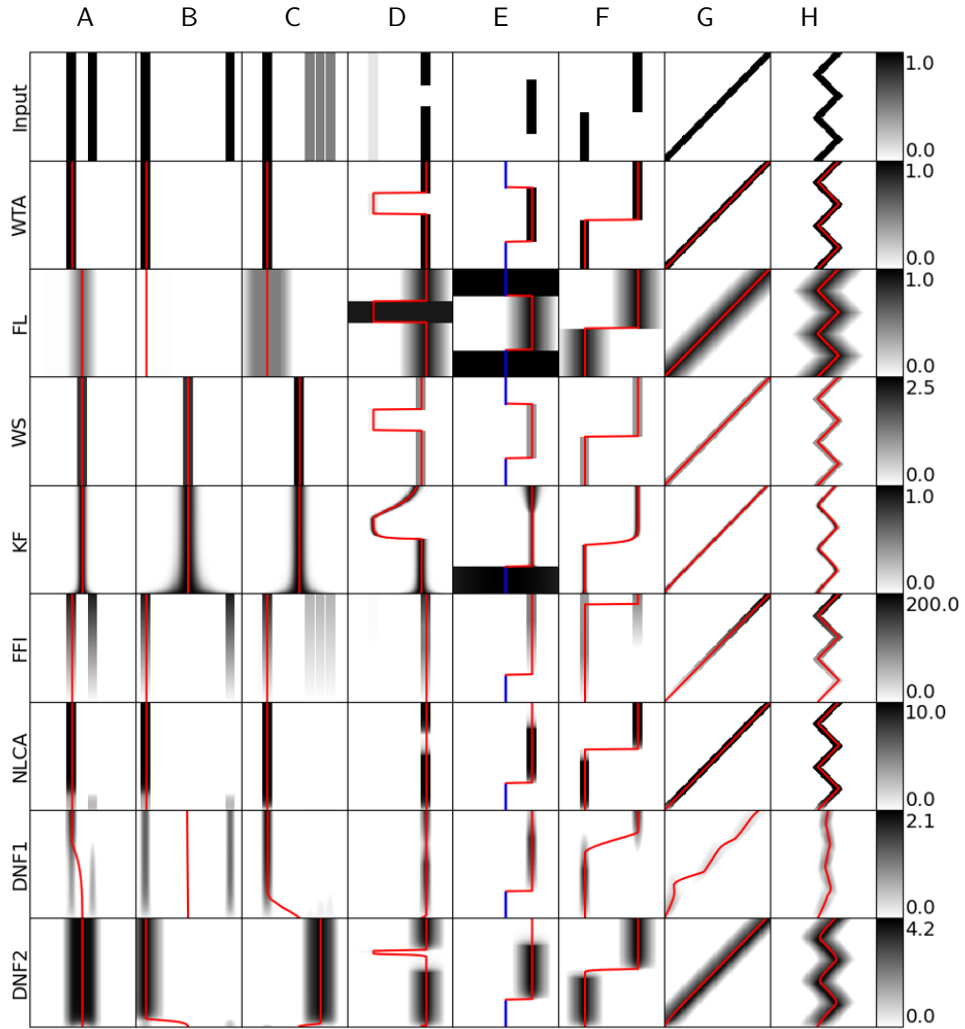


Figure 18 Evolution over time (y -axis, starting at bottom) of activities $\{y_i\}$ or \bar{y} (grayscale surfaces) and decisions (red lines) of 8 models (rows 2-9) in 8 scenarios (row 1). Blue segments indicate a default decision (in the middle of the field) when models have an empty or invalid output.

more than they self-excite. The shift is not visible for NLCA decision (red line) because it is discrete, but since DNF use a barycenter for aggregation, we can see clearly the potential shift from undecided competition to selection (DNF_1^A , DNF_2^B , DNF_1^C , DNF_2^C).

Selection and interpolation

As we can see in scenarios A and B, some models are specialized in selecting only the strongest stimulus (WTA, FFI, NLCA) and some at making an interpolation (WS,

KF). Two can implement both behaviors. FL will either select (FL^B) or interpolate (FL^A) depending on the proximity between the stimuli. The gap at which it switches behaviors can be controlled through its slope parameter α . This distinction can also be made with DNF, except it is controlled mostly by the width of lateral excitation σ_+ , which determines how close stimuli must be to be able to fit inside a single bubble of activity. But the width of lateral inhibition σ_- is not neglectible. DNF_1^B shows a case where neither selection nor interpolation occurs: the interaction kernel is too thin for the stimulated region to affect each other, so the two stimuli are selected separately. If this particular behavior is unwanted, it is common to use a global inhibition term (i.e. an infinite σ_-).

Greediness

Scenario C opposes one strong stimulus to a concentrated group of smaller stimuli of higher total intensity. Most models will take the greedy approach and pick the strongest stimulus, as selecting the group requires taking their proximity into account. WS and KF do it to a certain degree, as the bigger weight of the group attracts the barycenter towards it. Regarding DNF, the outcome will again depend on the width of the interaction kernel. DNF favor stimuli that match its kernel. With a thin kernel, the lone stimulus will be picked more easily than the group, in which every component goes in competition with one another. With a large kernel, the group can be merged into a single, big bubble, that prevails over more isolated stimuli.

Robustness to temporary obstruction

In scenario D, two targets are present. The question is whether the model will immediately start changing target when the one it initially focused temporarily disappears, or keep the focus for a certain time. Here, there is a clear difference between accumulator-based models and the others. The time constant will ensure that the disappearance of the target will not be integrated instantly. Parameter τ can be tuned to control the

update speed: the second instance of DNF, because it has a lower value of τ than the others, starts switching attention before the first target reappears.

Reaction time

Scenarios E and F are useful to compare the time dynamics of either KF and all accumulator models. For KF, when a new stimulus appears, the activity will start shifting instantly. For discrete accumulators, the change is taken into account, but there is a delay before the maximum activity changes side. DNF is a mix of both: like KF, the spatial continuity allows for a gradual shift towards the newer stimulus, except a time lag is induced by the temporal dynamics of the differential equation, similarly to FFI and NLCA.

The convergence time of KF can be tuned to a small degree via its parameter q , but it does not give as much leeway as some accumulators and their time constant τ . However, FFI, NLCA and DNF behave differently when a switch occurs between two stimuli. FFI will lower its activity at the first unit and increase its activity at the other, symmetrically to what it did before the switch. So the moment the model will actually change targets will depend only on the inputs (here, the delay between input switch and output switch is exactly the same as the duration for which the first stimulus was presented). For NLCA, it will depend mostly on its leakage term λ . For DNF, one has to also consider the strength of lateral interactions.

Tracking speed

This one is only relevant for topology-based models with a time dependency, i.e. KF^G , DNF_1^G and DNF_2^G . The others will obviously track instantly a single moving target. Both KF and DNF can track the target with a minor delay. But DNF, depending on its parameters, might fail to track the target smoothly. With a high integration time τ and a small kernel, the stimulus might shake off the current active bubble, so a new bubble ends up appearing at the new stimulus position from time to time.

Trajectory smoothing

Again, only KF and DNF can smooth a trajectory that changes frequently. Three behaviors can be obtained depending on parameters q and τ respectively: follow the target faithfully, including in sharp turns (high q , low τ); round the turns (low q , medium τ); or converge to a seemingly average position ($q = 0$, very high τ).

A summary of these observations is given in table 4. DNF are by far the most versatile, which is consistent with their higher number of parameters. The downside is that fitting them to achieve a specific task can prove to be difficult (Quinton, 2010; Forest et al, 2022b). An approximate computation time with a regular CPU for each model (not counting input processing or output aggregation) is also given in the table. Unsurprisingly, models with the least amount of steps, WS and WTA, are the fastest. However, this might be quite dependent to the (Python) implementation. For example, FFI is computed by multiplying the vector of inputs by a matrix containing 1 in its diagonal and $-w_-$ elsewhere. Matrix multiplication in `numpy`, Python’s standard mathematical library, is slower than other operations, including 1D convolution, which is why FFI seems slower than NLCA or DNF despite its simpler design. .

4 Discussion

Decision-making tasks can not all be achieved by a one-size-fits-all model. DNF appear to be the most versatile, failing only with sparse signals in a continuous domain, because it does not suffice to generate a peak of activity, and no interaction occurs. This is not a very realistic use case, and it can be avoided by “Gaussianizing” the stimuli. On the other hand, their theoretical and computational complexity may not be warranted in every scenario. In competition tasks where topology is not relevant, accumulator models such as NLCA show similar properties to DNF for a lower cost. Finding a trade-off between conflicting stimuli can be done by either KF or FL, the

latter being also able to switch between selecting the best (WTA) and interpolating between them (akin to WS) depending on their proximity.

Models tested here are quite bare, and there is always room for refining and extending them. Parameters can be tuned to change behavior. We show two different examples of it with DNF, but another one would be memory: increasing w_+ sufficiently leads to self-sufficient peaks to be formed, that stay in place even after the stimulus has disappeared. Furthermore, numerous extensions are available in the literature. WTA can be combined with a kernel to include neighbors in the aggregation. It can also be enhanced with iterative elements (winner takes most). FL has seen various implementations, most notably fuzzy inference systems by (Mamdani and Assilian, 1975) and (Sugeno and Yasukawa, 1993). WS can be expanded with kernel methods such as SVM. KF has numerous extensions, most notably the extended Kalman filter, one of the most used estimation algorithms for nonlinear systems (Julier and Uhlmann, 2004). DNF can be adapted to sparse inputs with a variation called sparse neural field (Quinton and Girau, 2010), though it is less robust. It can also be altered to incorporate predictive and active aspects (Quinton and Girau, 2011; Quinton and Goffart, 2018), which reinforce tracking abilities and robustness to distractor and oclusions.

One particular aspect of decision-making that is often overlooked is its relation to perception and cognition. More often than not, decision is more than a posthoc filtering of the model output: the data is already filtered inside the model through thresholding, attentional processes And like decision drives action, action also impacts decision, through predictive aspects for example. The decision-making algorithm must be put in context of the cognitive system it belongs to. The choice between one strong stimulus and a big group of weaker stimulus, between attending the expected position of a stimulus and exploring unexpected ones, etc., depends on both the task and the system cognition. For instance, ignoring a distractor may be more important when the system is moving or acting towards a previously-selected target, than when it

is still figuring out what to do. A decision-making system is often made of several components in perpetual interaction (e.g. extended KF and FL (Das et al, 2017)), and this is how more complex, interesting and robust behaviors may emerge.

Meanwhile, decision-making models have to face a variety of constraints. The first is an issue of scalability. Inputs may bear a high dimensionality, which are increasingly harder to process for models of high complexity. At the same time, algorithms may be faced with computational constraints: limited processing power, memory, time. Part of this can be mitigated with some optimizations: some algorithms may be required to perform numerical approximations (e.g., replacing convolutions by products in the frequency domain after a Fourier transform), data reduction (e.g., through PCA/SVD and projecting the data or kernels on a subspace to reduce algorithmic complexity), and changes in structure and coding of data. Another constraint is the possibility (or not) to parallelize computations for speed. Sometimes, it might even be a requirement to make an algorithm distributed (as in multi-agent systems), or centralized instead.

Acknowledgments

This research was supported over the 2010-2023 period by: Inria Nancy-Grand Est – LORIA; European FP7 program – EUCogIII (3rd European Network for the Advancement of Artificial Cognitive Systems, Interaction and Robotics) under grant agreement 269981; French program “Investissement d’avenir” through ANR, European Union (Auvergne European Regional Development Funds) and from “Région Auvergne” in the framework of the LabEx IMobS3 (Innovative Mobility: Smart and Sustainable Solutions) (ANR-10-LABX-16-01); Idex Paris-Saclay – iCODE (Institut pour le Contrôle et la Décision) (ANR-11-IDEX-0003); French region Auvergne-Rhône-Alpes through Pack Ambition Recherche initiative (AMPLIFIER project); French National Research Agency in the framework of the “Investissements d’avenir” (ANR-15-IDEX-02 and ANR-11-LABX-0025-01); Pôle Grenoble Cognition (FR 3381 CNRS, Univ.

Grenoble Alpes, Grenoble INP); French National Research Agency – 3AI institutes – MIAI@Grenoble Alpes (ANR-19-P3IA-0003).

Compliance with ethical standards

Conflict of interest

The authors have no relevant financial or non-financial interests to disclose.

Code availability

The Python program with which this review was made is openly available at **WIP**.

References

- Alais D, Burr D (2004) The ventriloquist effect results from near-optimal bimodal integration. *Current Biology* 14(3):257–262. <https://doi.org/10.1016/j.cub.2004.01.029>
- Amari SI (1977) Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics* 27(2):77–87. <https://doi.org/10.1007/BF00337259>
- Arulkumaran K, Deisenroth MP, Brundage M, et al (2017) Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine* 34(6):26–38
- Babaie-Janvier T, Robinson PA (2019) Neural field theory of corticothalamic attention with control system analysis. *Frontiers in Neuroscience* 13:1240
- Bajrami X, Dërmaku A, Demaku N (2015) Artificial neural fuzzy logic algorithm for robot path finding. *IFAC-PapersOnLine* 48(24):123–127
- Bellman RE, Zadeh LA (1970) Decision-making in a fuzzy environment. *Management science* 17(4):141–164

- Bicho E, Schöner G (1997) The dynamic approach to autonomous robotics demonstrated on a low-level vehicle platform. *Robotics and autonomous systems* 21(1):23–35
- Bicho E, Mallet P, Schöner G (2000) Target representation on an autonomous vehicle with low-level sensors. *The International Journal of Robotics Research* 19(5):424–447
- Bitzer S, Park H, Blankenburg F, et al (2014) Perceptual decision making: drift-diffusion model is equivalent to a bayesian model. *Frontiers in Human Neuroscience* 8
- Bogacz R, Brown E, Moehlis J, et al (2006) The physics of optimal decision making: a formal analysis of models of performance in two-alternative forced-choice tasks. *Psychological review* 113(4):700
- Bogacz R, Usher M, Zhang J, et al (2007) Extending a biologically inspired model of choice: multi-alternatives, nonlinearity and value-based multidimensional choice. *Philosophical Transactions of the Royal Society B: Biological Sciences* 362(1485):1655–1670
- Braitenberg V (1986) *Vehicles: Experiments in synthetic psychology*. MIT press
- Brooks RA (1991) Intelligence without representation. *Artificial intelligence* 47(1):139–159
- Buss AT, Spencer JP (2018) Changes in frontal and posterior cortical activity underlie the early emergence of executive function. *Developmental Science* 21(4):e12602
- Calvert G, Spence C, Stein B (2004) *The Handbook of Multisensory Processes*. A Bradford book, MIT Press

- Castanedo F (2013) A review of data fusion techniques. *The scientific world journal* 2013
- Chen SY (2011) Kalman filter for robot vision: a survey. *IEEE Transactions on industrial electronics* 59(11):4409–4420
- Conrey FR, Sherman JW, Gawronski B, et al (2005) Separating multiple processes in implicit social cognition: the quad model of implicit task performance. *Journal of personality and social psychology* 89(4):469–487
- Das TK, Harischandra PD, Abeykoon AHS (2017) Extended kalman filter based fusion of reliable sensors using fuzzy logic. In: 2017 Moratuwa Engineering Research Conference (MERCCon), IEEE, pp 58–63
- Doki K, Suyama K, Funabara Y, et al (2015) Robust localization for mobile robot by K-L divergence-based sensor data fusion. In: IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society, pp 2638–2643, <https://doi.org/10.1109/IECON.2015.7392499>
- Dubois D, Perny P (2016) A review of fuzzy sets in decision sciences: Achievements, limitations and perspectives. In: Greco S, Ehrgott M, Figueira JR (eds) *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer New York, New York, NY, p 637–691
- Dubois D, Foulloy L, Mauris G, et al (2004) Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. *Reliable computing* 10(4):273–297
- Ernst MO, Banks MS (2002) Humans integrate visual and haptic information in a statistically optimal fashion. *Nature* 415:429–433. <https://doi.org/10.1038/415429a>

- Escobar MJ, Alexandre F, Viéville T, et al (2022) Bio-inspired Robotics, Springer International Publishing, Cham, pp 161–194. https://doi.org/10.1007/978-3-319-40003-7_8
- Falandays JB, Spevack S, Pärnamets P, et al (2021) Decision-making in the human-machine interface. *Frontiers in Psychology* 12. <https://doi.org/10.3389/fpsyg.2021.624111>
- Fix J, Rougier N, Alexandre F (2011) A dynamic neural field approach to the covert and overt deployment of spatial attention. *Cognitive Computation* 3(1):279–293. <https://doi.org/10.1007/s12559-010-9083-y>
- Forest S, Quinton JC, Lefort M (2022a) Combining manifold learning and neural field dynamics for multimodal fusion. In: 2022 International Joint Conference on Neural Networks (IJCNN), pp 1–8, <https://doi.org/10.1109/IJCNN55064.2022.9892614>
- Forest S, Quinton JC, Lefort M (2022b) A dynamic neural field model of multimodal merging: application to the ventriloquist effect. *Neural Computation* 34(8):1701–1726. https://doi.org/10.1162/neco_a_01509
- Friston K, Schwartenbeck P, FitzGerald T, et al (2013) The anatomy of choice: active inference and agency. *Frontiers in human neuroscience* 7. <https://doi.org/10.3389/fnhum.2013.00598>
- Gepperth A, Lefort M (2016) Learning to be attractive: Probabilistic computation with dynamic attractor networks. In: 2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), pp 270–277, <https://doi.org/10.1109/DEVLRN.2016.7846831>
- Goertzel B (2014) Artificial general intelligence: concept, state of the art, and future prospects. *Journal of Artificial General Intelligence* 5(1):1

- Goertzel B, Pitt J, Wigmore J, et al (2011) Cognitive synergy between procedural and declarative learning in the control of animated and robotic agents using the OpenCogPrime AGI architecture. In: Twenty-Fifth AAAI Conference on Artificial Intelligence
- Gold JJ, Shadlen MN (2001) Neural computations that underlie decisions about sensory stimuli. *Trends in cognitive sciences* 5(1):10–16
- Gold JJ, Shadlen MN (2007) The neural basis of decision making. *Annual Review of Neuroscience* 30(1):535–574
- Griegen R, Tekülve J, Zibner SKU, et al (2020) Scene memory and spatial inhibition in visual search. *Attention, Perception, & Psychophysics* 82(2):775–798
- Incera S, Markis TA, McLennan CT (2013) Mouse-tracking reveals when the Stroop effect happens. *The Ohio Psychologist* 2013(August):33–34
- Julier SJ, Uhlmann JK (2004) Unscented filtering and nonlinear estimation. *Proceedings of the IEEE* 92(3):401–422
- Kaelbling LP, Littman ML, Moore AW (1996) Reinforcement learning: A survey. *Journal of artificial intelligence research* 4:237–285
- Kahneman D, Tversky A (2013) Prospect theory: An analysis of decision under risk. In: *Handbook of the fundamentals of financial decision making: Part I*. World Scientific, p 99–127
- Kalman RE (1960) A new approach to linear filtering and prediction problems. *Journal of Basic Engineering* 82(1):35–45
- Khansari-Zadeh SM, Billard A (2011) Learning stable nonlinear dynamical systems with gaussian mixture models. *IEEE Transactions on Robotics* 27(5):943–957

- Kingma DP, Welling M (2019) An introduction to variational autoencoders. *Foundations and Trends in Machine Learning* 12(4):307–392. <https://doi.org/10.1561/22000000056>
- Kohonen T (2012) *Self-organizing maps*, vol 30. Springer Science & Business Media
- Krishnapuram R, Keller JM (1992) Fuzzy set theoretic approach to computer vision: An overview. In: [1992 Proceedings] *IEEE International Conference on Fuzzy Systems*, IEEE, pp 135–142
- LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):436–444. <https://doi.org/10.1038/nature14539>
- Lepora NF, Gurney KN (2012) The basal ganglia optimize decision making over general perceptual hypotheses. *Neural Computation* 24(11):2924–2945
- Lepora NF, Pezzulo G (2015) Embodied choice: how action influences perceptual decision making. *PLoS computational biology* 11(4):e1004110
- Mamdani EH, Assilian S (1975) An experiment in linguistic synthesis with a fuzzy logic controller. *International journal of man-machine studies* 7(1):1–13
- Marr D (1982) *Vision. A Computational Investigation Into the Human Representation and Processing of Visual Information*. MIT Press
- Ming Y, Cao S, Zhang R, et al (2017) Understanding hidden memories of recurrent neural networks. In: *2017 IEEE conference on visual analytics science and technology (VAST)*, IEEE, pp 13–24
- Mnih V, Kavukcuoglu K, Silver D, et al (2015) Human-level control through deep reinforcement learning. *Nature* 518(7540):529–533

- Pezzulo G, Verschure PFMJ, Balkenius C, et al (2014) The principles of goal-directed decision-making: from neural mechanisms to computation and robotics. *Philosophical Transactions of the Royal Society B: Biological Sciences* 369(1655):20130470. <https://doi.org/10.1098/rstb.2013.0470>
- Pio-Lopez L, Nizard A, Friston K, et al (2016) Active inference and robot control: a case study. *Journal of The Royal Society Interface* 13(122):20160616. <https://doi.org/10.1098/rsif.2016.0616>
- Plataniotis KN, Hatzinakos D (2000) Gaussian mixtures and their applications to signal processing. In: Stergiopoulos S (ed) *Advanced Signal Processing Handbook: Theory and Implementation for Radar, Sonar, and Medical Imaging Real Time Systems* (1st ed.). CRC Press, p 89–124
- Quinton JC (2010) Exploring and optimizing dynamic neural fields parameters using genetic algorithms. In: *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pp 1–7, <https://doi.org/10.1109/IJCNN.2010.5596293>
- Quinton JC, Girau B (2010) A sparse implementation of dynamic competition in continuous neural fields. In: *Brain Inspired Cognitive Systems 2010 - BICS 2010*, Madrid, Spain
- Quinton JC, Girau B (2011) Predictive neural fields for improved tracking and attentional properties. In: *The 2011 International Joint Conference on Neural Networks*, IEEE, pp 1629–1636
- Quinton JC, Goffart L (2018) A unified dynamic neural field model of goal directed eye movements. *Connection Science* 30(1):20–52. <https://doi.org/10.1080/09540091.2017.1351421>

- Qureshi MS, Swarnkar P, Gupta S (2018) A supervisory on-line tuned fuzzy logic based sliding mode control for robotics: An application to surgical robots. *Robotics and Autonomous Systems* 109:68–85
- Rasmussen CE (2004) Gaussian processes in machine learning. In: Bousquet O, von Luxburg U, Rätsch G (eds) *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures*. Springer Berlin Heidelberg, Berlin, Heidelberg, p 63–71
- Ratcliff R (2018) Decision making on spatially continuous scales. *Psychological review* 125(6):888–935
- Ratcliff R, McKoon G (2008) The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation* 20(4):873–922
- Rohde M, van Dam LC, Ernst MO (2016) Statistically optimal multisensory cue integration: A practical tutorial. *Multisensory Research* 29(4-5):279–317. <https://doi.org/10.1163/22134808-00002510>
- Rougier NP, Vitay J (2006) Emergence of attention within a neural population. *Neural Networks* 19(5):573–581
- Roxin A (2019) Drift–diffusion models for multiple-alternative forced-choice decision making. *The Journal of Mathematical Neuroscience* 9(1):1–23
- Russo F, Ramponi G (1994) Fuzzy methods for multisensor data fusion. *IEEE transactions on instrumentation and measurement* 43(2):288–294
- Sandamirskaya Y (2014) Dynamic neural fields as a step toward cognitive neuromorphic architectures. *Frontiers in Neuroscience* 7:276. <https://doi.org/10.3389/fnins.2013.00276>

- Scarpina F, Tagini S (2017) The Stroop color and word test. *Frontiers in Psychology* 8
- Schöner G, Spencer J, DFT Research Group (2015) *Dynamic Thinking: A Primer on Dynamic Field Theory*. Oxford Series in Developmental Cognitive Neuroscience, Oxford University Press, <https://doi.org/10.1093/acprof:oso/9780199300563.001.0001>
- Sigaud O, Salaün C, Padois V (2011) On-line regression algorithms for learning mechanical models of robots: a survey. *Robotics and Autonomous Systems* 59(12):1115–1129
- Sobrevilla P, Montseny E (2003) Fuzzy sets in computer vision: An overview. *Mathware and Soft Computing* 10(2/3):71–83
- Somvanshi M, Chavan P, Tambade S, et al (2016) A review of machine learning techniques using decision tree and support vector machine. In: 2016 international conference on computing communication control and automation (ICCUBEA), IEEE, pp 1–7, <https://doi.org/10.1109/ICCUBEA.2016.7860040>
- Stroop JR (1935) Studies of interference in serial verbal reactions. *Journal of experimental psychology* 18(6):643–662
- Sugeno M, Yasukawa T (1993) A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on fuzzy systems* 1(1):7–31
- Sun R, Peterson T, Merrill E (1999) A hybrid architecture for situated learning of reactive sequential decision making. *Applied Intelligence* 11(1):109–127
- Taouali W, Goffart L, Alexandre F, et al (2015) A parsimonious computational model of visual target position encoding in the superior colliculus. *Biological Cybernetics* 109(4):549–559. <https://doi.org/10.1007/s00422-015-0660-8>

- Tekülve J, Fois A, Sandamirskaya Y, et al (2019) Autonomous sequence generation for a neural dynamic robot: Scene perception, serial order, and object-oriented movement. *Frontiers in Neurorobotics* 13
- Trappenberg TP, Munoz DP, Klein RM (2001) A model of saccade initiation based on the competitive integration of exogenous and endogenous signals in the superior colliculus. *Journal of Cognitive Neuroscience* 13(2):256–271. <https://doi.org/10.1162/089892901564306>
- Usher M, McClelland JL (2001) The time course of perceptual choice: the leaky, competing accumulator model. *Psychological review* 108(3):550
- Vickers D (1970) Evidence for an accumulator model of psychophysical discrimination. *Ergonomics* 13(1):37–58
- Vijayakumar S, Schaal S (2000) Locally weighted projection regression: An $o(n)$ algorithm for incremental real time learning in high dimensional space. In: *Proceedings of the seventeenth international conference on machine learning (ICML 2000)*, Morgan Kaufmann, pp 288–293
- Wakileh BAM, Gill KF (1988) Use of fuzzy logic in robotics. *Computers in industry* 10(1):35–46
- Wang XJ (2002) Probabilistic decision making by slow reverberation in cortical circuits. *Neuron* 36(5):955–968
- Wijeakumar S, Ambrose JP, Spencer JP, et al (2017) Model-based functional neuroimaging using dynamic neural fields: An integrative cognitive neuroscience approach. *Journal of Mathematical Psychology* 76:212–235

Wilimzig C, Schneider S, Schöner G (2006) The time course of saccadic decision making: Dynamic field theory. *Neural Networks* 19(8):1059–1074. <https://doi.org/10.1016/j.neunet.2006.03.003>

Wilson HR, Cowan JD (1973) A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik* 13(2):55–80

Zadeh LA (1965) Fuzzy sets. *Information and control* 8(3):338–353

A dynamic neural field model of multimodal merging: application to the ventriloquist effect

Simon Forest^{1, 2}, Jean-Charles Quinton¹, Mathieu Lefort²

¹Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, UMR 5224, F-38000, Grenoble, France

²Université de Lyon, Université Claude Bernard Lyon 1, CNRS, LIRIS, UMR 5205, F-69621, Villeurbanne, France

Keywords: Multimodal merging; dynamic neural fields; superior colliculus; selective attention

Abstract

Multimodal merging encompasses the ability to localize stimuli based on imprecise information sampled through individual senses such as sight and hearing. Merging decisions are standardly described using Bayesian models that fit behaviors over many trials, encapsulated in a probability distribution. We introduce a novel computational

model based on Dynamic Neural Fields able to simulate decision dynamics and generate localization decisions, trial by trial, adapting to varying degrees of discrepancy between audio and visual stimulations. Neural fields are commonly used to model neural processes at a mesoscopic scale, for instance neurophysiological activity in the superior colliculus. Our model is fit to human psychophysical data of the ventriloquist effect, additionally testing the influence of retinotopic projection onto the superior colliculus, and also providing a quantitative performance comparison to the Bayesian reference model. While models performs equally on average, a qualitative analysis of free parameters in our model allows insights into the dynamics of the decision and the individual variations in perception caused by noise. We finally show that the increase in the number of free parameters does not result in overfitting, and that the parameter space may either be reduced to fit specific criteria or exploited to perform well on more demanding tasks in the future. Indeed, beyond decision or localization tasks, our model opens the door to the simulation of behavioral dynamics as well as saccade generation driven by multimodal stimulation.

1 Introduction

Humans have versatile and diverse ways of perceiving the world around them. Senses provide a dense and continuous flow of data, yet our ability to process information is limited, so we need to select a subset of all available data in order to engage in adequate interactions with the environment. Performing relevant selection involves processes pertaining to (selective) attention.

Focusing on visual attention, human vision is constrained by the heterogeneous disposition of sensors on the retina, with a denser distribution near the center of the visual field (called fovea). As a consequence, humans will tend to gaze at objects of interest, in order to see them better. One outcome of this kind of overt attention is that it may trigger visual saccades towards objects located in the periphery of the retinotopic space. Because of its weaker resolution, saccades are less precise and more likely to be disturbed by artifacts.

That issue can be circumvented with the use of additional information from other modalities (Calvert et al., 2004). For example, a sound congruent to a visual stimulus may guide saccades to this particular target (Frens et al., 1995; Kapoula and Pain, 2020). Generally speaking, it is common to merge sensory data coming from multiple modalities. They might enhance each other (Meredith and Stein, 1986), complement one another (Newell et al., 2001), or even compete together to form an interpolation of different sensory inputs (McGurk and MacDonald, 1976; Alais and Burr, 2004). These mechanisms depend on the relative reliability of the modalities, with factors including stimulus noisiness (Ernst and Banks, 2002), sensor precision (Witten and Knudsen, 2005), and possible top-down interference (such as selective attention; Driver and Spence, 2004). Studies on this topic vary from macroscopic (at a behavioral level) to microscopic (neurological) scale, but it is common for such insights to be shared across these two domains (Calvert et al., 2004; Alais et al., 2010).

Our aim is to build a computational model of multisensory integration that can be embedded in attention processes. We will focus on audiovisual merging especially.

1.1 Biological inspiration

One source of inspiration for our computational model is the superior colliculus (SC). It has been reported to integrate cues from multiple modalities, including visual, auditory and somatosensory (Wallace and Stein, 1996; Calvert et al., 2004), which makes it a relevant neural structure to be used as a reference for our model. It is also involved in the generation of motor commands such as saccades (Gandhi and Katnani, 2011). However, please note that our purpose is not to build a biologically-accurate simulation of the SC, but rather get inspiration from the brain workflow, for which mesoscopic scale models of multisensory integration are available. Such scale should allow us to remain neurally plausible, as we later turn our attention to macroscopic observations and directly model behavioral data.

In previous works, the SC has already been used as a target of computational models of visual (Taouali et al., 2015) and multimodal (Casey et al., 2012; Bauer et al., 2015) perception. A common representation of a visual map in the SC is given by Ottes et al. (1986), where the retinotopic space is mapped to the collicular space using a logpolar transformation. That transformation has been suggested to lie at the core of complex mechanisms of visual attention (Taouali et al., 2015), including saccades (Manfredi et al., 2009).

1.2 Computational model

Computational neural models of the SC exist in various forms, both for multisensory integration (Bauer, 2015, chapter 3) and for saccade generation (Girard and Berthoz, 2005). One frequently used theoretical paradigm that encompasses both aspects, and

that has been predominant when it comes to visual processing in the SC, is that of dynamic neural fields (DNF) (Marino et al., 2012; Taouali et al., 2015; Quinton and Goffart, 2018). It originated as a mathematical model of neural dynamics (Amari, 1977), and has been used to model neural activity in sensorimotor maps at a mesoscopic scale (Schöner et al., 2015). DNF describe the evolution of mean field potential over a continuous domain (usually simply called a map), for instance the average membrane potential of neurons in the intermediate layers of the SC (Trappenberg et al., 2001; Wilimzig et al., 2006). While interactions at the microscopic scale may be of interest for many neural processes, focusing on neural fields at a mesoscopic scale helps to bridge the gap with behavioral data. This is not only useful to better understand adaptive functions found in living systems (Schöner et al., 2015), but also makes it possible to build artificial systems able to reproduce them (including decision-making and attentional capabilities based on noisy sensor data) and to implement them on robots (with topologies of sensors that differ from humans). Depending on their parametrization, DNF may for instance achieve selection or interpolation between several conflicting signals (Taouali et al., 2015), robust selective attention in presence of noise and distractors (Fix et al., 2011), working or long term memory of stimuli (Sandamirskaya, 2014).

DNF have long been used as models of visual attention (Fix et al., 2011) and (visuo)motor control (Wilimzig et al., 2006; Sandamirskaya, 2014; Quinton and Goffart, 2018). However, the literature is scarcer when it comes to using DNF for multimodal fusion (Schauer and Gross, 2004; Ménard and Frezza-Buet, 2005; Lefort et al., 2013). Schauer and Gross (2004) have shown promising results with a bio-inspired DNF-based model of audiovisual integration. With very little preprocessing, they achieved a sig-

nificant response enhancement when exposed to congruent visual and auditory signals, although they did not draw connections to known psychophysical phenomena.

1.3 Psychophysical reference

In this paper, we will show that applications of DNF go as far as to account for well known psychophysical effects of multisensory integration. As an illustration of such possibilities, we will use the ventriloquist effect (Alais and Burr, 2004), which is an example of audiovisual merging. From a human participant viewpoint exposed to spatially incongruent visual and audio stimuli, the position of a stimulus is shifted towards the other, depending on which modality has the highest relative precision. The effect takes its name from ventriloquist shows, where spectators have the illusion that a puppet is speaking, while the sound is actually produced by the ventriloquist holding it.

We will draw on psychophysical data reported in Alais and Burr (2004), because their experimental paradigm and protocol can easily be replicated *in silico*, they provide extensive results in all conditions, and their paper is a seminal contribution to the field, with results that have not yet been challenged. One might notice that in their experiment, only the visual precision varied. However, by manipulating the relative precision between the two modalities, they showed the multiple sides of the ventriloquist effect (either vision capturing audition, the reverse, or an interpolation between both). We want our computational model to exhibit the diversity of behaviors linked to multimodal fusion, so this experiment constitutes an interesting showcase.

In addition to empirical data, we will also compare the performance of our model to optimal Bayesian integration, usually considered as the golden standard among formal

and computational models of multisensory integration (Ernst and Bulthoff, 2004; Rohde et al., 2016). However, note that we do not strive for a perfect quantitative fit of our model to the data. Indeed, even though optimization and sensitivity analysis will be combined to assess the ability of our model to robustly converge with behavioral data, our model enables a broad set of perspectives by building on past DNF models, of which the ventriloquist effect is only one illustration.

The remainder of this article is structured as follows. In section 2, we describe our computational model and its evaluation criteria in the context of the ventriloquist effect. We present the results in section 3, and discuss further on the capabilities of our model in section 4.

2 Method

2.1 General model

From a neurophysiological standpoint, the (deep) SC has been reported to receive projections from different modalities on a series of multimodal neural maps (King, 2004). In this section, we first described how these maps are modeled, before turning to the projections they receive. An overview of our general model is given in figure 1.

2.1.1 Dynamic neural fields

Our model of a SC map activity is based on dynamic field theory (Schöner et al., 2015). DNF model the evolution of the neural activity over time on each point of a topological space \mathbf{X} that maps a portion of the brain. The mean field potential U at position $\mathbf{x} \in \mathbf{X}$

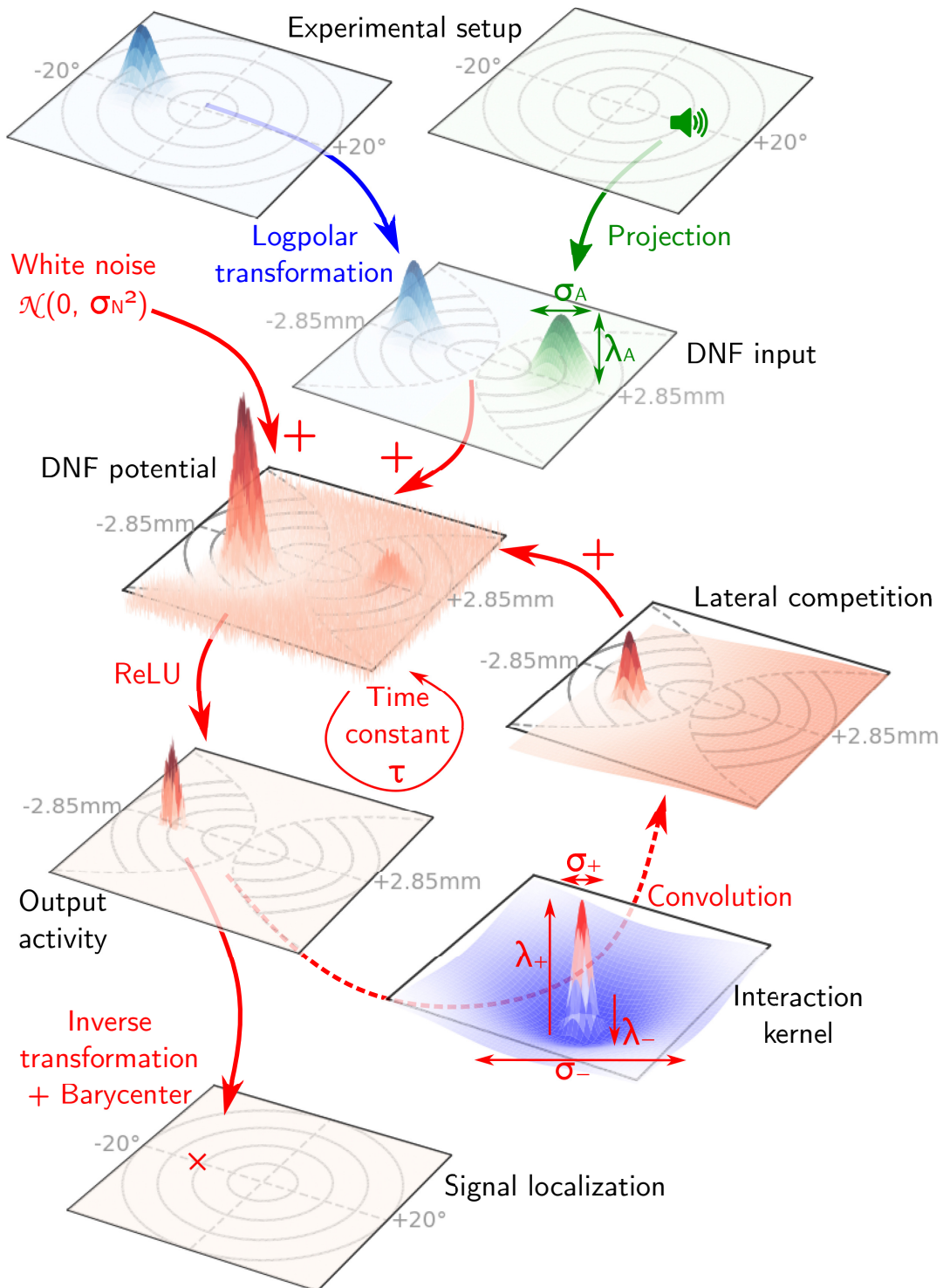


Figure 1: Visual representation of the audiovisual merging DNF model. Each rectangle represents a map, either in retinal space (shown with concentric circles) or SC (hourglass shape, obtained by performing a logpolar transformation on the visual map). The blue arrow and text relate to visual preprocessing, green to auditory. Steps and parameters from the model, other than preprocessing, are shown in red.

and time t is described by the following stochastic integro-differential equation:

$$\tau \frac{\partial U}{\partial t}(\mathbf{x}, t) = -U(\mathbf{x}, t) + I(\mathbf{x}, t) + \int_{\mathbf{x}' \in \mathbf{X}} W(\|\mathbf{x} - \mathbf{x}'\|) f(U(\mathbf{x}', t)) d\mathbf{x}' + \varepsilon \quad (1)$$

where τ is the time constant which determines the response timescale of the entire field, I is the input stimulation over the field and f is a non-linear activation function; as often chosen to simplify numerical simulations, we will use a ReLU function to approximate the mean firing rate of neurons (Quinton and Goffart, 2018). The last term ε represents noise which, like the entire dynamic neural fields, can be interpreted at either a neurological (a sum of numerous variations of activity induced by external neurons) or psychophysical level (e.g. perceptual noise) (Schöner et al., 2015, box 1.4, p. 36). Due to the variations being summed over a large population of neurons, white noise is often used, and ε is therefore sampled from a normal distribution $\mathcal{N}(0, \sigma_N)$.

Finally, the kernel approximating lateral interactions within the continuous population of neurons is defined by:

$$W(\Delta \mathbf{x}) = \lambda_+ \exp\left(-\frac{\Delta \mathbf{x}^2}{2\sigma_+^2}\right) - \lambda_- \exp\left(-\frac{\Delta \mathbf{x}^2}{2\sigma_-^2}\right) \quad (2)$$

with $\lambda_+ > \lambda_-$ and $\sigma_+ < \sigma_-$, thus giving rise to local excitation and more diffuse inhibition. In the case of visual attention models, with such constraints on parameters, and spatially coherent input stimulation reflecting the presence of localized objects within the visual field, the numerical simulation of the DNF equation will converge to a stereotypical peak of activity, filtering out noise (Fix et al., 2011; Quinton, 2010). In the case of overt attention, it is then possible to directly project the DNF activity to control eye movements (Quinton and Goffart, 2018), in agreement with visual fixations being correlated with a balance of activity in the SC (Gandhi and Katnani, 2011). In our

numerical simulations, we will simply estimate the stimulus position within the field as the barycenter of the field output $f(U)$ (Rougier, 2006).

The time course of field activity before convergence will not be the focus of this article, since we are mostly interested in the location of peaks after stabilization. Readers interested in activity evolution over time will find extensive insights in Schöner et al. (2015) and an illustration of SC dynamics simulation in (Taouali et al., 2015, figure 5).

2.1.2 Projections to the neural field

Empirical evidence supports that signals emanating from a common location in the environment, even through different modalities, will project to nearby locations in the SC (Wallace and Stein, 1996). At the same time, the structure of the SC can be linked back to retinotopic space (Ottens et al., 1986). Given these neurophysiological findings, we decompose the input I defined at each point of the DNF as the sum of a visual input I_V and an auditory input I_A . Although summing projections from different modalities introduces a strong assumption into the model, it is frequent in the literature (Sandamirskaya, 2014; Schöner et al., 2015).

The projection of visual stimuli from the retina to the SC has been modeled mathematically in the form of a logpolar transformation (Ottens et al., 1986). Formally, a visual signal at a position (u, v) in the retinotopic space will be mapped to the SC at a position $\mathbf{x} = (x, y)$ given by:

$$\begin{cases} x = B_x \log \left(\frac{\sqrt{(u+A)^2 + v^2}}{A} \right) \\ y = B_y \arctan \left(\frac{v}{u+A} \right) \end{cases} \quad (3)$$

A , B_x and B_y are constant parameters that originate from the literature (Ottens et al.,

1986). Their values are given in table 1.

As for the auditory inputs and to our knowledge, there is no mathematical formulation of their projection onto the SC. To avoid introducing additional model parameters or uninformed constraints, we thus simply aligned the audio stimuli to their spatially congruent visual counterparts, since we do not aim at modeling the learning of sensory maps in the current research work. As projections to the SC through complex neural pathways are usually quite distant from raw sensory stimulation, we generate population coded auditory inputs as gaussian blobs of amplitude λ_A and width (standard deviation) σ_A . While the gaussian blob associated to the auditory stimulation is directly projected without distortion to the SC neural map, a similar gaussian blob is generated for the visual stimulation yet transformed through equation (3) during its projection on the SC. Amplitude and width of the audio stimuli are added to the list of free parameters of the model, while visual amplitude is fixed (since redundant with λ_A) and visual width is driven by the experimental setup.

2.2 Application to the ventriloquist effect

Even with constraints imposed on projections to the DNF, the model of the SC presented in the previous section and recapped in figure 1 is designed to accomplish a variety of tasks related to audio-visual perception, attention or memory, building upon existing works on neural fields (Schauer and Gross, 2004; Sandamirskaya, 2014; Taouali et al., 2015). In order to validate its capabilities for multimodal fusion, we here apply and test this generic model using an experimental paradigm associated with the ventriloquist effect, this effect being largely documented, and human data available. We use the

seminal work by Alais and Burr (2004), using human performance as ground truth for the evaluation of audio-visual fusion in our model. In their article, they reported detailed psychophysical results aggregated over hundreds of trials per condition and participant, with psychometric functions estimated in both unimodal and bimodal blocks of trials. For the latter, they relied on a fully crossed experimental design, manipulating various fusion-relevant parameters of the stimuli. Among other things, this makes their study particularly fit to replication using their data as a ground truth for computer simulations.

2.2.1 Experimental data

For each bimodal trial, participants were exposed to a sequence of two presentations of audio-visual stimuli (conflicting and non-conflicting, in random order), and had to report which of them was perceived more leftward. In the non-conflict presentation, auditory information (1.5 ms sound click with position determined by the interaural time difference) and visual information (15 ms low-contrast Gaussian blob of controlled width, with standard deviation $\sigma_V \in \{2^\circ, 16^\circ, 32^\circ\}$) were perfectly aligned with each other, but their eccentricity relative to the center of the participant’s field of view was manipulated (from -20° to $+20^\circ$, as depicted on the horizontal axis of figure 1 of Alais and Burr, 2004). In the conflict presentation, stimuli were still aligned on the azimuthal axis, but an horizontal spatial discrepancy was introduced between the two, with the visual stimulus moving of $\Delta \in \{-5^\circ, -2.5^\circ, 0^\circ, 2.5^\circ, 5^\circ\}$ (from left to right) and the auditory stimulus moving of $-\Delta$ (horizontal positions in figure 2).

As a consequence, we aim at replicating the psychometric curves (proportion of conflict stimuli perceived rightward as a function of eccentricity of the non-conflict

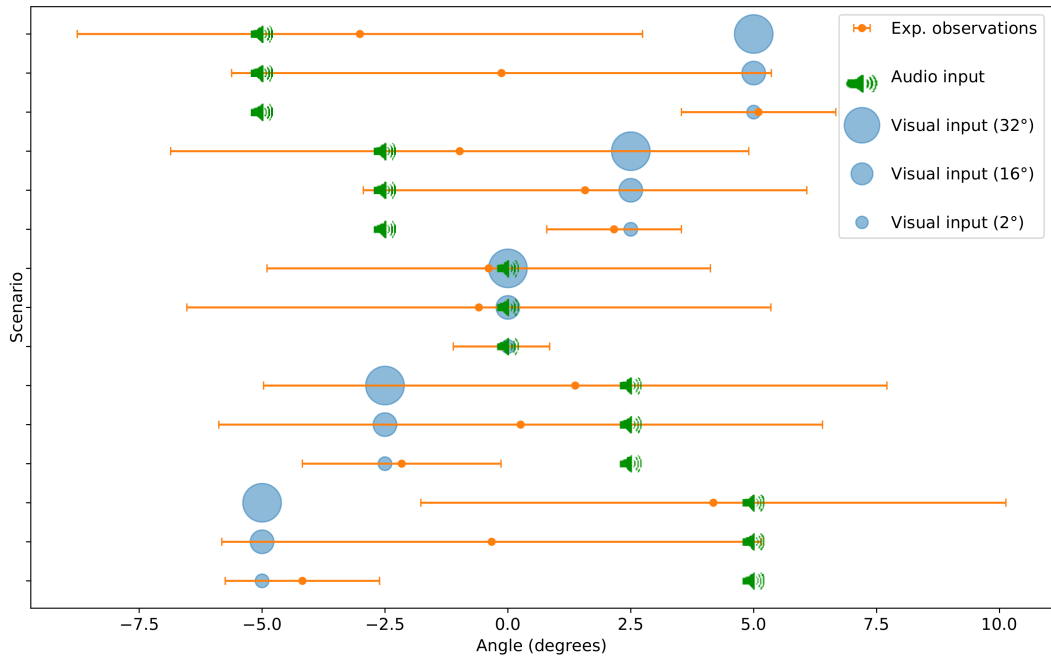


Figure 2: List of scenarios and experimental measures from Alais and Burr (2004). In each line: The green speaker symbol gives the position of the auditory stimulus in the conflicting presentation. The blue circle of growing size gives the position of the visual stimulus, of width $\sigma_V = 2^\circ, 16^\circ$ or 32° (not to scale). The measures of bimodal localization are represented by an orange error bar (mean \pm SD).

stimuli) obtained in the 15 scenarios of the original study (3 visual precisions \times 5 spatial distances). These psychometric curves were approximated by cumulative Gaussian functions (sigmoids with near-logistic shape; Bowling et al., 2009), thus reducing them to two parameters: median (also named point of subjective equality, equal to the mean for a Gaussian distribution) and standard deviation (accuracy). The Gaussian distributions associated to the unimodal and bimodal psychometric functions from Alais and Burr (2004) are reproduced on figure 2.

As a synthesis of their results, a thin visual stimulus ($\sigma_V = 2^\circ$) captures the location of the merged signal given its high accuracy. When it is very wide (32°), the auditory stimulus does. In-between (16°), the merging is located between both. In addition, the higher the precision of the inputs (e.g. 2° visual stimulus), the lower the standard deviation of the human localization distribution after fusion, reflecting that auditory and visual information were taken into account in a statistically optimal manner (Rohde et al., 2016).

2.2.2 Model constraints and simulation

For this specific operationalization of the ventriloquist effect, all presentations happen on a single azimuthal axis: $y = 0$. While the version of our DNF model presented in section 2.1.1 could be used as a suitable model of two-dimensional maps in the SC, it introduces parameters that are not directly supported by empirical data from the selected study, and would simply make optimization and interpretation more complex. Committing to the principle of parcimony, we have therefore chosen to restrict our model to a unidimensional projection of the SC, reducing the computational cost of the

simulations.

Whereas asking which stimuli were perceived as more leftward made sense experimentally to reduce task difficulty and prevent biases in responses, numerical simulations allow to directly estimate localization probability density functions. Yet given the noise and non-linearities from equation (1), we rely on the Monte Carlo method to sample the localization distribution under each condition through repeated simulation, and estimate summary statistics (mean and standard deviation of the empirical Gaussian distribution) for the conflict presentation alone. This means that the (static) inputs used in our model always consist of a bimodal signal, having a median location set at the fovea, and made of two unimodal components located opposite from each other. The non-conflict presentation is no longer necessary in this numerical setting. Since there is no generic analytical solution to this class of stochastic integro-differential equations, we rely on numerical resolution, which makes simulations computationally intensive and parameter estimation complex.

To correctly model the spatial distribution of stimuli used in the ventriloquist experiment, the simulated neural field covers angles from -20° to 20° in retinal space (which, after the transformation of equation (3), corresponds to ± 2.85 mm in SC) with a spatial resolution of 100 points ($\Delta x = 0.057$ mm). Similarly, to ensure a correct approximation of the temporal dynamics of the multimodal fusion and guarantee convergence to a stable localization, we solve equation (1) using the Euler scheme with a temporal resolution of 100 iterations per second ($\Delta t = 0.01$ s). All simulation constants are recapitulated in table 1. Algorithmically, the mean field potential (vector U) is initialized

to zero and updated by applying the following equation:

$$\begin{aligned}
\forall k \in K, U(k\Delta x, t + \Delta t) = & U(k\Delta x, t) \\
& + \frac{\Delta t}{\tau} \left(-U(k\Delta x, t) \right. \\
& \quad \left. + I(k\Delta x, t) \right. \\
& \quad \left. + \sum_{k' \in K} W(|k\Delta x - k'\Delta x|) f(U(k'\Delta x, t)) \right. \\
& \quad \left. + \varepsilon \right)
\end{aligned} \tag{4}$$

where $K = \{-50, -49, \dots, 50\} = \{\frac{-2.85}{\Delta x}, \frac{-2.85+\Delta x}{\Delta x}, \dots, \frac{+2.85}{\Delta x}\}$ and I can be decomposed according to section 2.1.2:

$$I(k\Delta x, t) = I_V(k\Delta x, t) + I_A(k\Delta x, t) \tag{5}$$

Table 1: Constant settings for all simulations. The values and descriptions of A , B_x and B_y are taken from Ottes et al. (1986). High spatial and temporal resolutions were chosen to prevent any qualitative impact on the results.

Constant	Value	Unit	Description
B_x	1.4	mm	x -axis scaling for the SC map
B_y	1.8	mm/°	y -axis scaling for the SC map
A	3	°	Shape of the mapping, relatively to $\frac{B_x}{B_y}$
Δt	0.01	s	Simulation time step
\mathbf{X}	[-2.85, 2.85]	mm	Spatial domain in SC
Δx	0.057	mm	Spatial discretization step

Given that we model a forced decision task (i.e. where human participants were asked to always answer even if they needed to guess), adequate parameters should always lead to the (quick) emergence of a stable activity pattern in presence of stimuli,

usually under the form of a stereotyped peak of activity on the neural field. An example of this output is given in figure 3, using artificial inputs and zero noise for the demonstration. We can see that, given two similar but conflicting stimuli, the DNF will in any case generate a prototypical peak of activity (an attractor in the dynamical system modelled by the set of differential equations), from which the barycenter can be used as the bimodal stimulus localization estimate, as developed at the end of section 2.1.1. The ensuing decision will either correspond to an interpolation between unimodal signals, or to the selection of the strongest one (barring random fluctuations not shown here). The choice between these two behaviors will depend on both the distance between the stimuli (as in this figure) and their relative precision (illustrated in the result section, with much lower stimuli precision).

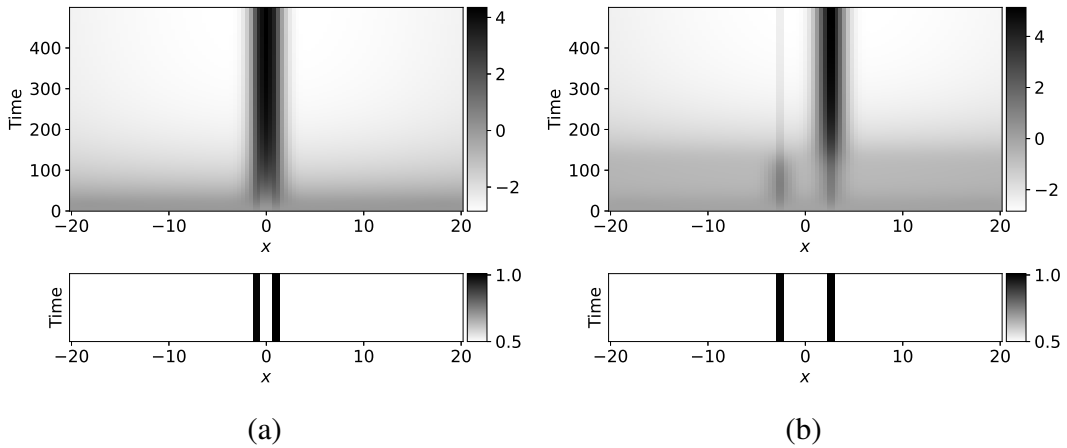


Figure 3: Evolution of DNF potential U on neural field (x) over time (top), using two different custom-made static inputs I (bottom). Parameters are taken from the “Selected” column of DNF+id in table 2, except noise is reduced to zero for explanatory purpose (on this figure only). To break the symmetry, in both subfigures, the right stimulus is 1% stronger than the left. Left and right subfigures differ by the distance between the stimuli.

2.3 Evaluation

While our task is not limited to a quantitative fit to empirical data, we will use the differences between model outputs and psychophysical results as a performance metric, which allows an indirect comparison of numerical models using human behavior as ground truth. As all (human and simulated) localization distributions roughly follow a Gaussian profile, performance will be computed based on estimated means and standard deviations on all scenarios from figure 2.

2.3.1 Compared models

The seminal experimental results on which we rely were already accompanied by a mathematical model (Alais and Burr, 2004). It is based on Bayesian modeling using maximum a posteriori estimates on localization distributions, which remains the dominant paradigm for multisensory integration (Rohde et al., 2016) to which we will compare. It explicitly relies on the hypothesis that the psychometric functions of visual and auditory stimuli are Gaussian cumulative distribution functions. The mean estimate and derived variance for their Bayes optimal combination are given by:

$$\hat{S}_{AV} = \frac{1/\sigma_V^2}{1/\sigma_V^2 + 1/\sigma_A^2} \hat{S}_V + \frac{1/\sigma_A^2}{1/\sigma_V^2 + 1/\sigma_A^2} \hat{S}_A \quad (6)$$

$$\sigma_{AV}^2 = \frac{\sigma_V^2 \sigma_A^2}{\sigma_V^2 + \sigma_A^2} \quad (7)$$

where \hat{S}_V and \hat{S}_A are the mean estimates of the visual and auditory signals positions respectively (assumed to coincide with the actual position of the sources), and σ_V^2 and σ_A^2 their variances (derived from the unimodal psychometric functions, as described in Rohde et al., 2016). The Bayesian model differs by design from ours, insofar that it uses

the unimodal performance to predict the bimodal behavior, whereas we fit our model directly on the bimodal scenarios, without prior knowledge of the unimodal variances.

In the case of our DNF model, for a given set of parameters allowing convergence to a stable localization decision through numerical resolution, each simulation should generate a single scalar output (between -20° and 20° after projecting back to the visual space). By replicating such simulations, the Monte Carlo method therefore produces an approximate localization distribution in each condition. As the 15 generated distributions (one per condition) are expected to be roughly Gaussian and were tested against extreme observations (to prevent biases in mean and standard deviation estimates due to statistical outliers), 50 simulations per condition were assessed as sufficient to extract accurate distribution parameters, and used as indices of model performance.

To test the usefulness of the logpolar transformation to correctly explain the experimental results for different eccentricities (confounded with varying degrees of audio-visual discrepancies), as well as to test the robustness of the DNF model to distortions in inputs projections, we will use two versions of our model: one where visual inputs go through a logpolar transformation following equation (3) (referenced as DNF+log in tables and figures); another where the transformation is replaced by an identity function (DNF+id), meaning $x = u$ and $y = v$. In the latter case, the DNF will operate directly on a visual map, i.e. $\mathbf{X} = [-20^\circ, 20^\circ]$, $\Delta x = 0.2^\circ$, and the auditory inputs need no realignment.

2.3.2 Model parametrization

Following previous definitions and constraints, our model has eight free parameters (see table 2): six from the DNF equation, and two from our modeling of auditory inputs in the SC as a Gaussian blob. This is true for both versions (DNF+log and DNF+id), since the logpolar transformation parameters are constant and derived from the literature. The behavior of a DNF depends mostly on the shape of its interaction kernel W . Therefore, fusion performance can mainly be correlated to the four parameters λ_+ , λ_- , σ_+ and σ_- . The dynamic and nonlinear nature of the DNF equation can make the dependencies very hard to comprehend, with strong interactions between parameters, especially when related to the kernel. Since we will also measure the variance of the model localization output, σ_N , which controls the amount of noise in the equation, will also play an important role; as well as τ , which controls the integration rate, and thus the weight of the noise compared to stimuli. Finally, while λ_A and σ_A do not intervene in the inner dynamics of the DNF, they can also be tweaked as part of the audio preprocessing of the model. They do have some interaction with the other parameters, as the shape of the interaction kernel determines which shape of input signals will be favored.

To ensure a fair comparison of models, free parameters had to be adjusted to the multimodal merging task. Within the high-dimensional parameter space, meta-heuristics that were already applied to the optimization of DNF parameters (such as Quinton, 2010) did not prove to be robust enough in the case of our multimodal fusion scenarios and evaluation procedure. Indeed, we could not easily combine into a single optimization criteria our two metrics: mean multimodal localization and localization variance. Trying to tackle this multicriteria optimization problem on stochastic integro-

differential equations also did not lead to acceptable Pareto-optimal sets of solutions.

Therefore, after a review of articles in the DNF literature, and extended preliminary simulations, we extracted for each parameter an interval in which suitable behavior was possible, and simply relied on an iterative and partial grid-search approach. Similarly to Jenkins et al. (2021), we started by picking some expertise-driven parameter values, then analyzed model performance as a function of one or two parameters at a time. Keeping the best values found, we iterated over sets of parameters until convergence. In a way similar to a simplex algorithm, we obtained the parameter values in column “Selected” of table 2. We have found that a change in σ_A was sufficient at first sight to compensate most of the distortion of visual inputs by the logpolar transformation. Consequently, it is possible to switch between DNF+log and DNF+id and obtain results of the same order of magnitude, by tweaking σ_A and leaving other parameters intact.

3 Results

Relying on the (locally) optimal parameters from table 2, this section first shows qualitative and illustrative behaviors of the DNF, before comparing performance between the different models described in section 2.3.1 (Bayesian, DNF+id, DNF+log), and then turning to a sensitivity analysis of the DNF model performance, studying the impact of pairs of parameters when keeping the others fixed. The objectives are to show that good performance from either DNF model versions cannot be attributed to over-parametrization (and thus overfit to the experimental data), and to study the effect of parameters on the DNF behavior.

Table 2: Model parameters. When one is fixed, its value is given in the “Selected” column. When one varies, either for exploration or visualization, it takes its values in the specified interval, discretized uniformly into 20 values. For DNF+log, values in italics have to be rescaled by a factor $\frac{2.85}{20}$ to accommodate for the change in field size from $[-20, 20]$ degrees to $[-2.85, 2.85]$ millimeters: while the transformation in the model is not linear, we use this field-wide rescaling to express all width and SD values in the same unit, opting for degrees. After the input is transformed, the DNF always operates on a regular space. σ_A has two different values for DNF+id and DNF+log respectively.

Parameter	Min.	Max.	Selected	Description
τ	0.05	0.5	0.15	Time constant
λ_+	0.1	1	0.425	Amplitude of lateral excitation
λ_-	0.05	0.2	0.15	Amplitude of lateral inhibition
σ_+	<i>0.2</i>	<i>2</i>	<i>0.85</i>	Width of lateral excitation
σ_-	<i>2</i>	<i>100</i>	<i>40</i>	Width of lateral inhibition
σ_N	0.5	5	2.8	Standard deviation of noise distribution
λ_A	0.1	2	1.1	Amplitude of auditory input
σ_A	<i>2</i>	<i>64</i>	<i>20 26</i>	Standard deviation of auditory input

3.1 Evolution of field potential

As a way to showcase the behaviors of our models, we start by observing their dynamics in realistic experimental conditions, complementing the illustration of qualitative differences in DNF outputs based on stimuli distance in section 2.2.2. For this subsec-

tion, we will make tests using the DNF+id model, as its output can be directly read and easily interpreted in the topological space of the source stimuli. We use the parameters from the “Selected” column of table 2. The inputs in the second experimental scenario ($\Delta = -5^\circ$, $\sigma_V = 16^\circ$) and related model activity are given in figure 4.

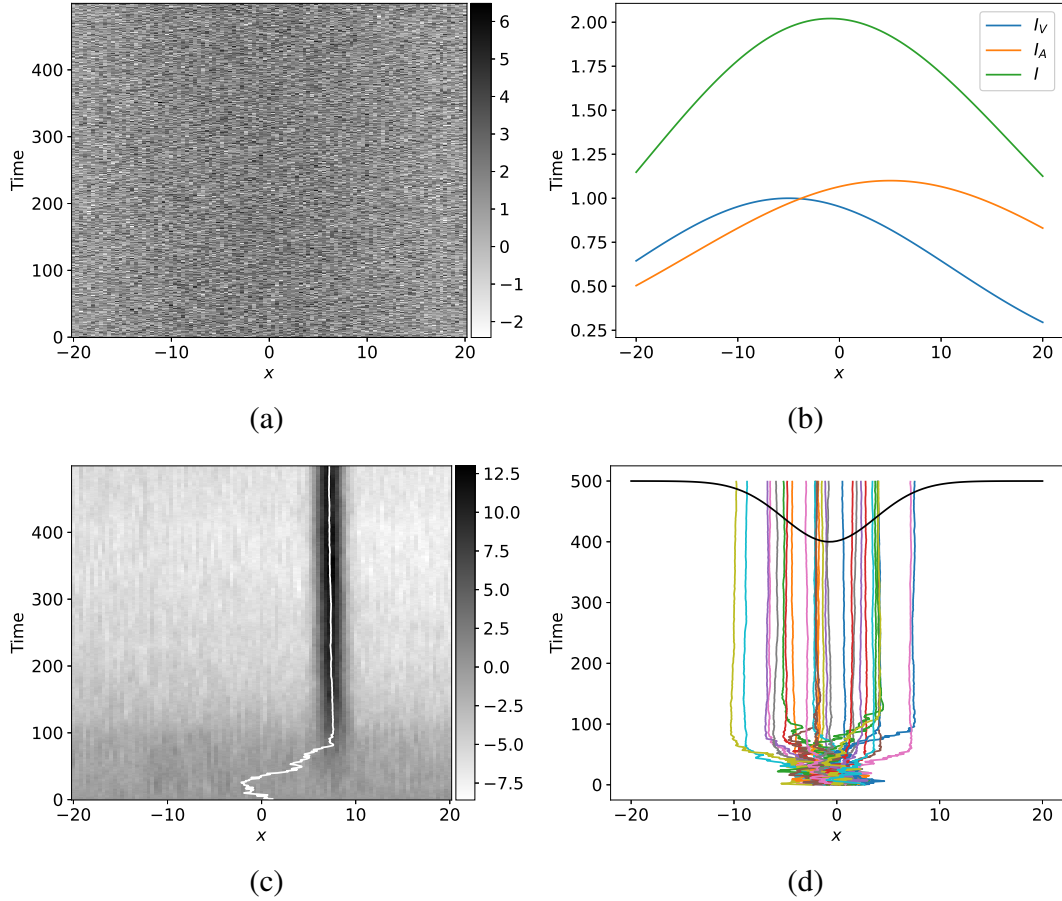


Figure 4: Evolution of DNF+id activity having $\Delta = -5^\circ$ and $\sigma_V = 16^\circ$. (a) Inputs summed with noise on neural field (x) over time. (b) Theoretical distribution of inputs in absence of noise. (c) Field potential U during one single run. The white line shows the evolution of the barycenter of field output $f(U)$. (d) Barycenters of DNF output for 30 other runs of the model. The black line shows the approximate Gaussian distribution obtained with the mean and SD of the final 30 positions.

As can be seen in subfigure (a), the amount of noise in the simulated data makes it almost impossible to distinguish the raw stimuli (b) with the naked eye. The evolution of DNF potential U is shown for one run of the model in subfigure (c). A peak forms at a seemingly random position, which is actually biased by the position of the stimuli. The underlying distribution of selected multimodal locations becomes apparent when the model is run multiple times (d). Some decisions do happen quite far from the source, which is consistent with stereotypical psychophysical studies, in which participants sometimes realize extreme guesses. But the distribution of selected multimodal locations shows that on average, decisions are made in between the two stimuli. The mean and variance of this DNF output distribution are the summary statistics used for model evaluation.

3.2 Model evaluation

Given the aforementioned models, we simulated the experimental scenarios to compare with the psychophysical data. The results are summarized on figure 5. As a reminder, we observe two metrics: the mean localization of a bimodal presentation (center of the intervals on figure 5) and its standard deviation (half-amplitude of the intervals). To mitigate the influence of extreme observations due to the stochasticity of the model, and thus provide accurate estimates, results presented in this section have been aggregated over 2500 runs instead of 50.

The quality of fit varies between scenarios. For example, DNF-based models achieve better fits in scenarios 6, 14 and 15, while the Bayesian model fares better in scenarios 3, 11 and 13. The distances between model and experimental outputs are summarized

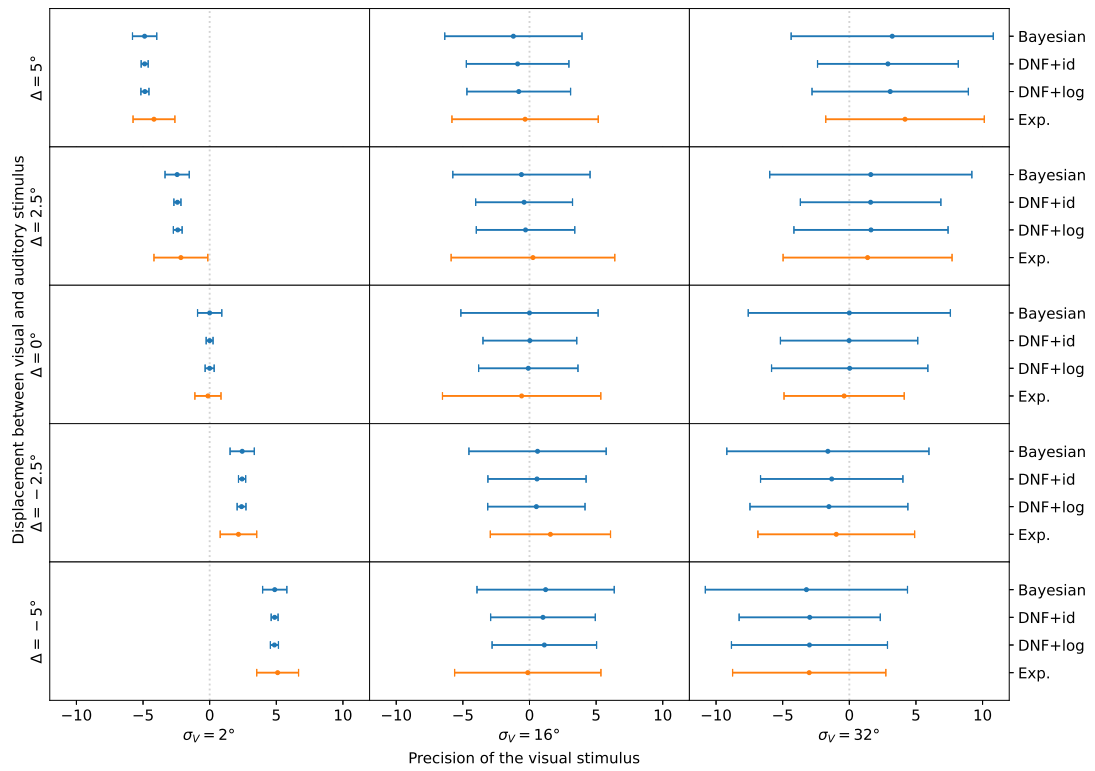


Figure 5: Experimental results of bimodal presentation (orange intervals, same as figure 2) and corresponding model outputs (in blue). For each error bar, the center dot represents the average localization, and the half-amplitude is the standard deviation.

in table 3. This shows a slight superiority of DNF+log over DNF+id, and a slight advantage of the Bayesian model when it comes to representing the localization variance only.

Table 3: Comparison between our model with logpolar transformation (DNF+log), without logpolar transformation (DNF+id), and the reference Bayesian model, using root mean square error between simulated and experimental data over the 15 scenarios.

	Error between means	Error between SD
DNF+log	0.626	1.33
DNF+id	0.638	1.38
Bayesian	0.677	1.28

Meanwhile, DNF come with the ability to model complex dynamical behaviors and are closer to known neurobiological mechanisms. So it is worth noting that our model enables a versatile point of view of multisensory integration, for a quantitative fit similar to the classical model. In particular, our model can simulate observations on a smaller scale (one run is one human decision) than Bayesian models (mostly focusing on the global distribution of the results). Our model can simulate all random variations between observations, while staying faithful to important mechanisms of multisensory integration.

3.3 Parameter exploration

Our model already shows quantitative results comparable to the most standard modeling paradigm, but there are other useful properties that can be displayed. In this section,

we will verify that performance is indeed consequent to our design choices, and not of overfitting. We will also show that there is still room for finetuning if one were to target some more specific criteria (such as a maximal fit of localization variance).

In order to emphasize parameter interactions in the most readable way, we have chosen to display the effects of two parameters at a time. In figures 6 to 8, six parameters keep the selected values mentioned in section 2.3.2, and two vary on a regular grid within the bounds given by table 2. We will only consider the DNF+log model from now on, our original and most complete version (even though similar analyses could be obtained with DNF+id).

We have found that depending on parameters, model behavior could fall into one of the following four categories. Only the first one is relevant to our simulation, the others will be masked in following figures.

1. For all scenarios, one single peak of activity emerges and stabilizes (often called a “bubble” in DNF literature). The rest of the field is inhibited thanks to lateral inhibition.
2. One bubble emerges but does not stabilize. The maximum potential increases indefinitely because of self-excitation. This is clearly implausible on a neural level.
3. No bubble emerges by lack of interaction, i.e. the term factored by W in equation (4) is negligible compared to the others. So the potential U will converge to an approximation of I . Two peaks will be observable when the stimuli are spatially discrepant, but they do not correspond to a bubble enhanced by self-

excitation. The outcome is that the decision-making role of DNF goes missing, which falls far away from our objectives.

4. In scenarios where stimuli are far apart, two distinct bubbles emerge. This happens when there is not enough long-range inhibition for one bubble to take over the other. Psychophysically, that would account for an observer explicitly noticing that there are two distinct stimuli. Alais and Burr (2004) do not report this happening in their experiments.

3.3.1 Pairwise variations

Our first step is to make all 8 parameters of our model vary by pairs. The results are compiled in two triangular matrices (one for each error measure) in figure 6 (means bottom left, SD top right), of which each element contains a 2D regular grid. The bounds of each parameter are listed in table 2.

First, we can see that τ and σ_N have a strong effect on the localization standard deviation, and a slight effect on the mean localization. In general, increasing σ_N or decreasing τ would give moderately less reliable localization means, but more plausible standard deviations. This is coherent with our simulation paradigm: increasing σ_N means adding more noise, and decreasing τ means a quicker integration of new information through time, both increasing the weight of the noise relatively to the stable audio and visual stimuli. We can also see that the mean localization is not completely smooth, and even less so for higher σ_N or lower τ . As a reminder, our results are by default aggregated over 50 runs for each parameter combination, for the purposes of smoothing the graphics. Fluctuations caused by extreme values are still expected, so it

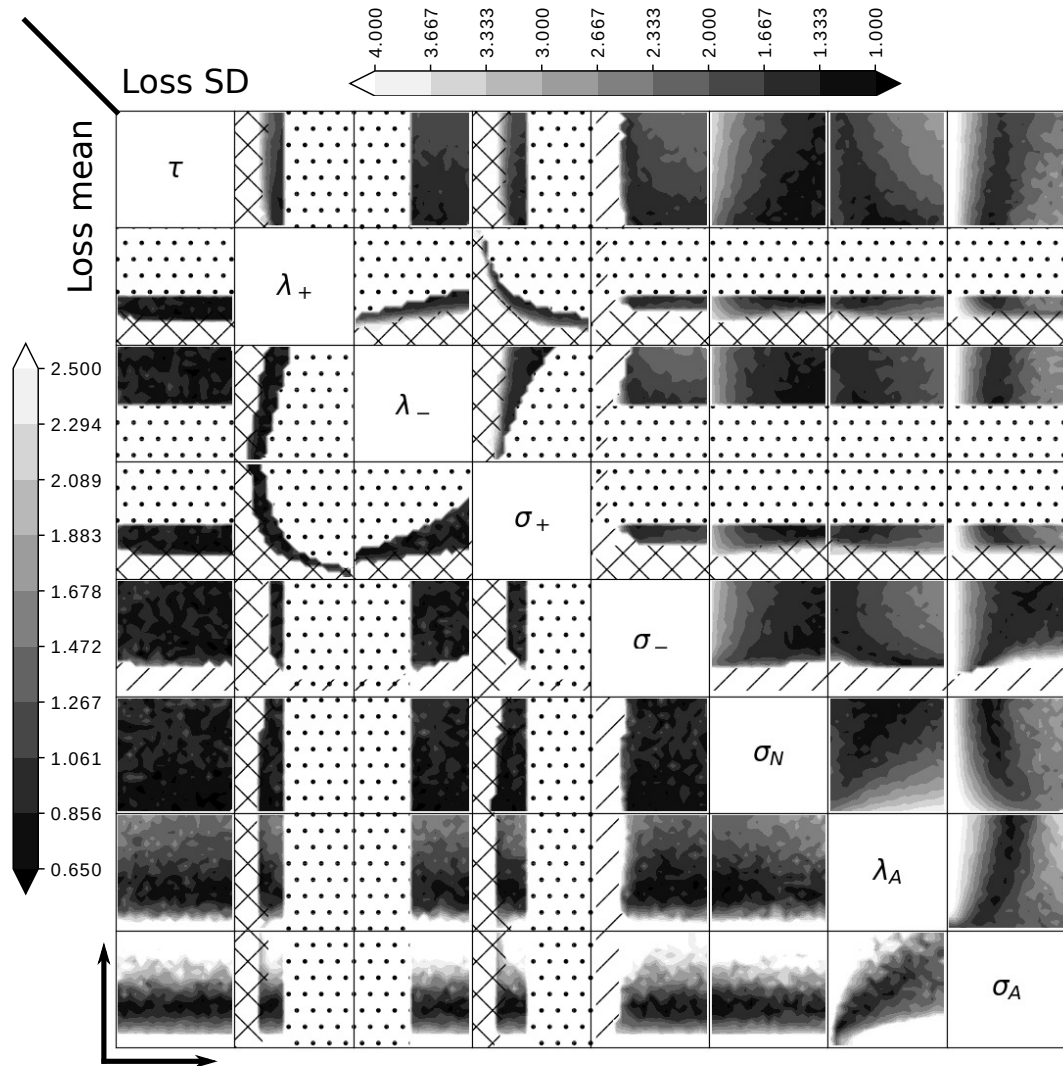


Figure 6: RMSE obtained by the DNF+log model depending on pairs of parameters. The bottom left triangular matrix is based on errors in mean localization of bimodal presentations, the top right one on their standard deviations. For each entry, the parameter labeled in row increases from bottom to top, and the parameter labeled in column increases from left to right. The blank areas filled with geometrical shapes designate parameter sets that fall out of scope of our simulation plan (cf. section 3.3). Dotted: no convergence, or overflowing activity (case 2). Hatched: more than one peak (cases 3 and 4). Crossed: no interaction (case 3).

is consistent that they become more apparent when the amount of noise in the system is increased.

There is some predictable interaction between λ_A and σ_A . The graphs outline a parabola-shaped ridge, along which these parameters can evolve with little impact on the results. It is worth noting that an increase of σ_A can be compensated by an increase of λ_A . That is a characteristic of the DNF. The model is designed to select in priority stimuli whose profile match the positive part of the interaction kernel, which is very thin in the case of the selected parameters ($\sigma_+ = 0.85^\circ$, or 0.12 mm after rescaling). When σ_A augments, the auditory stimulus strays further away from the thin template, and loses weight in the DNF integration. This loss of importance can be artificially compensated by an increase of λ_A .

Interaction kernel parameters λ_+ , λ_- , σ_+ and σ_- have clear bounds. In a DNF, when a peak forms due to self-excitation, a minimum amount of inhibition is necessary for the system to stabilize. Too much excitation or too little inhibition will cause the peak to increase in amplitude indefinitely, which does not fit plausibly to any neural mechanism. On the contrary, too little excitation and no peak will form, no interaction will happen and the model will simply replicate its inputs as outputs. This is out-of-scope because it is impossible to generate a saccade or focus for fine-grained processing two stimuli that lie in different locations of the visual field. It is worth noting that λ_+ has an impact on the thresholds for λ_- and σ_+ , and vice versa. That means that any of these parameters can be tweaked largely, as long as some ratios of excitation or inhibition are maintained. Interestingly enough, σ_- is less affected by the other three. The main use of this parameter is to ensure the presence of long-range inhibition, so it primarily

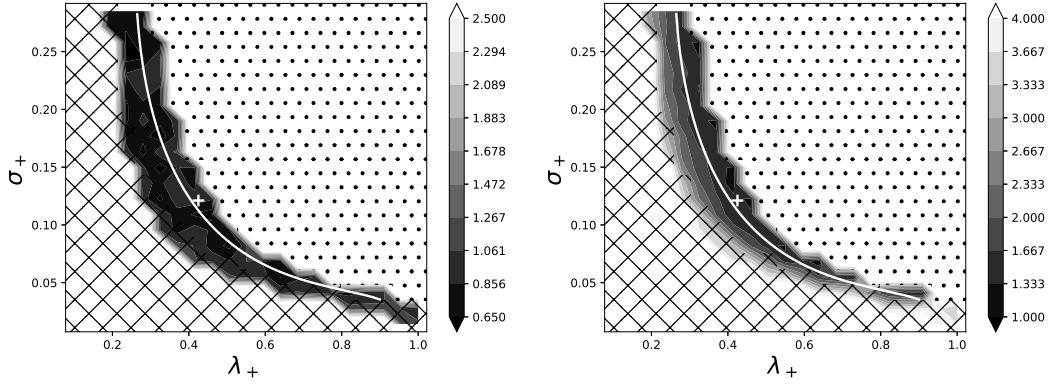


Figure 7: RMSE obtained by the DNF+log model depending on λ_+ and σ_+ (expanded from figure 6). The left graph is based on mean localization of bimodal presentations, the right graph on their standard deviations. The white cross indicates the default values used in the previous section. The white line shows the parametric curve that will be used for parameter reduction. The blank areas filled with geometrical shapes designate parameter sets that fall out of scope of our simulation plan. Dotted: no convergence, or overflowing activity. Crossed: no interaction (U replicates I).

needs to be sufficiently high. That is consistent with alternative implementations of DNF in the literature, where local inhibition in W is replaced by a constant global inhibition parameter, in situations where only one stimulus should be selected in the entire field (Schöner et al., 2015; Taouali et al., 2015). This can be seen as a reduction of equation (2) with σ_- tending to infinity. Our model does not make this restriction: while a multi-selection is irrelevant in our application to the ventriloquist effect, we did not make the assumption of a unique selection in the entire SC.

3.3.2 Reducing the dimensionality of the parameter space

Some regular grids present ridges along which the two parameters vary while the model error stays approximately constant. This is particularly clear for the pair (λ_+, σ_+) , allowing us to define a parametric curve on the optimal performance ridge which covers the whole range of parameter values. This curve is defined as a function of an abstract parameter p_+ , with the grids and curves for the localization mean and standard deviation reproduced on figure 7. The use of p_+ allows us to check for interaction with other parameters, with one less dimension, and to cancel the effect of the local excitation parameters on the model error. The new grids made with p_+ are given in figure 8.

We can see that there are no interaction effects left, including between p_+ and λ_- . This confirms that the model behavior remains approximately invariant to its excitation parameters as long as a certain ratio is kept. Consequently, the number of parameters in our model could be decreased: for each value of σ_+ within a certain range, there is a value of λ_+ that achieves a similar fit.

The representation of figure 8 also makes clear the tolerable range of certain parameters, and the latitude in their tuning. Inhibition parameters have to exceed a certain threshold ($\lambda_- > 0.11$, $\sigma_- > 5^\circ$), otherwise the self-excitation of the DNF will not be compensated, and the membrane potential U will increase endlessly. In addition, σ_- must be high enough (above approximately 30°) to ensure that only one peak is selected. We can see that a better fit in localization standard deviation can be attained by either decreasing τ or increasing σ_N , but at the detriment of the fit in mean localization. Similarly, λ_A and σ_A show vertical strips where the fit is maximal, but these strips do not coincide between both error measures. Given our goal of reproducing in general

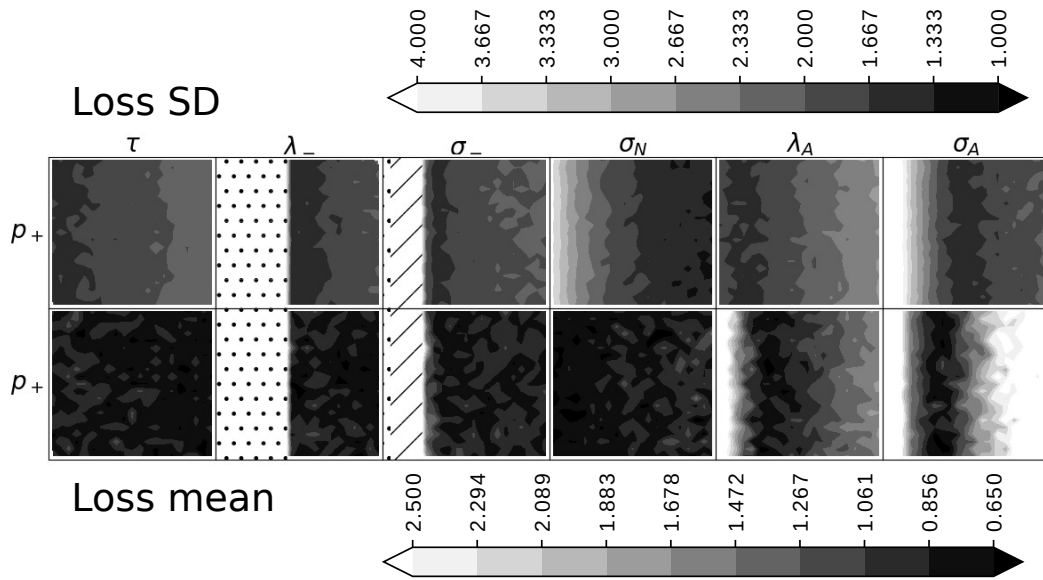


Figure 8: RMSE obtained by the DNF+log model depending on p_+ (from the parametric curve of figure 7) and other parameters. The bottom row is based on mean localization of bimodal presentations, the top row on their standard deviations. In each entry, the parameter labeled on the top increases from left to right. The bottom of a square corresponds to a low λ_+ and high σ_+ , the top corresponds to a high λ_+ and low σ_+ . See figure 6 for the rest of the legend.

aspects a psychophysical experiment, we have had to settle for a good quantitative fit in both criteria. But as we can see, if our objective was to fit either the mean localization or its standard deviation, performance could be increased substantially. There are no sharp ridges or spikes, and the local optima (see darkened areas on figure 8) are quite wide, so the parameter fitting would be relatively smooth, and the results we obtained in table 3 do not rely exclusively on finetuned values of many parameters.

In summary, there are several ways the number of parameters can be decreased. We have seen earlier that changes in λ_A and σ_A can compensate each other, so λ_A

could be fixed arbitrarily, and some finetuning would be feasible with σ_A alone. σ_A determines, together with the kernel parameters, the relative weight each stimulus will have in the DNF. For an estimation of the mean localization of the bimodal signal, if we assume that λ_- and σ_- always remain above a necessary threshold, and that λ_+ and σ_+ are restricted to the parametric curve in figure 7, then we are left with only two free parameters: p_+ and σ_A . Remaining parameters intervene in the dynamic capabilities of our model (e.g. to predict response times) and its ability to explain some of the inter-observational variations.

4 Conclusion

Models of multimodal merging in psychophysics come predominantly from the Bayesian paradigm. We have shown, using the ventriloquist effect as an illustrative example, that it is possible to model such a task using a neurally-inspired, population-based dynamical system. The model we created conciliates known characteristics of the superior colliculus and the paradigm of dynamic field theory, reaching a quantitative fit comparable to the classical paradigm. The difference between the two models has to be examined at a more theoretical level, given that they operate at different levels of abstraction. DNF are meant to model neural dynamics (Amari, 1977). While they do not constitute an exact simulation of neurons at a microscopic level, the behaviors that emerge from the dynamic system echo physically observable neural patterns at a larger scale, aggregating over thousands of neurons. Bayesian models of multimodal fusion, on the contrary, were not derived to accurately relate to biological mechanisms (although fine-grained

Bayesian models may be perfectly fit to model such mechanisms), but rather to estimate subjects' decision distributions at coarser spatiotemporal scales. Using the terminology from Marr (1982), the Bayesian model operates at the level of the computational theory, in that it describes the logic by which information coming from different sensory modalities will be integrated, without delving into the ways the inputs are represented or the algorithm is implemented. DNF models could be placed in the other two levels: either representation-algorithm, when the way inputs are transformed into a decision is described through mathematical equations; or hardware implementation, when we consider the discretized field where each neuron acts as a processing unit. Note that these levels are not mutually exclusive, and previous works have hinted at perspectives to analyze either Bayesian modeling (Ma et al., 2006) or DNF (Gepperth and Lefort, 2016) at the level of the other. In any case, this different positioning does not preclude the ability of any of these paradigms to generalize to a wide range of tasks and mechanisms. Both make sense at their own level, although it can be argued that Bayesian modeling might be too broad to capture some of the most subtle behaviors that may emerge from neural interaction (Jenkins et al., 2021). That additional precision of DNF comes at the cost of an extended parameter space.

It is worth noting that our choice of parameters is not detrimentally constraining. There is some latitude in the parameter tuning, thus our modeling hypotheses do not particularly weaken the value of our results. In particular, there is flexibility in the shape of auditory inputs (the model does not rely on one specific pair of values (λ_A, σ_A)), and quantitative fit did not discriminate against the use of the logpolar transformation.

The relative freedom in model optimization opens up new simulation perspectives.

First, there is room for additional parameters and tuning, not included in our current simulations as a first parsimonious approximation. For instance, in our model, as in many previous DNF models (Wilimzig et al., 2006; Fix et al., 2011), white noise is used while not spatially correlated. One could expect that spatially correlated noise (as used in Taouali et al., 2015; Jenkins et al., 2021) would help fit the variance better, especially in scenarios involving a very thin visual stimulus. Then, we have seen that the parameter dimensionality could be reduced (for example by removing σ_- and using global inhibition), and that some pairs of parameters could compensate one another in an optimization task (most notably, λ_+ and σ_+ , τ and σ_N , λ_A and σ_A). Consequently, we have reason to believe that our model can be used to fit more demanding tasks. A hypothetical situation would be to simulate a bimodal perception task and fit both the signal localization and an observer's response time. One could then consider locking pairs of parameters on parametric curves (as we did with λ_+ and σ_+) for localization fitting, and use the newly freed dimensions (such as p_+) to fit for the additional constraints.

Indeed, our model has room for the integration of additional functionalities, and the first novelty brought by DNF stands in its dynamic properties. DNF are fully capable of integrating any kind of time-dependant signals (so long as they can be projected onto a topological map). Moreover, their inner dynamics may account for behavioral responses of a human during the perception process. For instance, the peaks of activity in the DNF can generate population-coded motor commands for visual saccades (Wilimzig et al., 2006; Quinton and Goffart, 2018). While the experimental data we have used did not highlight any particular time-related merging effect, our model incorpo-

rates by design the groundwork for the modeling of new dynamic properties.

Additionally, we have seen that DNF are suitable when perceptive fields are not homogeneous across the map, as was showcased by the logpolar transformation. In that particular case, the expectation is that a visual stimulus that appears further away from the fovea will have an increased precedence in the audiovisual fusion. Indeed, in the periphery of the retina, the logpolar transformation will activate a smaller region of the multisensory map, and in our case the DNF matches thinner signals better. This situation is out of scope in the classical ventriloquist experiment, which centers on the fovea, with little eccentricity. This limitation in the experimental data may explain the lack of difference we found between DNF+id and DNF+log. But our simulation would still provide an interesting baseline for the modeling of eccentric audiovisual merging, especially with regards to saccade generation. A visual signal in the border of the field of view will be a likely target for a saccade, although (or, according to many models of saccade generation, because) it is seen less precisely. At the psychophysical level, how much this interferes with the general paradigm of multisensory integration (for which a less precise visual stimulus would actually be captured more easily by other modalities) is still an open question. However, on a computational level, our model reunites some of the keys to a common ground between multimodal fusion and active perception.

Acknowledgments

This work forms part of the project AMPLIFIER. It was funded by the French region Auvergne-Rhône-Alpes in the context of the “Pack Ambition Recherche” initiative.

Preparations to this work have been partially funded by the PERSYVAL-Lab LabEx (ANR-11-LABX-0025-01) under the French program *Investissement d'avenir*, as well as the ANR GAG and CNRS project APF².

Most the computations presented in this paper were performed using the GRICAD infrastructure (<https://gricad.univ-grenoble-alpes.fr>), which is supported by Grenoble research communities.

References

- Alais, D. and Burr, D. (2004). The ventriloquist effect results from near-optimal bimodal integration. *Current Biology*, 14(3):257–262.
- Alais, D., Newell, F. N., and Mamassian, P. (2010). Multisensory processing in review: from physiology to behaviour. *Seeing Perceiving*, 23(1):3–38.
- Amari, S.-I. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 27(2):77–87.
- Bauer, J. (2015). *One Computer Scientist's (Deep) Superior Colliculus: Modeling, understanding, and learning from a multisensory midbrain structure*. PhD thesis, University of Hamburg.
- Bauer, J., Magg, S., and Wermter, S. (2015). Attention modeled as information in learning multisensory integration. *Neural Networks*, 65:44–52.
- Bowling, S. R., Khasawneh, M. T., Kaewkuekool, S., and Cho, B. R. (2009). A logistic

- approximation to the cumulative normal distribution. *Journal of Industrial Engineering and Management*, 2(1):114–127.
- Calvert, G., Spence, C., and Stein, B. (2004). *The Handbook of Multisensory Processes*. A Bradford book. MIT Press.
- Casey, M. C., Pavlou, A., and Timotheou, A. (2012). Audio-visual localization with hierarchical topographic maps: Modeling the superior colliculus. *Neurocomputing*, 97:344–356.
- Driver, J. and Spence, C. (2004). Crossmodal spatial attention: Evidence from human performance. In Spence, C. and Driver, J., editors, *Crossmodal Space and Crossmodal Attention*. Oxford University Press.
- Ernst, M. O. and Banks, M. S. (2002). Humans integrate visual and haptic information in a statistically optimal fashion. *Nature*, 415:429–433.
- Ernst, M. O. and Bulthoff, H. (2004). Merging the senses into a robust percept. *Trends in Cognitive Sciences*, 8(4):162–169.
- Fix, J., Rougier, N., and Alexandre, F. (2011). A dynamic neural field approach to the covert and overt deployment of spatial attention. *Cognitive Computation*, 3(1):279–293.
- Frens, M. A., Van Opstal, A. J., and Van der Willigen, R. F. (1995). Spatial and temporal factors determine auditory-visual interactions in human saccadic eye movements. *Perception & Psychophysics*, 57(6):802–16.

- Gandhi, N. J. and Katnani, H. A. (2011). Motor functions of the superior colliculus. *Annual Review of Neuroscience*, 34(1):205–231.
- Gepperth, A. and Lefort, M. (2016). Learning to be attractive: Probabilistic computation with dynamic attractor networks. In *2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 270–277.
- Girard, B. and Berthoz, A. (2005). From brainstem to cortex: Computational models of saccade generation circuitry. *Progress in Neurobiology*, 77(4):215–251.
- Jenkins, G. W., Samuelson, L. K., Penny, W., and Spencer, J. P. (2021). Learning words in space and time: Contrasting models of the suspicious coincidence effect. *Cognition*, 210:104576.
- Kapoula, Z. and Pain, E. (2020). Differential impact of sound on saccades vergence and combine eye movements: A multiple case study. *Journal of Clinical Studies & Medical Case Reports*, 7:095.
- King, A. J. (2004). The superior colliculus. *Current Biology*, 14(9):335–338.
- Lefort, M., Boniface, Y., and Girau, B. (2013). SOMMA: Cortically inspired paradigms for multimodal processing. *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.
- Ma, W. J., Beck, J. M., Latham, P. E., and Pouget, A. (2006). Bayesian inference with probabilistic population codes. *Nature neuroscience*, 9(11):1432–1438.

- Manfredi, L., Maini, E. S., and Laschi, C. (2009). Neurophysiological models of gaze control in humanoid robotics. In Choi, B., editor, *Humanoid Robots*, chapter 10. IntechOpen, Rijeka.
- Marino, R. A., Trappenberg, T. P., Dorris, M., and Munoz, D. P. (2012). Spatial interactions in the superior colliculus predict saccade behavior in a neural field model. *J. Cognitive Neuroscience*, 24(2):315–336.
- Marr, D. (1982). *Vision. A Computational Investigation Into the Human Representation and Processing of Visual Information*. MIT Press.
- McGurk, H. and MacDonald, J. (1976). Hearing lips and seeing voices. *Nature*, 264(5588):746–748.
- Meredith, M. A. and Stein, B. E. (1986). Visual, auditory, and somatosensory convergence on cells in superior colliculus results in multisensory integration. *Journal of Neurophysiology*, 56(3):640–662.
- Ménard, O. and Frezza-Buet, H. (2005). Model of multi-modal cortical processing: Coherent learning in self-organizing modules. *Neural Networks*, 18(5):646–655. IJCNN 2005.
- Newell, F. N., Ernst, M. O., Tjan, B. S., and Bühlhoff, H. H. (2001). Viewpoint dependence in visual and haptic object recognition. *Psychological Science*, 12(1):37–42. PMID: 11294226.
- Ottes, F. P., Gisbergen, J. A. V., and Eggermont, J. J. (1986). Visuomotor fields of the superior colliculus: A quantitative model. *Vision Research*, 26(6):857–873.

- Quinton, J.-C. (2010). Exploring and optimizing dynamic neural fields parameters using genetic algorithms. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7.
- Quinton, J.-C. and Goffart, L. (2018). A unified dynamic neural field model of goal directed eye movements. *Connection Science*, 30(1):20–52.
- Rohde, M., van Dam, L. C., and Ernst, M. O. (2016). Statistically optimal multisensory cue integration: A practical tutorial. *Multisensory Research*, 29(4-5):279–317.
- Rougier, N. P. (2006). Dynamic neural field with local inhibition. *Biological Cybernetics*, 94(3):169–179.
- Sandamirskaya, Y. (2014). Dynamic neural fields as a step toward cognitive neuromorphic architectures. *Frontiers in Neuroscience*, 7:276.
- Schauer, C. and Gross, H. M. (2004). Design and optimization of Amari neural fields for early auditory-visual integration. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, volume 4, pages 2523–2528.
- Schöner, G., Spencer, J., and DFT Research Group (2015). *Dynamic Thinking: A Primer on Dynamic Field Theory*. Oxford Series in Developmental Cognitive Neuroscience. Oxford University Press.
- Taouali, W., Goffart, L., Alexandre, F., and Rougier, N. P. (2015). A parsimonious computational model of visual target position encoding in the superior colliculus. *Biological Cybernetics*, 109(4):549–559.

- Trappenberg, T. P., Munoz, D. P., and Klein, R. M. (2001). A model of saccade initiation based on the competitive integration of exogenous and endogenous signals in the superior colliculus. *Journal of Cognitive Neuroscience*, 13(2):256–271.
- Wallace, M. T. and Stein, B. E. (1996). Chapter 21: Sensory organization of the superior colliculus in cat and monkey. In Norita, M., Bando, T., and Stein, B. E., editors, *Extrageniculostriate Mechanisms Underlying Visually-Guided Orientation Behavior*, volume 112 of *Progress in Brain Research*, pages 301–311. Elsevier.
- Wilimzig, C., Schneider, S., and Schöner, G. (2006). The time course of saccadic decision making: Dynamic field theory. *Neural Networks*, 19(8):1059–1074. *Neurobiology of Decision Making*.
- Witten, I. B. and Knudsen, E. I. (2005). Why seeing is believing: merging auditory and visual worlds. *Neuron*, 48(3):489–496.

Combining Manifold Learning and Neural Field Dynamics for Multimodal Fusion

Simon Forest^{*†}, Jean-Charles Quinton^{*}, Mathieu Lefort[†]

^{*}Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, UMR 5224, F-38000, Grenoble, France

[†]Univ. Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR 5205, F-69622, Villeurbanne, France

{simon.forest, quintonj}@univ-grenoble-alpes.fr, mathieu.lefort@univ-lyon1.fr

Abstract—For interactivity and cost-efficiency purposes, both biological and artificial agents (e.g., robots) usually rely on sets of complementary sensors. Each sensor samples information from only a subset of the environment, with both the subset and the precision of signals varying through time depending on the agent-environment configuration. Agents must therefore perform multimodal fusion to select and filter relevant information by contrasting the shortcomings and redundancies of different modalities. For that purpose, we propose to combine a classical off-the-shelf manifold learning algorithm with dynamic neural fields (DNF), a training-free bio-inspired model of competition amid topologically-encoded information. Through the adaptation of DNF to irregular multimodal topologies, this coupling exhibits interesting properties, promising reliable localizations enhanced by the selection and attentional capabilities of DNF. In particular, the application of our method to audiovisual datasets (with direct ties to either psychophysics or robotics) shows merged perceptions relying on the spatially-dependent precision of each modality, and robustness to irrelevant features.

Index Terms—multimodal fusion, growing neural gas, manifold learning, dynamic neural field, selective attention

I. INTRODUCTION

When it comes to information processing and behavioral decision-making, the way we merge data coming from inputs of mixed nature is becoming increasingly important. Let us start with a toy example. A robot is given a task, for example: “touch the alarm clock when it goes off”. At first, the robot might be facing several objects resembling an alarm clock, which it should have no difficulty distinguishing. When a sound goes off, the robot should be able to locate its origin, but it is usually achieved with a low precision. Before taking an action, the robot has to select an object. Here, it should be the one clock-looking object that coincides most with the sound source localization. But how the modalities should be weighted depends not only on the task (a clock visible on the front has lower priority than sound coming from the side), but also on the reliability of the sensors (room reverberation can make sound orientation irrelevant).

The task in this example faces multiple challenges, starting with two: the fusion of sensory modalities of different availability and reliability, and the selection of (and attention towards) a target. To tackle these problems, most of model nowadays are based on deep learning. In this article, we propose another approach based on dynamic neural fields

(DNF), a bio-inspired model of neural activity [1]. It is a topologically-grounded continuous-time recurrent network, where weights are known and depend on the distance between neurons. With a mixture of short-range excitation and long-range inhibition, input stimuli are put in competition until a bubble of activity emerges, which can be interpreted as a decision of target selection and/or action. Additionally, temporal dynamics allows the bubble to remain stable despite input fluctuations and robust to potential distractors. DNF have seen various applications, including in robotics. In particular, the interaction properties of DNF make them very suitable for multimodal fusion [2], [3].

One limit that previous DNF implementations have faced lies in the nature of the manifold they evolve on. Most applications in the literature assume the existence of an underlying regular topology, most often 1D or 2D. But it is hardly representative of the disparities in the sensory space, disparities which become crucial when performing multimodal fusion. Indeed, let us take a look at the shape of stimuli perceived from the environment. The quantity of information available is huge, and the data an agent receives from its sensors is only a projection of it in a few given dimensions. Equipped with a standard camera, a robot will receive a 2D projection of the part of the environment it is facing. With one microphone, it can detect sounds from anywhere around it, but it can hardly locate them. Two microphones may enable some 1D sound localization along the axis on which they are aligned, usually azimuthal (with interaural time/level difference), and even a bit of 2D or 3D by exploiting the shape of pinnae with a head-related transfer function (HRTF) [4]. We must first account for the specificities of each sensory modality before we create behaviors that exploit it at best. Additionally, we must find a way to match complementary information from different modalities, which usually boils down to projecting stimuli onto a common manifold.

So, our first step will consist in learning unimodal manifolds. For this purpose, we will use growing neural gas (GNG) [5], a standard manifold learning algorithm which is quite parsimonious in light of the possible complexity of the sensory space. Then, we will suggest an easy-to-implement solution to create a multimodal manifold suitable for fusion. The main novelty of our work is that we will run DNF directly on this new topology, even though it lacks the regularity and low dimensionality of classical implementations. We will show that

This work was funded by French region Auvergne-Rhône-Alpes as part of the project AMPLIFIER.

properties of DNF in selection and attention are compatible with such fabricated manifolds, and that this coupling allows new possibilities for multimodal fusion taking into account the relative resolution of the modalities.

Our article is structured as follows. In section II, we will review the existing literature on manifold learning and DNF, and in particular their applications to multimodal fusion. Then we will describe our model in section III, and demonstrate its capabilities through three applications in section IV. We will conclude and discuss additional perspectives in section V.

II. PREVIOUS WORK

A. Manifold Learning

Sensors provide high-dimensional samples of the environment, but sensory spaces can often be projected onto manifolds of lower dimension. Deep learning methods are particularly suited for learning such manifold (see [6] for a review). For example, the last layers of a deep neural network have been shown to contain an intrinsic dimensionality that is smaller than the number of features in the data [7]. Dedicated methods such as variational autoencoders [8] learn structured embedding in an unsupervised manner. As our focus in this article is the study of coupling between DNF and irregular multimodal manifold, we will use simpler methods (i.e. self-organizing neural networks) that will provide more control and insight for the study.

In self-organizing maps (SOM), e.g. the Kohonen model [9], each neuron represents a prototypical input in the high-dimensional sensory space, so that the input space is projected onto a neural lattice of fixed shape and size. In neural gas (NG) [10], neurons are not arranged on a lattice, but are connected following a Hebbian rule, thus neurons with close prototypes are linked together. Eventually, the gas fills the input space in a way that matches the stimulus distribution. Growing neural gas (GNG) [5] is a derivative of NG, in which neurons are added (or deleted) over time until a chosen condition is met, thus adapting to the unknown input space spread.

Manifolds in multimodal fusion: Numerous articles have shown promising results in multimodal fusion using deep learning. Deep unsupervised learning can be used to project multimodal data on a low-dimensional manifold for use in robotics [11]. Inputs can be mixed during neural network training to exploit the correlations between modalities [12]. Reference [13] proposes a new type of deep neural network receiving multimodal inputs allocated through an attention module. Unfortunately, most of these works make the assumption that all multimodal data are related. Also, deep architecture are dedicated to one specific task and no generic architecture emerges [14].

We aim to create a new multimodal topology over which new dynamic properties could be applied, and self-organization offers solutions for a much lower cost [15]–[23]. SOM and their derivatives have long been used as models of multimodal fusion, but the ways modalities are combined can be very diverse. Map architectures can be divided in two categories. In the first, one SOM is trained for each

modality, then all unimodal maps are connected depending on a special learning rule [15]–[17]. In the second, unimodal maps link to a new multimodal SOM [18], [19] or NG [20] that combines all information. Additional layers of SOM can also be considered to create a hierarchical flow of information [21]–[23]. Additionally, models can be made more adaptive to time-dependant tasks with the help of “growing when required” maps [22], [23], an alternative to GNG designed for dynamic input distributions [24]. Some of these models have already been proof-tested for visual, auditory and/or proprioceptive modalities on hardware setups [21], [23] and robots [17], [19].

After multimodal maps and/or interconnected unimodal maps have been learned, we need a paradigm to dictate the way perception will occur. Multimodal perception can be seen as a form of decision pondering sensory inputs of different reliability and relevance. We follow the architectural choice made in [18] and [15], where dynamic neural fields (DNF) are used as the paradigm that governs fusion or segregation of stimuli in the multimodal topological space. DNF come with many useful properties for multimodal perception.

B. Dynamic Neural Fields

Originally stemming from neuroscience, DNF have various applications in robotics [25]. For example, visual attention may be cumulated with motor control to make a robot autonomously gaze at objects in its environment and learn a sensorimotor map [26]. DNF rely on a population of topologically connected units at a mesoscopic scale, where the apparent activity (or average membrane potential over assemblies of neurons) can be read to infer decisions at a behavioral level. The activity evolves through time depending on a sum of external stimulations and lateral interaction between neurons. Stimulated neurons will send strong excitation to their nearest neighbors, and moderate inhibition to neighbors located further apart, leading to the emergence of a stable bubble of activity. Depending on the parametrization, this can lead to several types of behavior [25]. With strong local excitation, the bubble can be self-sustaining, acting as long-term memory [26]. Long-range inhibition will create a competition between conflicting stimuli, until either one dominates the others, or they are merged in a single bubble at an interpolated position [3], [27]. Then, the self-maintaining bubble can be used for robust selective attention, able to ignore noise and minor distractors [28]. Ultimately, the output of DNF can be directly exploited to generate motor command [26], [29].

The properties of DNF can benefit greatly to multimodal fusion. It provides the tools not only to enhance robust decisions when modalities are congruent [2], but also to solve conflicts between modalities [3]. This is where the choice of the underlying manifold can be very important.

The vast majority of works using DNF assume the dynamics take place on a completely regular topology, e.g. a 2D lattice in the case of vision. However, there is no clear way of projecting two or more modalities onto the same lattice. In [2] and [3], strong assumptions are made on the shape of stimuli in a modality so that they fit in the topology of the other. To

tackle this issue, [15] proposes using separate manifolds for each modality, each learned by SOM, and apply DNF on each of them. Communication between modalities is ensured by a specific set of topographic connections.

The latter reference is actually one of the first to suggest using a learned manifold as the theater of neural dynamics. Otherwise, some attempts to alter the projection of inputs into the manifold have lead to satisfying results: [27] and [3] successfully reproduce biological behaviors after applying a logpolar transformation to visual stimuli, which models the discrepancies in the resolution of the human retina [30]. In [15], the projections received by neurons are altered, although they are still organized in a rectangular lattice. Since DNF are strongly dependant to the topography, and rely on a symmetrical interaction kernel¹, one may fear that breaking the regularity of the underlying topology may make DNF completely unpredictable.

An ensuing question would be how far from regular and/or rectangular can the underlying topology be for DNF to remain viable. If DNF could be made to operate on manifolds of unconstrained shape or dimension (easily accessible through GNG), then this would open the door to adding the properties of DNF to a new range of applications, starting with new capabilities in multimodal fusion like the ability to take into account the different resolution and reliability of all modalities. To our knowledge, this has not been tested. At best, suggestions have been made to approximate DNF activity using gaussian mixtures, sparsifying the space on which they operate to make them applicable in more complex topologies [33]. Yet, this latter approach still relies on a continuous regular space on which the lateral connectivity kernel function and Gaussian mixtures can be defined, which remains a strong limitation when processing high dimensional inputs.

III. MODEL

In this article, we use GNG to learn manifolds of the sensory space in each modality. We then assemble them into one multimodal graph, on which we use a DNF to produce behaviors that have, to our knowledge, never been implemented on this kind of manifold. These three steps are summarized in figure 1 and explained below.

A. Unimodal Topology Learning

In this part, we process modalities separately. As our focus in this article is not on tuning the unimodal topology learning on a specific task, we use the standard GNG algorithm with its original parameter values, as described in [5]. To summarize, GNG are trained by receiving a succession of randomly selected stimuli. Every time, the two neurons whose prototypical input match the stimulus best get a fresh connection. Then the best-matching unit (BMU) and its direct topological neighbors have their prototype moved towards the stimulus. Connections that have not been updated in a long time are removed, and

¹There have been suggestions to break the symmetry from the DNF side, either through asymmetrical kernels [31] or through distortions of the topology by predictive reinforcements [32], but both require an additional learning step.

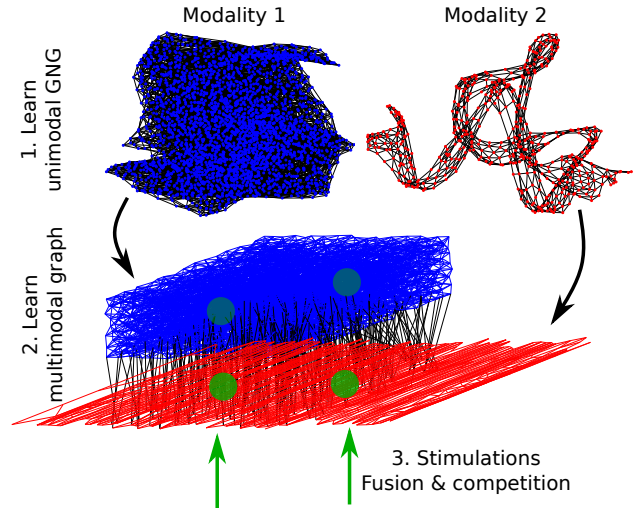


Fig. 1. Recap of the steps taken in this article. 1. Learn a growing neural gas in each modality. 2. Assemble them into one single graph by creating multimodal connections. 3. Present stimuli and compute multimodal activity.

isolated neurons as well. Then at fixed intervals, a new neuron is inserted. Its prototypical input is placed at the middle of the most activated connection.

B. Multimodal Topology Learning

For a first milestone, we will focus on bimodal architectures in the rest of this article. As a reminder, bimodal architectures in self-organization literature often merge data in one of two ways: a multimodal map is created that receives information from the unimodal ones, or new connections are added between the unimodal maps, each having its own processing unit. We propose an intermediate solution that is the most economical of all: we create a new bimodal graph that contains all nodes and edges from one modality, and all nodes and edges from the other. To create the crossmodal edges, we connect neurons of the two modalities that fire together, which is similar to an Hebbian learning. More precisely, the algorithm is: We draw a random multimodal input. If it lies in the sensory range of both modalities, we find the BMU in each GNG and connect them (if they are not already connected). We repeat until a certain proportion of nodes have at least one crossmodal edge.

C. Selection of Activity

Once the bimodal graph is created, its associated neurons can be stimulated by sensory inputs (through their respective modality), and we can use DNF to select and attend to a stimulus. DNF are usually expressed as an integro-differential equation in a continuous field of neurons, that is later discretized and computed using the Euler method. The integration of DNF is comparable to the simulation of continuous-time recurrent neural networks. In DNF, the distance between neurons plays an important role, as it determines whether they will excite or inhibit one another. Our model differs from others in the literature in that all neurons do not share a

common coordinate system. So, we need to adapt the DNF equation, so that the distances are defined on the graph, and only that. We rely on the standard distance from graph theory, i.e. the number of edges on the shortest path between any two vertices.

In our model, each neuron is tied to a specific modality. So, the external input received individually will be modality-specific (although the rest of DNF operations will not be). To ensure that the total amount of external stimulation is independent from the local resolution of a modality, we will order all neurons of a modality by their proximity to the stimulus (using the euclidian distance in the coordinate system of that modality), and stimulate them descendingly according to their rank. For each neuron indexed k , given a stimulus indexed i , we note $r_{k,i}$ the rank of proximity between the prototypical input of k and the coordinates of i . The external stimulation I_k received by k is given by:

$$I_k = \lambda_{m,i} e^{-\frac{r_{k,i}^2}{2\sigma_i^2}} \quad (1)$$

where $\lambda_{m,i}$ is the intensity of stimulus i with regards to k 's modality m . A neuron can only receive external inputs from its own modality.

Next, we compute the evolution of activity in the graph over time. The following is completely modality-agnostic. The potential U_k of neuron k is initialized as 0 and updated incrementally by²:

$$\Delta U_k = \frac{\Delta t}{\tau} \left(-U_k + I_k + \sum_{k'} W(\langle k, k' \rangle) f(U_{k'}) + h \right) \quad (2)$$

where Δt is the simulated time between steps, τ a time constant that determines the speed of DNF updates, f an activation function (ReLU), and h a negative resting level. $\langle \cdot, \cdot \rangle$ designates the minimal distance in number of edges between two nodes in the bimodal neural gas, and W is a weight function expressed as:

$$W(\delta) = \lambda_+ e^{-\frac{\delta^2}{2\sigma_+^2}} - \lambda_- e^{-\frac{\delta^2}{2\sigma_-^2}} \quad (3)$$

with amplitudes $\lambda_+ > \lambda_- > 0$ and widths $\sigma_+ < \sigma_-$. W can be seen as a kernel shaped like a mexican hat [1].

One possible way to interpret the outcome is to read the output $f(U)$. It is common to take a barycenter of the output as an estimator of the position targeted by the model. While we are not supposed to know an euclidian topology in which the positions of GNG nodes can be averaged, we can still use the input data to interpolate a corresponding location in a 2D euclidian space for each neuron. We will do that for our experimentations, but please note that this interpolation will not always be possible. Similarly, for the GNG, we will plot them by putting all nodes to their asserted location, only for visualization purposes.

²In this equation, only U_k is incremented over time, and the inputs I_k are static. However, none of our hypotheses prevent the inputs from being updated over time. We make this choice because dynamic inputs are not necessary for the results presented in this paper. Otherwise, equation (2) could be written by expressing $U_k(t)$ as a function of $U_*(t - \Delta t)$ and $I_k(t)$.

TABLE I
RANGES OF INPUTS IN THE EXTERNAL ENVIRONMENT

Section	Modality	X-range	Y-range	Z-range
IV-A	vis.	[0, 90]	[-45, 45]	-
	aud.	[0, 90]	[-45, 45]	-
IV-B	vis.	[-45, 45]	[-45, 45]	-
	aud.	[-90, 90]	[-45, 85]	-
IV-C	vis.	[-45, 45]	[-45, 45]	[0, 45]
	aud.	[-90, 90]	[-45, 85]	-

IV. RESULTS

Our results will be divided in three parts, with a common protocol for all. For this article, we will consider two modalities, vision and audition. That can correspond for example to a robot asked to locate a visual and/or audible stimulus. We test three setups that take into account challenges that might happen in the robot perception: differences of resolution within the same sensory space (section IV-A), high-dimensional feature space (IV-B), and non-relevant features (IV-C).

So, the main difference between the setups will be in the first step of our model, the generation of the unimodal manifolds (described in section III-A). For the GNG training, a stimulus location will be drawn within the subspace of the environment that is accessible to the appropriate sensors. For example, a robot's visual perception might be restricted to the space in front of them, while their auditory range might be all around them. Input ranges are listed in table I. Then, we simulate the information that would be received from the sensors if a real stimulus was sent from this position. The way they are preprocessed will be defined in each subsection.

We have set an upper limit to the number of neurons in the GNG. Otherwise, the resolution could become excessively high, increasing the computational cost for no valid reason. Once the limit is reached, the GNG is trained like a regular NG, except that nodes that have become irrelevant can still be removed and replaced. This is still more efficient than starting with all neurons and training a NG from the beginning.

The creation of a bimodal manifold is roughly the same in all setups. For the DNF, input stimuli will be specified in each scenario, depending on the properties to showcase. For the same reasons, parameters might need to be adjusted slightly from one setup to the next. All values are given in table II.

A. Bio-inspired Model of Audiovisual Processing

Our first experimentation is inspired from observations in neurophysiology. Human visual perception is affected by the heterogeneous distribution of sensors in the retina, giving a higher resolution in the center of the field of view (the fovea) than in its periphery. This disparity can be observed in brain regions processing visual information, such as the superior colliculus [30]. A mathematical model of the disparity between fovea and periphery, using a logpolar transformation, has been suggested by [30], and previous works have coupled it with DNF for visual [27] and audiovisual processing [3].

TABLE II
PARAMETERS USED IN OUR DNF IMPLEMENTATION. SPREAD
PARAMETERS ARE EXPRESSED IN ARBITRARY UNIT THAT DENOTES THE
MINIMAL NUMBER OF EDGES THAT SEPARATE TWO NEURONS.

Parameter	Value		Meaning
	IV-A	IV-B & IV-C	
Simulation settings			
Δt	0.01	0.01	Time step
σ_I	2.5	2.5	Spread of stimulus
$\lambda_{vis, A}$	2	2	Strength of visual bottom stimulus
$\lambda_{vis, B}$	2.4	2.02	Strength of visual top stimulus
$\lambda_{aud, A}$	2.4	1.5	Strength of audio bottom stimulus
$\lambda_{aud, B}$	2	0	Strength of audio top stimulus
DNF parameters			
τ	0.1	0.1	Time constant
λ_+	0.4	0.55	Amplitude of lateral excitation
σ_+	2.5/3/3.5	1.5	Spread of lateral excitation
λ_-	0.3	0.3	Amplitude of lateral inhibition
σ_-	$+\infty$	10	Spread of lateral inhibition
h	-1	-1	Resting level

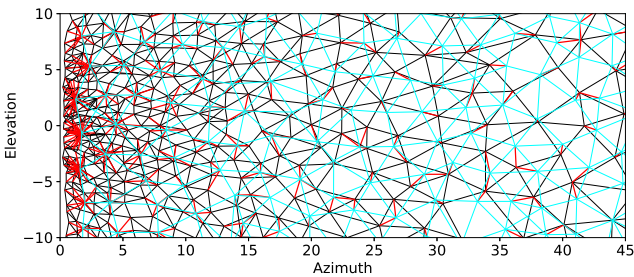


Fig. 2. Sample representation of a bimodal graph. Edges are colored depending on the modalities of the neurones they connect. Visual-visual: black. Auditory-auditory: cyan. Visual-auditory: red.

Models of the superior colliculus are not only useful for computational neuroscience. While cameras used by robots are supposed to have a homogeneous resolution, they might happen to have blurry spots because of dirt or wear. Other modalities may also have a high variance in resolution. The logpolar transformation is a straightforward way of testing these variations in a controlled setting. Additionally, even when the camera sensory space is perfectly regular, it has been suggested that adding a logpolar transformation on top of it could improve gaze control in robots [34].

1) *Sensory space*: In light of the aforementioned hypothesis, we take coordinates of a visual stimulus in a regular 2D visual hemifield, and displace them following the logpolar transformation in [30]. The new 2D coordinates are used as inputs for the visual GNG. Since we study the effect of variable resolutions in one modality, the other modality, audio, will be modeled as a regular 2D space as in [3], with the same range as vision (table I), so that it does not interfere with the analysis. Both GNG are given 1000 nodes maximum.

2) *Produced manifolds*: A sample of the bimodal graph is shown in figure 2. For visualization, visual nodes are placed according to a reverse logpolar transformation of their features,

and auditory nodes according to their raw coordinates. The unimodal GNG are superposed with different colors.

As expected, the visual GNG has a much higher resolution around the fovea (0°), as can be presumed by the high density of nodes. It gradually decreases as the azimuth augments. On the contrary, the auditory GNG has roughly the same resolution everywhere. Connections between neurons of different modalities are shown in red³. For azimuths between 0° and approximately 30° , vision has a better resolution than audition: most nodes from the audio GNG are connected to multiple visual nodes. The trend is reversed for higher azimuths.

3) *Resulting properties*: After the bimodal manifold is created, we are interested in seeing what a DNF would select when confronted to conflicting bimodal stimulus. It is expected that near the fovea, vision is more reliable, so it should have a bigger weight in the fusion than audition. To test this, we put two conflicting stimuli A and B at a common azimuth x , and elevations -5° and 5° respectively. Both stimuli can be both seen and heard, but A is 20% more audibly salient than B, and B is 20% more visually salient than A.

When tested on a unimodal manifold, the DNF has no trouble selecting either A or B. Every time, the most salient stimulus in its respective modality has a higher chance of being selected. Occasionally, the DNF forms a bubble in-between the stimuli. This is mostly visible for higher azimuths in the visual GNG. The reason is that the resolution is so low that A and B are separated by only a few edges. The DNF does not have access to the corresponding inputs of its neurons viewed from the exterior. Thus, when viewed from inside the model, they are topologically very close to each other. So, the DNF treats the stimuli as if they were right next to each other, and merges them into a bubble of activity located at their center of mass.

In the bimodal manifold, the stochasticity in the creation of the GNG starts having an impact, as it may seemingly give a locally higher resolution to a modality when it is not expected. A might be selected instead of B, when B is more salient, just because B stimulates a region with fewer neurons or connections than average. To separate the random effect caused by the creation of the GNG, we create 50 bimodal manifolds, and test a run of DNF on 90 different azimuths for each of them. The results are aggregated in figure 3. As we suspect that the distance at which stimuli are merged depends on the width of the DNF kernel, we couple in our analysis the effect of resolution with the value of σ_+ . We test three different values of σ_+ , represented by three different colors: green, red, blue from thinnest to widest.

The curves represent the outcome of two mixed logistic regressions. The fit of the black curve is obtained after

³For this model, we initially observed that a lot of visual neurons close to the fovea were never connected to auditory ones. Because there are so many of them in a very close space, a huge number of random draws is required before they are all visited. To ensure that the merging task would not be hindered by a lack of connectivity, we biased the draw of external stimuli so that the prototypical input of every neuron was drafted. We found that this manual bias has no effect on the graph connectivity outside the fovea. This draw method is not applicable to most scenarios, since we are not supposed to know the actual coordinates of the neurons in the external environment.

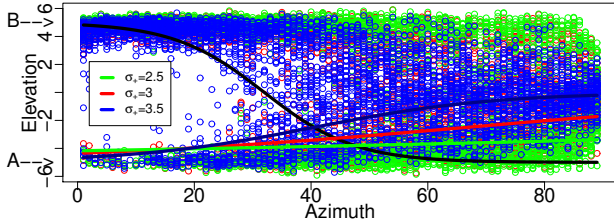


Fig. 3. Statistical model of the modality priority change (in black) and the stimulus merging. One point represents the barycenter of the output of one of the 3 differently parametrized DNF (green: $\sigma_+ = 2.5$, red: $\sigma_+ = 3$, blue: $\sigma_+ = 3.5$), on one of the 50 randomized GNG, with two bimodal stimuli A and B at azimuth x and elevations $\pm 5^\circ$. The black curve shows a logistic regression of the switch between preferred stimuli. Colored curves show logistic regressions of the stimulus merging effect depending on values of σ_+ .

cancelling the merging effect, and shows a clear switch of preference from B to A centered on 32° . B is more likely to be selected than A when the visual modality is the most reliable, and vice versa. Logically, this effect is independent of σ_+ variations. This amounts to the DNF automatically selecting a stimulus according to the most reliable sensor.

The fit of the colored curves are obtained by canceling the switch effect. We can see a convergence from $\pm 5^\circ$ to 0° elevations, although for lower values of σ_+ , the limit at 0° is not reached before the end of the field of view. Only the lower curves are displayed but the effect is symmetrical.

The results show two trends. First, from the higher concentration of points at the 5° elevation in the leftmost part of the figure, we can see that B (visually stronger) is more often selected in lower azimuths than A. Then A is preferred for higher azimuths. Second, we see that the probability of A and B being merged (manifesting as an increasing concentration of points around 0°) increases with the azimuths. As we expected, the distance at which they are merged depends a lot on the value of σ_+ . The larger the interaction kernel, the sooner the merging seems to happen.

B. Real-world Robotic Sensory Data

In the previous section, we used manufactured data to showcase DNF selection properties in manifolds of variable resolution, favoring the most reliable modality. In this section, we will partly use real experimental data and show that these properties are still available in more complex sensory spaces. Our main change will be on auditory preprocessing. One way of performing sound source localization for robots is to compute a HRTF, a function that associates spectral features (caused by interferences on the signal by the head and pinnae) to source orientations [4]. Meanwhile, vision is less of a challenge nowadays, as extracting the position of an object from an image is easily achievable, and one can reasonably expect to have a homogeneous resolution in most cases.

1) *Sensory space*: Data provided by [35] includes head-related impulse responses of a robot equipped with artificial pinnae, to a sound located at different angles. Given an external stimulus position in 2D, we can interpolate the responses received by the two robotic ears within a specific

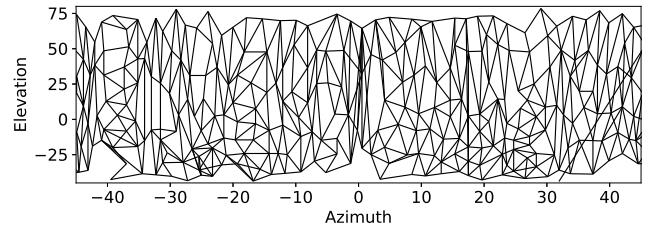


Fig. 4. Sample of the auditory graph obtained from HRTF data. The 2D location of neurons is not known by the GNG, it has been interpolated from their prototypical input in HRTF space, for visualization purposes only. Note that the x -axis and y -axis have different scales.

range (table I). We then compute their Fourier transform and make the difference between the ears to obtain a HRTF. In the end, each audio input is 100-dimensional.

For vision, we will consider a robot with an intact camera and assume it can roughly estimate the 2D coordinates of an object in front of it. We do not need visual and auditory perception to have the same range. Realistically, stimuli can be heard from more orientations than they can be seen. To keep resolutions approximately balanced, we will use respectively maximum 500 and 200 nodes for auditory and visual GNG.

2) *Produced manifolds*: The visual GNG is very similar to the auditory GNG in the previous section, which also directly received stimuli drawn from a regular 2D space. The new auditory one, however, has a distinct shape. Figure 4 shows what the GNG looks like after placing each node at the source location that would match its audio (100D) coordinates best. The graph appears to be stretched vertically.

3) *Resulting properties*: Like in the previous scenario, we test the DNF with two stimuli A and B. This time, they are separated both horizontally and vertically. Stimulus A has congruent audio and visual components, while B is not audible but visually more salient by 1%. It is expected that A should be selected over B, as A is consistent over modalities. Results are synthesized in figure 5.

In the visual-only manifold, B largely takes precedence. A is mostly inhibited, with some (negative) residual activity left. This is expected, as B is more visibly salient, but it is worth noting that the 1% difference between $\lambda_{\text{vis}, A}$ and $\lambda_{\text{vis}, B}$ matters. While not shown here, we have tested swapping the intensity values, and A does take precedence in that inverted case. We are in a situation where both stimuli are considered equally by the DNF, and a very small difference in intensity is enough to bias the competition towards one or the other. This is a very standard observation in DNF literature, but it is still worth noting considering the topology is not entirely regular.

In the audio-only manifold, A is trivially selected, but we can see some loss of precision in elevation: the barycenter is found 7° higher than the actual stimulus. This is very consistent with the general lack of elevation-wise precision in auditory perception.

The precision is improved in the bimodal manifold. As would be expected, audiovisual congruent stimulus A is selected over visual-only B. But the barycenter is also closer

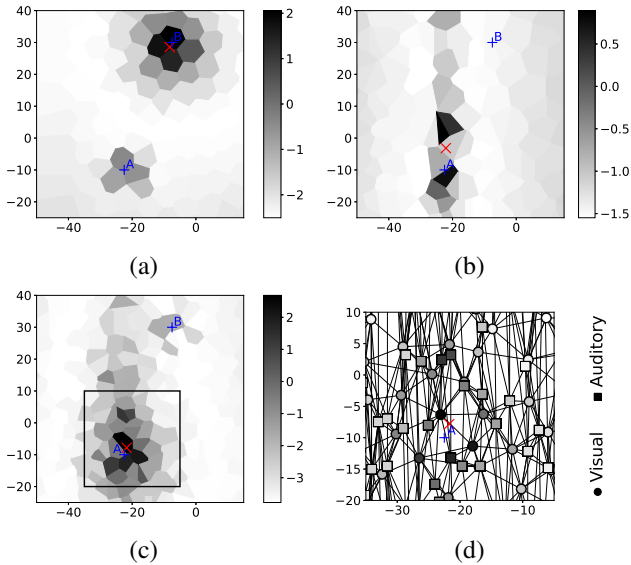


Fig. 5. Results of stimulus selection by DNF unimodal and bimodal GNG. These 2D depictions use neuron positions interpolated from the source data (for visualization). Shades of gray reflect neuron potential U . Red crosses indicate the barycenter of output activation $f(U)$ in the reconstructed 2D projection. (a) Visual-only neural gas with two stimuli located at A and B, with B slightly more salient. Nodes are represented by Voronoi cells, edges connecting nodes are not shown. (b) Auditory-only neural gas, with only one input at A. (c) Bimodal neural gas. Its input is the sum of the ones used for (a) and (b). (d) Zoom on (c) around A, where all nodes and edges are shown.

to the actual stimulus position than in the audio-only case, meaning the visual elevation-wide better precision had a positive impact. Again, the enhanced multimodal precision is a classical observation in either neuroscience or machine learning, but it is worth noting that it persists when working with a complex underlying topology.

When we look more closely at the nodes around A, we can see that despite there being a lot of edges in all directions, a few neurons form a discernable bubble. It is interesting that these neurons come indiscriminately from both modalities. One could have feared an outcome where only visual neurons interact with each other, and auditory neurons, less regularly distributed, only serve to transmit a little bit of auditory stimulation. On the contrary, the crossmodal connections play an important part, so that the DNF does not leave out one modality for the other. When both are useful, both are used.

C. Dealing with a Superfluous Dimension

1) *Sensory space*: This setup is similar to the previous one, except the visual sensory space is now 3D. We add a dimension that is not relevant to the task, e.g. color when a robot is asked to select an object designated by shape only. Since the visual space expands, and GNG are not advanced enough to reduce the dimensionality when the amount of possible inputs increases brutally, we also increase the number of neurons in the visual GNG to 3000. The rest of the setup remains the same.

2) *Resulting properties*: We did the same experiments as in section IV-B. Stimuli A and B are given the same color, so that

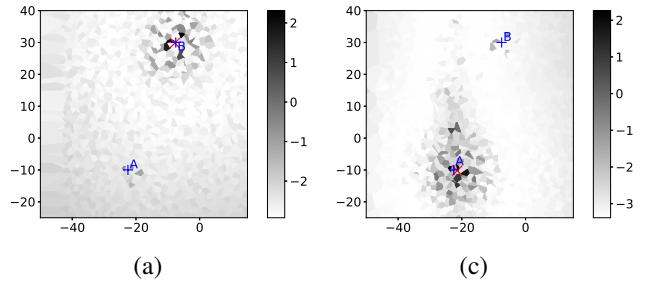


Fig. 6. Same as figure 5 with a supplementary dimension in the visual modality. The third dimension is orthogonal to the plane used in this representation.

their distance in the external environment remains the same as before. According to our preliminary tests, the conclusion would be the same with stimuli of different colors. Results are displayed in figure 6. Only the visual-only and audiovisual conditions are shown, since the auditory-only condition is the same as before, and the zoom-in picture with edges is hardly readable. As a reminder, the visualizations are still made using x - and y -axes, meaning the new color axis is completely flattened. These presentations are akin to looking at a cube from a side, hence the dense Voronoi tessellation and the scattered activity.

We find that the outputs are strikingly similar, i.e. a preference for multimodal consistent inputs and improving audio precision, despite a big increase in the number of neurons and edges, many of which are irrelevant to the task. This shows robustness of the model to distracting dimensions.

V. CONCLUSION AND PERSPECTIVES

Our model consists in two unimodal GNG, trained using the standard algorithm by [5], then connected to form one new multimodal manifold with a simple Hebbian rule. This manifold is used as a support for neural dynamics that are implemented by adapting the DNF paradigm [1]. Our model was tested on multiple setups, including real data. The main novelties of our work are twofold. First is the use of neural dynamics in a multimodal manifold of unspecified dimensionality or regularity, a capability of DNF that has not been showcased before. The field applies on a learned manifold that is faithful to each unimodal sensory space, and is not hindered by irrelevant dimensions. Second is the combination of the multimodal topology with DNF to obtain interesting properties such as the contribution of different modalities that depends on their respective learned resolution, the selection of the most relevant multimodal stimulus by using the best information each modality had to offer, and the filtering of irrelevant informations. These results are scalable to applications with more than two modalities.

As we have seen when adding a dimension, the number of neurons in the GNG necessary to keep the same resolution, and consequently the computational cost of the model, may increase drastically when the sensory space is broadened. This would not be an issue with deep neural networks, that are very

effective at finding intrinsic dimensions in data [7]. It would be interesting to see whether manifolds created by deep learning are also suitable vectors of neural dynamics. This would be complementary to existing approaches to encode topological maps with neural networks [36], [37].

In our model, learning of the multimodal topologies and their use for multimodal fusion are decoupled. An interesting perspective would be to perform them simultaneously, which raises some challenges like making the model robust to the temporal dynamics and to the detection of relevant features for learning and fusion. Another perspective is to study multimodal active perception, where the internal perception will be related to motor actions to explore the environment. DNF are well suited to model saccades [29]. This raises open questions related to multimodal attention and active perception.

ACKNOWLEDGMENT

Most of the computations presented in this paper were performed using the GRICAD infrastructure (<https://gricad.univ-grenoble-alpes.fr>), which is supported by Grenoble research communities.

REFERENCES

- [1] S.-I. Amari, "Dynamics of pattern formation in lateral-inhibition type neural fields," *Biological Cybernetics*, vol. 27, no. 2, pp. 77–87, 1977.
- [2] C. Schauer and H. M. Gross, "Design and optimization of Amari neural fields for early auditory-visual integration," in *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, vol. 4, 2004, pp. 2523–2528.
- [3] S. Forest, J.-C. Quinton, and M. Lefort, "A dynamic neural field model of multimodal merging: application to the ventriloquist effect," *Neural Computation*, in press. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-03600794>
- [4] S. Argentièri, P. Danès, and P. Souères, "A survey on sound source localization in robotics: From binaural to array processing methods," *Computer Speech & Language*, vol. 34, no. 1, pp. 87–112, 2015.
- [5] B. Fritzsche, "A growing neural gas network learns topologies," in *Advances in Neural Information Processing Systems*, G. Tesauro, D. Touretzky, and T. Leen, Eds., vol. 7. MIT Press, 1995.
- [6] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [7] A. Ansuini, A. Laio, J. H. Macke, and D. Zoccolan, "Intrinsic dimension of data representations in deep neural networks," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.
- [8] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014.
- [9] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.
- [10] T. Martinetz and K. Schulten, "A "neural-gas" network learns topologies," *Artificial neural networks*, vol. 1, pp. 397–402, 1991.
- [11] A. Droniou, S. Ivaldi, and O. Sigaud, "Deep unsupervised network for multimodal perception, representation and classification," *Robotics and Autonomous Systems*, vol. 71, pp. 83–98, 2015.
- [12] X. Yang, P. Ramesh, R. Chitta, S. Madhvanath, E. A. Bernal, and J. Luo, "Deep multimodal representation learning from temporal data," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5066–5074, 2017.
- [13] A. Jaegle, F. Gimeno, A. Brock, O. Vinyals, A. Zisserman, and J. Carreira, "Perceiver: General perception with iterative attention," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 2021, pp. 4651–4664.
- [14] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *ICML*, 2011.
- [15] M. Lefort, Y. Boniface, and B. Girau, "SOMMA: Cortically inspired paradigms for multimodal processing," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*, 2013, pp. 1–8.
- [16] L. Khacef, L. Rodriguez, and B. Miramond, "Brain-inspired self-organization with cellular neuromorphic computing for multimodal unsupervised learning," *Electronics*, vol. 9, no. 10, 2020.
- [17] N. Gonnier, Y. Boniface, and H. Frezza-Buet, "Input prediction using consensus driven SOMs," in *2021 8th International Conference on Soft Computing & Machine Intelligence (ISCMCI)*. IEEE, 2021, pp. 38–42.
- [18] O. Ménard and H. Frezza-Buet, "Model of multi-modal cortical processing: Coherent learning in self-organizing modules," *Neural Networks*, vol. 18, no. 5, pp. 646–655, 2005.
- [19] S. Lallec and P. F. Dominey, "Multi-modal convergence maps: from body schema and self-representation to mental imagery," *Adaptive Behavior*, vol. 21, no. 4, pp. 274–285, 2013.
- [20] M. Vavrečka and I. Farkaš, "A multimodal connectionist architecture for unsupervised grounding of spatial language," *Cognitive Computation*, vol. 6, no. 1, pp. 101–112, 2014.
- [21] M. Johnsson, M. Martinsson, D. Gil, and G. Hesslow, "Associative self-organizing map," in *Self Organizing Maps*, J. I. Mwasigi, Ed. Rijeka: IntechOpen, 2011, ch. 30.
- [22] G. I. Parisi, J. Tani, C. Weber, and S. Wermter, "Emergence of multimodal action representations from neural network self-organization," *Cognitive Systems Research*, vol. 43, pp. 208–221, 2017.
- [23] K. Huang, X. Ma, R. Song, X. Rong, X. Tian, and Y. Li, "An autonomous developmental cognitive architecture based on incremental associative neural network with dynamic audiovisual fusion," *IEEE Access*, vol. 7, pp. 8789–8807, 2019.
- [24] S. Marsland, J. Shapiro, and U. Nehmzow, "A self-organising network that grows when required," *Neural networks*, vol. 15, no. 8-9, pp. 1041–1058, 2002.
- [25] G. Schöner, J. Spencer, and DFT Research Group, *Dynamic Thinking: A Primer on Dynamic Field Theory*, ser. Oxford Series in Developmental Cognitive Neuroscience. Oxford University Press, 2015.
- [26] Y. Sandamirskaya, "Dynamic neural fields as a step toward cognitive neuromorphic architectures," *Frontiers in Neuroscience*, vol. 7, p. 276, 2014.
- [27] W. Taouali, L. Goffart, F. Alexandre, and N. P. Rougier, "A parsimonious computational model of visual target position encoding in the superior colliculus," *Biological Cybernetics*, vol. 109, no. 4, pp. 549–559, 2015.
- [28] J. Fix, N. Rougier, and F. Alexandre, "A dynamic neural field approach to the covert and overt deployment of spatial attention," *Cognitive Computation*, vol. 3, no. 1, pp. 279–293, 2011.
- [29] J.-C. Quinton and L. Goffart, "A unified dynamic neural field model of goal directed eye movements," *Connection Science*, vol. 30, no. 1, pp. 20–52, 2018.
- [30] F. P. Ottes, J. A. V. Gisbergen, and J. J. Eggermont, "Visuomotor fields of the superior colliculus: A quantitative model," *Vision Research*, vol. 26, no. 6, pp. 857–873, 1986.
- [31] M. Cerda and B. Girau, "Asymmetry in neural fields: a spatiotemporal encoding mechanism," *Biological cybernetics*, vol. 107, no. 2, pp. 161–178, 2013.
- [32] J.-C. Quinton and B. Girau, "Predictive neural fields for improved tracking and attentional properties," in *The 2011 International Joint Conference on Neural Networks*. IEEE, 2011, pp. 1629–1636.
- [33] —, "A sparse implementation of dynamic competition in continuous neural fields," in *Brain Inspired Cognitive Systems 2010 - BICS 2010*, Madrid, Spain, 2010.
- [34] L. Manfredi, E. S. Maini, and C. Laschi, "Neurophysiological models of gaze control in humanoid robotics," in *Humanoid Robots*, B. Choi, Ed. Rijeka: IntechOpen, 2009, ch. 10.
- [35] V. R. Algazi, R. O. Duda, D. M. Thompson, and C. Avendano, "The CIPIC HRTF database," in *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No. 01TH8575)*. IEEE, 2001, pp. 99–102.
- [36] P. Hartono, P. Hollensen, and T. Trappenberg, "Learning-regulated context relevant topographical map," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 10, pp. 2323–2335, 2014.
- [37] P. Hartono, "Mixing autoencoder with classifier: conceptual data visualization," *IEEE Access*, vol. 8, pp. 105 301–105 310, 2020.

Suréchantillonnage Actif pour Modérer l'Apprentissage de Biais Visuels

A. Devillers¹, B. Alcaraz², M. Lefort¹

¹ Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, F-69622 Villeurbanne, France

² Université du Luxembourg

alexandre.devillers@liris.cnrs.fr, benoit.alcaraz@uni.lu, mathieu.lefort@liris.cnrs.fr

Résumé

Les techniques de reconnaissance d'image sont devenues particulièrement performantes lors de cette dernière décennie. Cependant l'apprentissage sur des données biaisées, par exemple, la présence régulière d'un fond bleu derrière un poisson, réduit drastiquement les performances des approches actuelles qui ont tendance à apprendre ces biais. Dans ce papier, nous proposons ImRAN, une méthode d'apprentissage actif qui consiste à surreprésenter les données mal classifiées, et qui améliore l'état de l'art sur le jeu de données Biased MNIST lorsqu'il est très biaisé.

Mots-clés

Apprentissage profond non-biaisé, apprentissage de raccourcis, apprentissage actif, classification d'image.

Abstract

Image recognition techniques have become particularly powerful in the last decade. However, learning on biased data, for example, the regular presence of a blue background behind a fish, drastically reduces the performances of current approaches that tend to learn these biases. In this paper, we propose ImRAN, an active learning method that consists in over-representing misclassified data, which improves the state of the art on the Biased MNIST dataset when it is highly biased.

Keywords

De-biased deep learning, shortcut learning, active learning, image classification.

1 Introduction

Les techniques de reconnaissance d'image sont devenues particulièrement performantes lors de cette dernière décennie [15], en particulier grâce à l'utilisation de réseaux de neurones convolutifs [10]. Ces approches statistiques dépendent principalement de la qualité des données d'apprentissage. Cependant, en pratique certains jeux de données peuvent présenter un ou plusieurs biais, c'est-à-dire une corrélation plus ou moins forte entre un élément récurrent des images et leurs classes, sans pour autant que cet élément soit prédictif. Or, l'apprentissage d'un modèle se basant sur ces caractéristiques se traduirait par une faible ca-

pacité de généralisation hors du domaine d'entraînement. Les réseaux de neurones, ayant tendance à converger vers ces solutions lorsqu'elles sont simples et à exploiter ces raccourcis d'apprentissage [8], sont possiblement inopérants sur certains de ces jeux de données. Par exemple, dans une tâche de classification d'animaux, les photos de poissons ont régulièrement un fond bleu. Un modèle d'apprentissage profond risque ainsi de converger vers un modèle de décision classifiant toute image avec un fond bleu comme étant un poisson. Cela limite la reconnaissance de poissons dans d'autres décors, et crée des erreurs de classification pour d'autres images avec un fond bleu n'étant pas des poissons. Avoir un algorithme performant malgré la présence de biais est une propriété souhaitable, car il n'est pas toujours possible de connaître leur présence à l'avance dans le jeu de données. Une mauvaise gestion de ces derniers peut donc entraîner une perte importante de performances lors d'une utilisation applicative de ces algorithmes. Cela peut dans certains cas, comme les voitures autonomes, causer des dommages matériels ou physiques. Dans un autre contexte applicatif, une approche résiliente aux biais permettrait de limiter les biais sociétaux des bases de données, comme l'influence du genre, de la couleur de peau, ou autres facteurs pouvant dépendre des échantillons récoltés pour l'apprentissage [23]. Cependant, puisque la présence ou la nature des biais n'est pas toujours connue à l'avance, il est souhaitable qu'une méthode limitant l'apprentissage de biais ne dégrade pas les performances de base lorsqu'il n'y en a pas.

Dans ce papier, nous proposons la méthode *Image Representation Avoiding Naive learning* (ImRAN) qui modifie l'apprentissage d'une architecture de réseaux de neurones convolutionnels. Cette méthode consiste en un changement automatique et en ligne de la distribution du jeu de données d'apprentissage, ainsi qu'au bruitage des images afin de rajouter un peu de variabilité. Un élément sera d'autant plus vu que sa classification aux époques précédentes a été faible. En effet, si le modèle tend à apprendre le biais, les données mal classifiées seront principalement celles non-biaisées, et multiplier leurs occurrences dans le jeu d'apprentissage aura donc tendance à le débiaiser.

Le papier est divisé comme suit. Tout d'abord, la Section 2 recontextualise notre article dans l'état de l'art. En-

suite, nous introduisons notre méthode ImRAN dans la Section 3. Dans la Section 4, nous présentons le protocole et analysons les résultats obtenus. Enfin, nous discutons de notre approche et des travaux futurs dans la Section 5, puis concluons dans la Section 6.

2 Travaux antérieurs

La nature des réseaux de neurones à exploiter des raccourcis d'apprentissage, tel que les biais, réduit la confiance qui leur est accordée et peut limiter leur application pratique [8]. La forte présence de biais dans un jeu de données peut réduire à un niveau proche de l'aléatoire les performances des réseaux de neurones dans une tâche de classification binaire [16]. Un autre exemple où l'apprentissage de biais affecte les performances est dans les approches de *Visual Question Answering* (VQA), où les modèles ignorent la modalité visuelle pour reposer principalement sur des biais statistiques présents dans le langage [2]. Par exemple, une question comme "Quelle est la couleur de la banane présente à l'image?" aura comme réponse "Jaune" indépendamment de la couleur de la banane dans l'image. Dans cet article, nous nous concentrons sur de la classification d'images, bien que notre méthode puisse éventuellement être adaptée à d'autres tâches ou modalités.

Parmi les approches existantes visant à réduire l'apprentissage de biais visuels, certaines demandent d'avoir un certain nombre de connaissances expertes vis-à-vis des biais présents. Une partie des méthodes repose sur le fait de connaître au préalable, pour chaque image du jeu d'entraînement, la présence ou non d'un biais [17, 21, 23]. Bien que ces méthodes puissent présenter des performances importantes, elles sont difficilement applicables en pratique. En effet, un jeu de données n'est pas toujours connu à l'avance comme comportant des biais. De plus, dans l'éventualité où l'information serait disponible, l'accès à une annotation des biais reste peu fréquent. D'autres méthodes demandent uniquement à connaître le type de biais, ou bien exigent qu'il soit de bas niveau sémantique [3, 6, 19, 22]. Cependant, le spectre d'application de ces méthodes reste limité.

Certaines approches de la littérature, qu'elles utilisent ou non des informations sur le biais, sont basées sur la modification de la fonction de coût qu'optimise le modèle, et plus particulièrement sur l'ajout d'un terme de régularisation dont l'objectif est d'empêcher la convergence vers les raccourcis d'apprentissage exploitant les biais présents dans les données d'apprentissage. C'est par exemple le cas de EnD [21] qui force les représentations des données partageant un même biais à être différentes (ce qui demande de connaître le biais), tout en rapprochant entre elles les représentations des données d'une même classe. Des travaux se sont concentrés sur l'utilisation de réseaux de neurones naïfs, peu profonds et avec un champ perceptif réduit, afin de converger vers l'apprentissage des biais présents dans les jeux de données [3, 6, 19, 22], pour ensuite pénaliser l'apprentissage d'un autre modèle, plus complexe, afin qu'il évite les biais appris par les modèles naïfs. Ces approches font généralement l'hypothèse forte, appuyée

par des connaissances expertes, que les biais sont souvent des caractéristiques de bas niveau tandis que les caractéristiques prédictives des classes sont hautement sémantiques. C'est notamment le cas lorsque le biais relève de la présence d'une couleur ou d'une texture. Récemment, la fonction de coût *Generalized Cross Entropy* (GCE) [24], connue comme facilitant l'apprentissage des biais, a été utilisée pour remplacer les architectures naïves et accentuer l'apprentissage des biais du premier modèle [16, 19]. Enfin, l'approche RUBi [5] propose une fonction de coût qui force une rétro-propagation d'un gradient faible pour les exemples où le modèle a déjà un haut niveau de confiance dans la bonne réponse après *softmax*, et à l'inverse, rétro-propage un gradient fort si le modèle avait un taux de confiance faible. L'idée sous-jacente est d'attribuer une plus grande importance aux exemples non-biaisés qu'aux exemples biaisés, en faisant l'hypothèse qu'un exemple biaisé donnera un haut taux de confiance pour la bonne réponse. La méthode que nous présentons dans cet article se rapproche de RUBi dans l'idée de pondérer plus fortement les exemples non-biaisés pendant l'apprentissage, cependant, là où RUBi applique cette pondération au niveau de la rétro-propagation, ImRAN joue sur la distribution du jeu de données.

Une autre catégorie d'approches a cherché à modifier directement les images des jeux de données en entrée afin d'en retirer les biais à l'échelle des pixels [1, 7, 9, 18, 20, 23]. Toutefois, cela présente une limitation majeure due au fait que retirer certains biais peut être complexe, voire impossible, par exemple lorsque le biais relève de la texture [9] ou bien du genre des personnes [12]. Pour contourner cette limitation, [16] propose de manipuler les biais à l'échelle des représentations et de croiser les biais des exemples entre eux afin qu'ils ne soient plus prédictifs des différentes classes.

Pour finir, un autre type d'approches vise à changer directement la distribution des données afin d'augmenter la proportion de données non-biaisées comparativement aux données biaisées. Cela a pour effet de réduire la prédictivité des biais vis-à-vis des classes, supprimant ainsi les biais dans la nouvelle distribution obtenue. C'est ce que propose la méthode REPAIR [17], qui attribue à chaque exemple du jeu de données un poids. Ce poids est ensuite appris via rétro-propagation dans l'objectif de réduire la probabilité de tirage des exemples avec des biais. Cependant, REPAIR demande d'avoir des informations expertes sur les biais.

Dans cet article, nous proposons une méthode qui change activement la distribution des données d'apprentissage pour tendre à augmenter la proportion d'exemples non-biaisés. De plus, notre méthode ne demande pas de connaissance experte sur les biais présents.

3 Méthode proposée

Nous nous plaçons dans le cadre d'un jeu de données biaisé, c'est-à-dire qu'il y est plus facile de reconnaître les classes dans le jeu d'apprentissage via la détection d'un élément (appelé biais) qui est non pertinent pour la classification sur



FIGURE 1 – Exemples d’images du jeu de données *Biased* MNIST dans lequel la couleur de fond est corrélée avec la classe.

le jeu de test. Le jeu d’apprentissage n’est ainsi pas représentatif de celui du jeu de test. Pour pallier ce problème, nous proposons la méthode *Image Representation Avoiding Naive learning* (IMRAN) qui cherche à modifier l’échantillonnage des données dans le jeu d’entraînement pour essayer de la rendre plus similaire à celui du jeu de test, sans pour autant modifier l’algorithme d’apprentissage.

Le principe consiste à surreprésenter les exemples mal classifiés par le modèle dans le jeu d’apprentissage. En effet, dans le cas d’un jeu d’apprentissage biaisé, le modèle tend à apprendre le biais, ce qui implique que les données non-biaisées finissent généralement mal classifiées. La proportion de ces exemples est alors augmentée dans le jeu de données, permettant de réduire le biais dans le jeu d’apprentissage et donc le risque pour le modèle de l’apprendre. L’avantage de cette méthode par rapport à l’existant est double. Premièrement, nous n’avons pas besoin d’information ou d’annotation du biais, la détection se faisant uniquement sur la performance de classification de chaque exemple. Deuxièmement, si le jeu de données n’est pas biaisé, la remise des exemples mal classifiés ne devrait pas introduire de biais et donc affecter de façon de manière significative les performances issues de l’apprentissage.

Algorithm 1 Boucle principale d’apprentissage.

\mathcal{D} : jeu de données initial.
 \mathcal{D}'_i : jeu de données augmenté au pas de temps i .
 $n \in \mathbb{N}$: nombre d’époques pour l’apprentissage.

```

1:  $i \leftarrow 0$ 
2:  $\mathcal{D}'_i \leftarrow \mathcal{D}$ 
3: while  $i < n$  do
4:    $\text{learn}(\mathcal{D}'_i)$ 
5:    $\mathcal{D}'_i \leftarrow \text{duplique}(\mathcal{D}, \mathcal{D}'_i)$ 
6:    $i \leftarrow i + 1$ 
7: end while

```

La boucle d’apprentissage est décrite dans l’Algorithme 1. À chaque itération, la fonction $\text{duplique}(\mathcal{D}, \mathcal{D}'_{i-1})$ génère un nouveau jeu de données \mathcal{D}'_i à partir de la base de données originale \mathcal{D} , du nombre d’erreurs pour chaque duplication, et du nombre total de duplications pour chaque élément, du jeu de données \mathcal{D}'_i issu de la fonction duplique au pas de temps précédent. Plus précisément, la fonction $\text{duplique}(\mathcal{D}, \mathcal{D}'_{i-1})$, décrite dans l’Algorithme 2, fonctionne comme suit. Pour chaque entrée d du jeu de données originel \mathcal{D} , on retrouve l’ensemble des augmentations de d , c’est-à-dire des copies bruitées ou non, dans le jeu de données \mathcal{D}'_{i-1} généré au pas de temps précédent $i - 1$ par duplique . On calcule ensuite le ratio



FIGURE 2 – À gauche une image de *Biased* MNIST sans bruit, à droite, avec application d’un bruit gaussien de moyenne 0 et de déviation standard 1.

d’erreurs de prédiction du réseau sur ces données, arrondi à l’entier supérieur. Enfin, on génère un nombre d’augmentations entre 1 et K , proportionnel au taux d’erreur de prédiction. La fonction augmente prend en paramètre un élément d , un nombre de copies à générer, et une déviation standard pour le bruit, et retourne un ensemble de copies, bruitées par un bruit gaussien de moyenne nulle et de déviation standard σ .

Algorithm 2 Fonction duplique : génère un dataset augmenté \mathcal{D}'_i comportant 1 à K copies de chaque entrée du dataset originel \mathcal{D} au pas de temps i .

\mathcal{D} : dataset initial.
 \mathcal{D}'_{i-1} : dataset augmenté au pas de temps $i - 1$.
 $K \in \mathbb{N}$: nombre maximum de duplications.
 $C \in]0; 1]$: paramètre de la moyenne glissante exponentielle. (Pas de lissage pour une valeur de 1)

```

1: function  $\text{DUPLIQUE}(\mathcal{D}, \mathcal{D}'_{i-1})$  :
2:    $\mathcal{D}'_i \leftarrow \emptyset$ 
3:   for all  $d \in \mathcal{D}$  do
4:      $A \leftarrow \text{getAugmentations}(d, \mathcal{D}'_{i-1})$ 
5:      $\text{tauxErreur} \leftarrow \text{getTauxErreur}(A)$ 
6:      $\text{nbCopies} \leftarrow \text{tauxErreur} \times (K - 1) + 1$ 
7:      $\text{nbCopies} \leftarrow \lceil \text{nbCopies} \times C + |A| \times (1 - C) \rceil$ 
8:      $\mathcal{D}'_i \leftarrow \mathcal{D}'_i \cup \text{augmente}(d, \text{nbCopies}, \sigma)$ 
9:   end for
10:  return  $\mathcal{D}'_i$ 
11: end function

```

Une inertie est appliquée à la variation du nombre de copies entre chaque itération. Ce lissage permet de limiter les changements trop brusques entre les jeux de données d’époques consécutives. Le nombre d’occurrence d’une entrée est calculé comme la moyenne glissante entre son ancienne valeur et la nouvelle, pondéré par C .

4 Expérimentations

4.1 Protocole expérimental

4.1.1 *Biased* MNIST

Le jeu de données *Biased* MNIST [4] est composé de 60000 images de taille 28×28 biaisées synthétiquement. Les images sont des chiffres écrits à la main en blanc sur fond noir issues du jeu de données MNIST [14]; chaque chiffre représentant une classe. Pour y ajouter synthétiquement un biais, chaque classe se voit attribuer une couleur unique, et le fond noir présent sur les images est remplacé par la couleur correspondant au label associé à l’image (voire Figure 1). Les images non-biaisées ont quant à elles un fond

ρ	Vanilla	LearnedMixIn	RUBi	ReBias	ImRAN (avec bruit)	ImRAN (sans bruit)
.999	10.4	12.1	13.7	22.7	44.2 ± 2.8	33.5 ± 4.1
.997	33.4	50.2	43.0	64.2	74.8 ± 2.2	69.4 ± 2.3
.995	72.1	78.2	90.4	76.0	82.7 ± 1.3	79.7 ± 1.0
.990	89.1	88.3	93.6	88.1	90.5 ± 0.5	90.5 ± 0.5
.100	99.2	54.6	99.3	99.3	99.0 ± 0.1	99.2 ± 0.1

TABLE 1 – **Résultats pour *Biased* MNIST.** *Accuracy* des différentes méthodes sur le jeu de données *Biased* MNIST, suivant le taux d’images biaisées ρ . Chaque valeur est la moyenne de 10 exécutions pour ImRAN et 3 exécutions pour les autres méthodes. La dernière colonne compare les performances entre ImRAN avec et sans bruit appliqué aux données dupliquées.

d’une couleur aléatoire parmi celle des autres classes. La proportion des images biaisées peut ainsi être contrôlée pour créer des variantes du jeu de données. Le fait que le biais ne modifie que la couleur du fond garantit que les caractéristiques prédictives des classes (i.e. les chiffres) soient identiques quel que soit la proportion choisie. De cette façon, la difficulté de la tâche de classification à partir des caractéristiques pertinentes reste la même. C’est l’identification et l’extraction de ces caractéristiques prédictives qui devient plus difficile quand la présence de biais augmente. Par ailleurs, le fait que le biais soit toujours localement au même endroit sur une large surface (i.e., le fond), et qu’il soit composé d’une caractéristique de bas niveau (i.e., la couleur), le rend simple à apprendre et donc dur à éviter.

La proportion des images ayant la couleur de leur fond associée à leur classe est contrôlée par le paramètre ρ . De cette façon, pour $\rho = 1$, toutes les images sont biaisées, alors qu’avec $\rho = 0.1$, les images ne sont pas du tout biaisées par la couleur, étant donné qu’il y a 10 classes. Pour nos expérimentations, nous avons utilisé les valeurs 0.999, 0.997, 0.995, 0.990, et 0.1 pour ρ sur le jeu d’entraînement, valeurs classiquement utilisées dans la littérature [3], à l’exception de 0.1 qui est utilisé ici pour comparer les méthodes sur un jeu de données non-biaisé. Par la suite, les modèles sont évalués sur un jeu de test non-biaisé, mais cependant modifié synthétiquement avec $\rho = 0.1$ pour avoir une couleur de fond autre que du noir.

Enfin, la séparation entre images biaisées et non-biaisées est faite avant de procéder à l’entraînement, et la proportion reste la même au sein de chaque classe, les exemples non-biaisés restant donc bien les mêmes à travers tout l’entraînement.

4.1.2 Hyperparamètres

Afin d’être comparable avec la littérature, dans nos expérimentations, nous avons utilisé la même architecture et les mêmes hyperparamètres que dans ReBias [3]. Cette méthode de l’état de l’art n’utilise pas directement l’information en lien avec les biais présents, ce qui est aussi le cas pour notre méthode, et a historiquement les meilleures performances sur *Biased* MNIST avec $\rho = 0.999$, qui représente la variante la plus difficile testée dans nos expérimentations.

L’architecture que nous avons utilisée dans nos expérimentations est composée d’un réseau de neurones convolutif avec 4 couches de convolutions, avec un noyau de taille 7, et respectivement 16, 32, 64, et 128 canaux. Chacune de ces

couches de convolution est suivie d’une couche de *batch normalization* [11], ainsi que d’une fonction d’activation ReLU. La sortie de la dernière couche de convolution passe ensuite dans une couche d’*average pooling* pour produire un vecteur de dimension 128, avant de finir par une couche linéaire complètement connectée avec une taille de sortie à 10. Le classifieur est entraîné pendant 80 époques avec l’optimiseur Adam [13] et un taux d’apprentissage commençant à 0.001 qui est par la suite divisé par 10 toutes les 20 époques. Les images sont normalisées sur les 3 canaux avec 0.5 comme moyenne et 0.5 comme déviation standard. Pour ce qui est des hyperparamètres propres à notre méthode, nous les avons optimisé indépendamment afin d’avoir une bonne performance en un temps raisonnable sur $\rho = 0.999$. Ainsi, on fixe C le coefficient de la moyenne exponentielle mouvante à 0.001. Pour le nombre maximum de duplications K , nous avons choisis d’utiliser une valeur de 1000. Enfin, on fixe σ la déviation standard du bruit ajouté sur les images à 1 par défaut (e.g., Figure 2).

4.2 Résultats

Les résultats visibles dans la Table 4.1 montrent les performances de ImRAN, ainsi que les résultats de quatre autres méthodes dont l’approche standard (Vanilla) qui utilise la même architecture de réseau de neurones mais n’utilise aucun procédé particulier pour limiter l’effet des biais. Les méthodes utilisées pour la comparaison sont LearnedMixIn [6], RUBi [5] et ReBias [3].

La première ligne contient les résultats des différentes méthodes pour $\rho = 0.999$, là où le biais est le plus fortement présent. Dans ce contexte, l’approche Vanilla n’obtient que 10.4% d’*accuracy* ce qui est comparable à un choix aléatoire parmi les 10 classes possibles. Cependant, on remarque que ImRAN performe mieux que n’importe quelle autre méthode que ce soit avec ou sans bruit, avec respectivement 44.2% et 33.5% d’*accuracy*. ImRAN double quasiment les performances de ReBias qui est la deuxième meilleure méthode pour cette valeur de ρ . Pour $\rho = 0.997$, on remarque que ImRAN est toujours la meilleure méthode avec une marge significative. Cela confirme l’efficacité de notre méthode dans un contexte où le jeu de données est fortement biaisé.

On observe que pour toutes les variantes biaisées du jeu de données ($\rho \neq 0.100$) ImRAN performe toujours mieux que Vanilla, LearnedMixIn, et ReBias. De plus, en la présence de biais, ImRAN, ainsi que RUBi, sont les seules méthodes à toujours avoir une *accuracy* supérieure à l’ap-

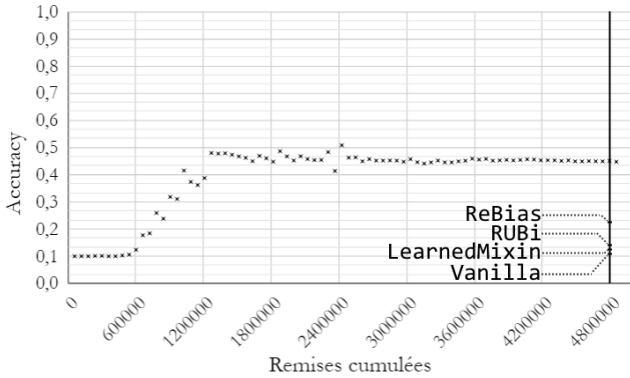


FIGURE 3 – Évolution de l’*accuracy* d’un apprentissage représentatif d’ImRAN en fonction du nombre d’exemples parcourus. La barre à 4 800 000 exemples représente 80 époques pour un jeu de données sans remise.

proche Vanilla. La méthode que nous proposons semble donc être un mécanisme efficace pour réduire l’apprentissage de biais lorsqu’il y en a. Néanmoins, sur des valeurs de ρ comme 0.995 et 0.990, qui sont des valeurs relativement faibles, l’approche RUBi obtient les meilleurs résultats, alors que ImRAN se place deuxième. Toutefois, RUBi a de mauvaises performances sur les valeurs élevées de ρ , or la proportion de biais représenté par ρ n’étant généralement pas connu, ImRAN semble être un choix plus versatile.

En parallèle de ces expérimentations, les méthodes ont été testées sur une version non-biaisée (i.e., $\rho = 0.1$) de MNIST, pour cela nous avons utilisé le code source officiel de ReBias qui propose aussi une implémentation de RUBi et LearnedMixIn. Le but est de s’assurer que les méthodes ne dégradent pas ou peu les performances de Vanilla lorsque le jeu de données n’a pas de biais. Les résultats, à la dernière ligne de la Table 4.1, montrent une *accuracy* très proche entre les différentes méthodes, à l’exception de LearnedMixIn qui performe faiblement. Cela montre que ImRAN ne dégrade quasiment pas les performances lorsque appliquée à un jeu de données sans biais.

Nous avons également testé notre modèle sans utilisation du bruitage des données (i.e., $\sigma = 0$), afin de quantifier l’influence de ce mécanisme dans les résultats. Ces derniers sont reportés à droite de la Table 4.1. On constate une augmentation significative des performances sur les valeurs élevées de ρ lorsque du bruit est ajouté aux augmentations. Pour les valeurs plus modérées de ρ , les résultats avec et sans bruit sont équivalents, voire légèrement meilleurs pour $\rho = 0.1$, ne dégradant ainsi pas du tout les performances comparées à Vanilla. Le bruit peut donc s’avérer légèrement pénalisant pour des données non-biaisées, là où il est fortement utile pour des données très biaisées.

Enfin, le fait d’augmenter la taille du jeu de données dans ImRAN, fait qu’à nombre d’époque égal, notre méthode apprend sur légèrement plus de données. La Figure 3 montre l’évolution de l’*accuracy* en fonction du nombre d’exemples cumulés pour un apprentissage de ImRAN avec pour $\rho = 0.999$. On peut voir que pour un

nombre d’exemples parcouru équivalent à 80 époques pour les autres approches, ImRAN a déjà convergé vers sa valeur finale. L’augmentation du nombre d’exemples vus n’a donc pas d’influence dans la hausse des performances obtenues par ImRAN.

5 Discussion et travaux futur

Le principal avantage de notre méthode, ImRAN, est qu’elle ne requiert ni de savoir si un biais est présent ou non, ni de disposer d’informations sur ce dernier comme son type (e.g., texture, couleur, forme) pour bien performer comme montré dans la Section 4.2. Comparée aux autres algorithmes de la littérature, notre méthode se montre très performante, allant jusqu’à presque doubler les performances de l’état de l’art dans les cas les plus extrêmes, et égaler les autres approches pour des valeurs de ρ plus faibles, à l’exception de RUBi. De surcroît, elle semble rester relativement efficace aussi bien dans ces cas extrêmes que sur des jeux de données à la proportion de biais plus modérée. Comme nous l’avons vu mentionné dans l’introduction, satisfaire ces deux cas est important puisque dans la plupart des jeux de données réels, il n’est pas possible de connaître à l’avance la présence de biais, ni dans quelles proportions. Cependant, ImRAN possède un léger défaut, le temps d’apprentissage est légèrement augmenté (entre 2% à 8% pour les variantes biaisées du jeu de données). Cela est dû à la duplication des éléments, et donc l’augmentation de la taille du jeu de données.

Ainsi, nos travaux futurs se focaliseront sur plusieurs points. Tout d’abord, l’expérimentation de notre approche sur plus de jeux de données afin de valider que la méthode est également efficace dans d’autres cas. En particulier avec des données plus réalistes et des biais plus complexes, comme par exemple avec le jeu de données bFFHQ qui contient des visages de personnes appartenant aux catégories jeune et âgée sachant que le genre de ces personnes est fortement corrélé avec leur tranche d’âge. Ensuite, la mise en place d’un mécanisme estimant la valeur de ρ pour ajuster la sélection des hyperparamètres, et notamment vis-à-vis du bruit. Enfin, nous souhaitons explorer des alternatives au bruit gaussien dans l’objectif de trouver une méthode générant une meilleure variance, et particulièrement au sein des exemples dupliqués.

6 Conclusion

Dans ce papier, nous avons présenté une version préliminaire de la méthode ImRAN, qui modifie activement la distribution de ses données d’apprentissage pour limiter l’apprentissage de représentations biaisées aussi bien sur des jeux de données peu ou grandement biaisés. Nous avons ensuite comparé ses résultats aux approches actuelles, puis discuté des résultats obtenus, ainsi que des potentielles pistes d’amélioration.

Remerciements

Cette recherche est supportée par le fond national de recherche luxembourgeois (FNR) :

IPBG2020/IS/14839977/C21, et a été réalisé à l'aide de GPU offerts par la société NVIDIA. Nous remercions sincèrement ces soutiens.

Références

- [1] Vedika Agarwal, Rakshith Shetty, and Mario Fritz. Towards causal vqa : Revealing and reducing spurious correlations by invariant and covariant semantic editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9690–9698, 2020.
- [2] Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. Analyzing the behavior of visual question answering models. *arXiv preprint arXiv :1606.07356*, 2016.
- [3] Hyojin Bahng, Sanghyuk Chun, Sangdoon Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pages 528–539. PMLR, 2020.
- [4] Hyojin Bahng, Sanghyuk Chun, Sangdoon Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pages 528–539. PMLR, 2020.
- [5] Remi Cadene, Corentin Dancette, Matthieu Cord, and Devi Parikh. Rubi : Reducing unimodal biases for visual question answering. *Advances in neural information processing systems*, 32, 2019.
- [6] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out : Ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv :1909.03683*, 2019.
- [7] Luke Darlow, Stanisław Jastrzębski, and Amos Storkey. Latent adversarial debiasing : Mitigating collider bias in deep neural networks. *arXiv preprint arXiv :2011.11486*, 2020.
- [8] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11) :665–673, 2020. Publisher : Nature Publishing Group.
- [9] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture ; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv :1811.12231*, 2018.
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [11] Sergey Ioffe and Christian Szegedy. Batch normalization : Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.
- [12] Eungyeup Kim, Jihyeon Lee, and Jaegul Choo. Biaswap : Removing dataset bias with bias-tailored swapping augmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14992–15001, 2021.
- [13] Diederik P Kingma and Jimmy Ba. Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*, 2014.
- [14] Yann LeCun. The mnist database of handwritten digits. <http://yann.lecun.com/exdb/mnist/>, 1998.
- [15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553) :436–444, 2015.
- [16] Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. Learning debiased representation via disentangled feature augmentation. *Advances in Neural Information Processing Systems*, 34 :25123–25133, 2021.
- [17] Yi Li and Nuno Vasconcelos. Repair : Removing representation bias by dataset resampling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9572–9581, 2019.
- [18] Matthias Minderer, Olivier Bachem, Neil Houlsby, and Michael Tschannen. Automatic shortcut removal for self-supervised representation learning. In *International Conference on Machine Learning*, pages 6927–6937. PMLR, 2020.
- [19] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure : Debiasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33 :20673–20684, 2020.
- [20] Rakshith Shetty, Bernt Schiele, and Mario Fritz. Not Using the Car to See the Sidewalk—Quantifying and Controlling the Effects of Context in Classification and Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8218–8226, 2019.
- [21] Enzo Tartaglione, Carlo Alberto Barbano, and Marco Grangetto. End : Entangling and disentangling deep representations for bias correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13508–13517, 2021.
- [22] Haohan Wang, Zexue He, Zachary C. Lipton, and Eric P. Xing. Learning robust representations by projecting superficial statistics out. *arXiv preprint arXiv :1903.06256*, 2019.
- [23] Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. Balanced datasets are not enough : Estimating and mitigating gender bias in deep image representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5310–5319, 2019.
- [24] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31, 2018.

Algorithmic learning, a next step for AI. An application to arithmetic operations

Frédéric Armetta¹, Anthony Baccuet¹ and Mathieu Lefort¹

¹Univ Lyon, UCBL, CNRS, INSA Lyon
LIRIS, UMR5205, F-69622
Villeurbanne, France

Abstract

Deep learning achieved state of the art performances in multiple domains (image recognition, natural language processing, etc.) One of the next steps is to be able to learn algorithms, as a way to provide some new forms of generalization for AI systems. This is currently a hard and challenging problem as it involves algorithmic recurrence, memory management and combination of subtasks, which leads to trainability problems. We show that even a simple algorithm of multiplication manifests trainability problems for neural networks. In this article, we present an original training method applied to multi-digit multiplication learning, called Unrolling Algorithmic Training (UAT). To learn the global algorithm, we use additional supporting tasks consisting in the successive subtasks composing the global algorithm (in our case the 1-digit multiplications and the final addition). The global end-to-end and the subtasks learning are then balanced with an active learning mechanism. This multi-task learning allows to overcome the problem of trainability encountered when learning directly the global algorithmic task. Our interpretation is that the network is able to somehow combine the subtasks in order to learn the global task. Moreover, we show that the global algorithm can be bootstrapped, fine-tuned and is even resilient without retraining it from scratch when we vary the size of recurrence provided to the network.

Keywords

Algorithmic Machine Learning, Deep Learning, Active learning

1. Introduction


Since the first success of convolutional neural networks on image classification [1], deep learning methods have pushed forward the state of the art in multiple domains [2]. This flows in tasks of increasing complexity such as language translation [3] or game playing [4]. One of the next steps is algorithmic learning, including mathematical expression calculation, as it will open the way to learn any task that can be expressed in a Turing machine and so to provide greater autonomy to AI systems.

Algorithmic resolution through neural networks is still an emerging area of research. It is a challenging problem as it requires to memorize values for a long period of time, to learn inferences, to combine procedures, to extrapolate to unknown domains, etc. More fundamentally this needs to fill the gap between symbolic understanding and statistical learning. Neural Turing Machine [5] was proposed to learn end-to-end algorithmic procedures, such as list sorting. It

✉ frederic.armetta@univ-lyon1.fr (F. Armetta); anthony.baccuet@gmail.com (A. Baccuet);
mathieu.lefort@liris.cnrs.fr (M. Lefort)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

aims to reduce the trainability problem faced by RNN, which are Turing complete [6], by adding specific mechanisms such as a memory and differentiable ways to access it. However, it also suffers from the trainability problem itself, i.e. the learning procedure is very sensitive to the hyperparameters and/or to the initial values and may lack generality, so that the performances are hard to reproduce and inconstant [7]. These problems are due to the depth of the proposed architecture [8] but also to the intrinsic complexity of the operations to learn, especially the long-term dependencies between data or variables to manipulate.

To overcome this trainability problem for an algorithm learning, an alternative way is rather to decompose this algorithm into its successive steps so that the neural network will compute and learn iteratively by feeding back its previous output thanks to a handmade recurrence external to the model. This procedure was applied to a transformer-inspired architecture for learning some algorithmic procedures [9] and to a MLP for arithmetic operations [10, 9]. In this paper, we aim to address the trainability problem end-to-end.

We choose to focus and illustrate our proposal on arithmetic operations, especially the multi-digit multiplication algorithm, which requires multiple one-digit multiplications to add properly in order to get the final result. This computation involves many operations and variables in the calculation stream. The complexity for such an operation is also related to the propagation of the carry e.g. [11]. Thus, learning directly the end-to-end multiplication leads to poor performances [12]. To tackle this trainability problem, we propose to mix during learning both intermediate sub-tasks and the global end-to-end multiplication, weighted by an active learning strategy. This is somehow similar to a multi-task learning, using intermediate computation as relevant complementary tasks to help the network to learn the global multiplication. However, with our approach, the network will perform all these tasks without any layer dedicated to each one. Thus, the network will have to accommodate intermediate operations in order to learn the global multiplication. Another difference is that this accommodation has to follow the course of the algorithm, by sequentially connecting tasks and propagating intermediate values through recurrences. We will show that this training procedure can also be used for pre-training the network, as a kind of bootstrap, for an efficient fine-tuning relying on the algorithmic target only (the multiplication).

We present the related work in section 2. We show in this section that language models such as chatGPT are also highly concerned by algorithmic learning, and suffer from trainability problems in the same way. This can be observed when the query involves lengthy algorithmic reasoning, as for arithmetic operations. We detail the use case of multi-digit multiplications in section 3. The chosen task is the decimal multi-digit multiplication, that is one of the hardest of the four arithmetic operations as it implies a lot of steps. Then, the model on which our learning procedure is applied is detailed in section 4 and next the protocol and results are presented in section 5. We conclude and discuss the perspectives of our work in section 6.

2. State of the art

Over the years, multiple deep learning models have been proposed to solve different kinds of problems [2]. Convolutional neural networks [1] are able to classify images with performances sometimes overcoming the ones of humans in specific scenarios. Recurrent neural networks

with dedicated cells such as Long Short Term Memory (LSTM) [13] or Gated Recurrent Units (GRUs) [14] can deal with temporal data.

Deep learning models are universal function approximators [15] but they are often hard to train. For instance, in theory even a shallow architecture is sufficient, but in practice, the ability of deep networks to learn relevant representations from data is much better [16]. Moreover, RNNs are Turing complete [6], but also face a trainability problem [5]. This limitation tends to be more pronounced for complex tasks, especially when long-term inferences are needed. As an example, the error rate of an arithmetic operation is related to the number of carries for a MLP [11].

Some global strategies were proposed to make network training easier and more effective. Among them we can mention active learning strategies that consist in finding the right next example to improve the training progress [17, 18]. A common way to do it is curriculum learning where tasks are ordered by increasing complexity [19]. Another strategy is multi-task learning that consists in combining different tasks with similar objectives so that the network can better generalize and learn the underlying structure of data [20]. Even if our proposal appears similar to multi-task learning, in our model the tasks are taken on the same network and are of algorithmic nature.

Neural Turing Machine (NTM) [5] was specifically designed to learn algorithmic tasks. To limit the problem of long time dependency encountered by LSTM, it mimics some of the principles of a Turing Machine by introducing a memory and reading/writing mechanisms. Differentiable Neural Computer [21] improves these accesses to learn more complex tasks as some NLP problems of inference and reasoning. However, these models are difficult to train as they seem to be very sensitive to initial weights [7, 22]. The neural GPU architecture [8] shares similar ideas with NTM but uses convolutional GRU in order to obtain parallel computing. It is able to learn some algorithmic tasks such as binary addition and multiplication, sequence reversing, etc. Its main achievement is its ability to generalize the learning to inputs with longer size in testing, e.g. from 20-digit binary multiplication in learning up to 2000-digit ones in testing, without any error. However, this model completely fails to learn decimal multiplications [23]. This limitation may be related to carry propagation that is hard to train and appears more frequently with decimal coding of numbers and can be partially overcome by using curriculum learning (from binary to quaternary then to decimal) [24].

In [22], multiple algorithms (LSTM with or without attention, transformers) were tested on various mathematical inference tasks. The main objective was to propose a dataset and to compare the models, especially on the attentional aspect, so once again detailed performances are missing. One of the conclusions is that long-term dependencies are the harder to learn which can be compromising for the addition of multiple operands in specific cases. A transformer with operands given as variables in the text, achieves good performance except for subtraction and multiplication [25]. [26] solves mathematical equations, including differential ones, with Seq2Seq models. The originality lies in the equation being written as the prefix notation of its tree representation. Another approach proposes to solve arithmetic expressions by composing single-digit sub-tasks. This hierarchical combination is learned by reinforcement learning with curriculum [27]. The system manages to generalize to some extent to longer sequences, but the performance sometimes drops, especially for multiplication. Neural Arithmetic Logic Units [28] are cells that are able to perform arithmetic operations by introducing specific computation

modules such as log, exp, etc. The aim here is not to learn arithmetic *per se* but to provide dedicated cells able to extrapolate learning with computations that extends to unknown domains. A direct extension dealing also with negative inputs was proposed in [29]. Other derived models can compute arithmetic operations on real numbers [30]. As we saw, there is a huge variety of tasks that was explored in the literature, so that comparing different approaches is difficult.

Algorithmic reasoning is also part of the required material for Natural Language Processing. Thus, large Language Models which have been extensively developed in recent years, need to learn reasoning to answer appropriately. They can for instance learn to perform numerical reasoning when learning few examples in a few-shot setting (using a query such as "Q: What is 24 times 18?", taking GPT-J-6B as a base). It was shown that the performance is highly correlated with the frequency of the terms in the pretraining corpus [31]. When co-occurrence of terms is low, accuracy is also low. These observations underline the trainability problem and lack of generalisation to algorithms learning. As a consequence, performance decreases as the size of operands increases [32]. Moreover, while addition seems to be less impacted by the size of operands, performance collapses for the multiplication which requires more reasoning or algorithmic processing. GPT-3 is among the most popular and large language model, using about 500 billion tokens for learning [33]. ChatGPT which is fine-tuned from GPT-3 also suffers from the limitations of GPT-based models. These observations suggest that model reasoning skills for such models are still limited. They highly rely on the size of the corpus available and statistics. The corpus will never contain all the combinations of terms or parameters of algorithms we have to learn. A better approach is then to enhance the ability to learn algorithms, which would allow to make better predictions for unknown configurations.

In this article, we choose to focus on the multi-digit multiplication of two decimal numbers. This is a simple enough task to not mix different problems but in the meantime it is challenging as multiple models are unable to learn it properly. This difficulty arises from long-term dependency due to the carry propagation, but also and more generally from the inherent complexity of algorithms which involves many operations and variables in the calculation flow.

Moreover, some articles precisely measured performance on this task. [12] proposes a MLP to learn addition and multiplication either from visual inputs or numerical encoding. In both cases, the accuracy achieved for multiplication is poor (see section 5).

3. Problem statement

In this article we will consider the decimal multi-digit multiplication. Formally, let n be the (maximum) number of digits of any of the two operands. The multiplication of these two operands leads to $n + 1$ sub-tasks: n single-digit multiplications and 1 final addition of the partial multiplications (see figure 1 for an example). In our data representation, each operation will correspond to two lines of computation: one for the carries and one for the result. The maximum size of any intermediate operation is $N = 2n$ (this maximal length will be obtained for the final addition with a carry generated at the most significant digit position). All the operands will be padded with zero digits to match a N size, yet there will be exactly $n + 1$ intermediate operations even if the first operand has less than n digits.

0023	(1)
×0048	(2)

0012	(3)
0184	(4)
0010	(5)
+0920	(6)

0110	(7)
1104	(8)

Figure 1: Example of the representation of a multiplication of two 2-digit operands. Note that the signs (+ and ×) and lines are only shown for clarity. Lines (1) and (2) are the two operands. Lines (3) and (4) (respectively (5) and (6)) represent the carries and result of the 1-digit multiplication of 8 (respectively 4) by 23. Lines (7) and (8) correspond to the carries and result of the addition of lines (4) and (6), which is also the final result of the global multiplication of 48 by 23, i.e. 1104.

4. Model

Our model is very similar to networks used for natural language processing. We choose a recurrent network in order to provide a potential for algorithm unrolling. We use an agnostic encoding for digits (no binary encoding) in order to prevent any facilities associated with the selected problem and focus on the algorithmic concerns.

4.1. Data representation

Each digit d is represented by a 10-dimension vector with a one-hot encoding, i.e. $(\delta_{di})_{i \in \{0, \dots, 9\}}$. This vector can also have two other values. First, a null vector represents an empty value in the input to form empty lines (see table 1). Note that it will also be used as the starting character for the decoder presented below. Second, a one vector corresponds to the end of the line either when reading or writing the data.

4.2. Model architecture

We use a Seq2Seq model as proposed in [34] (see figure 2). It is composed of an encoder that will sequentially read the data and recurrently embed it in some hidden states. From it, a decoder will iteratively produce a sequence from previously outputted characters, beginning with some predetermined code, until an ending character is written. Both encoder and decoder are implemented with an LSTM neural network model. During training, we used teacher forcing, i.e. the next character is not produced from the previous output but from the ground truth. The current output is then compared to the expected one and the error is backpropagated through the decoder and the encoder.

A headband allows to read and write successive lines two at a time, digit by digit. The digits, encoded in one-hot vectors, are then read from right to left, with the `<eol>` vector at the end of each line. Similarly, encoded one-hot vectors are written at the outputs. Both the inputs of the encoder and output of the decoder are 2×10 in size. Each task can then be encoded individually in order to infer the ground truth value (see below).

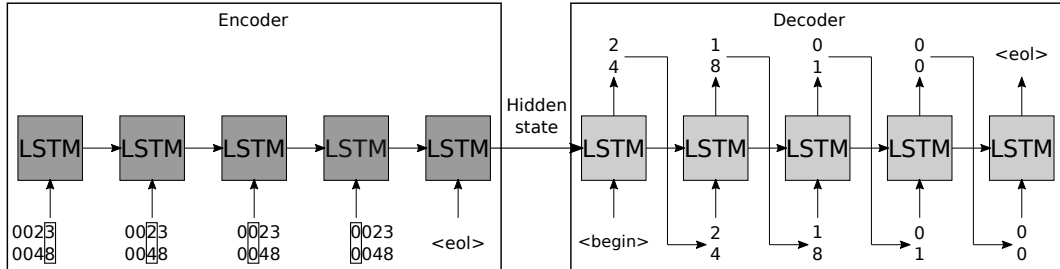


Figure 2: Seq2Seq architecture. The encoder receives successively the two digits (here boxed) of two lines (here the operands 23 and 48) from right to left. From the encoder embedding, the decoder recurrently produces the digits, from right to left, of the two next lines (here the carry and result of the first intermediate multiplication 8×23). In practice each digit is encoded as a one-hot vector (see section 4.1).

4.3. Tasks definition

The multiplication algorithm involves sub-tasks that can be submitted to the model (see table 1). The encoder records the formulated task, the decoder outputs the target value. The multiplication algorithm can also be inferred end-to-end in the same way. As formalized in section 3, the multiplication operation of two n -digit numbers can be decomposed into n single-digit multiplications and 1 final addition. These $n+1$ operations and the global end-to-end multiplication (ie computing the final result and carries from the two operands) compose the $n+2$ tasks that the network has to learn. For each task, the encoder reads the corresponding inputs and the decoder outputs the target value of the corresponding operation as illustrated in table 1. So that the network can differentiate the task of the first single digit multiplication and the end-to-end multiplication, which both require only the two operands, in the latter case we add $2n$ empty lines (corresponding to unfilled intermediate results lines).

Task	Encoder inputs	Decoder output
st1	(1//2)	(3//4)
st2	(1//2) (3//4)	(5//6)
st3	(1//2) (3//4) (5//6)	(7//8)
end-to-end	(1//2) (empty line * 4)	(7//8)

Table 1

Sub-tasks and end-to-end inferences associated with a 4-digit multiplication (see numbered lines in figure 1, lines are read and written two at a time ("//"))

4.4. Active learning mechanism

We have so far described several tasks. One can choose to select a bundle of tasks to train concurrently on the model. In this article, different settings are compared (sub-tasks only, all the tasks, end-to-end only) for training.

In order to balance the learning effort between the chosen bundle tasks, in all cases we use the same active learning mechanism as proposed in [10]. It consists in measuring the error rate, denoted err_{task} , of each involved task in the training dataset at the end of each training epoch. For the next epoch, the training dataset is constructed by randomly picking up examples from a global fixed set of examples so that each task is present with a fraction $F_{task} = \lambda \frac{err_t}{\sum_{ta \in taskList} err_{ta}} + (1 - \lambda) \frac{1}{card(taskList)}$ where λ is an hyperparameter and $taskList$ is the list of all tasks involved. The general idea is that the more difficult is a task to learn (first term), the more it is present in the next epoch with a lower bound depending on the λ value.

5. Experiments

5.1. Protocol

For training, 100000 unique couples of operands are generated. At each epoch, in order to update the dataset, we generate and attribute an operation to each of the couple so that the global distribution between operations matches the one decided by the active learning mechanism presented in section 4.4.

For the validation and test datasets, we use respectively 1000 and 10000 additional unique couples of operands. The size of the latent space of the encoder is set to 500 and the active learning parameter λ to 0.5. The model is trained using the ADAM optimizer with a learning rate of 10^{-4} and a batch size of 10. All the results presented in the next sections are averaged over 4 runs that learned during 500 epochs.

5.2. Results

	test = end-to-end		
	3 × 3 (6 output digits)	restricted 4 × 4 (7 output digits)	4 × 4 (8 output digits)
Hoshen et al.[12]	n.c	37.6 %	n.c.
UAT (train = end-to-end only)	4.05% (± 1.72%)	35.87% (± 28.10%)	92.42% (± 3.64%)
UAT (train = sub-tasks and end-to-end)	3.33% (± 1.32%)	4.51% (± 1.21%)	23.34% (± 14.69%)

Table 2

Error rate on the end-to-end multiplication task

In this section, we want to quantify the effect of our proposition to mix the learning of the sub-tasks with the global end-to-end multiplication. For the purpose of comparison, we

reproduce 4×4 multiplications, as presented in [12] and [10], with 4-digit inputs chosen so that the final outputs are restricted to 7-digit numbers.

5.2.1. Sub-tasks and end-to-end

The main purpose of this paper is to tackle the trainability problem encountered for the end-to-end multiplication task we choose as a first step towards general algorithmic learning. For the so considered bundle of tasks, sub-tasks and the end-to-end task are trained simultaneously on the same network following the active learning mechanism. We report in table 2 the performances for the end-to-end task, and compare with [12] which uses an MLP while we use a Seq2Seq. We also vary the scale and complexity of problems in order to show the improvements of our proposal.

Results clearly demonstrate the contribution of sub-tasks for the end-to-end task, lowering the error rate (from 35.87% to 4.51 % for the 4×4 restricted multiplications). This suggests that the model is able to self-organize and take advantage of sub-tasks, which facilitate the training, validating that our proposed training procedure is the key element of our model. While results are similar without the support of sub-tasks, i.e. both architectures face trainability problems, our training approach clearly outperforms [12]. The standard deviation reduction shows that stability of learning is also improved (with the exception of UAT end-to-end only 8 output digits that can be excluded from the comparison because of homogeneous but poor efficiency).

We report below other significant improvements for the most difficult problem presented (the 4×4 (8 output digits) problem) thanks to fine tuning and additional recurrences.

4x4 (8 output digits)	error rates
initial UAT train = sub-tasks + end-to-end	23.34% ($\pm 14.69\%$)
UAT fine tuning (pre-training = sub-tasks + end-to-end)	6.68% ($\pm 4.31\%$)
UAT fine tuning (pre-training = sub-tasks)	73.31% ($\pm 11.10\%$)

Table 3
Influence of fine tuning

free double lines	error rates
0	88.67% ($\pm 5.03\%$)
1	50.19% ($\pm 18.16\%$)
2	14.39% ($\pm 2.65\%$)
3	4.44% ($\pm 1.65\%$)
4	6.68% ($\pm 4.31\%$)
5	5.53% ($\pm 2.86\%$)
6	3.84% ($\pm 1.61\%$)
7	3.83% ($\pm 1.29\%$)

Table 4
Influence of the recurrence provided to the model when fine tuning (pre-training = sub-tasks + end-to-end, 4 free double lines)

5.2.2. End-to-end only (fine tuning)

To improve further the performance of our model, we propose to fine tune the end-to-end task only, during 500 additional epochs, for the hardest 4×4 problem. As we can see on table 3, when keeping the same recurrence to the network, the fine tuning leads to a significant improvement of the performance as the error drops from 23.34% to 6.68% for the 8 output digits problem. This result also shows that once the end-to-end task is bootstrapped (by simultaneous learning with subroutines), its performance can be further improved. On the contrary, table 3 shows that not including the end-to-end task in the pre-training bundle leads to trainability problems. This confirms that the model has to learn simultaneously the intermediate operations and the end-to-end one to overcome the trainability problem.

5.2.3. Adaptability

In our model, the input of the end-to-end multiplication is composed by some empty lines to match with the input size of the final addition subroutine (see table 1). These empty lines provide recurrence steps for the network (encoder). To test the influence of this factor, we ran multiple fine tuning of our model, setting each with a different number of empty lines (initial training done with 4 double lines).

We can observe on table 4 that the error rates tend to decrease monotonically with the number of free recurrences. This may sound logical as increasing the number of recurrences also increases the computational power of the model.

5.2.4. Active learning dynamic

In section 4.4, we introduced the active learning mechanism that we used to balance the learning effort between the different type of operations involved. In figure 3 is represented the evolution of their distribution in the training dataset. We can observe that it is relatively constant after some time and that the hardest task is the end-to-end one. However, one can notice that the second hardest task is the addition procedure and not any of the intermediate multiplication, whose mean error rate is even close to 0. This is surprising as in the literature the addition appears to be a simpler task to learn. This may be due to the fact that in this case the addition involves many operands.

5.2.5. Sub-tasks only

We will train the network only on the sub-tasks (ie excluding the end-to-end operation). To obtain the global result, we will provide the network with the successive intermediate tasks in the right order (with intermediate lines filled iteratively with previous outputs of the network) and consider only the output of the final addition as the global result. This method was also used in [9]. This task is easier to process (as the global operation is decomposed in its successive steps for its execution). Especially, we aim to estimate the effect of the recurrence available in our network knowing that [10] uses a network without recurrence.

Table 5 shows that our model reduces the error rate, in this context of execution. This shows that the recurrent model and data representation we use succeeds in capturing all the

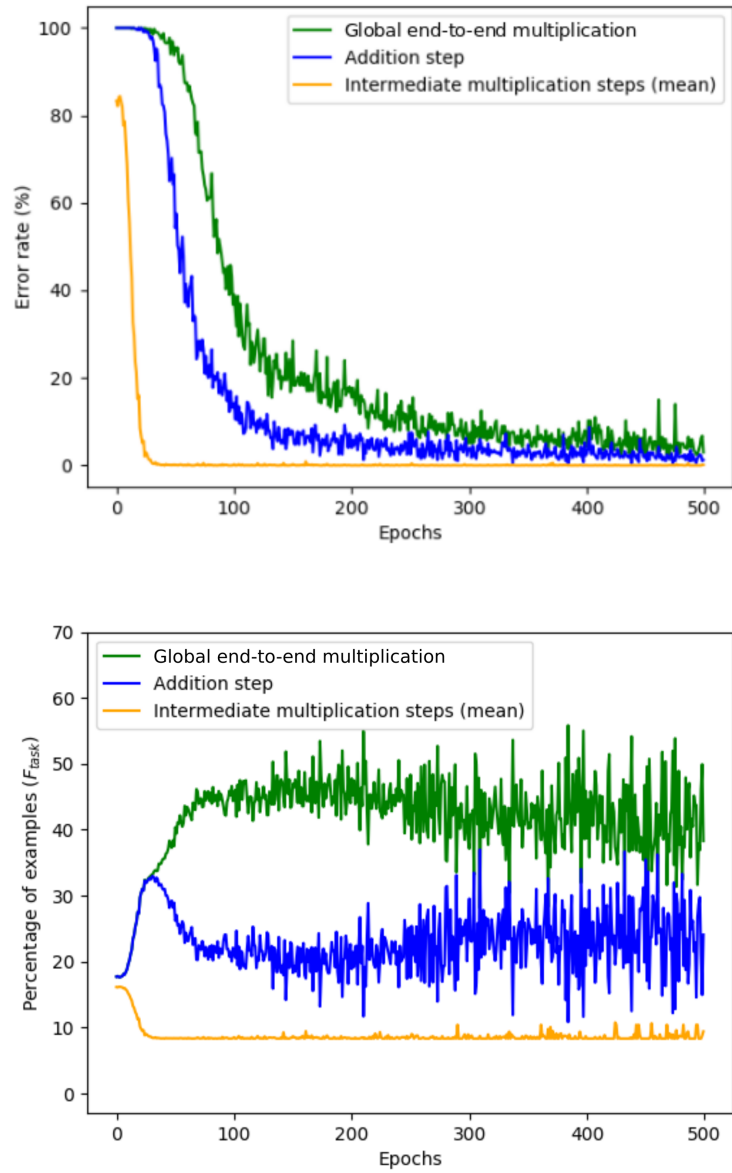


Figure 3: (Top) Evolution of the error rate for each kind of operation (estimated from the training dataset) used by the active learning. (Bottom) Resulting evolution of the distribution of the various tasks in the training dataset due to active learning

subroutines.

test = <i>handmade</i> recurrence	
restricted 4×4 (7 output digits)	
[10]	2 %
UAT	0.31% (\pm 0.11%)

Table 5

Error rate comparison between [Nollet et al., 2020] and UAT, when combining sub-tasks only thanks to an handmade recurrence

6. Conclusion and perspectives

Despite its multiple successes, deep learning architectures still struggle to learn algorithms, such as arithmetic operations. Algorithmic learning is highly expected as it would allow new classes of problems to be addressed, such as algorithmic generalisation. This is nevertheless a hard task, due to the complexity of the learning, especially the long-term dependencies, but also a trainability problem faced by multiple models. This limitation can as well be observed for language models whose performances come from the corpus available, and do not apply well to algorithmic inferences when variability in parameter values is large and usual generalisation do not apply as observed for arithmetic calculations, which underlines the need for new inference mechanisms.

In this article, we propose an original way to learn the end-to-end multi-digit multiplication of two (decimal) operands by guiding the model via the introduction of sub-tasks (or subroutines) concurrently with the targeted end-to-end task while applying an active learning strategy. While in the literature the addition appears to be a simple task to learn, multiplication is especially challenging. We show through analysis of our experiments that the multiplication can nevertheless be learned end-to-end, we measure improvements directly caused by UAT, the learning procedure we introduce. Learning the target end-to-end task after the sub-tasks can improve the results but the best way, by a large margin, is to mix all the tasks together during training in order to bootstrap the target end-to-end training task. Once bootstrapped, the targeted task can efficiently be fine-tuned alone. This process can be described as a new algorithmic transfer, i.e. the tasks complement and support each other to achieve the overall arithmetic task.

By fine tuning the model on the final end-to-end task we show an interesting additional property of our learning method. Once the end-to-end multiplication learned, the network can adapt to a number of recurrence different from the one it was first been trained for. This seems to indicate that the model is able not only to combine the sub-tasks to resolve the global end-to-end task but also to autonomously extract and to adapt some kind of high level algorithmic knowledge from it. Restricting the provided recurrences for the computing and maintaining accuracy looks like a constrained parallelization of the algorithmic task.

The main motivation for this work is not only the learning of the multiplication, but to provide a new method for alleviating the hard trainability problem that is observed for algorithms. This work raises multiple research questions. We want to investigate more precisely how the transfer from intermediate steps to the global task is achieved by the network. To overcome

the trainability problem encountered by the classical training procedure, we provide all the intermediate steps to the network during learning. A specific case that we want to study is to provide only some of the supporting tasks and see if the network can complement the unknown tasks by itself. For that we will investigate how the transfer from intermediate steps to the global task is achieved in the network. This understanding may give us a way to control the flow of interaction between supporting steps and the general dynamic of the algorithmic transfer, and thus get a more detailed understanding of how an AI could exploit and adapt its algorithmic knowledge in order to expand its capabilities.

Acknowledgment

This work was performed using HPC resources from GENCI-IDRIS and a GPU donated by @NVIDIA Corporation. We gratefully acknowledge this support.

References

- [1] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [2] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, *Deep learning*, volume 1, MIT press Cambridge, 2016.
- [3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., 2017, pp. 5998–6008. URL: <http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf>.
- [4] Y. Tian, J. Ma, Q. Gong, S. Sengupta, Z. Chen, J. Pinkerton, L. Zitnick, Elf opengo: An analysis and open reimplementation of alphazero, in: *International conference on machine learning*, PMLR, 2019, pp. 6244–6253.
- [5] A. Graves, G. Wayne, I. Danihelka, Neural turing machines, *CoRR abs/1410.5401* (2014). URL: <http://arxiv.org/abs/1410.5401>. arXiv: 1410 . 5401.
- [6] H. Siegelmann, E. Sontag, On the computational power of neural nets, *Journal of Computer and System Sciences* 50 (1995) 132 – 150. URL: <http://www.sciencedirect.com/science/article/pii/S0022000085710136>. doi:<https://doi.org/10.1006/jcss.1995.1013>.
- [7] M. Collier, J. Beel, Implementing neural turing machines, in: *Artificial Neural Networks and Machine Learning – ICANN 2018*, Springer International Publishing, Cham, 2018, pp. 94–104.
- [8] Ł. Kaiser, I. Sutskever, Neural gpus learn algorithms, *arXiv preprint arXiv:1511.08228* (2015).
- [9] Y. Yan, K. Swersky, D. Koutra, P. Ranganathan, M. Hashemi, Neural execution engines: Learning to execute subroutines, *CoRR abs/2006.08084* (2020). URL: <https://arxiv.org/abs/2006.08084>. arXiv: 2006 . 08084.
- [10] B. Nollet, M. Lefort, F. Armetta, Learning Arithmetic Operations With A Multistep Deep

- Learning, in: The International Joint Conference on Neural Networks (IJCNN), Glasgow, United Kingdom, 2020. URL: <https://hal.archives-ouvertes.fr/hal-02929738>.
- [11] S. Cho, J. Lim, C. Hickey, B.-T. Zhang, Problem difficulty in arithmetic cognition: Humans and connectionist models (2019).
 - [12] Y. Hoshen, S. Peleg, Visual learning of arithmetic operations, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16, AAAI Press, 2016, pp. 3733–3739. URL: <http://dl.acm.org/citation.cfm?id=3016387.3016429>.
 - [13] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation* 9 (1997) 1735–1780.
 - [14] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014. *arXiv:1406.1078*.
 - [15] G. Cybenko, Approximation by superpositions of a sigmoidal function, *Mathematics of control, signals and systems* 2 (1989) 303–314.
 - [16] J. Ba, R. Caruana, Do deep nets really need to be deep?, in: *Advances in neural information processing systems*, 2014, pp. 2654–2662.
 - [17] B. Settles, Active Learning Literature Survey, Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. URL: <http://axon.cs.byu.edu/~martinez/classes/778/Papers/settles.activelearning.pdf>.
 - [18] Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, A. A. Efros, Large-scale study of curiosity-driven learning, *arXiv preprint arXiv:1808.04355* (2018).
 - [19] Y. Bengio, J. Louradour, R. Collobert, J. Weston, Curriculum learning, in: *Proceedings of the 26th annual international conference on machine learning*, ACM, 2009, pp. 41–48.
 - [20] Y. Zhang, Q. Yang, A survey on multi-task learning, *arXiv preprint arXiv:1707.08114* (2017).
 - [21] A. Graves, G. Wayne, M. Reynolds, T. Harley, I. Danihelka, A. Grabska-Barwińska, S. G. Colmenarejo, E. Grefenstette, T. Ramalho, J. Agapiou, et al., Hybrid computing using a neural network with dynamic external memory, *Nature* 538 (2016) 471–476.
 - [22] D. Saxton, E. Grefenstette, F. Hill, P. Kohli, Analysing mathematical reasoning abilities of neural models, *arXiv preprint arXiv:1904.01557* (2019).
 - [23] K. Freivalds, R. Liepins, Improving the neural gpu architecture for algorithm learning, *arXiv preprint arXiv:1702.08727* (2017).
 - [24] E. Price, W. Zaremba, I. Sutskever, Extensions and limitations of the neural gpu, *arXiv preprint arXiv:1611.00736* (2016).
 - [25] A. Wangperawong, Attending to mathematical language with transformers, *arXiv preprint arXiv:1812.02825* (2018).
 - [26] G. Lample, F. Charton, Deep learning for symbolic mathematics, *arXiv preprint arXiv:1912.01412* (2019).
 - [27] K. Chen, Y. Dong, X. Qiu, Z. Chen, Neural arithmetic expression calculator, *CoRR abs/1809.08590* (2018). URL: <http://arxiv.org/abs/1809.08590>. *arXiv:1809.08590*.
 - [28] A. Trask, F. Hill, S. E. Reed, J. Rae, C. Dyer, P. Blunsom, Neural arithmetic logic units, in: *Advances in Neural Information Processing Systems* 31, Curran Associates, Inc., 2018, pp. 8035–8044. URL: <http://papers.nips.cc/paper/8027-neural-arithmetic-logic-units.pdf>.
 - [29] D. Schlör, M. Ring, A. Hotho, inalu: Improved neural arithmetic logic unit, *arXiv preprint*

arXiv:2003.07629 (2020).

- [30] A. Madsen, A. R. Johansen, Neural arithmetic units, in: International Conference on Learning Representations, 2020. URL: <https://openreview.net/forum?id=H1gNOeHKPS>.
- [31] Y. Razeghi, R. L. Logan IV, M. Gardner, S. Singh, Impact of pretraining term frequencies on few-shot numerical reasoning, in: Findings of the Association for Computational Linguistics: EMNLP 2022, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 840–854. URL: <https://aclanthology.org/2022.findings-emnlp.59>.
- [32] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, *Advances in neural information processing systems* 33 (2020) 1877–1901.
- [33] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al., Training language models to follow instructions with human feedback, arXiv preprint arXiv:2203.02155 (2022).
- [34] I. Sutskever, O. Vinyals, Q. V. Le, Sequence to sequence learning with neural networks, in: *Advances in Neural Information Processing Systems 27*, Curran Associates, Inc., 2014, pp. 3104–3112. URL: <http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf>.

Novelty detection for unsupervised continual learning in image sequences

Ruiqi Dai^{*§}, Mathieu Lefort^{†§}, Frédéric Armetta^{†§}, Mathieu Guillermin[‡] and Stefan Duffner^{*§}

^{*}Université de Lyon, INSA Lyon

[†]Université de Lyon, Université Claude Bernard Lyon 1

[‡]Université Catholique de Lyon, France

[§]LIRIS, UMR 5205 CNRS, France

Abstract—Recent works in the domain of deep learning for object recognition on common image classification benchmarks often address the representation learning problem under the assumption of i.i.d. input data. Although achieving satisfying results, this assumption seems not realistic when agents have to learn autonomously. An autonomous agent receives a continual visual flow of objects which is far from an i.i.d. distribution of objects. Moreover, agents have to construct their representations of the world and adapt to unknown environments, without relying on external sources of information such as labels that would be provided post-classification and are unavoidable when an over-segmentation is done. Then, in order to exploit the learned representation effectively for object recognition, a clear and meaningful relationship w.r.t. real object categories is required, which has been largely neglected in existing unsupervised algorithms.

In this paper, we propose a novelty detection method for continual and unsupervised object recognition, as an extension for the recent CURL model, which allows to moderate over-segmentation while preserving accuracy, in order to meet the requirements for autonomy. We experimentally validated our approach on two standard image classification benchmarks, MNIST and Fashion-MNIST, in this unsupervised and continual learning setting and improve the state of the art in terms of cluster purity, which is crucial for subsequent object recognition, since it facilitates clustering when information on ground truth labels is not available for free.

Index Terms—Continual learning, class-incremental learning, novelty Detection, object recognition, unsupervised learning.

I. INTRODUCTION

Let’s consider an agent interacting with objects in an unknown environment, continuously perceiving the objects through sensors. Being able to adapt to changes in the environment as well as continuously building a (visual) representation of objects of new classes while exploiting acquired knowledge is a crucial property for such a dynamic machine learning system. Classical deep learning models have shown excellent performance on image classification in an “off-line” setting, making the learning scenario comparatively simple in terms of representation learning since an iterative stochastic optimization of the loss function on i.i.d. data can be applied efficiently. However, when training data is not available all at once but sequentially, these models face some severe limitations. The literature on continual learning with neural networks [23], [24] partially responds to this issue, yet many of them are supervised (i. e. supervised learning) and in order to effectively

classify new observations of learned objects, extensive class labels are needed either at training time or after training to correctly attribute the numerous learned clusters to meaningful object categories. Even when addressed specifically, the lack of control over the way objects are introduced to the system lead to catastrophic forgetting phenomena for objects not seen for a long time, which is still a limitation to be overcome and considerably decreases the clustering performance. In fact, being able to learn new knowledge is an advantage coming from the plasticity of the network, but at the same time, the network should be stable enough to maintain the acquired knowledge, according to the stability-plasticity dilemma. A further limitation that we could identify, in the case of online and unsupervised learning, is the tendency to oversegment categories into many additional clusters [4], which makes the grouping of clusters inefficient during evaluation. Indeed, even if the clustering is done following an unsupervised approach, its evaluation is generally done thanks to ground truth labels assigned to the generated cluster, but hardly available online, which makes the methods ineffective when seeking autonomy. Moreover, we underline that over-segmentation facilitates the achievement of good accuracy while paradoxically reducing the autonomy of the system.

In this paper, we address the problem of unsupervised class-incremental representation learning for object recognition, in which objects are observed one after the other for a single period of time without storing any images in the long term. We will name it as *class-incremental* learning in the following sections. We propose a deep neural network model performing unsupervised class-incremental learning for visual object recognition, which is an extension of CURL [23], an approach dedicated to continual learning. Our main contribution lies in the integration of a new-class estimator based on statistics of the dynamics of the input sequence leveraging the temporal continuity of objects introduced and allowing to improve the detection of new objects. Furthermore, this guides the training with self-supervision by optimizing an adapted loss function.

The re-identification of learned objects that re-appear at different times during training could be addressed by using a classifier for category prediction which may require dedicated mechanisms. We choose to set aside this problem that can be addressed on its own as a second step. In this paper, we focus on the problem of performing an accurate automatic and

unsupervised novelty detection, in order to maintain clustering as close as possible to the original class labels provided by the dataset while keeping high accuracy.

II. RELATED WORK

A. Novelty Detection

In the literature, there are two types of tasks considering novelty detection: one-class classification approaches [19] that consider novelty detection as a binary classification problem of known/unknown, which are limited in scalability when there are numerous categories in the dataset; or multi-class approaches, also called open-set classification in the literature [1], [6]. For multi-class novelty detection, the estimation of the probability for unknown objects is a major challenge because existing classification approaches are usually based on a closed-world assumption [1], which estimates the probability distribution only over known categories, thus does not provide an appropriate estimation of the uncertainty when it comes to an unknown object. As a result, the model may wrongly “activate” an existing category with high confidence [16], this creates calibration problems in commonly used classification approaches using the softmax function. Some proposed approaches re-calibrate softmax, for example, ODIN [15] or G-OpenMax [5], [6]. Others use ensembles of deep learning models to predict uncertainty [12]; or treat this issue with a probabilistic approach based on the likelihood ratio between the inlier distribution and background knowledge [25], [28]. However, in continual learning, it is much more challenging to have a precise estimate on the background statistics for this sort of calibration.

B. Unsupervised object recognition

Different approaches for unsupervised image classification have been proposed in the literature, contrary to continual learning, common “off-line” approaches assume that training data are i.i.d [8], [30]. It is usually necessary to present the entire dataset several times in random order during training to ensure convergence and optimal performance. Recent advances in this domain make use of deep neural networks, in particular generative models [2], [7], [10] like Variational Auto-Encoders (VAE) [10], [31] and Generative Adversarial Networks (GAN) [2], [7]. These models learn to generate new data with the same statistics as a training set. Another family of unsupervised object recognition concerns clustering approaches like k-means [27] or DBSCAN [33]. These approaches work on the raw data without learning high-level features as deep neural networks, so applying them to images requires to use “hand-crafted” local feature extractor. Others include incremental clustering, for example SOINN [4] that learns the topology of dataset distribution, which will be introduced more in detail in section II-C. Common unsupervised object recognition algorithms have difficulties in determining the number of categories, as a result, they tend to mix similar categories, or reversely divide a category into several subcategories, requiring an extra effort of regrouping clusters during evaluation.

C. Continual learning

The literature in continual learning concerns two different scenarios: either solving a sequence of tasks/learning different datasets in the multi-task scenario, or learning new classes [18] incrementally in the single-task scenario. The state-of-the-art methods for continual learning with neural networks [14], [22] mainly focused on 3 categories of approaches:

- **Structural approaches** propose approaches that is network structure-related, for example, [26] proposes to dynamically add new nodes during training that modify the network structure with respect to the arrival of new tasks. Other algorithms [17] selectively activate parts of the network.
- **Regularization approaches** add a task-related regularization term to the cost function [34] to moderate changes in neurons involved in previous tasks while still allowing the network to learn new tasks. For example, in [11] the effect of catastrophic forgetting is contained by constraining the update of weights via a regularization term based on the Fisher information matrix extracted from previous tasks.
- **Experience replay** approaches try to alleviate catastrophic forgetting by regularly “replaying” past training examples [24], i. e. to train with both images from the current task/class and stored or generated samples [23]. The strategy of replay or the choice of examples to be stored is crucial to the model in terms of memory efficiency.

Most of these approaches are designed for *supervised* continual learning, showing strong dependence on accurate task identification and instance ground truth labels. Concerning unsupervised continual learning, the Self-Taught Associative Memory (STAM) [29] is an approach based on hierarchies of clustered image features that are continually learned by selecting centroids based on distance metrics. However, as opposed to neural network-based models, it is not clear to what extent the learned representation (i. e. hierarchical sets of image patches) can generalise to unseen object appearances and can be “re-used” for new object categories. Continual Unsupervised Representation Learning (CURL) [23] proposed a model based on VAE learning a Gaussian Mixture for different categories and alleviates catastrophic forgetting with generative replay, but it fails to automatically detect the number of clusters, thus requires to group clusters during evaluation. Therefore, *unsupervised* continual learning remains a challenging open research problem.

Common incremental clustering methods [3], [9] (such as BIRCH [35], incremental k-means [3]) are potential approaches to address incremental learning. Other approaches make use of topology learning [4]. Furoo et al. [4], for example, proposed a model called SOINN, for unsupervised and online topology learning for non-stationary data, with less memory consumption and allows for learning without knowing *a priori* the number of classes and the distribution of data. Yet in the domain of image sequences, to work effectively with more complex visual data streams, these approaches often

require either hand-crafted features or a pre-trained feature extraction model. Comparatively, approaches based on deep learning are more suitable due to the powerful representation capacity for visual data and images. Another limitation of these approaches is that they tend to create a large number of clusters and thus “over-segmenting” the original object classes [4]. This is also the case for some of the unsupervised continual learning approaches mentioned previously, cf. CURL [23]. This makes subsequent classification more complex as supervision is required afterwards to assign each cluster to the corresponding object class.

We propose a model that improves the clustering effectiveness by exploiting the constraints of the addressed scenario where objects are presented one after the other in the data stream. Our model is based on a previously proposed generative deep neural network [23] that we extended by modifying and improving the loss function and the new object class detection process.

III. PROPOSED APPROACH

Regarding the context and the class-incremental setting (cf. section I) of unsupervised and continual learning of object representation, we propose a generative neural network model extending CURL [23] that has been originally designed for the single-task sequential learning. We will first briefly outline this base model in section III-A, and then present our contributions (sections III-B and III-C).

A. Model and learning algorithm

CURL is a model that learns robust representations for different classes in a continuous manner based on a derivative of Variational Auto-Encoder (VAE), as shown in Fig. 1. Concretely, the core of the model is a Variational Auto-Encoder (VAE) which allows to approximate the distribution of the latent variable with a Gaussian or component. CURL extends the VAE by dynamically introducing a new dedicated component, for each new outlier image. It alleviates the effect of catastrophic forgetting by continuously generating synthetic training examples of previously learnt classes.

The model optimizes a modified ELBO (Evidence Lower Bound) objective (maximizing the likelihood of the data), with input images x , categorical variable y (the index of the Gaussian component), latent variable z corresponding to the internal representation (formed by the GMM):

$$E(x) = \sum_{n=1}^K q(y = k|x) \left[\log p(x|\tilde{z}^{(k)}) - KL((q(z|x, y = k)||p(z|y = k))) - KL(q(y|x)||p(y)) \right], \quad (1)$$

where $q(y = k|x)$ represents the component posterior, computed by a dense layer with softmax, marked as yellow nodes in Fig. 1, $\tilde{z}^{(k)} \sim q(z|x, y = k)$ is the latent code sampled from the k th Gaussian component each modelled by a dense layer of latent encoder head, $\log p(x|\tilde{z}^{(k)})$ corresponds to the component-wise reconstruction loss of input images,

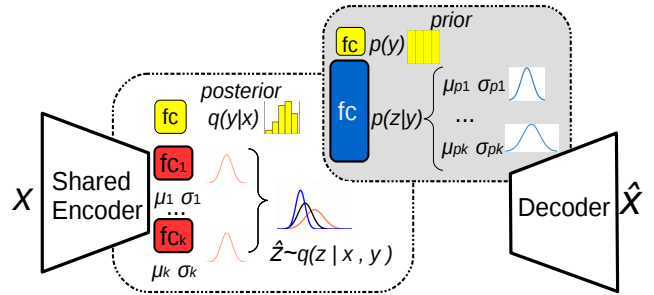


Fig. 1. The neural architecture of CURL: a Variational Auto-Encoder, X representing the input images, y the category variable. The encoder maps the input images to a shared representation for all the categories. Its output is used as the input of a fully-connected (fc, in yellow) softmax layer to estimate the object category $q(y|x)$; and updates the parameters μ_k, σ_k of the corresponding component(s) k . Posterior latent variable distribution $q(z|x, y)$ is approximated with component-specific latent encoders (a GMM). Also, the prior $p(z|y)$ of latent variable z follows a Gaussian distribution. Then, the image \hat{X} is reconstructed from the resampled \hat{Z} using the decoder. For more details, refer to section III.

with reconstructed image \hat{X} at the output, $KL((q(z|x, y = k)||p(z|y = k)))$ is a Kullback-Leibler divergence acting as the component-wise regularizer and enforcing a (Gaussian) embedding and $KL(q(y|x)||p(y))$ is the categorical regularizer that ensures that classes are well balanced, approaching the assumed uniform prior distribution $p(z|y)$ of each category. By maximizing Eq. 1, the model learns to reconstruct the input images and at the same time, due to the two regularization terms, to cluster objects into different classes in the latent space z by dynamically assigning them to different components. Poorly modelled instances whose ELBO is inferior to a threshold are considered as possible new category candidates and are thus stored in a temporary buffer which, once filled, is used to create and initialize a new component in the model. For more details, refer to [23].

This base model gives promising clustering performance. However, a major limitation is that the number of clusters resulting from the model does not allow to stay close to the original number of classes. In fact, it tends to create excessive clusters that therefore do not reflect the actual distribution of object categories. The number of introduced clusters is expected to stay close to the ground truth distribution, to facilitate eventually the categorization during evaluation.

Due to the fact that objects are presented sequentially for a certain amount of time, we consider that continuity is present in the perception of an autonomous agent evolving in a continual environment. Thus, the purpose of this study relies on measuring the additional value allowed by considering such a hypothesis on the accuracy of the system, training time and the need to keep under control the number of introduced components.

We propose two modifications of the original CURL model: a new category detection process (section III-B) that guides the learning with self-supervision optimizing a modified loss function (section III-C).

B. Detection of new classes

We hereby consider the case where the agent perceives objects class by class in the environment, not in a completely random way but in a class incremental way as it is mentioned in section I. In this paper, we choose to focus on novelty detection and improvements that can be achieved through the use of the continuity of perceived objects hypothesis, illustrated by the continuity in classes presented to the system. This shows the potential for such an approach, which is generally not exploited in machine learning, to study later the resilience of the process in a more noisy environment.

In this context, our contribution consists in the automatic detection of new classes by integrating an adaptive change detection algorithm, the Page-Hinckley test [21] applied to ELBO likelihood, a common approach applied in the domain of concept drift detection to detect abrupt changes in sequential input data. Formally, let $x_t \in \mathcal{X} = \{x_0, \dots, x_T\}$ be the examples presented in sequence of input training examples. In accordance with CURL, in our model, poorly modelled examples, are considered as new category candidates, i.e. for which the *unsupervised* ELBO objective $E(x)$ (Eq. 1) is below a threshold θ , since the *unsupervised* ELBO objective $E(x)$ marginalizes over all the existing categories which might reduce false-positive new category detection that corresponds to a category learned in the past instead of a new one. In our model, we apply the Page-Hinckley test that computes the decision function $g(t)$ for each new arriving example. We compare ELBO objective $E(x)$ with a threshold θ , noted by H the Heaviside step function that will equal to 1 if $E(x)$ is smaller than a threshold θ (implying an outlier). It is smoothed by a running average noted by $p_n(t)$, counting the average times that the outliers occur. We adopted a variant of the Page-Hinckley test as defined in [20], with N being the number of samples the agent has seen since the previous category change, and v being the tolerated change for each step:

$$g(t) = \max(0, g(t-1) + p_n(t) - \mu_{p_n}(t) - v) \quad (2)$$

$$\mu_{p_n}(t) = \frac{(N-1)}{N} \mu_{p_n}(t-1) + \frac{1}{N} p_n(t) \quad (3)$$

$$p_n(t) = \alpha * p_n(t-1) + (1 - \alpha) * H(\theta - E(x_t)) . \quad (4)$$

If $g(t)$ is greater than a threshold θ_n , then a new category is detected, i.e. a Gaussian is added to the GMM in the VAE and we reinitialize $g(t)$ to 0. Contrary to CURL that might be affected by noise in the ELBO loss, under the hypothesis of temporal continuity, our proposal of detecting new categories by Eq. 2-Eq. 4. helps to smooth these fluctuations and to obtain a cleaner supervision signal in the presence of outliers and alleviate category "over-segmentation".

Another modification of CURL in our model is that we propose for the original CURL model concerns the usage of the buffer storing recent examples in the incoming data stream. In our model since the proposed Page-Hinckley test detects abrupt changes, once a category change is captured, the buffer is filled with all the following instances in the sequence until reaching its maximum size n . However, the examples in

the (unfilled) buffer are not used for training immediately to prevent over-fitting resulting from too few training instances and to ensure having enough observations for each object class. Once the buffer is full, the training of the new class is initiated and the buffer is released.

C. Loss function

We use self-supervision deduced from our new-category detection algorithm to adapt the loss function that is used for training the model. We propose to optimize a supervised version of the ELBO objective function $E_{sup}(x)$ that CURL [23] originally used for a supervised baseline comparison of their algorithm. However, we integrate it differently in our approach. That is, we create an internal supervision signal $y_m \in \mathbb{N}$ based on the detection of new classes for training. $y_m \in \mathbb{N}$ that corresponds to the class of the instance determined by our model. Note that our proposed approach is still completely unsupervised as no ground truth labels are used. More specifically, y_m is incremented if the presence of the new, unseen object class is detected and maintained constant otherwise.

$$y_m = \begin{cases} y_m + 1, & \text{if } g_t \geq \theta_n \\ y_m, & \text{otherwise.} \end{cases}$$

The objective is defined as:

$$E_{sup}(x) = \log q(y = y_m | x) + \log p(x | \tilde{z}^{y_m}, y = y_m) - KL(q(z|x, y = y_m) || p(z|y = y_m)) , \quad (5)$$

and we continue to use the same variable definition as in Eq. 1. where the first term trains a fully connected layer with softmax to predict the label, the second term minimises the auto-encoded reconstruction error and the last term again represents the Kullback-Leibler divergence between the variational posterior of z and its corresponding Gaussian prior distribution.

IV. EXPERIMENTS

A. Dataset

To compare our approach to the state of the art, we evaluated our model on two standard datasets: MNIST (images of handwritten digits from 0 to 9) [13] and Fashion-MNIST [32] (Zalando's images with the classes {T-shirt, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, ankle boot}). Both datasets contain objects from 10 classes, with 60000 images for training and 10000 images for testing. The size of images in both datasets is 28×28 . For each class, there are around 6000 images for training and around 1000 for the test set. During training we present images in a class by class order, from 0 to 9. It can be noticed that compared to the MNIST dataset, the Fashion-MNIST dataset is more complex. In the Fashion-MNIST dataset, images of different classes can be more similar (than in MNIST), for example, dresses resemble coats. The MNIST dataset, however, is a comparatively simple task for the reason of the well-alignment of digits in each category. And comparatively, objects in the Fashion-MNIST dataset are more diverse and more complex.

B. Experimental protocol

Our model is trained in a continuous way, as stated in section I, i. e. the data are presented to the model sequentially and class-by-class. Thus, each class is seen only once, but during training, each image of the current class can be presented several times until a new class is detected. For a fair comparison with CURL, as detailed in section IV-C, we preserved the model architecture and setting for generative replay from [23], meanwhile tuning the other parameters (the threshold for outliers) with respect to clustering performances, and most importantly, with regard to the optimum number of categories detected that approaches the true distribution of categories and results in the best clustering score of AMI/ARI while preserving the same performance (i. e. accuracy) in clustering. We also compared the clustering performance of our proposal with two incremental clustering algorithms, BIRCH [35] and incremental k-means [27]. The two mentioned incremental algorithms do not provide explicit feature extracting methods, to avoid retraining a neural network, we take the flattened image as input without extracting features. In addition, we compare our model with SOINN [4], which was originally designed for offline clustering, that we adapted in an online setting that presents objects class by class. As an ablation study, we have also tested two variants of our model: Ours w/o P-H test, where we perform a simple running average p_n while comparing ELBO objective with a threshold; Ours w/o p_n , where we apply the Page-Hinckley test directly on ELBO objective without running average p_n smoothing.

C. Hyperparameters

For both datasets, we fixed the neural network architecture and the learning rate to 10^{-3} while using the Adam optimizer. To compare our model with CURL, we use the same neural network structure as in [23]: a 4-layer MLP as encoder {1200, 600, 300, 150}, and a linear layer with 64 dimensions to compute the mean and variance for the 32-dimensional latent variable z . For the decoder a two-layer MLP {500, 500} was used. The total number of iterations is 100000 counting all the categories, for each category 10000 iterations, where at each iteration, the size of batch is 100. We applied the mechanism of generative replay in the same way as CURL, i. e. images of previous classes are generated at fixed intervals (every 10000 steps) and stored into a buffer. For the mixture generative strategy, we continue to use the one of CURL, that is to create a mixture between real images of the current category and generated images of other learned components. To this end, every two steps, a batch of generated images is mixed with the batch of real images for training. We suppose that images of a class are visible for at least 100 steps in both experiments and we use a buffer of size 100 that stores outlier candidates. For the value of θ , the threshold on the ELBO loss, we have chosen $\theta = -150$ for all the experiments on MNIST, resulting in the best accuracy. For Fashion-MNIST, we set $\theta = -300$ for CURL and our model without Page-Hinckley test, and $\theta = -190$ for our model. Concerning the Page-Hinckley test, we set $\alpha = 0.85$ for both datasets, and

$v = 0.3$ and $\theta_n = 1.5$ for parameters in the experiments with Page-Hinckley test applied on running average p_n ; $v = 55.0$ and $\theta_n = 1500.0$ for parameters in the experiments with Page-Hinckley test applied on *negative* unsupervised ELBO loss without p_n smoothing.

D. Evaluation measures

To evaluate the quality of the learned clustering, we used three standard metrics: the clustering accuracy assigning to each component its most represented class, for labelisation of each component on the test set in correspondence to classes and measuring the proportion of correctly classified instances, the Adjusted Mutual Information (AMI) and the Adjusted Random Index (ARI) computed between learned clustering prediction and that of the ground truth. AMI measures the mutual information between two assignments of partitions. ARI measures the similarity between two partitions by counting the difference of assignment of pairs of samples between two partitions. Both metrics are adjusted w. r. t. the chance to remove the bias induced by the inequality in the number of clusters in both partitions. All measures are in $[0, 1]$, where higher values are better.

The clustering accuracy gives a general idea about the classification performance if labels were available. However, it does not completely reflect the quality of the clustering. For example, let's consider the case where the algorithm creates a partition that correctly separates different classes, but creates many excessive clusters from the same class (over-segmentation). We need at least one ground-truth label per cluster to regroup them into correct classes, i. e. requiring supplementary effort on data annotation, which considerably decreases the level of autonomy of the algorithm in an unsupervised continual learning setting.

E. Results

The results on MNIST and Fashion-MNIST are shown in Table I and in Table II respectively. Note that one needs to choose the trade-off between optimizing the number of clusters, reaching better AMI/ARI scores, while detecting all the changes which allows high clustering accuracy. For MNIST, our model achieves a very good trade-off and creates fewer additional components, i. e. the closest to the real number of classes (10), and scores the highest in terms of AMI and ARI compared to CURL and SOINN. For Fashion-MNIST, our model outperforms CURL on the AMI and ARI measures, with a slightly inferior accuracy. But as shown in Table II, CURL creates 120 components exceeding by far the number of real categories in the dataset. This indicates that the clusters created by our model follow the true distribution of different categories and avoid over-segmentation.

In Fig. 2 and Fig. 3, we further show the confusion matrix between ground truth classes and clusters. We can observe that in our model, samples of the same class are represented principally by one cluster. On the contrary, the confusion matrix of CURL shows that CURL tends to separate samples of the same class into different clusters. We equally illustrate the

Model	accuracy	AMI	ARI	nb components
CURL [23]	0.822 \pm 0.0102	0.557 \pm 0.006	0.28 \pm 0.025	93.85 \pm 1.884
CURL supervised [23]	0.855 \pm 0.006	0.749 \pm 0.006	0.6997 \pm 0.012	10 \pm 0
SOINN [4]	0.925 \pm 0.0011	0.39 \pm 0.002	0.018 \pm 0.0008	1204 \pm 39.6
BIRCH [35]	0.3026 \pm 0.002	0.184 \pm 0.014	0.10 \pm 0.0113	10 \pm 0
Incram. k-means [27]	0.338 \pm 0.017	0.2545 \pm 0.013	0.124 \pm 0.013	10 \pm 0
Ours w/o P-H test	0.849 \pm 0.008	0.735 \pm 0.0102	0.685 \pm 0.015	22 \pm 1.07
Ours w/o p_n	0.854 \pm 0.005	0.748 \pm 0.00424	0.6996 \pm 0.0085	10.67 \pm 0.47
Ours	0.8537 \pm 0.006	0.746 \pm 0.0096	0.70 \pm 0.013	10 \pm 0

TABLE I

COMPARISON OF OUR METHOD WITH THE STATE OF THE ART ON THE MNIST (AVERAGE OVER 3 RUNS) FOR EACH METRIC MEAN \pm SD.

Model	accuracy	AMI	ARI	nb components
CURL [23]	0.686 \pm 0.013	0.445 \pm 0.004	0.137 \pm 0.002	120 \pm 0.0
CURL supervised [23]	0.654 \pm 0.007	0.57 \pm 0.006	0.4336 \pm 0.006	10 \pm 0
SOINN [4]	0.796 \pm 0.003	0.365 \pm 0.001	0.022 \pm 0.008	755 \pm 16.54
BIRCH [35]	0.328 \pm 0.023	0.286 \pm 0.019	0.124 \pm 0.0139	10 \pm 0
Incram. k-means [27]	0.404 \pm 0.004	0.38 \pm 0.0105	0.237 \pm 0.013	10 \pm 0
Ours w/o P-H test	0.644 \pm 0.009	0.537 \pm 0.015	0.415 \pm 0.022	64.6 \pm 9.5
Ours w/o p_n	0.65 \pm 0.0098	0.547 \pm 0.007	0.42 \pm 0.007	25.67 \pm 9.534
Ours	0.6526 \pm 0.0056	0.558 \pm 0.00856	0.442 \pm 0.0117	13.0 \pm 2.19

TABLE II

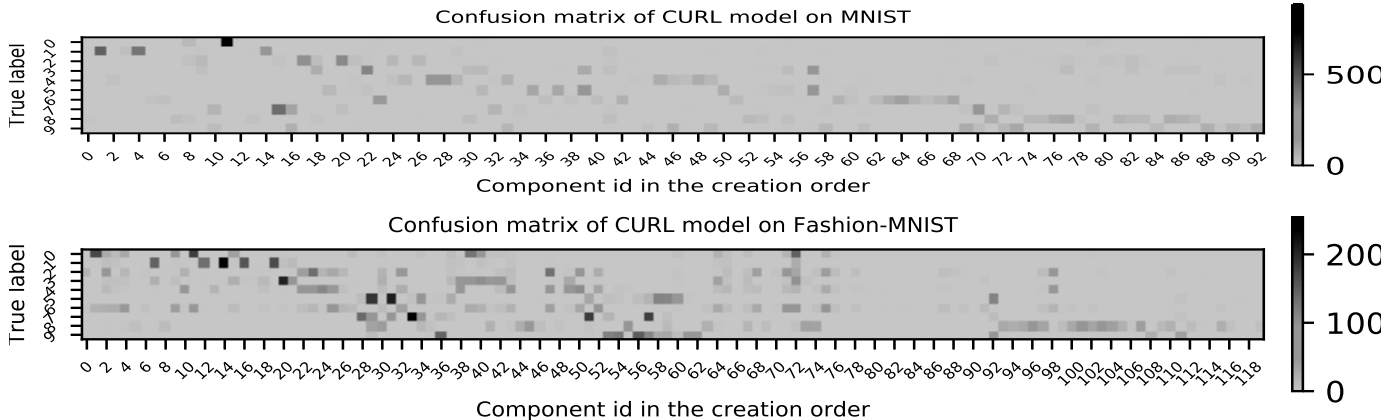
COMPARISON OF OUR METHOD WITH THE STATE OF THE ART ON FASHION-MNIST (AVERAGE OVER 3 RUNS) FOR EACH METRIC MEAN \pm SD.

Fig. 2. Confusion matrices between ground truth and predicted cluster components for CURL on the MNIST test set (above) and the Fashion-MNIST test set (below) (the darker a cell the more instances it represents).

2D t-SNE projection of the learned embedding vector of our model and CURL on the MNIST test set in Fig. 4. Different colors represent different categories according to the ground truth label. This not only shows that our approach reduces the phenomenon of over-segmentation in the clustering but also that the different clusters are more consistent with the real object classes. In addition, the clusters are overall more compact and better separated.

Finally, we explicitly studied the relationship between clustering accuracy and the amount of available annotated training data during evaluation, as shown in Fig. 5. We illustrated the variation of clustering accuracy, while using a limited number of examples on the test set to attribute the majority class to each component. Examples used for labeling were chosen at

random and with a permutation at each evaluation. Compared to CURL, our model can achieve its maximum accuracy with a very small amount of labelled examples during evaluation, while CURL requires much more examples. The over-segmentation clearly increases the requirement of annotated data during evaluation and may thus limit the classification performance in practical applications.

To validate the individual contributions of our method, we compared it to a variant of CURL using our loss (Eq. 5) supervised by the ground truth and with buffer, called "CURL supervised" in Tables I and II. These two experiments demonstrate the effectiveness of our new-category detection algorithm, since our model with Page-Hinckley test applied on p_n is capable of reaching a comparable performance in

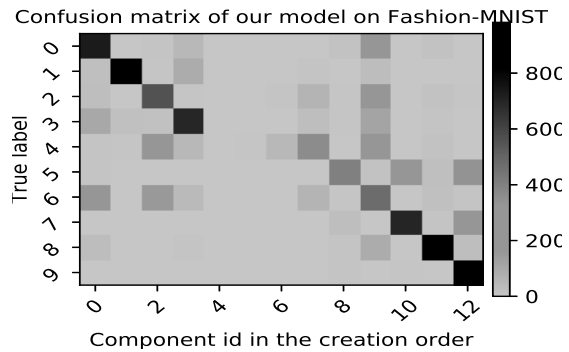
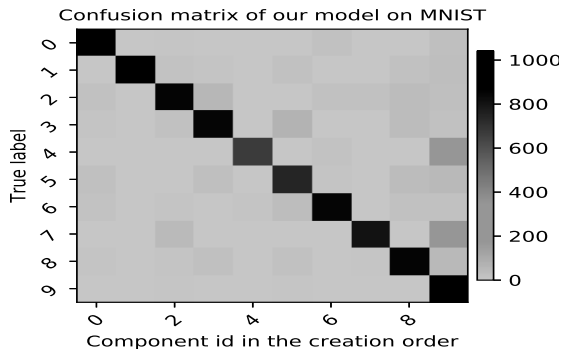


Fig. 3. Confusion matrices between ground truth and predicted cluster components for our complete approach on MNIST (left) and Fashion-MNIST (right) (the darker a cell the more instances it represents).

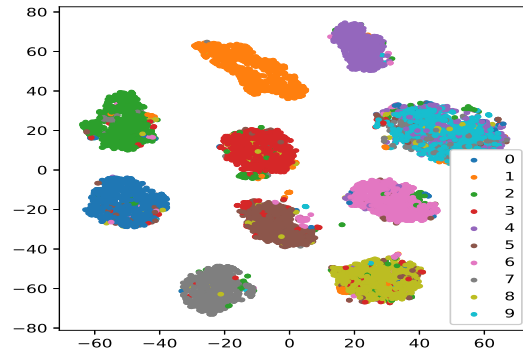
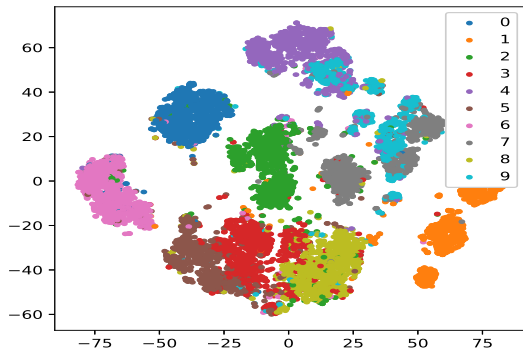


Fig. 4. 2D t-SNE projection of one run of CURL (left) and our model (right) on the MNIST test set: different colors represent different ground truth labels. There are about 93 clusters in CURL representing 10 categories, thus several "sub-clusters" for each category

terms of new category detection on MNIST with respect to supervision using the ground truth.

But both our model and CURL are outperformed by SOINN in terms of clustering accuracy. Only in terms of accuracy, SOINN performs better, which is not surprising given the excessive number of components (about 1200 on MNIST and about 755 nodes on Fashion-MNIST) reducing thus the probability of impure ground-truth clusters but at the same time needing much more additional supervision to label these clusters, as demonstrated in Fig. 5, one could observe that if we only use part of the test set to label components by their majority class, a drop in the clustering performance could be remarked from Fig. 5. The SOINN model converges the slowest compared to CURL and our model.

The results of BIRCH and incremental k-means are much below the performance of the other methods on both datasets showing a clear limitation of such classical incremental clustering algorithms in this context.

V. CONCLUSION AND DISCUSSION

Recent works have focused on creating an efficient neural network model for continual learning, as it is the case for CURL which is unsupervised and provides a generative replay mechanism while making use of a rich multivariate Gaussian Mixture Model. In this paper, we improved the new category detection process by moderating the number of components created for class categorization in order to stay close to the

real distribution. We consider that, for an autonomous agent, some continuity is present and images of its environment are not perceived in a totally random order. Thus, we proposed a completely unsupervised approach based on an extension of CURL, a VAE-based model [23], that takes advantage of continuity in the introduced object class and applying a supervised ELBO loss with self-supervision. To this end, we proposed to use the statistical Page-Hinckley test to improve the performance of new-class detection, and p_n a running average for each instance, to smooth fluctuations in the ELBO loss, leading to a robust class change detector. When compared to the baseline, our proposal allows to considerably reduce the introduction of additional clusters while keeping accuracy, which improves autonomy. Indeed, over-segmentation of clusters leads to further supervision for classification which is not always available online, or can only be done in a restrained way. This work appears as a first step and shows how unsupervised learning can take advantage of temporal continuity of objects perceived to better categorize objects online. Further work will study how this proposal behaves under increasing noise in the input sequences.

REFERENCES

- [1] Bendale, A., Boulton, T.E.: Towards open set deep networks. In: CVPR. pp. 1563–1572 (2016)
- [2] Bojanowski, P., Joulin, A., Lopez-Pas, D., Szlam, A.: Optimizing the latent space of generative networks. In: International Conference on Machine Learning. pp. 600–609 (2018)

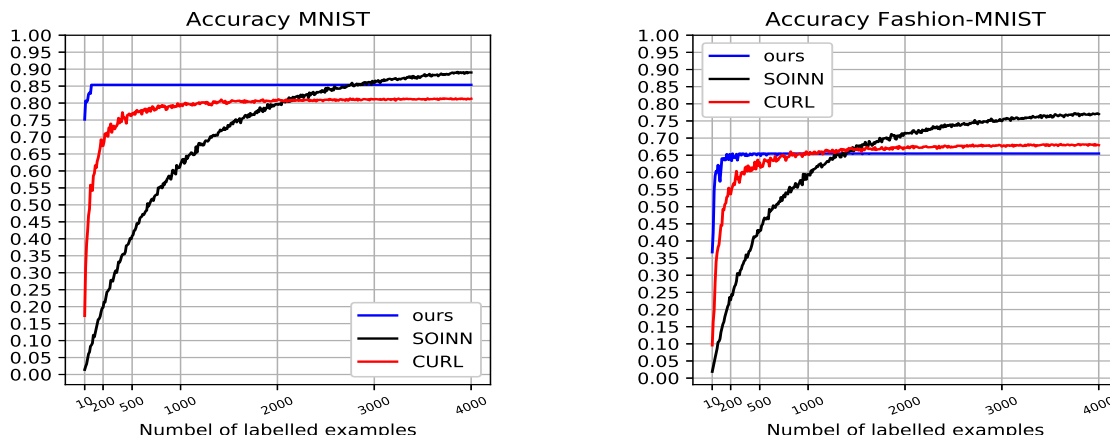


Fig. 5. Influence of the number of annotated examples used for labeling the test set on the accuracy, for MNIST (left) and Fashion-MNIST (right).

- [3] Dey, L., Chakraborty, S., Nagwani, N.K.: Performance comparison of incremental k-means and incremental DBSCAN algorithms. *International Journal of Computer Applications* **27**(11), 14–18 (2011)
- [4] Furoo, S., Hasegawa, O.: An incremental network for on-line unsupervised classification and topology learning. *Neural networks* **19**(1), 90–106 (2006)
- [5] Ge, Z., Demyanov, S., Chen, Z., Garnavi, R.: Generative openmax for multi-class open set classification. In: *BMVC* (2017)
- [6] Geng, C., Huang, S.J., Chen, S.: Recent advances in open set recognition: A survey. *IEEE transactions on pattern analysis and machine intelligence* (2020)
- [7] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: *NeurIPS*. pp. 2672–2680 (2014)
- [8] Goyal, S., Benjamin, P.: Object recognition using deep neural networks: A survey. *arXiv preprint arXiv:1412.3684* (2014)
- [9] Joshi, P., Kulkarni, P.: Incremental learning: Areas and methods—a survey. *International Journal of Data Mining & Knowledge Management Process* **2**(5), 43 (2012)
- [10] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. *stat* **1050**, 1 (2014)
- [11] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., et al.: Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* **114**(13), 3521–3526 (2017)
- [12] Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: *NeurIPS*. pp. 6402–6413 (2017)
- [13] LeCun, Y.: The MNIST database of handwritten digits. <http://yann.lecun.com/exdb/mnist/> (1998)
- [14] Lesort, T., Lomonaco, V., Stoian, A., Maltoni, D., Filliat, D., Díaz-Rodríguez, N.: Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges. *Information fusion* **58**, 52–68 (2020)
- [15] Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. In: *ICLR* (2018)
- [16] Liu, W., Wang, X., Owens, J.D., Li, Y.: Energy-based out-of-distribution detection. *arXiv preprint arXiv:2010.03759* (2020)
- [17] Mallya, A., Davis, D., Lazebnik, S.: Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In: *ECCV*. pp. 67–82 (2018)
- [18] Maltoni, D., Lomonaco, V.: Continuous learning in single-incremental-task scenarios. *Neural Networks* **116**, 56–73 (2019)
- [19] Markou, M., Singh, S.: Novelty detection: a review—part 1: statistical approaches. *Signal processing* **83**(12), 2481–2497 (2003)
- [20] Montiel, J., Read, J., Bifet, A., Abdesslem, T.: Scikit-multiflow: A multi-output streaming framework. *Journal of Machine Learning Research* **19**(72), 1–5 (2018), <http://jmlr.org/papers/v19/18-251.html>
- [21] Page, E.S.: Continuous inspection schemes. *Biometrika* **41**(1/2), 100–115 (1954)
- [22] Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., Wermter, S.: Continual lifelong learning with neural networks: A review. *Neural Networks* **113**, 54–71 (2019)
- [23] Rao, D., Visin, F., Rusu, A., Pascanu, R., Teh, Y.W., Hadsell, R.: Continual unsupervised representation learning. In: *Advances in Neural Information Processing Systems*. pp. 7645–7655 (2019)
- [24] Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: icarl: Incremental classifier and representation learning. In: *CVPR*. pp. 2001–2010 (2017)
- [25] Ren, J., Liu, P.J., Fertig, E., Snoek, J., Poplin, R., DePristo, M.A., Dillon, J.V., Lakshminarayanan, B.: Likelihood ratios for out-of-distribution detection. In: *NeurIPS* (2019)
- [26] Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R.: Progressive neural networks. *arXiv preprint arXiv:1606.04671* (2016)
- [27] Sculley, D.: Web-scale k-means clustering. In: *Proceedings of the 19th international conference on World wide web*. pp. 1177–1178 (2010)
- [28] Serrà, J., Alvarez, D., Gómez, V., Slizovskaia, O., Núñez, J.F., Luque, J.: Input complexity and out-of-distribution detection with likelihood-based generative models. In: *ICLR* (2020)
- [29] Smith, J., Taylor, C., Baer, S., Dovrolis, C.: Unsupervised progressive learning and the STAM architecture. In: *IJCAI* (2021)
- [30] Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: *Thirty-first AAAI conference on artificial intelligence* (2017)
- [31] Tolstikhin, I., Bousquet, O., Gelly, S., Schoelkopf, B.: Wasserstein auto-encoders. In: *ICLR* (2018)
- [32] Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747* (2017)
- [33] Xu, R., Wunsch, D.: Survey of clustering algorithms. *IEEE Transactions on neural networks* **16**(3), 645–678 (2005)
- [34] Zenke, F., Poole, B., Ganguli, S.: Continual learning through synaptic intelligence. In: *International Conference on Machine Learning*. pp. 3987–3995 (2017)
- [35] Zhang, T., Ramakrishnan, R., Livny, M.: BIRCH: An efficient data clustering method for very large databases. In: *Proc. of the ACM SIGMOD Intern. Conf. on Management of Data*. p. 103–114 (1996)

6.10 Curriculum Vitae

6.10.1 Informations générales

Institution: Université Lyon 1	✉: LIRIS - Bâtiment Nautibus
Grade: Maître de Conférences CN	25 avenue Pierre de Coubertin
Depuis: 1 ^{er} septembre 2015	69622 Villeurbanne Cedex
Section CNU: 27	☎: +33 (0)4 72 43 12 51
Département: Polytech	@: mathieu.lefort 'at' univ-lyon1.fr
Laboratoire: LIRIS	🌐: https://perso.liris.cnrs.fr/mathieu.lefort/

6.10.1.1 Formation

- 2012 **Diplôme de thèse en informatique** (Université Nancy 2, Nancy, France)
 Soutenue: 04/07/2012
 Titre: Apprentissage spatial de corrélations multimodales par des mécanismes d'inspiration corticale
 Supervision: Bernard Girau (Université de Lorraine, LORIA), Yann Boniface (Université de Lorraine, LORIA)
- 2007 **Diplôme d'ingénieur en informatique** (ENSEEIH, Toulouse, France) avec spécialité en 3^{ème} année en robotique et systèmes embarqués (ENSTA, Paris, France)
- 2002 **Baccalauréat section scientifique** (spécialité SVT)

6.10.1.2 Parcours professionnel

- depuis 2015 **Maître de conférence** LIRIS (équipe SyCoSMA) - Université Lyon 1 (Polytech)
- 2015 **Post doctorant** LRI, U2IS - Université Paris Sud, ENSTA ParisTech
 Supervision: Michèle Sebag, Alexander Geppert
- 2013-2015 **Post doctorant** U2IS, équipe Flowers - ENSTA ParisTech, Inria Bordeaux Sud-Ouest
 Supervision: Alexander Geppert
- 2011-2012 **ATER** LORIA - IUT Nancy
 Supervision: Bernard Girau, Yann Boniface
- 2010-2011 **demi ATER** LORIA - Université Nancy 1
 Supervision: Bernard Girau, Yann Boniface
- 2007-2010 **Doctorant** (bourse ministérielle + ACE) LORIA - Université Nancy 2
 Supervision: Bernard Girau, Yann Boniface

6.10.2 Enseignement

6.10.2.1 Cours

2023-2024	256h eqTD	Lyon 1 (Polytech+département info)
	10h eqTD	Univ. Grenoble Alpes (master sciences cognitives)
2022-2023	96h eqTD	Lyon 1 (Polytech+dép. info) - demi délégation CNRS
2021-2022	246h eqTD	Lyon 1 (Polytech+département info)
	10h eqTD	Univ. Grenoble Alpes (master sciences cognitives)
2020-2021	252h eqTD	Lyon 1 (Polytech+dép. info)
	10h eqTD	Univ. Grenoble Alpes (master sciences cognitives)
2019-2020	192h eqTD	Lyon 1 (Polytech+dép. info)
	11h eqTD	Univ. Grenoble Alpes (master sciences cognitives)
2018-2019	192h eqTD	Lyon 1 (Polytech+dé. info)
	11h eqTD	Univ. Grenoble Alpes (master sciences cognitives)
2017-2018	150h eqTD	Lyon 1 (Polytech+dép. info) - décharge 42h nouvel arrivant
2016-2017	150h eqTD	Lyon 1 (Polytech+dép. info) - décharge 42h nouvel arrivant
2015-2016	150h eqTD	Lyon 1 (Polytech+dép. info) - décharge 42h nouvel arrivant
2011-2012	192h eqTD	ATER IUT Nancy
2010-2011	96h eqTD	demi ATER Nancy 1
2009-2010	64h eqTD	ACE Nancy 2
2008-2009	64h eqTD	ACE Nancy 2
2007-2008	64h eqTD	ACE Nancy 2

6.10.2.2 Responsabilités principales

2022-en cours	Co-responsable du Master 2 IA (Lyon 1)
2022-en cours	Membre du groupe de travail DDRS (Polytech Lyon)
2022-en cours	Membre du groupe de travail sobriété énergétique (Lyon 1)
2020-en cours	Responsable S7 (1er semestre équivalent M1, stage) (Polytech Lyon)
2018-2023	Membre du Conseil de Gouvernance (Polytech Lyon)
2023-2024	Membre du Conseil de département (Polytech Lyon)

6.10.3 Recherche

6.10.3.1 Projets

Je porte les projets suivants:

2024 - 2026	Data Annotation Technology Advancement With Innovative Solutions for Efficiency (DATAWISE) Financement : région AuRA R&D Booster Montant : 555k€ (dont 135k€ pour la partie académique) Partenaires : INSA, NeoVision (entreprise) Description : apprentissage auto-supervisé de représentations pour l'assistance à l'annotation de données industrielles et réduction de l'empreinte de calcul nécessaire
2024 - 2027	Apprentissage profond sensori-moteur Financement : Soutien Enseignant Chercheur (SENS) Université Lyon 1 Montant : ~ 130k€ (bourse doctorale)

- Description : apprentissage auto-supervisé de représentations prédictives incluant l'action en lien avec les théories-sensorimotrices
- 2024 - 2029 Multimodal deep SensoriMotor Representation learning (MeSMRise)
 Financement : ANR PRC
 Montant : 511k€
 Partenaires : Université Lyon 1, Univ. Grenoble Alpes, Institut Pascal (Clermont Ferrand)
- Description : apprentissage auto-supervisé de représentations multimodales multi-niveaux avec perception et apprentissage actif dans un environnement simulé en lien avec les théories-sensorimotrices

J'ai porté les projets suivants:

- 2022 - 2024 ACtive Multimodal mErging : from psychophysics to computational modeling to robotics (ACME)
 Financement : Fédération Informatique Lyon
 Montant : 10k€
 Partenaires : LIRIS, LJK (Grenoble)
 Description : projet interdisciplinaire (informatique, mathématiques appliquées, psychophysique) sur la modélisation de données psychophysiques sur la fusion active de données chez l'humain
- 2022 - 2024 Multimodal Effective Representation Learning of Evolution of birds (MERLE)
 Financement : Fédération Informatique Lyon
 Montant : 7k€
 Partenaires : LIRIS, LGL (Lyon)
 Description : projet interdisciplinaire (informatique, biologie) de représentations de caractéristiques d'oiseaux qui soient significatives pour la biologie évolutive
- 2022 - 2023 VariatiOnal ContrAstive Learning (VoCaL)
 Financement : projet transverse LIRIS
 Montant : 3.5k€
 Partenaires : LIRIS (SyCoSMA + Imagine)
 Description : étude de l'apprentissage auto-supervisé de représentations avec des aspects variationnels
- 2017 - 2022 Active Multisensory Perception and LearnIng For InteractivE Robots (AMPLIFIER)
 Financement : région AuRA Pack Ambition Recherche
 Montant : 205k€
 Partenaires : LIRIS, CRNL (Lyon), LJK (Grenoble), LPNC (Grenoble), Gipsa-lab (Grenoble), Hoomano (entreprise)
 Description : projet interdisciplinaire (informatique, mathématiques appliquées, psychophysique) sur l'analyse et la modélisation d'un effet psychophysique et l'apprentissage de modèles neuro-inspirés de fusion de données
- 2017 Active Perception For Autonomous Predictive Fusion (APF²)
 Financement : PEPS Mission pour l'interdisciplinarité
 Montant : 25k€

- Description : projet interdisciplinaire (informatique, mathématiques appliquées, psychophysique) pour la mise en place d'un protocole de psychophysique sur la fusion active de données chez l'humain
- 2016 Étude de la montée en abstraction dans l'apprentissage constructiviste à base de système multi-agents
Financement : BQR accueil
Montant : 6.7k€
- Description : étude de méthodes multi-agents constructivistes
- 2015 Rewarded Multimodal Online Deep Learning
Financement : DIGITEO
Montant : allocation post-doctorale 12 mois
- Description : fusion prédictible de données multimodales et apprentissage actif
- 2015 Séjour de deux semaines de Jean-Charles Quinton à l'ENSTA ParisTech
Financement : LIDEX iCode
Montant : 4k€
- 2015 Représentations multimodales distribuées: intérêts fonctionnels et passage à l'échelle
Financement : LIDEX iCode
Montant : 2.7k€
Description : étude de la fusion prédictible de données multimodales

Je suis également collaborateur dans les projets suivants:

- 2023 - en cours Adaptive Co-Construction of Ethics for Lifelong trustworthy AI (ACCELER-AI)
Financement : ANR PRC
Montant : 430k€
Partenaires : LIRIS, LIMOS (Mines Saint-Étienne), CSHRC (UCLy)
Description : projet interdisciplinaire (informatique, philosophie) sur l'apprentissage de comportements éthiques pour l'IA co-construit avec l'humain
- 2021 - en cours Interacting with Pepper : mutual learning of turn-taking practices in HRI (PepperMint)
Financement : Labex ASLAN
Montant : 230k€
Partenaires : LIRIS, ICAR (Lyon), GenZ (Université d'Oulu, Finlande)
Description : projet interdisciplinaire (informatique, linguistique) sur la détection et l'apprentissage d'indices multimodaux pour la prise de parole lors d'une interaction humain-robot

J'ai également été collaborateur dans les projets suivants:

- 2018 - 2023 ARTIFICIAL Constructivist Agents that Learn ETHics in Human-Involved Co-Construction (Ethics.ai)
Financement : région AuRA Pack Ambition Recherche
Montant : 200k€

- Partenaires : LIRIS, LHC (Saint-Étienne), UCLy, Ubiant (entreprise)
Description : projet interdisciplinaire (informatique, philosophie) sur l'étude de modèles d'IA éthique (proposition d'un cadre de réflexion et de modèles d'apprentissage de comportements)
- 2017 - 2020 BEHAVIORS.AI is an Engine enhancing verbal and non-Verbal Interactions of Robots based on Artificial Intelligence (Behaviors.ai)
Financement : projet ANR labcom
Montant : 300k€
Partenaires : LIRIS, Hoomano (entreprise)
Description : étude de la perception empathique d'un comportement dans le cadre de robots sociaux
- 2017 - 2019 Reconnaissance et suivi d'objets dans une séquence vidéo par des paradigmes d'apprentissage constructiviste
Financement : projet transverse LIRIS
Montant : 5k€
Description : apprentissage auto-supervisé de représentations à partir de vidéos
- 2017 - 2019 AnimIA
Financement : projet transverse LIRIS
Montant : 5k€
Description : apprentissage de représentations pour l'animation de squelettes

6.10.3.2 Encadrement

J'encadre 2 doctorants:

- 2022 - en cours Pierre-Elliott Thiboud (Co-encadrement 33% - Michaël Sdika 33% - Nicolas Duchateau 33%)
Financement : Bourse CIFRE
Titre : Structure et explicabilité des réseaux de neurones pour la prévention du sepsis
- 2021 - en cours Alexandre Devillers (Co-encadrement 65% (100% depuis 03/2023) - Salima Hassas 35%)
Financement : Bourse MESRI
Titre : Structuration des représentations visuelles pour l'amélioration de la généralisation, en particulier pour l'apprentissage auto-supervisé

J'ai encadré 5 doctorant.e.s:

- 2020 - 2023 Anaëlle Badier (Co-encadrement 25% - Nathalie Guin 50% - Marie Lefevre 25%)
Financement : Bourse CIFRE
Soutenue le 08/12/2023
Titre : Adaptation continue du processus d'Adaptive Learning, via une découverte automatique de connaissances et en interaction avec les acteurs du processus d'apprentissage
- 2018 - 2022 Simon Forest (Co-encadrement 45% - Salima Hassas 10% - Jean-Charles Quinton 45%)

- Financement : projet AuRA AMPLIFIER
Soutenue le 16/09/2022
Titre : Fusion multimodale : de la psychophysique à la robotique sociale
- 2018 - 2022 Ruiqi Dai (Co-encadrement - Stefan Duffner 25% - Frédéric Armetta 50%
- Mathieu Guillermin 25%)
Financement : Bourse MESRI
Soutenue le 14/09/2022
Titre : Apprentissage continu non supervisé pour la reconnaissance
d'objets
- 2017 - 2019 Alexandre Galdeano (Co-encadrement 33% - Salima Hassas 33% - Amélie
Cordier 33%)
Financement : Bourse CIFRE
Titre : Apprentissage développemental de comportements suggérant
l'empathie pour des robots d'interactions hétérogènes
- 2016 - 2019 Victor Lequay (Co-encadrement 50% - Salima Hassas 50%)
Financement : Bourse CIFRE
Soutenue le 11/12/2019
Titre : Gestion décentralisée et collaborative à base de multi-agents de
l'énergie dans un microgrid par apprentissage et partage multi-critère de
ressources

J'ai également encadré 23 stages de niveau M2, 6 stages de niveau L2 à M1, 2 post doctorants (pour une durée cumulée d'environ 6 mois) et 3 ingénieurs. De plus, 3 nouveaux doctorants vont ou devraient commencer prochainement:

- 07/10/2024 Nadir Bendoukha (Co-encadrement - Stefan Duffner - Jochen Triesch)
Financement : Projet ANR MeSMRise
Titre : Apprentissage profond sensori-moteur multimodal
- 01/10/2024 Nathan Salazar (Co-encadrement - Emmanuel Dellandrea - Alexandre
Meyer)
Financement : Bourse MESRI
Titre : Vers la construction d'un modèle de fondation des mouvements
humains pour l'analyse et la synthèse des actions et expressions corporelles
- 2024 Axel Bessy (Co-encadrement - Alexandre Meyer - Hamid Ladjal)
Financement : Bourse CIFRE (dossier déposé)
Titre : Fusion multimodale d'imagerie médicale thoracique pour l'aide au
diagnostic : vers un modèle général

6.10.3.3 Responsabilités

- 2023 - en cours Responsable du WP2 Arqus living lab (thème IA) de l'alliance eu-
ropéenne Arqus
- 2023 - en cours Co-responsable de l'équipe SyCoSMA
- 2023 - en cours Membre du conseil de la FIL
- 2022 - en cours Co-responsable du thème 3IA de la FIL
- 2022 - en cours Membre de la commission impact environnementaux du LIRIS
- 2022 - 2023 Responsable de l'équipe SyCoSMA (durant l'arrêt maladie de Salima
Hassas)

2020 - en cours Membre de la commission égalité femme-homme du LIRIS

J'ai également été reviewer pour un certain nombre de conférences (ECML-PKDD, AAAI, ALIFE, etc.), journaux (Transactions on Cognitive and Developmental Systems, Neural Processing Letters) et agences de financement (ANR, Initiative d'excellence Paris Seine). J'ai aussi été examinateur pour les thèses de Subhy Albakour (soutenue en 2023 à l'Institut Polytechnique) et de Duc-Canh Nguyen (soutenue en 2018 à l'Université Grenoble Alpes), et membre d'un comité de sélection de recrutement MCF à l'IUT Lyon 1 en 2023.

6.10.4 Médiation

- 2023 Rencontre discussion lycéens/chercheurs dans le cadre de l'association DÉCLICS
- 2021-en cours Participation à des week-end de création d'activités de médiation
- 2019-2021 Membre du CS pour la création de l'exposition « Entrez dans le monde de la l'IA » de la MMI / IHP (Institut Henri Poincaré) / Fermat science
- 2020, 2022, 2024 Présentation vulgarisée sur l'apprentissage profond à l'université ouverte, au meet up Data Science et à la bibliothèque Part Dieu
- 2018 Participation au comité scientifique du forum Pop' Sciences de l'Université de Lyon sur l'Intelligence Artificielle
- 2017 Mentor des savanturiers du numérique – encadrement d'un projet d'une classe de CM1
- 2016-en cours Diverses activités (ateliers informatique débranchée, *scientific dating*, ciné débat, etc.) dans différents événements/festivals grand public et scolaire de vulgarisation scientifique (fête de la science, Mix Teen, Pint of Science, coupe du monde de robotique, ...)
- 2016 Participation à la formation des enseignants du secondaire à l'enseignement de l'informatique en lien avec la réforme du brevet
- 2015-en cours Membre du groupe de travail médiation au LIRIS
- 2015 Participation à la rédaction d'un article de Planète Robots

Publications

International Publications

Journals

- [1] Lucien Tisserand, Brooke Stephenson, Heike Baldauf-Quilliatre, Mathieu Lefort, and Frédéric Armetta. Unraveling the thread: Understanding and addressing sequential failures in human-robot interaction anonymous. *Frontiers in Robotics and AI*, 11:1359782.
- [2] Simon Forest, Jean-Charles Quinton, and Mathieu Lefort. A dynamic neural field model of multimodal merging: application to the ventriloquist effect. *Neural Computation*, 34(8):1701–1726, 2022.

Conferences

- [3] Anaëlle Badier, Mathieu Lefort, and Marie Lefevre. Recommendation model for an after-school e-learning mobile application. In *International Conference on Computer Supported Education (CSEDU)*, pages 80–87, 2023.
- [4] Anaëlle Badier, Mathieu Lefort, and Marie Lefevre. Understanding the usages and effects of a recommendation system in a non-formal learning context. In *International Conference on Intelligent Tutoring Systems (ITS)*, pages 54–65. Springer, 2023.
- [5] Alexandre Devillers and Mathieu Lefort. Equimod: An equivariance module to improve visual instance discrimination. In *International Conference on Learning Representations (ICLR)*, 2023.
- [6] Alexandre Devillers, Valentin Chaffraix, Frédéric Armetta, Stefan Duffner, and Mathieu Lefort. The impact of action in visual representation learning. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 30–35. IEEE, 2022.
- [7] Simon Forest, Jean-Charles Quinton, and Mathieu Lefort. Combining manifold learning and neural field dynamics for multimodal fusion. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.
- [8] Ruiqi Dai, Mathieu Lefort, Frédéric Armetta, Mathieu Guillermin, and Stefan Duffner. Novelty detection for unsupervised continual learning in image sequences. In *International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 493–500. IEEE, 2021.

-
- [9] Ruiqi Dai, Mathieu Lefort, Frédéric Armetta, Mathieu Guillermin, and Stefan Duffner. Self-supervised continual learning for object recognition in image sequences. In *Neural Information Processing (ICONIP)*, pages 239–247. Springer, 2021.
- [10] Bastien Nollet, Mathieu Lefort, and Frédéric Armetta. Learning Arithmetic Operations With A Multistep Deep Learning. In *The International Joint Conference on Neural Networks (IJCNN)*, Glasgow, United Kingdom, July 2020.
- [11] Laurianne Charrier, Alisa Rieger, Alexandre Galdeano, Amélie Cordier, Mathieu Lefort, and Salima Hassas. The rope scale: a measure of how empathic a robot is perceived. In *International Conference on Human-Robot Interaction (HRI)*, pages 656–657. IEEE, 2019.
- [12] Alexander Gepperth and Mathieu Lefort. Learning to be attractive: probabilistic computation with dynamic attractor networks. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 270–277. IEEE, 2016.
- [13] Victor Lequay, Mathieu Lefort, Saber Mansour, and Salima Hassas. Flexible load shedding using gossip communication in a multi-agents system. In *International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, pages 31–39. IEEE, 2016.
- [14] Alexander Gepperth, Thomas Hecht, Mathieu Lefort, and Ursula Korner. Biologically inspired incremental learning for high-dimensional spaces. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 269–275. IEEE, 2015.
- [15] Alexander Gepperth, Mathieu Lefort, Thomas Hecht, and Ursula Körner. Resource-efficient incremental learning in very high dimensions. In *International Conference on Artificial Neural Networks (ESANN)*, 2015.
- [16] Thomas Hecht, Mathieu Lefort, and Alexander Gepperth. Using self-organizing maps for regression: the importance of the output function. In *European Symposium on artificial neural networks (ESANN)*, pages 107–112, 2015.
- [17] Mathieu Lefort and Alexander Gepperth. Active learning of local predictable representations with artificial curiosity. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 228–233. IEEE, 2015.
- [18] Mathieu Lefort and Alexander Gepperth. Learning of local predictable representations in partially learnable environments. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2015.
- [19] Alexander Gepperth and Mathieu Lefort. Latency-based probabilistic information processing in recurrent neural hierarchies. In *International Conference on Artificial Neural Networks (ESANN)*, pages 715–722. Springer, 2014.
- [20] Mathieu Lefort and Alexander Gepperth. Discrimination of visual pedestrians data by combining projection and prediction learning. In *European Symposium on artificial neural networks (ESANN)*, 2014.

- [21] Mathieu Lefort and Alexander Gepperth. Propre: Projection and prediction for multimodal correlations learning. an application to pedestrians visual data discrimination. In *International Joint Conference on Neural Networks (IJCNN)*, pages 2718–2725. IEEE, 2014.
- [22] Mathieu Lefort, Thomas Kopinski, and Alexander Gepperth. Multimodal space representation driven by self-evaluation of predictability. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 319–324. IEEE, 2014.
- [23] Mathieu Lefort, Yann Boniface, and Bernard Girau. Somma: Cortically inspired paradigms for multimodal processing. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2013.
- [24] Mathieu Lefort, Yann Boniface, and Bernard Girau. Coupling bcm and neural fields for the emergence of self-organization consensus. In *From Brains to Systems: Brain-Inspired Cognitive Systems*, pages 41–56. Springer, 2011.
- [25] Mathieu Lefort, Yann Boniface, and Bernard Girau. Unlearning in the bcm learning rule for plastic self-organization in a multi-modal architecture. In *International Conference on Artificial Neural Networks (ICANN)*, pages 93–100. Springer, 2011.
- [26] Mathieu Lefort, Yann Boniface, and Bernard Girau. Self-organization of neural maps using a modulated bcm rule within a multimodal architecture. In *From Brains to Systems: Brain-Inspired Cognitive Systems*, page 26, 2010.
- [27] Jean-Charles Quinton, Bernard Girau, and Mathieu Lefort. Competition in high dimensional spaces using a sparse approximation of neural fields. In *From Brains to Systems: Brain-Inspired Cognitive Systems*, pages 123–137. Springer, 2010.

Workshops

- [28] Frédéric Armetta, Anthony Baccuet, and Mathieu Lefort. Algorithmic learning, a next step for ai. An application to arithmetic operations. In *International Workshop on Artificial Intelligence and Cognition (AIC)*, 2023.
- [29] Lucien Tisserand, Frédéric Armetta, Heike Baldauf-Quilliatre, Antoine Bouquin, Salima Hassas, and Mathieu Lefort. Sequential annotations for naturally-occurring hri: first insights. In *Workshop Human-Robot Conversational Interaction (HRCI) in International Conference on Human-Robot Interaction (HRI)*, 2023.
- [30] Arthur Aubret, Mathieu Lefort, Céline Teulière, Laetitia Matignon, Salima Hassas, and Jochen Triesch. Compressed information is all you need: unifying intrinsic motivations and representation learning. In *NeurIPS 2022 Workshop on Information-Theoretic Principles in Cognitive Systems*, 2022.
- [31] Simon Forest, Mathieu Lefort, and Jean-Charles Quinton. Biological constraints in neural field models of sensor fusion. In *International Workshop on Intrinsically Motivated Open-ended Learning (IMOL)*, 2019.

-
- [32] Laurianne Charrier, Alexandre Galdeano, Amélie Cordier, and Mathieu Lefort. Empathy display influence on human-robot interactions: a pilot study. In *Workshop on Towards Intelligent Social Robots: From Naive Robots to Robot Sapiens at International Conference on Intelligent Robots and Systems (IROS)*, page 7, 2018.
- [33] Alexandre Galdeano, Alix Gonnot, Clément Cottet, Salima Hassas, Mathieu Lefort, and Amélie Cordier. Developmental learning for social robots in real-world interactions. In *1st Workshop on Social Robots in the Wild at International Conference on Human-Robot Interaction (HRI)*, page 5, 2018.
- [34] Mathieu Lefort, Jean-Charles Quinton, Marie Avillac, and Adrien Techer. Active multisensory perception and learning for interactive robots. In *Workshop on Computational Models for Crossmodal Learning at International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, page 2, 2017.

National Publications

Journals

- [35] Marie Lefevre, Anaëlle Badier, Mathieu Lefort, and Nathalie Guin. Adaptive learning en contexte parascolaire : comprendre les usages et effets via l'analyse des traces d'un déploiement industriel. *Revue STICEF*, 2024.
- [36] Mathieu Guillermin and Mathieu Lefort. De la signification éthique des limites. *La personne transformée. Nouveaux enjeux éthiques et juridiques*, pages 91–112, 2019.
- [37] Victor Lequay, Mathieu Lefort, Saber Mansour, and Salima Hassas. Ajustement diffus et adaptatif de la consommation électrique résidentielle par un système multi-agent auto-adaptatif. *Revue des Sciences et Technologies de l'Information-Série RIA: Revue d'Intelligence Artificielle*, 31(4/2017):427–447, 2017.

Conferences

- [38] Frédéric Armetta, Anthony Baccuet, and Mathieu Lefort. L'apprentissage algorithmique, une nouvelle étape pour l'ia. une application aux opérations arithmétiques. In *Conférence Nationale en Intelligence Artificielle, PFIA*, pages 31–39, 2023.
- [39] Anaëlle Badier, Mathieu Lefort, and Marie Lefevre. Comprendre les usages et effets d'un système de recommandations pédagogiques en contexte d'apprentissage non-formel. In *Conférence sur les Environnements Informatiques pour l'Apprentissage Humain (EIAH)*, 2023.
- [40] Alexandre Devillers, Benoît Alcaraz, and Mathieu Lefort. Suréchantillonnage actif pour modérer l'apprentissage de biais visuels. In *Conférence sur l'Apprentissage Automatique (CAp)*, 2023.

- [41] Simon Forest, Jean-Charles Quinton, and Mathieu Lefort. Champ neuronal et apprentissage profond de topologies pour la fusion multimodale. In *Conférence Nationale en Intelligence Artificielle, PFIA*, pages 40–49, 2023.
- [42] Alexandre Devillers and Mathieu Lefort. Towards considering explicit sensitivity to augmentation in visual instance discrimination tasks. In *Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA)*, 2022.
- [43] Anaëlle Badier, Mathieu Lefort, and Marie Lefevre. Système de recommandation de ressources pédagogiques pour un apprentissage sur application mobile parascolaire. In *Conférence sur les Environnements Informatiques pour l'Apprentissage Humain (EIAH)*, pages 294–299, 2021.
- [44] Victor Lequay, Mathieu Lefort, Saber Mansour, and Salima Hassas. Ajustement diffus et adaptatif de la consommation électrique résidentielle par un système multi-agents. In *Journées Francophones sur les Systèmes Multi-Agents (JFSMA)*, pages 171–180, 2016.
- [45] Mathieu Lefort, Yann Boniface, and Bernard Girau. Auto-organisation d'une carte de neurones bcm sous contrainte multimodale. In *5ème Conférence française de Neurosciences Computationnelles (Neurocomp)*, 2010.
- [46] Mathieu Lefort, Yann Boniface, and Bernard Girau. Auto organisation d'une carte de neurones par modulation de la règle bcm dans un cadre multimodal. In *17e congrès francophone AFRIF-AFIA Reconnaissance des Formes et Intelligence Artificielle-RFIA*, pages 423–430, 2010.
- [47] Mathieu Lefort, Yann Boniface, and Bernard Girau. Feedback modulation of bcm's neurons in multi modal environment. In *4ème Conférence française de Neurosciences Computationnelles (Neurocomp)*, pages P–20, 2009.

Workshops

- [48] Mathieu Lefort, Jean-Charles Quinton, Simon Forest, Adrien Techer, Alan Chauvin, and Marie Avillac. Influence of eye-movements on multisensory stimulus localization: experiments, models and robotics applications. In *Grenoble Workshop on Models and Analysis of Eye Movements*, page 1, 2018.

Others

Posters

- [49] Nawel Medjkoune, Frédéric Armetta, Mathieu Lefort, and Stefan Duffner. Autonomous object recognition in videos using siamese neural networks. In *EUCognition Meeting (European Society for Cognitive Systems) on "Learning: Beyond Deep Neural Networks"*, page 4, 2017.
- [50] Mathieu Lefort, Yann Boniface, and Bernard Girau. The somma model: cortically inspired maps for multimodal learning. In *Cognitive Systems*, 2012.

-
- [51] Mathieu Lefort, Yann Boniface, and Bernard Girau. Perceptive self-organizing maps based on the coupling of neural fields with the bcm learning rule for multi modal association. 2011.
- [52] Mathieu Lefort, Yann Boniface, and Bernard Girau. Self-organizing neural maps for multi-modal associations. *BMC Neuroscience*, 12(Suppl 1):P125, 2011.
- [53] Thomas Girod, Mathieu Lefort, and Jean-Charles Quinton. Cortically-inspired computational models for multimodality. In *Third EUCogII Members Conference-Multisensory integration*, 2010.
- [54] Mathieu Lefort, Yann Boniface, and Bernard Girau. Multi-sensory integration by constrained self-organization. In *Third EUCogII Members Conference-Multisensory integration*, 2010.

Theses

- [55] Mathieu Lefort. *Apprentissage spatial de corrélations multimodales par des mécanismes d'inspiration corticale*. PhD thesis, Université de Lorraine, 2012.
- [56] Mathieu Lefort. Développement d'un algorithme décentralisé de planification dans un jeu stochastique fini non escompté à horizon fini à deux joueurs. Master thesis. 2007.

Bibliography

- [57] Sumyeong Ahn, Seongyeon Kim, and Se-Young Yun. Mitigating dataset bias by using per-sample gradient. *arXiv preprint arXiv:2205.15704*, 2022.
- [58] David Alais and David Burr. The ventriloquist effect results from near-optimal bimodal integration. *Current Biology*, 14(3):257–262, 2004. ISSN 0960-9822. doi: 10.1016/j.cub.2004.01.029.
- [59] V Ralph Algazi, Richard O Duda, Dennis M Thompson, and Carlos Avendano. The CIPIC HRTF database. In *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No. 01TH8575)*, pages 99–102. IEEE, 2001.
- [60] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, pages 139–154, 2018.
- [61] Shun-Ichi Amari. Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 27(2):77–87, June 1977. ISSN 1432-0770. doi: 10.1007/BF00337259.
- [62] S. Argentieri, P. Danès, and P. Souères. A survey on sound source localization in robotics: From binaural to array processing methods. *Computer Speech & Language*, 34(1):87–112, 2015. ISSN 0885-2308. doi: 10.1016/j.csl.2015.03.003.
- [63] Pradeep K Atrey, M Anwar Hossain, Abdulmotaleb El Saddik, and Mohan S Kankanhalli. Multimodal fusion for multimedia analysis: a survey. *Multimedia systems*, 16:345–379, 2010.
- [64] Arthur Aubret, Laetitia Matignon, and Salima Hassas. A survey on intrinsic motivation in reinforcement learning. *arXiv preprint arXiv:1908.06976*, 2019.
- [65] Arthur Aubret, Céline Teulière, and Jochen Triesch. Self-supervised visual learning from interactions with objects. *arXiv preprint arXiv:2407.06704*, 2024.
- [66] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pages 528–539. PMLR, 2020.
- [67] Ruzena Bajcsy, Yiannis Aloimonos, and John K Tsotsos. Revisiting active perception. *Autonomous Robots*, 42:177–196, 2018.

-
- [68] Michele Banko and Eric Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pages 26–33, 2001.
- [69] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.
- [70] Horace B Barlow et al. Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1(01):217–233, 1961.
- [71] Yoshua Bengio, Ian Goodfellow, and Aaron Courville. *Deep learning*, volume 1. MIT press Cambridge, MA, USA, 2017.
- [72] Mark H Bickhard. The interactivist model. *Synthese*, 166:547–591, 2009.
- [73] Elie L Bienenstock, Leon N Cooper, and Paul W Munro. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *Journal of Neuroscience*, 2(1):32–48, 1982.
- [74] Rafal Bogacz, Marius Usher, Jiaxiang Zhang, and James L McClelland. Extending a biologically inspired model of choice: multi-alternatives, nonlinearity and value-based multidimensional choice. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1485):1655–1670, 2007.
- [75] Jeannette Bohg, Karol Hausman, Bharath Sankaran, Oliver Brock, Danica Kragic, Stefan Schaal, and Gaurav S Sukhatme. Interactive perception: Leveraging action in perception and perception in action. *IEEE Transactions on Robotics*, 33(6): 1273–1291, 2017.
- [76] Ali Borji, Saeed Izadi, and Laurent Itti. iLab-20M: A large-scale controlled object dataset to investigate deep learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2221–2230, 2016.
- [77] Valentino Braitenberg. *Vehicles: Experiments in synthetic psychology*. MIT press, 1986.
- [78] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a” siamese” time delay neural network. In *Advances in Neural Information Processing Systems*, pages 737–744, 1994.
- [79] Rodney A Brooks. Elephants don’t play chess. *Robotics and autonomous systems*, 6(1-2):3–15, 1990.
- [80] Yves Burnod. *An adaptive neural network: the cerebral cortex*. Masson editeur, 1990.
- [81] Remi Cadene, Corentin Dancette, Matthieu Cord, and Devi Parikh. Rubi: Reducing unimodal biases for visual question answering. *Advances in neural information processing systems*, 32, 2019.
- [82] Gemma Calvert, Charles Spence, and Barry E Stein. *The handbook of multisensory processes*. MIT press, 2004.

- [83] Lawrence Cayton et al. *Algorithms for manifold learning*. eScholarship, University of California, 2008.
- [84] Arslan Chaudhry, Naeemullah Khan, Puneet Dokania, and Philip Torr. Continual learning in low-rank orthogonal subspaces. *Advances in Neural Information Processing Systems*, 33:9900–9911, 2020.
- [85] Kaiyu Chen, Yihan Dong, Xipeng Qiu, and Zitian Chen. Neural arithmetic expression calculator. *CoRR*, abs/1809.08590, 2018.
- [86] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.
- [87] Andrew A Chien, Liuzixuan Lin, Hai Nguyen, Varsha Rao, Tristan Sharma, and Rajini Wijayawardana. Reducing the carbon impact of generative ai inference (today and in 2035). In *Proceedings of the 2nd workshop on sustainable computer systems*, pages 1–7, 2023.
- [88] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don’t take the easy way out: Ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv:1909.03683*, 2019.
- [89] Rumen Dangovski, Li Jing, Charlotte Loh, Seungwook Han, Akash Srivastava, Brian Cheung, Pulkit Agrawal, and Marin Soljačić. Equivariant Contrastive Learning. *arXiv preprint arXiv:2111.00899*, 2021.
- [90] Sophie Deneve, Peter E Latham, and Alexandre Pouget. Efficient computation and cue integration with noisy population codes. *Nature neuroscience*, 4(8):826–831, 2001.
- [91] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [92] Benjamin Devillers, Léopold Maytié, and Rufin VanRullen. Semi-supervised multimodal representation learning through a global workspace. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.
- [93] Marc O Ernst and Martin S Banks. Humans integrate visual and haptic information in a statistically optimal fashion. *Nature*, 415(6870):429–433, 2002.
- [94] Erwin Erwin, Klaus Obermayer, and Klaus Schulten. Self-organizing maps: ordering, convergence properties and energy functions. *Biological cybernetics*, 67(1):47–55, 1992.
- [95] Samuel Felton, Elisa Fromont, and Eric Marchand. Deep metric learning for visual servoing: when pose and image meet in latent space. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 741–747. IEEE, 2023.
- [96] Jeremy Fix, Nicolas Rougier, and Frederic Alexandre. A dynamic neural field approach to the covert and overt deployment of spatial attention. *Cognitive Computation*, 3(1):279–293, March 2011. doi: 10.1007/s12559-010-9083-y.

-
- [97] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.
- [98] Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, Giovanni Pezzulo, et al. Active inference and learning. *Neuroscience & Biobehavioral Reviews*, 68:862–879, 2016.
- [99] Bernd Fritzsche. A growing neural gas network learns topologies. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7. MIT Press, 1995.
- [100] S. Furao, T. Ogura, and O. Hasegawa. An enhanced self-organizing incremental neural network for online unsupervised learning. *Neural Networks*, 20(8), 2007.
- [101] Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning. *arXiv preprint arXiv:2403.00504*, 2024.
- [102] Jacques Gautrais and Simon Thorpe. Rate coding versus temporal order coding: a theoretical approach. *Biosystems*, 48(1-3):57–65, 1998.
- [103] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. Publisher: Nature Publishing Group.
- [104] Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G Bellemare. Deepmdp: Learning continuous latent space models for representation learning. In *International conference on machine learning*, pages 2170–2179. PMLR, 2019.
- [105] C. Geng, S.-J. Huang, and S. Chen. Recent advances in open set recognition: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2020.
- [106] James Jerome Gibson. *The senses considered as perceptual systems*. Houghton Mifflin, 1966.
- [107] Gabriel Gohau. Karl r. popper, la logique de la découverte scientifique, collection bibliothèque scientifique payot, 1973. *Raison présente*, 32(1):121–124, 1974.
- [108] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.
- [109] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *CoRR*, abs/1410.5401, 2014.
- [110] Tom George Grigg, Dan Busbridge, Jason Ramapuram, and Russ Webb. Do Self-Supervised and Supervised Methods Learn Similar Visual Representations?, December 2021. arXiv:2110.00528 [cs, stat].

- [111] Jean-Bastien Grill, Florian Strub, Florent Alché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, and Mohammad Gheshlaghi Azar. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.
- [112] Aldric Hama. The invisible gorilla: and other ways our intuitions deceive us. *The Journal of Social, Political and Economic Studies*, 35(4):537–543, 2010.
- [113] Moritz Hardt and Benjamin Recht. *Patterns, predictions, and actions: Foundations of machine learning*. Princeton University Press, 2022.
- [114] Nicholas Hay, Michael Stark, Alexander Schlegel, Carter Wendelken, Dennis Park, Eric Purdy, Tom Silver, D Scott Phoenix, and Dileep George. Behavior is everything: Towards representing concepts with sensorimotor contingencies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.
- [115] Donald Olding Hebb. *The organization of behavior: A neuropsychological theory*. Psychology press, 1949.
- [116] Richard Held and Alan Hein. Movement-produced stimulation in the development of visually guided behavior. *Journal of comparative and physiological psychology*, 56(5):872, 1963.
- [117] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.
- [118] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [119] Alan Lloyd Hodgkin and Andrew Fielding Huxley. Propagation of electrical signals along giant nerve fibres. *Proceedings of the Royal Society of London. Series B-Biological Sciences*, 140(899):177–183, 1952.
- [120] Harold Hotelling et al. The generalization of student’s ratio. 1931.
- [121] Dapeng Hu, Shipeng Yan, Qizhengqiu Lu, Lanqing Hong, Hailin Hu, Yifan Zhang, Zhenguo Li, Xinchao Wang, and Jiashi Feng. How well does self-supervised pre-training perform with streaming data? *arXiv preprint arXiv:2104.12081*, 2021.
- [122] Tianyu Hua, Wenxiao Wang, Zihui Xue, Sucheng Ren, Yue Wang, and Hang Zhao. On feature decorrelation in self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9598–9608, 2021.
- [123] Yanping Huang and Rajesh PN Rao. Predictive coding. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(5):580–593, 2011.
- [124] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 20(11):1254–1259, 1998.
- [125] Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *arXiv preprint arXiv:2110.09348*, 2021.

-
- [126] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):4037–4058, 2020.
- [127] Lynn H Kaack, Priya L Donti, Emma Strubell, George Kamiya, Felix Creutzig, and David Rolnick. Aligning artificial intelligence with climate change mitigation. *Nature Climate Change*, 12(6):518–527, 2022.
- [128] Lukasz Kaiser and Ilya Sutskever. Neural gpu learn algorithms. *arXiv preprint arXiv:1511.08228*, 2015.
- [129] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 03 1960.
- [130] Andrew J. King. The superior colliculus. *Current Biology*, 14(9):335–338, May 2004. ISSN 0960-9822. doi: 10.1016/j.cub.2004.04.018.
- [131] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [132] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A Rusu, K. Milan, J. Quan, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.
- [133] Teuvo Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9): 1464–1480, 1990.
- [134] Soheil Kolouri, Phillip E Pope, Charles E Martin, and Gustavo K Rohde. Sliced wasserstein auto-encoders. In *International Conference on Learning Representations*, 2018.
- [135] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.
- [136] Thomas S Kuhn. *The structure of scientific revolutions*, volume 962. University of Chicago press Chicago, 1997.
- [137] Benjamin J Kuipers, Patrick Beeson, Joseph Modayil, and Jefferson Provost. Bootstrap learning of foundational representations. *Connection Science*, 18(2): 145–158, 2006.
- [138] Alban Laflaquiere. A sensorimotor perspective on contrastive multiview visual representation learning. *IEEE Transactions on Cognitive and Developmental Systems*, 14(2):269–278, 2021.
- [139] Alban Laflaquière, J Kevin O’Regan, Bruno Gas, and Alexander Terekhov. Discovering space—grounding spatial topology and metric regularity in a naive agent’s sensorimotor experience. *Neural Networks*, 105:371–392, 2018.
- [140] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1817–1824. IEEE, 2011.

- [141] Guillaume Lample and François Charton. Deep learning for symbolic mathematics. *arXiv preprint arXiv:1912.01412*, 2019.
- [142] Laura Leal-Taixé, Cristian Canton-Ferrer, and Konrad Schindler. Learning by tracking: Siamese CNN for robust target association. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 33–40, 2016.
- [143] Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard, and Lawrence Jackel. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2, 1989.
- [144] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database, 2010.
- [145] Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. Learning debiased representation via disentangled feature augmentation. *Advances in Neural Information Processing Systems*, 34:25123–25133, 2021.
- [146] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges. *Information fusion*, 58:52–68, 2020.
- [147] Yi Li and Nuno Vasconcelos. Repair: Removing representation bias by dataset resampling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9572–9581, 2019.
- [148] Xiaodan Liang, Si Liu, Yunchao Wei, Luoqi Liu, Liang Lin, and Shuicheng Yan. Towards computational baby learning: A weakly-supervised approach for object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 999–1007, 2015.
- [149] Ran Liu. Understand and Improve Contrastive Learning Methods for Visual Representation: A Review. *arXiv preprint arXiv:2106.03259*, 2021.
- [150] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 7765–7773, 2018.
- [151] Marie Martel, Lucilla Cardinali, Alice C Roy, and Alessandro Farnè. Tool-use: An open window into body representation and its plasticity. *Cognitive neuropsychology*, 33(1-2):82–101, 2016.
- [152] Thomas M Martinetz, Stanislav G Berkovich, and Klaus J Schulten. ‘neural-gas’ network for vector quantization and its application to time-series prediction. *IEEE transactions on neural networks*, 4(4):558–569, 1993.
- [153] Marc Masana, Xialei Liu, Bartłomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost Van De Weijer. Class-incremental learning: survey and performance evaluation on image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5513–5533, 2022.

-
- [154] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5:115–133, 1943.
- [155] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [156] Marvin Minsky and Seymour Papert. An introduction to computational geometry. *Cambridge tiass., HIT*, 479(480):104, 1969.
- [157] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. *arXiv preprint arXiv:1406.6247*, 2014.
- [158] Hans Moravec. *Mind children: The future of robot and human intelligence*. Harvard University Press, 1988.
- [159] Erik Myin and J Kevin O’Regan. Perceptual consciousness, access to modality and skill theories. a way to naturalize phenomenology? *Journal of Consciousness Studies*, 9(1):27–46, 2002.
- [160] Lorenzo Natale, Giorgio Metta, and Giulio Sandini. Learning haptic representation of objects. In *International Conference on Intelligent Manipulation and Grasping*, page 43. Genoa, 2004.
- [161] J Kevin O’regan and Alva Noë. A sensorimotor account of vision and visual consciousness. *Behavioral and brain sciences*, 24(5):939–973, 2001.
- [162] Fenno P. Ottes, Jan A.M. Van Gisbergen, and Jos J. Eggermont. Visuomotor fields of the superior colliculus: A quantitative model. *Vision Research*, 26(6):857–873, 1986. ISSN 0042-6989. doi: 10.1016/0042-6989(86)90144-6.
- [163] Pierre-Yves Oudeyer. Computational theories of curiosity-driven learning. *arXiv preprint arXiv:1802.10546*, 2018.
- [164] Pierre-Yves Oudeyer and Frederic Kaplan. What is intrinsic motivation? a typology of computational approaches. *Frontiers in neurorobotics*, 1:108, 2007.
- [165] Giovanni Pezzulo, Paul F. M. J. Verschure, Christian Balkenius, and Cyriel M. A. Pennartz. The principles of goal-directed decision-making: from neural mechanisms to computation and robotics. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1655):20130470, 2014. doi: 10.1098/rstb.2013.0470.
- [166] Rolf Pfeifer and Josh Bongard. *How the body shapes the way we think: a new view of intelligence*. MIT press, 2006.
- [167] Jean Piaget. The construction of reality in the child. *Journal of Consulting Psychology*, 19(1):77, 1955.
- [168] Alexandre Pitti, Arnaud Blanchard, Matthieu Cardinaux, and Philippe Gaussier. Gain-field modulation mechanism in multimodal networks for spatial perception. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 297–302. IEEE, 2012.

- [169] Jean-Charles Quinton. Exploring and optimizing dynamic neural fields parameters using genetic algorithms. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2010.
- [170] Jean-Charles Quinton and Laurent Goffart. A unified dynamic neural field model of goal directed eye movements. *Connection Science*, 30(1):20–52, 2018.
- [171] Monique Radeau and Paul Bertelson. The after-effects of ventriloquism. *The Quarterly journal of experimental psychology*, 26(1):63–71, 1974.
- [172] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [173] Marc’Aurelio Ranzato. On learning where to look. *arXiv preprint arXiv:1405.5488*, 2014.
- [174] Dushyant Rao, Francesco Visin, Andrei Rusu, Razvan Pascanu, Yee Whye Teh, and Raia Hadsell. Continual unsupervised representation learning. In *Advances in Neural Information Processing Systems*, pages 7645–7655, 2019.
- [175] Roger Ratcliff and Gail McKoon. The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation*, 20(4):873–922, 04 2008.
- [176] Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. Impact of pretraining term frequencies on few-shot numerical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 840–854, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.
- [177] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022.
- [178] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.
- [179] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization (1958). 1958.
- [180] David E Rumelhart, James L McClelland, PDP Research Group, et al. *Parallel distributed processing, volume 1: Explorations in the microstructure of cognition: Foundations*. The MIT press, 1986.
- [181] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Pearson, 2016.
- [182] Olawale Salaudeen and Moritz Hardt. Imagenot: A contrast with imagenet preserves model rankings. *arXiv preprint arXiv:2404.02112*, 2024.

-
- [183] Yulia Sandamirskaya. Dynamic neural fields as a step toward cognitive neuromorphic architectures. *Frontiers in Neuroscience*, 7:276, 2014. doi: 10.3389/fnins.2013.00276.
- [184] Nikunj Saunshi, Jordan Ash, Surbhi Goel, Dipendra Misra, Cyril Zhang, Sanjeev Arora, Sham Kakade, and Akshay Krishnamurthy. Understanding contrastive learning requires incorporating inductive biases. In *International Conference on Machine Learning*, pages 19250–19286. PMLR, 2022.
- [185] David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. Analysing mathematical reasoning abilities of neural models. *arXiv preprint arXiv:1904.01557*, 2019.
- [186] Dominik Schmidt and Minqi Jiang. Learning to act without actions. *arXiv preprint arXiv:2312.10812*, 2023.
- [187] Gregor Schöner and John P Spencer. *Dynamic thinking: A primer on dynamic field theory*. Oxford University Press, 2016.
- [188] John R Searle. Minds, brains, and programs. *Behavioral and brain sciences*, 3(3): 417–424, 1980.
- [189] H.T. Siegelmann and E.D. Sontag. On the computational power of neural nets. *Journal of Computer and System Sciences*, 50(1):132 – 150, 1995. ISSN 0022-0000. doi: <https://doi.org/10.1006/jcss.1995.1013>.
- [190] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676): 354–359, 2017.
- [191] Daniel A Slutsky and Gregg H Recanzone. Temporal and spatial dependency of the ventriloquism effect. *Neuroreport*, 12(1):7–10, 2001.
- [192] James Smith, Cameron Taylor, Seth Baer, and Constantine Dovrolis. Unsupervised progressive learning and the stam architecture. *arXiv preprint arXiv:1904.02021*, 2019.
- [193] Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial life*, 11(1-2):13–29, 2005.
- [194] Georgi Stojanov and Andrea Kulakov. Interactivist approach to representation in epigenetic agents. 2003.
- [195] Freek Stulp and Olivier Sigaud. Many regression algorithms, one unified model: A review. *Neural Networks*, 69:60–79, 2015.
- [196] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc., 2014.

- [197] Wahiba Taouali, Laurent Goffart, Frédéric Alexandre, and Nicolas P. Rougier. A parsimonious computational model of visual target position encoding in the superior colliculus. *Biological Cybernetics*, 109(4):549–559, October 2015. ISSN 1432-0770. doi: 10.1007/s00422-015-0660-8.
- [198] Enzo Tartaglione, Carlo Alberto Barbano, and Marco Grangetto. End: Entangling and disentangling deep representations for bias correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13508–13517, 2021.
- [199] René Thom. *Modélisation et scientificité*. Maloine, 1979.
- [200] Andrew Trask, Felix Hill, Scott E Reed, Jack Rae, Chris Dyer, and Phil Blunsom. Neural arithmetic logic units. In *Advances in Neural Information Processing Systems 31*, pages 8035–8044. Curran Associates, Inc., 2018.
- [201] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.
- [202] Francisco J Varela, Evan Thompson, and Eleanor Rosch. *The embodied mind, revised edition: Cognitive science and human experience*. MIT press, 2017.
- [203] Shashanka Venkataramanan, Mamshad Nayeem Rizve, João Carreira, Yuki M Asano, and Yannis Avrithis. Is imagenet worth 1 video? learning strong image encoders from 1 long unlabelled video. *arXiv preprint arXiv:2310.08584*, 2023.
- [204] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [205] Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5310–5319, 2019.
- [206] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.
- [207] Yifei Wang, Qi Zhang, Yisen Wang, Jiansheng Yang, and Zhouchen Lin. Chaos is a Ladder: A New Theoretical Understanding of Contrastive Learning via Augmentation Overlap, May 2022. arXiv:2203.13457 [cs, stat].
- [208] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.
- [209] Tete Xiao, Xiaolong Wang, Alexei A. Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. *arXiv preprint arXiv:2008.05659*, 2020.
- [210] Yihao Xue, Eric Gan, Jiayi Ni, Siddharth Joshi, and Baharan Mirzasoleiman. Investigating the Benefits of Projection Head for Representation Learning, March 2024. arXiv:2403.11391 [cs].

-
- [211] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *arXiv preprint arXiv:1708.01547*, 2017.
- [212] L A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.
- [213] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.
- [214] Yunzhe Zhang, Yao Lu, and Qi Xuan. How Does Contrastive Learning Organize Images? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 497–506, 2024.
- [215] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31, 2018.