



Learning Event-Based Perception Without Supervision. A spiking predictive coding model for learning efficient visual representations

Antony N'Dri

► To cite this version:

Antony N'Dri. Learning Event-Based Perception Without Supervision. A spiking predictive coding model for learning efficient visual representations. Neural and Evolutionary Computing [cs.NE]. Université Clermont Auvergne, 2025. English. ⟨NNT : 2025UCFA0100⟩. ⟨tel-05526587⟩

HAL Id: tel-05526587

<https://theses.hal.science/tel-05526587v1>

Submitted on 25 Feb 2026

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.



HAL Authorization

Learning Event-Based Perception Without Supervision

A spiking predictive coding model for learning efficient
visual representations

Apprentissage non supervisé de perception basée événements

*Un modèle impulsif de codage prédictif pour l'apprentissage de représentations visuelles
efficaces*

Thèse présentée par **Antony N'DRI**

Pour obtenir le grade de **Docteur d'Université**

Spécialité : **Image, Système de Perception et Robotique**

Unité de recherche : **Institut Pascal**

Soutenue publiquement le **10 Octobre 2025** devant le jury composé de

Fred HAMKER	Rapporteur, Président	Professeur des Universités Technische Universität Chemnitz.
Timothée MASQUELIER	Rapporteur	Directeur de Recherche CerCo.
Jean MARTINET	Examinateur	Professeur des Universités Université Côte d'Azur.
Yulia SANDAMIRSKAYA	Examinatrice	Professeure des Universités ZHAW.
Jochen TRIESCH	Directeur	Professeur des Universités FIAS.
Céline TEULIERE	Encadrante	Maîtresse de Conférences Université Clermont Auvergne.

REMERCIEMENTS

Cette thèse a été une épreuve difficile sur plusieurs points émotionnels et psychiques. Durant ces 3 années, j'ai découvert mes plus profonds retranchements, principalement en matière de résilience et de rigueur, dans cette poursuite accrue de connaissance qu'est le doctorat. Malgré tous les obstacles intellectuels et psychologiques rencontrés, qui, je pense, resteront à jamais gravés dans ma mémoire, je ressors assurément grandi de cette expérience doctorale. Ce voyage unique et personnel m'a permis de me découvrir une fascination pour le cerveau via mes recherches en ingénierie neuromorphique mêlant neuroscience computationnelle et sciences de l'ingénierie. Cela marque, en mon sens, une étape importante dans la définition de mes intérêts scientifiques personnels ainsi que celle de mes futurs choix professionnels.

Premièrement, j'adresse mes remerciements aux rapporteurs et examinateurs de ma thèse pour le temps précieux qu'ils ont porté à l'évaluation de ma thèse, avec des critiques et discussions captivantes, enrichissantes et très pertinentes pour la poursuite et l'amélioration de ces travaux. J'adresse un remerciement particulier à Fred Hamker pour son déplacement en présentiel à Clermont-Ferrand pour ma soutenance de thèse. Ensuite, je souhaiterais adresser mes plus profonds remerciements à mes encadrants, Céline et Jochen. J'ai pu sortir de ce doctorat en tant que chercheur grâce à leur encadrement remarquable qui m'a toujours poussé dans le bon sens. Humainement je me considère particulièrement chanceux d'avoir pu travailler avec eux et d'avoir pu échanger divers moments et discussions chaleureux et personnels à Clermont-Ferrand comme à Francfort durant ces 3 années. Aussi, je souhaiterais adresser un remerciement à Benoît Thuilot pour son accompagnement académique formidable durant mes années universitaires qui a pu me mener à cette thèse.

Enfin, je remercie sincèrement tous mes collègues et amis de l'équipe ComSee du laboratoire, ainsi que ceux en dehors, pour leur soutien et leur présence dans cette étape importante de ma vie. Des remerciements particuliers pour Thomas, Carlos, Sara, Mathieu, Eliana et Priyanka. Bien entendu, j'émets des remerciements inconditionnels envers ma famille: mon père, ma mère et ma soeur aînée. Un merci à Nagatsuki Tappei-*sensei* pour la rédaction du *light novel* "ReZero" dont la lecture m'a incroyablement soutenu pendant la thèse. Pour conclure, un remerciement à toute personne que je n'ai pu mentionner ici par souci d'espace, ainsi qu'au présent lecteur pour avoir daigné prendre le temps de jeter un coup d'oeil à ma thèse.

ABSTRACT

Current artificial neural networks based on deep learning are facing a number of challenges in terms of energy consumption and adaptability. A new category of work called “neuromorphic” has come to light in the past decades. It addresses the issues encountered by deep learning models by proposing software and hardware of low energy consumption by mimicking the brain’s information coding via discrete voltage spikes. However, the design of such spiking neural networks, despite their low energy consumption faces many challenges due to the significant difference in information coding strategies.

In the context of computer vision in particular, new sensors named “event cameras” have recently been commercialized and exhibit many desirable qualities for various applications (e.g. object tracking, robot control) with a high dynamic range in the order of 120 dB, a low energy consumption in the order of milliwatts, a low latency and fast acquisition rate, both in the order of microseconds. Essentially, pixels in such sensors respond to brightness changes by emitting asynchronous events whenever brightness increases (or decreases) above (or below) a threshold. Despite their remarkable properties, the asynchronous nature of event camera’s data make them complex to use with current neural networks as they cannot leverage their diverse advantages in terms of energy efficiency and fast processing.

In this thesis, we tackle this issue by taking further inspiration from information processing in the brain. Notably, the brain is known to learn different visual features with little to no supervision, contrary to artificial neural networks that owe their success to supervised learning. One of the key mechanisms by which the brain learns without supervision is believed to be spike timing-dependent plasticity (STDP), by which biological neurons increase the strength of those synapses that receive inputs inputs shortly *before* the neuron is activated, and decrease the strength of synapses which receive inputs shortly *after* the neuron’s activation. Interestingly, STDP is fully local and does not require any form of backpropagation to achieve unsupervised learning. In addition to this unsupervised learning scheme, we also consider in our work, arguably the most eminent theory of brain function of this early 21st century, *predictive coding*. This theory postulates that neurons in the brain strip away redundancy from their responses, to only send out the unpredictable components of their responses.

Combining STDP and predictive coding ideas, we propose two spiking frameworks: the Predictive Coding Light (PCL) and the Spiking Predictive Recurrent Inhibitory and Excitatory Recall (SPRIER) networks. Those two frameworks are built to receive inputs from an event camera and, due to their spiking nature, can leverage the advantages of the sensor, compared to current artificial neural networks.

In our work, we propose an alternative to conventional predictive coding, and offer a concrete implementation in an artificial spiking neural network. Instead of comparing bottom-up inputs to top-down predictions and transmitting only prediction errors to higher levels, our networks learn to suppress the most predictable spikes via recurrent and top-down inhibitory connections and pass a compressed representation of the sensory input to higher processing stages.

We demonstrate first with PCL that STDP and the predictive coding ideas described above, can be used to learn *without supervision* different visual features that are reminiscent of the ones exhibited by simple and complex cells in the primary visual cortex (V1) of mammals.

Second, we show that PCL achieves a remarkable reduction in energy consumption through its predictive coding scheme, by the reproduction, again without supervision, of extra-classical receptive field effects observed in V1. We also show how this can be applied to challenging classification tasks and demonstrate a significantly reduced energy consumption with a high classification accuracy.

Third, we learn invariant representations with deeper PCL networks and test different projection types observed in the visual cortex of mammals, to assess their functional role in the emergence of invariance properties in the visual system.

Lastly, with SPRIER we extend the PCL network to learn, again without supervision by using *local* STDP rules, how to represent a memory of past inputs (similar to so-called recurrent neural networks in deep learning). We show the correspondence between SPRIER’s neurons’ responses and biological responses, and demonstrate that it can be applied to neuromorphic vision datasets as well.

Keywords: predictive coding, spiking neural network, unsupervised representation learning, working memory, neuromorphic vision

RÉSUMÉ

Les réseaux de neurones artificiels actuels basés sur l'apprentissage profond sont confrontés à un certain nombre de défis en termes de consommation énergétique et d'adaptabilité. Une nouvelle catégorie de travaux appelée « neuromorphique » a vu le jour au cours des dernières décennies. Elle aborde les problèmes rencontrés par les modèles d'apprentissage profond en proposant des logiciels et architectures matérielles à faible consommation énergétique qui imitent le codage d'information du cerveau en utilisant des données impulsionnelles discrètes qui dépendent de potentiels neuronaux. Cependant, la conception de tels réseaux impulsionnels, malgré leur faible consommation énergétique, est confrontée à de nombreux défis en raison des différences significatives de stratégies de codage d'information.

Dans le contexte de la vision par ordinateur en particulier, de nouveaux capteurs appelés « caméras à événements » ont été récemment commercialisés et présentent diverses qualités souhaitables pour plusieurs applications (par exemple, le suivi d'objets et la commande de robots). Ces avantages sont notamment: une plage dynamique large de l'ordre de 120 dB, une faible consommation énergétique de l'ordre du milliwatt, un faible temps de latence et un taux d'acquisition rapide, tous deux de l'ordre des microsecondes. Essentiellement, les pixels de ces capteurs réagissent aux changements de luminosité en émettant des événements asynchrones chaque fois que la luminosité augmente (ou diminue) au-dessus (ou en-dessous) d'un seuil. Malgré leurs propriétés remarquables, la nature asynchrone des données des caméras à événements les rend difficiles à utiliser avec les réseaux de neurones actuels, car ils ne peuvent pas tirer parti de leurs divers avantages en termes de faible consommation énergétique et de rapidité de traitement.

Dans cette thèse, nous nous attaquons à ce problème en nous inspirant du traitement d'information du cerveau. En particulier, le cerveau est connu pour apprendre différentes caractéristiques visuelles avec peu voire aucune supervision, contrairement aux réseaux de neurones artificiels qui doivent leur succès à l'apprentissage supervisé. L'un des principaux mécanismes connus permettant au cerveau d'apprendre sans supervision est la plasticité fonction du temps d'occurrence des impulsions (STDP en anglais), par laquelle les neurones biologiques augmentent la force des synapses qui reçoivent des entrées peu *avant* l'activation du neurone, et diminuent la force des synapses qui reçoivent des entrées peu *après* l'activation du neurone. Il est

intéressant de noter que la STDP est entièrement locale et ne nécessite aucune forme de rétropropagation pour réaliser un apprentissage non supervisé. Outre ce schéma d'apprentissage non supervisé, nous prenons également en compte dans notre travail la théorie sans doute la plus éminente du fonctionnement du cerveau de ce début de 21^{ème} siècle, le *codage prédictif*. Cette théorie postule que les neurones du cerveau éliminent la redondance de leurs réponses, pour n'envoyer que les composantes imprévisibles de leurs réponses.

En combinant les idées de STDP et de codage prédictif, nous proposons deux *frameworks* : les réseaux Predictive Coding Light (PCL) et Spiking Predictive Recurrent Inhibitory and Excitatory and Recall (SPRIER). Ces deux réseaux sont conçus pour recevoir des entrées d'une caméra à événements et, en raison de leur caractère impulsionnel, ils peuvent tirer parti des avantages du capteur par rapport aux réseaux de neurones artificiels actuels.

Dans notre travail, nous proposons une alternative aux approches conventionnelles de codage prédictif et offrons une implémentation dans un réseau de neurones impulsionnels. Plutôt que de comparer les entrées ascendantes aux prédictions descendantes et de transmettre uniquement les erreurs de prédiction aux couches supérieures, nos réseaux apprennent à supprimer les impulsions les plus prévisibles par le biais de connexions inhibitrices récurrentes et descendantes. Nos réseaux transmettent donc une représentation compressée de l'entrée sensorielle aux couches supérieures.

Tout d'abord, nous démontrons avec PCL que la STDP et les idées de codage prédictif décrites ci-dessus, peuvent être utilisées pour apprendre *sans supervision* différentes caractéristiques visuelles similaires aux caractéristiques visuelles dont sont sensibles les cellules simples et complexes du cortex visuel primaire (V1) des mammifères.

Deuxièmement, nous montrons que PCL permet une réduction remarquable de la consommation énergétique grâce à son principe de codage prédictif, par la reproduction, toujours sans supervision, des effets de champ récepteur non-classique observés dans V1. Nous montrons également comment cela peut être appliqué à des tâches de classification difficiles et démontrons une réduction significative de consommation énergétique de PCL avec une conservation d'un taux de performance élevé de classification.

Troisièmement, nous apprenons des représentations invariantes avec des réseaux PCL plus profonds et nous testons différents types de projection observés dans le cortex visuel des mammifères, afin d'évaluer leur rôle fonctionnel dans l'émergence des propriétés d'invariance du système visuel.

Enfin, avec SPRIER, nous étendons le réseau PCL pour qu'il apprenne, toujours sans supervision en utilisant des règles *locales* STDP, comment représenter une

mémoire des entrées du réseau (par analogie aux réseaux de neurones récurrents en apprentissage profond). Nous montrons la correspondance entre les réponses des neurones de SPRIER et les réponses biologiques, et nous démontrons que le réseau peut également être appliqué à des bases de données de vision neuromorphique.

Mots-Clés : codage prédictif, réseau de neurone impulsionnel, apprentissage non supervisé de représentations, mémoire de travail, vision neuromorphique

CONTENTS

Remerciements	iii
Abstract	v
Résumé	vii
List of Figures	xvii
List of Tables	xxi
Glossary	xxiii
Acronyms	xxv
1 General introduction	3
1.1 Motivation	4
1.2 Context	6
1.3 Contributions	7
1.4 Manuscript outline	9
2 Background	11
Overview	12
2.1 Neuromorphic vision	12
2.1.1 Biological vision and spiking neural networks	12
2.1.2 Spike Timing-Dependent Plasticity	19
2.1.3 Spiking models of V1	21
2.1.4 Neuromorphic sensors and circuits	21
2.2 General Predictive Coding	23
2.2.1 Concept and historical background	24

2.2.2	Analogy with efficient coding	28
2.2.3	Predictive coding and the brain	30
	Conclusion	32
3	Spiking Predictive Coding Review	33
	Overview	34
3.1	Challenges	34
3.2	Historical Antecedents and Early Works	36
3.2.1	Early work by Ballard, Rao, and collaborators.	36
3.2.2	Early work by Denève.	37
3.2.3	Conclusion	37
3.3	Spiking predictive coding with explicit error neurons (Class 1)	38
3.3.1	Mismatch negativity with Spiking PC.	38
3.3.2	SNN-PC: Spiking Neural Network for Predictive Coding. . . .	39
3.3.3	Conclusion.	40
3.4	Spiking predictive coding with membrane potential error encoding (Class 2): Spike Coding Networks	40
3.5	Spiking predictive coding with implicit error encoding (Class 3) . . .	42
3.5.1	Efficient Coding through Suppression.	42
3.5.2	Prediction Errors through Inhibitory Plasticity.	43
3.5.3	Predictions and Prediction Errors through Learning Rule and Correction Neurons.	44
3.5.4	Conclusion.	45
3.6	Neuromorphic applications of predictive coding	46
3.6.1	Implementations of Spiking Predictive Coding on Hardware and for Robotics.	46
3.6.2	Conclusion.	48
3.7	Conclusion	48
4	Predictive Coding Light	51

Overview	52
4.1 Introduction	53
4.2 Methods	54
4.2.1 Network architecture	54
4.2.2 Synaptic plasticity mechanisms	56
4.2.3 Network parameters	59
4.3 How inhibitory STDP removes redundant spikes	62
4.3.1 Experimental protocol	63
4.3.2 Results	63
4.4 How PCL learns simple and complex cell-like responses	64
4.4.1 Experimental protocol	65
4.4.2 Results	67
4.5 PCL exhibits efficient information coding on digit and gesture datasets	76
4.5.1 Experimental protocol	77
4.5.2 Results	78
4.6 Discussion	85
Conclusion	86
5 Deep Predictive Coding Light	89
Overview	90
5.1 Introduction	90
5.2 How to achieve spatial invariance with the PCL network	91
5.2.1 Dataset generation and network architecture	92
5.2.2 The PCL network requires specific neuronal projections to learn invariant object representations	95
5.2.3 Residual connections improve classification accuracy when used in intermediate layers	100
5.3 Discussion	103
5.4 Conclusion	103

6	SPRIER	105
	Overview	106
6.1	Introduction	107
6.2	Methods	109
6.2.1	From PCL to SPRIER	109
6.2.2	New synaptic plasticity mechanisms	110
6.3	Sensory learning effects	113
6.3.1	Impact of synaptic delays	113
6.3.2	Impact of recurrent excitation	115
6.4	Learning spatiotemporal sequences of gratings	118
6.4.1	Reproducing V1-like effects	118
6.4.2	Impact of the synaptic delays	119
6.5	Learning to predict gestures	121
6.6	Discussion	123
	Conclusion	125
7	Discussions and Perspectives	127
7.1	Conclusions and discussions	128
7.2	Perspectives on improvements	129
7.2.1	Neurobiological aspects	129
7.2.2	Application aspects	130
7.3	Perspectives on future applications	131
A	Supplementary material	135
B	Publications and communications	137
C	Résumé étendu en Français	139
	Introduction	139
C.1	Motivation	139
C.2	Contexte	142

C.3 Contributions	143
C.4 Résumé des chapitres	145
Bibliography	149

LIST OF FIGURES

2.1	Processing streams of visual perception.	12
2.2	Retinal processing	13
2.3	The primary visual cortex	14
2.4	Bank of Gabor filters.	15
2.5	V1 cell's orientation tuning curve.	15
2.6	Sinusoidal gratings.	16
2.7	Surround suppression	16
2.8	Biological neuron.	17
2.9	LTP and LTD mechanisms.	19
2.10	Event camera and standard camera output.	22
2.11	Illustration of Predictive Coding.	24
2.12	The dress.	25
2.13	The free-energy principle.	25
2.14	Principle of efficient coding.	28
2.15	A canonical microcircuit for predictive coding.	30
3.1	Classes of spiking predictive coding models.	35
3.2	History of spiking predictive coding models.	36
3.3	Cognitive and spatial organization of predictive coding models.	48
4.1	Predictive Coding Light (PCL) network.	54
4.2	Spike Timing-Dependent Plasticity rules.	56
4.3	Toy example on how inhibitory STDP removes the most predictable spikes.	62

4.4	Development of simple and complex cell-like receptive fields in the PCL network.	68
4.5	Quantification of simple and complex cells.	69
4.6	Influence of the absence of a causal LTP on the emergence of simple cells in a PCL network.	70
4.7	Influence of recurrent connections and membrane time constants on the emergence of complex cells in a PCL network.	71
4.8	Different STDP rules allow the emergence of simple and complex cell-like response properties in a PCL network.	73
4.9	PCL gives rise to surround suppression effects as observed in primary visual cortex.	75
4.10	Complex cells replicate some suppressive effects.	76
4.11	Evolution of the classification accuracy of simple and complex cells on N-MNIST and gestures during training.	78
4.12	PCL achieves substantial reduction of spiking with only small to moderate loss of downstream task performance.	79
4.13	Classification based on simple and complex cell activity.	79
4.14	PCL's simple cells show increased population sparseness.	81
5.1	The v2e simulator.	92
5.2	3-layer Predictive Coding Light (PCL) networks.	95
5.3	Sensory learning in 3-layer PCL networks	97
5.4	Experimental protocol for the classification of digits with different 3-layer PCL networks	98
5.5	4-layer Predictive Coding Light (PCL) network.	102
6.1	Spiking patterns of a SPRIER network.	107
6.2	Network architecture.	109
6.3	Effects of feedforward synaptic delays on sensory learning in the SPRIER network.	114
6.4	Effect of synaptic delays on simple cells' weights.	115

6.5	Effects of synaptic delays on the emergence of suppression in a SPRIER network.	116
6.6	Impact of recurrent excitation on sensory learning in a SPRIER network.	117
6.7	The SPRIER network learns sequences of counterphase sinusoidal gratings.	118
6.8	Effect of different recurrent excitatory delays on the emergence of V1-like memory effects.	120
6.9	SPRIER learns to predict different gestures through its spontaneous activity.	121
6.10	Spiking representations of the gestures for different SPRIER networks.	123

LIST OF TABLES

4.1	Connectivity, neurons' membrane potential and plasticity parameters of the PCL network.	60
4.2	Performance metrics of simple and complex cells for all parameters. .	73
4.3	Influence of different time bins on the classification accuracy of the PCL network on the N-MNIST dataset and on the DVS128 Gesture dataset.	80
4.4	Statistical significance of PCL's population sparseness for different gestures.	82
4.5	Classification accuracy of different spiking neural network models on digits datasets.	83
4.6	Classification accuracy of different spiking neural network models on the DVS128 Gesture dataset.	84
5.1	Connectivity, neurons' membrane potential and plasticity parameters of a 3-layer PCL network.	96
5.2	Comparison of different 3-layer deep PCL network architectures . . .	99
5.3	Connectivity, neurons' membrane potential and plasticity parameters of a 4-layer PCL network.	101
5.4	Comparison of different 4-layer deep PCL network architectures . . .	102
6.1	Neurons' membrane potential and plasticity parameters of the SPRIER network.	111
A.1	Influence of top-down on sensory learning in deep PCL networks . . .	135

GLOSSARY

AMPA The α -amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid receptor (AMPA receptor) is a structure that controls excitation in biological neurons. [39](#)

conditional probability The conditional probability refers to a measure of the probability of an event A happening given that another event B has already happened. [26](#)

cortical columns It consists of the grouping of cortical neurons with similar tuning properties into a cylindrical structure. Cortical columns are observed throughout the mammalian's cortex. [32](#), [95](#)

Dale's law It is a principle that states that biological neurons can only perform either excitation or inhibition, and not both. [55](#), [85](#), [86](#), [110](#), [128](#), [129](#)

eye-blink classical conditioning It is a type of conditioning that pairs an auditory or visual stimulus with a second stimulus that should elicit an eyeblink in an individual. [47](#)

feedback Feedback corresponds to the reciprocal responses that a deeper neuron sends out to a superficial neuron. [6](#), [7](#), [14](#), [15](#), [16](#), [19](#), [31](#), [37](#), [38](#), [39](#), [40](#), [44](#), [45](#), [46](#), [54](#), [55](#), [108](#), [128](#), [143](#)

GABA The gamma-aminobutyric acid receptor (GABA receptor) is a structure that controls inhibition in biological neurons. [39](#)

hidden causes Hidden causes are the set of parameters that are responsible for the emergence of a given output. [37](#), [47](#)

joint probability The joint probability refers to a measure of the probability of two events A and B happening simultaneously. [26](#), [27](#)

log-odds It is the logarithm of the odds ratio, where the odds-ratio is a statistic measure to quantify how strongly two different quantities are related to each other. [37](#)

marginal The marginal distribution is the probabilities of the different values that a variable can take. [26](#), [27](#)

- MEG** Magnetoencephalography (MEG) refers to a neuroimaging technique to measure magnetic fields produced by the brain's electrical currents. [39](#)
- mismatch negativity** It is an effect observed in the brain in which the response of a neuron changes when it is stimulated by a deviant stimulus after stimulation with a sequence of the same stimulus over a long time. [xii, 29, 33, 38](#)
- NMDA** The N-methyl-D-aspartate receptor (NMDA receptor) is another structure, as AMPA receptors, that controls excitation in biological neurons. [39](#)
- oscillations** Oscillations are repetitive patterns observed in neurons responses. They happen at the neuronal population level at different frequency bands (e.g. alpha frequency band or gamma frequency band) across different areas of the cortex. [18, 37](#)
- posterior probability** The posterior probability refers to the probability that a variable has a certain value, taking into account new data observations and the prior. [26, 27](#)
- prior probability** The prior probability refers to the probability that a non-observed variable has a certain value. [26, 27](#)
- rate-based** It is the act of computing information with neurons according to the continuous quantity represented by the activation value of the neurons. This activation value represents the time-averaged firing rate of neurons. [12, 23, 34, 38](#)
- receptive field** In neuroscience, a receptive field refers to the spatial and temporal regions that effectively causes a neuron to give a response. [6, 13, 15, 16, 21, 27, 30, 37, 52, 53, 54, 55, 60, 61, 62, 65, 66, 67, 68, 69, 70, 72, 73, 74, 76, 90, 91, 96, 97, 98, 99, 100, 101, 113, 114, 115](#)
- suppression** It is a biological neural processing phenomenon in which the response of a single neuron or a population of neurons is reduced, either in time or in space. [xii, xvii, xviii, xix, 8, 9, 16, 29, 31, 33, 42, 43, 52, 53, 54, 65, 66, 74, 75, 76, 85, 114, 116, 124, 144, 146](#)
- surround** The surround of a particular neuron refers to the set of neurons that are spatially adjacent to this neuron. The surround can be quantified according to precise spatial ranges. [xvii, xviii, 8, 9, 16, 29, 38, 52, 53, 66, 74, 75, 76, 85, 114, 116, 124, 144, 146](#)
- tuning curve** It refers to the response of neuron as a function of parameters of a specific stimulus parameter. [65, 68, 71](#)

ACRONYMS

- AdEx** Adaptive exponential integrate-and-fire 39
- AEC** Active Efficient Coding 6, 7, 131, 132
- aSTDP** Acausal Spike Timing-Dependent Plasticity 56, 57, 72
- cSTDP** Causal Spike Timing-Dependent Plasticity 56, 57, 69, 70, 72, 78, 95
- ELBO** Evidence Lower Bound 25
- ICA** Independent Component Analysis 29, 36
- IT** Inferior Temporal area 14, 90, 91, 92, 95, 98, 100, 103, 130
- L4** Layer 4 14, 15, 31
- LGN** Lateral Geniculate Nucleus 13, 14, 15, 36
- LIF** Leaky Integrate-and-Fire 17, 45, 55, 103
- LTD** Long Term Depression 19, 20
- LTP** Long Term Potentiation xviii, 19, 20, 57, 58, 60, 70, 74, 96, 101, 111, 112
- MT** Middle Temporal area 14, 90
- PC** Predictive Coding xii, 33, 38, 39, 40, 48, 49, 50
- PCA** Principal Component Analysis 36
- PCL** Predictive Coding Light vi, viii, xiii, xvii, xviii, xxi, 43, 51, 52, 53, 54, 55, 57, 58, 59, 61, 64, 65, 66, 67, 68, 69, 70, 71, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 85, 86, 87, 89, 90, 91, 92, 94, 95, 96, 97, 98, 100, 101, 102, 103, 106, 108, 109, 110, 118, 122, 123, 124, 125, 128, 129, 130, 131, 135
- SCN** Spike Coding Network 41
- SPRIER** Spiking Predictive Inhibitory and Excitatory Recall vi, viii, ix, xviii, xix, xxi, 106, 107, 108, 109, 110, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 128, 129, 131
- sSTDP** Symmetric Spike Timing-Dependent Plasticity 56, 57, 72, 112
- STDP** Spike Timing-Dependent Plasticity v, vi, vii, viii, xiii, xvii, xviii, 5, 6, 9, 19, 20, 21, 23, 32, 37, 39, 43, 44, 45, 46, 47, 51, 52, 53, 56, 57, 58, 61, 62, 63, 64, 67, 70, 71, 72, 73, 74, 85, 86, 90, 91, 95, 100, 103, 108, 110, 119, 123, 128, 129, 130, 146
- V1** Primary visual cortex vi, viii, xiv, xix, 6, 9, 10, 13, 14, 15, 16, 21, 36, 53, 55, 59, 64, 72, 73, 74, 75, 85, 90, 92, 100, 105, 106, 118, 119, 120, 123, 124, 128, 146

Even though it's filled with smoke,
We can still see the sky,
We can still see the ocean,
This place isn't Hell.

– *Usopp from the manga "One Piece"*
by Eiichiro Oda

GENERAL INTRODUCTION

1.1	Motivation	4
1.2	Context	6
1.3	Contributions	7
1.4	Manuscript outline	9

1.1 Motivation

The brain is a remarkable organ that has been optimized by million years of evolution. Its development has been key to the survival of the human species as it allowed us to perceive the world through our different senses and make decisions based on our prior experiences. For instance, in the case of vision, while the retina allows us to see the world by sensing light and converting it into signals that can be interpreted by the brain, the retina by itself cannot explain perception. Without any further analysis of the retinal signals, we would be incapable of recognizing, segmenting or tracking objects – that is, we would not be able to *perceive* the world. This applies not only to vision but also to other senses such as hearing, touch and smell. The brain is essentially the key organ that leverages sensory stimuli to create perception. Fundamentally, through perception, we are able to interpret the world to then interact with it through action. Considering how crucial it has been to our survival and our everyday life, scientists have been questioning for centuries, and even millennia, what could be the mechanisms behind the emergence of perception.

Neurons are cells that populate the brain and have been understood as being its primary element of computation. Understanding how neural computation is done should help us also comprehend how the brain generates perception from sensory stimuli. The ideas behind different theories of sensory coding and perception such as efficient coding [20] and predictive coding [195] have been around for decades now and can be tested in computer simulations of neural networks to guide neuroscientific research and reach an understanding on how perception occurs. Furthermore, science has recently entered a new era in which the design of intelligent systems is at the center of research across different fields (e.g. cognitive science, biotechnology, medical imaging, computer science, robotics). Therefore, taking inspiration from the most intelligent system that we know, the brain, to build new intelligent systems should lead to scientific advances and allow the development of machines that outperform existing ones.

Current state-of-the-art models of machine perception consist of deep learning approaches [136] and actually find their origins in models that attempt to simulate brain processing. Deep learning has now revolutionized perception algorithms and is used with software apps to assist humans in cognitive tasks (e.g. chatbots, surveillance softwares, cancer diagnosis, etc.) but it is also deployed in mechanical systems such as robots to directly aid humans in physical tasks (e.g. robotic-assisted surgery, autonomous rescue robots, robotic-powered prostheses, etc.). Through the combination of control and deep learning algorithms, machines are thus reaching new heights. However, deep learning suffers from crucial well-known challenges such as its:

- High energy consumption: New models are based on complex network architectures which require a lot of energy for both training and inference [189].
- Lack of explainability: Deep learning owes its results to the backpropagation algorithm which minimizes a given error function, and it is unclear what the underlying policy under which a deep learning agent gives an output actually is.
- Low adaptability: Most models are trained for specific tasks, which lead to the supervised training of entirely new algorithms for different tasks instead of an end-to-end adaptive (and unsupervised) training. Though a new type of model named foundation models [35] have been proposed to address this issue, they still require to be later adapted to different tasks through a supervised fine-tuning procedure.

Considering those limitations, machine perception algorithms are at a stage where it would be interesting to increase the level of biological inspiration. One of the most distinct differences between the brain and deep learning is the reliance on continuous signals in deep learning as opposed to asynchronous spiking signals observed in the brain. Taking more inspiration from the brain would thus demand to move the current artificial neurons to the spiking domain with so-called spiking neural networks [232]. While such a change in computation is complex, the advantages may outweigh the disadvantages. For instance, the brain uses only about 20 W of energy [199]. Lastly, the brain is highly adaptive as it can perform a wide variety of tasks with no supervision. Essentially, research on brain-inspired approaches would in the long term lead to significant improvements on the capabilities of current machines, while also increasing our understanding of the neural mechanisms behind perception. Different research labs have already started working on such brain-inspired spiking neural network models [171] and notable advancements have also been made on brain-inspired hardware with the creation of the “silicon retina” commonly known as event camera now [80]. Such parallel, but complementary, software and hardware research is crucial for novel brain-inspired machines to come to light in the future.

Working with spiking neurons involve looking at computation in an entirely different way. The brain tends to use two different types of neural communication processes: excitatory and inhibitory. Excitatory neurons would, as their name indicate, excite other neurons – that is, increase their responses. On the other hand, inhibitory neurons would inhibit other neurons – that is, reduce their responses. In order to build spiking systems capable of perception, it is necessary to study how such excitatory-inhibitory computations give rise to brain-like response properties. Spike timing-dependent plasticity (STDP) [27] is a good candidate for the emergence of such perceptual patterns as this synaptic plasticity mechanism has been widely observed in different cortical areas. However, the use of STDP and excitatory-inhibitory networks is subject to multiple important questions such as: 1) the role of each connection

in the emergence of perception, 2) the neural architecture required, 3) the balance between excitation and inhibition for stable neural activity. While those questions have puzzled neuroscientists for a long time now, it is only in the past decade that the interplay of excitatory and inhibitory plasticity has started being studied in networks of spiking neurons [11], [38], [99], [245]. When taking the example of vision, the use of inhibition for perceptual learning (e.g. visual cortex-like receptive fields) in deep spiking neural networks has shown promising results [121], [122], [155]. Those works however only use excitatory plasticity, and keep the inhibition fixed. Furthermore, the brain demonstrates a remarkable recurrent and feedback connectivity, which has thus not been evaluated in spiking networks for visual perception.

In this thesis, we are thus interested in developing visual cortex-inspired spiking networks that learn to perform visual perception. Our findings suggest that different brain response patterns find their origin in excitatory and inhibitory STDP, as well as different feedback pathways. In addition to those neuroscientific observations, we also make a link between brain responses and applied tasks such as object and motion recognition and highlight the advantages of the feedback pathways in the brain’s information processing and energy coding. Our framework is designed to be compatible with neuromorphic hardware which also mimics the brain, and thus proposes another view on artificial brain-inspired computing systems for machine perception.

1.2 Context

In the past years, a framework named “Active Efficient Coding” (AEC) was developed [63], [236]. This model rests on the hypothesis that biological systems learn motor behaviors, i.e., actions that allow for *efficient* sensory coding. An efficient sensory coding would for instance point to a few number of active neurons paired with a low response activity, such that information transmission is optimal. Models of AEC aim at learning autonomous agents that can perform smooth pursuit eye movements as well as eye vergence for binocular vision. Those models have however been developed with standard artificial neural networks and not spiking networks.

This thesis’ work aims at moving the AEC framework to the spiking domain using inputs from an event camera. A first thesis had been realized on the subject by Barbier [16] from October 2019 to January 2023. In that thesis, a spiking neural network library, preliminary results on the development of V1-like s (receptive fields) as well as the first version of a spiking AEC model [19] had been proposed.

Importantly, this thesis constitutes the second thesis on this body of work and builds on Barbier’s work, by adding new functionalities to the spiking neural network

library as well as testing and validating other hypotheses (here, predictive coding) of brain-like perception in vision using event-based natural image stimuli as well as neuromorphic vision datasets. This work demonstrates in a spiking neural network the neuroscientific validity as well as the computer vision capabilities on which the AEC hypothesis is built upon. This thesis thus poses a significant groundwork for future research on spiking AEC for robotics with event camera.

1.3 Contributions

In this thesis, we propose two spiking neural network models: the *Predictive Coding Light* model and the *Spiking Predictive Recurrent Inhibitory and Excitatory Recall* model. We show that different brain-like response properties emerge within those models after exposition to event-based natural image stimuli as well as spatiotemporal event-based sequences. All the synaptic connections of the two models are learnt using spike timing-dependent plasticity rules, and we show that feedback pathways coupled with plastic inhibition and excitation allow for the development of visual perception and for the management of energy resources in our brain-inspired networks.

Our contributions are as follows:

Review of spiking predictive coding models - Published as [171] and currently undergoing review in a peer-reviewed journal.

We review different spiking predictive coding models [171] which all targeted widely different applications and had thus been scattered across different research fields, which made the identification of open research questions complicated. In this review, we thus surveyed historical and present research efforts on the realization of predictive coding with spiking neural networks, separated them according to a common notion that was recurring amongst all predictive coding methods — the prediction-error. This review represents, in our opinion, a valuable new contribution to the community to guide and encourage future research on spiking neural networks for predictive coding.

The Predictive Coding Light model - *Published partially as [173] and [172].*

With Predictive Coding Light, we challenge current models of predictive coding and propose a new fully plastic model of spiking predictive coding for low-energy visual perception. Predictive Coding Light learns simple cell and complex cell-like response properties after being trained on natural images using spike timing-dependent plasticity. Predictive Coding Light further exhibits surround suppression, orientation-tuned and cross-orientation suppression effects, which are widely accepted to be signatures of efficient codes in predictive coding networks [23]. Also, Predictive Coding Light showcases the effect of recurrent and top-down inhibitory connections in the learning of efficient V1-like visual representations and in the realization of downstream task such as digit and gesture classification. Those results notably highlight the role of inhibition in saving energy by discarding highly predictable information in the neural code. Finally, Predictive Coding Light learns invariant visual representations in deep networks of different layers and demonstrates the influence of different projection patterns in the emergence of invariance when learning feature hierarchies.

The Spiking Predictive Recurrent Inhibitory and Excitatory Recall model

With the Spiking Predictive Recurrent Inhibitory and Excitatory Recall model, we extend Predictive Coding Light to include synaptic delays, distant lateral excitation and top-down excitation in its architecture. We introduce new mechanisms to ensure network stability and reproduce biological findings on learning of spatiotemporal sequences of gratings as well as test the network on gestures to validate the learning of more complex sequences. The learning of spatiotemporal sequences in the Spiking Predictive Recurrent Inhibitory and Excitatory Recall network is attributable to the inhibitory and excitatory spike timing-dependent plasticity and synaptic delays. Our results give an insight on how recurrent excitatory pathways might lead to the formation of memory in the brain.

Different publications, submissions for publication, oral communications in international and national workshops as well as attendance to international workshops were realized during this thesis. Details on these communications are given in Appendix B. Our source code (C++ and Python) and datasets are available on <https://github.com/comsee-research/Predictive-Coding-Light> and <https://zenodo.org/records/16813455>

1.4 Manuscript outline

This thesis follows this organization:

Chapter 2 provides a general overview on the different concepts discussed in this thesis. The chapter first introduces concepts of neuromorphic vision by covering neuroscientific knowledge required for the implementation of vision-based spiking networks and briefly presents neuromorphic sensors and hardware. Second, the chapter details the concepts of the predictive coding theory by introducing the principle of predictive coding, its link with the efficient coding hypothesis as well as the relationship between predictive coding and the brain.

Chapter 3 reviews different models of predictive coding with spiking neural networks. It proposes a separation of spiking predictive coding models into three different categories of works based on the way they encode prediction-errors: 1) using explicit prediction-error neurons, 2) using the membrane potential to encode prediction-errors and 3) using an implicit encoding of the prediction-errors in the neural activities.

Chapter 4 presents our spiking neural network framework *Predictive Coding Light*. The chapter describes how inhibitory and excitatory STDP can be used to learn simple and complex cell-like representations and compare the results to biological findings made in mammals' V1. It details the conditions under which simple cells and complex cells emerge and compares the effect of different spike timing-dependent plasticity rules in the emergence of visual cortex-like features. Next, the chapter demonstrates that surround suppression, orientation-tuned suppression and cross-orientation suppression effects observed in mammals' V1 all emerge in the Predictive Coding Light network. Lastly, the chapter presents how a Predictive Coding Light network can be used to perform object classification and show the effects of suppression which reduces the number of spikes (and therefore energy consumption) in the network while leading to only a minor drop in classification accuracies.

Chapter 5 evaluates different deeper instances of the Predictive Coding Light network on invariant representation learning. The chapter presents the effect of different type of projection patterns, e.g. residual connections or one-to-one “pooling-like” feature projections, and shows that depending on a layer's depth as well as its

network's input zone, a mixture of projection patterns are required to learn invariant visual representations in a hierarchical spiking neural network.

Chapter 6 describes a new framework called the *Spiking Predictive Recurrent Inhibitory and Excitatory Recall* network which builds on the Predictive Coding Light network. This network proposes distant lateral excitation, top-down excitation, synaptic delays and new synaptic learning mechanisms to learn using spike timing-dependent plasticity spiking networks that exhibit memory capabilities. The chapter shows how different learning approaches with synaptic delays and recurrent excitation affect sensory learning. Following this, it presents the results in the framework in the reproduction of different biological findings on temporal sequences learning made in mice V1. Finally, the chapter shows that the network can also be used to learn complex spatiotemporal sequences such as gestures.

Chapter 7 reviews the different contributions of this thesis with the frameworks: the *Predictive Coding Light* network and its extension the *Spiking Predictive Recurrent Inhibitory and Excitatory Recall* network. The chapter summarizes how the two frameworks can be used to reproduce many biological response properties as well as perform downstream tasks, using brain-inspired connectivity and learning mechanisms. Finally, the chapter lists a number of perspectives on improvements as well as future applications of this thesis.

BACKGROUND

Overview	12
2.1 Neuromorphic vision	12
2.1.1 Biological vision and spiking neural networks	12
2.1.2 Spike Timing-Dependent Plasticity	19
2.1.3 Spiking models of V1	21
2.1.4 Neuromorphic sensors and circuits	21
2.2 General Predictive Coding	23
2.2.1 Concept and historical background	24
2.2.2 Analogy with efficient coding	28
2.2.3 Predictive coding and the brain	30
Conclusion	32

Overview

In this chapter, we will be giving the theoretical background upon which this thesis is based. We will first explore in Sec. 2.1 the different areas of neuromorphic vision, by reviewing their biological correlates, their current algorithmic principles and offer an insight on the available sensors and hardware. Second, we will introduce in Sec. 2.2 the predictive coding theory and its underlyings in rate-based networks.

2.1 Neuromorphic vision

This section introduces the background behind neuromorphic vision (computer vision with so-called event-based/neuromorphic cameras) with a strong highlight on spiking neural network-based solutions. We will first describe how visual and neuronal processing are thought to occur in the biological brain and then present the spike timing-dependent plasticity learning rule used in brain-inspired learning approaches. Finally, we will review spiking works that model biological vision and briefly showcase current neuromorphic sensors and hardware.

2.1.1 Biological vision and spiking neural networks

2.1.1.1 Visual pathway

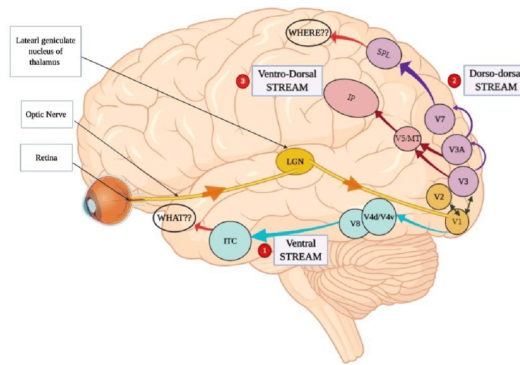


Figure 2.1: **Processing streams of visual perception** from [2].

Vision is a complex perception phenomenon that occurs thanks to the intricate successive integration of signals at different superficial and deep layers of the brain to extract information on object representations and motions. In this thesis, we study

different models inspired by the biological vision system of mammals. Figure 2.1 gives an illustration of visual processing streams.

In the biological vision system of mammals, light is received by the photoreceptors of the retina. Different types of retinal cells process light at different intensities and different wavelengths of lights (rod and cone photoreceptors). The responses of those photoreceptors are transmitted to the so-called retinal ganglion cells via bipolar, horizontal and amacrine cells.

When the intensity of the light received by the photoreceptors gradually increases, some bipolar cells are active and are called “ON” bipolar cells, whereas in case of decrease of light, other bipolar cells called “OFF” bipolar cells are instead active. This response pattern depends on the light intensity change in the scene and is allowed by an intricate excitatory and inhibitory synaptic connectivity between bipolar, amacrine and horizontal cells. See Fig. 2.2 for an illustration of retinal processing.

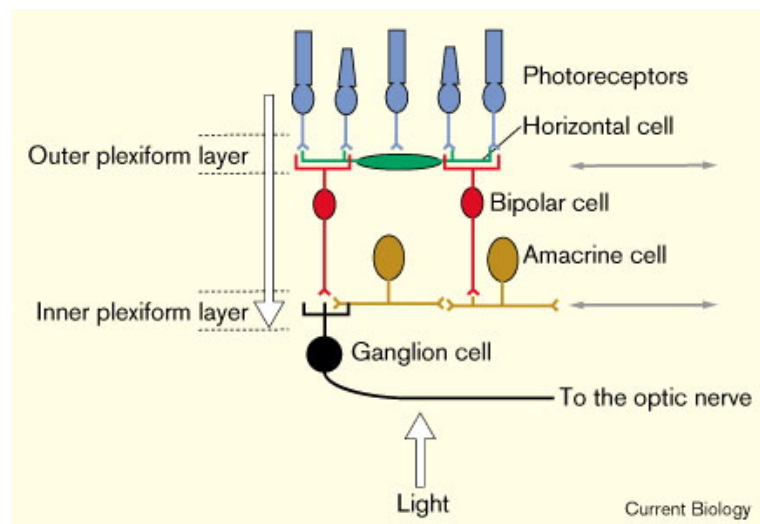


Figure 2.2: **Retinal processing** from [131]. Photoreceptors (either rods or cones) sense the light and transmit it to ganglion cells through horizontal, bipolar and amacrine cells. Ganglion cells send signals to the brain through the optic nerve.

Ganglion cells in turn transmit information to the lateral geniculate nucleus (LGN) which has long been known as a relay station of visual signals to sensory processing areas of the brain. While it is a relay station, the LGN has been observed to also perform some sort of computation to the inputs with modulation of visual responses [84].

The visual signal is finally transmitted to the visual cortex which is the central area responsible for the processing of vision. The signal first goes to the primary visual cortex (or V1) which extracts primary visual features by responding to moving edges (see next subsection for more details). Importantly, the receptive fields of V1 cells span only on a small retinal zone, which ought to preserve spatial information. The responses of V1 neurons are then transmitted to deeper areas of the visual cortex. In each subsequent area, the receptive field of the neurons are significantly increased, such that precise spatial information is lost, most likely for the benefit of invariance to the position of specific signals and to extract more complex features [36]. Crucially,

V1 projects to V2 which is also known to respond to moving edges. From V2, the responses are transmitted to two distinct pathways known as the ventral stream or “what” pathway and the dorsal stream or “where” pathway [241] (Figure 2.1). The ventral stream is thought to be responsible for object recognition and memory, whereas the dorsal stream is mainly associated with motion as it is thought to encode object locations and guide eye and arm movements. In the ventral stream, responses of V2 go to V4 whose neurons are thought to be tuned to simple shapes [188]. V4 then connects to the inferior temporal cortex (IT cortex) where neurons have been observed to be selective for complex features such as faces [56]. On the other hand, in the dorsal stream, V2 projects to V6 and then finally to V5 also known as the middle temporal area (MT). V6 neurons are thought to be responsive to self-motion stimuli of simple visual edges (e.g. long extended bars) while V5 neurons are on the other hand responsive to all sorts of motions including complex shapes and objects as well as different speeds [13], [148].

2.1.1.2 The primary visual cortex

The primary visual cortex or V1 is one of the most studied cortical areas. It is a central part of the visual system as it is the first area to receive “raw” retinal information and to significantly change the input signal for the extraction of complex signals to then project to the ventral and dorsal stream. Considering the relevance of V1 for this thesis’ work, we will here be highlighting the architecture of V1 as well as the key functions realized by this visual processing hub.

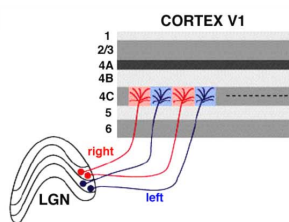


Figure 2.3: **The primary visual cortex** from [127]. The LGN sends signals to L4.

Architecture: Feedforward and feedback pathways.

First, V1 has a specific layered organization. Figure 2.3 shows this layered organization. Note that it is in fact a common structure across most cortical areas [217]. The LGN signals are transmitted first to layer 4 (L4) of the V1. Most of the cells that populate this layer are part of the so-called group of “simple cells” [167] (which can also be found in V4). Simple cells in L4 then project to L2/3, still in V1. Cells in L2/3 on the other hand mainly consist of so-called “complex cells” [166] (which can also

be found in other cortical areas e.g. V2). Note that the connections from L4 to L2/3 are excitatory as they drive L2/3 [85]. Signals of L2/3 are finally transmitted to V2. The LGN-L4-L2/3 projections are thus mostly feedforward. In addition to that feedforward architecture, V1 receives strong feedback connections from higher visual areas in L1 and L2/3. L1 is a particular layer that is mainly constituted of apical dendrites, and the actual neurons reside in L2/3. The feedback from L1-L2/3 is then transmitted through excitatory connections to both L5 and L6. L5/6 then

on their turn project to both L4 and the LGN through inhibitory connections. Thus, the projections through L1-L2/3-L5/6 would represent the feedback pathway of V1.

Simple cells. They have a very small spatial resolution.

They receive input from the LGN in the form of ON/OFF responses (respectively a depolarization of bipolar cells or an hyperpolarization of bipolar cells to signal photoreceptors' brightness increases and decreases). The responses of simple cells add a clear level of complexity to the brightness comparator performed at the retina. Indeed, simple cells have been shown to be selective to moving bars of different orientations. That is, simple cells will give a maximal response to a bar of a specific orientation that is moving in its receptive field in a direction orthogonal to the bar's orientation (see Fig. 2.5 for the orientation tuning curve of such a cell). The responses of simple cells are best represented by the so-called Gabor function [79], [116]. Figure 2.4 shows examples of Gabor functions. Importantly, neuroscientists characterized qualitatively the responses properties of simple cells to so-called counterphase sinusoidal gratings. Counterphase sinusoidal gratings (Figure 2.6) are gratings that consist in a succession of bars of different polarity/color (e.g. a black bar followed by a white bar), that can take different orientations, have different frequencies (the size and the number of bars), have different phases (the position of a bar in a receptive field), and whose bars' polarities invert in time according to a sinusoidal signal. Simple cells have been found to respond best to counterphase sinusoidal gratings of a specific orientation, a specific frequency, a specific phase and a specific polarity.

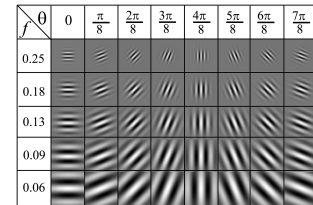


Figure 2.4: **Bank of Gabor filters** from [224].

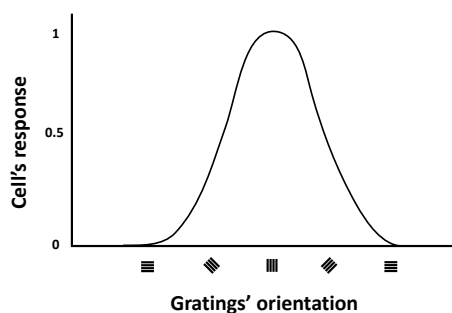


Figure 2.5: **Illustration of the orientation tuning curve of a simple or a complex cell tuned for a vertical grating.**

Complex cells. They receive inputs from simple cells, and their receptive field cover a larger zone than them. Complex cells add another degree of complexity to visual processing by integrating spatial and polarity invariance. As simple cells, complex cells are selective for bars of specific orientation and frequency in their receptive fields, however, they are not selective for the phase – that is, the position – of the bars, nor are they selective for the polarities of the bars. The invariance exhibited by complex cells basically allows a fast detection of moving edges in a large zone, which could be required for position-invariant object recognition.

Receptive field effects in V1 cells. Simple and complex cells have been shown to exhibit so-called nonclassical response properties where the way they respond is modulated according to which stimuli are present outside of their classical receptive fields. Such response properties are mainly thought to be arising from top-down feedback [120], [195]. Some of those effects observed in simple and complex cells are: surround suppression [115] and orientation-tuned suppression [186]. Figure 2.7 illustrates the principle of suppression.

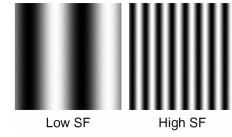


Figure 2.6: **Sinusoidal gratings** from [133]. “SF” denotes the spatial frequency of a grating.

Basically, for surround suppression, when a center V1 cell is excited with a grating, its response is maximal when the grating is localized only inside the cell’s receptive field. However, as the grating’s size increases beyond the cells classical receptive field, the center V1 cell’s response decreases. On the other hand, for orientation-tuned suppression, when a center V1 cell and neighboring cells are simultaneously excited with gratings of the cells’ preferred orientations, the response of the center V1 cell is minimal. However, if the neighboring cells are excited with a grating of orthogonal orientation to the center, the response is instead maximal. Those effects are thought to occur as an effort by V1 to remove redundant information and perform an efficient encoding [203]. In addition to these nonclassical receptive field effects, other effects include classical receptive field ones, such as cross-orientation suppression [165]. Cross-orientation suppression refers to the minimal response of a neuron when it is excited by the superposition of a grating of its preferred orientation and a grating orthogonal to this preferred orientation.

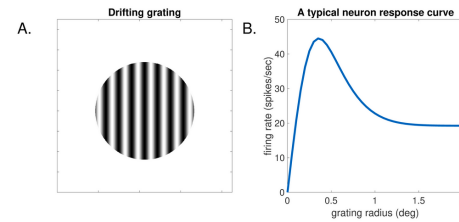


Figure 2.7: **Surround suppression** from [143]. The neuron’s classical receptive field size is about 0.4 degrees. When the grating’s radius goes over this size, the response of the neuron decreases.

2.1.1.3 Models of spiking neurons

In the brain, communication between neurons occurs not with a continuous (and fixed) signal ranged between 0 and 1, as is usually the case in artificial neural networks [136], but rather with asynchronous impulses named “spikes” [102]. The generation of such spikes depends on various chemical quantities (Ca^{2+} , Na^{2+} , Cl^{-} ionic flux) but also of the morphology and connectivity of a neuron (dendrites, astrocytes, axons, synapses) that are all interacting between each other.

Essentially, different electrical input currents are integrated by synapses and propagated through a dendritic tree to finally reach the soma of the neuron. The integration of those currents modifies the electrical state, or so-called membrane

potential, of the neuron. Different biophysical mechanisms are in place at the neuron level and aim at restoring this membrane potential to a resting value. However, if the synaptic input is sufficient enough to make the neuron's membrane potential go over some internal pre-defined threshold value, that neuron will then “fire” or “spike” and send out this spike to other neurons, through its axon. See Fig. 2.8 for a depiction of a neuron.

Different computational models have been made to model the biological neuron's behavior, with different levels of biological realism. We will here be reviewing three of such models that boasts relative popularity in the literature: the Leaky Integrate-and-Fire neuron, the Izhikevich neuron and the Hodgkin-Huxley neuron.

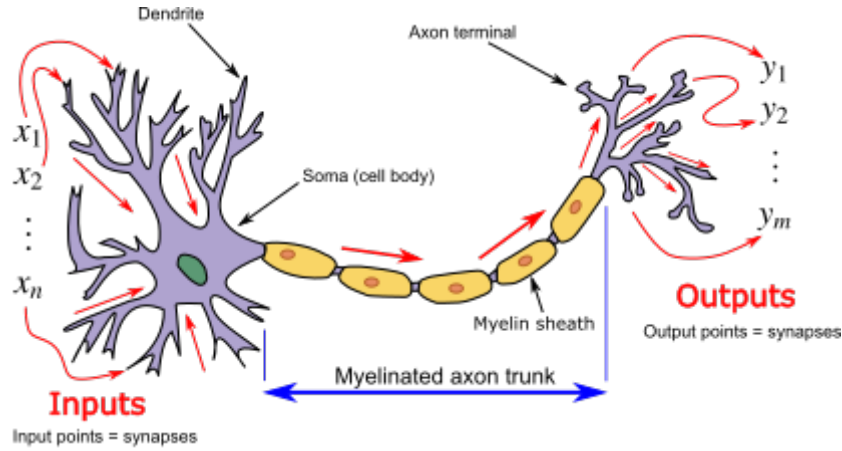


Figure 2.8: **Biological neuron.** From wikipedia.

The Leaky Integrate-and-Fire neuron. The Leaky Integrate-and-Fire or LIF neuron is one of the simplest models of a biological neuron. It finds its origins in the Perfect Integrate-and-Fire neuron first introduced by Louis Lapicque in 1907. In a LIF neuron, the voltage of the neuron increases with the current $I(t)$ received, but also decreases with time due to a leak term $\frac{V_m(t)}{R_m}$. The dynamics of a LIF neuron follow:

$$C_m \frac{dV_m(t)}{dt} = I(t) - \frac{V_m(t)}{R_m}, \quad (2.1)$$

where C_m is the membrane capacitance, V_m is the membrane voltage and R_m is the membrane resistance. In the absence of any input electrical current, due to the leak term, the membrane voltage will decay back to a resting value with time, according to the strength of the membrane resistance. If the membrane voltage goes above a (usually constant) threshold V_{th} , the neuron emits a spike (a dirac-delta function) and the membrane voltage gets forcefully reset to an arbitrary (constant) reset value. As the LIF neuron does not model more accurate neuronal parameters and dynamics, it reduces the computational costs while providing a relatively faithful model of neuronal computation. Other variations of the Integrate-and-Fire neuron model that add biological realism exist, such as the so-called Exponential-Integrate-and-Fire [71]

and Adaptive Exponential-Integrate-and-Fire neuron [39] models. In this work, we use a form of Leaky Integrate-and-Fire neuron.

The Izhikevich neuron. The Izhikevich neuron was proposed by Eugene M. Izhikevich in 2003 [111]. It is a neuron model with little computational cost and that is capable of exhibiting different spiking and bursting behavior consistent with cortical neurons. The neuron model is defined as follows:

$$\frac{dV_m}{dt} = 0.04V^2 + 5V + 140 - U_m + I_m, \quad (2.2)$$

$$\frac{dU_m}{dt} = a(bV_m - U_m), \quad (2.3)$$

where V_m is the membrane voltage of the neuron, U_m is a membrane recovery variable that represents the activation and inactivation of ionic flux in a biological neuron to get the membrane potential back to its resting value, I_m is the synaptic current, a is a hyperparameter that controls the time-scale of the recovery variable and b is another hyperparameter that controls the sensitivity of the recovery variable to the fluctuations of the membrane voltage. If V_m goes above the threshold V_{th} , the neuron emits a spike and the membrane voltage/recovery variables is reset according to:

$$if \ v \geq V_{th}, then \begin{cases} V_m \leftarrow c \\ U_m \leftarrow U_m + d, \end{cases} \quad (2.4)$$

where c is the reset value of the membrane voltage, and d is a hyperparameter that controls the reset of the membrane recovery variable. By carefully setting all of the different parameters, the Izhikevich neuron model is able to reproduce various spiking patterns such as: regular spiking, burst spiking (bursts of several spikes in quick succession), chattering spiking (spaced bursts of spikes) and so on. Note that Izhikevich also proposed the Resonate-and-Fire neuron model [110] in which subthreshold oscillations affect the way a neuron spikes, and is therefore more sensitive than the other models to the temporal structure of an input.

The Hodgkin-Huxley neuron. This neuron is one of the most realistic depiction of a biological neuron. It was proposed by Alan Hodgkin and Andrew Huxley in 1952 [102]. It accurately describes the membrane voltage V_m of a cell and the behavior of the flux of ions/synaptic currents I_{m_i} across a cell as follows:

$$C_m \frac{dV_m}{dt} = - \sum_i I_{m_i}(t, V_m), \quad (2.5)$$

$$I_{m_i}(t, V_m) = g_i(t, V_m) \times (V_m - V_{m_i}), \quad (2.6)$$

where i is the index of the ionic current, V_{m_i} is the reversal potential of the i -th ion channel, C_m is the membrane capacitance and $g_i(t, V_m)$ is the conductance of the i -th

ion channel. Thus with this model, it is possible to precisely model different ionic currents and their influence on the membrane voltage of a given neuron. Hodgkin and Huxley were awarded a Nobel prize in Physiology or Medicine in 1963 for this work.

More information on such spiking neurons and their behavior can be found in [70], [254]. Using such spiking neurons within different network architecture types (e.g. feedforward, or recurrent with lateral and feedback connections, convolutional) can bring different neuronal dynamics and input representations.

Importantly, as can be observed in the above spiking neuron models, spiking neurons evolve in **time**, contrary to artificial neural networks. As the computations performed by spiking neurons are vastly different from the ones seen in artificial neural networks, they also require other types of algorithms to learn synaptic weights.

2.1.2 Spike Timing-Dependent Plasticity

Learning with spiking neural networks can grossly be distinguished into two types of approaches: **unsupervised hebbian-based** and **supervised gradient-based**. The first approach is derived from observations on brain plasticity and comes from Donald Hebb’s famous postulates [97] of: “neurons that fire together wire together” and a cell 1 needs to “take part in firing” cell 2. The second approach on the other hand moves the backpropagation algorithm [251], widely used in deep learning [136] to the spiking domain, with so-called surrogate gradients or with the eventprop algorithm [179], [252].

One of the most well-known hebbian-based learning rules is **spike timing-dependent plasticity (STDP)** [41]. As this PhD work uses only STDP, we will here present only the STDP learning algorithm.

Brain plasticity is thought to be experience-dependent and local [97]. What has been observed in the brain is that when a pre-synaptic neuron emits a spike shortly before the post-synaptic neuron’s spike, the strength of the synapse between the two neurons will increase. Hence, according to the type of stimulus a neuron receives, its synaptic connections will grow more for the inputs that tend to make it spike. This mechanism is most commonly known as “long-term potentiation” (LTP) [30], [31] and describes the initial theory of learning proposed by Donald Hebb [97]. LTP is “long-term” in the sense that this plasticity effect can still be measured a long time (in the scale of hours) after the triggering event. However, biological neurons have not only been observed to perform this sort

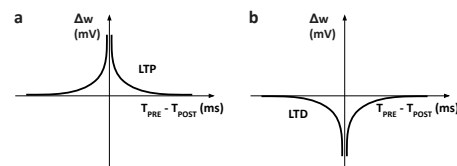


Figure 2.9: **LTP and LTD mechanisms.** a, LTP. b, LTD.

of LTP synaptic updates, but also, the so-called “long-term depression” (LTD) [142]. LTD refers to another mechanism in which, two neurons that are connected to each other such that the pre-synaptic neuron emits a spike shortly after the post-synaptic neuron’s spike, will see the strength of their shared synapse decrease. Figure 2.9 shows the LTP and LTD mechanisms.

STDP was discovered by Markram et al. [153] toward the end of the 20th century. It describes a specific type of hebbian synaptic plasticity in which neuronal synapses undergo either LTP or LTD according to the timing of the spikes. STDP has been found in multiple regions of the brain such as the hippocampus [27] and the cortex [66]. What was found was that, synaptic strength could get increased or decreased if the timings between the pre-synaptic spike and the post-synaptic spike were in the order of magnitude of milliseconds (approx. between 20 and 50 ms [27]). LTP and LTD would be weaker the larger the timing difference was. In STDP, both LTP and LTD happens at a synapse depending on the sign of the difference between a pre-synaptic spike time and a post-synaptic spike time. The most commonly observed (and understood) version of STDP is the causal rule, in which, LTP is induced when the pre-synaptic spike happens shortly *before* the post-synaptic spike, and LTD is induced when the pre-synaptic spike happens shortly *after* the post-synaptic spike. Concretely, in such a causal STDP rule, a neuron would basically increase the efficacies of synapses that were useful in making it spike, and decreasing the efficacies of synapses whose inputs arrived too late.

The expression of such a (causal) STDP rule generally takes the following form:

$$\Delta W = \begin{cases} \eta_{LTP} e^{-\frac{\Delta t}{\tau_{LTP}}} & , if(\Delta t > 0) \\ -\eta_{LTD} e^{\frac{\Delta t}{\tau_{LTD}}} & , if(\Delta t < 0) \end{cases} , \quad (2.7)$$

where ΔW denotes the change of the synaptic weight/strength, η_{LTP} and η_{LTD} are learning rates, Δt is $(t_{post} - t_{pre})$, the difference between the timing of the post-synaptic spike and the timing of the pre-synaptic spike, τ_{LTP} and τ_{LTD} are time constants that control the time scale of the exponential plasticity window.

Although we do not use them, other extensions of STDP called triplet STDP learning rules [86], [190] also exist. In such triplet STDP, rather than considering only pairs of spikes, a triplet of them is considered. This triplet can either be the two last pre-synaptic spikes and one post-synaptic spike, or one pre-synaptic spike and the two last post-synaptic spikes. Triplet STDP is able to account for more biological observations than the classical STDP protocol regarding the weight changes induced by the timing and frequency of more than one pre-synaptic/post-synaptic spike [222], [249]. In addition to the standard and triplet STDP, reward-modulated STDP rules also exist and are mainly used to perform supervised learning [168] and reinforcement learning [19].

2.1.3 Spiking models of V1

The responses of biological simple cells have long been proved to be well described by the Gabor filter model [116]. Inspired by this finding, different spiking representational frameworks modeled simple cells with Gabor filters, and complex cells with a max pooling operation [121], [155], [234], [261]. While those works studied the use of STDP to learn intermediate features (e.g. shapes), the modeling of simple and complex cells was purely mechanistic, that is, without learning. Other mechanistic spiking works also include bio-plausible modeling of the cortex [8], [29], [263]. Such mechanistic works do not answer how the cortex might learn simple and complex cells.

In terms of learning models, the emergence of simple cells following the exposition to natural images, have been explained via sparse coding models [125], [264] that used local synaptic plasticity rules, but also independent component analysis [210], that used causal STDP learning rules. While they did not use natural images but rather standard computer vision datasets, Kheradpisheh et al. [122] modeled a retina-like input with a Difference-of-Gaussians, and observed the emergence of simple cell-like receptive fields in their spiking neurons. A key characteristic of simple and complex cells are their response properties to counterphase sinusoidal gratings. While those models accounted for Gabor function-like receptive fields, they did not assess that their learnt synapses allowed neurons to give simple cell-like responses. Other models explain direction-selectivity characteristics of biological simple cells using plastic inhibition [134], [135].

Recently, Sabatini et al. [202] proposed a fully mechanistic spiking model of simple cells, that they run both on simulation and neuromorphic hardware. They characterized the effect of their parameters on orientation and frequency tuning of simple cells. Taylor et al. [235] also recently proposed a predictive processing-based learning model of both simple and complex cells. In their model, neurons predict from their activity their future input. Learning is done by minimizing a loss on both prediction and metabolic constraints, using surrogate gradients [179]. The authors observe the emergence of neurons with receptive fields similar to simple and complex cells. While their model accounts for spiking statistics and tuning distributions similar to V1, the authors did not explicitly replicate response properties (orientation, frequency, phase and polarity) of simple and complex cells.

2.1.4 Neuromorphic sensors and circuits

A new class of technologies termed “neuromorphic” has attracted the attention of many researchers in computational neuroscience, artificial intelligence and robotics (computer vision and control mainly). Neuromorphic technology proposes a biomimetic

approach in which electronic (or software) systems mimic — more closely than deep learning — the functioning of biological neurons [43], [159]. We will here be reviewing key instances of neuromorphic sensors and circuits.

Neuromorphic or event cameras. Initially called “silicon retinas” [160], event cameras are bio-inspired sensors that mimic the processing done by the retina of mammals. Similarly to the retina, cells (pixels) in event cameras asynchronously respond to brightness increases and decreases, such that when brightness increases above some internal variable threshold, the cell outputs an “ON” event (or spike), and when the brightness decreases below another threshold, the cell instead outputs an “OFF” event (or spike).



Figure 2.10: **Event camera and standard camera output.**

Events take the following form: $e_k = (x, y, t, p)$ where x and y are the pixel’s coordinates, t is the timestamp of the event and p is its polarity. Figure 2.10 showcases an event camera’s output and its analogous frame. Crucially, rather than perform-

ing computation with excitatory and inhibitory circuits as done by rod, cones, bipolar, horizontal and amacrine cells in the retina, event cameras process data in another way by using electronic amplifiers and comparators to generate events. Event cameras possess characteristics that differ from standard cameras, such as: a high temporal resolution (in the order of microseconds), a low latency (in the order of microseconds for microseconds under high lighting but milliseconds under low lighting), a high dynamic range (in the order of 120 decibels) and a low power consumption (in the order of milliwatts). Overall, the sparse data output of event cameras coupled with their fast reaction time as well as adaptability to different lighting scenarios make them interesting for robotics where sensors’ efficiency constrain processing. This all fostered the creation of a new field of computer vision named “neuromorphic vision” [42], [80] interested in event-based vision processing. This marked a significant advancement in neuromorphic research where industrial interests in event cameras also played a key role in the development of these sensors. Companies such as Prophesee and iniVation are currently market leaders in the commercialization of event cameras and have also highly participated in the software development, which enabled an easier use of event cameras and increased interest in them. It is also important to note that in addition to vision, neuromorphic sensors for audition (neuromorphic cochlea) and olfaction (neuromorphic olfactory sensor) also exist [198] and recruit the attention of many researchers.

Neuromorphic circuits In addition to sensors, neuromorphic circuit boards that mimic neural computation have also been developed [107]. Those circuits use electronic devices to model neurons’ membrane voltage, synaptic communication as well as internal neuronal dynamics (e.g. adaptive threshold, spiking, spike frequency adaptation etc.). As the computations done by biological systems strongly vary between type of neurons and areas, different neuromorphic circuits can be built depending on the task at hand. Crucially, such circuits would be interesting from a neuroscientific perspective to test neural processing hypotheses, but they also show compelling features in robotics as they have a sub-millisecond processing speed and low energy consumption. Different industry manufacturers (e.g. Intel with Loihi, SpiNNcloud with SpiNNaker2 or SynSense with Speck and Xylo) now propose neuromorphic chips that could be used jointly with neuromorphic vision systems. An issue that however arises when using such neuromorphic circuits is the lack of software modularity for a direct and simplified implementation of novel approaches. Importantly, the design of such circuits is still undergoing much research and is also growing with the discovery of new materials and devices such as memristors/memristive systems [255] which have certain properties (e.g. resistive switching) that should allow significant progress in the development of neuromorphic circuits [161] (in-memory computing, STDP learning with memristors as synapses etc.).

As this thesis focuses on brain-inspired learning with spiking neural networks, we will not be reviewing non-spiking visual cortex-like applications of neuromorphic vision. However, details on recent models and datasets can be found in [42]. See the previous subsection for a review on spiking models of the visual cortex.

2.2 General Predictive Coding

In this section, we will present the fundamentals of general (or rate-based) predictive coding. We will give an overview of the theory by: 1) giving a historical and theoretical background under which predictive coding came to be, 2) making an analogy with another grand theory of brain function, the efficient coding hypothesis, 3) summarizing key findings of evidence of predictive coding in the brain. As predictive coding encompasses multiple paradigms, we hope for this overview to be general and explain the central elements of predictive coding.

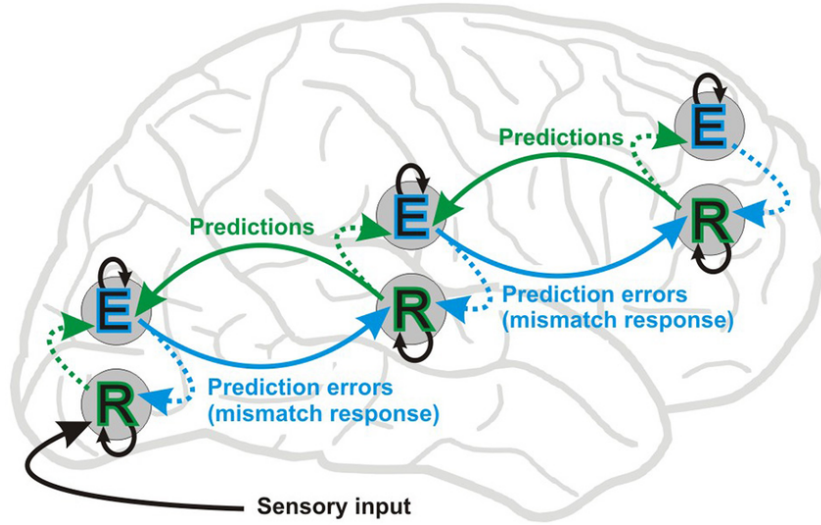


Figure 2.11: Illustration of Predictive Coding from [226].

2.2.1 Concept and historical background

2.2.1.1 Principle

Predictive coding [3], [77], [105], [164], [225] is a prominent theory of brain function that was introduced by Rao and Ballard [195]. This framework considers that the brain forms an internal model of the world. The concrete idea behind predictive coding is that, using this internal model, deeper neurons send out “top-down predictions” to earlier layers in order to cancel out information that has already been predicted. This way, only a residual “prediction error” is transmitted in the hierarchy, to update the brain’s internal model. Figure 2.11 shows the principle of predictive coding. As can be observed in the figure, neurons are separated into different classes, namely so-called “prediction” or “representation” neurons and “prediction error” neurons.

In their seminal paper, Rao and Ballard demonstrated that such an algorithm could be used for visual processing. Thanks to its generic formulation, predictive coding has also since then been applied to the processing of different sensory areas, e.g. in the auditory cortex [144] and motor cortex [196].

2.2.1.2 Background

The first ideas behind predictive coding can be traced back to Ibn al-Haytham and Descartes (see [96] for a review). These ideas have been formally formulated in Helmholtz’s theories of “unconscious inference” [246]. Helmholtz posited that visual perception was due to ingrained and unconscious judgments which guided the understanding of a visual scene.



Figure 2.12: **The dress** from wikipedia. An online viral phenomenon in which a dress is seen either as white and gold, or as blue and black by different observers.

An example of such unconscious inference would be the existence of optical illusions (see Fig. 2.12), where, a given visual stimulus is falsely interpreted as another due to perceptual biases. In Helmholtz’s theory, inference is realized by the *association* of a stimulus with *prior knowledge* to generate perception.

In the past decades, building on such Helmholtzian ideas, Friston explained predictive coding in the language of information theory and statistics [73]. As can be seen in Fig. 2.13, ideas of unconscious inference can be formulated in the form of a probabilistic Bayesian model to describe how perception occurs. Friston showed that this unconscious inference process must be realized by the minimization of the so-called “free-energy”. This postulate is named the **free-energy principle**. The free-energy (equivalent to a negative ELBO – Evidence Lower Bound –, used in machine learning and variational Bayesian inference) is essentially an upper bound on the surprise of a layer’s sensory states.

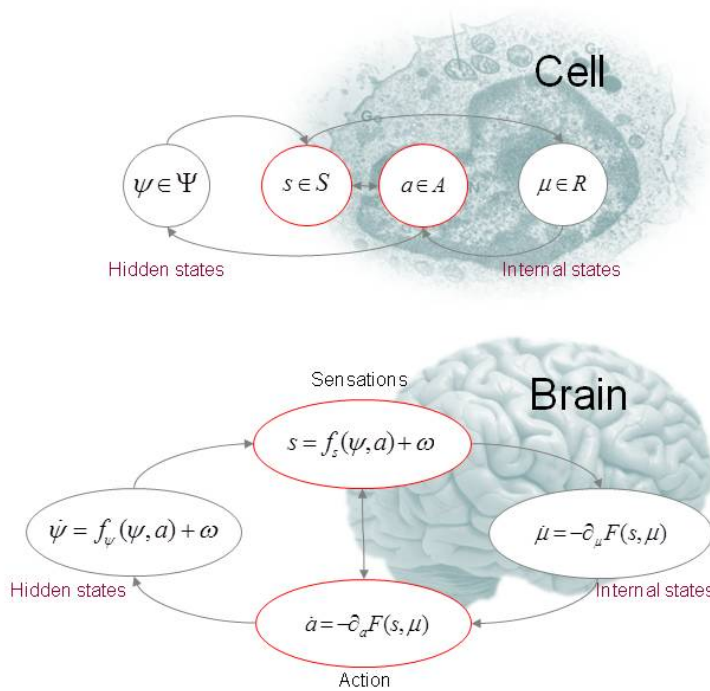


Figure 2.13: **The free-energy principle at the cell level (top) and at the level of the brain (bottom).** From wikipedia.

2.2.1.3 The Free-Energy Principle

For the sake of completeness regarding the concepts behind predictive coding, we will here give the formal definition of the free-energy principle.

Let us suppose that the brain receives a stimulus y . This stimulus is hypothesized to be induced by a hidden cause x . The goal of the brain would be to approximate the generative model $g(x; \theta)$ behind the production of the stimulus. From this generative model, the free-energy principle can be derived by performing a statistical analysis of the model and data, which will be done subsequently in the next paragraphs. As we will be using common jargon of Bayesian inference, the reader is invited to check the glossary for definitions. The generative model g is defined as follows:

$$y = g(x; \theta), \quad (2.8)$$

where g is an arbitrary function (e.g., a neural network) and θ is a vector of parameters (e.g. weights) which parameterizes g .

In general, the stimulus y is noisy, which makes the mapping between y and x difficult. For this reason, we define $p(y|x)$, which is a conditional probability distribution that characterizes the noise of the model. In addition to this characterization of the noise, the brain should have a prior knowledge on the value of the cause behind the emergence of a stimulus. We define such prior knowledge with $p(x)$ which is a probability distribution that describes the probability of the cause x to take a given value. The joint probability of y and x can then be expressed as: $p(y, x) = p(y|x)p(x)$. Note that those distributions can be initialized as different probability density functions, such as the normal distribution.

With the conditional probability distribution and the prior probability distribution, we can use Bayes' theorem to find the posterior probability distribution $p(x|y)$. It is defined as follows:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}, \quad (2.9)$$

where $p(y|x)$ and $p(x)$ still denote the conditional probability distribution and the prior probability distribution, while $p(y)$ is known as the marginal distribution, which essentially gives the probability of observing the input stimulus.

The expression of the marginal distribution is:

$$p(y) = \int p(y|x)p(x)dx, \quad (2.10)$$

where $p(y|x)$ and $p(x)$ still denote the conditional and prior probability distributions.

In general, the calculation of the marginal distribution is intractable as it requires integrating over the entire space of causes x . To solve this problem, usually an approximate posterior distribution $q(x|y)$ is defined. Note that similarly to the

conditional and prior probability distributions, the approximate posterior distribution can be initialized as any sort of probability density functions.

Friston [73] defines the free-energy as follows:

$$\begin{aligned}
 F &= - \int q(x|y) \ln \frac{p(y, x)}{q(x|y)} dx \\
 &= \underbrace{-\ln p(y)}_{\text{Surprise}} + \underbrace{KL[q(x|y), p(x|y)]}_{\text{Divergence}},
 \end{aligned} \tag{2.11}$$

where $q(x|y)$ and $p(y, x)$ still denote the approximate posterior distribution and the joint probability distribution, $p(y)$ still denotes the marginal distribution, $p(x|y)$ still denotes the (true) posterior probability distribution, and $KL[a, b]$ is the Kullback-Leibler divergence between a distribution a and a distribution b .

Essentially, the second line of Eq. (2.11) shows us that the free-energy can be divided into two terms. The first term is the negative logarithm of the marginal distribution. The second term is the Kullback-Leibler divergence between the approximate posterior and the true posterior probability distributions, which measures the statistical distance between the approximate and true posterior distributions.

As the Kullback-Leibler divergence is always positive, from Eq. (2.11), we make the following observation on the free-energy:

$$F \geq \underbrace{-\ln p(y)}_{\text{Surprise}}, \tag{2.12}$$

where $p(y)$ is always comprised between 0 and 1 as it is a probability, and the negative logarithm is therefore positive and bounds the function.

Interestingly, as the marginal distribution (see Eq. (2.10)) represents the probability of observing the input stimulus, its negative logarithm quantifies the amount of “surprise” perceived by the model.

Hence, the free-energy is an upper-bound on both, the sensory surprise, and the divergence between the true posterior and the approximate posterior probability distributions. The minimization of the free-energy thus reduces sensory surprise and improves the approximate posterior; this observation is known as the free-energy principle.

The free-energy principle has been shown to account for a wide variety of brain-like mechanisms and phenomena such as the brain’s hierarchical organization, hebbian plasticity, receptive fields’ properties and late evoked responses [74]. It has also been extended for active perception by modeling both action and perception learning [78]. It is also worth noting that in addition to those contributions, Friston also showed that different theories of brain functions (e.g. the Bayesian brain hypothesis, the

efficient coding hypothesis, reinforcement learning and motor control etc.), and the free-energy principle are all related [76].

This all positioned the free-energy principle as a possible unifying brain theory, and a clear hallmark in the theoretical understanding of predictive coding.

For a detailed walkthrough on how a predictive coding network can be derived from the free-energy principle, we refer the reader to [34].

2.2.2 Analogy with efficient coding

The efficient coding hypothesis is another theory of perception, which is centered around how the brain performs sensory coding. It was introduced in 1961 by Barlow [20] as an effort to guide research in sensory neuroscience. As the brain exhibits a complex anatomical structure which is home to a wide variety of complex phenomena, without a clear idea on *how* perception is generated, its study would be vastly inefficient and complicated. For instance, an example to justify this approach was the one of “wings”. Barlow argued that if one did not know that birds could fly, one could end up characterizing wings with extreme precision and miss its most important characteristic which is that they are used by animals to fly. For this reason, Barlow thought that it was vital to have *concepts* behind the generation of perception, so that experimentalists could aim their efforts at investigating specific structures, rather than an everlasting number of them. The efficient coding hypothesis has no defined generalized form. Barlow based it on 3 major ideas: 1) the brain has filters to detect meaningful information, 2) those filters are adaptive, and 3) the brain reduces information redundancy. The first 2 ideas presuppose that the brains responds to specific inputs and that those responses vary according, either to the context, or the type of input received. The last idea on the other hand presupposes that this is all done while ensuring that redundant information is removed – that is, the sensory signal is compressed. This is based on information theory [215] where “information”, “channel capacity” and “redundancy” are formally defined.

Succinctly, efficient coding views the brain from an information-theory perspective and postulates that it should be aiming at minimizing redundancy (as defined mathematically) between neurons’ responses, by filtering an input according to some sort of internal motive (e.g. visual/audio recognition, action etc.).

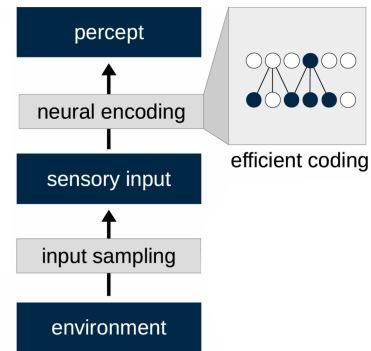


Figure 2.14: **Principle of efficient coding** from [63].

As the filters prescribed by efficient coding must reduce redundancy and change according to the context, the efficient coding hypothesis predicts that the filters should in theory be adapted to the sensory area they are located in and responses should be overall reduced and nonlinear. Examples of such key predictions of efficient coding in the brain can be found in the visual and auditory cortices, where neurons are respectively optimized for coding images and sounds, but also exhibit a significantly decreased response, compared to the input stimulus. Neurons' responses of those cortices also depend on the sensory input's context, as can be seen by the emergence of different nonlinear effects such as surround and repetition suppression but also mismatch negativity. Thus efficient coding provides an elegant way of understanding sensory processing in the cortex.

While the efficient coding hypothesis does not ascribe any particular algorithm to its realization, a popular model of efficient coding is the Independent Component Analysis (ICA) algorithm [44], [106], [117]. ICA basically assumes that a given signal can be linearly separated into a set of independent signals. Independent signals denote signals that do not share mutual information (so that redundancy is reduced). In the context of neurons, the ICA algorithm basically posits that a given sensory input would be separated into different independent neural responses. There have been different approaches for performing ICA, and one of the proposed formulations is the one of Bell and Sejnowski [24], [26] which is derived from the infomax principle [145]. The infomax principle aims at maximizing mutual information between an input signal and its outputs. Using this principle, the ICA algorithm of Bell and Sejnowski works by maximizing the joint entropy of the output signals with gradient ascent. Under different constraints (relationship between the nonlinear function applied to the outputs, and the probability density function of the independent responses), such an algorithm has been shown to give independent components/neural responses [25], [176]. Interestingly, sparse coding [180] can also be encompassed by such type of algorithms, despite no enforcement of a sparsity term in the method.

Predictive coding can be seen as another way of formulating efficient coding ideas. Friston [76] argued that the infomax principle, which has been used for some forms of ICA, can be also solved within the free-energy principle. Concretely, the free-energy defined in the previous sections, can be rewritten into the difference between “complexity” and “accuracy” terms. The “complexity” term indicates the computational costs, while the “accuracy” term indicates how well the sensory data is explained. Thus, the minimization of the free-energy should give: a minimization of complexity (and therefore an increase in efficiency), and a maximization of accuracy (and therefore a maximization of the mutual information between input and output), which should be analogous to the infomax principle. Other definitions of predictive coding formally place it under the Minimum Description Length framework [105], which aims at finding efficient codes and can be argued to be consistent with, if not tightly related to Barlow's initial postulate.

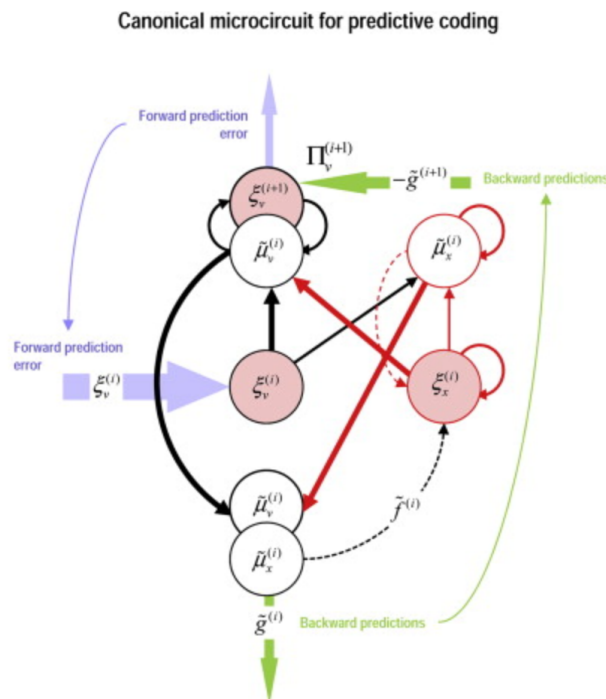


Figure 2.15: **A canonical microcircuit for predictive coding.** Circles are associated with different neural populations, and symbols represent predictive coding quantities. Red denotes inhibitory connections and black denotes excitatory connections. Details on the quantities can be found in [23].

Overall, the initial ideas suggested by Barlow are generic enough to be implemented by different types of algorithms. Essentially, the predictive coding theory proposes such efficient brain-like responses, and has therefore widely been understood as a computational framework that falls under the efficient coding hypothesis.

2.2.3 Predictive coding and the brain

The brain is a fascinating structure that is home to hierarchical and recurrent neural connections which are thought to create complex neural dynamics responsible for the emergence of multiple perceptual effects. As mentioned in Secs. 2.2.1.1 and 2.2.1.3, the predictive coding theory under the free-energy principle has successfully replicated various brain phenomena for both classical and nonclassical receptive fields in different sensory areas. However, those results do not clearly establish how predictive coding relates to the brain’s underlying neural circuitry. As a theory of brain function, it is of great interest to place predictive coding under concrete brain-like constraints to see how much this theory and the brain actually correlate. For this purpose, there have been various efforts at mapping explicitly predictive coding-like architectures to detailed brain circuits.

The notion of a “canonical microcircuit” has been around for over 30 years now. It was introduced by Douglas and Martin ([60], [61]) in 1989 and 1991. They performed

an in-vivo study of cat's visual cortex by recording (intracellularly) neurons' responses to electric stimulation, and characterized the excitatory and inhibitory dynamics of those biological neurons to propose a model that could replicate such dynamics. In their model, they separated neurons according to their biological circuit, and used simple computational models to represent them. This defines a microcircuit of the visual cortex as it explicitly models detailed population of neurons according to known excitatory and inhibitory synaptic connections between each population. The authors hypothesized that this circuit could be "canonical", that is, recurring across diverse areas of the cortex.

Such ideas have been explored by considering the predictive coding (under the free-energy principle) framework. The most recent propositions and discussions on such predictive coding canonical microcircuit have been proposed by Bastos et al. [23] and Keller and Mrsic-Flogel [120].

Bastos et al. first reviewed across different articles findings on synaptic connections in cortices' areas. Some key observations they made were for instance that:

- **Microcircuits in sensory ([60], [119], [238]) and motor ([139], [250]) cortices have a similar structure:** The number of layers and their anatomies stay the same in different areas. Cells' synaptic connections also show much similarity.
- **The cortex is hierarchically organized such that feedforward connections originate from superficial levels, and feedback connections originate from deep levels ([21], [67], [244]):** Feedforward input is transmitted in a given area through cells in L4, while the top-down/feedback input from other areas is initially received by (deep) L1 and then goes to pyramidal cells in L2/3 to finally project back to L4 through a connection in L5/6.
- **Feedforward connections mainly drive neural response [216], while feedback connections both drive ([46], [51], [163]) and modulate ([48], [216]) a neuron's response:** External sensory input is mainly conveyed by the driving excitatory feedforward connections. Feedback connections tend to modulate responses according to the sensory context, but have also been seen to exhibit some driving-like behavior.
- **Feedback connections might be mainly inhibitory [5], [94], [162], [169], [170], [229], [230], [239]:** When excited with a predictable stimulus, a neuron's response is minimal, however when this predictability no longer holds, its response increases. This suggests that feedback connections suppress predictable inputs. Note that local adaptation alone cannot explain such suppression [94]. Driving responses could be caused by disinhibition.

Considering all those constraints, Bastos et al. proposed a canonical microcircuit based on the predictive coding theory by mapping the predictive coding equations (see [34]) to specific layers, and types of neurons (e.g. excitatory and inhibitory neurons grouped as cortical columns). Figure 2.15 illustrates this canonical microcircuit.

In the case of Keller and Mrsic-Flogel [120], while they did not propose a microcircuit, they also discuss how predictive coding and cortical computation are related. Precisely, they highlighted behavioral elements that suggest the existence of some sort of predictive processing. They also considered some fundamental elements of predictive processing such as prediction error and precision signals. Suppressive effects in particular could be interpreted as some sort of “prediction error” signals. Importantly, they noted that different sorts of prediction error units would be needed in a biological system to convey prediction errors. The existence of such specific prediction error signals is still subject to much debate and under research by scientists in the field [237].

Finally, a key aspect of brain computation that has been weakly considered by most predictive coding methods is the fact that real neurons use **spikes** for computation. A new alley of research named “spiking predictive coding” aims at solving this issue.

Conclusion

In this chapter, we introduced the different background knowledge, theories and models upon which this thesis is based. Essentially, this work rests on different fields, with aspects rooted in computational neuroscience (spiking neural networks, visual cortex, STDP), others rooted in engineering (event-based vision) and lastly aspects rooted in cognitive science (efficient coding and predictive coding). In the next chapter, we will be reviewing the different models of predictive coding with spiking neural networks.

REVIEW OF MODELS OF PREDICTIVE CODING WITH SPIKES

Overview	34
3.1 Challenges	34
3.2 Historical Antecedents and Early Works	36
3.2.1 Early work by Ballard, Rao, and collaborators.	36
3.2.2 Early work by Denève.	37
3.2.3 Conclusion	37
3.3 Spiking predictive coding with explicit error neurons (Class 1)	38
3.3.1 Mismatch negativity with Spiking PC.	38
3.3.2 SNN-PC: Spiking Neural Network for Predictive Coding. .	39
3.3.3 Conclusion.	40
3.4 Spiking predictive coding with membrane potential error encoding (Class 2): Spike Coding Networks	40
3.5 Spiking predictive coding with implicit error encoding (Class 3)	42
3.5.1 Efficient Coding through Suppression.	42
3.5.2 Prediction Errors through Inhibitory Plasticity.	43
3.5.3 Predictions and Prediction Errors through Learning Rule and Correction Neurons.	44
3.5.4 Conclusion.	45
3.6 Neuromorphic applications of predictive coding	46
3.6.1 Implementations of Spiking Predictive Coding on Hardware and for Robotics.	46
3.6.2 Conclusion.	48
3.7 Conclusion	48

Overview

This chapter has been published as a preprint on arXiv [171] and is undergoing review. As models of predictive coding for spiking neural networks had never been surveyed, which posed significant difficulties in identifying related works, we decided to review existing models.

We identify this survey as one of the main contributions of this thesis considering the growing influence of predictive coding works and spiking neural networks, while such a survey was lacking from the literature.

In this chapter, we will be reviewing succinctly elements of the so-called “spiking predictive coding” framework, which aims at moving the general/standard predictive coding theory to the domain of spikes. We will first highlight the challenges faced when using spikes with predictive coding, second, introduce the historical antecedents of predictive coding, third, showcase the 3 classes of spiking predictive coding that have been identified, and lastly review instances of neuromorphic (hardware) and robotic implementations of predictive coding.

3.1 Challenges

Though early spiking predictive coding approaches date around the same time as the proposition of mechanistic predictive coding (see next section), different challenges have impeded early breakthroughs and works with spiking neural networks and predictive coding.

Rate-based vs Spikes. A dominating vision on how neurons encode inputs is that only neurons’ spiking rate matters. As this vision was prominent (and still quite is), there have been more interest into developing the predictive coding framework on the rate-based domain.

Learning with spikes. Due to their special nature, learning with spikes – and interpreting spiking responses – had been historically challenging. However, with the development of neuromorphic sensors, more interest have been attracted and there now exist different methods to learn synaptic weights in spiking simulations.

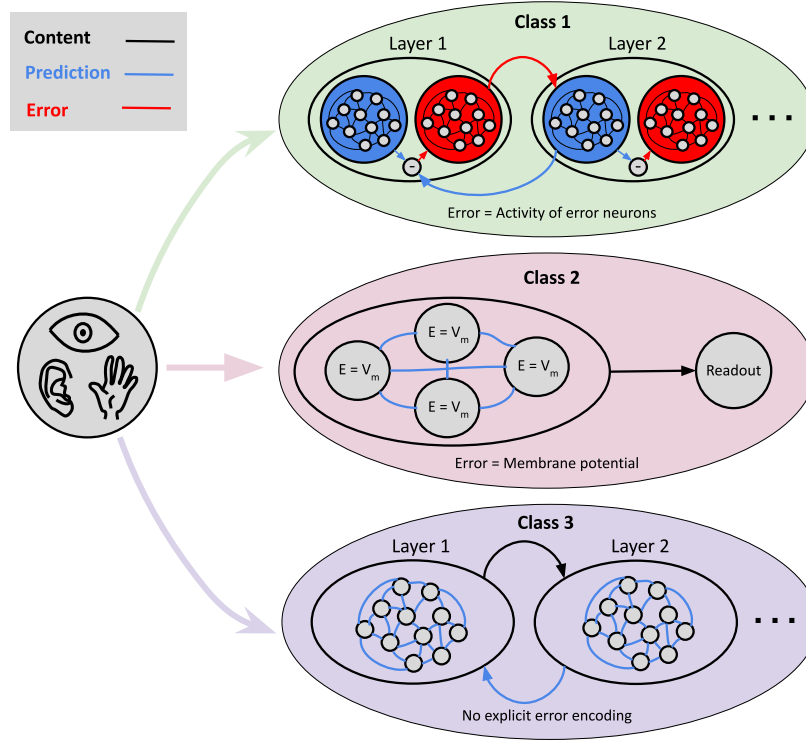


Figure 3.1: **Classes of spiking predictive coding models.** We organize the models according to how the prediction error is formulated or represented. In Class 1, there are explicit groups of neurons coding for prediction errors. In Class 2, prediction errors are represented in the membrane potentials of individual neurons. In Class 3, prediction errors are not represented explicitly, but rather treated as error-like responses that are modelled in, for instance, competition between inhibitory and excitatory synapses.

Spiking input patterns. Computer simulations necessitated to use inputs either as continuous signals, or to convert them into spikes. The now availability of different neuromorphic sensors allow for a direct use of spike input which makes more interesting the use of spiking processing approaches.

Maturity of the framework. Despite its initial promising results, the lack of clear explanation as to “why” and “how” predictive coding models can replicate brain phenomena, made them complex to evaluate and use. It is thanks to Friston’s mathematical characterization of predictive coding [76] with the free energy principle to encompass multiple theories of brain computation, that the framework got pushed toward a new state of maturity in which the behavior of predictive coding models can be explained theoretically. This type of theoretical work leads to new stage of questions such as how to map predictive coding models to spikes.

In this survey of instances of predictive coding with spikes, we identify three classes of work that we separated according to the way they model errors (Figure 3.1). Class 1 reviews methods encoding prediction errors with explicit error neurons while class 2 reviews those encoding prediction errors in the membrane potential of the neurons. The last class encode implicitly the prediction error in the spiking activity.

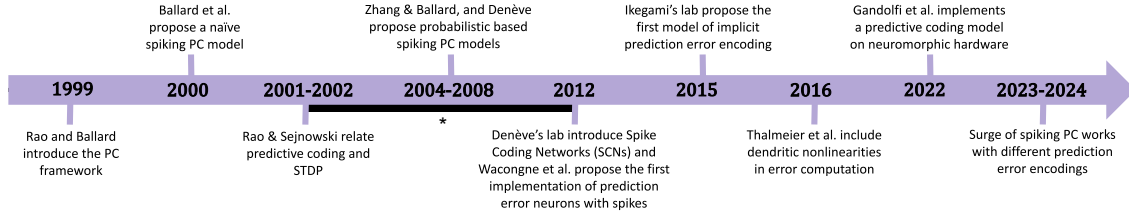


Figure 3.2: **History of spiking predictive coding models.** Timeline of key research efforts related to the development of spiking predictive coding. The black bar extending from 2001 to 2012 marked with an asterisk (*) indicates early works on sparse codes, ICA and PCA with spiking neurons by various groups.

As they are not key to the understanding of this thesis, we only here review some examples for Class 1 and 2 and refer the reader to read our survey paper [171] for more details on those models.

3.2 Historical Antecedents and Early Works

There have been different attempts at moving classical predictive coding approaches to the spiking domain. These start with the earliest works of Rao, Ballard and collaborators, and continue on to Denève's formulation of Bayesian inference on spiking neurons as a form of predictive coding. A common factor behind some of those historical approaches is an attempt to explicitly define how spikes encode information, e.g. as probabilities. Early spiking predictive coding models are based on the idea that a neuron spikes to improve predictions of its future inputs.

3.2.1 Early work by Ballard, Rao, and collaborators.

While the predictive coding framework was formally introduced as a rate-coded method by Rao and Ballard in their seminal paper in 1999 [195], research has provided growing evidence of the existence of precise spike timings in the neural code [47], [152]. Such evidence motivated early work on spiking predictive coding models. Ballard et al. proposed in [14] one of the first models in which an input is encoded as synchronous codes that are repeated in time.

Following this effort, still with the same motivations, Zhang and Ballard proposed another spiking predictive coding model [260]. They suggested a functional interpretation of spikes, by which neurons' spikes explicitly encode information as probabilities. They learnt synaptic weights by performing gradient descent in a recurrent network modeling the lateral geniculate nucleus (LGN) and the primary visual cortex (V1). Their model could account for a number of biological observations such as simple

cell-like receptive fields and neuronal activities that exhibited oscillations in the gamma frequency band.

In parallel, Rao and Sejnowski, in a series of papers, established a link between spike timing-dependent plasticity (STDP) and prediction-centered learning. The authors highlighted the property of STDP rules in learning predictions and suggested that STDP-induced inhibition and excitation could be akin to positive and negative feedback observed in standard predictive coding.

While recent models use Hebbian learning, e.g. STDP, as their primary learning rule instead of gradient descent, as in Ballard’s initial approaches, the information-theoretic view proposed in these early models, including those based on STDP-induced prediction learning, are still actively used (see models of class 1 and 3).

3.2.2 Early work by Denève.

Crucially, Denève proposed, in a series of papers [52]–[54], an extension of Bayesian inference in spiking neurons to predictive coding. Bayesian inference consists of making statistical inferences that are based on the confidence that a particular hidden cause is present or a response or event will happen according to previous beliefs about the world, i.e. priors. Denève’s model argues that a neuron should only spike if that spike conveys new information about the presence of hidden causes. It is efficient as it should encode only information that could not be explained by previous spikes.

Denève relates the spiking dynamics of the Bayesian neuron model to leaky integrate-and-fire neurons by showing that the membrane potential of those neurons can be interpreted as the difference between the log-odds ratio of a stimulus being present based on the input (synaptic evidence accumulation) and the log-odds ratio based on the neuron’s own prediction of the presence of the stimulus. This introduces the formalism of the membrane voltage as a prediction error used in some of the recent work in spiking predictive coding (see Class 2).

3.2.3 Conclusion

Different works have attempted to move predictive coding to spiking networks. However, it is important not to mistake general Bayesian models of inference using spiking implementations (e.g. [192]) for predictive coding models, as most of those general Bayesian models explicitly decode information from spiking networks without any constraints on efficiency, contrary to predictive coding models. While the research efforts described were introduced around the same time frame as classical predictive coding (see timeline in Fig. 3.2), they motivated only a few researchers

at that time to explore spiking networks. As there is now more interest in spiking neural networks and predictive coding, it is important to revisit the historical efforts described above in this section and observe their relationship with present spiking models of predictive coding. For instance, the work of Denève gave rise to Class 2, composed of so-called spike coding networks, which we review in Sec. 3.4. Similarly, the different works related to efficient sensory codes as well as the seminal works of Ballard and co-workers are precursors to the models of Class 1 and Class 3, which we review in Sec. 3.3 and 3.5. Overall, these early works exposed the difficulties of applying predictive coding to spiking networks and introduced many of the ideas that underpin more recent forms of spiking predictive coding.

3.3 Spiking predictive coding with explicit error neurons (Class 1)

The first class of spiking predictive coding models that we will consider are those in which prediction errors are coded by designated error neurons. In this category, error neurons are explicitly defined and charged with computing the mismatch between the sensory representation of an external input and predictions (produced by top-down pathways or feedback) of this sensory representation. Standard predictive coding appeared difficult to reconcile with spiking networks, since it requires error neurons that can encode both positive and negative prediction errors. How can the activity of spiking error neurons, whose firing rates can only be positive, encode both positive and negative numbers? One initial solution might be to represent the signed error as the positive or negative deviation of the firing rate with respect to a baseline firing rate value (that represents zero or the absence of prediction error). However, to solve this problem, some models of this first class define two types of error neurons for representing positive vs. negative prediction errors (inspired by on-center off-surround and off-center on-surround types of cells in the retina), while others use rate-based error units in conjunction with spiking prediction units. The presence of such units in the cortex has been reported in certain studies but is still a highly debated subject in the field [120]. We review below example predictive coding models that try to apply this particular way of modeling error neurons in spiking systems.

3.3.1 Mismatch negativity with Spiking PC.

Wacongne et al. proposed the first spiking implementation of prediction error neurons [248]. Specifically, this mechanism mimicked so-called mismatch negativity (MMN)

exhibited in a network of Izhikevich neurons. The authors considered a 3-layered network consisting of a prediction error layer, a prediction layer and a memory trace layer. In addition, a thalamic-like input was used to excite the network. In this architecture, prediction error neurons received feed-forward AMPA excitation from two populations of thalamic neurons as well as recurrent feedback GABA inhibition from prediction neurons. Prediction error neurons would then send back errors to prediction neurons which would subsequently update memory neurons. The memory neurons then accumulated information and sent this back via a recurrent excitatory pathway to the prediction neurons. In addition to those AMPA/GABA currents, the authors also consider an NMDA receptor-dependent current that is added when receiving excitation. While most of the model's connection weights are random, the synaptic connections between memory neurons and prediction neurons were adjusted with STDP. Importantly, this work was able to successfully replicate biological observations of MMN. Furthermore, while MMN could be also explained by synaptic habituation, this effort opposed this explanation of MMN and instead appealed to a fully predictive coding perspective. An MEG experiment was performed to show that a predictive model accounted for biological observations of MMN which could not be explained by synaptic habituation alone. Wacongne later also showed that their model could account for changes in MMN observed in people affected by schizophrenia [247].

3.3.2 SNN-PC: Spiking Neural Network for Predictive Coding.

The spiking neural network for predictive coding (SNN-PC) model was proposed by Lee et al. [138] as an advancement in biological realism. Through the use of a neuronal architecture built with adaptive exponential integrate and fire (AdEx) neurons, an NMDA-mediated component for synaptic transmission, and error units, the authors built a fully spiking predictive coding system.

One of the authors' main contributions to improving biological realism comes from their reformulation of the traditional error unit. Specifically they separated every error unit into two subcomponents; one coding for positive error values and another for negative values. The two components work off of the same signals, bottom-up excitatory inputs and top-down inhibitory inputs, but the formulated combination of these inputs is inverted.

The other contribution of Lee et al. is their use of feedforward gist pathways in their model. It is believed that visual cortical processing takes place within two processes: a fast approximation of the world/niche and a slower, more detailed process. The second of these is tackled by the predictive coding model and the error nodes described previously. The first, higher level representation is crafted

by the gist pathway. In the authors' work the input image is transformed through sparse random connections, the gist unit, and then passed through to each of the representation units to help bias the representations towards a specific image.

3.3.3 Conclusion.

'Class 1' predictive coding networks represent some of the first efforts to cast prediction error (or free energy) minimizing dynamics in terms of systems of discrete, pulse-emitting neuronal systems. Ultimately, these models rest on the assumption that mismatch/error signals come from explicit error neurons (the sum of whose activities over time approximate a free energy function) and, as a result, build their message-passing architectures around this building block. The primary difference in these models is the tactic that they take in modeling these error neurons (for instance, some model these changes being produced by different concentrations of neurotransmitters through variable traces while others design units that encode differently the signs of the various error signals) and the biophysical mechanisms that they instantiate for synaptic plasticity as well as the transmission of feedback and expectation signals across the full neuronal structure.

3.4 Spiking predictive coding with membrane potential error encoding (Class 2): Spike Coding Networks

In the second class of spiking predictive coding models that we review here, prediction errors are not encoded by explicitly designated error neurons but instead distributed throughout the network in the membrane potentials of the neurons themselves. In other words, the membrane potentials of the neurons represent the prediction error between the input received by a network and its decoded activity which forms the estimate of (a function of) the input. In this sense, these particular kinds of PC networks unite the encoding and decoding perspectives on neural coding – they show that efficient encoding cannot be formulated independently from the decoding algorithm itself and that what an efficient code is depends inherently on the decoder used and the task under consideration. Here, we will show the original formulation of spike coding networks (SCNs). Details on their extensions and derivations, as well as other formalisms that calculate prediction errors within the membrane potentials of neurons can be found in the full survey [\[171\]](#).

The derivation of the SCN takes a different perspective than how decoding questions are traditionally formulated. Instead of asking the question ‘What is the optimal decoder for this neural activity?’, the derivation starts with the question ‘Given input signal $s(t)$, a decoder X , an error function $E(s, \hat{s})$, and a cost function C that limits the neural activity, what should my network structure look like for the network to optimally (under E and C) generate activity that can be decoded to an estimate $\hat{s}(t)$ with decoder X ?’. In their original formulation, [32], [33], Boerlin, Machens and Denève used this approach, with a linear decoder with exponential decay and an $L2$ loss function, to derive a network where recurrent synaptic connections of neurons that have similar decoding weights are inhibitory, whereas recurrent connections of neurons that have opposite sign decoding weight values excite one another. Each neuron has a local estimate of the network’s current estimate $\hat{s}$ through the recurrent input from the other neurons and compares the future estimate given that it would generate a spike or not. Only if the spike would contribute to an improved estimate $\hat{s}$ does the neuron fire a spike. So similarly to the Bayesian neuron (Sec. 3.2.2), each neuron compares continually the input to its own output and effectively only generates a spike if that contributes to an estimate of the input. So in this case too, as with the Bayesian neuron, the parameter that is being tracked continuously and compared to a threshold, i.e. the ‘membrane potential’, is a prediction error itself. This process can also be interpreted as the network tracking the input within a ‘bounding box’ with a precision that is given by the threshold and the number of neurons [40]. Note that the computations that the network can perform are not limited to autoencoders (i.e. $s \approx \hat{s}$) but also include polynomial dynamical systems [178] as well as optimal control [223].

Surprisingly, the derived networks described above exhibit properties that were difficult to explain using more traditional decoders. One of these properties is that the networks show irregular activity with a high trial-to-trial variability. Importantly, this irregular activity should not be confused with noise. Rather, it is a result of degenerate coding: since there are multiple ‘solutions’ to the problem of tracking the stimulus, the network will use different solutions given different initial conditions. This over-representation makes the networks also strongly robust to noise, perturbations, and cell loss [22]. The error-correcting properties of the network also reflect a form of the neurobiological balance of excitation and inhibition [55], [213]. Finally, even though the ideal connectivity of the network can be mathematically derived using the method described above, the network connectivity can also be usefully learned via local unsupervised learning rules [4], [37], [38].

In their original formulation, SCNs represented a class of networks that can explain observed biophysical properties of biological neuronal networks, such as a balance of excitation and inhibition, large trial-to-trial variability, irregular firing, and robustness against noise and cell loss, in a functional context. This functional context is that of predictive coding, in which each of the neurons tracks the error between (a

function of) the input and the network estimate in the membrane potentials of the neurons. These codes are very efficient (in terms of sparsity) since spikes are only fired if they contribute to the network estimate.

Overall, the derivation of ‘class 2’ predictive coding networks all follow a similar path: a goal computation, an error function, a cost function and decoder are defined, and the neuron and network properties are derived or learned as a result of this. In the resultant network structures, neurons keep track of the prediction error in their membrane potential and only spike if this reduces the error, making these networks inherently very efficient. The connectivity and properties of the network depend on the choice of computation, error, cost and decoder, but the networks share a tight balance between excitation and inhibition, irregular firing, a large trial to trial variability and robustness against noise and neuron loss.

3.5 Spiking predictive coding with implicit error encoding (Class 3)

In the predictive coding models reviewed so far, errors were always explicitly encoded in some fashion. Class 1 used specific error units whereas class 2 derived spiking networks that encoded errors within neuronal membrane potentials. While these models for error encoding are attractive, they are not the only ones proposed in the literature. Due to the link between predictive coding and the general efficient coding hypothesis, we see the influence of previous spiking models of efficient sensory coding (see Sec. 3.2) on recent efforts that fall under a third class of models that propose to implement spiking predictive coding with different error encodings. Most of these models exhibit prediction error-like responses with no explicit definition of error encoding. Instead, they implicitly encode errors; for instance, this is done via competition between plastic excitatory and inhibitory synapses. In addition to this implicit encoding via excitation/inhibition, alternative methods have been proposed including the implementation of correction neurons or the design of a predictive learning rule. Methods of this third class are amongst the most recent predictive coding approaches (see timeline in Fig. 3.2).

3.5.1 Efficient Coding through Suppression.

According to the efficient coding hypothesis, an efficient code would represent a stimulus with as few spikes as possible. In the context of predictive coding, this would be achieved with a direct suppression of redundant codes through some sort of prediction neurons. Sinapayen et al. proposed “learning by stimulus avoidance”

(LSA) [219], [220], an STDP-based spiking neural network learning framework that exhibited predictive coding-like effects such as suppression. LSA is based on the idea that neurons learn to respond to external stimulation by having neural responses that minimize the chance of receiving more excitation. This learning framework was derived from the experiments of Shahaf and Maron on cortical neurons of rats [154], [214]. In those experiments, neurons were excited until the rats performed a target behavior. As soon as the target behavior was achieved, stimulation was removed. Hence, neurons learnt to optimally respond to a stimulus by aiming to remove it. During the training of actions, stimuli were therefore removed only when the expected behavior was achieved by the overall network (of Izhikevich neurons). At the neuron scale, STDP naturally leads to stimulus removal as inputs that lead to a desired action have their synaptic strength increased. This work demonstrated that this effect can also appear at the scale of a network and that populations can essentially learn selective behaviors. While LSA was introduced as a control approach without an explicit predictive aspect, Masumori et al. showed that LSA, through the inclusion of inhibitory STDP, could be used to perform predictive coding [157]. They connected a pool of input excitatory neurons to inhibitory neurons. After presentations of a pair of signal/target stimuli, inhibitory neurons learned to predict (and therefore suppress) target signals. The authors also observed that, if synaptic delays were utilized, their network could suppress signals relatively far in time.

Importantly, our work, Predictive Coding Light (PCL), that we detail in the next chapter is part of this category.

3.5.2 Prediction Errors through Inhibitory Plasticity.

It has been observed experimentally that neuronal responses change as a function of the frequency at which a stimulus is presented. Neurons repeatedly excited with a certain stimulus exhibit a decreased response (“repetition suppression”) compared to those excited with a novel or deviant stimulus. This change in response signals a decreasing prediction error in the predictive coding framework. While this is generally explained with an increase of the response of error neurons, Schulz et al. challenged this idea by proposing inhibitory plasticity as an explanation of such cortical prediction error responses [212]. The authors specifically modeled a biologically plausible spiking network model of the mammalian cortex in which neurons were recurrently connected via both excitatory and inhibitory pathways. The different synaptic connections in this model were plastic and learnt via a triplet spike timing-dependent plasticity rule (triplet STDP) [86]. Schulz et al. notably showed, in a series of experiments, that not only the learnt inhibition generated prediction errors/novelty responses but also explained stimulus-specific adaptation (SSA).

Zhu and Rosenbaum [262] also analyzed the emergence of such predictive coding-like effects with homeostatic plasticity. Similar studies were also conducted using rate-coded neurons [100], [101]. While the authors considered populations of neurons that received fixed bottom-up input and top-down feedback, the computation of prediction errors did not arise from a direct mismatch of the signals but rather from the learning of recurrent inhibition via STDP. In their experiments, the authors showed that, when trained with pairs of bottom-up input and top-down feedback, inhibition reduced activity to target firing rates. However, they further observed that when bottom-up and top-down signals no longer matched, inhibition failed to stabilize network activity, which led to the emergence of prediction errors. In addition, these researchers analyzed their model's network dynamics by deriving rate codes from their spiking model using mean-field theory.

Similarly, Asabuki et al. also studied the emergence of prediction errors but particularly in networks that received no top-down feedback [11]. While in the classical predictive coding view, prediction errors arise from the arrival of feedback from deeper brain areas, recent experimental results intriguingly indicate that top-down feedback might not be necessary for their emergence. A circuit of excitatory and inhibitory neurons recurrently connected might be enough to observe prediction error responses. The authors therefore considered a recurrent circuit of a pool of excitatory neurons that received bottom-up excitation and could recurrently excite each other. The researchers excited a pool of inhibitory neurons that were also connected with each other. At each point in time, the membrane potential of a neuron depended only on its immediate excitatory and inhibitory inputs. The authors' model notably spiked under an inhomogeneous Poisson process, where the firing rate of a neuron depended on the value of its membrane potential. They developed a custom learning rule [9] in which recurrent synaptic weights were potentiated if they could predict the activity of postsynaptic neurons. The authors showed that such a framework facilitated the learning of synaptic weight values that resulted in the emergence of a wide variety (biphasic, expectation dependent, context-dependent) of prediction error signals. The authors explained that prediction errors in their framework came from changes in the balance of excitatory-inhibitory currents under conditions different from training.

3.5.3 Predictions and Prediction Errors through Learning Rule and Correction Neurons.

Taking inspiration from general predictive processing, Saponati and Vinck proposed a predictive learning rule. The network in this work adapted its synaptic weight values by anticipating future inputs by firing ahead of time [209]. The objective of any single neuron in this model was to predict its next inputs from its membrane potential and its feedforward synaptic weights such that it minimized prediction errors directly

encoded in the loss function of the network. The derived learning rule potentiated connection strengths from inputs that were found to be useful in making predictions of future inputs, whereas inputs that did not enable the prediction of future inputs had their connections depressed. The authors demonstrated that, in a recurrently-connected spiking neural network of LIF-like neurons, such a predictive learning rule permitted the forecasting/prediction of future input sequences while also accounting for the emergence of spike timing-dependent plasticity rule patterns. While there exist other spiking methods that also implement predictive-like constraints in their learning rules, we do not review them here as part of class 3 [10], [88], [92], [235]. For instance, Taylor et al.’s [235] work uses an external decoder to generate predictions during sensory learning, and therefore predictions do not have a functional use in anticipating and suppressing predictable data, unlike in standard predictive coding models. Asabuki and Fukai [10] on the other hand proposed a model in which neurons predict each other’s firing rates to generate spontaneous activity in the absence of stimuli, in lieu of removing predictable data. Other spiking models [88], [92] are mostly based on contrastive learning ideas, to learn to disentangle neural representations by having neurons encoding similar representations generate spikes with a similar rate, instead of suppressing predictable inputs.

In terms of alternative predictive coding approaches, Van Driel et al. proposed an explanation of prediction mismatch responses/prediction errors with “correction neurons” [62]. They considered a pool of recurrently-connected spiking LIF-like neurons that received feedforward excitation as well excitatory and inhibitory feedback (predictions). All connections in this model were fixed. Predictions or feedback lead to early (anticipatory) firing of neurons and, when these predictions and the input did not match, a class of neurons called ‘correction neurons’ were activated. Those neurons bore similarity to error neurons but, instead of signaling an error, they fired to signal that anticipated spikes were “wrong”. Those spikes have thus a functional use as a correction of the neural code instead of being a coding error. Van Driel et al. showed that such a mechanistic network could account for prediction errors in a minimal network; this work desirably reproduced the prediction errors/prediction mismatch responses’ experiment on a mouse’s visual cortex in Fiser et al. [68].

3.5.4 Conclusion.

The methods proposed in this third class of models questioned the need for explicit error encoding in general. For instance, without any explicit definition of an error, STDP-learned inhibition alone is able to learn efficient sensory codes in which redundant stimuli are suppressed and only unpredictable stimuli generate spikes. Similarly, when looking at the response of neurons to frequent and deviant stimuli, methods of class 3 showed prediction mismatch responses – that we defined in this work as prediction errors – that arose from the competition of, again, STDP-learned excitatory

and inhibitory synapses. The different uses of STDP for predictions are further reminiscent of Rao and Sejnowski’s earlier works on predictive learning [193], [194], [197] (see Sec. 3.2). We also reviewed under class 3 various novel approaches that learnt predictions through a predictive rule, which further replicated observations of prediction errors through ‘correction neurons’ instead of classical ‘error neurons’. In general, the results achieved by methods in this third class suggest that an explicit error encoding might not be a vital requirement for predictive coding with spikes.

3.6 Neuromorphic applications of predictive coding

When performing predictive coding, predictions and error computations are performed locally by neurons that explicitly connect to one another. This has even been argued to offer an alternative to backpropagation, the workhorse of deep learning [6], [164], [185], [206], [257]. The local computations performed by predictive coding makes it not only a biologically plausible alternative to backpropagation but also a fitting candidate for applications on neuromorphic hardware. Despite these attributes, however, predictive coding has been only scarcely implemented on neuromorphic platforms. It is well known that standard deep learning approaches suffer from the weight transport problem as well as expensive computational costs for training (such as the forward and backward locking problems [185], [206]), which render their implementation on neuromorphic hardware complicated. Very recent efforts to instantiate predictive coding neuromorphically have also begun to call into question if predictive coding itself also still suffers constraints similar to those that hinder backprop-based deep networks [69], [91]; these efforts generally offer workaround solutions to these potential constraints, using schemes as feedback alignment and neuronal reservoirs to implement a more hardware-friendly version of predictive coding. As those efforts are not crucial to this thesis, we instead review here only applications of neuromorphic predictive coding with a mapping of predictive coding algorithms based on Friston’s work to spiking neurons, including some implementations on hardware and for robotics.

3.6.1 Implementations of Spiking Predictive Coding on Hardware and for Robotics.

In Friston’s formulation of predictive coding, a network operating under the free energy principle minimizes the so-called variational free energy which calculates the difference between the priors (an internal model of the causes behind an observation) and the posterior (how the observation actually evolves as inferred by data/evidence).

While this Fristonian view of predictive coding has been generally applied to rate coding modeling approaches, Palacios et al. extended it to networks of spiking neurons [187], similar to the work reviewed under ‘Class 1’ models. In their framework, each neuron tries to predict when other neurons will spike by minimizing the variational free energy via so-called ‘variational message passing’. In this approach, a network knows the exact generative model from which an observation is generated, as well as the inverse (inference/recognition) model. The network makes use of this inverse model: each time a new observation is received, the network infers the hidden causes (spikes) behind it. It uses this information, the spiking rate of the network, and predictions of future activity to update the firing rate of all neurons. Neurons then spike only when their firing rate is above a particular threshold. Such a scheme has been shown to lead to synchronous networks that are robust to perturbations and develop sparse connectivity between neurons. Gandolfi et al. built on this work by reproducing so-called delayed eye-blink classical conditioning effects with a cerebellum-like network structure of mutually inferring free-energy-minimizing neurons [81]. Their network consisted of neurons with both excitatory and inhibitory connectivities that respect Dale’s law. Crucially, Gandolfi et al. implemented digital neurons on low-power microcontrollers based on these computational principles. This neuromorphic hardware instantiation of predictive coding strongly reduced the number of neurons required (and therefore also power consumption) compared to other methods that replicated the eye-blink classical conditioning effects. The results obtained on hardware furthermore outperformed software results. Critically, these results mark an important step towards low energy consumption and increased performance of brain-inspired efficient computing on neuromorphic hardware.

Barbier et al. [18] recently proposed a fully spiking robot controller that is based on predictive coding, reinforcement learning concepts, and more generally the so-called Active Efficient Coding framework [63], [147], [236]. In the authors’ model, a 2-layer spiking neural network first extracts a sparse representation of a visual signal from an event-based vision sensor. Neurons of this network implement predictive coding in a fashion similar to the work presented in this thesis (see next chapter), by predicting each other’s spikes through STDP-learnt lateral and top-down connections. The framework devised by Barbier et al. makes use of the learnt visual representations to excite pools of actor and critic [128] spiking neurons. Concretely, the internal predictions of the representational layers are used to generate rewards and thus adapt the behavior of the agent/robot. With such types of intrinsically motivated rewards, the agent/robot learns to generate actions that induce best predicted (and thus suppressed/sparse) sensory representations. The authors validate their framework on a robotic simulation platform, demonstrating the self-calibration of active visual tracking and stabilization of moving patterns. Importantly, this work gives an insight into how spiking predictive coding could be used to learn motor behaviors in the context of robotic applications.

3.6.2 Conclusion.

In this final section, we reviewed efforts that took inspiration from the free-energy principle to propose spiking implementations in both software and hardware, as well as alternative approaches that use predictive coding and reinforcement learning for robotic applications. The scarcity of robotics applications and hardware implementations of spiking predictive coding suggests an important open area of research for fast, efficient and biologically plausible computing applied to real-world tasks.

3.7 Conclusion

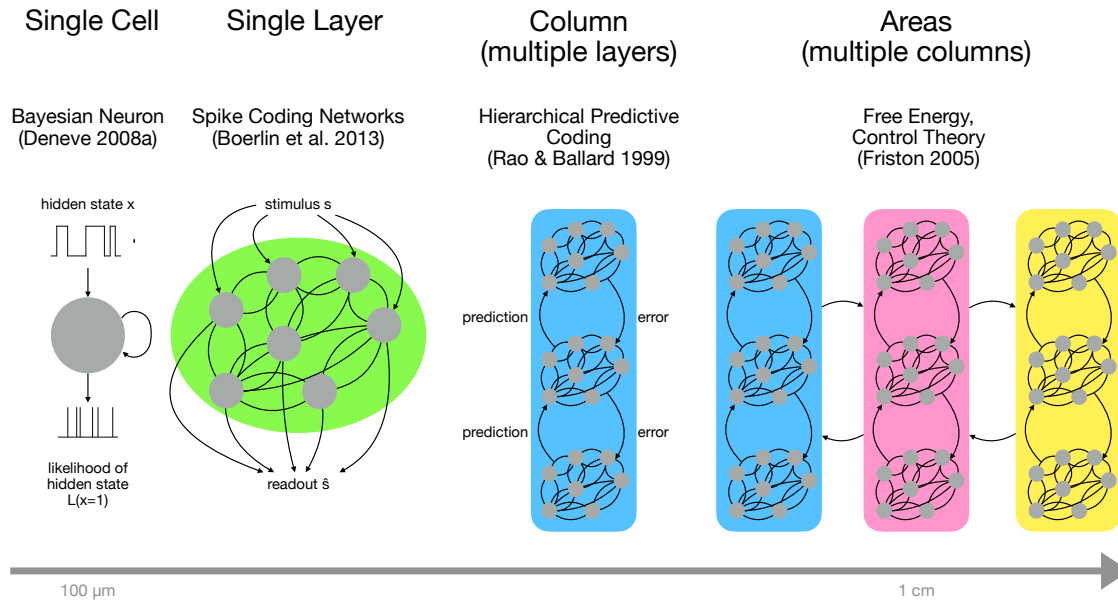


Figure 3.3: **Cognitive and spatial organization of predictive coding models.** Although we organize the predictive coding models discussed in this chapter according to how the prediction error is formulated/represented, alternative organizations are possible. For instance, the models presented here can be organized at different spatial and cognitive levels/resolutions.

In this chapter, we have surveyed approaches that have aimed to implement predictive coding (PC) in terms of spiking neural networks. There are two main reasons for the growing interest in this topic. First, predictive coding ideas feature prominently in theories of brain function and it is of great importance to understand their possible implementations at the level of spiking neurons in the brain; this is a computational neuroscience motivation. Second, there is enormous interest in constructing more energy-efficient AI systems and spiking neural networks are interesting candidates to facilitate this, as reflected in the growing interest in neuromorphic hardware; this is a neuromorphic engineering motivation. Given these two distinct

motivations, it is not surprising that the literature on the topic is somewhat scattered across different fields and venues.

We chose to organize this survey based on how prediction errors are encoded under various computational modeling approaches and schemes. Specifically, we decomposed these under three broad categories: models utilizing specific error units (Class 1), models encoding errors through a unit’s membrane potential (Class 2), and implicit error models without any direct error encoding mechanisms (Class 3). This was a pragmatic choice meant to highlight the fundamental differences between spiking PC approaches. However, other organizations are also been possible. For instance, we found a dichotomy between approaches that aim to better understand implementations of PC in the brain through computational modeling (thus reflecting computational neuroscience motivations) and approaches that focus on technological advances (thus reflecting neuromorphic engineering motivations). In the present organization of the survey, these approaches have been interspersed among the classes, which could be considered a shortcoming. In fact, instances of these two approaches are somewhat difficult to compare since they use very different “measures” of success. While computational neuroscience puts an emphasis on improving our understanding of biological phenomena (reproduction and explanation of biological data, generation of testable predictions), what matters for neuromorphic engineering is performance in applications (accuracy, speed, energy efficiency). Nevertheless, we view the organization presented in this review as an important starting point and stepping stone for fostering cross-fertilization across these two perspectives. Ultimately, computational neuroscience research could greatly benefit from neuromorphic computing hardware to speed up simulation times, which are often a limiting factor in computational neuroscience research and scientific inquiry. At the same time, neuromorphic engineering may harvest novel ideas from computational neuroscience as to what biological mechanisms may be worth implementing in the next generation of neuromorphic systems and hardware platforms.

Our division of modeling approaches into the three different classes raises the obvious question of whether one class should be considered superior to the others; what is the best modeling paradigm for spiking predictive coding? At this point, we do not think that any approach has sufficiently demonstrated its superiority to rule out the other ones, particularly, as mentioned before, approaches across these classes use different measurements of success depending on their scientific goals and the research questions that they seek to answer. Furthermore, it is important to realize that the answer may depend on which of the two motivations – neuroscience versus neuromorphic engineering – is driving a specific research effort. It may well turn out that one class of approaches turns out to be a better model of PC in the brain whereas a different class prevails in technical applications in neuromorphic systems such as neurorobots. After all, recalling David Marr’s classic analogy, we do not build airplanes with feathers and flapping wings.

Even within the field of computational neuroscience, the different implementations of spike-based PC do not all offer explanations at the same level of abstraction and at the same spatial scale (see Fig. 3.3). Whereas the Bayesian neuron and spike coding networks focus on how errors are represented in neurons and how single functions of input stimuli are represented, hierarchical predictive coding models offer explanations at higher cognitive and network levels of abstraction and bigger spatial scales, specifically shedding light on how information is shared between layers and across areas.

Finally, the research topic of spiking and neuromorphic predictive coding (and predictive processing) is still relatively young, as indicated by the overall relative dearth of explicit efforts in this direction. In this thesis, we propose a new model of spiking predictive coding that we hope will encourage future endeavors in this research topic. For this purpose, we will be studying in the following chapters the different brain-inspired frameworks developed during this PhD, which model different visual cortex-like results and show diverse interests in neuromorphic engineering with low energy consumption as well as the building of a memory to perform tasks robustly.

PREDICTIVE CODING LIGHT: A HIERARCHICAL SPIKING NEURAL NETWORK THAT LEARNS V1-LIKE RESPONSES PROPERTIES

Overview	52
4.1 Introduction	53
4.2 Methods	54
4.2.1 Network architecture	54
4.2.2 Synaptic plasticity mechanisms	56
4.2.3 Network parameters	59
4.3 How inhibitory STDP removes redundant spikes	62
4.3.1 Experimental protocol	63
4.3.2 Results	63
4.4 How PCL learns simple and complex cell-like responses	64
4.4.1 Experimental protocol	65
4.4.2 Results	67
4.5 PCL exhibits efficient information coding on digit and gesture datasets	76
4.5.1 Experimental protocol	77
4.5.2 Results	78
4.6 Discussion	85
Conclusion	86

Overview

In the two previous chapters, we gave a theoretical background on the different notions discussed in this thesis. Here, we introduce the framework we developed during this thesis: Predictive Coding Light (PCL).

The contributions of this chapter are as follows:

- We propose Predictive Coding Light, a hierarchical spiking neural network that introduces an alternative way of how predictive coding ideas could be implemented in the brain.
- We reproduce many neuroscientific findings from the visual cortex of mammals with the emergence of simple and complex cell-like receptive fields and the emergence of surround suppression, orientation-tuned suppression and cross-orientation suppression.
- We use only biologically plausible local spike timing-based learning rules and systematically compare the impact of different versions of those rules on the emergence of simple and complex cell-like receptive fields.
- We demonstrate strong performance and high energy savings on digit and gesture recognition problems.

Findings of this chapter have been published in the proceedings of the CVPR Workshop on Event-based Vision [173] and in the journal Nature Communications [172]. Some of the results described in this chapter have also been presented in a talk at the workshop Spiking Neural networks as Universal Function Approximators (SNUFA) 2023.

The chapter will be organized as follows: we will first be introducing PCL. Second, we will describe the results obtained on the emergence of simple and complex cell responses properties in PCL and study the impact of various STDP rules on the emergence of such responses properties. Third, we will demonstrate that PCL showcases different extra-classical receptive field effects that are generally seen as hallmarks of predictive coding. Lastly, we will evaluate PCL on two datasets used in neuromorphic vision, to demonstrate the interest of the extra-classical receptive field effects in information coding.

4.1 Introduction

As discussed in Chapter 2, predictive coding has been shown to encompass many other theories of brain function such as efficient coding and sparse coding, and has thus acquired significant popularity as one of the potential strategies employed by the brain to attain high energy efficiency. As spikes are a major source of energy consumption in the brain [12], [140], and that neurons in a predictive coding network reduce spikes by sending only prediction-errors, we here employ a novel predictive coding scheme that we implement with spiking neural networks, *Predictive Coding Light* (PCL).

To build PCL we take inspiration from the seminal work of Rao and Ballard [195] introduced in Chapter 2. Specifically, we also use the PCL network to mimic V1's simple cells but also complex cells. We also take inspiration from spiking models of predictive coding reviewed in Chapter 3. For instance, PCL share similarities with seminal works on spiking predictive coding such as [193], as both works use STDP to learn synaptic connections that can be used to predict network activity.

Essentially, the PCL network is a hierarchical spiking neural network composed of 2 layers that model, respectively, simple cells and complex cells observed in V1, such that simple cells take inputs from an event-based sensor and then send their output spikes to complex cells (Figure 4.1a). The PCL network is part of Class 3 of spiking predictive coding models as it implements spiking predictive coding neither with explicit error neurons as in Class 1 or in the membrane potential of neurons as in Class 2. Rather, similarly to other models of Class 3, PCL's prediction errors are implemented *implicitly* within the network's activity by using different lateral inhibitory connections amongst simple cells, as well as top-down connections from complex cells to simple cells. Using this scheme and paired with inhibitory STDP learning, any simple cell spike that can be sufficiently well predicted by a neighboring cell, or by a complex cell, would be suppressed through inhibition. In this sense it is a "light" version of predictive coding that may be easier to reconcile with neurophysiological evidence and offer advantages for the engineering of neuromorphic information processing systems.

PCL follows a simple motivation: reducing energy consumption by removing predictable spikes. It realizes this function in a novel way compared to other models of Class 3 as it is, to our knowledge, the only model of that class that can replicate simple and complex cell-like responses properties while exhibiting also the different aforementioned nonclassical receptive field effects (surround suppression, orientation-tuned suppression and cross-orientation suppression). Other models of Class 3 study how prediction-error like responses can occur in a spiking neural network with toy examples, rather than more complex ones as done with PCL. Such neurons' responses

have also not been observed in models of Class 1 and 2. The information coding study done by the PCL network with digit and gesture recognition is also to our knowledge the only example of the impact of suppression on downstream tasks, in a spiking neural network.

We next present the methodology employed to implement the PCL network.

4.2 Methods

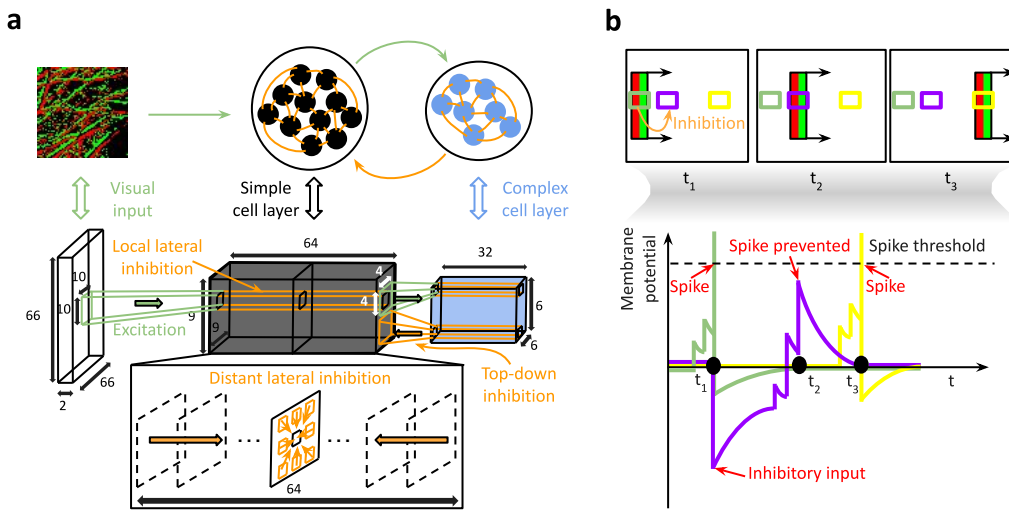


Figure 4.1: **Predictive Coding Light (PCL) network.** **a, top:** Illustration of a two-layer PCL network comprising a “simple cell” (black) and a “complex cell” (blue) layer driven by visual input from an event-based vision sensor. Feedforward connections (green) are excitatory while feedback and recurrent connections (orange) are inhibitory. All connections are learnt via unsupervised spike timing-based learning rules. **bottom:** Detailed network architecture illustrating sizes of receptive fields and numbers of feature channels. **b,** suppression of predictable spikes through learnt inhibitory connections. The top panel illustrates a hypothetical visual input comprising a vertical bar moving from left to right. The leading and trailing edge of the bar create *on* (green) and *off* (red) events on the sensor array, respectively. At different time points, these events drive activity in three simple cells with different receptive field locations (colored boxes). The bottom panel illustrates the expected behavior of the neurons’ membrane potentials for this input. The appearance of the bar is assumed unexpected and reliably drives firing in the green neuron. In contrast, strong activation of the middle neuron (purple) is highly predictable and has caused the growth of a strong inhibitory connection onto this neuron. This inhibition manages to cancel the predictable spike (thereby rendering it less predictable). Finally, the yellow neuron, not being under the influence of strong inhibition, generates a spike.

4.2.1 Network architecture

The PCL network follows a 2-layer architecture where the first layer is inspired from simple cells and receives a visual input from an event-based vision sensor, whereas the second layer is inspired from complex cells and take inputs from the simple cell layer. Figure 4.1a illustrates the compact and detailed network’s connectivity.

Essentially, we model the properties of V1’s simple and complex cells through a combination of both local and distant plastic lateral inhibition, and feedforward excitation initiated from a spiking input received by an event-based vision sensor. A unit in the PCL network can therefore excite some of its targets and inhibit others, i.e., the PCL network does not respect Dale’s law. All synaptic connections in the PCL network are learnt via different spike timing-dependent learning rules, that we describe in Sec. 4.2.2.1. All of the PCL network’s connectivity, plasticity and neural parameters are summarized in Tab. 4.1.

Simple cells Units in the simple cell layer receive events from the event camera and excite units in the complex cell layer. Simple cells also receive three kinds of inhibition. The first is a local lateral inhibition from other simple cells with identical receptive field location. This mutual inhibition facilitates simple cells developing diverse tuning properties. The second is a distant lateral inhibition. This allows the network to learn to suppress spikes that are highly predictable based on the surrounding context. The third is a top down inhibition from the complex cell layer. Similar to the distant lateral inhibition it enables the network to learn to suppress highly predictable simple cell spikes based on the compressed and more abstract representation in the complex cell layer. The idea behind the distant lateral inhibition and the top-down inhibition is described in Fig. 4.1b, where a neuron cancels through inhibition the spike of another neuron that is located retinotopically elsewhere.

Complex cells Units in the complex cell layer are driven by excitatory inputs from the simple cell layer. Similar to the simple cells, complex cells also use a local lateral inhibition scheme to facilitate the learning a broad range of tuning properties. Furthermore, complex cells send inhibitory feedback connections to simple cells to cancel highly predictable spikes.

4.2.1.1 Neuron model

Model neurons in the PCL network are leaky integrate-and-fire (LIF, see Chapter 2) units with a relative refractory period mechanism. In the absence of any input, a neuron’s membrane potential will decay to a resting value V_{rest} which we define as 0 mV in this work. When a neuron receives an input spike, its membrane potential will increase or decrease according to the sign and weight of the synaptic connection. To ensure that a neuron’s membrane potential does not reach unrealistically low values, we also define a minimum membrane potential value V_{min} that we set to -20 mV. Our simulation is event-based, i.e., membrane potentials are updated only

when a synaptic input w_i is received according to:

$$\tilde{V}(t + \Delta t) = \max\{V_{\min}, V(t)e^{\frac{-\Delta t}{\tau_m}} + w_i(t) - \eta_{RP}e^{-\frac{t+\Delta t-t_s}{\tau_{RP}}}\}, \quad (4.1)$$

where Δt is the time since the last update of the membrane potential, τ_m is the membrane time constant, η_{RP} is the amplitude of the trace of the relative refractory period, t_s is the time since the neuron last spiked and τ_{RP} is the time constant of the relative refractory period.

If the membrane potential of a neuron reaches the firing threshold $V_\theta = 30$ mV, the neuron generates a spike that is sent to other neurons. After a spike, the membrane potential is set to a reset value $V_{\text{reset}} = -10$ mV. The membrane potential therefore follows:

$$V(t + \Delta t) = \begin{cases} \tilde{V}(t + \Delta t) & : \tilde{V}(t + \Delta t) < V_\theta \\ V_{\text{reset}} & : \tilde{V}(t + \Delta t) \geq V_\theta. \end{cases} \quad (4.2)$$

The firing threshold V_θ and the minimum membrane potential $V_{\min}$ were chosen such that their difference to the resting membrane potential V_{rest} is in the same range as biologically observed values.

4.2.2 Synaptic plasticity mechanisms

4.2.2.1 Spike Timing-Dependent Plasticity rules

We use three forms of spike timing-dependent plasticity (STDP): a causal STDP (cSTDP), an acausal STDP (aSTDP) and a symmetric STDP (sSTDP) (see Fig. 4.2) that are all observed in different regions of the brain [41]. We systematically test the effect of each form on the learning of synaptic excitatory and inhibitory weights.

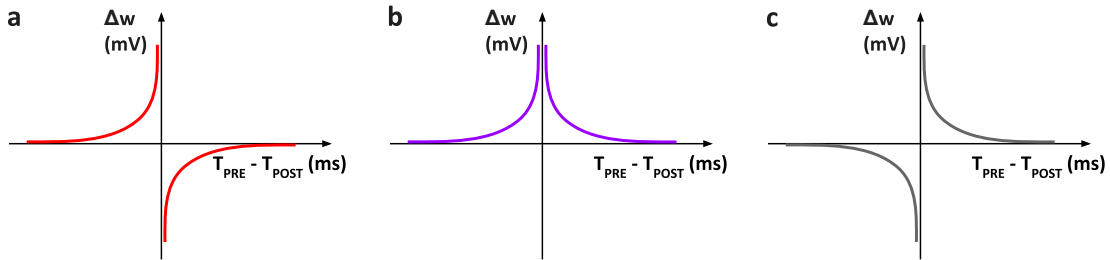


Figure 4.2: **Spike Timing-Dependent Plasticity rules:** T_{PRE} designates the timing of the presynaptic spike and T_{POST} designates the timing of the postsynaptic spike. **a**, Causal STDP (cSTDP). A presynaptic spike occurring shortly before a postsynaptic spike leads to a weight increase, whereas a presynaptic spike that occurs shortly after a postsynaptic spike leads to a weight decrease. **b**, Symmetric STDP (sSTDP). A presynaptic that occurs close to the time of the postsynaptic spike (whether before or after) leads to a weight increase. **c**, Acausal STDP (aSTDP). A presynaptic spike occurring shortly before the postsynaptic spike leads to a weight decrease, whereas a presynaptic spike that occurs shortly after a postsynaptic spike leads to a weight increase.

Causal STDP It is the most standard form of STDP in which depending on the timing of a neuron’s pre and postsynaptic spikes, we perform long term potentiation and depression (LTP and LTD). For an i -th synaptic input to a neuron, both LTP and LTD are triggered when there is a postsynaptic spike. At the moment of the spike, the synaptic weight is updated according to:

$$\tilde{\Delta}w_i^{\text{LTP,c}} = \eta_{\text{LTP}} e^{\frac{t_i - t_s}{\tau_{\text{LTP}}}}, \quad (4.3)$$

$$\tilde{\Delta}w_i^{\text{LTD,c}} = -\eta_{\text{LTD}} e^{\frac{t_{s-1} - t_i}{\tau_{\text{LTD}}}}, \quad (4.4)$$

where t_i is the time when the last spike from input i was received, t_s is the spike time of the postsynaptic neuron, η_{LTP} and τ_{LTP} set the scale and time constant of the potentiation temporal window, t_{s-1} is the previous spike time of the postsynaptic neuron, and η_{LTD} and τ_{LTD} set the scale and time constant of the depression temporal window. To foster the learning of more invariant representations in complex cells, they use larger LTP and LTD time constants than simple cells (40 ms instead of 7 ms, see Tab. 4.1) In the “standard” version of a PCL network, we use cSTDP for all of our connections.

Acausal STDP Oppositely to synaptic updates with cSTDP, in aSTDP, potentiation and depression windows are reversed:

$$\tilde{\Delta}w_i^{\text{LTP,a}} = \eta_{\text{LTP}} e^{\frac{t_{s-1} - t_i}{\tau_{\text{LTP}}}}, \quad (4.5)$$

$$\tilde{\Delta}w_i^{\text{LTD,a}} = -\eta_{\text{LTD}} e^{\frac{t_i - t_s}{\tau_{\text{LTD}}}}, \quad (4.6)$$

where all parameters stay the same.

Symmetric STDP Contrary to cSTDP and aSTDP, there is no depression part in sSTDP. The weights of an input neuron are updated as follows:

$$\tilde{\Delta}w_i^{\text{LTP,s}} = \eta_{\text{LTP}} \left(e^{\frac{t_i - t_s}{\tau_{\text{LTP}}}} + e^{\frac{t_{s-1} - t_i}{\tau_{\text{LTP}}}} \right), \quad (4.7)$$

where all parameters again stay the same.

4.2.2.2 Soft-bound synaptic plasticity

We use a soft-bound mechanism to keep synaptic weights within allowed maximum and minimum values w_{max} and w_{min} . This operates via the two functions:

$$A_+(w_i) = (w_{\text{max}} - w_i)\eta_+, \quad (4.8)$$

$$A_-(w_i) = (w_i - w_{\text{min}})\eta_-, \quad (4.9)$$

where η_+ and η_- are positive constants. The overall STDP updates are given by:

$$\Delta w_i^{LTP} = A_+(w_i) \tilde{\Delta} w_i^{LTP}, \quad (4.10)$$

$$\Delta w_i^{LTD} = A_-(w_i) \tilde{\Delta} w_i^{LTD}, \quad (4.11)$$

For simplicity, we choose $w_{\min} = 0$ for all synaptic weights.

Note that should a synaptic weight w_i become larger than $w_{\max}$, events that normally trigger LTP will instead produce LTD, since the softbound function $A_+(w_i)$ will be negative for $w_i > w_{\max}$. An analogous effect occurs if a weight becomes smaller than $w_{\min}$.

4.2.2.3 Weight normalization

The PCL network uses a simple weight normalization to model the competition of a neuron's synapses for a limited supply of synaptic building blocks such as neurotransmitter receptors or scaffolding molecules [240]. Immediately after the update of a neuron's weight w_i

$$w_i \leftarrow w_i + \Delta w_i^{LTP} - \Delta w_i^{LTD} \quad (4.12)$$

all the neuron's weights of the same type (e.g. excitatory) are normalized according to:

$$w_i \leftarrow \lambda \frac{w_i}{\sum_i w_i} \quad \forall i, \quad (4.13)$$

where λ is the normalization target or amount of synaptic resources for this connection type.

4.2.2.4 Weight sharing

The PCL network uses weight sharing for the excitatory and local lateral inhibitory connections formed by both simple and complex cells. The weight sharing is implemented such that whenever the excitatory or local inhibitory weights of *one* neuron are changed via the strictly local STDP rule (described below), all *other* neurons encoding the same feature (but at different retinal locations) also get their weights modified to the same value. Specifically, after each individual neuronal weight update, all excitatory and local lateral inhibitory synapses are further updated as follows:

$$\forall \mathbf{z}, \forall \mathbf{j} \neq \mathbf{k}, \mathbf{w}_i^{\mathbf{z}, \mathbf{j}} \leftarrow \mathbf{w}_i^{\mathbf{z}, \mathbf{k}}, \quad (4.14)$$

where z gives the feature channel, j gives the retinal location of the neuron, k is the retinal location of the neuron that received the last synaptic weight update and w_i denotes an excitatory or local lateral inhibitory synaptic weight indexed by i . While

such weight sharing is not biologically plausible, it reduces the number of trainable parameters. Other biologically inspired models [155], [200] also report the use of such a mechanism. We emphasize, however, that distant lateral and top-down inhibitory connections are **not** shared but specific to each neuron.

4.2.3 Network parameters

A common problem when working with spiking neural networks is their large number of parameters. As a spiking framework, the PCL network is no exception of this rule and also requires the choice of many parameters. Table 4.1 summarizes the different parameters used in the experiments of this chapter. We will here be giving a short walkthrough to gain an insight on the choice of some of those parameters.

4.2.3.1 Connectivity parameters

The most common parameters to first pick when building any type of neural network are the architecture/connectivity parameters. While event cameras, such as iniVation’s DVS346, give frames of dimensions $346 \times 260 \times 2$, (note that the last dimension designates the polarity of the event) we here reduced the dimensionality of the input to a pixel map of dimensions $66 \times 66 \times 2$ to reduce our number of PCL neurons which has a direct effect on the computational demands of the simulation.

As real neurons can have overlapping receptive fields, we also allowed simple (respectively complex) cells to have an overlap between them, such that any input would be reasonably centered on at least one simple (respectively complex) cell to give a strong response. Inputs received at the border of the retinal map represent however an exception, as we do not implement padding. We chose the overlap between cells as well as the number of neurons (and thus a neuron’s input zone) as a tradeoff between computational constraints (memory and processing speed) and network complexity.

In addition to those standard architectural parameters, the novelty in PCL compared to other spiking frameworks modeling V1, is the introduction of distant lateral inhibition and top-down inhibition. For the distant lateral inhibition in simple cells, we picked a large range of dimensions $4 \times 4 \times 64$ which means that simple cells can inhibit in any direction of the retinal map, other simple cells (not sharing the same input zone) within this range. As for the top-down inhibition, any simple cell is inhibited back by complex cells they project to.

Table 4.1: Connectivity, neurons' membrane potential and plasticity parameters. Connectivity parameters values indicate dimensions of the following form: $(x \times y \times z)$ cells.

Simple cells' connectivity parameters:

Retina size	Input zone	N_{cells}
$(346 \times 260 \times 2)$	$(66 \times 66 \times 2)$	$(9 \times 9 \times 64)$
receptive field	Overlap	Plastic lateral inhibition range
$(10 \times 10 \times 2)$	$(3 \times 3 \times 2)$	$(4 \times 4 \times 64)$

Complex cells' connectivity parameters:

Input size	Input zone	N_{cells}
$(9 \times 9 \times 64)$	$(9 \times 9 \times 64)$	$(6 \times 6 \times 32)$
receptive field	Overlap	Plastic top-down inhibition range
$(4 \times 4 \times 64)$	$(3 \times 3 \times 64)$	$(4 \times 4 \times 64)$

Simple cells' membrane potential and plasticity parameters:

V_{reset} (mV)	V_{thresh} (mV)	$\tau_{\text{m/I}}$ (ms)	τ_{RP} (ms)	τ_{LTP} (ms)	τ_{LTD} (ms)
-10	10	18	5	7	7
η_{RP} (mV)	w_{min} (mV)	w_{max} (mV)	$\eta_{\text{LTP/LTD}}$ (mV)	$\eta_{+/-}$	λ
10	0	excit: 3 dist. inhib.: 50 loc. inhib.: 50 td. inhib.: 50	excit: ± 0.000204 dist. inhib.: ± 2.652 loc. inhib.: ± 0.816 td. inhib.: ± 1.428	excit: 0.33 dist. inhib.: 0.02 loc. inhib.: 0.02 td. inhib.: 0.02	excit: 50 dist. inhib.: 6500 loc. inhib.: 1500 td. inhib.: 3500

Complex cells' membrane potential and plasticity parameters:

V_{reset} (mV)	V_{thresh} (mV)	τ_{m} (ms)	τ_{RP} (ms)	τ_{LTP} (ms)	τ_{LTD} (ms)
-10	3	50	5	40	40
η_{RP} (mV)	w_{min} (mV)	w_{max} (mV)	$\eta_{\text{LTP/LTD}}$ (mV)	$\eta_{+/-}$	λ
5	0	excit: 4 loc. inhib.: 25	excit: ± 0.08 loc. inhib.: ± 0.048	excit: 0.25 loc. inhib.: 0.04	excit: 1000 loc. inhib.: 600

4.2.3.2 Membrane parameters

The computations done by spiking neurons are highly dependent on the parameters controlling the behavior of their membrane potential. Such parameters are: the membrane time constant, the spiking threshold, the reset value of the membrane potential, the minimum value of the membrane potential, the amplitude and the time constant of the refractory period. Depending on their choice, a neuron's firing rate could either be:

- too high: and therefore computationally expensive as each new spike leads to a new set of instruction since the simulation is event-based
- too low: and therefore causes information loss as neurons cannot encode and transmit input spikes.

In order to constrain the parameters and avoid the aforementioned extreme cases, we take inspiration from biological neurons and set all of those parameters in a range similar to what would be expected of real neurons. In general, the membrane time constant is of the order of milliseconds and increases with layer's depth. Similarly, the refractory period of a neuron lasts only some milliseconds, and its membrane potential tends to be strictly kept within a range that spans about 40 mV. The span of our PCL neurons' membrane potential is below this range, with 30 mV for simple cells and 23 mV for complex cells to generate spikes with all types of inputs considered here.

4.2.3.3 Synaptic weight and plasticity parameters

Finally, the scale of the synaptic weights as well as the speed at which plasticity occurs and its timescale also play a major role on what a neuron would (learn to) encode.

In terms of synaptic weight scales, we had to find values purely empirically after fixing the membrane potential parameters. Essentially, the scales λ , w_{min} , w_{max} are all set in order to reach some "stable" level of neural activity which is arbitrarily decided according to known constraints (e.g. computational cost of spiking, coding accuracy). Please note however that the distribution of synaptic weights of a neuron (whether excitatory or inhibitory) is initialized from a random uniform distribution.

As for the plasticity parameters, we here again take a look at biology to set them. In general, the STDP windows of layers of neurons tend to increase the deeper the layer is. This observation is thought to arise to build invariance. An example to illustrate this hypothesis would be the case of a neuron being excited by an input randomly moving in its receptive field: with a large STDP window, inputs at distant

timesteps can contribute equally to the spiking of a neuron, and the neuron would therefore become invariant to the exact position of the input in its receptive field. For this reason, we use a larger STDP window for complex cells compared to simple cells. Regarding the learning rates of STDP, similar to ANNs, we had to find them empirically. Admittedly, the process of finding them is more arduous than ANNs due to the spiking nature of the input (activity changes at each level and the learning rate thus needs to be different at each layer), and the number of synapse types.

4.3 How inhibitory STDP removes redundant spikes

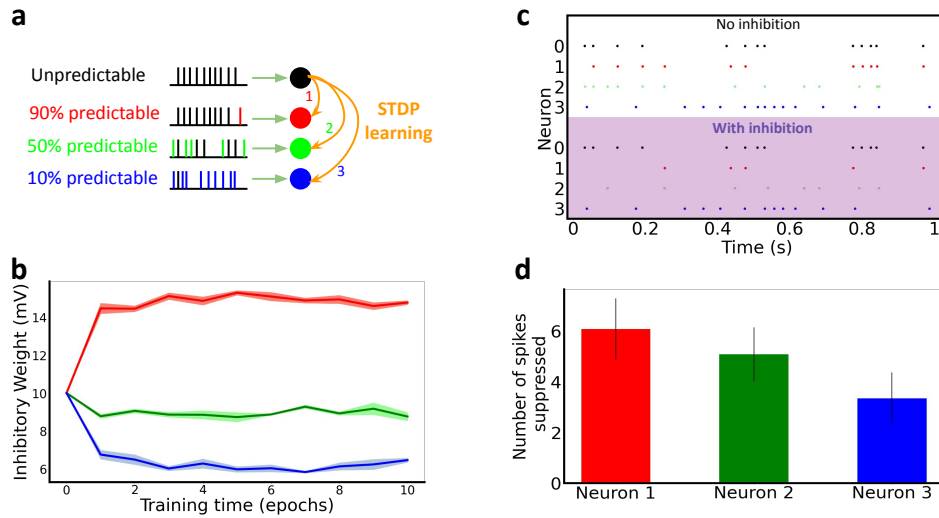


Figure 4.3: **An inhibitory STDP rule leads to the removal of the most predictable spikes in a spiking neural neural network.** **a**, Experimental protocol. 4 neurons named “neuron 0”, “neuron 1”, “neuron 2” and “neuron 3” receive poisson spike trains as input. Neuron 0 is considered the “main neuron” that receives fully unpredictable poisson spike train inputs, while other neurons exhibit redundancy with this initial spike train (90% for neuron 1, 50% for neuron 2 and 10% for neuron 3). Neuron 0 grows inhibitory connections to neurons 1 to 3 that are learnt via an STDP rule on 140 samples for 10 epochs, while the excitatory weights are set to a same constant value of 15 mV with a neural threshold at 10 mV. The inhibitory connections undergo competition through L1-normalization. **b**, Evolution of the inhibitory synaptic weights during training averaged on 3 networks. The shaded area denotes the standard error. **c**, Raster plot of the network’s responses without inhibition (top) and with inhibition (bottom) on an example test sample. Responses of neurons 1 and 2 are significantly suppressed when inhibition is present, contrary to the case without inhibition. **d**, Suppression of neurons 1-3 averaged over 3 networks. Error bars denote the standard error.

The central idea behind Predictive Coding Light is that inhibitory STDP (iSTDP) permits the network to learn to suppress the most predictable spikes. Although the iSTDP rule does not put an explicit constraint on the predictability of any spikes, it naturally gives rise to the suppression of the most predictable spikes via inhibitory synapses.

Specifically, if a presynaptic neuron tends to spike shortly before a postsynaptic one, the causal iSTDP rule will grow the inhibitory synapse between the two neurons. Once this synapse becomes strong enough it will become effective at suppressing

the predictable spikes of the postsynaptic neuron. At the network level, iSTDP will come to capture the statistics of neural firing and allow the removal of the most predictable spikes.

4.3.1 Experimental protocol

To illustrate this idea we realize a simple experiment with just 4 neurons (see Fig. 4.3a). In this experiment, the presynaptic neuron (black) is driven by excitatory inputs from a poisson spike train and has inhibitory synapses onto three other neurons (red, green, blue). These neurons are partly driven by the same poisson input, albeit with a small delay, such that a percentage of their spikes is predictable from the black neuron’s spike train. The inhibitory synapses from the black neuron to the other 3 neurons are initialized to the same value and evolve according to the causal STDP rule.

We first generate four different poisson spike trains (one for each neuron) with a constant firing rate using a poisson spike generator. We then randomly remove a certain percentage of the spikes from neuron 1–3’s input spike trains (90% for neuron 1, 50% for neuron 2 and 10% for neuron 3) and replace the removed spikes with randomly selected spikes (without replacement) from the spike train of neuron 0, but shifted 1 ms in time. This way, the predictability of the input spike trains to neurons 1–3 given the spike train of neuron 0 is fully controlled.

We generate 140 spike trains of 1 second length with a firing rate of 20 Hz to train the inhibitory weights, and use 35 additional spike trains for testing the effects of these connections. These spike trains are fed to neurons 0–3 with fixed excitatory synaptic weights that have a value of 15 mV. The parameters of all neurons and the inhibitory synaptic plasticity are the same as the ones described in Tabs. 4.1 (for the distant lateral inhibition).

Importantly, while the L1-normalization is normally applied to *incoming* weights, for this simple example, due to the presence of only one inhibitory synapse for each neuron, we instead apply it on *outgoing* lateral inhibitory weights. Fundamentally, for neurons in the brain there is arguably competition for synaptic resources among both outgoing and incoming synapses which motivates the use of either form of normalization.

4.3.2 Results

Figure 4.3b shows the evolution of the inhibitory synaptic weights during training. As expected, the iSTDP rule causes the inhibitory synapses to grow stronger the higher the predictability of the postsynaptic spikes, with neuron 1 reaching the

highest inhibitory weight and neuron 3 reaching the lowest one while neuron 2's is in-between the two.

The resulting inhibition has a direct impact on the spiking activity of the network as it leads to different levels of suppression of spikes for neurons 1–3, as depicted in Fig. 4.3c for one example test sample. In this example sample, we observe that when inhibition is deactivated in the network, all neurons have a high spiking activity, meanwhile the activation of inhibition leads to a significantly reduced activity for neuron 1 and only a minimal reduction for neurons 2 and 3.

This observation holds up over multiple samples, with neuron 1 having more spikes suppressed than neurons 2 and 3 (Figure 4.3d). Inhibition onto neurons that exhibit highly predictable spikes can therefore lead to the suppression of these spikes when the inhibition is strong enough to overcome the excitatory driving input.

Overall, those results highlight the capability of iSTDP to learn inhibitory synaptic connections capable of removing the most predictable spikes from the network's activity. In the next sections, we show how the PCL network can use this effect in more complex cases and acquire neural tuning properties similar to V1.

4.4 How PCL learns simple and complex cell-like responses

With the PCL network, we are interested in modeling V1's simple and complex cells. Considering the spiking nature of the network, it is necessary to use a spiking input which event cameras readily produce as they mimic the biological retina (see Chapter 2). In mammals' V1, the responses patterns of simple and complex cells are thought to emerge after exposure to a wide variety of input stimuli after birth. For instance, to develop vision, humans have to experience different type of visual stimuli such as stimuli found in nature (e.g. trees, grass, lakes etc.) but also visual sequences of buildings as well as other humans. It is only after such exposure that mature simple and complex cells are able to emerge, without supervision, in V1.

In order to see if the PCL network can reproduce those properties, we thus use different types of event-based stimuli that we use to train and test our network. Essentially, we use an event-based camera simulator to generate those stimuli, such that we have:

- a training dataset where the Van-Hateren natural images database [242] composed of different images observed during natural vision experience have been converted to events.

- different grating datasets for testing of the general response properties of simple cells and complex cells as well as suppression.

We next give details as to how those datasets were generated, before providing the results after training of the PCL network.

4.4.1 Experimental protocol

We first present our event-based camera simulator and then describe the generated training and testing datasets. Afterward, we present the models against which simple and complex cells' tuning curve are compared.

4.4.1.1 Event-based camera simulator

We generate event-based stimuli from RGB image sequences using the PIX2NVS event-based camera simulator [28]. This simulator calculates the logarithmic difference between pixel values of two successive frames. When this difference exceeds a moving threshold, the pixel can fire a random number of temporally jittered events (between 1 and $n_{\max} = 5$, depending on the ratio between the logarithmic difference and the threshold). The moving threshold is updated individually for each pixel after any event and has minimum and maximum allowed values of 0.2 and 0.4. Other simulators such as the v2e [104] and ICNS event-based camera simulators¹ could be used to provide more authentic event streams (notably with noise and motion blur non-idealities), but the PIX2NVS event-based camera simulator offers a simpler pipeline sufficient for the purposes of the current study.

4.4.1.2 Natural Image and Grating Datasets

First, we generate a training set composed of event sequences of the Van Hateren natural images database [242] with events moving in 16 random directions that cover the scope of 360° . Event sequences from the natural images dataset comprise 2000 sequences that each have a duration of 800 ms. We use such sequences to learn simple and complex cells' receptive fields, as those sequences are observed during natural vision experience. During training, each sequence is repeated twice and the duration experienced by the network is actually 1600 ms. Second we generate four different test sets:

- a sinusoidal counterphase grating set of various orientations, spatial frequencies and phases for the testing of simple and complex cell response properties.

¹<https://github.com/neuromorphicsystems/IEBCS>

- a drifting grating test set composed of sequences of different orientations and different patch sizes, but with a fixed spatial frequency, motion direction and speed, for the testing of surround suppression
- a second drifting grating test set composed of sequences of different orientations between center and surround, but with a fixed spatial frequency, motion direction and speed for the testing of orientation-tuned suppression.
- a final drifting grating test set in which we superpose two gratings of different orientations but identical spatial frequency, motion direction and speed for the testing of cross-orientation suppression.

All grating sequences have a duration of 2 s. Please see respectively Figs 4.4a (lefthand side) and 4.9a for examples of event-based natural images samples used for training and gratings used for testing the selectivity of simple cells/complex cells and suppressive effects.

Example videos are available at: <https://zenodo.org/records/16813455>.

4.4.1.3 Gabor and energy models of simple and complex cells

We compare receptive fields of units in the PCL network with classic Gabor filter and energy models of simple and complex cells in primary visual cortex as discussed by Dayan & Abbott [50]. In the following, we briefly present these models.

The Gabor filter [79] has traditionally been used to approximate the spatial receptive field of simple cells. It is given by:

$$D_s(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2})} (\cos(kx - \phi)), \quad (4.15)$$

where x and y are pixel coordinates, σ_x and σ_y characterize the extent of the receptive field in x and y directions, k is the preferred spatial frequency and ϕ is the preferred spatial phase. The temporal characteristics of the receptive field are described by the function proposed by Adelson et al. [1]:

$$D_t(\tau) = \alpha e^{-(\alpha\tau)(\frac{(\alpha\tau)^5}{5!} - \frac{(\alpha\tau)^7}{7!})}, \quad (4.16)$$

where τ indexes time and α is a constant. Using those two functions, the linear estimate of the response of a simple cell to a counterphase sinusoidal grating is given by:

$$L(t) = L_s L_t(t) \quad (4.17)$$

where, L_s is the spatial linear estimate and $L_t(t)$ is the temporal one. Their expressions are given by:

$$L_s = \int dx dy D_s(x, y) A \cos(Kx \cos(\Theta) + Ky \sin(\Theta) - \Phi), \quad (4.18)$$

$$L_t(t) = \int_0^\infty d\tau D_t(\tau) \cos(\omega(t - \tau)), \quad (4.19)$$

where, A is the contrast amplitude of the sinusoidal grating, K is the spatial frequency of the grating, Θ is the grating's orientation, Φ is the grating's phase, ω is the temporal frequency of the grating. t indexes time and D_s and $D_t(\tau)$ are the spatial and temporal components of the simple cell receptive field. The final simple cell response is calculated by applying a half-wave rectification to $L(t)$ to avoid negative responses.

The response of a complex cell can be described using the energy model. The energy model first adds the squared linear responses $L_1(t)^2$ and $L_2(t)^2$ of two simple cells with the same orientation and frequency preferences but of opposite phase preference. The full complex cell response to a counter-phase grating can be written as:

$$r(t) = r_0 + G(L_1(t)^2 + L_2(t)^2) \quad (4.20)$$

where, L_1 and L_2 are the linear responses of simple cells with opposite phase preference, r_0 is a baseline firing rate and G is a scaling factor. Details on the parameters and derivations of the above equations can be found in Chapter 2 of the textbook by Dayan & Abbott [50].

4.4.2 Results

4.4.2.1 Simple and complex cell-like response properties arise from unsupervised learning on natural images.

Given its biological inspiration, we wondered if cells in the PCL network might reproduce findings about simple and complex cells in visual cortex. To this end we train the network on a database of natural images (Fig. 4.4a). After training with a causal STDP rule, we observe the formation of simple and complex cell-like receptive fields in the network. Most simple receptive fields are well fit by Gabor functions (see examples in Figure 4.4b and analysis in Fig. 4.5a-b). Figure 4.4b (top right-hand side) visualizes learnt complex cell receptive fields. For each complex cell we plot the set of simple cell receptive fields that have the strongest connection to this complex cell from a particular retinal location. Brightness of the simple cell receptive field indicates the strength of the connection. Complex cells tend to pool responses from simple cells with similar orientation and frequency preferences (Figure 4.5c-d). In contrast, the phase and polarity of the pooled simple cell receptive fields is quite variable, as would be expected for biological complex cells.

To quantify the tuning properties of the PCL network's simple and complex cells, we stimulate the network with sinusoidal counterphase gratings (Fig. 4.4a) and record the neural responses. Figures 4.4c and d compare the tuning properties of simple and complex cells in the PCL network to responses of the classic Gabor filter

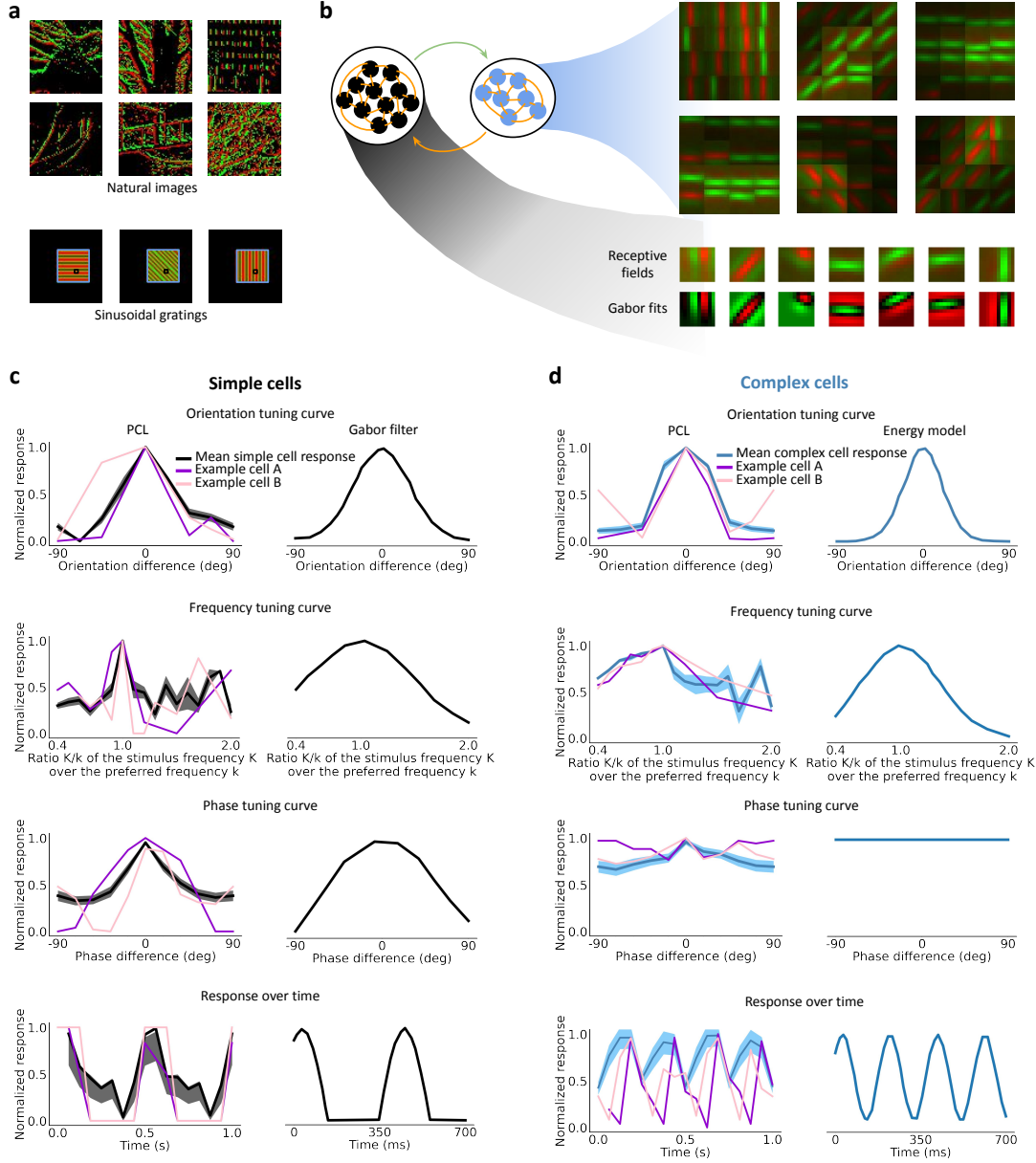


Figure 4.4: **Development of simple and complex cell-like receptive fields in the PCL network.** **a**, The PCL network is trained on event streams generated by natural images (top) and tested with sinusoidal gratings (bottom). Red and green colors denote *on* and *off* events, respectively. **b**, Visualization of example receptive fields learnt by simple and complex cells. Example simple cell receptive fields are shown on top of their Gabor fits. Complex cells receptive fields are visualized by showing for each of six example complex cells the receptive fields of their input simple cells that have the strongest connection to them for each of 16 simple cell receptive field locations. The brighter the colors the stronger the connection. **c**, Comparison of the tuning properties of simple cells in the trained PCL network to tuning properties of the classic Gabor filter model [79], [116]. In each plot we show two example simple cells and an average over all 64 simple cells types. Responses are averaged over 2 trials. Shaded regions indicate one standard error across trials. We show (top to bottom) the orientation tuning curve, the frequency tuning curve, the phase tuning curve and the response over time to sinusoidal gratings. **d**, Comparison of the tuning properties of complex cells in the trained PC network to tuning properties of the classic energy model of complex cells [1]. The mean complex cell responses were calculated over all 32 complex cells. Note the virtually flat phase tuning curve (3rd from top) and the frequency doubling (relative to the simple cells) in response to a moving grating (bottom).

[79], [116] and energy models [1], respectively, which are considered the standard model of biological simple and complex cells [50]. We see that we learn complex

cells selective for the orientation and the frequency. However, they are invariant to the phase of the gratings and also to their polarities, as they keep giving a maximal response in time, despite the switch in polarity of the gratings. Simple cells on the other hand appear sensitive to the polarity and the phase of the bars, on top of their orientation and frequency.

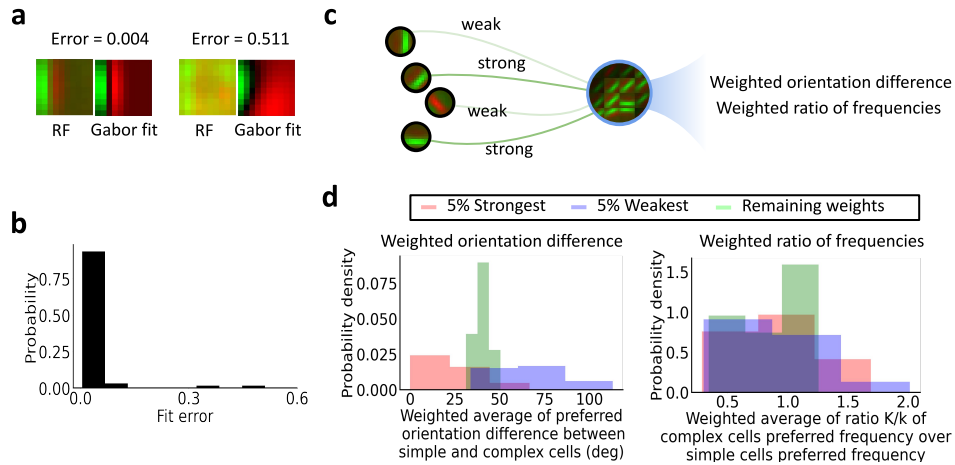


Figure 4.5: **Quantification of simple and complex cells.** **a**, Example of high and low error Gabor fits of simple cells' receptive fields. **b**, Distribution of the Gabor fit errors of the 64 simple cells' receptive fields of one PCL network. **c**, Description of the quantification process of complex cells. A complex cell with a certain tuning property should have stronger connections with simple cells of similar tuning properties. **d**, Tuning of complex cells weights between $4 \times 4 \times 64$ simple cells and 32 complex cells for one PCL network. Left shows the distribution of the weighted average of preferred orientation difference. Right shows the distribution of the weighted average of the ratio of the preferred frequencies. The weighted averages were obtained by weighting neurons' contributions according to the synaptic weights of simple cells connecting to a complex cell. Three histograms are constructed according to the relative strength of the connections: either part of the 5% strongest weights, 5% weakest weights or in-between.

Overall, both simple and complex cells in the PCL network show qualitatively similar tuning properties as their respective standard models.

4.4.2.2 Complex cells emerge from plastic inhibition and depend on simple cells' membrane time constants and plasticity windows

After observing the emergence of simple and complex cell-like response properties in our cells, we investigate the conditions behind the emergence of complex cells. While models of simple cells learning have been proposed before, complex cells are mainly understood as a pooling layer and there has been to our knowledge no spiking model that attempts to explain their emergence.

We first questioned the effect of learning the local lateral inhibition and replaced it with a constant one (fixed to the median local inhibition value of a fully cSTDP learnt network) for both simple and complex cells. Before even getting to complex cells, moving to such a static inhibition strongly impaired the learning of simple cells' receptive fields in our framework such that we could no longer get Gabor filter-like features (Figure 4.6a). As complex cells are dependent on the development of simple

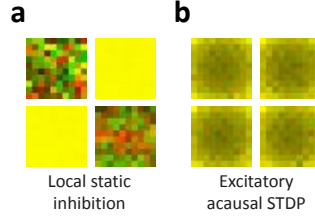


Figure 4.6: **Non-emergence of simple cell-like receptive fields in a PCL network in the absence of causal LTP at all connections.** **a**, Example of 4 simple cells' receptive fields learnt using a cSTDP rule but fixing the value of the local lateral inhibition during training. Some cells (top left and bottom right) stay close to initialization, while others (top right and bottom left) are updated but they do not become selective to Gabor filters. **b**, Example of 4 simple cells' receptive fields learnt using fully plastic connections but an acausal STDP for feedforward connections. All cells are updated but they do not become selective to Gabor filters.

cells, no complex cell-like receptive fields arose in the case where the local lateral inhibition is not plastic.

Second, we tested the importance of our recurrent connections (distant lateral and top-down inhibition) for the emergence of complex cells. We recorded the responses of different PCL networks: 1) our standard PCL network trained with a causal STDP rule for all synaptic connections, 2) a PCL network in which we deactivated our recurrent connections (distant lateral and top-down inhibitions), 3) a PCL network in which we used short STDP windows and membrane time constants for complex cells (7 ms instead of 40 ms for the STDP windows and 21 ms instead of 50 ms for the membrane time constant) and 4) a PCL network in which we used long STDP windows and membrane time constants for simple cells (40 ms instead of 7 ms for the STDP window and 50 ms instead of 21 ms for the membrane time constant).

We see that deactivating our recurrent connections completely breaks the emergence of any sort of selectivity in our complex cells (Figure 4.7a, black curve). In this case, we thus observe complex cells invariant to the orientation, the frequency and the phase of the gratings. Interestingly, the use of a short STDP window and membrane time constants (instead of long ones) for complex cells, still gives results similar to our standard PCL (Figure 4.7a, red curve). However, using long STDP windows and membrane time constants for simple cells (instead of short ones), surprisingly leads to complex cells of poor selectivity (Figure 4.7a, green curve). For instance, such a network gives complex cells sensitive to the phase and the polarity of the gratings instead of cells invariant to them. Furthermore, complex cells of this case also exhibit a weaker orientation selectivity than those of our standard PCL network. On top of those response properties, we also see in Fig. 4.7b that a PCL network without recurrent connections as well as a PCL network that uses simple cells learnt with long STDP windows and time constants, both exhibit fewer orientation diversity, compared to a standard PCL network.

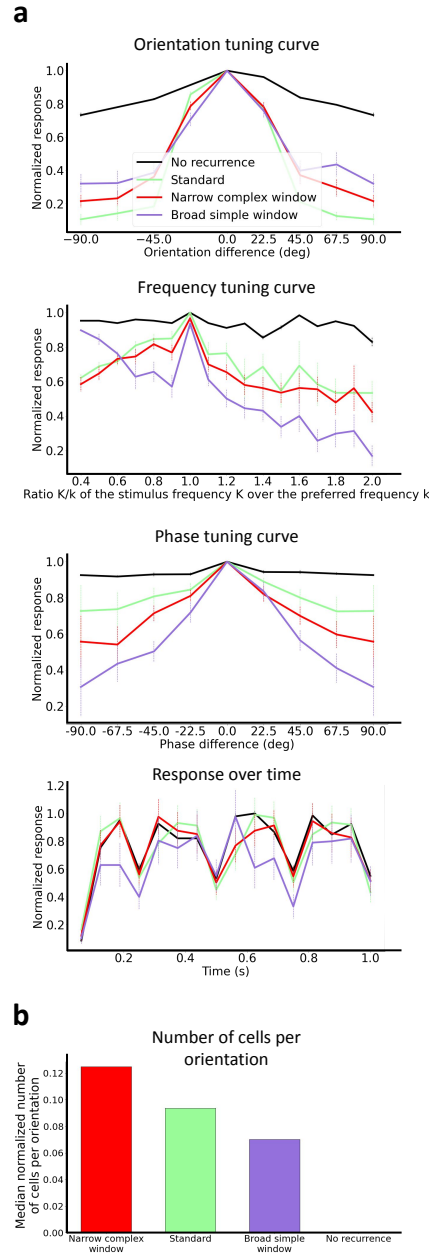


Figure 4.7: Recurrent connections, STDP windows and membrane time constants control the emergence of complex cells in a PCL network. **a**, Tuning properties of complex cells of 4 trained PCL networks: with no top-down/distant lateral inhibition, with a narrow STDP window and short time constant for complex cells, with a broad STDP window and a large time constant for simple cells, and the standard PCL network. In each plot we average over all 32 complex cells types. Responses are averaged over 2 trials. Error bars indicate one standard error across trials. We show (top to bottom) the orientation tuning curve, the frequency tuning curve, the phase tuning curve and the response over time to sinusoidal gratings. **b**, Median normalized number of complex cells tuned to a given orientation in our 4 PCL networks. For each network, each complex cell was tested against different orientations and assigned a favorite orientation according to its maximal response. The number of cells tuned to all orientations was then counted and normalized by dividing it against the total number of cells. The median normalized cell count per orientation of a network was then calculated and plotted here for each network.

Overall, the emergence of complex cells in our PCL network appears to be dependent on the plasticity of inhibition (whether local or recurrent), and on the shape of the STDP windows and time constants of the complex and simple cells.

4.4.2.3 Different STDP rules account for the emergence of simple and complex cells

Different forms of STDP exist but the influence of each of them on sensory learning is still an open question. For this reason, we test various STDP rules (cSTDP, sSTDP and aSTDP) and combine them differently in excitatory and inhibitory synapses to see how they shape simple and complex cells' response properties.

To keep the number of synaptic plasticity rule combinations low, we decided to only use one rule for all excitatory connections, and another rule for all inhibitory connections (distant local, distant lateral and top-down inhibitory connections). This gives a total of 9 combinations. However, we observed during our experiments that using an acausal STDP in excitatory plasticity prevents the emergence of Gabor filter-like simple cells receptive fields (Figure 4.6), so we decided not to consider excitatory aSTDP networks any further and restricted our analysis to the 6 remaining combinations of STDP rules.

We define a simple metric to evaluate the “performance” of our cells with respect to their expected biological properties.

Performance metric In the following paragraphs, we will be using the subscripts o, f, ph and po for P to denote the performance on replicating respectively V1-like orientation, frequency, phase and polarity-dependence effects. The use of different subscripts at once signify that the formula is the same for them. Let the performance metric $P_{o,f,ph}$ of a cell be defined as:

$$\tilde{P}_{o,f,ph} = \frac{1}{N} \sum_{i=0}^N (r_{pref} - r_i), \quad (4.21)$$

where N is the number of stimuli (i.e. the number of gratings with each grating having a varying orientation or a varying frequency or a varying phase), r is the normalized response of a cell to a stimulus, and r_{pref} is the maximal (also normalized) response of the cell. As for the performance on the polarity of the gratings, we first separate the response of a given cell in different time intervals, and take the average response of the cell over those different time intervals, such that: $\tilde{P}_{po} = \frac{1}{T} \sum_{i=0}^T (r_i)$ where T is the number of time intervals, r is again the normalized response and the subscript po denotes the polarity of the grating. For a straightforward visualization of the results, we further made it so that for all type of cells and features, the higher $P_{o,f,ph,po}$ is, the better the “performance” of the network is. Table 4.2 details the expression of the metric for all parameters for simple and complex cells. As simple cells are expected to be selective to the polarity of a stimulus (and therefore have a weak $\tilde{P}_{po}$), we take the opposite of the $\tilde{P}_{po}$ and add 1 to it so that the value stays positive. The same process is applied to complex cells, as they are expected to be

Table 4.2: Performance metrics of simple and complex cells for all parameters. In each case, N still denotes the number of stimuli (i.e. the number of gratings with each grating having a varying orientation or a varying frequency or a varying phase), r still denotes the normalized response of a cell to a stimulus, r_{pref} still denotes the maximal (and normalized) response of the cell and T still denotes the number of time intervals.

	Cell type	
	Simple	Complex
P_o (orientation)	$\frac{1}{N} \sum_{i=0}^N (r_{pref} - r_i)$	$\frac{1}{N} \sum_{i=0}^N (r_{pref} - r_i)$
P_f (frequency)	$\frac{1}{N} \sum_{i=0}^N (r_{pref} - r_i)$	$\frac{1}{N} \sum_{i=0}^N (r_{pref} - r_i)$
P_{ph} (phase)	$\frac{1}{N} \sum_{i=0}^N (r_{pref} - r_i)$	$ 1 - \frac{1}{N} \sum_{i=0}^N (r_{pref} - r_i) $
P_{po} (polarity)	$ 1 - \frac{1}{T} \sum_{i=0}^T (r_i) $	$\frac{1}{T} \sum_{i=0}^T (r_i)$

invariant to the phase of a stimulus (and therefore have a weak $\tilde{P}_{ph}$). Using this performance metric, we systematically compare the effect of different rules on the emergence of V1-like receptive fields for both simple and complex cells, to help us pick the rule that works best.

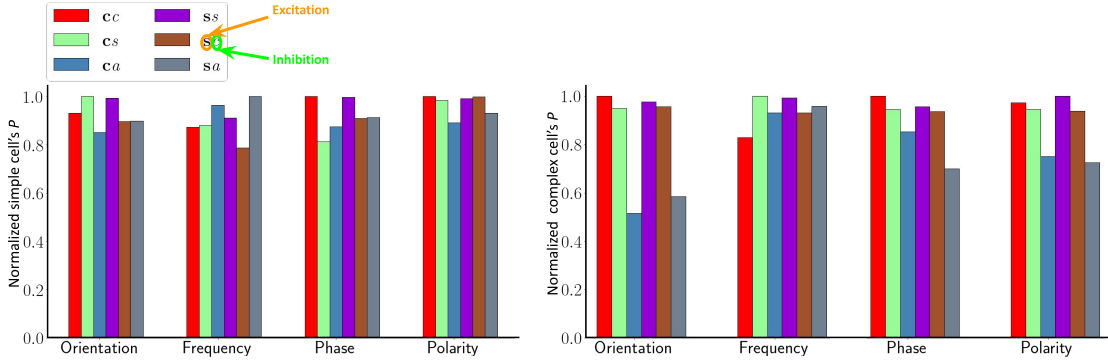


Figure 4.8: **Different STDP rules allow the emergence of simple and complex cell-like response properties in a PCL network.** Left shows the relative performance of PCL’s simple cells with 6 different combinations of STDP rules, while right shows the performance of PCL’s complex cells with the same rules. Performance scores have been normalized such that the highest score in each comparison equals one.

Figure 4.8 shows the performance of the different networks for simple and complex cells. For simple cells, all combinations of rules appear to give similar results. The performances of the various networks are in the same range. It is however interesting to note that the use of only one set of rule (either causal or symmetric) gives slightly better results than other combinations. For complex cells, we can observe distinct results as the use of an acausal inhibitory STDP rule always gives a worse performance than the other rules. Crucially, those results indicate that the combination of a causal and a symmetric STDP can still lead to simple and complex cell-like response properties, but an acausal STDP (whether excitatory or inhibitory) systematically prevents the emergence of V1-like cells.

It is interesting to note that both the causal and the symmetric STDP rule share the identical “causal” LTP behavior for a presynaptic spike arriving shortly before

a postsynaptic spike. Sensory learning might thus strongly depend on this causal LTP behavior, while the depression of the weights might be done either through a causal LTD or, in the case of the symmetric STDP rule (which only produces LTP) through neurons' competition over synaptic resources [240], that we model here with the L1 normalization of the synaptic weights.

4.4.2.4 Emergence of V1-like suppressive effects

Biological simple cells are known to exhibit different classical and nonclassical receptive field effects (see Chapter 2). For example, when a neuron's classical receptive field and its surround are excited simultaneously, simple cells exhibit so-called surround suppression [115] and orientation-tuned suppression [186]. Rao and Ballards's classic predictive coding model has offered an explanation for such phenomena [195], but it remains an open question if spiking predictive coding approaches including PCL exhibit similar effects. So far, other spiking predictive coding models (of all classes) have not presented such findings.

Therefore, we tested if the PCL network exhibits similar effects. Testing the PCL network with test stimuli for surround suppression, orientation-tuned suppression, and cross-orientation suppression (Fig. 4.9a), we find that it qualitatively reproduces experimental findings from primary visual cortex (Fig. 4.9b-d).

First, simple cells in the PCL network exhibit surround suppression when stimulated by gratings of increasing size (Fig. 4.9b). Second, simple cells also exhibit cross-orientation suppression when confronted with a superposition of gratings of different orientations (Fig. 4.9c). Third, simple cells also exhibit orientation-tuned suppression when stimulated with gratings of different orientation in their classical receptive field vs. surround (Fig. 4.9d).

We wondered which of the inhibitory connection types in the PCL network contributes how much to these effects. To investigate this, we selectively disabled one or the other connection type (Figure 4.9b-d (right)). The results reveal that all inhibitory connection types contribute to the reduction in network activity. Abolishing the distant lateral inhibition has only a mild effect on the results in all stimulus conditions (compare blue vs. red curves). In contrast, results can change drastically if the top-down inhibition is removed (orange curves in b,c. The local lateral inhibition also plays a major role (compare green and yellow curves in c).

Next, we wondered to what extent suppression effects may occur in the complex cells of the PCL network. Interestingly, we found that complex cells also exhibit orientation-tuned and cross-orientation suppression but not surround suppression (Figure 4.10). Overall, responses of complex cells are reduced due to the distant lateral and top-down inhibition, but a complex cell's response does not diminish due

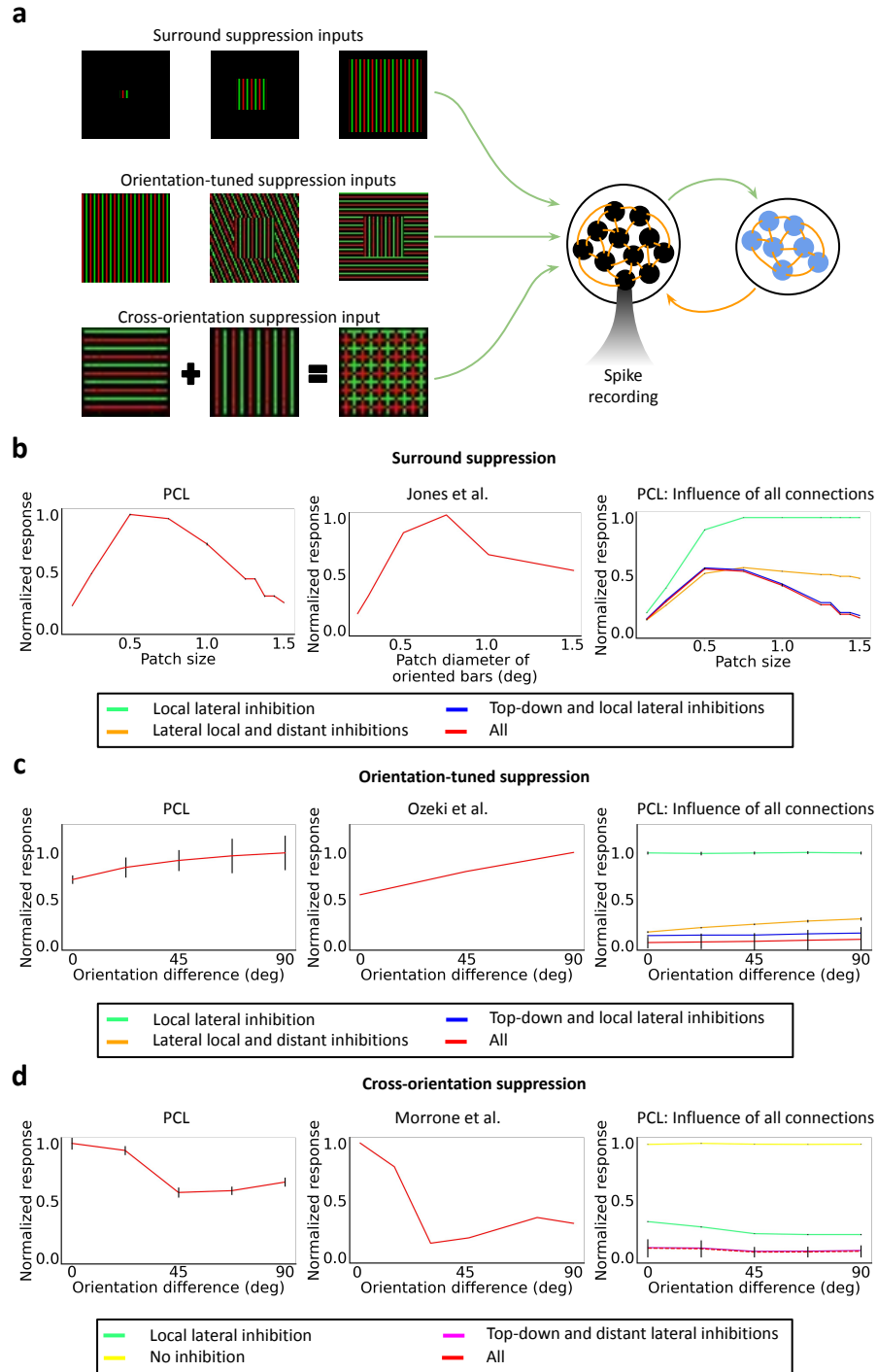


Figure 4.9: **PCL gives rise to surround suppression effects as observed in primary visual cortex.** **a**, Test stimuli and evaluation protocol. A PCL network is excited with gratings of increasing size to reveal surround suppression, gratings of distinct orientations between center and surround for orientation-tuned suppression and a superposition of gratings of distinct orientations for cross-orientation suppression. **b**, Surround suppression in a PCL network compared to data from macaque V1 (reproduced from [115]). Error bars denote one standard error across trials. Responses are averaged over all 64 simple cells and 2 trials. **c**, Orientation-tuned suppression in PCL network compared to data from cat V1 (reproduced from [186]). **d**, Cross-orientation suppression in PCL network compared to data from cat V1 (reproduced from [165]). Rightmost panels in (b–d) show simple cell responses for different types of inhibitory connections being either present or absent.

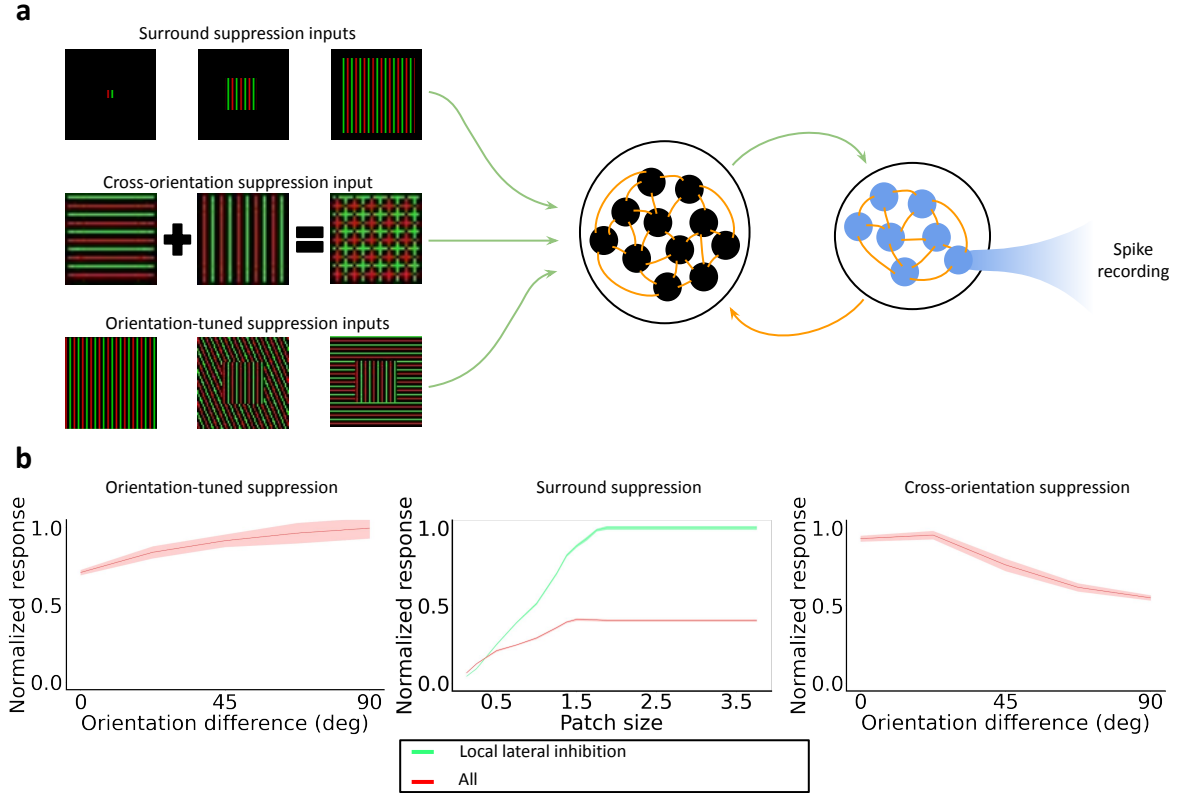


Figure 4.10: **Complex cells also replicate some suppressive effects.** **a**, Test stimuli and evaluation protocol. A PCL network is excited with gratings of increasing size to reveal surround suppression, a superposition of gratings of distinct orientations for cross-orientation suppression and gratings of distinct orientations between center and surround for orientation-tuned suppression. **b**, Orientation-tuned suppression, surround suppression and cross-orientation suppression in a PCL network's complex cells. Responses are averaged over all 32 complex cells and 2 trials. Shaded regions denote one standard error across trials.

to excitation of its extra-classical receptive field. We suspect that this is due to the absence of top-down inhibition to complex cells.

4.5 PCL exhibits efficient information coding on digit and gesture datasets

Above we showed that the PCL network reproduces a range of neurophysiological findings from primary visual cortex. Now we are interested in testing the PCL network on more complex datasets to see first if it can perform downstream tasks (here, classification), and second to verify the impact that suppression has on PCL's performance in those tasks.

For this purpose, we here use two well-known benchmark neuromorphic datasets, the N-MNIST dataset [182] composed of different digits and the DVS128 Gesture dataset [7] composed of gestures. We first give the experimental protocol by giving a

brief description of the datasets and of the classifier that we use. Second, we present the results achieved by the PCL network.

4.5.1 Experimental protocol

4.5.1.1 Datasets

N-MNIST Dataset The Neuromorphic-MNIST (N-MNIST) dataset [182] is a dataset of 60 000 training and 10 000 testing samples of handwritten digits. Samples of the dataset have a dimension of 28×28 pixels. The dataset was made by capturing original MNIST images with an ATIS event-based vision sensor. In this work, we move the center of the samples to the center of the input zone of our network. To ease the computational burden we only use 30 000 samples for training and 5000 for testing. For the comparison with the other methods described in Tabs. 4.5 and 4.6, we consider the full dataset, however.

DVS128 Gesture Dataset The DVS128 Gesture Dataset [7] was recorded with a DVS128 event-based vision sensor and is composed of samples of 29 individuals performing 11 hand gestures under 3 different scene illuminations. We move the center of the samples to the center of the input zone of our network and scale down the spatial extent by a factor of 1.94 to match the size of the PCL network’s input zone. The gestures of the first 23 individuals are used for training and the remaining are used for testing [7]. We use only the first 4 s of each gesture sample for both training and testing. For the comparison with the other methods described in Tabs. 4.5 and 4.6, we consider the entire duration of each gesture, however.

4.5.1.2 Classifier

We compare the influence of our learnt spike removal scheme of PCL and a random spike removal scheme by training classifiers on the resulting representations. We construct feature descriptors based on the spiking activity of all simple cells, all complex cells or both. Spike trains are binned (bin size varying between 4 s for gestures and 60 ms for N-MNIST) and we count the number of spikes in each bin for each cell. These values are combined into a vector X of size $B \times N$ where B is the number of time bins and N is the number of cells used for classification (either all simple cells, all complex cells, or both). We train separate linear SVM classifiers for the PCL network and the random spike removal scheme using this descriptor. For the comparison with other methods depicted in Tabs. 4.5 and 4.6, we use a fixed bin width of the duration of the entire input stimulus for both the N-MNIST and gestures datasets. Furthermore, to improve the results we also perform a min-max

normalization of the descriptor before training the SVM such that $X \leftarrow \frac{X - X_{min}}{X_{max} - X_{min}}$, with X_{min} and X_{max} , being the minimum and maximum spike count of X and the division being an element-wise division.

4.5.2 Results

By removing the most predictable and therefore most redundant spikes, PCL aims to save energy with only moderate loss of information. To characterize the network’s ability to trade off reduction in spiking activity (which we use as a proxy for energy savings) against information loss, as mentioned above, we test the PCL network on two visual classification problems, the DVS128 gesture benchmark [7] and the N-MNIST problem [182] (Fig. 4.12a,b). We compare the PCL network to a version where inhibitory connections have been disabled and instead the same number of spikes are removed randomly (Fig. 4.12c) to reveal if PCL retains more information for the same energy expenditure (number of spikes).

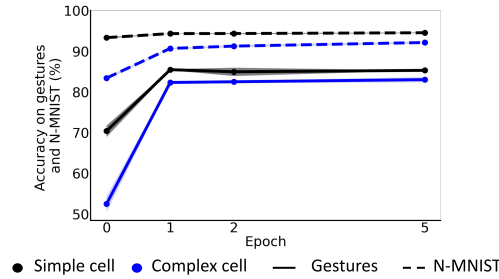


Figure 4.11: **Classification accuracy of simple and complex cells, without recurrent inhibition, on N-MNIST and gestures as a function of training time. Results show averages over 2 networks. Shaded regions denote one standard error across networks.**

We begin with the PCL network trained on natural images with cSTDP and fine-tune its distant lateral and top-down inhibition (again with cSTDP) on the two classification datasets. Then we train a classifier (linear support vector machine) on the (binned) spike trains of the PCL network’s simple (and/or complex) cells. By training on the natural images, the PCL network learns visual representations that improve classification accuracy for both simple and complex cells (Figure 4.11).

We systematically vary the amount of inhibition by scaling the fine-tuned inhibitory weights to reveal the PCL network’s trade-off between energy savings and information loss. Figure 4.12d confirms that more inhibition reduces spiking activity in the PCL network, down to a lower bound. Please, note that we use the amount of inhibition as the x-axis to showcase the fact that more inhibition indeed leads to a decrease in number of spikes. More importantly, the classification results reveal that the PCL network consistently outperforms the random spike removal scheme. For Gestures, performance remains high over the entire range of inhibition levels tested, while for N-MNIST, performance drops gradually with increasing levels of

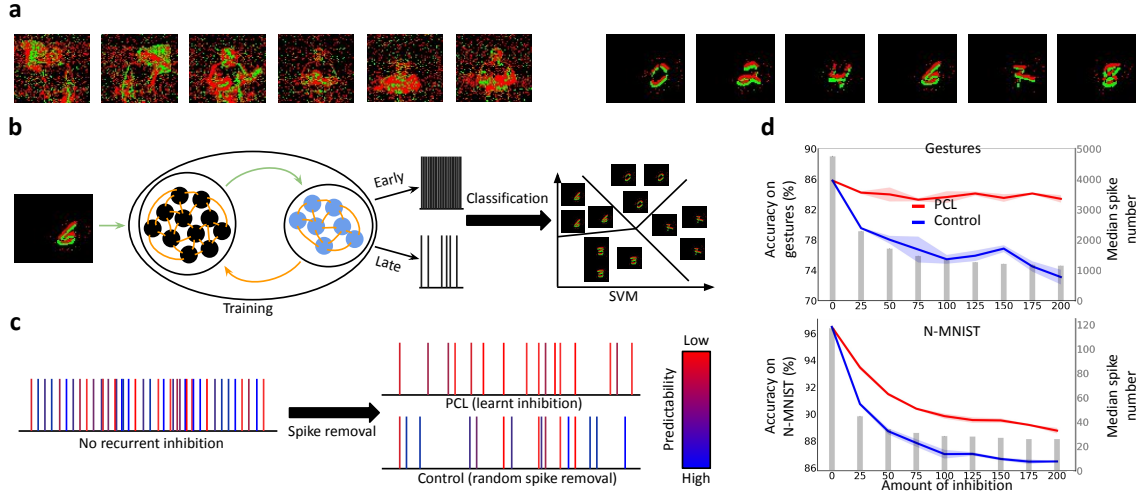


Figure 4.12: **PCL achieves substantial reduction of spiking with only small to moderate loss of downstream task performance.** **a**, Example of event-based inputs from the DVS128 Gesture dataset and the N-MNIST dataset. **b**, Experimental protocol. The inhibitory connectivity in the PCL network is fine-tuned on inputs from the respective datasets. Afterwards an SVM is trained to classify the visual input sequences from the spike patterns of the simple cells. **c**, Removal of most predictable spikes in the PCL network vs. random removal of the same number of spikes in the control condition. **d**, Effect of different amounts of inhibition on classification accuracy (left y-axis) for gesture (top) and N-MNIST datasets (bottom). 100% of inhibition corresponds to our default parameters. Grey bars show the median number of spikes (right y-axis). Results show averages over 3 networks with inhibition trained from different random seeds. Shaded regions denote one standard error across networks.

inhibition (Fig. 4.12d). We also consider classifiers trained on the activity of only the complex cells or the combination of simple and complex cells (Figure 4.13). These experiments confirm that for the N-MNIST dataset, higher levels of inhibition tend to gradually impair classification performance, whereas for gestures, substantial reductions in spiking activity are possible with only marginal loss of performance. They also reveal that classification based on the highly compressed complex cell representations is worse than that for simple cells, while classification based on the combined simple and complex cell activities can outperform classification based on simple cell activities alone.

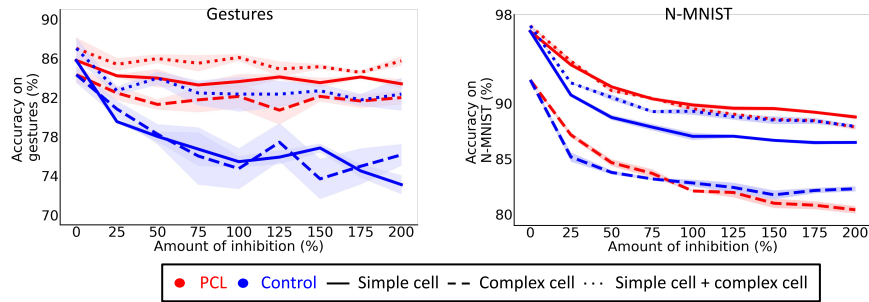


Figure 4.13: **Classification based on simple and complex cell activity.** Effect of different amounts of inhibition on classification accuracy for gestures (left) and N-MNIST datasets (middle and right) using the spiking patterns of the simple cells, complex cells and both of them conjointly. The experiment protocol is the same as the one depicted in Fig. 4.5b-c. 100% corresponds to our default parameters. Results show averages over 3 networks with inhibition trained from different random seeds. Shaded regions denote one standard error.

Table 4.3: Influence of different time bins on the classification accuracy of the PCL network on the N-MNIST dataset and on the DVS128 Gesture dataset. Results show average over 3 networks with one standard error across networks.

<i>N-MNIST dataset</i>				
	% of Inhib.	Bin size = 60 ms	Bin size = 150 ms	Bin size = 300 ms
Simple cells	0	96.48±0.08%	95.32±0.04%	95.25±0.15%
	100	89.84±0.18%	88.65±0.14%	87.84±0.16%
	200	88.76±0.2%	87.06±0.1%	87.43±0.27%
Complex cells	0	92.1±0.03%	90.05±0.03%	90.75±0.28%
	100	82.09±0.04%	79.38±0.29%	79.94±0.44%
	200	80.4±0.32%	77.58±0.56%	77.77±0.65%
Simple and complex cells	0	96.98±0.08%	95.43±0.01%	93.23±0.14%
	100	89.5±0.24%	87.51±0.24%	87.18±0.25%
	200	87.85±0.21%	85.77±0.08%	86.13±0.1%

<i>DVS128 Gesture dataset</i>				
	% of Inhib.	Bin size = 50 ms	Bin size = 2 s	Bin size = 4 s
Simple cells	0	68.69±0.85%	84.56±0.6%	85.79±0.1%
	100	68.19±0.78%	83.74±1.1%	83.62±0.77%
	200	69.94±0.38%	80.82±0.58%	83.39±0.48%
Complex cells	0	70.99±0.41%	84.91±0.76%	84.33±0.81%
	100	66.31±0.5%	81.41±0.72%	82.11±1.19%
	200	65.85±0.67%	81.52±0.53%	81.99±0.67%
Simple and complex cells	0	72.16±0.1%	86.67±1.01%	87.02±1.08%
	100	67.84±0.34%	85.85±0.42%	86.08±0.35%
	200	68.54±0.42%	83.86±1.09%	85.73±0.48%

We also test the effect of different bin widths when binning the spike trains on classification performance. For the N-MNIST dataset, we observe that performance does not critically depend on the chosen bin width. In contrast, for the gesture dataset performance is best for a coarse binning of the entire 4 s spike train into a single bin (Table 4.3).

To elucidate how the PCL network manages to retain high amounts of information about the input despite massively reducing the number of spikes, we analyze the firing statistics of units in the PCL network (Figure 4.14). We observe that the PCL network exhibits increased population sparseness. We measure population sparseness as the number of *active* neurons, i.e., neurons that fire at least one spike in response to a particular stimulus. A smaller number of active neurons corresponds to a higher population sparseness. We compare the number of active neurons for the PCL network and networks without recurrent inhibition and control networks with random spike removal (Figure 4.14 and Tab.4.4). Figure 4.14 (left-hand side) shows example spike trains of the different networks for three example gestures. It is apparent that the PCL network uses a smaller number of active neurons in each

case. This observation is confirmed quantitatively in Figure 4.14 (right-hand side), where we plot the number of active neurons in each condition. Table 4.4 confirms the statistical significance of this finding for all gestures with just one exception: For the “other gesture” class, which is composed of many different and unrelated gestures, the sparsity is not significantly increased by the PCL network. We suspect that this is due to a lack of systematic and learnable structure in this class. In conclusion, the PCL network provides an efficient code by reducing the number of neurons actively involved in the representation of any particular stimulus.

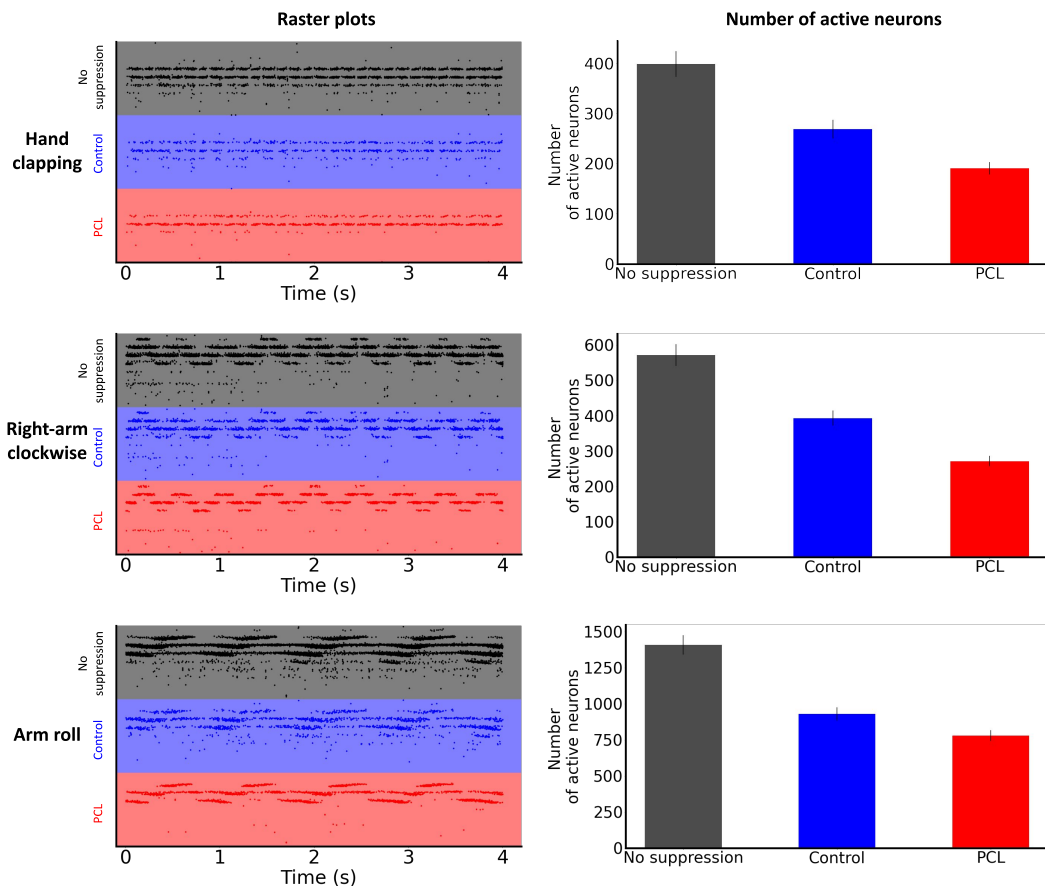


Figure 4.14: **PCL’s simple cells show increased population sparseness.** **a**, Raster plots of a network without distant lateral and top-down inhibition, a PCL network, and a control network in which spikes are removed randomly, for three different gestures. The right side shows the average number of active neurons (firing at least one spike) for 77 samples of each gesture. Error bars denote one standard error.

We compare the classification performance of the PCL network (without suppression) to that of other spiking neural networks that have been tested on the same or similar tasks. Table 4.5) compares different spiking neural network approaches that have been applied to digit recognition (using the N-MNIST, MNIST and MNIST-DVS datasets). Comparisons of methods tested on different datasets are generally difficult, but we have still included MNIST and MNIST-DVS in order to cover more approaches. Supervised methods generally tend to show the strongest performance (top part of table). Among the unsupervised approaches (bottom part of table),

Table 4.4: Statistical significance of PCL’s population sparseness for different gestures. Errors denote the standard error over 77 samples. P-values were calculated via the independent samples t-Test.

Gestures	Number of active neurons			P-value	
	PCL	Control	No suppression	Control - PCL	No suppression - PCL
Hand clapping	191 $\pm$ 13	269 $\pm$ 19	399 $\pm$ 25	< 0.01	< 0.01
Right hand wave	242 $\pm$ 14	371 $\pm$ 21	526 $\pm$ 30	< 0.01	< 0.01
Left hand wave	271 $\pm$ 15	393 $\pm$ 21	571 $\pm$ 30	< 0.01	< 0.01
Right arm clockwise	859 $\pm$ 155	1023 $\pm$ 48	1514 $\pm$ 71	< 0.05	< 0.01
Right arm counter clockwise	837 $\pm$ 43	1013 $\pm$ 51	1475 $\pm$ 71	< 0.01	< 0.01
Left arm clockwise	790 $\pm$ 39	948 $\pm$ 47	1446 $\pm$ 69	< 0.01	< 0.01
Left arm counter clockwise	779 $\pm$ 39	930 $\pm$ 45	1408 $\pm$ 67	< 0.05	< 0.01
Arm rolls	292 $\pm$ 10	418 $\pm$ 14	615 $\pm$ 21	< 0.01	< 0.01
Air drums	361 $\pm$ 21	470 $\pm$ 23	681 $\pm$ 34	< 0.01	< 0.01
Air guitar	240 $\pm$ 11	343 $\pm$ 17	534 $\pm$ 28	< 0.01	< 0.01
Other gestures	976 $\pm$ 69	1043 $\pm$ 68	1761 $\pm$ 101	0.49	< 0.01

those that rely on biologically implausible non-local learning rules show particularly strong performance. Among the models that rely only on local learning rules and that have been tested on N-MNIST, PCL performs best, even though its feature representation has only been trained on natural images.

Results of different approaches on the DVS128 Gesture dataset are compared in Table 4.6. Again, supervised approaches (top part of table) tend to show the strongest performance. Among unsupervised approaches based on local learning rules, only the model by Nadafian et al.[175] shows slightly better results than PCL (90.52 % accuracy vs. 89.12 %). Again, it is important to keep in mind that the PCL network’s feature representation was only trained on natural images and its architecture and parameters have not been optimized for any classification task.

It is also important to note that the PCL network performs better on both gestures and digits than a classifier trained directly on the events alone, which achieve 88.42% on digits and 69.82% on gestures.

Overall, our results indicate that the PCL network, by suppressing the most predictable spikes, establishes a favorable trade-off between energy consumption and expressiveness of the learned representation and permits strong downstream classification performance on two challenging benchmark problems.

Table 4.5: Classification accuracy of different spiking neural network models on digits datasets.

<i>Supervised feature learning</i>					
Paper	Dataset	Learning rule	Classifier head	Acc. (%)	Local update
Zhao et al. [261]	MNIST-DVS	Fixed weights for simple features and Tempotron [90] for classification	Clusters of output neurons	88.14	✗
Lee et al. [137]	N-MNIST	Backpropagation	Output layer	98.76	✗
Jin et al. [114]	N-MNIST	Backpropagation	Output layer	98.88	✗
Liu et al. [146]	N-MNIST	Fixed weights for simple features and STDP for classification	Clusters of output neurons	89.7	✗
Lan et al. [132]	N-MNIST	Predictive coding inspired backpropagation	Output layer	98.5	✗
Samadzadeh et al. [207]	N-MNIST	Backpropagation	Output layer	99.6	✗
<i>Unsupervised feature learning</i>					
Paper	Dataset	Learning rule	Classifier head	Acc. (%)	Local update
Diehl and Cook [58]	MNIST	STDP	Clusters of output neurons	95	✓
Kheradpisheh et al. [122]	MNIST	Fixed weights for simple features and STDP for intermediate features	SVM	98.4	✓
Tavanaei et al. [233]	MNIST	STDP	Output layer	98.6	✓
Putra and Shafique [191]	MNIST	STDP	Clusters of output neurons	92.8	✓
Zhang et al. [258]	MNIST	STDP and Self-backpropagation (SBP)[258]	Output layer	97.89	✗
Ororbia [183]	MNIST	Spike-Triggered LRA [183]	Maximum entropy log-linear classifier	97.53	✓
Lee et al. [138]	MNIST	Rate-based hebbian learning	Linear classifier	90~	✓
Iyer and Basu [108]	N-MNIST	STDP	Clusters of output neurons	80.63	✓
Safa et al. [205]	N-MNIST	Lasso regression with gradient descent paired with STDP	SVM	99.26	✗
PCL (this work)	N-MNIST	STDP	SVM	95.62 (S) 93.09 (C) 95.81 (S+C)	✓

Table 4.6: Classification accuracy of different spiking neural network models on the DVS128 Gesture dataset.

Supervised feature learning

Paper	Learning rule	Classifier head	Acc. (%)	Local update
Massa et al. [156]	ANN-SNN conversion	Output layer	89.64	✗
Singh et al. [221]	ANN-SNN conversion	Output layer	92.466	✗
Fang et al. [65]	Backpropagation	Output layer	97.92	✗
Cordone et al. [45]	Backpropagation	Output layer	87.5	✗
Zhang et al. [258]	STDP and Self-backpropagation (SBP)[258]	Output layer	84.76	✗
Samadzadeh et al. [207]	Backpropagation	Output layer	96.7	✗
Yao et al. [256]	Backpropagation	Output layer	96.53	✗
Samadzadeh et al. [207]	ANN-SNN conversion	Output layer	96.7	✗

Unsupervised feature learning

Paper	Learning rule	Classifier head	Acc. (%)	Local update
George et al. [83]	STDP	Linear classifier	65	✓
Iyer and Chua [109]	STDP for feature learning and Tempotron[90] for classification	Output layer	60.37	✓
Safa et al. [205]	Lasso regression with gradient descent paired with STDP	SVM	92.5	✗
Nadafian et al. [175]	STDP and unsupervised delay learning [174] for feature learning, R-STDP [72] and reward-modulated delay learning [175] for classification	Output layer	90.52	✓
PCL (this work)	STDP	SVM	81.75 (S) 87.37 (C) 89.12 (S+C)	✓

4.6 Discussion

We have presented Predictive Coding Light (PCL), a hierarchical spiking neural network that not only models many features of V1 neurons, but also offers an account for surround suppression, orientation-tuned suppression and cross-orientation suppression. Our study shows that via STDP-learning, the PCL network encodes with few spikes non-trivial visual stimuli such as digits and gestures. Interestingly, we observed that removing spikes at random worsens the neural representation and leads to a poor performance. This is, to our knowledge, the first demonstration of the advantages of such an adaptive suppression in neural coding within a fully spiking system.

Although the PCL network reproduces many findings about primary visual cortex (at least at a qualitative level), it is not meant to be a faithful model of the underlying neurobiology. Some shortcomings in this respect are the violation of Dale’s law by units in the PCL network, i.e., they excite some of their targets and inhibit others. Other shortcomings are the absence of recurrent excitation and feedforward inhibition. Nevertheless, the PCL network offers an arguably minimal description of how a wide range of visual cortex phenomena can arise through generic spike timing-based learning rules. It remains to be seen if other features of information processing in primary visual cortex can also be demonstrated in PCL networks.

The PCL network contrasts with the generic formulation of predictive coding. In standard predictive coding, neurons transmit only a prediction-error response, which has made popular the idea of a possible class of specific error neurons [120], [227]. While this idea appears attractive at first glance, the PCL network here shows that such explicit prediction error neurons might not be required to compute predictive (and efficient) codes. In the PCL network, rather than representing an error explicitly with explicit neurons, the sensory neurons simply respond preferentially to specific inputs, and their level of activity changes according to how redundant all neuronal spikes are. In this configuration, the activity of the network might be implicitly encoding some sort of prediction-error response as we observe that novel/unpredictable stimuli cause high activity (due to weak inhibition), while redundant/predictable stimuli cause low activity (due to strong inhibition).

Interestingly, other models of predictive coding with spikes such as Spike Coding Networks (SCNs) [32], [33] encode errors in the membrane potential of neurons, however, they use unstructured networks in which representations arise from the optimization for specific processes and tasks rather than from direct exposure to stimuli and STDP-learning. Essentially, since the encoding of the error in the PCL network is implicit, there is no indubitable proof that an error does not arise in the membrane potential of the neurons, and both type of networks show similarities in

the reproduction of biological response properties. As SCNs are based on similar theoretical assumptions as the PCL network, an interesting problematic would therefore be to ascertain explicitly the resulting encoding differences between the two of them. It is an open question whether or not for a particular application, the PCL network and SCNs would not be solving the same optimization problem and achieving similar solutions. Similarly, the relationship between the STDP-learning happening in a PCL network and the free-energy principle [75] is also an open question. Fundamentally, the results of the PCL network and the noticeable similarities and differences with other predictive coding approaches suggest that spiking neurons could encode prediction errors in many seemingly different, but perhaps functionally equivalent ways.

Importantly, we also began work on the implementation of a light version of a PCL network on neuromorphic hardware platforms such as the Loihi 2 chip [181]. However, we experienced a diverse number of software and hardware limitations (absence of complex mathematical operations, fixed-point computation, low memory resources etc.) which left us to postpone this application. During the CapoCaccia Neuromorphic Workshop, we also tried out the SpiNNaker2 board [158] which showed some level of software maturity but still lacked modularity when adding new features. Essentially, PCL has undeniably desirable properties for hardware implementation but the current state of neuromorphic hardware and the associated software tools does not allow the necessary coding freedom.

Ultimately, however, the reliance on generic local learning rules makes the PCL network an interesting candidate architecture for implementation on neuromorphic hardware for energy efficient information processing and continual learning. The violation of Dale’s law by the PCL network may even be seen as a strength in this context, as it obviates the duplication of precious resources (separate excitatory and inhibitory units with identical tuning that might be required otherwise). An important open challenge is to extend the approach to deep architectures with many layers. This could pave the way for many interesting real world applications. Furthermore, the incorporation of recurrent excitation could instill additional desirable computational features such as a fading memory, pattern completion, evidence accumulation and others. However, maintaining healthy network dynamics when incorporating recurrent excitatory connections that learn with local learning rules remains a formidable challenge.

Conclusion

In this chapter, we presented our brain-inspired event-based framework for visual perception: Predictive Coding Light (PCL). PCL lies at the intersection of computational neuroscience, cognitive science and computer vision. Crucially, this work builds upon the vast theoretical background of efficient coding and predictive coding to apply it on non-trivial spiking input data for both neuroscientific applications and downstream classification tasks. We believe that such a framework that spans multiple disciplines could allow the development of novel ultra-low energy approaches for sensory perception and control in robotics.

In the next two chapters, we add and test new functionalities of the PCL network by learning invariant hierarchical representations with deeper networks, and integrating recurrent excitatory connections.

DEEP PREDICTIVE CODING LIGHT: INVARIANT OBJECT REPRESENTATIONS

Overview	90
5.1 Introduction	90
5.2 How to achieve spatial invariance with the PCL network	91
5.2.1 Dataset generation and network architecture	92
5.2.2 The PCL network requires specific neuronal projections to learn invariant object representations	95
5.2.3 Residual connections improve classification accuracy when used in intermediate layers	100
5.3 Discussion	103
5.4 Conclusion	103

Overview

In this chapter, we test different network architectures for deeper instances of the PCL network and demonstrate how to build invariant representations with such networks of spiking neurons. Specifically the building of invariant representations that led to this study rests on the following idea: if the PCL network is excited successively with the same input at different locations, similarly to what would arise through saccades, our intuition would be that the temporal proximity between two positions of the same input should allow the STDP rule to associate them and learn invariant representations hierarchically, as a neuron’s receptive field size increases.

We hope to submit this work in a peer-reviewed journal in the future. The contributions of this work are as follows:

- We show that one-to-one feature projections between layers allow the PCL network to achieve position invariant representations.
- We show that residual connections play a role in the learning of hierarchical representations in deep PCL networks.

This chapter is organized as follows: we will first introduce the motivations behind this study. Next, we will demonstrate how spatial invariance can be achieved with the PCL network by presenting our datasets and different network architectures and then showcasing the results achieved by PCL.

5.1 Introduction

In the previous chapter, we presented the PCL network and showed how it learnt simple and complex cell-like response properties. As we studied specifically V1, PCL consisted only of a 2-layer network. However, as introduced in Chapter 2, visual processing does not stop at V1 but rather continues in two different processing streams, the ventral and dorsal stream thought to be responsible for object and motion processing, respectively. Considering how the PCL network could learn V1-like features, deeper versions of the PCL network might be able to build representations similar to either the IT cortex or area MT. Following this intuition, we here study deeper versions of the PCL network for the learning of visual features.

While there have been different deep models of spiking neural networks proposed in the literature, many approaches first train standard non-spiking artificial neural

networks and only later convert them to spiking neural networks, rather than directly training a spiking network [124]. Other works do training using surrogate gradients [179]. Both types of methods thus learn visual features with non-local learning rules, which is biologically implausible. A third class of hierarchical spiking neural network models includes those that instead use local learning rules (e.g. STDP), such as works that use “fully-connected”-like networks instead of convolutional-like networks inspired from the cortex that perform a retinotopical mapping of the input [58], [191]. Other approaches, notably those from Masquelier’s group [121], [122], [155] have proposed diverse implementations of hierarchical cortex-like spiking neural networks. However, in those works, not all connections are learnt with STDP (e.g. pooling is performed with Integrate-and-Fire neurons with a fixed excitation, as well as fixed lateral connections). A few other recent works have proposed hierarchical cortex-like spiking neural networks [87], [175], but they rather aim at performing end-to-end classification using reward-modulated spike timing-dependent plasticity (R-STDP). Interestingly, other works are rooted in predictive coding such as [138], [184], but they do not consider the retinotopical mapping of the visual cortex. Most importantly, those models do not evaluate per se the invariance of the learnt features as they use inputs that appear in single spatial regions, rather than in different ones. Here, we construct hierarchical instances of the Predictive Coding Light network, which learn from inputs that can appear in very different spatial regions and evaluate the ability of different layers of the network to classify this data. This allows us to uncover a variety of architectural choices that allow a cortex-like hierarchical network to exhibit spatial invariance similar to IT cortex.

To test our network, we use an MNIST-like dataset that we convert to events using an event-based simulator. We next present the results obtained with different deeper instances of the PCL network, and showcase architectural changes (a precise one-to-one cortical mapping, residual/bypass connections) that help to improve the “performance” of the network, which we evaluate through a classification task.

5.2 How to achieve spatial invariance with the PCL network

As mentioned above, we are motivated to learn invariant object representations with the PCL network. The IT cortex is known to be populated by neurons that respond to intermediate and complex visual features regardless of their locations in the neurons’ receptive fields. So, we wondered if by adding more layers to the PCL network, it would be able to achieve similar invariant response patterns.

To test the invariance learning properties of PCL, we use simple inputs that consist of digits. We consider again the well-known MNIST dataset, but we convert the standard MNIST frames to events that randomly appear at different locations of our network’s input zone. In this configuration, cells in superficial layers see only patches of the digits, while cells in deeper layers see entire digits that can appear at random locations within the input region.

In the visual cortex of mammals, different projections (e.g. with so-called residual or skip connections) are seen between different cortical areas (e.g. V1 to V4). Additionally, different studies report feature-specific projections from V4 to IT cortex [231]. It is so far unknown whether such connectivity structures play a major role in how the visual system achieves invariant object representations. Inspired by such findings, we here also test different connectivity patterns in our PCL network and assess their impact on invariance learning. We detail below how we generate the input stimuli as well as the different network architectures. Following that, we present the results achieved by PCL.

5.2.1 Dataset generation and network architecture

In this chapter, contrary to previous Chapter 4 and the following chapter (Chapter 6), we convert frames to events not with the PIX2NVS simulator [28] but with a more recent event-based simulator named v2e [104]. We detail in the following the functioning of the simulator and how we use it to generate our dataset that we name cMNIST-ev for contrastive MNIST-events.

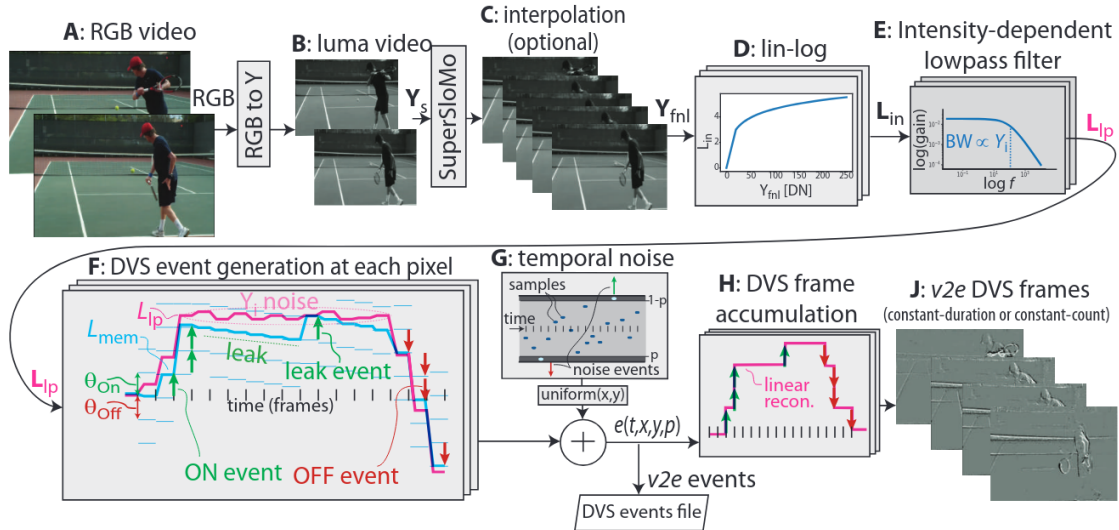


Figure 5.1: Modules of the v2e simulator from [104].

The v2e event-based simulator. v2e was proposed by Delbruck’s group [104] to correct various false assumptions on event cameras and thus have a realistic simulator of event-based visual data. The simulator emulates outputs from the DVS (Dynamic Vision Sensor) using 6 modules:

1. RGB to grayscale conversion: Frames of an RGB video are all first converted to grayscale/luma values. In this step, frames can optionally be resized to other output height and width.
2. Synthetic slow-motion: Luma frames are interpolated using the Super-SloMo video interpolation network [113]. This permits to increase the temporal resolution of the video and generate fine-grained events similar to the real sensor.
3. Logarithmic mapping: v2e bases its conversion of linear pixel intensities to the logarithmic ones used in event-based sensors on [118]. Essentially, linear luma frames are converted to logarithmic values using a lin-log mapping in which small luma intensities (smaller than 20) are mapped linearly to a logarithmic intensity, whereas higher luma intensities (above 20) follow a logarithmic function as depicted in Fig. 5.1.
4. Finite intensity-dependent photoreceptor bandwidth: v2e adds an (optional) low-pass filter that filters the pixel logarithmic intensity. This low-pass filter is used to model responses of event cameras under low illumination, as in an event camera the pixel bandwidth increases with the pixel value. For pixels of low intensities, an event camera exhibits a phenomenon akin to “motion blur” where the pixel takes longer time to settle to a new value and thus emits a trail of events at moving edges.
5. Event generation model: Each pixel keeps a memory of its previous logarithmic intensity value. According to the difference between this memorized logarithmic intensity and the new intensity, a number N_e of “ON” or “OFF” events are generated if this difference is above or below a threshold. Thresholds are initialized from a gaussian distribution similar to the real sensor. In addition to this standard event generation process, “hot pixels” and “leak noise” events are also applied to the generation model to mimic pixels that fire events even in the absence of movement in the scene, and spontaneous leak events caused by the characteristics of the sensor.
6. Temporal noise: v2e uses a Poisson process to model the temporal noise normally observed in a real event camera.

Figure 5.1 gives an overview of the different modules of the v2e simulator. Please, see [104] for more information on the simulator.

The cMNIST-ev dataset. Using the above v2e simulator, we convert MNIST frames to events. The process of dataset generation is the following:

1. Creation of sequences of digits: We take an MNIST frame (28×28 *pixels*) and randomly initialize its location in a zone of dimensions 66×66 *pixels*. We then change the pixel intensities of the digits such that digits' luminance switch from black (0) to white (255), following a sinusoidal signal. Once half a period (50 ms) has passed, we randomly change the location of the digit, still in the same zone of 66×66 *pixels*. This way, events of different polarities can be generated, and digits can be located in different regions of the network's input zone. We use a frame rate of 1000 images/s.
2. Conversion: The digits are converted using the v2e simulator with the "ideal" parameter settings provided by the simulator's software. This "ideal" setting considers an idealized version of an event camera. This is equivalent to considering only steps 1, 2 and 5 of the v2e simulator. However, considering that we use a very high frame rate, we do not interpolate our sequences and the generation process thus only consists of steps 1 and 5 of the v2e simulator. Please note that in this "idealized" setting, hot pixels and leak noise are also removed from the event generation model.

Please see Fig. 5.3 (left) for an illustration of those digits. We generate a total of 1800 event-based sequences (each lasting 1.6 s) that we use to train our PCL network from scratch. To evaluate the ability of the PCL network, we however generate 9000 additional event-based sequences (900 samples/class) that we use to train and test an SVM classifier (detailed in the next paragraph). Contrary to the sensory learning set, they instead last 300 ms. We use 80 % of the samples for training and the remaining 20 % for testing.

Network architecture and parameters. As stated before, different projection types are observed in the biological vision system. It is currently unknown which role they may play in the learning of invariant representations. So, as we here are building deeper versions of the initial Predictive Coding Light network, we decided to test different of those projection types to establish their role. We first simply build a 3-layer network (one more layer than the standard PCL network). The different tested network architectures are depicted in Fig. 5.2. In this 3-layer network, L1 and L2 follow the exact same projection type as in the standard PCL network. However, we test three types of projection patterns on L3:

- (a) a PCL network which uses residual connections that directly connect L1 to L3. L3 thus receives two feedforward inputs, the direct input from L2, and the residual input from L1. L3 neurons can inhibit each other through local lateral inhibition and also send top-down inhibition to both L1 and L2.

- (b) a PCL network with a basic projection type where L3 receives feedforward input only from L2. L3 neurons can also inhibit each other through local lateral inhibition and also send top-down inhibition to L2.
- (c) a PCL network with a one-to-one projection type where each *slice* of features (akin to cortical columns) of L2 is projected **only** to the same feature slice in L3. This feature preserving projection type is motivated by studies of inferior temporal areas TE and TEO (found in the IT cortex) which report a precise organization of projections of cortical columns from V4 to area TE. In this configuration, we thus allow L3 neurons to send top-down inhibition to their input feature slices, but we **do not** allow local lateral inhibition among L3 neurons as they now share different input receptive fields in the feature space.

The parameters of those different versions of the PCL network can be found in Tab. 5.1. We do not use our distant lateral inhibition here in any of the layers as an effort to reduce computational resources. Please note that apart from the deactivation of the distant lateral inhibition, all networks depicted here are still learnt using a cSTDP rule as well and use all the plasticity mechanisms described in Chapter 4. The logic behind the choice of the parameters also follow the one described in that same chapter.

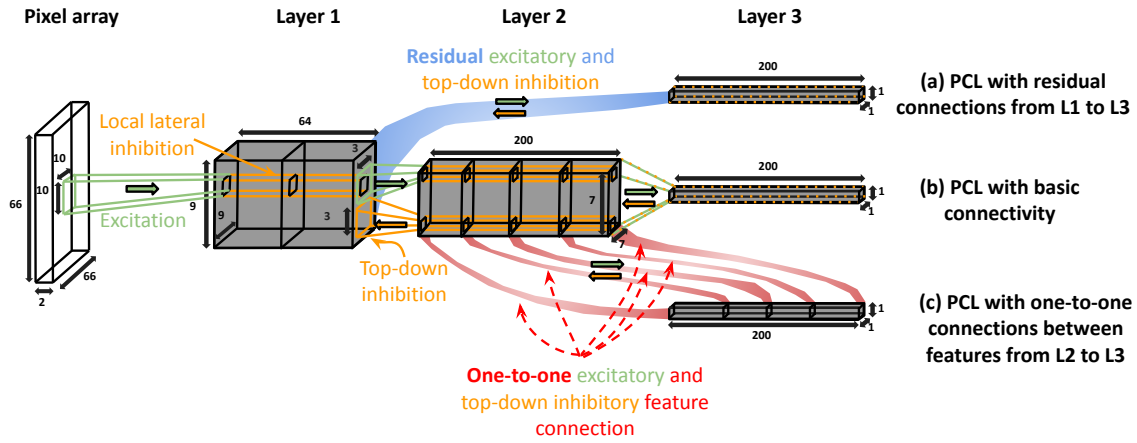


Figure 5.2: Detailed network architecture of different 3-layer PCL networks.

5.2.2 The PCL network requires specific neuronal projections to learn invariant object representations

We train our three PCL networks from scratch, using a cSTDP rule for all synaptic connections (Figure 5.2) on the 1800 elements of the sensory learning database of cMNIST-ev for 5 epochs.

After training, L1 and L2 respectively exhibit selectivity for features of simple to intermediate complexity (Fig. 5.3 bottom). Indeed, L1 neurons appear to be

Table 5.1: Connectivity, neurons' membrane properties and plasticity parameters of a 3-layer PCL network. Connectivity parameter values indicate dimensions of the following form: $(x \times y \times z)$ cells.

L1 connectivity parameters:

Retina size	Input zone	N_{cells}
$(346 \times 260 \times 2)$	$(66 \times 66 \times 2)$	$(9 \times 9 \times 64)$
receptive field	Overlap	Plastic lateral inhibition range
$(10 \times 10 \times 2)$	$(3 \times 3 \times 2)$	$(0 \times 0 \times 64)$

L2 connectivity parameters:

Input size	Input zone	N_{cells}
$(9 \times 9 \times 64)$	$(9 \times 9 \times 64)$	$(7 \times 7 \times 200)$
receptive field	Overlap	Plastic top-down inhibition range
$(3 \times 3 \times 64)$	$(2 \times 2 \times 64)$	$(3 \times 3 \times 64)$

L3 connectivity parameters:

Input size	Input zone	N_{cells}
$(7 \times 7 \times 200)$	$(7 \times 7 \times 200)$	$(1 \times 1 \times 200)$
receptive field	Overlap	Plastic top-down inhibition range
$(7 \times 7 \times 200)$	$(0 \times 0 \times 200)$	case (a)/(b): $(7 \times 7 \times 200)$ case(c): $7 \times 7 \times 1$

Layers' membrane potential and plasticity parameters:

V_{reset} (mV)	V_{thresh} (mV)	$\tau_{\text{m/I}}$ (ms)	τ_{RP} (ms)	τ_{LTP} (ms)	τ_{LTD} (ms)
-10	L1: 30 L2: 3 L3: 3	L1: 20 L2: 90 L3: 110	5	L1: 10 L2: 80 L3: 100	L1: 10 L2: 80 L3: 100
η_{RP} (mV)	w_{min} (mV)	w_{max} (mV)	$\eta_{\text{LTP/LTD}}$ (mV)	$\eta_{+/-}$	λ
10	0	excit: $V_{\text{thresh}} + 1$ loc. inhib.: 50 td. inhib.: 50	L1 excit: ± 0.00054 L2 excit: ± 0.5184 L3 excit: ± 0.575 loc. inhib. (all): ± 3.1 td. inhib. (all): ± 3.1	$\frac{1}{w_{\text{max}}}$	L1-L2-L3 res. excit.: $0.25 \times RF_{\text{size}}$ L3 excit.: $3 \times RF_{\text{size}}$ loc. inhib. (all): $25 \times N_{\text{features}}$ td. inhib.: $5 \times TD_{\text{range}}$

tuned to sharp bars located at different locations in the receptive fields. However, L2 neurons here appear to exhibit tuning properties to different shapes that are usually observed in digits.

L3 neurons' receptive fields give different results according to the projection type considered (Fig. 5.3 right). L3 neurons' receptive fields from case (a) and (b) do not appear to exhibit any specific structure, with neurons being tuned to different unstructured shapes throughout their receptive fields. We however note that certain regions of the receptive fields appear to display the same feature repetitively (a bit more in case (a) than case (b)). Still, the learnt receptive fields of both cases (a) and (b) are not reminiscent of the underlying patterns seen in digits.

For case (c), the specific one-to-one feature slice projection from L2 to L3 allow neurons to be tuned for a unique feature throughout their input zone. The features displayed by their receptive fields resemble shapes that would be seen in different digits (e.g. in digit "0" or digit "3").

Importantly, rather than becoming selective for one given location in their receptive fields, all cases exhibit broad synaptic connectivity patterns everywhere in the neurons' receptive fields, which is in line with the random digit locations of the cMNIST-ev dataset.

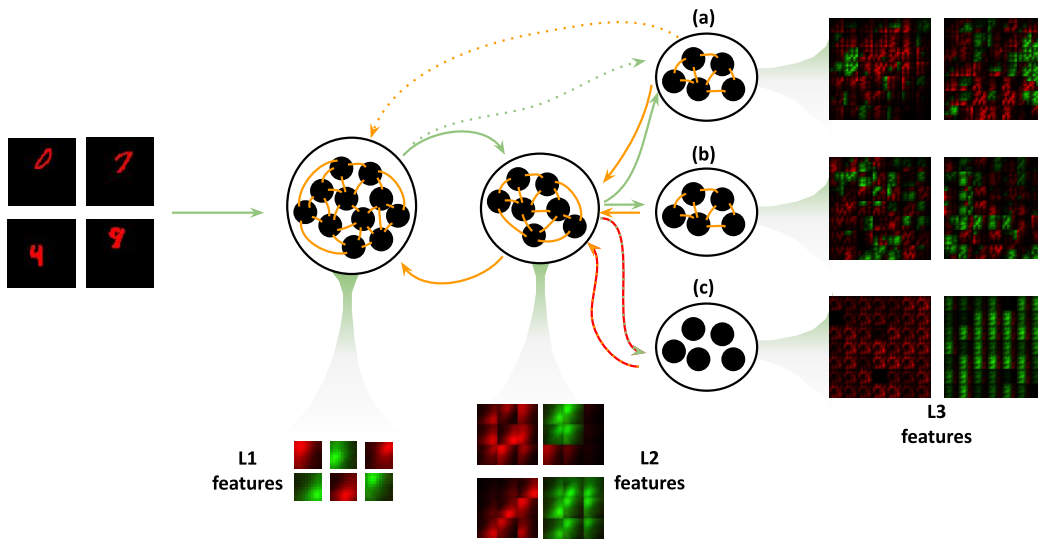


Figure 5.3: Schematic illustrating different 3-layer PCL networks as well as the learnt sensory representations. L2 and L3 receptive fields are built by showing for each retinal location the input cell's receptive field with the strongest connection to the subsequent cell.

It should also be noted that as a digit size is 28×28 *pixels*, L1 receives only inputs from different parts of the digits as its receptive field size is 10×10 *pixels*. L2 feeds from a region of 3×3 L1 *neurons*, which when converted in pixels is equal to 24×24 *pixels*, due to the overlap between L1 neurons. Finally, L3 feeds from the entire input zone. L3 is thus given the arduous task of separating (with no

supervision) the class of the digit regardless of its location in the input zone. This might explain why L3 neurons’ receptive fields of cases (a) and (b) do not exhibit any visually observable structure, contrary to the receptive fields of case (c), which “only” have to learn how to become tuned to various input locations, rather than further separating the features of different digits.

Next, we used the different trained PCL networks to classify the digits. As in Chapter 4, we froze the trained weights and used the spiking patterns of the different layers of the network to train an SVM classifier (Figure 5.4). We here perform classification with the top-down inhibition deactivated, as we observed in Chapter 4 that it tended to reduce classification accuracy. We later discuss in Chapter 7 how top-down inhibition might be used during inference without causing information loss. The idea here would be that: the deeper a layer is, the higher its classification performance should be. As superficial layers see only patches of the input, they should not be able to classify digits as well as layers whose neurons “see” the entire input. L3 neurons in particular should accumulate evidence on a digit class over different spatial locations to reach the highest accuracy (which would be expected of IT cortex neurons).

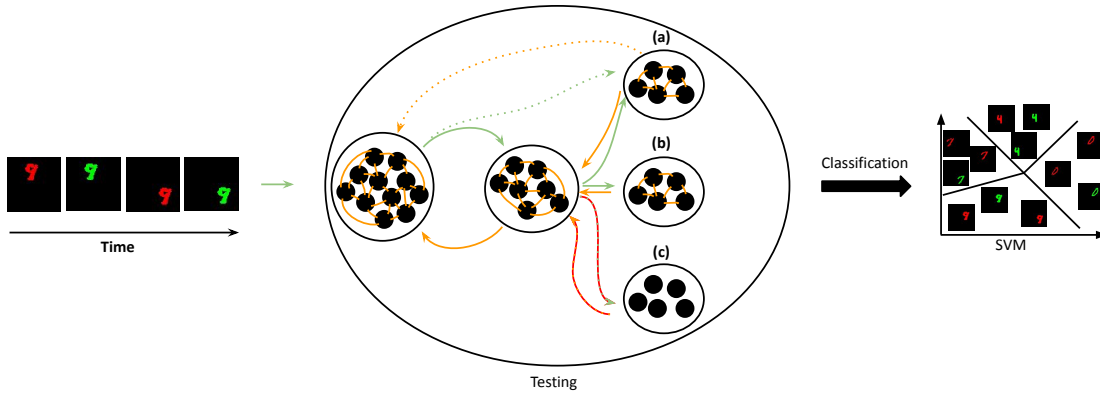


Figure 5.4: Experimental protocol for the classification of digits with different 3-layer PCL networks

We describe again briefly below how the classification operates.

Classification Spike trains are accumulated over the entire input sequence and we count the number of spikes for each cell. These values are combined into a vector X of size N where N is the number of cells used for classification (either L1, L2, L3 cells or all of them together). Furthermore, to improve the results we also perform a min-max normalization of the descriptor before training the SVM such that $X \leftarrow \frac{X - X_{min}}{X_{max} - X_{min}}$, with X_{min} and X_{max} , being the minimum and maximum spike count of X and the division being an element-wise division. We train linear SVM classifiers for the PCL network with our cMNIST-ev dataset (7000 samples) using this descriptor. The testing database consists of 1800 samples.

Table 5.2: Comparison of different 3-layer network architectures on the emergence of invariant representations in a deep PCL network with the cMNIST-ev dataset. Results show averages on 2 network simulations that were each trained from scratch using random initialization seeds. Errors on the other hand denote the standard error.

	Case (a)	Case (b)	Case (c)	Control of case (c)	Control L1/L2
Median number of spikes	L1: 591 ± 2	L1: 587 ± 3	L1: 591 ± 2		
	L2: 201 ± 1	L2: 204 ± 0	L2: 196 ± 0	L3: 73 ± 0	L1: 539 ± 1
	L3: 59 ± 0	L3: 14 ± 0	L3: 77 ± 1	All: 861 ± 0	L2: 219 ± 0
	All: 851 ± 1	All: 805 ± 4	All: 866 ± 1		
Accuracy	L1: $31.15 \pm 0.14\%$	L1: $29.81 \pm 0.22\%$	L1: $31.28 \pm 1.26\%$		
	L2: $44.33 \pm 0\%$	L2: $44.23 \pm 1.3\%$	L2: $44.42 \pm 0.2\%$	L3: $50.55 \pm 0.12\%$	L1: $21.61 \pm 0.08\%$
	L3: $19.36 \pm 0.06\%$	L3: $13.15 \pm 0.14\%$	L3: $53.19 \pm 0.53\%$	All: $51.12 \pm 0.43\%$	L2: $13.2 \pm 0.3\%$
	All: 48.83 ± 0.79	All: $48.17 \pm 0.24\%$	All: $52.45 \pm 0.1\%$		

The results are reported in Tab. 5.2. First, from one layer to the next, we observe a drop in the number of spikes. Also note due to the architecture, the number of neurons also diminishes per layer. The representations learnt by the network thus have to become more compact with each layer.

Second, as expected, the classification increases from L1 to L2. However when classifying the network with L3, depending on the projection type considered, we either observe a decrease in performance (case (a) and (b)) or an increase in performance (case (c)). The use of a basic projection type in case (b) leads to a huge drop in classification accuracy from L2 to L3, which is consistent with the lack of structure observed on the representations of the receptive fields of this network’s L3. When residual connections from L1 to L3 are added (case (a)), the classification performance (as well as the number of spikes) weakly improves compared to case (b) but still stays very much below the accuracy obtained with L2 cells’ spiking patterns.

Finally, using a one-to-one feature projection between L2 to L3 (case (c)) gives the best results by a large margin with results that show a slight increase from L2 to L3.

Using all layers to classify the digits gives the best classification results for cases (a) and (b) but not case (c), interestingly.

We also verified that the learnt spatial connectivity in case (c) plays a role in spatial invariance by comparing the network’s accuracy with a control network (Control of case (c)) in which the weights from L2 to L3 are initialized randomly. This control network, while giving higher accuracy than cases (a) and (b) stays below our learnt network of case (c). We also consider a second control case (Control L1/L2) where we keep random weights for L1 and L2 input synaptic weights. We show with that second control (Control L1/L2) that learning the connections of L1 and L2 plays a major role in classification accuracy (see last column of Tab. 5.2).

Also, while we do not use the top-down inhibition during inference, we notice that using it during learning helps to learn visual representations that perform better (see Appendix).

Importantly, the structure of this dataset as well as the number of samples are vastly different from the N-MNIST dataset used in the previous chapter. Additionally, the networks' architectures are also different, which make any sort of direct comparison with our previous chapters' results complex. However, considering the high classification performance already reported on N-MNIST at weight initialization, we believe that the networks used here would perform similarly well to the PCL network of the previous chapter on the N-MNIST problem.

Overall, those results surprisingly suggest that STDP alone might not be sufficient to learn invariant representations, but that a one-to-one feature mapping from a cortical area to the next (here, from L2 to L3) may be required. It is however concerning that the residual connections, despite their presence in the visual system, do not seem to significantly increase the performance of L3. For this reason, we wondered if the residual connections' impact did not depend on the architecture type. In the brain, the pathway follows V1-V2-V4-IT, with the existence of some residual connections from V1 to V4 [177]. While the receptive field's size of V4 neurons is larger than V1 and V2's, it isn't as gigantic as the one of IT where neurons can significantly vary in position. For this reason, we next evaluate a 4-layer network which models the four areas, and we surprisingly observe a significant impact on classification accuracy when residual connections are used.

5.2.3 Residual connections improve classification accuracy when used in intermediate layers

We now use a 4-layer PCL network in which contrary to Chapter 4 where we modeled specifically V1's processing with simple and complex cell layers, we here, for the sake of simplicity, consider that each layer of the PCL network represents one visual area. We use the 4-layer PCL network to model the V1-V2-V4-IT pathway of the visual cortex of mammals. Following the previous section, to model the projection pattern of V4 to the IT cortex, we again use the one-to-one feature projection between cells of the two layers. We train two networks of this architecture where one contains the residual connections from V1 to V2 and one does not contain any residual connection. Both networks are trained for 5 epochs on the cMNIST-ev database using again a cSTDP rule for all synaptic connections. Figure 5.5 shows the network's architecture (with residual connections). The network's parameters can be found in Tab. 5.4. Importantly, in this new network's architecture, the input size of the different layers is as follows: L1 feeds from a zone of 6×6 *pixels*, L2 from a zone of 15×15 *pixels*, L3 from a zone of 21×21 *pixels* and L4 from the entire input zone of the network. In this configuration, the receptive field sizes of L2 and L3 are not vastly different and both should be excited by almost entire digits.

Table 5.3: Connectivity, neurons' membrane potential and plasticity parameters of a 4-layer PCL network. Connectivity parameters values indicate dimensions of the following form: $(x \times y \times z)$ cells.

L1 connectivity parameters:

Retina size	Input zone	N _{cells}
$(346 \times 260 \times 2)$	$(66 \times 66 \times 2)$	$(21 \times 21 \times 64)$
receptive field	Overlap	Plastic lateral inhibition range
$(6 \times 6 \times 2)$	$(3 \times 3 \times 2)$	$(0 \times 0 \times 64)$

L2 connectivity parameters:

Input size	Input zone	N _{cells}
$(21 \times 21 \times 64)$	$(21 \times 21 \times 64)$	$(18 \times 18 \times 64)$
receptive field	Overlap	Plastic top-down inhibition range
$(4 \times 4 \times 64)$	$(3 \times 3 \times 64)$	$(4 \times 4 \times 64)$

L3 connectivity parameters:

Input size	Input zone	N _{cells}
$(18 \times 18 \times 64)$	$(18 \times 18 \times 64)$	$(8 \times 8 \times 200)$
receptive field	Overlap	Plastic top-down inhibition range
$(4 \times 4 \times 64)$	$(2 \times 2 \times 64)$	$(4 \times 4 \times 64)$

L4 connectivity parameters:

Input size	Input zone	N _{cells}
$(8 \times 8 \times 200)$	$(8 \times 8 \times 200)$	$(1 \times 1 \times 200)$
receptive field	Overlap	Plastic top-down inhibition range
$(8 \times 8 \times 200)$	$(0 \times 0 \times 200)$	$(8 \times 8 \times 1)$

Layers' membrane potential and plasticity parameters:

V _{reset} (mV)	V _{thresh} (mV)	$\tau_{m/I}$ (ms)	τ_{RP} (ms)	τ_{LTP} (ms)	τ_{LTD} (ms)
-10	L1: 10 L2-4: 3	L1: 20 L2: 50 L3: 90 L4: 110	5	L1: 10 L2: 40 L3: 80 L4: 100	L1: 10 L2: 40 L3: 80 L4: 100
η_{RP} (mV)	w _{min} (mV)	w _{max} (mV)	$\eta_{LTP/LTD}$ (mV)	$\eta_{+/-}$	λ
10	0	excit: $V_{thresh} + 1$ loc. inhib.: 50 td. inhib.: 50	L1 excit: ± 0.00054 L2-L3 excit: ± 0.5184 L4 excit: ± 0.575 loc. inhib. (all): ± 3.1 td. inhib. (all): ± 3.1	$\frac{1}{w_{max}}$	L1-L2-L3-L3 res. excit.: $0.25 \times RF_{size}$ L4 excit.: $3 \times RF_{size}$ loc. inhib. (all): $25 \times N_{features}$ td. inhib.: $5 \times TD_{range}$

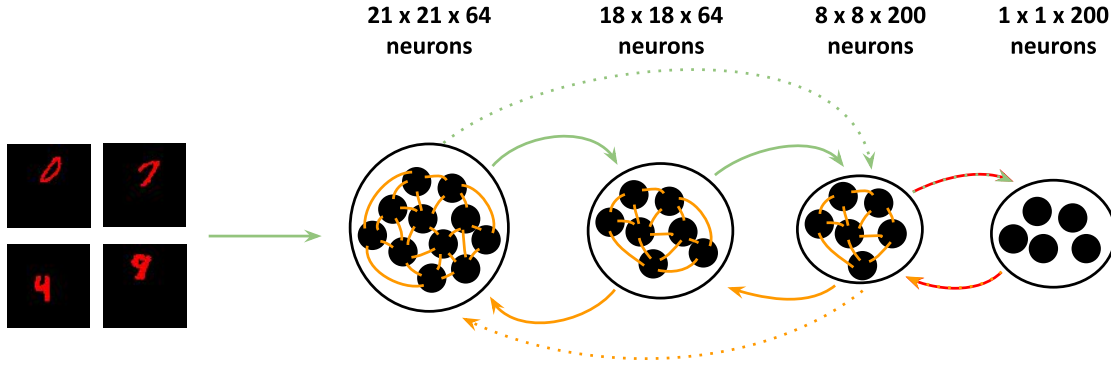


Figure 5.5: Schematic illustrating a 4-layer PCL network.

Table 5.4: Comparison of different 4-layer network architectures on the emergence of invariant representations in a deep PCL network with the cMNIST-ev dataset. Results show averages on 2 network simulations that were each trained from scratch using random initialization seeds. Errors on the other hand denote the standard error.

	No res.	With res.	Control (res.)
Median number of spikes	L1: 1662 ± 6	L1: 1665 ± 6	
	L2: 1066 ± 9	L2: 1057 ± 14	L3: 612 ± 1
	L3: 203 ± 2	L3: 741 ± 3	L4: 203 ± 0
	L4: 70 ± 0	L4: 228 ± 2	All: 3540 ± 9
	All: 3006 ± 5	All: 3696 ± 5	
Accuracy	L1: $49 \pm 0.31\%$	L1: $47.81 \pm 0.53\%$	
	L2: $68.64 \pm 0.18\%$	L2: $67.14 \pm 0.3\%$	L3: $38.11 \pm 0.59\%$
	L3: $47.08 \pm 0.88\%$	L3: $68.03 \pm 0.38\%$	L4: $41.33 \pm 0.35\%$
	L4: $50.45 \pm 0.51\%$	L4: $71.25 \pm 0.1\%$	All: $66.95 \pm 0.04\%$
	All: $70.64 \pm 0.49\%$	All: $76.7 \pm 0.49\%$	

We again use the trained networks to perform classification of the different digits, using the same protocol as Sec. 5.2.2. Table 5.4 displays the results achieved by the networks. Surprisingly, in the absence of residual connections between L1 and L3, the accuracy starts to decrease in L3 and L4. On the other hand, with the use of residual connections, accuracy increases with layers. Residual connections significantly increase the number of spikes in L3 which might be the cause for L3's increased accuracy. For this reason, we also tested the PCL network trained with residual connections, with random residual connections (Control). We observe a large drop in accuracy (from 68.03 to 38.11 %).

Overall, those results show that for learning hierarchical representations in spiking networks, residual connections between layers can be helpful. This could explain the importance of such pathways in the visual system of mammals, to build robust representations regardless of the depth of an area.

5.3 Discussion

With deeper versions of the PCL network, we showed how hierarchical spiking neural networks might be used to build invariant representations. Crucially, we show that different synaptic projection strategies (e.g. residual connections and feature-specific connections) might be important to avoid information loss with network's depth. Considering how PCL is inspired from the visual system of mammals, this study also gives some insight as to the potential role of the diverse projection types observed in the brain, notably in the ventral stream.

The most important idea we showed here for the building of invariant representations, is this notion of one-to-one feature projections from one layer to the next. This is in effect similar to the pooling operation used with so-called convolutional neural networks (CNNs) [130]. Similar schemes had been proposed by Kheradpisheh et al. [122] however our work instead goes further by learning this pooling with STDP using LIF neurons (and not IF neurons). Furthermore, with this study, we also evaluate the use of other projection types, i.e. residual connections, but also test on event-based data curated for the evaluation of translation-invariance.

The one-to-one feature projection used in this study, while inspired from observations of precise feature connectivity between different brain areas (from V4 to area TE in IT) could be further made biologically plausible. We hypothesize that in the brain, rather than feature "slices" being connected so precisely to one other area, it might rather be that a pool of similar features project to the same neuron. We discuss in Chapter 7 how that might be implemented in networks of spiking neurons.

Overall, with the such deeper instances of the PCL network, we showed that PCL can be used to perform invariant object recognition, using only a local STDP rule to learn **all** synaptic connections.

5.4 Conclusion

In this chapter, we evaluated different deeper versions of the PCL network. However, a missing element of our framework thus far would be: recurrent excitatory connections. To tackle this aspect, in the next chapter, we present an extension of the PCL network which uses excitatory connections but also synaptic delays.

SPIKING PREDICTIVE RECURRENT INHIBITORY AND EXCITATORY NETWORK: LEARNING A WORKING MEMORY

Overview	106
6.1 Introduction	107
6.2 Methods	109
6.2.1 From PCL to SPRIER	109
6.2.2 New synaptic plasticity mechanisms	110
6.3 Sensory learning effects	113
6.3.1 Impact of synaptic delays	113
6.3.2 Impact of recurrent excitation	115
6.4 Learning spatiotemporal sequences of gratings	118
6.4.1 Reproducing V1-like effects	118
6.4.2 Impact of the synaptic delays	119
6.5 Learning to predict gestures	121
6.6 Discussion	123
Conclusion	125

Overview

In this chapter, we build on the Predictive Coding Light network presented in Chapters 4 and 5 to propose a new spiking neural architecture that we name “Spiking Predictive Recurrent Inhibitory and Excitatory Recall” (SPRIER) networks. The SPRIER network is an effort to respond to some of the limitations of the PCL network in which mechanisms such as synaptic delays and recurrent excitatory pathways are absent despite their relevance in biological networks. Those mechanisms are crucial as they are thought to be behind the emergence of complex network dynamics associated with memory.

This work will be submitted in a peer-reviewed venue in the near future, and its contributions to the literature are as follows:

- We present an extension of the PCL network with synaptic delays and recurrent excitatory synaptic connections and introduce new engineering-based modulated plasticity approaches to solve the issue of runaway activity in spiking networks.
- The SPRIER network can reproduce spatiotemporal sequence recognition and prediction learning effects observed in V1 [82].
- The SPRIER network also achieves learning of complex spatiotemporal sequences (e.g. gestures [7]) to build a memory of past spiking activity patterns which can be used to perform classification tasks.

The organization of this chapter is as follows: we first provide insight into the motivations behind the SPRIER network and briefly discuss related work. Second, we introduce the mechanisms of the SPRIER network. Third, we describe the results obtained in the reproduction of biological findings on visual recall effects. Fourth, we test our SPRIER network on real-world event-based data (gestures).

6.1 Introduction

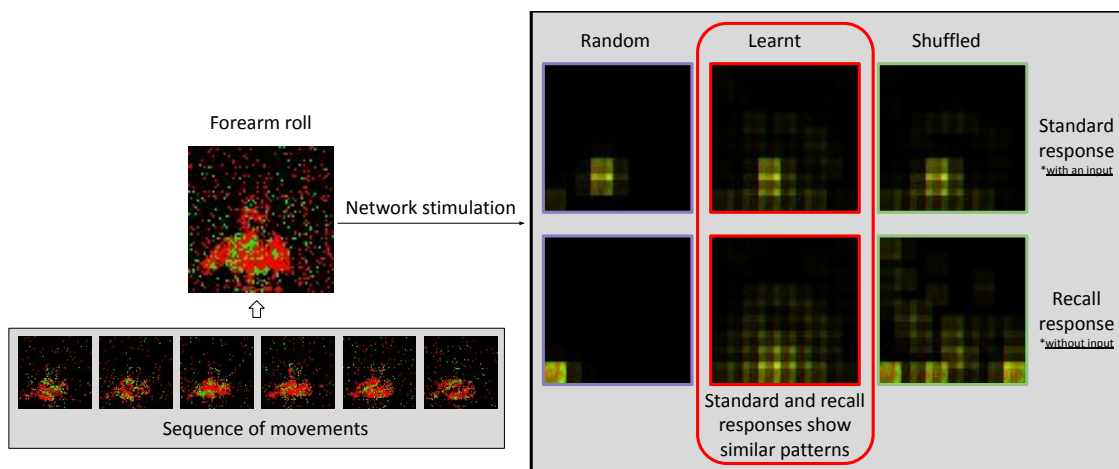


Figure 6.1: **Spiking patterns of a SPRIER network.** Top left shows an event-based input pattern representing a forearm roll gesture, with bottom left showing example movements composing this gesture. On the right side of the figure, the SPRIER network is stimulated with such event-based gestures and two response patterns: a standard one in which the network is excited by an external input, and a recall response in which the network is active from its spontaneous activity alone, are shown. Three networks are considered with a random SPRIER network, a learnt SPRIER network and a shuffled SPRIER network. Zones that exhibit neural activity are represented in the frames such that bright colors indicate strong network activity and dark colors indicate weak activity. The learnt network exhibits similar patterns for both its standard and recall responses.

The ability to recall past information and use it to predict future events in time has been key to the success of artificial neural networks when processing sequential inputs.

While it is believed that the brain achieves this task by using recurrent excitatory pathways and by integrating inputs at different timescales, how to develop a working memory in a network of spiking neurons still remains an open question [64], [129], [211].

Maintaining a memory of the past is commonly associated with attractor networks [98], [103]. In this framework, neurons are recurrently connected such that when they are excited with inputs, they start firing and their global activity evolves towards a stable state which encodes memory. Those firing patterns often happen either in space or in time. See [15] for a review of models of working memory. Regarding spiking works with attractor networks, Dynamic Neural Fields (DNFs) [208] have been suggested for neuromorphic implementation of cognitive processes. DNFs formalize the activation patterns of a group of recurrently connected neurons such that the presentation of an input leads to attractor dynamics. They have been implemented as networks of spiking neurons on neuromorphic hardware for event-based object tracking, and modeling of attention [64], [243]. Those works show promising results but use fixed weights for both sensory processing and attractor dynamics. [129] learnt simple event-based visual sequences in a population of spiking neurons implemented

on a neuromorphic electronic device. While their results are impressive, they do not extract complex features, as they use Poisson spike trains and simple event-based blinking laser as sensory inputs. Interestingly, another category of works use so-called “liquid state machines” (LSM) [149], [150] that consist of a filter made of spiking neurons with randomly initialized fixed (or learnt [228]) connections between each other. LSM map input functions that depend of time to output functions. Thus, in this configuration, memory is directly integrated by LSM. Overall, allowing a network to memorize its inputs implies that neurons will spike even when there is no input to the network. Since it is also known that spiking is a source of energy consumption in the brain [140], as we discussed previously, this suggests that biological systems need to find a trade-off between memorization of input (which leads to more spikes) and efficiency (which leads to a sparse code).

Building on the PCL network, we thus introduce SPRIER: a Spiking Predictive Recurrent Inhibitory and Excitatory Recall network, that learns to maintain a memory of past inputs. SPRIER uses the previous feedforward excitation, local lateral, distant lateral and top-down inhibitions showcased in the PCL network, but also adds distant lateral and top-down excitations paired with synaptic delays. Importantly, all synaptic connections in the SPRIER network are learnt without supervision using STDP rules.

The behavior of the SPRIER network can be described succinctly as follows: neurons remove (immediate) redundant neural spikes through inhibitory connections, and learn to predict (distant) future inputs, implicitly establishing a memory of past inputs that aid in predicting future inputs.

While SPRIER implements “memory” through very long delays of recurrent excitatory connections, a similar effect could arise from the activity of many excitatory neurons being connected in space [59] and time [112]. This interpretation of SPRIER would be more consistent with what is known of sensory cortex, where the presence of different excitatory feedback pathways is well established [201].

Figure 6.1 illustrates the response of a SPRIER network with learned connections and compares it to SPRIER networks with random and shuffled connections, when excited with a sequential visual input representing a “forearm roll” gesture.

In the next section, we detail how the SPRIER network is built.

6.2 Methods

6.2.1 From PCL to SPRIER

The SPRIER network is a direct extension of the PCL network as an effort to model memory effects. Figure 6.2 shows the architecture of the SPRIER network. We describe below the new synaptic connectivity.

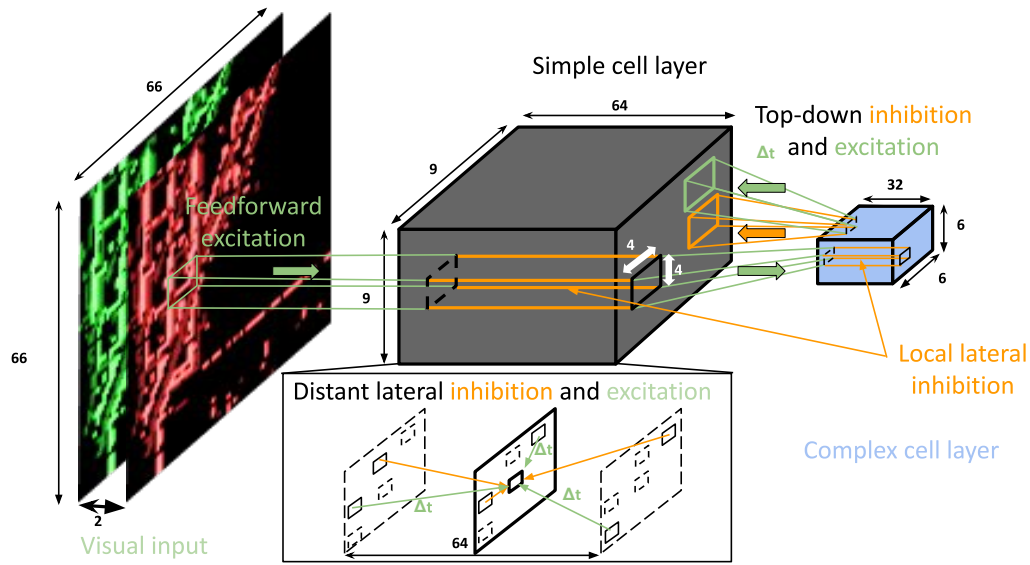


Figure 6.2: **Network architecture.**

Recurrent synaptic excitation We introduce distant lateral and top-down excitation to promote the retention of past spiking patterns and thus function akin to a “memory”. We however still also use our previous distant lateral and top-down inhibitory connections to remove highly predictable spikes and stabilize network activity. Indeed, spiking neural networks with recurrent excitatory connections are prone to runaway activity where neurons’ activity levels keep increasing indefinitely due to self-excitation. Therefore, inhibition is necessary for healthy network dynamics in addition to allowing the suppressive effects studied in Chapter 4. The connectivity pattern of distant lateral excitatory connections are the same as inhibitory ones: simple cells can receive excitation from a 3-dimensional range of other simple cells, as long as they don’t share the same input zone. The same logic as top-down inhibition is also applied for top-down excitation, where complex cells can also excite all simple cells in their input zone. However, to avoid that excitation and inhibition cancel each other due to this connectivity pattern, we allow distant lateral and

top-down excitation to happen later in time by introducing synaptic delays (see next paragraph). However, all weights are still learnt via STDP rules.

Synaptic delays We use synaptic delays in our recurrent excitatory synaptic connections, and also test their effect on other synaptic connections (feedforward and recurrent inhibitory connections). The idea behind synaptic delays is as follows: by introducing potentially long transmission times, neurons can easily learn about temporal correlations between spikes that happen far in time. Delays have been characterized as mechanisms that allow networks to exhibit complex spiking patterns [112]. While we do test the influence of synaptic delays on a grand number of connections, in the final version of the SPRIER network, we keep them only for the recurrent excitatory connections and not the recurrent inhibitory connections. This choice is made to keep things simple and avoid long simulation times. For simplicity, unless specified otherwise, we initialize delays from a wide uniform distribution [100 ms, 500 ms]. Please note that we do not learn the synaptic delays as in other approaches [17], [112].

6.2.2 New synaptic plasticity mechanisms

Stability issues of recurrent artificial neural networks (whether spiking or not) have been well-known for many years now with either mechanistic (fixed weight) solutions [59] or through stabilization with inhibitory synapses [204]. Those studies highlight the formidable challenge that is the generation of healthy network dynamics with recurrent excitation. As a spiking neural network framework, the SPRIER network also faces the same issue. To address it, we propose two new plasticity mechanisms (which we describe below) that we empirically found to help in generating balanced inhibitory and excitatory spiking networks: a plasticity modulation of the recurrent excitatory connections and a modulation of the inhibition. The intuition behind those mechanisms lie in 1) gating recurrent excitatory plasticity to prevent an exploding number of spikes (runaway activity), and 2) avoiding the direct cancellation of inhibition and excitation in a network that breaks Dale’s law. Essentially, since the SPRIER network breaks Dale’s law to reduce the number of neurons (and therefore also the computational cost), such engineering-based mechanisms are necessary for the implementation of recurrent excitation. For instance, in a spiking network with separate (but also unstructured) circuits of excitatory and inhibitory neurons, it would be possible to directly control which neurons would be active from the visual input, which neurons would be active from recurrent excitation and lastly have inhibition target separately one or the other pool of neurons. In SPRIER, we use our new plasticity mechanisms as they allow us to achieve similar operations with fewer neurons. Please note that we are still using all of the previous mechanisms (STDP, softbound synaptic plasticity and weight normalization) of the PCL network.

The connectivity parameters of the networks are the same as in Chapter 4 but some of the membrane potential and plasticity parameters are slightly different, e.g. thresholds, learning rates and synaptic resources, to improve the network’s dynamics (see Tab. 6.1).

Table 6.1: Neurons’ membrane potential and plasticity parameters.

Simple cells					
V_{reset} (mV)	V_{thresh} (mV)	$\tau_{\text{m/I}}$ (ms)	τ_{RP} (ms)	τ_{LTP} (ms)	τ_{LTD} (ms)
-10	30	18	5	7	7
η_{RP} (mV)	w_{min} (mV)	w_{max} (mV)	$\eta_{\text{LTP/LTD}}$ (mV)	$\eta_{+/-}$	λ
10	0	f-f. excit: 3	f-f. excit: ± 0.000204	f-f. excit: 0.33	f-f. excit: 50
		dist. inhib.: 30	dist. inhib.: ± 2.652	dist. inhib.: 0.033	dist. inhib.: 6500
		td. inhib.: 30	td. inhib.: ± 1.428	td. inhib.: 0.033	td. inhib.: 3500
		loc. inhib.: 30	loc. inhib.: ± 0.612	loc. inhib.: 0.033	loc. inhib.: 1500
		lat. excit.: 30	lat. excit.: ± 3.06	lat. excit.: 0.033	lat. excit.: 750
		td. excit.: 30	td. excit.: ± 1.02	td. excit.: 0.033	td. excit.: 250
Complex cells					
V_{reset} (mV)	V_{thresh} (mV)	τ_{m} (ms)	τ_{RP} (ms)	τ_{LTP} (ms)	τ_{LTD} (ms)
-10	3	50	5	40	40
η_{RP} (mV)	w_{min} (mV)	w_{max} (mV)	$\eta_{\text{LTP/LTD}}$ (mV)	$\eta_{+/-}$	λ
5	0	excit: 4	excit: ± 0.08	excit: 0.25	excit: 1000
		loc. inhib.: 25	loc. inhib.: ± 0.048	loc. inhib.: 0.04	loc. inhib.: 600

Plasticity modulation In a simple hierarchical network that has only one neural population such as ours, recurrent excitation can lead to runaway activity. We regulate recurrent excitatory synaptic plasticity according to feed-forward and recurrent excitatory input levels to avoid this problem. First, we define the trace I of a synaptic input as:

$$I(t + \Delta t) = I(t)e^{-\frac{\Delta t}{\tau_1}} + w_i(t + \Delta t), \quad (6.1)$$

where Δt is still the time since the last update of the membrane potential, τ_1 controls the decay of the trace and $w_i(t + \Delta t)$ is a synaptic input. We denote as I_{ffExc} and I_{rExc} , respectively, the traces of feedforward and recurrent excitatory synaptic

inputs received by a given neuron. We modulate recurrent excitatory plasticity so that it follows an sSTDP rule that switches from fully LTP to LTD when recurrent excitation becomes too high. This modulation should encourage recurrent excitation in learning correlations in spikes caused by the external world. The coincidence of feedforward and recurrent excitatory activities thus push the two synaptic connections to become similar, whereas in the absence of feedforward activity, recurrent excitatory connections are reduced which pushes the network to a state of feedforward-driven activity. Arguably, to build a memory, the brain must be performing some similar sort of operation, which we here provide with this innovative model. Plasticity in recurrent excitatory synapses follows:

$$\Delta w_i^{\text{rExc},s} = \begin{cases} \log\left(\frac{I_{\text{ffExc}}}{\beta \times I_{\text{rExc}}}\right) \tilde{\Delta} w_i^{\text{sSTDP}} & \text{if } I_{\text{rExc}} > 0, \\ 0 & \text{if } I_{\text{rExc}} = 0, \end{cases}$$

where $\Delta w_i^{\text{rExc},s}$ refers to the synaptic weight update of simple cells' recurrent excitatory weights and β is a parameter that controls the influence of the ratio of excitatory inputs. We set β to 1.

In the above equation, LTP happens when the logarithm is greater than zero (more feedforward excitation than recurrent excitation) and LTD happens when it is smaller than zero (more recurrent excitation than feedforward excitation). Furthermore, to avoid having excessive updates of the weights, we keep the logarithm term bounded within the interval $[-2, 2]$. We set the logarithm term to -2 when $I_{\text{ffExc}} = 0$.

Inhibition modulation When presented with weak inputs, a spiking neuron recruits both recurrent excitation and inhibition. In the model presented so far, a neuron receiving a weak input would receive as much inhibition as a neuron that receives a strong input. To prevent this, we modulate distant lateral and top-down inhibitory currents so that their strengths depend on the amount of feed-forward and recurrent excitatory inputs. We reduce inhibition when a neuron spikes despite little feed-forward excitation, such that its response is amplified. We motivate such a mechanism by biological observations of disinhibition during sensory experience that suggest its importance in learning and the formation of memory [141]. Let $\hat{w}_i^{rI}$ be the modulated amount of distant lateral or top-down inhibitory input. It evolves according to:

$$\hat{w}_i^{rI} = w_i^{rI} e^{-\frac{I_{\text{rExc}}}{I_{\text{ffExc}} + \alpha \times I_{\text{rExc}}}}, \quad (6.2)$$

where α is a constant that we set here to 0.5. When a neuron receives little to no excitation (both feed-forward and recurrent), it also receives little to no inhibition.

This allows a network to enter directly a “recall” mode whenever no outside input is received.

6.3 Sensory learning effects

As we added new learning mechanisms and connection types, we wondered about the impact they would have on the network. We first study the effect of delays on feedforward excitation, distant lateral inhibition and top-down inhibition. Second, we analyze the effect of recurrent excitation and thus defend the use of our new plasticity mechanisms.

6.3.1 Impact of synaptic delays

We analyze the effect of synaptic delays on the feedforward excitatory connections of our simple cells. Please note that we do not activate the recurrent excitatory connections here. Each pixel in the input zone of a simple cell have a different synaptic delay (Figure 6.3a) that we draw from either a random lognormal distribution or a random uniform distribution. We share those delays across simple cells at different retinal positions and then train our network on our event-based natural images dataset (see Chapter 4) for 5 epochs.

We initially expected synaptic delays to have only a mild impact on the learning of Gabor-like receptive fields as connections between biological neurons also have different transmission speeds, however we instead observed that the Gabor fit error of our simple cells significantly increased with the range of the delays (Figure 6.3b). This effect is observed with both uniform and lognormal distributions of synaptic delays as not only the mean value of the delays increase, but the variance of the delays also do (Figure 6.3c-d).

Simple cells’ receptive fields appear to become noisier with a strong variability in synaptic strength across one receptive fields, which led us to the hypothesis that large delays might cause a weakening of the synaptic strength between neuron in sensory learning which induces this sort of “noisy” effect in learning. This is indeed the case in our SPRIER network as synaptic strength reduces for synapses with high delays for both uniform and lognormal delay distributions (Figure 6.4a-b).

Those results thus lead us to the conclusion that for the event-based visual input we are considering here simple cells require synaptic inputs with a narrow delay distribution to form Gabor-shaped receptive fields. For this reason, we decide not to

use any sort of synaptic delay for our feedforward synaptic connections. As for the local lateral inhibition, due to its local nature, we do not consider this case.

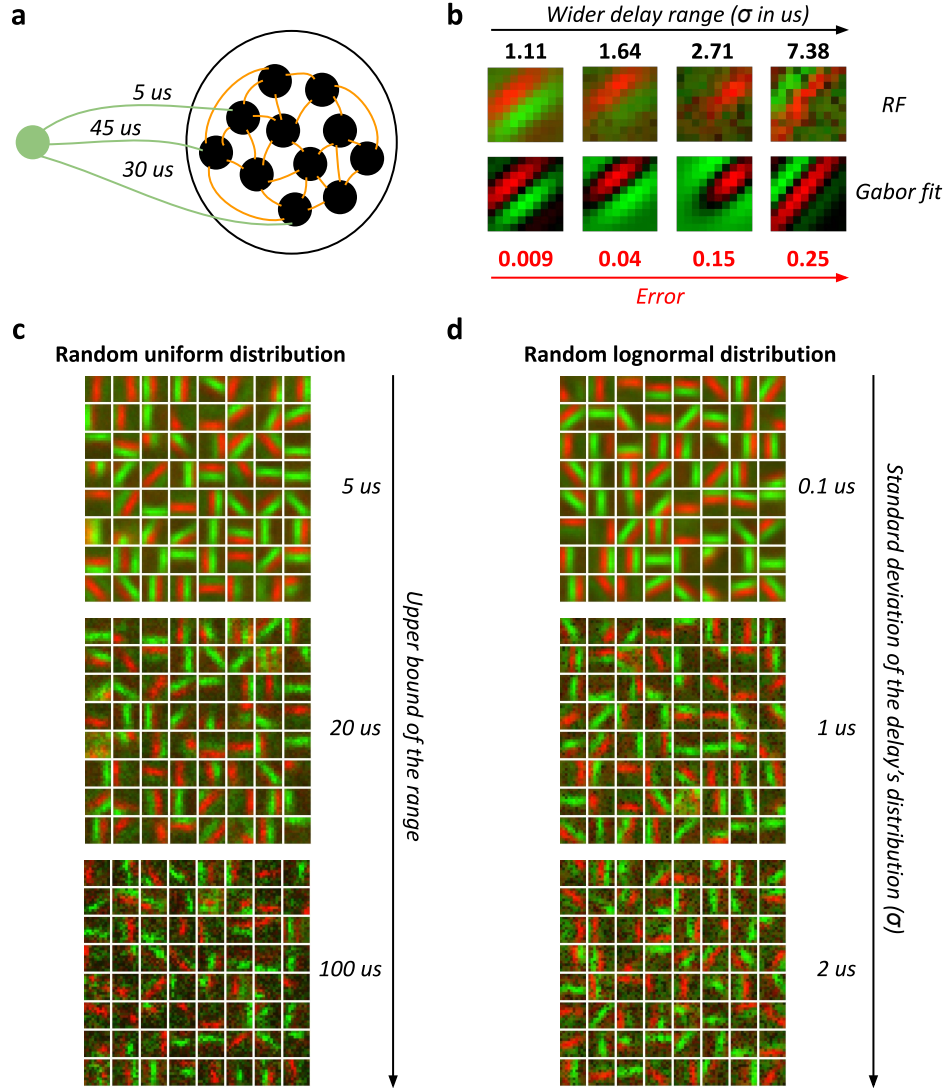


Figure 6.3: **Effects of feedforward synaptic delays on sensory learning in the SPRIER network.** **a**, Simplified network architecture of the simple cell layer of the SPRIER network. Synapses connecting input events to simple cells each have different delays that are shared between simple cells at different retinal locations. **b**, Effect of a wider delay range on simple cell learning and Gabor fit. **c**, Simple cells' receptive fields learnt with different uniform delay distributions that have different upper bounds on their range. **d**, Same as c but with different lognormal delay distributions that have different standard deviations.

Next, we consider the impact of synaptic delays on our distant lateral and top-down inhibition and train a network on natural images with delays only for those connections. Similarly to our feedforward excitatory connections, we observe a reduction of the weight strength with the delay of the synapse for both types of inhibitory connections (Figure 6.5a-b).

Interestingly, surround suppression increases with the width of the delay's underlying distribution (Figure 6.5c). Our explanation of this result is that the network can now more easily cancel spikes at various future times and not only immediate

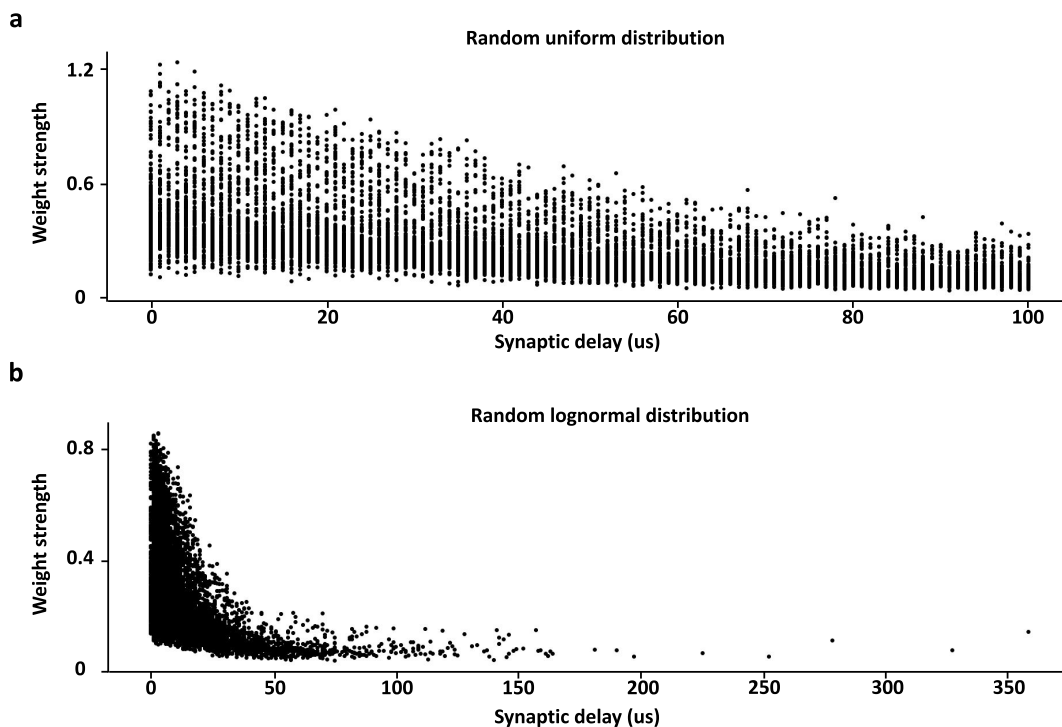


Figure 6.4: **Effect of synaptic delays on simple cells' weights.** **a**, Weight strength of simple cells' synapses for delays initialized from a random uniform distribution. **b**, Same as a but for delays initialized from a random lognormal distribution.

ones. As we use moving input patterns to train the network, there is redundant information in the movement of the stimulus, which can be better removed when distant lateral inhibition and top-down inhibition have synaptic delays.

Nevertheless, in SPRIER, to reduce computational burden while allowing a difference between recurrent inhibition and excitation, we decide not to use any synaptic delays for our distant lateral inhibition and top-down inhibition, but keep them for their excitatory counterparts.

In summary, in the “default” version of the SPRIER network that we use in the next sections, **only** distant lateral excitation and top-down excitation use synaptic delays, while all other connections have an instantaneous spike transmission.

6.3.2 Impact of recurrent excitation

Figure 6.6 shows a simplified representation of the SPRIER network including its recurrent excitatory connections. When added without caution, this recurrent excitation has a severe impact on sensory learning on natural images as it prevents the emergence of simple cell-like receptive fields (Left side of Fig. 6.6b).

In addition to this effect on sensory learning, recurrent excitation also tends to lead to runaway activity with non-terminating simulations due to loops of excitatory

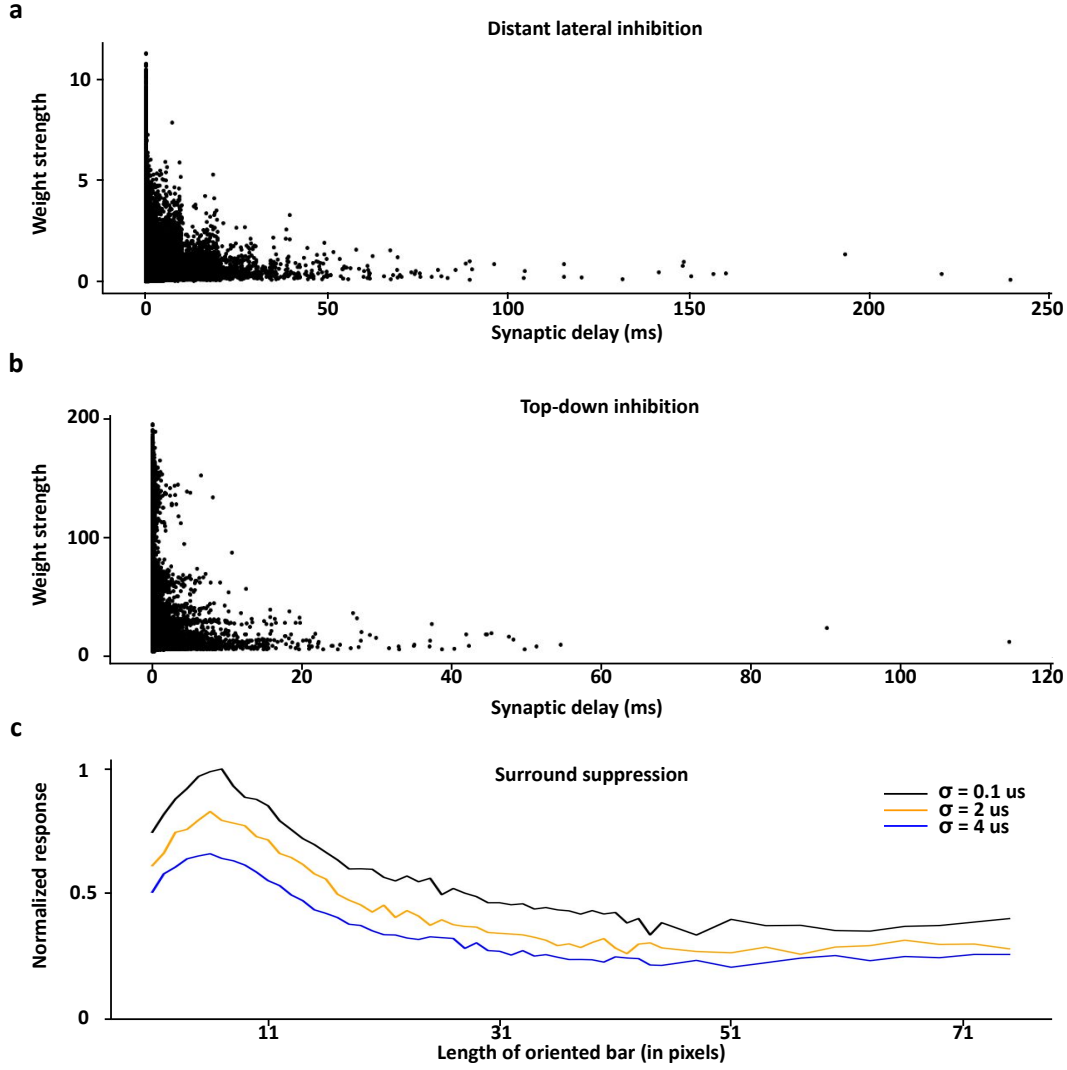


Figure 6.5: **Effects of synaptic delays on the emergence of suppression in a SPRIER network.** **a**, Weight strength of distant lateral inhibitory synapses for delays initialized from a random lognormal distribution. **b**, Weight strength of top-down inhibitory synapses for delays initialized from a random lognormal distribution. **c**, Surround suppression of a SPRIER network for different standard deviations of the lognormal distribution behind the generation of the synaptic delays.

connections that cause endless spikes in the network, even with the presence of distant lateral inhibition and top-down inhibition. To prevent this issue, we carefully choose the λ parameter (see Eq. (4.13) and Table 6.1) of our recurrent excitatory weights as it controls the ability of the network to excite itself. For too high values of λ , the softbound of the network is inefficient in preventing the excessive growth of some synaptic weights (Figure 6.6c).

Ultimately, to achieve sensory learning, we: 1) empirically find recurrent excitatory hyperparameters (λ and softbound) that do not overpower inhibition, 2) train feedforward excitation, local lateral inhibition, distant lateral inhibition and top-down inhibition with fixed weights for recurrent excitation and 3) freeze the feedforward excitatory and local lateral inhibition connections and train simultaneously distant

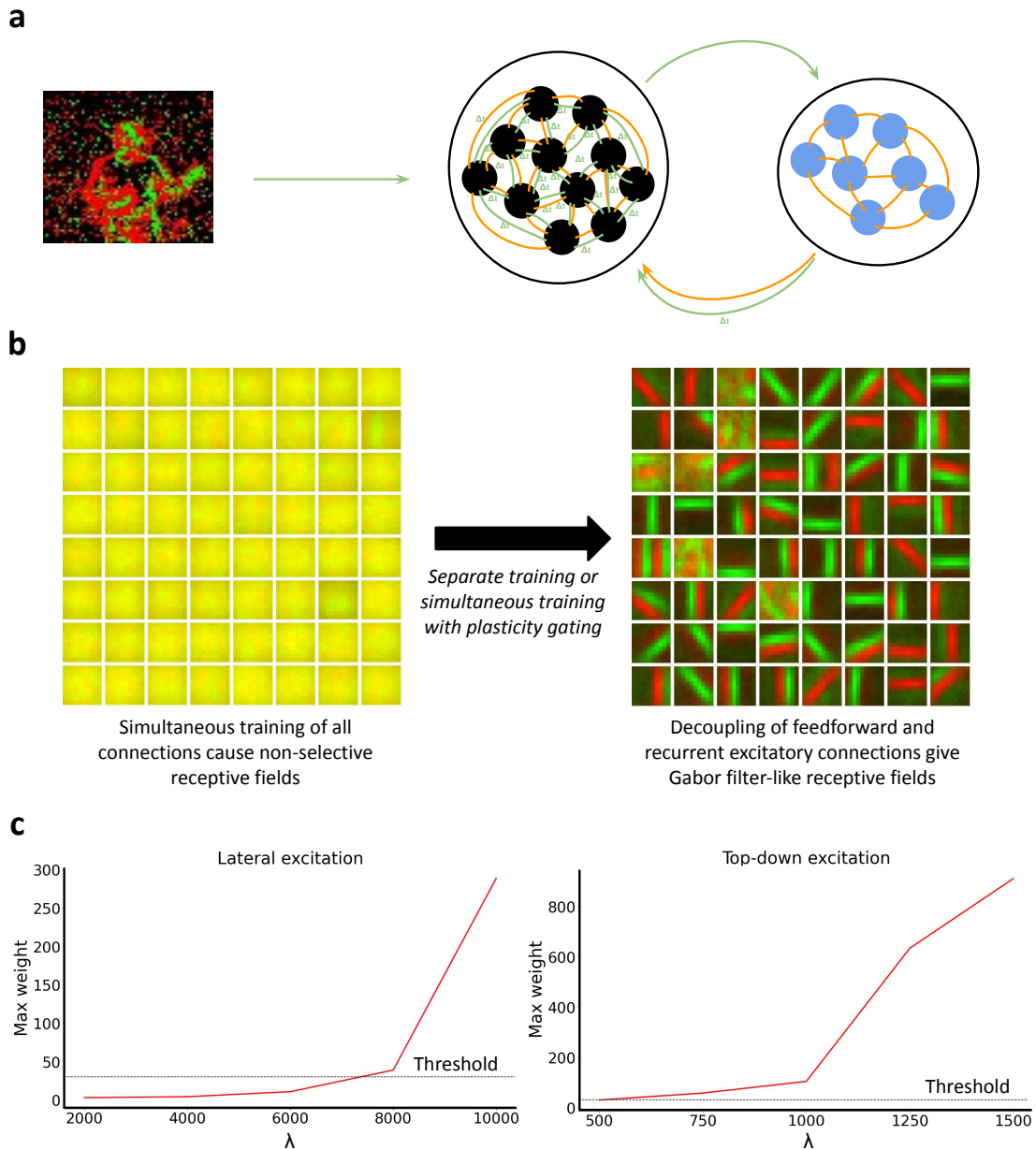


Figure 6.6: **Impact of recurrent excitation on sensory learning in a SPRIER network.** **a**, Simplified network architecture. Dark circles represent simple cells and blue circles represent complex cells. **b**, Effect of a naive training (left) of simple cells' weights with a SPRIER network, and the training approach (right) used in our simulations. **c**, Impact of different excitatory synaptic resources on weight growth for the distant lateral excitation (left) and the top-down excitation (right).

lateral inhibition, top-down inhibition, distant lateral excitation and top-down excitation.

Interestingly, it could be possible to train simultaneously all connections by gating plasticity when the feedforward input is minimal or absent (Figure 6.6b).

In the next sections, we show the effect of recurrent excitation on the learning of spatiotemporal sequences, and examine the use of different delay distributions.

6.4 Learning spatiotemporal sequences of gratings

6.4.1 Reproducing V1-like effects

Response amplification and memory effects have been observed after exposure to sequences of gratings in mouse V1 [82]. Two groups of mice were distinguished: experimental and control. Experimental mice were exposed to a repeating sequence of gratings, while control mice observed random permutations of the grating sequences. Mice from the experimental group demonstrated an increased response to the fixed sequence compared to mice from the control group. They also demonstrated a spontaneous response to an omitted stimulus of the learned sequence, which suggested that strong synaptic connections formed to memorize or anticipate a given temporal sequence.

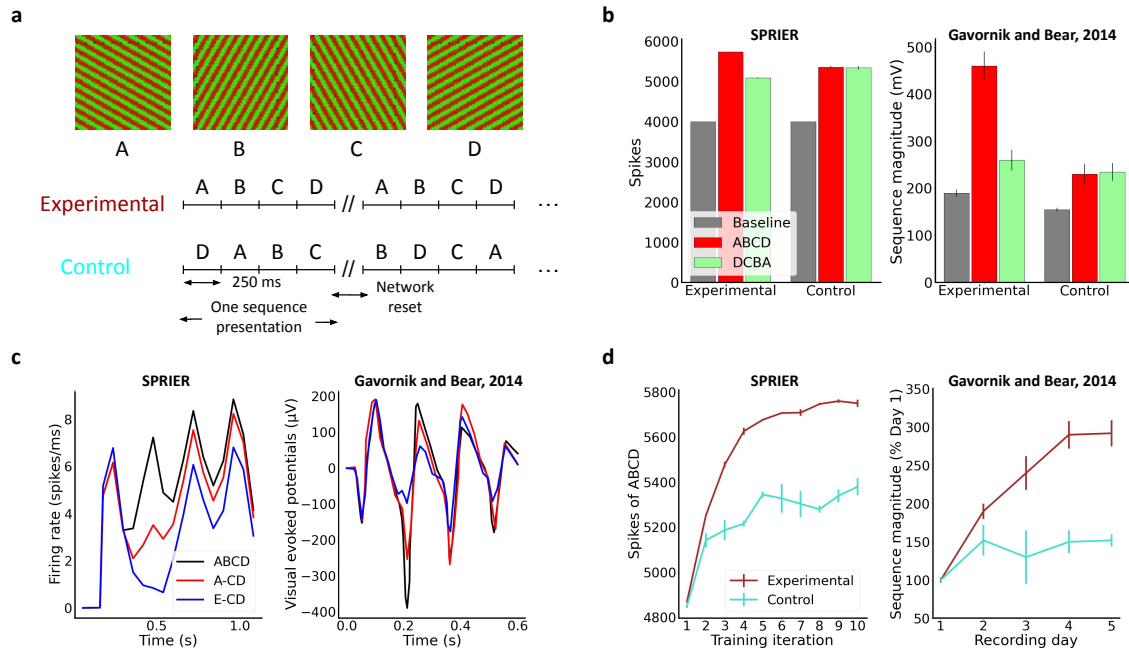


Figure 6.7: **Learnt representations of sequences of counterphase sinusoidal gratings.** **a**, Experimentation process. An experimental network is trained for 10 presentations with a sequence ABCD. After each presentation, the state of the network is reset. A control network is also trained for 10 presentations with sequences of randomly permuted elements of ABCD. **b**, Network responses after training. Left shows the response of the experimental and the control networks and right shows the results observed in mouse V1. The baseline refers to the response of SPRIER with recurrent excitated deactivated. Error bars indicate the standard error across 3 trainings. **c**, Response of the experimental network to an omitted element of ABCD. Left and right again show the results of SPRIER and those observed in mouse V1. **d**, Evolution of network activity during training. Left again shows the response of SPRIER and right shows the results observed in mouse V1. Error bars again indicate the standard error across 3 trainings.

Similarly to the PCL network, we use our natural image database for training the feedforward excitatory and inhibitory connections. We later train our recurrent

excitatory connections and finetune the recurrent inhibitory connections on either sinusoidal gratings or gestures. We describe below the dataset we generate to test the learning of spatiotemporal gratings.

Counterphase sinusoidal gratings We generate 250 ms sinusoidal gratings of various orientations (0° , 30° , 60° , 120° and 150°) with a phase of 0° , a spatial frequency of 1/6 cycles per degree and a temporal frequency of 4 Hz. We construct sequences of such grating stimuli to train our recurrent excitation and inhibition. Please see Fig. 6.7a for an example of such a sequence of gratings.

We replicate the response amplification and memory effects observed in mouse V1 with our SPRIER network using the above event-based sequences of sinusoidal gratings. Our simple and complex cells are first trained on natural images, and their recurrent inhibitory and excitatory connections are next trained on sequences of gratings. Similarly to [82], we consider two networks, an experimental one which sees multiple repetitions of a fixed sequence “ABCD” and a control one which sees at each iteration a different randomly permuted sequence of these elements (Figure 6.7a). Both networks are trained for a total of only 10 sequence presentations as we use here high learning rates.

Similarly to mouse V1, after exposure to sequences of gratings, the experimental SPRIER network responds more effectively to the trained “ABCD” sequence than to an untrained “DCBA” sequence, while the control SPRIER network does not exhibit a preference for one or the other sequence (Figure 6.7b). Also, responses after training are generally stronger than during a baseline period recorded before training. Additionally, the experimental SPRIER network learns to “fill in” missing elements of a sequence. This is tested with the “A_CD”, where “_” denotes an omitted stimulus (Figure 6.7c). Furthermore, we confirm that this sensitivity to spatiotemporal sequences arises through STDP-learning in our SPRIER network and saturates after a certain number of training iterations for both experimental and control networks (Figure 6.7d).

Overall, the SPRIER network reproduces different effects observed in [82], which suggests that recurrent excitatory connections learning via STDP are a good candidate mechanism to explain these phenomena.

6.4.2 Impact of the synaptic delays

We next wondered about the impact of synaptic delays on the emergence of these effects in the SPRIER network. To study the effect of these delays, we here replace the uniform distribution of delays with a **unique** delay for all recurrent excitatory synaptic connections and then proceed to test different delays.

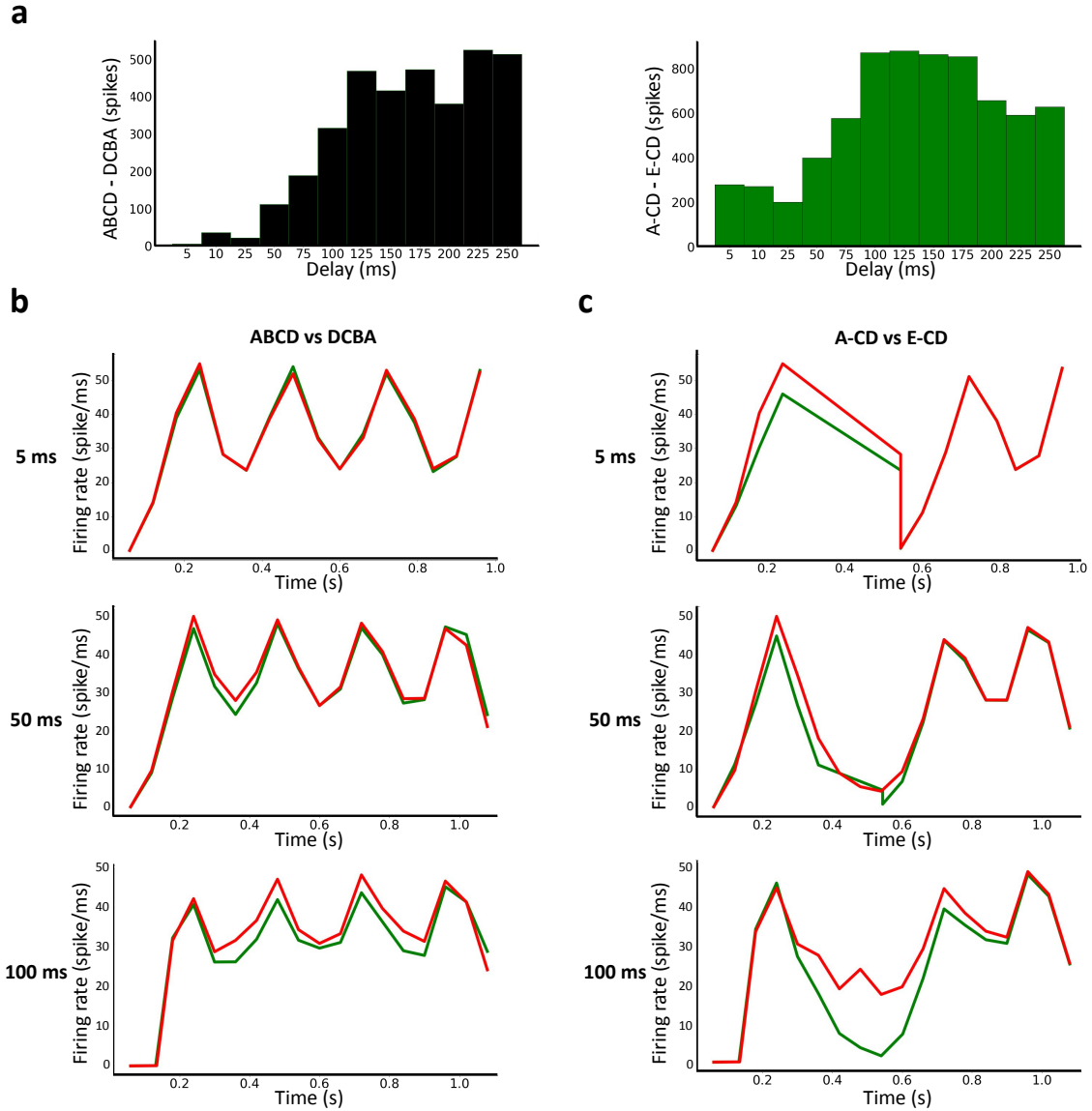


Figure 6.8: **Effect of different recurrent excitatory delays on the emergence of V1-like memory effects.** **a**, Difference in the number of spikes generated for sequences “ABCD” and “DCBA” (left), and for sequences “A_CD” and “E_CD” (right) for different synaptic delays. **b**, Spiking rate over time of the SPRIER network for 3 recurrent excitatory delays after excitation with gratings sequences “ABCD” (in red) and “DCBA” (in green). **c**, Same as **b** but for gratings sequences “A_CD” (in red) and “E_CD” in green.

Our intuition is the following: to bind one sequence element to another in time, delays need to be sufficiently long. In our grating sequences, each element lasts 250 ms, so it would be helpful for delays to be of this order of magnitude.

Our results confirm this hypothesis as short delays (below 100 ms) cannot elicit memory effects (Figure 6.8a). The increased response of the SPRIER network to one particular sequence as well as its ability to fill in omitted inputs thus strongly depend on the synaptic delays used. Figures 6.8b-c show examples of the responses, in time, for different delays.

Essentially, information is able to propagate in time to bind different inputs in the neural representation. In the SPRIER network, this occurs thanks to having large delays, but biological neurons use only very short delays of the order of milliseconds or tens of milliseconds. However, biological neurons might be realizing such long delays with “chains” of neurons that propagate information across multiple synapses. As we want to keep a low number of neurons in our simulations, the use of a range of synaptic delays allows our SPRIER network to capture associations between inputs with different relative timing.

6.5 Learning to predict gestures

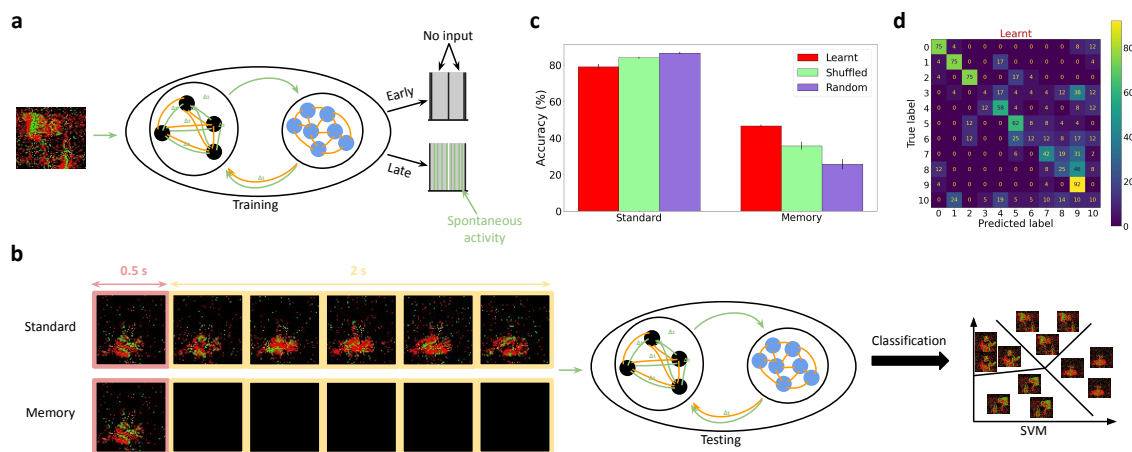


Figure 6.9: **SPRIER learns to predict different gestures through its spontaneous activity.** **a**, Experimental training protocol. The SPRIER network’s recurrent excitatory connections are trained on different gestures. Those learnt excitatory connections allow for the network to be active even in the absence of inputs, after training. **b**, Experimental testing protocol. The SPRIER network is excited with two types of inputs: a standard one in which gestures are shown to the network for 2.5 s, and a memory one in which a cue gesture stimulus is shown for 0.5 s and a blank stimulus is shown for the remaining 2 s. Spikes occurring between 0.5 s and 2 s are used for classifying the gestures using a SVM classifier. **c**, Performance of a learnt SPRIER network, a shuffled one and a random one. Error bars indicate the standard error across 3 trainings. **d**, Confusion matrix of the learnt SPRIER network.

Following the learning of spatiotemporal sequences of gratings, we next wondered if the SPRIER network could learn more complex sequences. We here again consider the DVS128 Gesture dataset. As this dataset contains a wide variety of gestures that cover different retinal regions and have different temporal dynamics, accurately predicting future inputs appears to be a formidable challenge. To characterize the ability of the SPRIER network to learn the temporal structure in these sequences, we first fine-tune the original SPRIER network trained on natural images on the DVS128 Gesture Dataset. After training (Figure 6.9a) for 10 epochs our distant lateral inhibition, distant lateral excitation, top-down inhibition and top-down excitation on a set of 110 gestures (10 elements of each class), we next evaluate the network with a classification task, where it needs classify a gesture although part of the gesture

sequence is omitted. The rationale is as follows: if the SPRIER network properly learnt the spatio-temporal structure of the gestures, then it would be able to “fill in” omitted inputs and the resulting activity might suffice for classifying the gestures — even in the absence of feedforward activity. Therefore, we test SPRIER by first stimulating the network with a 500 ms long “cue” gesture input and then use only the spike train generated from the recurrent excitation of the network for classification. We describe briefly in the next paragraph the process of gesture classification.

Gestures classification. We perform gesture classification using the same process as for the PCL network (see Chapter 4). In this experimental protocol, however, we use either the spikes obtained from our SPRIER network after excitation from 2 s of gestures’ events, or instead use the spontaneous activity generated by the network after presentation of a cue (see Fig. 6.9b). Crucially, we named “standard” the case of using actual gestures, and “memory” the case of using spontaneous activity alone. In the memory case, a sample is fed to the linear SVM (for both training and testing) if it actually generated activity (a non-zero descriptor). During testing, we lower the spiking threshold of our networks from 30 mV to 18 mV to allow more spikes to happen. We excite the network with a cue sensory input that lasts 500 ms and record for 2 s the spikes occurring from SPRIER’s spontaneous activity (Figure 6.9b).

To see if the learnt recurrent excitation is fundamental to the working memory of SPRIER, we perform this experiment with 3 networks: a random network where recurrent excitatory weights are initialized randomly, a learnt network and a shuffled network where recurrent excitatory connections of our learnt network are randomly permuted for all cells. We train for each of those networks’ descriptors different SVM classifiers. We then proceed to test the different networks’ memory abilities with the classifiers that were trained on the standard (full) sequence.

Figure 6.9c shows that our classifiers all achieve accuracies around 70% when presented with a full sequence. Surprisingly, we observed that the network with learnt recurrent excitation performed worse than others when excited with a full sequence. As different gestures might share similar features, we hypothesize that this effect arises from the network trying to encode too many gestures with too few number of features which might end up degrading the representation. The random and shuffled networks do not suffer from this problem as they blindly encode information without learning. We also see on Fig. 6.9c that when the sequence is cut short and the classifier is presented only with spontaneous activity, the network with learnt connections outperforms both random and shuffled networks. This result suggests that the learnt network uses the initial cue to create a spiking representation that persists in time. Figure 6.9d shows the confusion matrix. For most classes, the network tends to give correct predictions.

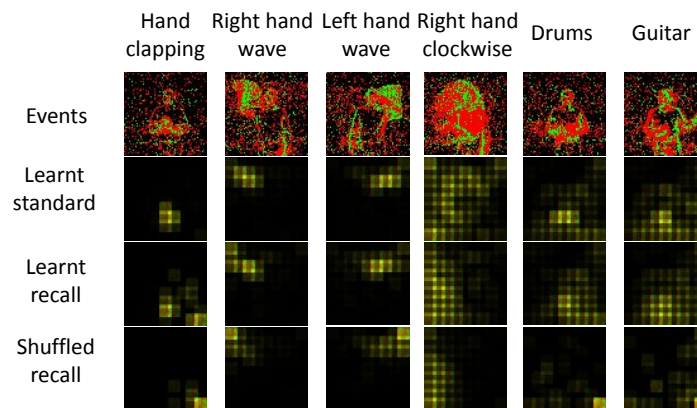


Figure 6.10: **Spiking representations of the gestures for different SPRIER networks.** Bright colors indicate zones that elicit strong network activity while dark colors indicate zones that elicit weak activity.

Figure 6.10 shows examples of the zones of the network that are activated during a standard presentation and during only spontaneous activity. We see that the learnt network has activity patterns that resemble its evoked response, i.e., it is successfully filling in the missing information. The shuffled network, however, has a poor ability to fill in this missing information, which is reflected in its localized and minimal activity.

Overall, we conclude that the interplay of fast inhibition and delayed excitation might play an important role in permitting recurrent spiking networks to learn efficient codes while also retaining memories of past events.

6.6 Discussion

The Spiking Predictive Recurrent Inhibitory and Excitatory (SPRIER) network proposes novel ways of maintaining stability in recurrent excitatory spiking neural networks. Similarly to the PCL network, SPRIER learns simple and complex cell-like representations but also adds further capabilities by learning to fill in missing information via its recurrent excitatory connections. The SPRIER network exhibits such effects with gratings, reminiscent of findings in mouse V1, but also demonstrated such effects for complex gestures. Interestingly, despite the overlapping positions and features between gestures, the filled-in activity in the SPRIER network can still be effectively used to classify the gesture. STDP had been already observed as a crucial plasticity mechanism in neural learning and the SPRIER network further highlights the ability of this generic learning rule to generate different response patterns using a wide variety of synaptic connections, including delayed ones.

In the gesture classification experiments performed here, we observed relatively low accuracy compared to other classification methods such as [218], [253]. This can be explained by our choice to classify samples using only a minimum part of their time-span (2 s vs 6 s), to reduce simulation time. Furthermore, we also considered here a fairly simple spike descriptor based on temporal binning of spikes. Regarding the findings for gratings, other models such as [95], [126] have also reproduced biological sequence learning effects via STDP of recurrent excitatory connections, but their input stimuli consist of frames and not event-based data as in our work. Furthermore, contrary to our framework, they do not learn V1-like visual features.

Similarly to PCL, the SPRIER network is also closely related to the predictive coding theory. Essentially, the distant lateral and top-down inhibition still allow for hallmarks of predictive coding such as suppressive effects (e.g. surround suppression) to emerge in the network. However, the addition of recurrent excitatory connections allows spontaneous activity and the emergence of a fading memory in the neural network. Crucially, despite the additional activity occurring in the SPRIER network, any new spike still causes inhibition (albeit modulated by the frequency of the input events). The PCL network was only able to remove predictable inputs and lacked the generative aspect displayed by other predictive coding methods. By having recurrent excitation, SPRIER can now implicitly represent “positive prediction error” signals and neurons can spike more when they “over”-predict inputs by attempting to fill in missing inputs. Such “over”-prediction is suppressed in time (similar to a fading memory), thanks to the distant lateral and top-down inhibition. Essentially, the SPRIER network stays consistent with the predictive coding theory.

We used some mechanisms to facilitate learning in SPRIER, which may not be biologically plausible. Most notable are our plasticity modulation and inhibitory modulation mechanisms. We believe that they allow a simple use of recurrent excitation while still exhibiting similarities to biology regarding the effects they produce. In the context of neuromorphic computing, with limited number of resources, it is interesting to explore such novel ideas that may help to reduce the number of neurons required for learning a certain task while still maintaining good performance. While we only considered fixed delays, learning delays has been shown to have important benefits [89], [93], [151]. Learning delays would therefore be an exciting area of research and could permit a more precise binding between different elements in temporal sequences.

Conclusion

In this chapter, we built on Predictive Coding Light (PCL) to propose a new model, the Spiking Predictive Recurrent Inhibitory and Excitatory (SPRIER) network. It adds recurrent excitation and synaptic delays to give the network a fading memory and permit the “filling in” of missing sensory inputs.

DISCUSSIONS AND PERSPECTIVES

7.1	Conclusions and discussions	128
7.2	Perspectives on improvements	129
7.2.1	Neurobiological aspects	129
7.2.2	Application aspects	130
7.3	Perspectives on future applications	131

7.1 Conclusions and discussions

In this thesis, we presented a novel efficient spiking neural network framework called *Predictive Coding Light* (PCL) and an extension of PCL, which we denote *Spiking Predictive Recurrent Inhibitory and Excitatory Recall* (SPRIER).

First, the PCL network is inspired from the visual cortex of mammals and has been shown in this thesis to: 1) model the response properties of simple and complex cells, 2) replicate surround suppression, orientation-tuned suppression and cross-orientation suppression that are seen as hallmarks of predictive coding, 3) encode with low energy (few spikes) complex spatiotemporal event-based sequences such as digits and gestures, 4) learn invariant visual representations with residual connections and one-to-one feature projections.

Crucially, in the PCL network, **all** synaptic connections are learnt using only STDP rules, which is an important advance relative to most other models of the visual cortex (see Chapter 2), where only a subset of the connections are learnt. PCL demonstrates the flexibility of spiking approaches of predictive coding with its novel approach of modeling prediction errors *implicitly*. PCL is also, to our knowledge, the only spiking predictive coding work that **learns** visual cortex-like responses and shows clear energy efficiency in information processing. Essentially, with the PCL network, we have proposed an unsupervised learning framework that develops **efficient** and **invariant** visual representations. The PCL framework can both be used to model cortex-like responses (e.g. V1-like response properties) but also for engineering applications (e.g. classification). Interestingly, due to its generic architecture and learning rules, its principles could be readily applied to other sensory modalities, but this topic is left for future work.

The SPRIER network extends PCL by exhibiting a memory of past synaptic inputs through the use of recurrent excitatory connections and synaptic delays. As for PCL, its learning rules are based on STDP. SPRIER has been shown to replicate effects observed in the visual cortex of mammals with the learning of spatiotemporal sequences of gratings. Additionally, we have demonstrated that the network can also be applied to more complex sequences such as gestures. Concretely, we have shown how the SPRIER network learns to maintain a memory of those sequences after their disappearance.

While both PCL and SPRIER violate Dale’s law, both networks are still interesting as neurobiological models given the wide variety of effects that they reproduce. We have shown the functional role of feedback pathways in the emergence of different brain-like response patterns. We have also demonstrated in both frameworks how different types of excitatory and inhibitory synapses could be used together to learn

sensory representations and achieve healthy spiking dynamics. Importantly, we have also provided systematic evaluations of various brain observations (e.g. learning rules, projection types between areas) to provide an overview of their effect on sensory learning.

On the application side, our frameworks have both been evaluated with event-based camera data. This allowed us to establish advantages regarding energy efficiency, information coding and memory capabilities, which make these networks appear suitable for technical vision applications that we discuss in Sec. 7.3.

Undeniably, the core strengths of the frameworks presented in this thesis are their direct use with event cameras and their unsupervised learning with STDP rules. This thesis thus highlights the versatility of brain-inspired learning rules, and their potential in computer vision. Furthermore, with new spiking models being built based on predictive coding ideas, we hope that our models will inspire future generations of neuromorphic information processing systems.

Overall, our studies have provided novel conceptual insights and they have highlighted the potential of efficient brain-inspired computing systems for technical applications.

7.2 Perspectives on improvements

7.2.1 Neurobiological aspects

Neural computation We made a number of simplifications in our frameworks compared to biological computations. For instance, neurons in both PCL and SPRIER do not respect Dale’s law, which poses a number of limitations on neural computations as could be seen with the SPRIER network. While the violation of Dale’s law could appear as a strength in the context of neuromorphic implementations (as it reduces the number of neurons), it still reduces the capabilities of the models in the reproduction of brain-like response patterns, and in the guiding of future biological experiments. For this reason, we believe that the networks could be improved by respecting Dale’s law. Additionally, to address the issues of biological plausibility observed within the SPRIER network with its plasticity mechanisms, a potential improvement that might allow the removal of those mechanisms could be the implementation of more realistic conductance-based synapses [57]. Such synapses could effectively allow excitatory inputs to be more effective when a neuron is hyperpolarized and inhibitory inputs to be more effective when a neuron is depolarized. This type of model might be able to alleviate some of the stability issues

faced when working with networks that have recurrent excitation. Thus, this appears to be an interesting approach to improve the networks' biological plausibility.

Feedback effect In our simulations, inhibitory feedback is instantaneous and thus lead to information loss the more spikes are suppressed. Though STDP ends up keeping only the most unpredictable spikes, information is still lost. To solve this issue, it would be interesting to mimic more biological processing by first accumulating sufficient evidence [49], [259] (allowing a sufficient number of spikes to happen) before letting feedback remove spikes through inhibition. We could for instance model this effect with the activation of feedback based on the “level” of information present in the neural code (e.g. a threshold on the number of spikes). Another alternative could be to allow this accumulation of information by implementing delays in our feedback connections, at the price of additional computational cost.

Deep PCL pooling connectivity In our studies on deeper instances of the PCL network, we observed that a one-to-one feature mapping from one area to the other allowed neurons to develop spatial invariance. It is unlikely that the brain has such a precise connectivity, so an improvement could perhaps be to consider groups of neurons representing “similar” features. Crucially, in the visual cortex of mammals, the columnar organization creates a feature space that would allow features to be projected according to their similarities. We believe that this one-to-one connection can be thus improved by estimating the distance between features of our PCL network, and project groups of features together, to perform pooling and achieve spatial invariance. An idea to define such a distance could be to modify the PCL network's local lateral inhibition such that an organized feature space is explicitly enforced. Neurons that would be “close” in feature space would be able to exchange only weak local lateral inhibition, whereas neurons that are “far” in feature space would be able to strongly inhibit each other. We believe that such an implementation would further improve the biological plausibility of our model while still allowing spatial invariance to arise.

7.2.2 Application aspects

In all of our simulations, to test the network on downstream tasks, we had to use an extra classifier (an SVM) to then classify the spiking patterns. While it is indeed thought that different neurons of the IT cortex spike together to represent inputs [123], from an application standpoint, it is unpractical to have different modules for the realization of a task. An improvement for the framework could thus be to implement such a supervised aspect directly in the spiking neural network, for instance with R-STDP [168]. Interestingly, such R-STDP could further be used to

learn feedback pathways, and might avoid strong information loss. Essentially, an end-to-end learner would significantly improve the current framework and further motivate the use of brain-inspired networks.

7.3 Perspectives on future applications

Extension of PCL to motion learning We showed that PCL could be used to learn hierarchical representations and thus functionally model the ventral stream. We are also interested in evaluating if when stimulated with moving inputs, the PCL network would be able to learn different type of motions, as seen in the dorsal stream. This type of study could show much interest in the context of autonomous robots, notably for the estimation of ego-motion as well as object tracking.

Evaluation of PCL and SPRIER on additional neuromorphic datasets It would also be interesting and add much more depth to the results if we validated further the results on invariance learning but also memory effects on additional and more complex event-based datasets. This would give a larger basis for comparison of our work with other models and would allow us to test if our approach generalizes to a wide variety of datasets. This could enable to better evaluate if our frameworks could compete more with other classification frameworks.

Extension to AEC and robotics As part of the AEC framework, future applications of this work include an extension to AEC. Preliminary results had been previously obtained by Barbier et al. [19] but did not account for fully learnable synapses, deep networks as well as recurrent excitation and synaptic delays. Combining the PCL and SPRIER networks with the AEC framework would be interesting in two respects: 1) neurobiological, where the response patterns of the network could be evaluated regarding their similarity with brain responses across visual and motor cortex, 2) robotics, where the AEC framework could provide a conceptual basis for building self-calibrating and energy-efficient active perception systems for, e.g., mobile robots.

Neuromorphic chip implementation Considering the conceptual feasibility of the implementations of our networks on neuromorphic chips, we would be interested in assessing directly the energy gain of our networks and demonstrate the efficiency of brain-inspired computing systems on downstream tasks. In the future, we hope

to implement the AEC framework on neuromorphic chips and apply it in different scenarios including research on mobile robots.

Appendices

APPENDIX A

SUPPLEMENTARY MATERIAL

Table A.1: Influence of the activation of top-down inhibition during the training of deep PCL networks on the cMNIST-ev dataset. Classification was done however by deactivating the top-down inhibition of the networks. Results show averages on 2 network simulations that were each trained from scratch using random initialization seeds. Errors on the other hand denote the standard error. The networks were presented in Chapter 5 and their architectures consist of the 3-layer PCL network of case (c) and of the 4-layer PCL network with residual connections activated. Results show that although the top-down inhibition is deactivated during this classification task, its presence during learning enhances the quality of the learnt representations as classification accuracy increases for networks trained with top-down inhibition.

	3-layer PCL network		4-layer PCL network	
	Trained with 0% top-down	Trained with 100% top-down	Trained with 0% top-down	Trained with 100% top-down
Median number of spikes	L1: 605 ± 1	L1: 591 ± 2	L1: 1602 ± 6	L1: 1665 ± 6
	L2: 186 ± 3	L2: 196 ± 0	L2: 1018 ± 8	L2: 1057 ± 14
	L3: 69 ± 1	L3: 77 ± 1	L3: 804 ± 29	L3: 741 ± 3
	All: 860 ± 4	All: 866 ± 1	L4: 214 ± 15	L4: 228 ± 2
			All: 3596 ± 57	All: 3696 ± 5
Accuracy	L1: $32.25 \pm 0.73\%$	L1: $31.28 \pm 1.26\%$	L1: $43.87 \pm 0.14\%$	L1: $47.81 \pm 0.53\%$
	L2: $40.39 \pm 0.43\%$	L2: $44.42 \pm 0.2\%$	L2: $62.7 \pm 0.37\%$	L2: $67.14 \pm 0.3\%$
	L3: $48.56 \pm 0.08\%$	L3: $53.19 \pm 0.53\%$	L3: $62.96 \pm 0.76\%$	L3: $68.03 \pm 0.38\%$
	All: 49.42 ± 0.06	All: $52.45 \pm 0.1\%$	L4: $65.92 \pm 0.1\%$	L4: $71.25 \pm 0.1\%$
			All: $72.34 \pm 0.47\%$	All: $76.7 \pm 0.49\%$

PUBLICATIONS AND COMMUNICATIONS

The contributions of this thesis have been published in an international conference, submitted to two international journals, and presented in two talks and two posters.

International Proceedings and Journals

Work related to the first contribution on the review on spiking predictive coding is currently undergoing review in a peer-reviewed journal. It has been published as an arXiv preprint

[171] A. W. N’dri *et al.*, *Predictive coding with spiking neural networks: A survey*, 2024

Work related to the second contribution on *Predictive Coding Light* has been published as

[173] A. W. N’dri *et al.*, “Predictive Coding Light: learning compact visual codes by combining excitatory and inhibitory spike timing-dependent plasticity*,” *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, vol. 00, pp. 3997–4006, 2023 **(accepted for a poster presentation)**

[172] A. W. N’dri *et al.*, “Predictive coding light,” *Nature Communications*, vol. 16, no. 1, p. 8880, 2025

Work related to our extension of the Predictive Coding Light network to deeper architectures as well as our third contribution on *Spiking Predictive Recurrent Inhibitory and Excitatory Recall* have not been submitted to journals thus far.

Proceedings & Workshops without act

Other communications of this work include three talks as

Antony W. N'dri, “Predictive Coding Light”, *Neuromorphic Symposium, Paris*, September, 2025.

Antony W. N'dri, “Combining various types of spike timing-dependent plasticity to learn efficient neural codes”, *Spiking Neural networks as Universal Function Approximators Workshop, Online* (SNUFA Workshop), November, 2023.

Antony W. N'dri, “Combining various types of spike timing-dependent plasticity to learn efficient neural codes”, *Journée GDR ISIS Caméras à événements appliquées à la robotique, Paris* (GDR ISIS), November, 2023.

Lastly, this thesis work has also been presented include three poster presentations

Antony W. N'dri, “Predictive Coding Light”, *Bernstein Conference, Frankfurt*, October, 2025. – poster presented by Jochen Triesch

Antony W. N'dri, “Event-based visual perception learning”, *Journée Scientifique de l'École Doctorale Sciences Pour l'Ingénieur* (JS-EDSPI), June, 2024.

Antony W. N'dri, “Event-based visual perception learning”, *Journée Scientifique du programme DATA* (JS-DATA), June, 2024.

RÉSUMÉ ÉTENDU EN FRANÇAIS

Introduction	139
C.1 Motivation	139
C.2 Contexte	142
C.3 Contributions	143
C.4 Résumé des chapitres	145

Introduction

Cette dernière partie de ce manuscrit résume de manière substantielle en **Français** les travaux réalisés pendant cette thèse. Dans un premier temps, nous détaillerons les motivations de ces travaux de thèse. Ensuite nous donnerons son contexte puis les contributions de cette thèse. Enfin, nous résumerons brièvement le contenu des différents chapitres de cette thèse.

C.1 Motivation

Le cerveau est un organe impressionnant qui a été optimisé par des millions d'années d'évolution. Son développement a été un point clé de la survie de l'espèce humaine dans la mesure où il nous a permis de percevoir le monde à travers nos différents sens, et ainsi nous a donné la faculté d'adapter notre comportement selon nos expériences antérieures. A titre d'exemple, dans le cas précis de la vision, bien que la rétine nous permette de voir le monde en captant la lumière pour la convertir

en signaux qui seraient interprétés par le cerveau, la rétine à elle seule ne peut expliquer la perception. Sans aucun traitement additionnel des signaux de la rétine, nous ne serions pas capable de reconnaître, segmenter ou encore suivre des objets – c’est à dire dire, nous ne serions pas capable de *percevoir* le monde. Cela s’applique non seulement à la vision mais aussi aux autres sens tels que l’audition, le toucher et l’odorat. Le cerveau est essentiellement l’organe clé qui exploite les stimuli sensoriels pour permettre la perception. Fondamentalement à travers les facultés de perception, nous sommes capables d’interpréter le monde et interagir avec lui via nos actions. Voyant l’importance de ces éléments dans notre survie et notre vie de tous les jours, les scientifiques ont questionné pendant des siècles, voire millénaires, les mécanismes neuronaux derrière l’émergence de la perception.

Les neurones sont des cellules qui sont trouvées dans le cerveau et sont l’élément principal de “calcul” par lequel ce dernier traite les données. Comprendre comment les calculs neuronaux sont effectués nous aiderait à mieux appréhender comment le cerveau génère une perception à partir de stimuli sensoriels. Différentes théories de codage sensoriel et perceptuel tels que le codage efficace [20] et le codage prédictif [195] ont été introduites depuis des décennies maintenant et peuvent être testées via des simulations de réseaux de neurones sur ordinateur. Ces simulations permettraient de guider la recherche en neuroscience et donc d’améliorer notre compréhension sur les mécanismes derrière l’émergence de la perception. De plus, la science est récemment entrée dans une nouvelle ère dans laquelle le développement de systèmes intelligents est au centre de la recherche de divers domaines (e.g. science cognitive, biotechnologie, imagerie médicale, informatique, robotique). De ce fait, pour construire de nouveaux systèmes intelligents, il fait sens de prendre inspiration du système le plus intelligent connu à l’heure actuelle, soit le cerveau. Le développement de tels systèmes bio-inspirés devrait conduire à de nouvelles avancées scientifiques et permettre la construction de nouvelles machines qui surpassent celles déjà existantes.

L’état de l’art actuel en termes de modèles de “perception machine” consiste principalement en des méthodes d’apprentissage profond [136]. Ces dernières trouvent leurs origines dans des modèles de bio-mimétisme qui essaient de simuler le processus du cerveau. L’apprentissage profond a donc révolutionné les algorithmes de perception et est utilisé via des logiciels pour assister les êtres humains dans des tâches cognitives (e.g. chatbots, logiciels de surveillance, diagnostic de cancer, etc.) mais aussi déployé pour une aide directe dans des tâches physiques (e.g. chirurgie assistée par la robotique, robots sauveteurs, prothèses robotiques, etc.). A partir de la combinaison d’algorithmes de commande et d’apprentissage profond, les machines atteignent désormais de nouveaux sommets. Cependant, l’apprentissage profond souffre de problèmes bien connus qui sont les suivants:

- Consommation énergétique importante: Les nouveaux modèles sont basés sur des architectures complexes de réseaux qui consomment beaucoup d'énergie pour l'entraînement comme pour l'inférence [189].
- Manque d'explicabilité: Les techniques d'apprentissage profond doivent leurs résultats impressionnants à l'algorithme de rétropropagation qui minimise une fonction de coût. Le comportement du réseau de neurone suite à la minimisation de cette fonction de coût n'est malheureusement pas directement identifiable mathématiquement, ce qui rend complexe l'interprétabilité des réseaux de neurones profonds.
- Manque d'adaptabilité: La plupart des modèles sont entraînés sur des tâches précises. Cela entraîne un entraînement supervisé sur chaque tâche plutôt qu'un entraînement de bout en bout adaptatif (et sans supervision) du réseau. Bien qu'un nouveau type de modèle d'apprentissage profond appelé modèles de fondation [35] a récemment été proposé pour aborder ce problème, cela requiert toujours un fine-tuning du réseau sur des tâches individuelles.

Compte tenu de ces limitations, les algorithmes de perception machine sont à un stade où il serait intéressant d'augmenter le niveau d'inspiration biologique. L'une des différences les plus distinctes entre le cerveau et l'apprentissage profond est la dépendance aux signaux continus dans l'apprentissage profond, par opposition aux signaux impulsionnels asynchrones observés dans le cerveau. S'inspirer davantage du cerveau exigerait donc de changer les neurones artificiels actuels vers le domaine impulsionnels avec des réseaux neuronaux impulsionnels [234]. Bien qu'un tel changement de calcul soit complexe, les avantages peuvent l'emporter sur les inconvénients. Par exemple, le cerveau n'utilise qu'environ 20 W d'énergie [199]. Enfin, le cerveau est hautement adaptatif car il peut effectuer une grande variété de tâches sans supervision. Essentiellement, la recherche sur les approches inspirées du cerveau conduirait à long terme à des améliorations significatives des capacités des machines actuelles, tout en améliorant notre compréhension des mécanismes neuronaux derrière la perception. Différents laboratoires de recherche ont déjà commencé à travailler sur de tels modèles de réseaux neuronaux à impulsions inspirés du cerveau [171] et des avancées notables ont également été réalisées sur le matériel inspiré du cerveau avec la création de la "rétine de silicone" communément appelée maintenant caméra d'événements [80]. Une telle recherche parallèle, mais complémentaire, en logiciel et en matériel est cruciale pour que de nouvelles machines inspirées du cerveau voient le jour à l'avenir.

Travailler avec des neurones impulsionnels implique de considérer le calcul d'une manière entièrement différente. Le cerveau a tendance à utiliser deux types différents de processus de communication neuronale : excitateur et inhibiteur. Les neurones excitateurs, comme leur nom l'indique, exciteraient d'autres neurones — c'est-à-dire,

augmenteraient leurs réponses. D'autre part, les neurones inhibiteurs inhiberaient d'autres neurones — c'est-à-dire, réduiraient leurs réponses. Afin de construire des systèmes de neurones à impulsions capables de perception, il est nécessaire d'étudier comment de tels calculs excitateurs-inhibiteurs donnent naissance à des propriétés de réponse similaires à celles du cerveau. La plasticité fonction du temps d'occurrence des impulsions (STDP en anglais) [27] est un bon candidat pour l'émergence de tels motifs perceptuels, car ce mécanisme de plasticité synaptique a été largement observé dans différentes zones corticales. Cependant, l'utilisation de la STDP et des réseaux excitateurs-inhibiteurs est soumise à de multiples questions importantes telles que : 1) le rôle de chaque connexion dans l'émergence de la perception, 2) l'architecture neuronale requise, 3) l'équilibre entre excitation et inhibition pour une activité neuronale stable. Bien que ces questions aient intrigué les chercheurs en neurosciences depuis longtemps, ce n'est qu'au cours de la dernière décennie que l'interaction entre la plasticité excitatrice et inhibitrice a commencé à être étudiée dans des réseaux de neurones à impulsions [11], [38], [99], [245]. En prenant l'exemple de la vision, l'utilisation de l'inhibition pour l'apprentissage perceptuel (par exemple, des champs récepteurs semblables à ceux du cortex visuel) dans des réseaux neuronaux impulsionnels profonds a montré des résultats prometteurs [121], [122], [155]. Ces travaux utilisent cependant uniquement la plasticité excitatrice et maintiennent l'inhibition fixe. De plus, le cerveau démontre une connectivité récurrente et de descendante remarquable, qui n'a donc pas été évaluée dans les réseaux de neurones à impulsions pour la perception visuelle.

Dans cette thèse, nous nous intéressons donc au développement de réseaux de neurones à impulsions inspirés du cortex visuel qui apprennent à effectuer la perception visuelle. Nos résultats suggèrent que différents schémas de réponse cérébrale trouvent leur origine dans le STDP excitateur et inhibiteur, ainsi que dans différentes connexions récurrentes. En plus de ces observations neuroscientifiques, nous démontrons l'usage de tels mécanismes bio-inspirés sur des tâches appliquées telles que la reconnaissance d'objets et de mouvements, et nous mettons en évidence les avantages des connexions récurrentes dans le traitement de l'information et le codage énergétique du cerveau. Notre framework est conçu pour être compatible avec le matériel neuromorphique qui imite également le cerveau, et propose ainsi une autre perspective sur les systèmes informatiques artificiels inspirés du cerveau pour la perception des machines.

C.2 Contexte

Au cours des dernières années, un modèle nommé « Codage Actif Efficace » (AEC) a été développé [63]. Ce modèle repose sur l'hypothèse que les systèmes biologiques

apprennent des comportements moteurs, c'est-à-dire des actions qui permettent un codage sensoriel efficace. Un codage sensoriel efficace pointerait par exemple vers un petit nombre de neurones actifs associés à une faible activité de réponse, de sorte que la transmission d'informations soit optimale. Les modèles d'AEC visent à apprendre à des agents autonomes à effectuer des mouvements oculaires de poursuite fluide ainsi que la vergence oculaire pour la vision binoculaire. Ces modèles ont cependant été développés avec des réseaux de neurones artificiels standard et non avec des réseaux de neurones à impulsions.

Le travail de cette thèse vise à déplacer le cadre AEC vers le domaine des réseaux de neurones à impulsions en utilisant les entrées d'une caméra à événements. Une première thèse avait été réalisée sur le sujet par Barbier [19] d'octobre 2019 à janvier 2023. Dans cette thèse, une bibliothèque de réseaux neuronaux à impulsions, des résultats préliminaires sur le développement de champs réceptifs de type V1 ainsi que la première version d'un modèle AEC à impulsions [19] avaient été proposés.

Il est important de noter que cette thèse constitue la deuxième thèse sur ce corpus de travail et s'appuie sur les travaux de Barbier, en ajoutant de nouvelles fonctionnalités à la bibliothèque de réseaux neuronaux à impulsions ainsi qu'en testant et en validant d'autres hypothèses (ici, le codage prédictif) de la perception cérébrale dans la vision en utilisant des stimuli d'images naturelles basés sur des événements ainsi que des ensembles de données de vision neuromorphique. Ce travail démontre, dans un réseau neuronal impulsif, la validité neuroscientifique ainsi que les capacités de vision par ordinateur sur lesquelles repose l'hypothèse AEC. Cette thèse pose donc des bases significatives pour les recherches futures sur l'AEC impulsif pour la robotique avec caméra événementielle.

C.3 Contributions

Dans cette thèse, nous proposons deux modèles de réseaux de neurones à impulsions : le modèle *Predictive Coding Light* et le modèle *Spiking Predictive Recurrent Inhibitory and Excitatory Recall*. Nous montrons que différentes propriétés de réponse similaires à celles du cerveau émergent au sein de ces modèles après exposition à des stimuli d'images naturelles basés sur des événements ainsi qu'à des séquences spatio-temporelles basées sur des événements. Toutes les connexions synaptiques des deux modèles sont apprises en utilisant des règles de plasticité dépendantes du temps des pics, et nous montrons que les connexions descendantes (feedback) couplées à l'inhibition et à l'excitation plastiques permettent le développement de la perception visuelle et la gestion des ressources énergétiques dans nos réseaux inspirés du cerveau.

Nos contributions sont les suivantes :

Revue des modèles de codage prédictif impulsionnels - *Pré-publié sur arXiv [171] et actuellement en cours de révision dans un journal.*

Nous passons en revue différents modèles de codage prédictif impulsionnel [171] qui ciblaient tous des applications très différentes et étaient ainsi dispersés dans différents domaines de recherche, ce qui rendait l'identification des questions de recherche ouvertes compliquée. Dans cette revue, nous avons donc examiné les efforts de recherche historiques et actuels sur la réalisation du codage prédictif avec des réseaux de neurones à impulsions, en les séparant selon une notion commune qui revenait parmi toutes les méthodes de codage prédictif — l'erreur de prédiction. Cette revue représente, à notre avis, une nouvelle contribution précieuse à la communauté pour guider et encourager les recherches futures sur les réseaux neuronaux à impulsions pour le codage prédictif.

Le modèle Predictive Coding Light - *Publié partiellement sous [173] et [172].*

Avec Predictive Coding Light, nous remettons en question les modèles actuels de codage prédictif et proposons un nouveau modèle entièrement plastique de codage prédictif impulsionnel pour la perception visuelle à faible énergie. Predictive Coding Light apprend les propriétés de réponse des cellules simples et des cellules complexes après avoir été entraîné sur des images naturelles en utilisant la STDP. Predictive Coding Light présente en outre des effets de suppression des contours (surround suppression), des contours orientés (orientation-tuned suppression) et de superposition d'orientations (cross-orientation suppression), qui sont largement acceptés comme des signatures de codes efficaces dans les réseaux de codage prédictif [23]. De plus, Predictive Coding Light met en évidence l'effet des connexions inhibitrices récurrentes et descendantes dans l'apprentissage de représentations visuelles efficaces similaires au V1, et dans la réalisation de tâches appliquées telles que la classification de chiffres et de gestes. Ces résultats soulignent notamment le rôle de l'inhibition dans l'économie d'énergie en éliminant les informations hautement prévisibles dans le code neuronal. Enfin, Predictive Coding Light apprend des représentations visuelles invariantes dans des réseaux profonds de différentes couches et démontre l'influence de différents schémas de projection dans l'émergence de l'invariance lors de l'apprentissage des hiérarchies de caractéristiques.

Le modèle Spiking Predictive Recurrent Inhibitory and Excitatory Recall

Avec le modèle de Spiking Predictive Recurrent Inhibitory and Excitatory Recall, nous étendons le modèle Predictive Coding Light pour inclure des délais synaptiques, une excitation latérale distante et une excitation descendante dans son architecture. Nous introduisons de nouveaux mécanismes pour assurer la stabilité du réseau et reproduire les découvertes biologiques sur l'apprentissage des séquences spatiotemporelles de barres visuelles, ainsi que pour tester le réseau sur des gestes afin de valider l'apprentissage de séquences plus complexes. L'apprentissage des séquences spatiotemporelles dans le réseau Spiking Predictive Recurrent Inhibitory and Excitatory Recall est attribuable aux STDP inhibitrice et excitatrice et aux délais synaptiques. Nos résultats donnent un aperçu de la manière dont les connexions excitatrices récurrentes pourraient conduire à la formation d'une mémoire de travail et d'une mémoire associative dans le cerveau.

Différentes publications, soumissions pour publication, communications orales dans des ateliers internationaux et nationaux ainsi que la participation à des ateliers internationaux ont été réalisées au cours de cette thèse. Les détails de ces communications sont donnés dans l'Annexe B. Notre code source (C++ et Python) et nos ensembles de données sont disponibles sur <https://github.com/comsee-research/Predictive-Coding-Light> et <https://zenodo.org/records/16813455>

C.4 Résumé des chapitres

Le chapitre 2 offre un aperçu général des différents concepts abordés dans cette thèse. Le chapitre introduit d'abord les concepts de la vision neuromorphique en couvrant les connaissances neuroscientifiques nécessaires à la mise en œuvre de réseaux de neurones impulsionnels basés sur la vision et présente brièvement les capteurs et le matériel neuromorphiques. Deuxièmement, le chapitre détaille les concepts de la théorie du codage prédictif en introduisant son principe, son lien avec l'hypothèse du codage efficace ainsi que la relation entre le codage prédictif et le cerveau.

Le chapitre 3 passe en revue différents modèles de codage prédictif avec des réseaux de neurones à impulsions. Il propose une séparation des modèles de codage prédictif impulsionnels en trois catégories différentes de travaux selon la manière dont ils encodent les erreurs de prédiction : 1) en utilisant des neurones explicites de prédiction d'erreur, 2) en utilisant le potentiel de membrane pour encoder les erreurs de prédiction et 3) en utilisant un encodage implicite des erreurs de prédiction dans les activités neuronales.

Le chapitre 4 présente notre framework de réseau de neurones impulsionnels *Predictive Coding Light*. Le chapitre décrit comment les plasticités inhibitrice et excitatrice via STDP peuvent être utilisées pour apprendre des représentations similaires à celles des cellules simples et complexes observées dans le cerveau. Il compare les résultats aux découvertes biologiques faites dans le V1 des mammifères. Il détaille les conditions dans lesquelles les cellules simples et les cellules complexes émergent et compare l'effet de différentes règles de STDP dans l'émergence de caractéristiques semblables à celles du cortex visuel. Ensuite, le chapitre démontre que les effets de suppression basée sur les contours (surround suppression), de suppression basée sur l'orientation (orientation-tuned suppression) et de suppression basée sur une superposition d'orientation (cross-orientation suppression) observés dans le V1 des mammifères émergent tous dans le réseau Predictive Coding Light. Enfin, le chapitre montre comment un réseau Predictive Coding Light peut être utilisé pour effectuer la classification d'objets et montrer les effets de la suppression, qui réduit le nombre de pics (et donc la consommation d'énergie) dans le réseau tout en n'entraînant qu'une légère baisse des précisions de classification.

Le chapitre 5 évalue différentes instances plus profondes du réseau Predictive Coding Light sur l'apprentissage de représentations invariantes. Le chapitre présente l'effet de différents types de schémas de projection, par exemple les connexions résiduelles ou les projections de caractéristiques de type "pooling" un à un, et montre que, selon la profondeur d'une couche ainsi que la zone d'entrée de son réseau, un mélange de schémas de projection est nécessaire pour apprendre des représentations visuelles invariantes dans un réseau neuronal impulsionnel hiérarchique.

Le chapitre 6 décrit un nouveau cadre appelé le réseau *Spiking Predictive Recurrent Inhibitory and Excitatory Recall* qui s'appuie sur le réseau Predictive Coding Light. Ce réseau propose une excitation latérale distante, une excitation descendante, des délais synaptiques et de nouveaux mécanismes d'apprentissage synaptique pour apprendre en utilisant des réseaux de neurones impulsionnels via la STDP, pour développer une mémoire de travail. Le chapitre montre comment différentes approches d'apprentissage avec des délais synaptiques et une excitation récurrente affectent l'apprentissage sensoriel. Ensuite, il présente les résultats dans le cadre de la reproduction de différentes découvertes biologiques sur l'apprentissage des séquences temporelles réalisées dans le V1 de souris. Enfin, le chapitre montre que le réseau peut également être utilisé pour apprendre des séquences spatiotemporelles complexes telles que des gestes.

Le chapitre 7 passe en revue les différentes contributions de cette thèse avec les cadres : le réseau *Predictive Coding Light* et son extension le réseau *Spiking Predictive*

Recurrent Inhibitory and Excitatory Recall. Le chapitre résume comment les deux cadres peuvent être utilisés pour reproduire de nombreuses propriétés de réponse biologiques ainsi que pour effectuer des tâches en aval, en utilisant une connectivité inspirée du cerveau et des mécanismes d'apprentissage. Enfin, le chapitre énumère un certain nombre de perspectives d'amélioration ainsi que les applications futures de cette thèse.

BIBLIOGRAPHY

- [1] E. H. Adelson and J. R. Bergen, “Spatiotemporal energy models for the perception of motion,” *Josa a*, vol. 2, no. 2, pp. 284–299, 1985.
- [2] P. Ahirwal, “Correlation of neurochemical profile to the retinal fundus imaging features: Pilot experiment,” *Master’s thesis*, 2020.
- [3] L. Aitchison and M. Lengyel, “With or without you: Predictive coding and bayesian inference in the brain,” *Current opinion in neurobiology*, vol. 46, pp. 219–227, 2017.
- [4] A. Alemi, C. Machens, S. Denève, and J.-J. Slotine, “Learning arbitrary dynamics in efficient, balanced spiking networks using local plasticity rules,” *arXiv:1705.08026 [q-bio]*, Aug. 2017.
- [5] A. Alink, C. M. Schwiedrzik, A. Kohler, W. Singer, and L. Muckli, “Stimulus predictability reduces responses in primary visual cortex,” *Journal of Neuroscience*, vol. 30, no. 8, pp. 2960–2966, 2010.
- [6] N. Alonso and E. Neftci, “Tightening the biological constraints on gradient-based predictive coding,” in *International Conference on Neuromorphic Systems 2021*, 2021, pp. 1–9.
- [7] A. Amir, B. Taba, D. Berg, *et al.*, “A low power, fully event-based gesture recognition system,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7243–7252.
- [8] A. Arkhipov, N. W. Gouwens, Y. N. Billeh, *et al.*, “Visual physiology of the layer 4 cortical circuit in silico,” *PLoS computational biology*, vol. 14, no. 11, e1006535, 2018.
- [9] T. Asabuki and C. Clopath, “Embedding stochastic dynamics of the environment in spontaneous activity by prediction-based plasticity,” *bioRxiv*, pp. 2023–05, 2023.
- [10] T. Asabuki and T. Fukai, “Predictive learning rules generate a cortical-like replay of probabilistic sensory experiences,” *eLife*, vol. 13, 2024.
- [11] T. Asabuki, C. J. Gillon, and C. Clopath, “Learning predictive signals within a local recurrent circuit,” *bioRxiv*, pp. 2023–06, 2023.

- [12] D. Attwell and S. B. Laughlin, “An energy budget for signaling in the grey matter of the brain,” *Journal of Cerebral Blood Flow & Metabolism*, vol. 21, no. 10, pp. 1133–1145, 2001.
- [13] J. F. Baker, S. E. Petersen, W. T. Newsome, and J. M. Allman, “Visual response properties of neurons in four extrastriate visual areas of the owl monkey (*aotus trivirgatus*): A quantitative comparison of medial, dorsomedial, dorsolateral, and middle temporal areas,” *Journal of neurophysiology*, vol. 45, no. 3, pp. 397–416, 1981.
- [14] D. H. Ballard, R. P. N. Rao, and Z. Zhang, “A single-spike model of predictive coding,” *Neurocomputing*, vol. 32, pp. 17–23, 2000.
- [15] O. Barak and M. Tsodyks, “Working models of working memory,” *Current opinion in neurobiology*, vol. 25, pp. 20–24, 2014.
- [16] T. Barbier, “Autonomous learning in a neuromorphic vision system: An active efficient coding spiking model applied to vision,” Ph.D. dissertation, Université Clermont Auvergne, 2023.
- [17] T. Barbier, C. Teulière, and J. Triesch, “Unsupervised learning of spatio-temporal receptive fields from an event-based vision sensor,” in *International Conference on Artificial Neural Networks*, Springer, 2020, pp. 622–633.
- [18] T. Barbier, C. Teulière, and J. Triesch, “A spiking neural network for active efficient coding,” *Frontiers in Robotics and AI*, vol. 11, p. 1435197, 2024.
- [19] T. Barbier, C. Teulière, and J. Triesch, “A spiking neural network for active efficient coding,” *Frontiers in Robotics and AI*, vol. 11, p. 1435197, 2025.
- [20] H. B. Barlow *et al.*, “Possible principles underlying the transformation of sensory messages,” *Sensory communication*, vol. 1, no. 01, pp. 217–233, 1961.
- [21] P. Barone, A. Batardiere, K. Knoblauch, and H. Kennedy, “Laminar distribution of neurons in extrastriate areas projecting to visual areas v1 and v4 correlates with the hierarchical rank and indicates the operation of a distance rule,” *Journal of Neuroscience*, vol. 20, no. 9, pp. 3263–3281, 2000.
- [22] D. G. T. Barrett, S. Denève, and C. K. Machens, “Optimal compensation for neuron loss,” *Computational and Systems Neuroscience (CoSyNe)*, Salt Lake City, 2014.
- [23] A. M. Bastos, W. M. Usrey, R. A. Adams, G. R. Mangun, P. Fries, and K. J. Friston, “Canonical microcircuits for predictive coding,” *Neuron*, vol. 76, no. 4, pp. 695–711, 2012.
- [24] A. J. Bell and T. J. Sejnowski, “An information-maximization approach to blind separation and blind deconvolution,” *Neural computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

-
- [25] A. J. Bell and T. J. Sejnowski, “Fast blind separation based on information theory,” in *Proc. Intern. Symp. on Nonlinear Theory and Applications, Las Vegas*, 1995.
 - [26] A. J. Bell and T. J. Sejnowski, “The “independent components” of natural scenes are edge filters,” *Vision research*, vol. 37, no. 23, pp. 3327–3338, 1997.
 - [27] G.-q. Bi and M.-m. Poo, “Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type,” *Journal of neuroscience*, vol. 18, no. 24, pp. 10 464–10 472, 1998.
 - [28] Y. Bi and Y. Andreopoulos, “Pix2nvs: Parameterized conversion of pixel-domain video frames to neuromorphic vision streams,” in *2017 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2017, pp. 1990–1994.
 - [29] Y. N. Billeh, B. Cai, S. L. Gratiy, *et al.*, “Systematic integration of structural and functional data into multi-scale models of mouse primary visual cortex,” *Neuron*, vol. 106, no. 3, pp. 388–403, 2020.
 - [30] T. V. Bliss and A. R. Gardner-Medwin, “Long-lasting potentiation of synaptic transmission in the dentate area of the unanaesthetized rabbit following stimulation of the perforant path,” *The Journal of physiology*, vol. 232, no. 2, p. 357, 1973.
 - [31] T. V. Bliss and T. Lømo, “Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path,” *The Journal of physiology*, vol. 232, no. 2, pp. 331–356, 1973.
 - [32] M. Boerlin and S. Denève, “Spike-based population coding and working memory,” *PLoS computational biology*, vol. 7, no. 2, e1001080, 2011.
 - [33] M. Boerlin, C. K. Machens, and S. Denève, “Predictive Coding of Dynamical Variables in Balanced Spiking Networks,” *PLoS Computational Biology*, vol. 9, no. 11, e1003258, 2013.
 - [34] R. Bogacz, “A tutorial on the free-energy framework for modelling perception and learning,” *Journal of mathematical psychology*, vol. 76, pp. 198–211, 2017.
 - [35] R. Bommasani, D. A. Hudson, E. Adeli, *et al.*, “On the opportunities and risks of foundation models,” *arXiv preprint arXiv:2108.07258*, 2021.
 - [36] M. Booth and E. T. Rolls, “View-invariant representations of familiar objects by neurons in the inferior temporal visual cortex.,” *Cerebral cortex (New York, NY: 1991)*, vol. 8, no. 6, pp. 510–523, 1998.
 - [37] R. Bourdoukan, D. G. T. Barrett, C. K. Machens, and S. Denève, “Learning optimal spike-based representations,” in *Advances in Neural Information Processing Systems 25*, P. Bartlett, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., 2012, pp. 2294–2302.

- [38] W. Brendel, R. Bourdoukan, P. Vertechi, C. K. Machens, and S. Denève, “Learning to represent signals spike by spike,” *PLoS computational biology*, vol. 16, no. 3, e1007692, 2020.
- [39] R. Brette and W. Gerstner, “Adaptive exponential integrate-and-fire model as an effective description of neuronal activity,” *Journal of neurophysiology*, vol. 94, no. 5, pp. 3637–3642, 2005.
- [40] N. Calaim, F. A. Dehmelt, P. J. Gonçalves, and C. K. Machens, “The geometry of robustness in spiking neural networks,” *eLife*, vol. 11, M. Meister, Ed., e73276, May 2022.
- [41] N. Caporale and Y. Dan, “Spike timing–dependent plasticity: A hebbian learning rule,” *Annu. Rev. Neurosci.*, vol. 31, no. 1, pp. 25–46, 2008.
- [42] B. Chakravarthi, A. A. Verma, K. Daniilidis, C. Fermuller, and Y. Yang, “Recent event camera innovations: A survey,” *arXiv preprint arXiv:2408.13627*, 2024.
- [43] D. V. Christensen, R. Dittmann, B. Linares-Barranco, *et al.*, “2022 roadmap on neuromorphic computing and engineering,” *Neuromorphic Computing and Engineering*, vol. 2, no. 2, p. 022 501, 2022.
- [44] P. Comon, “Independent component analysis, a new concept?” *Signal processing*, vol. 36, no. 3, pp. 287–314, 1994.
- [45] L. Cordone, B. Miramond, and S. Ferrante, “Learning from event cameras with sparse spiking convolutional neural networks,” in *2021 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2021, pp. 1–8.
- [46] E. N. Covic and S. M. Sherman, “Synaptic properties of connections between the primary and secondary auditory cortices in mice,” *Cerebral cortex*, vol. 21, no. 11, pp. 2425–2441, 2011.
- [47] C. L. Cox, W. Denk, D. W. Tank, and K. Svoboda, “Action potentials reliably invade axonal arbors of rat neocortical neurons,” *Proceedings of the National Academy of Sciences*, vol. 97, no. 17, pp. 9724–9728, 2000.
- [48] F. Crick and C. Koch, “Constraints on cortical and thalamic projections: The no-strong-loops hypothesis,” *Nature*, vol. 391, no. 6664, pp. 245–250, 1998.
- [49] Á. Darriba and F. Waszak, “Predictions through evidence accumulation over time,” *Scientific reports*, vol. 8, no. 1, p. 494, 2018.
- [50] P. Dayan and L. F. Abbott, *Theoretical neuroscience: computational and mathematical modeling of neural systems*. MIT press, 2005.
- [51] R. De Pasquale and S. M. Sherman, “Synaptic properties of corticocortical connections between the primary and secondary visual cortical areas in the mouse,” *Journal of Neuroscience*, vol. 31, no. 46, pp. 16 494–16 506, 2011.

-
- [52] S. Deneve, “Bayesian Spiking Neurons I: Inference,” *Neural Computation*, vol. 20, no. 1, pp. 91–117, 2008.
 - [53] S. Deneve, “Bayesian Spiking Neurons II: Learning,” *Neural Computation*, vol. 20, no. 1, pp. 118–145, 2008.
 - [54] S. Denève, “Bayesian inference in spiking neurons,” *Advances in neural information processing systems*, 2004.
 - [55] S. Denève and C. K. Machens, “Efficient codes and balanced networks,” *Nature neuroscience*, vol. 19, no. 3, pp. 375–82, 2016.
 - [56] R. Desimone, “Face-selective cells in the temporal cortex of monkeys,” *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 1–8, 1991.
 - [57] A. Destexhe, “Conductance-based integrate-and-fire models,” *Neural Computation*, vol. 9, no. 3, pp. 503–514, 1997.
 - [58] P. U. Diehl and M. Cook, “Unsupervised learning of digit recognition using spike-timing-dependent plasticity,” *Frontiers in computational neuroscience*, vol. 9, p. 99, 2015.
 - [59] R. J. Douglas, C. Koch, M. Mahowald, K. A. Martin, and H. H. Suarez, “Recurrent excitation in neocortical circuits,” *Science*, vol. 269, no. 5226, pp. 981–985, 1995.
 - [60] R. J. Douglas and K. Martin, “A functional microcircuit for cat visual cortex,” *The Journal of physiology*, vol. 440, no. 1, pp. 735–769, 1991.
 - [61] R. J. Douglas, K. A. Martin, and D. Whitteridge, “A canonical microcircuit for neocortex,” *Neural computation*, vol. 1, no. 4, pp. 480–488, 1989.
 - [62] K. van Driel, L. Rudelt, V. Priesemann, and F. A. Mikulasch, “Prediction mismatch responses arise as corrections of a predictive spiking code,” *bioRxiv*, pp. 2023–11, 2023.
 - [63] S. Eckmann, L. Klimmasch, B. E. Shi, and J. Triesch, “Active efficient coding explains the development of binocular vision and its failure in amblyopia,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 11, pp. 6156–6162, 2020.
 - [64] M. Evanusa, Y. Sandamirskaya, *et al.*, “Event-based attention and tracking on neuromorphic hardware,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.
 - [65] W. Fang, Z. Yu, Y. Chen, T. Huang, T. Masquelier, and Y. Tian, “Deep residual learning in spiking neural networks,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 21 056–21 069, 2021.
 - [66] D. E. Feldman, “Timing-based ltp and ltd at vertical inputs to layer ii/iii pyramidal cells in rat barrel cortex,” *Neuron*, vol. 27, no. 1, pp. 45–56, 2000.

- [67] D. J. Felleman and D. C. Van Essen, "Distributed hierarchical processing in the primate cerebral cortex.," *Cerebral cortex (New York, NY: 1991)*, vol. 1, no. 1, pp. 1–47, 1991.
- [68] A. Fiser, D. Mahringer, H. K. Oyibo, A. V. Petersen, M. Leinweber, and G. B. Keller, "Experience-dependent spatial expectations in mouse visual cortex," *Nature neuroscience*, vol. 19, no. 12, pp. 1658–1664, 2016.
- [69] K. Flügel, D. Coquelin, M. Weiel, C. Debus, A. Streit, and M. Götz, "Feed-forward optimization with delayed feedback for neural networks," *arXiv preprint arXiv:2304.13372*, 2023.
- [70] L. Fortuna and A. Buscarino, "Spiking neuron mathematical models: A compact overview," *Bioengineering*, vol. 10, no. 2, p. 174, 2023.
- [71] N. Fourcaud-Trocmé, D. Hansel, C. Van Vreeswijk, and N. Brunel, "How spike generation mechanisms determine the neuronal response to fluctuating inputs," *Journal of neuroscience*, vol. 23, no. 37, pp. 11 628–11 640, 2003.
- [72] N. Frémaux, H. Sprekeler, and W. Gerstner, "Functional requirements for reward-modulated spike-timing-dependent plasticity," *Journal of Neuroscience*, vol. 30, no. 40, pp. 13 326–13 337, 2010.
- [73] K. Friston, "Learning and inference in the brain," *Neural Networks*, vol. 16, no. 9, pp. 1325–1352, 2003.
- [74] K. Friston, "A theory of cortical responses," *Philosophical transactions of the Royal Society B: Biological sciences*, vol. 360, no. 1456, pp. 815–836, 2005.
- [75] K. Friston, "A theory of cortical responses," *Philosophical Transactions of the Royal Society B*, vol. 360, pp. 815–836, 1456 2005.
- [76] K. Friston, "The free-energy principle: A unified brain theory?" *Nature reviews neuroscience*, vol. 11, no. 2, pp. 127–138, 2010.
- [77] K. Friston, "Does predictive coding have a future?" *Nature neuroscience*, vol. 21, no. 8, pp. 1019–1021, 2018.
- [78] K. Friston, J. Kilner, and L. Harrison, "A free energy principle for the brain," *Journal of physiology-Paris*, vol. 100, no. 1-3, pp. 70–87, 2006.
- [79] D. Gabor, "Theory of communication. part 1: The analysis of information," *Journal of the Institution of Electrical Engineers-part III: radio and communication engineering*, vol. 93, no. 26, pp. 429–441, 1946.
- [80] G. Gallego, T. Delbrück, G. Orchard, *et al.*, "Event-based vision: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 1, pp. 154–180, 2020.
- [81] D. Gandolfi, F. M. Puglisi, G. M. Boiani, *et al.*, "Emergence of associative learning in a neuromorphic inference network," *Journal of Neural Engineering*, vol. 19, no. 3, p. 036 022, 2022.

-
- [82] J. P. Gavornik and M. F. Bear, “Learned spatiotemporal sequence recognition and prediction in primary visual cortex,” *Nature neuroscience*, vol. 17, no. 5, pp. 732–737, 2014.
 - [83] A. M. George, D. Banerjee, S. Dey, A. Mukherjee, and P. Balamurali, “A reservoir-based convolutional spiking neural network for gesture recognition from dvs input,” in *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020, pp. 1–9.
 - [84] M. Ghodrati, S.-M. Khaligh-Razavi, and S. R. Lehky, “Towards building a more complex view of the lateral geniculate nucleus: Recent advances in understanding its role,” *Progress in neurobiology*, vol. 156, pp. 214–255, 2017.
 - [85] C. D. Gilbert and T. N. Wiesel, “Morphology and intracortical projections of functionally characterised neurones in the cat visual cortex,” *Nature*, vol. 280, no. 5718, pp. 120–125, 1979.
 - [86] J. Gjorgjieva, C. Clopath, J. Audet, and J.-P. Pfister, “A triplet spike-timing-dependent plasticity model generalizes the bienenstock-cooper-munro rule to higher-order spatiotemporal correlations,” *Proceedings of the national academy of sciences*, vol. 108, no. 48, pp. 19 383–19 388, 2011.
 - [87] G. Goupy, P. Tirilly, and I. M. Bilasco, “Paired competing neurons improving stdp supervised local learning in spiking neural networks,” *Frontiers in Neuroscience*, vol. 18, p. 1 401 690, 2024.
 - [88] L. Graf, Z. Su, and G. Indiveri, “Echospike predictive plasticity: An online local learning rule for spiking neural networks,” *arXiv preprint arXiv:2405.13976*, 2024.
 - [89] A. Grimaldi and L. U. Perrinet, “Learning heterogeneous delays in a layer of spiking neurons for fast motion detection,” *Biological Cybernetics*, vol. 117, no. 4, pp. 373–387, 2023.
 - [90] R. Gütig and H. Sompolinsky, “The tempotron: A neuron that learns spike timing-based decisions,” *Nature neuroscience*, vol. 9, no. 3, pp. 420–428, 2006.
 - [91] N. Hagiwara, T. Kunimi, K. Ando, M. Akai-Kasaya, and T. Asai, “Design and evaluation of brain-inspired predictive coding networks based on the free-energy principle for novel neuromorphic hardware,” *Nonlinear Theory and Its Applications, IEICE*, vol. 15, no. 1, pp. 107–118, 2024.
 - [92] M. S. Halvagal and F. Zenke, “The combination of hebbian and predictive plasticity learns invariant object representations in deep sensory networks,” *Nature neuroscience*, vol. 26, no. 11, pp. 1906–1915, 2023.
 - [93] I. Hammouamri, I. Khalfaoui-Hassani, and T. Masquelier, “Learning delays in spiking neural networks using dilated convolutions with learnable spacings,” *arXiv preprint arXiv:2306.17670*, 2023.

- [94] L. M. Harrison, K. E. Stephan, G. Rees, and K. J. Friston, “Extra-classical receptive field effects measured in striate cortex with fmri,” *Neuroimage*, vol. 34, no. 3, pp. 1199–1208, 2007.
- [95] C. Hartmann, A. Lazar, B. Nessler, and J. Triesch, “Where’s the noise? key features of spontaneous activity and neural variability arise through learning in a deterministic network,” *PLoS computational biology*, vol. 11, no. 12, e1004640, 2015.
- [96] G. Hatfield, “Perception as unconscious inference,” in *Perception and the Physical World: Psychological and Philosophical Issues in Perception*, D. Heyer, Ed., John Wiley and Sons, 2002, pp. 113–143.
- [97] D. O. Hebb, “Organization of behavior. new york: Wiley,” *J. Clin. Psychol*, vol. 6, no. 3, pp. 335–307, 1949.
- [98] D. O. Hebb, *The organization of behavior: A neuropsychological theory*. Psychology press, 2005.
- [99] G. Hennequin, E. J. Agnes, and T. P. Vogels, “Inhibitory plasticity: Balance, control, and codependence,” *Annual review of neuroscience*, vol. 40, no. 1, pp. 557–579, 2017.
- [100] L. Hertäg and C. Clopath, “Prediction-error neurons in circuits with multiple neuron types: Formation, refinement, and functional implications,” *Proceedings of the National Academy of Sciences*, vol. 119, no. 13, e2115699119, 2022.
- [101] L. Hertäg and H. Sprekeler, “Learning prediction error neurons in a canonical interneuron circuit,” *Elife*, vol. 9, e57541, 2020.
- [102] A. L. Hodgkin and A. F. Huxley, “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *The Journal of physiology*, vol. 117, no. 4, p. 500, 1952.
- [103] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities,” *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.
- [104] Y. Hu, S.-C. Liu, and T. Delbruck, “V2e: From video frames to realistic dvs events,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1312–1321.
- [105] Y. Huang and R. P. Rao, “Predictive coding,” *Wiley Interdisciplinary Reviews: Cognitive Science*, vol. 2, no. 5, pp. 580–593, 2011.
- [106] A. Hyvärinen and E. Oja, “Independent component analysis: Algorithms and applications,” *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000.
- [107] G. Indiveri, B. Linares-Barranco, T. J. Hamilton, *et al.*, “Neuromorphic silicon neuron circuits,” *Frontiers in neuroscience*, vol. 5, p. 73, 2011.

-
- [108] L. R. Iyer and A. Basu, “Unsupervised learning of event-based image recordings using spike-timing-dependent plasticity,” in *2017 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2017, pp. 1840–1846.
 - [109] L. R. Iyer and Y. Chua, “Classifying neuromorphic datasets with tempotron and spike timing dependent plasticity,” in *2020 international joint conference on neural networks (IJCNN)*, IEEE, 2020, pp. 1–8.
 - [110] E. M. Izhikevich, “Resonate-and-fire neurons,” *Neural networks*, vol. 14, no. 6-7, pp. 883–894, 2001.
 - [111] E. M. Izhikevich, “Simple model of spiking neurons,” *IEEE Transactions on neural networks*, vol. 14, no. 6, pp. 1569–1572, 2003.
 - [112] E. M. Izhikevich, “Polychronization: Computation with spikes,” *Neural computation*, vol. 18, no. 2, pp. 245–282, 2006.
 - [113] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, “Super slo-mo: High quality estimation of multiple intermediate frames for video interpolation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9000–9008.
 - [114] Y. Jin, W. Zhang, and P. Li, “Hybrid macro/micro level backpropagation for training deep spiking neural networks,” *Advances in neural information processing systems*, vol. 31, 2018.
 - [115] H. Jones, K. Grieve, W. Wang, and A. Sillito, “Surround suppression in primate v1,” *Journal of neurophysiology*, vol. 86, no. 4, pp. 2011–2028, 2001.
 - [116] J. P. Jones and L. A. Palmer, “An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex,” *Journal of neurophysiology*, vol. 58, no. 6, pp. 1233–1258, 1987.
 - [117] C. Jutten and J. Herault, “Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture,” *Signal processing*, vol. 24, no. 1, pp. 1–10, 1991.
 - [118] M. L. Katz, K. Nikolic, and T. Delbruck, “Live demonstration: Behavioural emulation of event-based vision sensors,” in *2012 IEEE International Symposium on Circuits and Systems (ISCAS)*, IEEE, 2012, pp. 736–740.
 - [119] D. Kätzel, B. V. Zemelman, C. Buetfering, M. Wölfel, and G. Miesenböck, “The columnar and laminar organization of inhibitory connections to neocortical excitatory cells,” *Nature neuroscience*, vol. 14, no. 1, pp. 100–107, 2011.
 - [120] G. B. Keller and T. D. Mrsic-Flogel, “Predictive processing: A canonical cortical computation,” *Neuron*, vol. 100, no. 2, pp. 424–435, 2018.
 - [121] S. R. Kheradpisheh, M. Ganjtabesh, and T. Masquelier, “Bio-inspired unsupervised learning of visual features leads to robust invariant object recognition,” *Neurocomputing*, vol. 205, pp. 382–392, 2016.

- [122] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, “Stdp-based spiking deep convolutional neural networks for object recognition,” *Neural Networks*, vol. 99, pp. 56–67, 2018.
- [123] R. Kiani, H. Esteky, K. Mirpour, and K. Tanaka, “Object category structure in response patterns of neuronal population in monkey inferior temporal cortex,” *Journal of neurophysiology*, vol. 97, no. 6, pp. 4296–4309, 2007.
- [124] S. Kim, S. Park, B. Na, and S. Yoon, “Spiking-yolo: Spiking neural network for energy-efficient object detection,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, 2020, pp. 11 270–11 277.
- [125] P. D. King, J. Zylberberg, and M. R. DeWeese, “Inhibitory interneurons decorrelate excitatory cells to drive sparse code formation in a spiking model of v1,” *Journal of Neuroscience*, vol. 33, no. 13, pp. 5475–5485, 2013.
- [126] C. Klos, D. Miner, and J. Triesch, “Bridging structure and function: A model of sequence learning and prediction in primary visual cortex,” *PLOS Computational Biology*, vol. 14, no. 6, e1006187, 2018.
- [127] H. Kolb, E. Fernandez, B. Jones, and R. Nelson, “Webvision: The organization of the retina and visual system [internet],” 1995.
- [128] V. Konda and J. Tsitsiklis, “Actor-critic algorithms,” *Advances in neural information processing systems*, vol. 12, 1999.
- [129] R. Kreiser, D. Aathmani, N. Qiao, G. Indiveri, and Y. Sandamirskaya, “Organizing sequential memory in a neuromorphic device using dynamic neural fields,” *Frontiers in neuroscience*, vol. 12, p. 407 706, 2018.
- [130] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, 2012.
- [131] L. Lagnado, “Retinal processing: Amacrine cells keep it short and sweet,” *Current biology*, vol. 8, no. 17, R598–R600, 1998.
- [132] M. Lan, X. Xiong, Z. Jiang, and Y. Lou, “Pc-snn: Supervised learning with local hebbian synaptic plasticity based on predictive coding in spiking neural networks,” *arXiv preprint arXiv:2211.15386*, 2022.
- [133] M. Landy, *Perception lecture notes: Spatial frequency channels [internet]*, 2002.
- [134] R. Larisch, L. Gönner, M. Teichmann, and F. H. Hamker, “Sensory coding and contrast invariance emerge from the control of plastic inhibition over emergent selectivity,” *PLOS Computational Biology*, vol. 17, no. 11, e1009566, 2021.

-
- [135] R. Larisch and F. H. Hamker, “A systematic analysis of the joint effects of ganglion cells, lagged lgn cells, and intercortical inhibition on spatiotemporal processing and direction selectivity,” *Neural Networks*, vol. 186, p. 107273, 2025.
 - [136] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
 - [137] J. H. Lee, T. Delbruck, and M. Pfeiffer, “Training deep spiking neural networks using backpropagation,” *Frontiers in neuroscience*, vol. 10, p. 508, 2016.
 - [138] K. Lee, S. Dora, J. F. Mejias, S. M. Bohte, and C. M. Pennartz, “Predictive coding with spiking neurons and feedforward gist signaling,” *Frontiers in Computational Neuroscience*, vol. 18, p. 1338280, 2024.
 - [139] S. Lefort, C. Tómm, J.-C. F. Sarria, and C. C. Petersen, “The excitatory neuronal network of the c2 barrel column in mouse primary somatosensory cortex,” *Neuron*, vol. 61, no. 2, pp. 301–316, 2009.
 - [140] P. Lennie, “The cost of cortical computation,” *Current biology*, vol. 13, no. 6, pp. 493–497, 2003.
 - [141] J. J. Letzkus, S. B. Wolff, and A. Lüthi, “Disinhibition, a circuit mechanism for associative learning and memory,” *Neuron*, vol. 88, no. 2, pp. 264–276, 2015.
 - [142] W. B. Levy and O. Steward, “Synapses as associative memory elements in the hippocampal formation,” *Brain research*, vol. 175, no. 2, pp. 233–245, 1979.
 - [143] Y. Li and L.-S. Young, “Unraveling the mechanisms of surround suppression in early visual processing,” *PLoS computational biology*, vol. 17, no. 4, e1008916, 2021.
 - [144] F. Lieder, K. E. Stephan, J. Daunizeau, M. I. Garrido, and K. J. Friston, “A neurocomputational model of the mismatch negativity,” *PLoS computational biology*, vol. 9, no. 11, e1003288, 2013.
 - [145] R. Linsker, “Self-organization in a perceptual network,” *Computer*, vol. 21, no. 3, pp. 105–117, 1988.
 - [146] Q. Liu, G. Pan, H. Ruan, D. Xing, Q. Xu, and H. Tang, “Unsupervised aer object recognition based on multiscale spatio-temporal features and spiking neurons,” *IEEE transactions on neural networks and learning systems*, vol. 31, no. 12, pp. 5300–5311, 2020.
 - [147] L. Lonini, S. Forestier, C. Teulière, Y. Zhao, B. E. Shi, and J. Triesch, “Robust active binocular vision through intrinsically motivated learning,” *Frontiers in neurorobotics*, vol. 7, p. 20, 2013.
 - [148] L. L. Lui, J. A. Bourne, and M. G. Rosa, “Functional response properties of neurons in the dorsomedial visual area of new world monkeys (callithrix jacchus),” *Cerebral Cortex*, vol. 16, no. 2, pp. 162–177, 2006.

- [149] W. Maass, “Liquid state machines: Motivation, theory, and applications,” *Computability in context: computation and logic in the real world*, pp. 275–296, 2011.
- [150] W. Maass, T. Natschläger, and H. Markram, “Real-time computing without stable states: A new framework for neural computation based on perturbations,” *Neural computation*, vol. 14, no. 11, pp. 2531–2560, 2002.
- [151] W. Maass and M. Schmitt, “On the complexity of learning for spiking neurons with temporal coding,” *Information and Computation*, vol. 153, no. 1, pp. 26–46, 1999.
- [152] Z. F. Mainen and T. J. Sejnowski, “Reliability of spike timing in neocortical neurons,” *Science*, vol. 268, no. 5216, pp. 1503–1506, 1995.
- [153] H. Markram, J. Lübke, M. Frotscher, and B. Sakmann, “Regulation of synaptic efficacy by coincidence of postsynaptic apss and epsps,” *Science*, vol. 275, no. 5297, pp. 213–215, 1997.
- [154] S. Marom and G. Shahaf, “Development, learning and memory in large random networks of cortical neurons: Lessons beyond anatomy,” *Quarterly reviews of biophysics*, vol. 35, no. 1, pp. 63–87, 2002.
- [155] T. Masquelier and S. J. Thorpe, “Unsupervised learning of visual features through spike timing dependent plasticity,” *PLoS computational biology*, vol. 3, no. 2, e31, 2007.
- [156] R. Massa, A. Marchisio, M. Martina, and M. Shafique, “An efficient spiking neural network for recognizing gestures with a dvs camera on the loihi neuromorphic processor,” in *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020, pp. 1–9.
- [157] A. Masumori, L. Sinapayen, and T. Ikegami, “Predictive Coding as Stimulus Avoidance in Spiking Neural Networks,” *arXiv*, 2019.
- [158] C. Mayr, S. Hoeppner, and S. Furber, “Spinnaker 2: A 10 million core processor system for brain simulation and machine learning-keynote presentation,” in *Communicating Process Architectures 2017 & 2018*, IOS Press, 2019, pp. 277–280.
- [159] C. Mead, “Neuromorphic electronic systems,” *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1629–1636, 1990.
- [160] C. A. Mead and M. A. Mahowald, “A silicon model of early visual processing,” *Neural networks*, vol. 1, no. 1, pp. 91–97, 1988.
- [161] A. Mehonic, A. Sebastian, B. Rajendran, O. Simeone, E. Vasilaki, and A. J. Kenyon, “Memristors—from in-memory computing, deep learning acceleration, and spiking neural networks to the future of neuromorphic and bio-inspired computing,” *Advanced Intelligent Systems*, vol. 2, no. 11, p. 2000085, 2020.

-
- [162] T. Meyer and C. R. Olson, “Statistical learning of visual transitions in monkey inferotemporal cortex,” *Proceedings of the National Academy of Sciences*, vol. 108, no. 48, pp. 19 401–19 406, 2011.
 - [163] M. Mignard and J. G. Malpeli, “Paths of information flow through visual cortex,” *Science*, vol. 251, no. 4998, pp. 1249–1251, 1991.
 - [164] B. Millidge, A. Tschantz, A. Seth, and C. L. Buckley, “Relaxing the constraints on predictive coding models,” *arXiv preprint arXiv:2010.01047*, 2020.
 - [165] M. C. Morrone, D. C. Burr, and L. Maffei, “Functional implications of cross-orientation inhibition of cortical visual cells. i. neurophysiological evidence,” *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 216, no. 1204, pp. 335–354, 1982.
 - [166] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst, “Receptive field organization of complex cells in the cat’s striate cortex,” *The Journal of physiology*, vol. 283, no. 1, pp. 79–99, 1978.
 - [167] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst, “Spatial summation in the receptive fields of simple cells in the cat’s striate cortex,” *The Journal of physiology*, vol. 283, no. 1, pp. 53–77, 1978.
 - [168] M. Mozafari, S. R. Kheradpisheh, T. Masquelier, A. Nowzari-Dalini, and M. Ganjtabesh, “First-spike-based visual categorization using reward-modulated stdp,” *IEEE transactions on neural networks and learning systems*, vol. 29, no. 12, pp. 6178–6190, 2018.
 - [169] S. O. Murray, D. Kersten, B. A. Olshausen, P. Schrater, and D. L. Woods, “Shape perception reduces activity in human primary visual cortex,” *Proceedings of the National Academy of Sciences*, vol. 99, no. 23, pp. 15 164–15 169, 2002.
 - [170] S. O. Murray, C. A. Olman, and D. Kersten, “Spatially specific fmri repetition effects in human visual cortex,” *Journal of neurophysiology*, vol. 95, no. 4, pp. 2439–2445, 2006.
 - [171] A. W. N’dri, W. Gebhardt, C. Teulière, *et al.*, *Predictive coding with spiking neural networks: A survey*, 2024.
 - [172] A. W. N’dri, T. Barbier, C. Teulière, and J. Triesch, “Predictive coding light,” *Nature Communications*, vol. 16, no. 1, p. 8880, 2025.
 - [173] A. W. N’dri, T. Barbier, C. Teulière, and J. Triesch, “Predictive Coding Light: learning compact visual codes by combining excitatory and inhibitory spike timing-dependent plasticity*,” *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, vol. 00, pp. 3997–4006, 2023.

- [174] A. Nadafian and M. Ganjtabesh, “Bio-plausible unsupervised delay learning for extracting temporal features in spiking neural networks,” *arXiv preprint arXiv:2011.09380*, 2020.
- [175] A. Nadafian, M. Mozafari, T. Masquelier, and M. Ganjtabesh, “Brain-inspired computational modeling of action recognition with recurrent spiking neural networks equipped with reinforcement delay learning,” *arXiv preprint arXiv:2406.11778*, 2024.
- [176] J.-P. Nadal and N. Parga, “Nonlinear neurons in the low-noise limit: A factorial code maximizes information transfer,” *Network: Computation in neural systems*, vol. 5, no. 4, pp. 565–581, 1994.
- [177] H. Nakamura, R. Gattass, R. Desimone, and L. G. Ungerleider, “The modular organization of projections from areas v1 and v2 to areas v4 and teo in macaques,” *Journal of Neuroscience*, vol. 13, no. 9, pp. 3681–3691, 1993.
- [178] M. Nardin, J. W. Phillips, W. F. Podlaski, and S. W. Keemink, “Nonlinear computations in spiking neural networks through multiplicative synapses,” *Peer Community Journal*, vol. 1, e68, 2021.
- [179] E. O. Neftci, H. Mostafa, and F. Zenke, “Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks,” *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 51–63, 2019.
- [180] B. A. Olshausen and D. J. Field, “Emergence of simple-cell receptive field properties by learning a sparse code for natural images,” *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.
- [181] G. Orchard, E. P. Frady, D. B. D. Rubin, *et al.*, “Efficient neuromorphic signal processing with loihi 2,” in *2021 IEEE Workshop on Signal Processing Systems (SiPS)*, IEEE, 2021, pp. 254–259.
- [182] G. Orchard, A. Jayawant, G. K. Cohen, and N. Thakor, “Converting static image datasets to spiking neuromorphic datasets using saccades,” *Frontiers in neuroscience*, vol. 9, p. 437, 2015.
- [183] A. Ororbia, “Spiking neural predictive coding for continually learning from data streams,” *Neurocomputing*, vol. 544, p. 126 292, 2023.
- [184] A. Ororbia and D. Kifer, “The neural coding framework for learning generative models,” *Nature communications*, vol. 13, no. 1, p. 2064, 2022.
- [185] A. Ororbia, A. Mali, A. Kohan, B. Millidge, and T. Salvatori, “A review of neuroscience-inspired machine learning,” *arXiv preprint arXiv:2403.18929*, 2024.
- [186] H. Ozeki, I. M. Finn, E. S. Schaffer, K. D. Miller, and D. Ferster, “Inhibitory stabilization of the cortical network underlies visual surround suppression,” *Neuron*, vol. 62, no. 4, pp. 578–592, 2009.

-
- [187] E. R. Palacios, T. Isomura, T. Parr, and K. Friston, “The emergence of synchrony in networks of mutually inferring neurons,” *Scientific reports*, vol. 9, no. 1, p. 6412, 2019.
 - [188] A. Pasupathy and C. E. Connor, “Shape representation in area v4: Position-specific tuning for boundary conformation,” *Journal of neurophysiology*, 2001.
 - [189] D. Patterson, J. Gonzalez, Q. Le, *et al.*, “Carbon emissions and large neural network training,” *arXiv preprint arXiv:2104.10350*, 2021.
 - [190] J.-P. Pfister and W. Gerstner, “Triplets of spikes in a model of spike timing-dependent plasticity,” *Journal of Neuroscience*, vol. 26, no. 38, pp. 9673–9682, 2006.
 - [191] R. V. W. Putra and M. Shafique, “Fspinn: An optimization framework for memory-efficient and energy-efficient spiking neural networks,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 11, pp. 3601–3613, 2020.
 - [192] R. P. N. Rao, “Hierarchical Bayesian inference in networks of spiking neurons,” *Advances in neural information processing systems*, vol. 17, 2004.
 - [193] R. P. N. Rao and T. J. Sejnowski, “Predictive sequence learning in recurrent neocortical circuits,” *Advances in neural information processing systems*, vol. 12, 1999.
 - [194] R. P. N. Rao and T. J. Sejnowski, “Predictive coding, cortical feedback, and spike-timing dependent plasticity,” *Probabilistic models of the brain*, p. 297, 2002.
 - [195] R. P. Rao and D. H. Ballard, “Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects,” *Nature neuroscience*, vol. 2, no. 1, pp. 79–87, 1999.
 - [196] R. P. Rao, D. C. Gklezakos, and V. Sathish, “Active predictive coding: A unified neural framework for learning hierarchical world models for perception and planning,” *arXiv preprint arXiv:2210.13461*, 2022.
 - [197] R. P. Rao and T. J. Sejnowski, “Spike-timing-dependent hebbian plasticity as temporal difference learning,” *Neural computation*, vol. 13, no. 10, pp. 2221–2237, 2001.
 - [198] S. Rastogi, N. Dennler, M. Schmuker, and A. Van Schaik, “A neuromorphic electronic nose design,” *IEEE Sensors Journal*, 2025.
 - [199] D. Richter, *Metabolism of the nervous system*. Elsevier, 2013.
 - [200] M. Riesenhuber and T. Poggio, “Hierarchical models of object recognition in cortex,” *Nature neuroscience*, vol. 2, no. 11, pp. 1019–1025, 1999.
 - [201] Y. B. Saalman, I. N. Pigarev, and T. R. Vidyasagar, “Neural mechanisms of visual attention: How top-down feedback highlights relevant locations,” *Science*, vol. 316, no. 5831, pp. 1612–1615, 2007.

- [202] S. P. Sabatini, V. Baruzzi, and G. Indiveri, “Recurrent models of orientation selectivity enable robust early-vision processing in mixed-signal neuromorphic hardware,” *Research Square preprint*, 2023.
- [203] R. N. Sachdev, M. R. Krause, and J. A. Mazer, “Surround suppression and sparse coding in visual and barrel cortices,” *Frontiers in neural circuits*, vol. 6, p. 43, 2012.
- [204] S. Sadeh and C. Clopath, “Inhibitory stabilization and cortical computation,” *Nature Reviews Neuroscience*, vol. 22, no. 1, pp. 21–37, 2021.
- [205] A. Safa, I. Ocket, A. Bourdoux, H. Sahli, F. Catthoor, and G. G. Gielen, “Event camera data classification using spiking networks with spike-timing-dependent plasticity,” in *2022 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2022, pp. 1–8.
- [206] T. Salvatori, A. Mali, C. L. Buckley, *et al.*, “Brain-inspired computational intelligence via predictive coding,” *arXiv preprint arXiv:2308.07870*, 2023.
- [207] A. Samadzadeh, F. S. T. Far, A. Javadi, A. Nickabadi, and M. H. Chehreghani, “Convolutional spiking neural networks for spatio-temporal feature extraction,” *Neural Processing Letters*, vol. 55, no. 6, pp. 6979–6995, 2023.
- [208] Y. Sandamirskaya, “Dynamic neural fields as a step toward cognitive neuromorphic architectures,” *Frontiers in neuroscience*, vol. 7, p. 71 560, 2014.
- [209] M. Saponati and M. Vinck, “Sequence anticipation and spike-timing-dependent plasticity emerge from a predictive learning rule,” *Nature Communications*, vol. 14, no. 1, p. 4985, 2023.
- [210] C. Savin, P. Joshi, and J. Triesch, “Independent component analysis in spiking neurons,” *PLoS computational biology*, vol. 6, no. 4, e1000757, 2010.
- [211] C. Savin and J. Triesch, “Emergence of task-dependent representations in working memory circuits,” *Frontiers in computational neuroscience*, vol. 8, p. 57, 2014.
- [212] A. Schulz, C. Miehl, M. J. Berry II, and J. Gjorgjieva, “The generation of cortical novelty responses through inhibitory plasticity,” *Elife*, vol. 10, e65309, 2021.
- [213] M. A. Schwemmer, A. L. Fairhall, S. Denève, and E. T. Shea-Brown, “Constructing Precisely Computing Networks with Biophysical Spiking Neurons,” *Journal of Neuroscience*, vol. 35, no. 28, pp. 10 112–10 134, Jul. 2015.
- [214] G. Shahaf and S. Marom, “Learning in networks of cortical neurons,” *Journal of Neuroscience*, vol. 21, no. 22, pp. 8782–8788, 2001.
- [215] C. E. Shannon, “A mathematical theory of communication,” *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

-
- [216] S. M. Sherman and R. Guillery, “On the actions that one nerve cell can have on another: Distinguishing “drivers” from “modulators,”” *Proceedings of the National Academy of Sciences*, vol. 95, no. 12, pp. 7121–7126, 1998.
 - [217] S. Shipp, “Structure and function of the cerebral cortex,” *Current Biology*, vol. 17, no. 12, R443–R449, 2007.
 - [218] S. B. Shrestha and G. Orchard, “Slayer: Spike layer error reassignment in time,” *Advances in neural information processing systems*, vol. 31, 2018.
 - [219] L. Sinapayen, A. Masumori, and T. Ikegami, “Learning by stimulation avoidance: A principle to control spiking neural networks dynamics,” *PloS one*, vol. 12, no. 2, e0170388, 2017.
 - [220] L. Sinapayen, A. Masumori, N. Virgo, and T. Ikegami, “Learning by stimulation avoidance as a primary principle of spiking neural networks dynamics,” in *Artificial Life Conference Proceedings*, MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info ..., 2015, pp. 175–182.
 - [221] S. Singh, A. Sarma, S. Lu, A. Sengupta, V. Narayanan, and C. R. Das, “Gesture-snn: Co-optimizing accuracy, latency and energy of snns for neuromorphic vision sensors,” in *2021 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, IEEE, 2021, pp. 1–6.
 - [222] P. J. Sjöström, G. G. Turrigiano, and S. B. Nelson, “Rate, timing, and cooperativity jointly determine cortical synaptic plasticity,” *Neuron*, vol. 32, no. 6, pp. 1149–1164, 2001.
 - [223] F. S. Slijkhuis, S. W. Keemink, and P. Lanillos, “Closed-Form Control With Spike Coding Networks,” *IEEE Transactions on Cognitive and Developmental Systems*, pp. 1–1, 2023.
 - [224] R. E. Soans, D. C. Lim, B. T. Keenan, A. I. Pack, and J. A. Shackleford, “Automated protein localization of blood brain barrier vasculature in brightfield ihc images,” *PloS one*, vol. 11, no. 2, e0148411, 2016.
 - [225] M. W. Spratling, “A review of predictive coding algorithms,” *Brain and cognition*, vol. 112, pp. 92–97, 2017.
 - [226] G. Stefanics, J. Kremláček, and I. Czigler, “Visual mismatch negativity: A predictive coding view,” *Frontiers in human neuroscience*, vol. 8, p. 666, 2014.
 - [227] P. Sterzer and G. B. Keller, “Predictive processing: Layer-specific prediction error signals in human cortex,” *Current Biology*, vol. 34, no. 10, R496–R498, 2024.
 - [228] A. Subramoney, F. Scherr, and W. Maass, “Reservoirs learn to learn,” in *Reservoir Computing: Theory, Physical Implementations, and Applications*, Springer, 2021, pp. 59–76.

- [229] C. Summerfield, E. H. Trittschuh, J. M. Monti, M.-M. Mesulam, and T. Egner, “Neural repetition suppression reflects fulfilled perceptual expectations,” *Nature neuroscience*, vol. 11, no. 9, pp. 1004–1006, 2008.
- [230] C. Summerfield, V. Wyart, V. M. Johnen, and V. De Gardelle, “Human scalp electroencephalography reveals that repetition suppression varies with expectation,” *Frontiers in Human Neuroscience*, vol. 5, p. 67, 2011.
- [231] K. Tanaka, “Inferotemporal cortex and object vision,” *Annual review of neuroscience*, vol. 19, no. 1, pp. 109–139, 1996.
- [232] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, “Deep learning in spiking neural networks,” *Neural networks*, vol. 111, pp. 47–63, 2019.
- [233] A. Tavanaei, Z. Kirby, and A. S. Maida, “Training spiking convnets by stdp and gradient descent,” in *2018 international joint conference on neural networks (IJCNN)*, IEEE, 2018, pp. 1–8.
- [234] A. Tavanaei, T. Masquelier, and A. S. Maida, “Acquisition of visual features through probabilistic spike-timing-dependent plasticity,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2016, pp. 307–314.
- [235] L. Taylor, F. Zenke, A. J. King, and N. S. Harper, “Temporal prediction captures key differences between spiking excitatory and inhibitory v1 neurons,” *bioRxiv*, pp. 2024–05, 2024.
- [236] C. Teulière, S. Forestier, L. Lonini, *et al.*, “Self-calibrating smooth pursuit through active efficient coding,” *Robotics and Autonomous Systems*, vol. 71, pp. 3–12, 2015.
- [237] E. R. Thomas, J. Haarsma, J. Nicholson, D. Yon, P. Kok, and C. Press, “Predictions and errors are distinctly represented across v1 layers,” *Current Biology*, vol. 34, no. 10, pp. 2265–2271, 2024.
- [238] A. M. Thomson, D. C. West, Y. Wang, and A. P. Bannister, “Synaptic connections and small circuits involving excitatory and inhibitory neurons in layers 2–5 of adult rat and cat neocortex: Triple intracellular recordings and biocytin labelling in vitro,” *Cerebral cortex*, vol. 12, no. 9, pp. 936–953, 2002.
- [239] A. Todorovic, F. van Ede, E. Maris, and F. P. de Lange, “Prior expectation mediates neural adaptation to repeated sounds in the auditory cortex: An meg study,” *Journal of Neuroscience*, vol. 31, no. 25, pp. 9118–9123, 2011.
- [240] J. Triesch, A. D. Vo, and A.-S. Hafner, “Competition for synaptic building blocks shapes synaptic plasticity,” *Elife*, vol. 7, e37836, 2018.
- [241] L. G. Ungerleider, “Two cortical visual systems,” *Analysis of visual behavior*, vol. 549, chapter–18, 1982.

-
- [242] J. H. Van Hateren and A. van der Schaaf, “Independent component filters of natural images compared with simple cells in primary visual cortex,” *Proceedings of the Royal Society of London. Series B: Biological Sciences*, vol. 265, no. 1394, pp. 359–366, 1998.
 - [243] B. C. de Vangel, C. Torres-Huitzil, and B. Girau, “Event based visual attention with dynamic neural field on fpga,” in *Proceedings of the 10th International Conference on Distributed Smart Camera*, 2016, pp. 142–147.
 - [244] J. Vezoli, A. Falchier, B. Jouve, K. Knoblauch, M. Young, and H. Kennedy, “Quantitative analysis of connectivity in the visual cortex: Extracting function from structure,” *The Neuroscientist*, vol. 10, no. 5, pp. 476–482, 2004.
 - [245] T. P. Vogels, H. Sprekeler, F. Zenke, C. Clopath, and W. Gerstner, “Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks,” *Science*, vol. 334, no. 6062, pp. 1569–1573, 2011.
 - [246] H. Von Helmholtz, *Handbuch der physiologischen Optik*. Voss, 1867, vol. 9.
 - [247] C. Wacongne, “A predictive coding account of MMN reduction in schizophrenia,” *Biological psychology*, vol. 116, pp. 68–74, 2016.
 - [248] C. Wacongne, J.-P. Changeux, and S. Dehaene, “A neuronal model of predictive coding accounting for the mismatch negativity,” *Journal of Neuroscience*, vol. 32, no. 11, pp. 3665–3678, 2012.
 - [249] H.-X. Wang, R. C. Gerkin, D. W. Nauen, and G.-Q. Bi, “Coactivation and timing-dependent integration of synaptic potentiation and depression,” *Nature neuroscience*, vol. 8, no. 2, pp. 187–193, 2005.
 - [250] N. Weiler, L. Wood, J. Yu, S. A. Solla, and G. M. Shepherd, “Top-down laminar organization of the excitatory network in motor cortex,” *Nature neuroscience*, vol. 11, no. 3, pp. 360–366, 2008.
 - [251] P. J. Werbos, “Backpropagation through time: What it does and how to do it,” *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.
 - [252] T. C. Wunderlich and C. Pehle, “Event-based backpropagation can compute exact gradients for spiking neural networks,” *Scientific Reports*, vol. 11, no. 1, p. 12 829, 2021.
 - [253] Y. Xing, G. Di Caterina, and J. Soraghan, “A new spiking convolutional recurrent neural network (scrnn) with applications to event-based hand gesture recognition,” *Frontiers in neuroscience*, vol. 14, p. 590 164, 2020.
 - [254] K. Yamazaki, V.-K. Vo-Ho, D. Bulsara, and N. Le, “Spiking neural networks and their applications: A review,” *Brain sciences*, vol. 12, no. 7, p. 863, 2022.
 - [255] J. J. Yang, D. B. Strukov, and D. R. Stewart, “Memristive devices for computing,” *Nature nanotechnology*, vol. 8, no. 1, pp. 13–24, 2013.

- [256] M. Yao, G. Zhao, H. Zhang, *et al.*, “Attention spiking neural networks,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 8, pp. 9393–9410, 2023.
- [257] U. Zahid, Q. Guo, and Z. Fountas, “Predictive coding as a neuromorphic alternative to backpropagation: A critical evaluation,” *Neural Computation*, vol. 35, no. 12, pp. 1881–1909, 2023.
- [258] T. Zhang, X. Cheng, S. Jia, M.-m. Poo, Y. Zeng, and B. Xu, “Self-backpropagation of synaptic modifications elevates the efficiency of spiking and artificial neural networks,” *Science advances*, vol. 7, no. 43, eabh0146, 2021.
- [259] Z. Zhang, C. Yin, and T. Yang, “Evidence accumulation occurs locally in the parietal cortex,” *Nature Communications*, vol. 13, no. 1, p. 4426, 2022.
- [260] Z. Zhang and D. H. Ballard, “A single spike model of predictive coding,” *Neurocomputing*, vol. 58, pp. 165–171, 2004.
- [261] B. Zhao, R. Ding, S. Chen, B. Linares-Barranco, and H. Tang, “Feedforward categorization on aer motion events using cortex-like features in a spiking neural network,” *IEEE transactions on neural networks and learning systems*, vol. 26, no. 9, pp. 1963–1978, 2014.
- [262] V. Zhu and R. Rosenbaum, “Evaluating the extent to which homeostatic plasticity learns to compute prediction errors in unstructured neuronal networks,” *Journal of Computational Neuroscience*, vol. 50, no. 3, pp. 357–373, 2022.
- [263] W. Zhu, M. Shelley, and R. Shapley, “A neuronal network model of primary visual cortex explains spatial frequency selectivity,” *Journal of computational neuroscience*, vol. 26, pp. 271–287, 2009.
- [264] J. Zylberberg, J. T. Murphy, and M. R. DeWeese, “A sparse coding model with synaptically local plasticity and spiking neurons can account for the diverse shapes of v1 simple cell receptive fields,” *PLoS computational biology*, vol. 7, no. 10, e1002250, 2011.