Efficacité des méthodes locales pour la classification d'images et la regression d'énergie en physique

Louis Thiry

PSL UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à l'École Normale Supérieure
et à Sorbonne Université

On the efficiency of local methods in image classification and energy regression in physics

Soutenue par
Louis Thiry
Le 9 Juillet 2021

École doctorale n°386

École Doctorale De Sciences Mathématiques de Paris

Spécialité

Mathématiques Appliquées

Composition du jury :

Gabriel Stoltz Professeur, École nationale des ponts et chaussées	*Président du jury*
Tony Lelièvre Professeur, École nationale des ponts et chaussées	*Rapporteur*
Matthew Blashko Professeur, Université Catholique de Louvain	*Rapporteur*
Mihai-Cosmin Marinica Chercheur HDR, CEA Saclay	*Examinateur*
Pierre Monmarché Maître de conférence HDR, Sorbonne Université	*Examinateur*
Stéphane Mallat Professeur, Collège de France	*Directeur de thèse*

PSL SORBONNE UNIVERSITÉ

" L'optimisme est une fausse espérance à l'usage des lâches et des imbéciles. L'espérance est une vertu, « virtus », une détermination héroïque de l'âme. La plus haute forme de l'espérance, c'est le désespoir surmonté. "

Georges Bernanos, *La Liberté, pour quoi faire ?* (1953)

À ma mère pour la joie et l'espérance qu'elle a toujours eues. À mon père pour la liberté d'esprit qu'il m'a transmise.

Remerciements

Je remercie en premier lieu mon directeur de thèse Stéphane Mallat de m'avoir proposé de travailler à la croisée de plusieurs disciplines. Cela m'a permis de diversifier les sources d'inspirations, de discuter avec des personnes de communautés scientifiques variées, et de rebondir dans le travail lorsque l'on bloquait sur un aspect ou un problème. Je remercie Meire Fortunato, pour son accueil chaleureux chez DeepMind, Théophane Weber, pour les nombreuses discussions que nous avons eues là-bas, Aron Cohen, qui m'a fait découvrir la théorie de la fonctionnelle de densité, et James Kirkpatrick qui a pris le projet en main. Je remercie Gabor Csanyi et Alexandre Tkatchenko pour la collaboration sur les énergies longues portées dans le graphène. Je remercie Cosmin Marinica et Clovis Lapointe du CEA pour notre fructueuse collaboration, et à qui j'ai toujours rendu visite avec plaisir à Saclay. Je remercie Edouard Oyallon pour les nombreuses discussions sur la classification d'images autour des *BagNet*, à la suite desquelles je me suis lancé sur la classification d'images par patch, et pour notre récente collaboration avec Michael et Eugène.

Une partie non négligeable de mon travail a concerné le monitorat challenge data, et le développement en Django du site challengedata.ens.fr. Je remercie Félix Purchiopoincom et Jean-Rémi pour leur aide précieuse, ainsi que Tanguy Marchand pour sa reprise du site.

J'ai également beaucoup apprécié l'ambiance au labo. Je voudrais remercier Georgios Exarchakis pour m'avoir encadré puis épaulé pendant mon stage et ma thèse, Mathieu Andreux pour son accueil au sein de l'équipe et Roberto Leonarduzzi (alias Leonardo Robertuzzi) pour m'avoir fait redécouvrir le maté. Je remercie Gaspar Rochette pour avoir relevé le style vestimentaire de l'équipe, je remercie fguth rusty-le-clown pour les slurm et Rudy Rastignac pour sa fougue dans les débat. Je remercie particulièrement Tomas Angles pour les discussions à bâtons rompus que nous avons eues sur toute sorte de sujets, et pour son exigence de précision (*Qu'est-ce que tu veux dire par précision?*). Je remercie particulièrement John Zarka, maître tartinier hors classe, pour ses saillies sur les zadistes de l'ENS, et les bonnes infos du Figaga qui ont nourri nos discussions parfois houleuses, mais toujours sincères. Je remercie enfin Louis Ty mon double.

Je voudrais remercier Sophie Jaudon et tout spécialement Lise-Marie Bivard qui ont été d'extraordinaires interlocutrices pour toutes les démarches administratives durant ces trois années. Je remercie également Ludovic Ricariou et Jacques Beigbeder pour leur précieux travail d'administration logicielle et matérielle.

Je tiens beaucoup au travail effectué avec mon ami Thomas Jean-Daniel Kerdreux Deglisposti, que j'ai rencontré il y a plus de dix ans au club de rugby de Fontenay-aux-roses. Grâce

à sa curiosité, à sa force de travail et à son exigence de vérité, nous nous sommes lancés sur l'interaction créative entres machines et artistes. Nous souhaitons à travers ce travail apporter un regard juste et éclairant sur l'utilisation grandissante des ordinateurs dans la création artistique, en cette époque charnière de *transformation digitale* et de *vie sans contact*. Je remercie également Raphaël Dallaporta pour notre collaboration sur l'*indice de volatilité*.

Je voudrais maintenant remercier ma mère pour le formidable témoignage de vie et d'espérance qu'elle a été pendant ses quatre derniers mois. Je remercie mon père qui m'a transmis sa liberté d'esprit, et qui m'a appris à estimer les gens de par leur qualités et non leur titres et statuts. Je remercie Jufess, Cel, Mary, Joku et MK pour la joyeuse fratrie que nous formons. Je remercie ma Lolo pour tout ce que l'on partage, et aussi Gangan et Zazzouze. Je remercie Solenn et Agnès, mes amies de longue date. Je remercie Valaté Malbès et Rémi pour notre belle cohabition dans la rue du quai de la Seine pendant les deux premères années de thèse, ainsi que les Jingles qui ont inspiré quelques chansons. Je remercie Sophie Hervé qui m'a accompagné sur le chemin de la voix, en parallèle de ma thèse. Je remercie Les Temps Dérobés pour les moments et concerts magiques passés ensembles. Je remercie Martin pour les parties de carton du jeudi.

Résumé

Les réseaux de neurones profonds ont permis récemment d'importants progrès dans les problèmes d'apprentissage en grande dimension, notamment en classification d'images et en régression d'énergie en physique. Ces deux problèmes sont de nature multi-échelle. En effet, l'énergie des molécules et des solides résulte d'interactions à différentes échelles, avec par exemple les liaisons ioniques et covalentes à petite échelle, les interactions de Van-der-Waals aux échelles moyennes et les interactions de Coulomb à grande échelle. De même, on peut classifier une image en utilisant des informations de texture à petite échelle, des informations de motif à moyenne échelle ou des informations de forme à l'échelle de l'objet. De plus, il existe une analogie naturelle entre les techniques de classification d'images dites locales, basées sur des petits patch d'image, et les techniques de régression énergétique dites locales, basées sur la description de petits voisinages atomiques dans les molécules ou les solides.

Dans ce manuscrit, nous étudions l'efficacité des méthodes locales pour la classification d'images et la régression d'énergie en physique. On observe que les méthodes locales sont étonnamment performantes pour ces deux problèmes, et ce malgré la nature multi-échelle de ces problèmes. Tout d'abord, nous étudions comparativement des techniques multi-échelles et locales pour la régression d'énergie de molécules et solides. Nous constatons que les méthodes locales sont très performantes, même pour les solides avec des composantes énergétiques à longue portée. Nous présentons une nouvelle méthode pour la régression d'entropie vibrationnelle dans les solides. Là encore, nous observons qu'une méthode utilisant des descripteurs locaux donne de bien meilleurs résultats que la stratégie multi-échelle étudiée. Pour la classification d'images, nous présentons un réseau de neurones convolutif structuré basé sur l'encodage de patch. Cette architecture donne des performances comparables à des réseaux convolutifs standards sur la base de données ImageNet. Enfin nous présentons un classificateur d'images basé sur des calculs de K-plus-proches-voisins de patch d'images, et dont les performances surprenantes suggèrent une forme de basse dimension des patch d'images. Nous terminons ce manuscrit par une ouverture sur les dispositifs interactifs humain-machine pour la création artistique.

Abstract

The recent success of convolutional neural networks in high-dimensional learning problems motivated the development of new machine learning techniques in different fields. In particular, a lot of progress has been made in image classification and energy regression in physics in the last years. These two problems are multi-scale. Molecules' and solids' energy results from interactions at different scales, e.g., ionic and covalent bonds at the short scale, Van-der-Waals interactions at the mesoscale, Coulomb interactions at the large scale. One can classify an image using texture information at the small scale, pattern information at a larger scale, or shape information at the object scale. There is a natural analogy between local image classification techniques based on small image patches and local energy regression techniques based on small atomic neighborhood descriptions.

This dissertation studies the efficiency of local methods in image classification and energy regression. We observe that local methods perform surprisingly well in image classification and energy regression despite these two problems' multi-scale nature. We first study comparatively multi-scale and local energy regression techniques for molecules and solids. We notice that local methods perform very well, even for solids with long-range energy components. We present a new strategy to regress the vibrational entropy in solids. Again, we observe that a local method based on atomic neighborhood description has better predictive power than a multi-scale strategy. We introduce a structured convolutional network architecture for image classification based on patch encoding that reaches an accuracy that is competitive with standard convolutional networks on ImageNet. We present an image classifier based on image patches K-nearest-neighbors computations that achieves state-of-the-art performance as a non-learned representation. We end this dissertation with an opening on artistic creativity in the context of human-machine interactive systems.

Contents

Chapter 1

Introduction

1.1 Curse of dimensionality in supervised learning

Supervised learning consists in learning a model from a set of input-output pairs to predict the output on new inputs. From a deterministic point of view, it consists in approximating a function F that maps an input x to a deterministic output $y = F(x)$. From a probabilistic point of view, it consists in modeling $F(x) = p(y|x)$, the conditional probability of y given x.

Under usual regularity assumptions such as L-Lipschitz continuity, the number N of pairs (x_i, y_i) needed to approximate the function F with a precision ϵ grows exponentially with the dimension D. There is a *curse of dimensionality*. In usual supervised learning problems, the number N of available input-output pairs goes from 10^3 to 10^9. With such numbers of pairs, dimensions D above 10 are already too high to approximate an L-Lipschitz function correctly.

Most of the practical supervised learning problems are very high-dimensional. In image classification, for example, the input x is an image with typically $D = 3 \times 256^2 = 2.10^5$ pixels and the output is the class of the object in the image. In speech recognition, the input x is a recording of typically 10 seconds of speech sampled at 10^4kHz, so $D = 10^5$, and the output is the list of spoken words. In energy regression in physics, the input $x = (r_1, \dots r_{N_a})$ is the set of 3-dimensional positions r_{N_a} of N_a atoms. N_a being typically equal to 10^3, the input dimension is $D = 3.10^3$, and the output is the atomic system's physical energy.

In recent years, deep neural networks have emerged as a potential class of non-linear functions F that allowed avoiding the curse of dimensionality for image classification [Krizhevsky et al., 2012], speech recognition [Graves et al., 2013] and energy regression in physics [Schütt et al., 2018]. In these examples, 10^4 to 10^7 samples are sufficient to approximate the function F with good precision. These numbers are far below an exponential of the input dimension D.

1.2 Local methods

In the second chapter *Règles de la méthode* of the *Discours de la méthode*, Descartes [1637] states four rules for the scientific method. "*The second [rule is] to divide each of the difficulties I would examine into as many plots as I could, and as many as would be required to solve them best.*" This general rule takes the following form in the context of supervised learning.

The input $x = (x^1, \dots, x^D) \in \mathbb{R}^D$ is separated into a set of sub-variables $s^k \in \mathbb{R}^{D_k}, k \in [\![1, K]\!]$ that are in dimension $D_k \ll D$. These sub-variables can be, for example, subsets of the native variables x^j. They can also be defined using (non-)linear transformations $\Phi^k : \mathbb{R}^D \mapsto \mathbb{R}^{D_k}$ of the input variable x.

$$s^k = \Phi^k(x)$$

From a functional analysis point of view, this separation hypothesis states that the function F to be approximated is the sum of K functions F_k of the sub-variables s^k

$$F(x) = \sum_{k=1}^{K} F_k(s^k) \tag{1.1}$$

This separation allows avoiding the curse of dimensionality if the sub-variables are sufficiently low-dimensional. The number K of functions does not affect the dimensionality of the problem.

When the input variable has a spatial topology, one can formulate the aforementioned separation hypothesis by gathering neighboring variables. It results in a particular type of separation method that we call **local methods**.

1.2.1 Local methods in physics

In energy regression in physics, local methods consist of separating the energy into atomic neighborhood contributions. The neighborhood of the atom i is denoted by $\mathcal{N}_i$. $\mathcal{N}_i$ is typically a ball of radius r_{cut} centered on the atomic position r_i. The energy is assumed to be a sum over atomic contributions. For the energy contribution of the atom i, one considers only the atoms j that are in the neighborhood $\mathcal{N}_i$. Under these assumptions, the energy function E becomes

$$E(r_1, \dots, r_{N_a}) = \sum_{i}^{N_a} E\left(\{r_j, j \in \mathcal{N}_i\}\right) \tag{1.2}$$

Since the number of atoms in the neighborhood $\mathcal{N}_i$ can vary, one represents the neighborhood $\mathcal{N}_i$ with a descriptor $q(\mathcal{N}_i)$ of fixed dimension. This descriptor has to be invariant w.r.t rotations and translations of the atoms. It is also supposed to be uniquely determined by the atomic neighborhood $\mathcal{N}_i$, up to rotations and translations. This short-range separation hypothesis allows *breaking the curse of dimensionality* as the dimensionality goes from $3N_a$ to a fixed dimension d of the descriptor, which is of the order of 10^2 typically.

This local separation hypothesis has not only been used in energy regression. In Density Functional Theory [Hohenberg and Kohn, 1964, Kohn and Sham, 1965], this hypothesis is known as *nearsightedness* [Kohn, 1996]. It assumes the locality of the electronic interactions in the exchange-correlation energy. The local density approximation [Ceperley and Alder, 1980, Perdew and Zunger, 1981] and generalized gradient approximation [Perdew et al., 1996] assume the *nearsightedness* of electronic interactions.

1.2.2 Local methods in image classification

Before the supremacy of deep neural networks, state-of-the-art image classification techniques relied on the image's separation into patches [Perronnin et al., 2010, Wang et al., 2010]. An

image x with $H \times W$ pixels $x_{ij}, i < H, j < W$ is decomposed into patches p with Q^2 pixels $p_{i,j}, i, j < Q$. It allows reducing the dimensionality of the problem from dimension $3HW$ ($\sim 10^5$ typically) to $3Q^2$ ($\sim 10^3$ typically). One can further reduce the dimension to $\sim 10^2$ computing Scale-invariant feature transform (SIFT, Lowe [2004]) or Histograms of Oriented Gradients (HOG, Dalal and Triggs [2005]) descriptors $D(p)$ of theses patches p. This descriptor $D(p)$ is then encoded into $\Phi(D(p))$, using, for example, a finite-dimensional approximation of the Fisher kernel [Perronnin et al., 2010]. Finally, Perronnin et al. [2010], Wang et al. [2010] average the encoding $\Phi(D(p))$ over image patches p. They apply a linear operator W for the classification. The average over spatial indices and the linear operations over the encoding $\Phi(D(p))$ dimension can be inverted. The classification decision $F(x)$ is a hence sum over the patches p of the image x

$$F(x) = \sum_{p \in x} W\Phi(D(p)) \ .$$

Contrarily to physics, where the distances are absolute, and the size of the neighborhood limited to a few Angstroms ($\dot{A}$), the size of the patch in an image is related to the resolution of the image. A 8×8 image patch of a 32×32 CIFAR-10 and A 8×8 image patch of a 256×256 ImageNet image contain qualitatively different information. The size of the patch in an image corresponds hence to a certain scale. Patch separation is implicitly a form of scale separation with a scale corresponding to the ratio between the patch size and the image size.

1.2.3 Convolutional neural networks

Convolutional neural networks (CNNs) show impressive results on image classification [Krizhevsky et al., 2012, He et al., 2016] or energy regression in physics [Schütt et al., 2018]. Contrarily to local methods, CNNs process the input as a whole. They do not separate the image into patches nor a molecule into atomic neighborhoods. They yield higher accuracy than methods based on separation in image classification and energy regression in physics.

On the one hand, removing local separation increases the input dimension: the function is harder to approximate. On the other hand, enforcing a local separation of the input can discard some relevant information for the classification or regression task. Classifying an image using the whole image instead image's patches allows analyzing global shape shapes that are not contained in patches. LeCun et al. [2015] propose the following explanation of the success of CNNs in image classification: *" The first layer [of the convolutional network] represents the presence or absence of edges [...] in the image. The second layer typically detects motifs [...]. The third layer may assemble motifs into [...] parts of familiar objects, and subsequent layers would detect objects as combinations of these parts."* A similar interpretation is given by Kriegeskorte [2015]: *"the [deep convolutional] network acquires complex knowledge about the kinds of shapes associated with each category. [...] High-level units appear to learn representations of shapes occurring in natural images."*

The energy of an atomic system results from complex many-body interactions. Moreover, methods based on short-range separation only account for interaction in the range of 0 to 5Å typically. One can hypothesize that they can not capture long-range interactions such

as Van-der-Waals interactions ranging up to 15Å. The *SchNet* convolutional neural network introduced by Schütt et al. [2018] can theoretically model interactions up to a range of 30Å.

1.2.4 Local methods' efficiency for image classification and energy regression

In this dissertation, we study image classification and energy regression techniques relying on local separation. What are the benefits of using local separation for the interpretability of the predictions? Do the local methods performs significantly worse than non-local methods on image classification and energy regression? If yes, how can we capture non-local components of the function we are trying to approximate? If no, what does it tells us about the underlying regularity properties of the supervised learning problem?

Image classification and energy regression in physics are two high-dimensional supervised learning problems. These two problems share several similarities that make our study all the more interesting :

- Atomic neighborhoods in atomic systems can be seen as the equivalent of patches in images.
- Invariance properties have driven the design of descriptors of atomic neighborhood or image patches: rotation and translation invariance for atomic neighborhoods, scale, lightening, and deformation invariance for image patch patterns.
- The dimension of image patches and atomic neighborhoods descriptors presented in the literature is $\sim 10^2$, which is significantly lower than the initial dimension but is still not a low dimension, i.e., a dimension ≤ 10.
- These two problems are multi-scale problems. Energy results from interactions at different scales, e.g., ionic and covalent bonds at short range, Van-der-Waals interactions at the mesoscale, and long-range Coulomb interactions. One can classify an image using texture information at a small scale, pattern information at a larger scale, or shape information at the image scale.

These two problems also have notable differences that make the study complementary:

- One problem is about regression, the other is about classification.
- The atomic positions in physics are in the continuous 3D space. Natural images are sampled on a finite grid ranging from 32×32 to 1024×1024 pixels typically.
- In physics, distances are absolute: the distance between an atom and its nearest neighbor is in the order of 1Å. In image classification, the number of pixels that separate two components (e.g., two edges) of an object in the image depends upon the sampling grid.
- In terms of performance, kernel methods are on par with CNNs in energy regression in physics. In image classification, CNNs are far above kernel methods.

Alongside the release of open-source Python implementation of the presented techniques[1], the main contributions of this thesis are the following. In the field of energy regression in physics:

- The Solid Harmonic Scattering Transform [Eickenberg et al., 2017] is an atomic environment descriptor relying on scale separation. It is implemented as a multi-scale convolutional neural network involving a cascade of wavelet transforms. We study this descriptor comparatively with existing local descriptors. We focus on two different energy regression benchmarks. One is about small organic molecules energy regression. The other is about long-range energy regression in graphite solid. Results show that local descriptors are very efficient to regress energies in these two cases.

- We present a method to regress the vibrational entropy in atomic systems, a free energy component. We study comparatively the Solid Harmonic Scattering transform and a local descriptor called Angular Fourier Series (AFS) [Bartók et al., 2013]. Results show a significantly better predictive power of the local AFS descriptor. Moreover, the presented regression model trained on small systems can extrapolate to very large atomic systems.

In the field of image classification:

- We present a structured convolutional neural network architecture that classifies images using small patches. Performances are comparable with BagNet [Brendel and Bethge, 2019], a state-of-the-art CNN that relies on patch-separation.

- We demonstrate that one can classify images with a non-trivial accuracy using K-nearest-neighbors computations between raw image patches solely. The presented technique significantly outperforms existing non-learned visual representations such as Scattering Transform [Bruna and Mallat, 2013] on CIFAR-10 and ImageNet databases with a linear classifier.

We end this dissertation with an opening on a different topic. Algorithms developed for supervised learning techniques have been used recently in the field of artistic creation. For example, image classification algorithms can be used to perform artistic style transfer on images [Gatys et al., 2015]. Image generation algorithms were used to generate "artistic" images. Some of these generated images are sold on the art market and shown during a public exhibition in the *Centre Pompidou* in Paris. These phenomena raise new questions about the notions of creation and creativity. We study this notion of creativity from the human-machine interaction perspective. The contributions of this thesis regarding this topic are the following:

- We propose a new form of human-machine interaction consisting of interactive rounds of creation between artists and an algorithm on a canvas. Alongside fostering creativity, it is a case-study of painter-algorithm interactions on a canvas.

- We present interactive painting processes in which a painter and various neural style transfer algorithms interact on a real canvas. We study and characterize the influence

[1] https://github.com/louity

of algorithms' outputs on the final canvas. This allows describing the creative agency of the algorithm in our interactive painting experiments.

In the following of this introduction, we review image classification and energy regression techniques focusing on local methods. Then we present the contributions of this dissertation.

1.3 Image classification

Image classification consists in assigning to an input image one class of a given set of classes. This set of classes can be a digit from 0 to 9 for handwritten digit recognition or a set of object classes (e.g., car, truck, table ...). The publication of large annotated image databases has fostered the development of image classification techniques. It began in the 90s with the MNIST handwritten digit image database [LeCun et al., 1990, 1998]. In 2004, the Caltech101 database [Fei-Fei et al., 2004] containing 9,146 images divided into 101 object classes was the first large database of objects. It was followed in 2005 by the PascalVOC image databases and challenges published every year between 2005 and 2012, containing a few thousand images divided into 19 object classes. A major shift was the publication of the ImageNet database [Russakovsky et al., 2015] in 2010, with the corresponding ImageNet Large Scale Visual Recognition Challenge (ILSVRC) held every year between 2010 and 2017. Major improvements in image classification, such as AlexNet [Krizhevsky et al., 2012], VGG [Simonyan and Zisserman, 2014], and ResNet [He et al., 2016] were presented on the occasion of this competition.

1.3.1 Patch-based image classification

Patch decomposition is a standard in image processing. The JPEG standard for image compression [Wallace, 1992] relies on the decomposition of the image into 8×8 patches. Early visual texture synthesis models [Efros and Leung, 1999] and image inpainting methods [Criminisi et al., 2004] are based on patch decomposition. SIFT descriptors of patches [Lowe, 2004] were at the core of image retrieval, image stitching, and 3D modeling. These descriptors are invariant to translations, rotations, and scaling transformations in the image domain and robust to moderate perspective transformations and illumination variations.

Before the supremacy of deep networks, state-of-the-art image classification methods were using patch descriptors. The ISLRVC 2010 challenge was won with 71.8% top-5 accuracy using non-linear coding [Wang et al., 2010] upon SIFT and Local Binary Patterns [He and Wang, 1990] patch descriptors, followed by a linear SVM classifier. Sánchez et al. [2013] have won the ISLRVC 2011 challenge with 74.3% top-5 accuracy using patches descriptors encoded with Fisher Vectors (FV) [Perronnin et al., 2010]. This classification pipeline relies on 24×24 patches extraction followed by a SIFT and Local Color Statistics (LCS) [Ah-Pine et al., 2008] encoding. These SIFT and LCS descriptors are encoded with a finite-dimensional approximation of the Fisher Kernel and a normalization step. Finally, a linear SVM classifier is applied. Perronnin et al. [2010] justify the use of patches as *"a standard approach to describe an image for classification."* It is not a hypothesis made on the classification function on purpose.

1.3.2 Convolutional neural networks

Krizhevsky et al. [2012] have won the ISLRVC 2012 challenge with $\sim 85.1\%$ top-5 accuracy using a convolutional neural network (CNN) [LeCun et al., 1989] called AlexNet. This method differs from the previous image classification techniques like Fisher Vectors as it does not rely on patch separation. It processes the input image as a whole.

Neural networks CNNs are a particular type of neural network. From a formal point of view, a neural network architecture is a class of parametric functions $\{F_\theta, \theta \in \mathbb{R}^P\}$ where P is the number of parameters. A neural network is a parametric function $F_\theta : x \in \mathbb{R}^d \mapsto F(x)$. The output $F(x)$ is computed with a succession of linear and non-linear operations:

$$F_\theta(x) = W_L \rho \left(W_{L-1} \left(\dots \rho \left(W_0 x \right) \right) \right) \tag{1.3}$$

where $W_l \in \mathbb{R}^{h_l \times h_{l+1}}$ are linear operators, called the weights of the neural network.

The following hyper-parameters define a neural network architecture:

- L: the depth of the neural networks.
- ρ: the non-linear function.
- h_l : the width of the l^{th} layer of the neural network. h_{L+1} is the dimension of the output of the neural network.

A neural network architecture defines a fixed class of parametric functions, parametrized by $\theta = \{W_0, \dots, W_L\}$.

Convolutional neural networks In a standard neural network, an image x with C channels and $H \times W$ pixels is considered as a vector of $\mathbb{R}^{C \times H \times W}$. The first linear operator W_0 maps $\mathbb{R}^{C \times H \times W}$ to $\mathbb{R}^{h_1}$. It is called a "fully-connected" linear operator since each coordinate of $W_0 x$ depends upon all the coordinates of the input x.

Convolutional neural networks (CNNs) are a particular class of neural network architecture. In CNNs, "fully-connected" linear operators are replaced with discrete convolution operators (see Appendix for an introduction to discrete convolutions). The hidden size h_l of the fully-connected operator is replaced by the number of channels C_l and spatial size S_l of the convolutional operator W_l. In addition to convolutions, a pooling operator P (local average pooling, max-pooling) allows reducing the layers' spatial size progressively. An example of CNN architecture is shown in Figure 1.1.

Receptive field in CNN The CNN's receptive field at a layer l is the region's size in the input image upon which depend a pixel at the layer l. For example, a single convolution operator's receptive field is equal to the spatial size S of the convolutional operator W_0. The receptive field of a cascade of two convolution operators of sizes S_1 and S_2 is equal to $S_1 + S_2 - 1$. In CNN, non-linear operations are usually pointwise, and they do not change the receptive field. On the contrary, pooling operations increase the receptive field. In usual CNNs such as AlexNet [Krizhevsky et al., 2012], VGG [Simonyan and Zisserman, 2014], and ResNet [He et al., 2016], the receptive field of the last convolutional layer is equal to the whole image size.

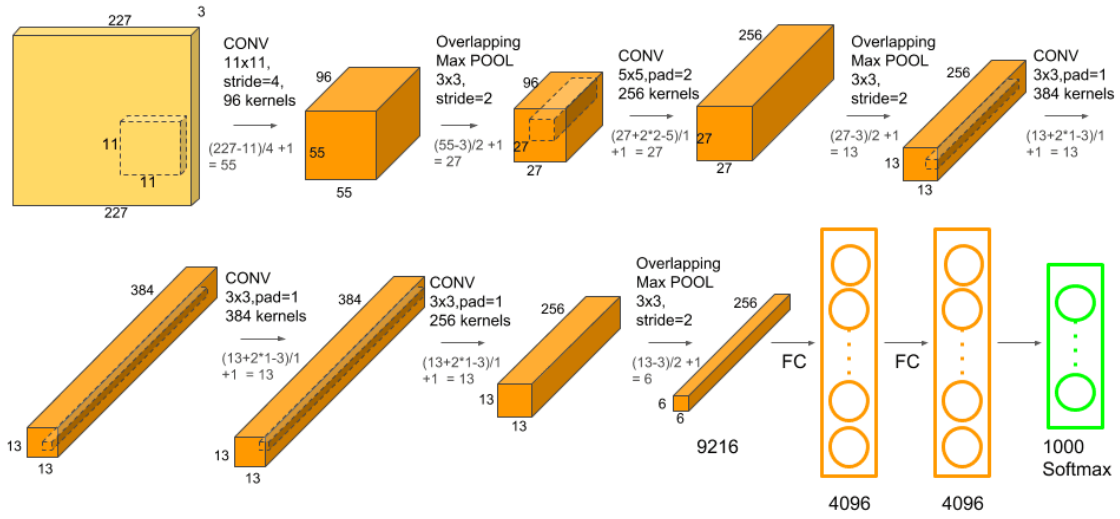


Figure 1.1: The architecture of the AlexNet CNN [Krizhevsky et al., 2012]. Convolution operations are represented by an arrow with CONV, the spatial size of the convolution, the stride of the convolution, and the number of convolutional kernels (i.e., filters). Convolutions are followed by a ReLU non-linearity which is not represented in the figure. Max-pooling operations are represented by an arrow with Max POOL, the spatial size of the region, and the stride. Fully-connected operators linear operators are represented by an arrow with FC and are followed by a ReLU non-linearity.

1.3.3 Patch-based convolutional neural networks

BagNets [Brendel and Bethge, 2019] have shown that an image can be classified as a bag-of-patches with a competitive classification accuracy (87.5% top5 accuracy on ImageNet). A BagNet is a simple variant of the ResNet-50 architecture [He et al., 2016]. Replacing most of the ResNet's 3×3 convolutions by 1×1 convolutions, the receptive field of the last convolutional layer is limited to $Q \times Q$ pixels, with $Q \in \{9,\ 17,\ 33\}$. With a global average pooling before the linear classification, this CNN's classification decision is a sum of classification decisions over $Q \times Q$ image patches. It demonstrates that close to state-of-the-art performance can be obtained with a patch separation hypothesis on the classification function.

More recently, Dosovitskiy et al. [2021] replaced the usual convolutional architecture with a Transformer architecture for image classification. This architecture was initially developed by Vaswani et al. [2017] for natural language processing tasks. In this technique, image patches play the role of words in a sentence. With a few adaptations from text to image, the accuracy obtained is 88.4% **top1** (top5 accuracy not mentioned) at a fraction of the computational cost of CNNs that have similar performances. This technique still uses positional embedding to encode the patch position, but an ablation study shows that removing this positional information results in an accuracy drop of about 4%. Without this positional embedding, the classification

decision does not consider the spatial ordering of the patches. This architecture treats the image as a "bag-of-patches."

These two results show that even in the framework of convolutional neural networks, local methods based on image patches can obtain very good accuracy. They encourage the presented study of local methods in the context of image classification.

1.4 Energy regression in physics

1.4.1 Potential energy surface

Energy regression in physics consists in fitting a deterministic function. This function, the potential energy surface (PES), maps the atomic positions to the energy of a set of atoms. The Born-Oppenheimer approximation in quantum mechanics guarantees the existence of this function. The potential energy of a system is the lowest eigenvalue of an eigenvalue problem in a functional space. Functions in this functional space map $\mathbb{R}^{3N_e}$ to $\mathbb{C}$, where N_e is the number of electrons of the system, usually greater than the number of atoms. The number of basis functions needed to represent a function in this functional space grows exponentially with the dimension $3N_e$. There is hence another form of curse of dimensionality in this problem. Since one can not represent these functions numerically, one can not solve the eigenvalue problem. One can thus not access the actual value of this function.

Electronic structure methods such as Hartree-Fock [Hartree, 1928, Fock, 1930] or Density Functional Theory [Hohenberg and Kohn, 1964, Kohn and Sham, 1965] give access to approximate values of this function. However, they require solving a complex optimization problem whose computational cost scales like $\mathcal{O}(N^2)$ to $\mathcal{O}(N^3)$. This computational cost is prohibitive for systems with a large number of atoms. One can use a coarser approximation with the use of a potential function.

1.4.2 Empirical potentials

Empirical (or parametric) potentials are functions F of the atomic positions $r_1, \ldots, r_{N_a}$. They are meant to avoid the computational cost of electronic structure methods and are generally less accurate. The simplest potentials only consider pairwise interactions and are called pair-potential. For example, the widely used Lennard-Jones potential [Jones, 1924] is a pair-potential. Its expression is

$$F_{\text{LJ}}(r_1, \ldots r_{N_a}) = 4\epsilon \sum_{i,j}^{N_a} \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right]$$

where $r_{ij} = \|r_i - r_j\|$ is the distance between atoms i and j, ϵ is a depth parameter, and σ is the distance at which the potential crosses zero. The attractive term proportional to $1/r^6$ in the potential comes from the Van-der-Waals forces' scaling.

The Stillinger-Weber potential [Stillinger and Weber, 1985] originally developed for Silicon atoms considers pairs and triplets interactions. An important class of potentials for metals is

based on the embedded-atom model (EAM) [Daw and Baskes, 1984]. They have the following form:

$$F(r_1, \ldots, r_{N_a}) = \sum_i^{N_a} f_i \left(\sum_j \rho(r_{ij}) \right) + \frac{1}{2} \sum_{i,j}^{N_a} F_2(r_{ij})$$

where F_i is called an embedding function that take as input a sum of pseudo electron densities $\rho(r_{ij})$. The term F_2 is usually a repulsive pair (two-body) potential.

1.4.3 Machine learning potentials

According to Behler [2016], a paradigm shift is taking place in the development of potentials. While empirical potentials derive from physical approximations, the design of new potentials is treated as a supervised learning problem. These new potentials, called Machine Learning (ML) potentials, are functions of the atomic positions like empirical potentials. The construction of a machine learning potential follows two steps:

1. **descriptor**. A descriptor $\Phi(r_1, \ldots, r_{N_a})$ of the atomic position is computed. It is supposed to be invariant to translations and rotations of the atomic positions since the energy is invariant to these transformations. The descriptor has to be differentiable with respect to the atomic positions $(r_1, \ldots, r_{N_a})$ as the derivative of the energy corresponds to forces.

2. **regression**. A supervised learning regression technique, e.g., linear regression, kernel regression, is employed to regress the energy. Using linear regression on top of a descriptor $\Phi(r_1, \ldots, r_{N_a})$, the potential F has the following form:

$$F(r_1, \ldots, r_{N_a}) = \langle \theta, \Phi(r_1, \ldots, r_{N_a}) \rangle + b$$

One of the first descriptors presented in the literature is the Atom-centered Symmetry Functions (ACSF) by Behler and Parrinello [2007]. This descriptor is a local descriptor. Behler and Parrinello [2007] hypothesize that the total energy is the sum of local energy contributions

$$E = \sum_{i=1}^{N_a} E_i$$

where E_i is the local contribution of the atom i to the global energy E. The energy E_i depends upon the neighborhood $\mathcal{N}_i$ of the atom i, which is a ball of center r_i and of radius R_{cut}, called the cutoff radius. This neighborhood $\mathcal{N}_i$ is described with a set of atom-centered symmetry function (ACSF) descriptors. These functions take as input the native variables, i.e., the positions r_j of the neighboring atoms j in the neighborhood $\mathcal{N}_i$ of the atom i.

There are two types of symmetry functions:

- the radial functions g_{rad} defined by

$$g_{\text{rad}}(\mathcal{N}_i) = \sum_{j \in \mathcal{N}_i} \exp(-\eta(r_{ij} - r_s)^2) f_{\text{cut}}(r_{ij})$$

- the angular function g_{ang} defined by

$$g_{\text{ang}}(\mathcal{N}_i) = 2^{1-\xi} \sum_{i \neq j \in \mathcal{N}_i} (1 + \lambda \cos(\theta_{ijk}))^{\xi} \exp(-\eta(r_{ij}^2 + r_{ik}^2 + r_{jk}^2)) f_{\text{cut}}(r_{ij}) f_{\text{cut}}(r_{ik}) f_{\text{cut}}(r_{jk}),$$

where f_{cut} is a radial cutoff function that vanished for $r > R_{\text{cut}}$, r_{ij} is the pairwise distance between atom i and atom j, and θ_{ijk} is the angle of the triplet of atoms i, j, k centered on i. η, ξ, λ and r_s are parameters that take several different values and yield different symmetry functions. The ACSF descriptor is the concatenation of all the symmetry functions. Typically 50 to 100 symmetry functions are used per atom differing in the values of the parameters η, ξ, λ and r_s .

ACSF descriptor provides a rotation and translation invariant description of the atomic systems as it depends on the pairwise distances r_{ij} and angles θ_{ijk}. It differentiable w.r.t. atomic position since it involves differentiable functions. It is invariant to any permutation of atoms of the same species in the environment.

Behler and Parrinello [2007] combined this ACSF descriptor with a neural network regression to model the energy of bulk silicon atomic systems.

1.5 Contributions in energy regression in physics.

After the seminal work on atomic descriptors by Behler and Parrinello [2007] to regress the potential energy surface of Silicon, Rupp et al. [2012] published the QM7 database of $7,165$ organic molecules with their corresponding atomization energies. Then Ramakrishnan et al. [2014] published the QM9 database with the atomization energies of $130,000$ small organic molecules. As for image classification, the publication of these databases fostered the development of new machine learning potentials. These machine learning potentials' novelty lies in the proposed descriptor functions rather than in new regression techniques.

The first descriptor published together with the QM7 database [Rupp et al., 2012] was the Coulomb Matrix. The majority of the descriptors published afterwards are local and rely on neighborhood separation. For example, the Bag-of-Bonds (BoB) descriptor [Hansen et al., 2015], the Bonds, Angles, Machine-learning (BAML) descriptor [Huang and Von Lilienfeld, 2016], the Histograms of Distances, Angles and Dihedral angles (HDAD) [Faber et al., 2017] or the Smooth overlapping atomic positions (SOAP) descriptor [Bartók and Csányi, 2015] assume that the energy is a sum of local contributions

$$E = \sum_{i=1}^{N_a} E_i \ .$$

The description of atomic neighborhoods is limited to a certain cutoff distance like for the Atom-centered symmetry functions.

Hirn et al. [2016] proposed a different approach based on scale separation and inspired by the Scattering Transform [Mallat, 2012], an image descriptor for image classification. Hirn et al. [2016] introduced the Wavelet Scattering for energy regression. The first step transforms the atomic positions into a fictitious image. Then they compute a multi-scale descriptor of this

image. Linear regression with this multi-scale descriptor predicts the energy of the molecule. Hirn et al. [2016] restricted the application to the 454 planar molecules of the QM7 database and the 4357 planar molecules of the QM9 database. This resctriction to planar molecules allowed the use of 2D images and simplified computations. This technique's regression accuracy is competitive with the Coulomb Matrix with Kernel Regression [Rupp et al., 2012] despite a much smaller training set.

Eickenberg et al. [2017, 2018] extended this technique to three-dimensional molecules. It results in the Solid Harmonic wavelet scattering transform, a multi-scale descriptor of three-dimensional atomic systems. In a nutshell, this descriptor separates scale information into different coefficients. Contrarily to BoB [Hansen et al., 2015], BAML [Huang and Von Lilienfeld, 2016], HDAD [Faber et al., 2017] or SOAP [Bartók et al., 2013] descriptors, the description of the interactions is not limited to a certain cutoff radius. The interactions between different scales are also described in the so-called *second-order* coefficients of this descriptor.

Energy is a complex property of the atomic state. It depends on various interactions in the system. These interactions occur at different scales, e.g., ionic and covalent bonds at short scale, Van-der-Waals interactions at the mesoscale, and long-range Coulomb interactions. Is this Solid Harmonic wavelet scattering transform, based on scale separation, more appropriate than existing local descriptors like SOAP or HDAD? Is it more efficient to regress molecules' energy, solids' energies, or solids' vibrational entropy?

1.5.1 Efficiency of local methods for energy regression

This contribution has been published in part in Eickenberg, Exarchakis, Hirn, Mallat, and Thiry [2018].

We study the Solid Harmonic Scattering transform [Eickenberg et al., 2017, 2018] and local descriptors for molecules and solids energy regression performances for the regression comparatively in this first contribution. We first present the Solid Harmonic Scattering Transform descriptor. We derive properties of this descriptor and study the effect of sampling errors on these descriptors' numerical computations.

Then we study comparatively Solid Harmonic Scattering transform and local descriptors for the regression of small organic molecules. Results with Solid Harmonic Scattering transform are competitive with the state-of-the-art on the QM9 database with 130,000 molecules. Local descriptors like HDAD [Faber et al., 2017] and SOAP [Bartók et al., 2013, Barker et al., 2017] descriptors obtain comparable performances.

Hence, energies of QM9 molecules happen to be essentially local energies. Is the multi-scale separation more accurate than other methods on problems with long-range energy terms?

To have the beginning of an answer to this question, we consider a particular type of atomic system with long-range interactions. We build a database of graphene systems, i.e., solids of carbon atoms at temperature $\sim$ 1500 Kelvin. Graphene systems are known to have long-range Van-der-Waals interactions [Chung, 2002, Novoselov et al., 2004, Bolotin et al., 2008]. We compute the energy with the Many-Body Dispersion [Tkatchenko et al., 2012] electronic structure method. Many-Body dispersion is capable of modeling such long-range Van-der-Waals energies. We hypothesize that local descriptors can not directly capture the

total energy of such systems.

On this database, we compare two machine learning potentials, each relying on a different form of separation:

1. A multi-scale Solid Harmonic Scattering Transform descriptor with linear regression. This potential relies on a multi-scale separation.

2. A local SOAP descriptor [Bartók et al., 2013] with kernel regression plus a long-range pair potential. This potential relies on spatial separation in short-range and long-range terms. Long-range energy is first regressed with a simple pair-potential. The remaining energy is regressed with a local SOAP descriptor.

Numerical results show that these two techniques perform similarly. Long-range forces are efficiently captured by a pair potential in the second technique. The short-range part of the energy is efficiently captured with a single-scale local descriptor (SOAP). This shows that a simple two-scales separation with short-range and long-range energies is sufficient to regress the system's energy. Systematic multi-scale separation does not significantly improve these results, while spatial separation offers a more interpretable result.

Numerical experiments on the QM9 database, the construction of the graphene database and the comparative study are developed in chapter 2.

1.5.2 A local method for vibrational entropy regression

The present contribution has been published in Lapointe, Swinburne, Thiry, Mallat, Proville, Becquart, and Marinica [2020].

In physical systems at a constant temperature T, the relevant energy is the free energy A. It indicates whether a physical process is likely or not in the system. It is defined as the energy $F(r_1, \dots, r_{N_a})$ minus the temperature T times the entropy $S(r_1, \dots, r_{N_a})$.

$$A(r_1, \dots, r_{N_a}) = F(r_1, \dots, r_{N_a}) - TS(r_1, \dots, r_{N_a}).$$

The vibrational entropy represents the entropy $S(r_1, \dots, r_{N_a})$ under the harmonic approximation. It is defined only for atomic positions $(r_1^m, \dots, r_N^m)$ at a local minimum of the energy.

The computational cost of vibrational entropy scales like $\mathcal{O}(N_a^3)$, which is prohibitive for systems with typically $N_a > 10^4$. Speeding up this computation using a vibrational entropy regression technique is the first motivation.

In the context of maximum-entropy stochastic process modeling, Scattering Transform has shown promising results [Bruna and Mallat, 2019, Zhang and Mallat, 2019]. Moreover, the scattering transforms' multi-scale separation is a key aspect for this modeling as it allows to correlate non-linearly different scales. The vibrational entropy presented here is a different quantity, but there are connections between these definitions. Is the multi-scale Solid Harmonic Scattering Transform efficient for vibrational entropy regression? How does it compare with existing local descriptors?

In this contribution, we present an efficient regression technique of vibrational entropy in iron crystals. It is the first published technique for vibrational entropy regression in bulk

atomic systems, to the best of our knowledge. We test two descriptors based on two types of separation :

1. The first one is a Solid Harmonic Scattering descriptor based on scale separation.

2. The second one is the Angular Fourier Series (AFS) local descriptor [Bartók et al., 2013].

Using linear regression, the AFS descriptor shows a superior predictive power on various large systems of body-centered cubic structures of iron atoms with defects. The presented method extrapolates on very large atom systems. Indeed, formation entropies in a range of 250 k_B are predicted with less than 1.6 k_B error from a training database whose formation entropies span only 25 k_B (training error less than 1.0 k_B).

Numerical experiments and implementation details are described and discussed in chapter 3.

1.6 Contributions in image classification.

The different studies we have performed in the field of energy regression suggest that one can efficiently capture long-range energy terms with pair potentials. The difficulty of energy regression lies in the local geometries whose modeling does not necessarily require scale separation. Descriptors of local atomic neighborhoods perform well on this task.

Like energy regression in physics, image classification is a multi-scale problem. There are typical edges and texture patterns at the patch level and global shapes at the image level for a given object. There is a natural analogy between atomic neighborhoods in atomic systems and patches in images.

The findings presented above suggest that the scale separation is not necessary to have a low energy regression error in physics. We thus ask the same question for image classification. Is a multi-scale approach necessary to have a good performance on standard classification benchmarks like ImageNet? With the BagNets, Brendel and Bethge [2019] have demonstrated that a competitive accuracy (87.5% top5) can be obtained with a CNN relying on patch separation. We further study this aspect in the following contributions.

1.6.1 A structured CNN for patch-based image classification

The present contribution has not been published yet. It served as preliminary experiments for the publication by Zarka, Thiry, Angles, and Mallat [2019].

Oyallon [2017] introduced a structured convolutional network architecture that reaches AlexNet accuracy (79.6 % top-5) on the ImageNet 2012 database. It is first composed of a scattering transform, [Mallat, 2012, Bruna and Mallat, 2013], which can be seen as an encoding of image patches of size $\sim 16 \times 16$. This scattering encoding is then fed to a cascade of 1×1 convolutions. It ends up with a big two-hidden layer neural-network classifier. Because of the use of the two-hidden layer classifier, this architecture does not rely on patch separation.

Can we remove this big classifier and obtain a competitive accuracy? Can we propose a structured convolutional neural network architecture that relies on patch separation?

In this contribution, we positively answer this specific question. We use a Scattering Transform encoding of $\sim 16 \times 16$ patches. We use it directly to have $\sim 16 \times 16$ patches encoding, or we concatenate Scattering descriptors to encode 32×32 image patches. On top of this descriptor, we learn a non-linear encoding for classification. This encoding is implemented with a sequence of N layers of 1×1 convolutions with C channels. Finally, a global spatial average pooling is performed, followed by a linear classifier. Like in the BagNets, the global spatial average pooling ensures the patch-separation of the classification decision.

With this architecture, we obtain an accuracy of 78.8% and 84.5% top-5 on ImageNet with patch size 16×16 and 32×32 respectively. It is comparable with the BagNet accuracies of 81.2% and 87.5% top5 with patch sizes 17×17 and 33×33 respectively. We obtain a competitive with a relatively shallow 10-layers encoding compared to the 50 layers encoding of the BagNet. It also suggests that the spatial component does not need to be learned to have competitive accuracy. We perform an ablation study to analyze the relative importance of the different parameters for classification accuracy.

The proposed architecture and numerical experiments are described and discussed in chapter 4.

1.6.2 Image classification with patches K-nearest-neighbors.

The present contribution has been published in Thiry, Arbel, Belilovsky, and Oyallon [2021].

One can accurately classify an image using patch information. But one has to learn a non-linear patch encoding to have good image classification performance. This learned representation is hard to understand and interpret. Hence, we have few insights into the reasons for the success of patch-based methods.

To understand this success, we first look at the performances of conceptually simple image classification techniques based on patches. The simplest baseline we can think of is a K-nearest-neighbor classifier. Do we get a non-trivial performance using a patch-based K-nearest-neighbor-based classifier? How does it compare with other predefined visual representations for classification?

In this contribution, we present an image classifier based on patches K-nearest-neighbors. We compute the K-nearest Neighbors of each patch of an image and a fixed dictionary of patches $\mathcal{D}$, with size $|\mathcal{D}|$ using the Mahanalobis Euclidean distance [Chandra et al., 1936]. For a fixed dataset, this dictionary $\mathcal{D}$ is obtained by uniformly sampling patches from images over the whole training set. This neighborhood representation is then fed to a linear classifier. On CIFAR-10, we obtain an accuracy of 86.6%, in the performance range of sophisticated convolutional kernel methods. On ImageNet, we obtain a non-trivial accuracy of 54.7%, outperforming predefined visual representation such as Scattering Transform [Mallat, 2012, Bruna and Mallat, 2013]. As such, this technique is a new baseline for object recognition without representation learning methods.

According to Beyer et al. [1999], *" scenario, where high-dimensional nearest-neighbors are meaningful, occurs when the underlying dimensionality of the data is much lower than the actual dimensionality."* This performance we obtain with K-nearest-neighbors suggests that the natural image patches have a low underlying dimension. We study this aspect using

existing dimensionality measures.

We present this method, the numerical experiments performed, and study the patches' low-dimensional properties in chapter 5.

1.7 Convolutional neural networks for artistic creation

In the last part of this thesis, we drive away from the central question about local methods in energy regression and image classification to study algorithms' applications in artistic creation. The VGG convolutional neural network initially developed by Simonyan and Zisserman [2014] for image classification has been used by Gatys et al. [2015] to perform artistic style transfer on images. The success of CNNs for image classification fostered the development of CNNs for image generation [Goodfellow et al., 2014]. These algorithms were used to generate "artistic" images. Some of these generated images are sold on the art market and shown during a public exhibition in the Centre Pompidou in Paris.

The use of these algorithms and the emergence of a new AI-art movement raise several questions on creativity and art. To study the notion of creativity, we present two interactive creation processes involving artists and algorithms. The experiments allow us to characterize the creative agencies in such an interactive process.

1.7.1 Dialog on a canvas with a machine

The first artist-algorithm interaction process we present consists of a succession of interactive rounds of creation between artists and machines. The algorithm and the artist repetitively paint on the canvas with a different color one after the other. After Charly and Tina have drawn a stroke, the algorithm partially completes the drawing using machine learning algorithms. The completion is projected directly on the canvas, and the artists are free to insert or modify it.

Thanks to its simplicity, this process is a powerful case study of the creative agencies in an interactive process. We present this work in the first part of chapter 6. It has been published in Cabannes, Kerdreux, Thiry, Campana, and Ferrandes [2019].

1.7.2 Neural style transfer with artists

The second artist-algorithm interaction process we present consists of an interactive painting process in which a painter and various neural style transfer algorithms interact on a real canvas. The principle of style transfer allows the painter to interact with his own style. Moreover, the generated images' diversity was perceived as a source of inspiration for human painters, portraying the machine as a computational catalyst.

We present this work in the second part of chapter 6. This work has been published in Kerdreux, Thiry, and Kerdreux [2020].

Chapter 2

Efficiency of local methods for energy regression

After the seminal work by Behler and Parrinello [2007] for Silicon, Rupp et al. [2012] initiated the development of small organic molecules' energy regression. They published a database of $7,185$ organic molecules whose energies were computed using density functional theory. They proposed to regress these energies using a kernel ridge regression and a Coulomb matrix descriptor and obtained promising results [Rupp et al., 2012]. Ramakrishnan et al. [2014] published the QM9 database with the atomization energies of $130,000$ three-dimensional organic molecules containing up to 9 non-hydrogen atoms. A variety of descriptors have been proposed to regress QM9 molecules' energies. A large part of these descriptors is based on spatial separation. For example, Bag of Bonds [Hansen et al., 2015], the Histogram of distances, angles, and Dihedral angles [Faber et al., 2017] and Smooth Overlapping Atomic Positions [Bartók et al., 2013] local descriptors represent a neighborhood of radius 3 to 5 $\dot{A}$ around the atoms.

Yet, we know that energy is not a local quantity in general. Energy depends on interactions at different scales in the system, e.g., ionic and covalent bonds at short range, Van-der-Waals interactions at the mesoscale, and long-range Coulomb interactions. Can we build a descriptor that describes the geometry of the systems at different scales? How does such descriptor compare with local descriptors like SOAP to regress molecule's and solid's energies?

Hirn et al. [2016] proposed a descriptor of two-dimensional molecules that separates scale information in different coefficients. The competitive results they obtained motivated this technique's three-dimensional extension. This three-dimensional extension is the Solid Harmonic wavelet scattering transform [Eickenberg et al., 2017, 2018].

In this chapter, we study the efficiency of local and multi-scale methods for energy regression in molecules and solids. Solid Harmonic wavelet scattering transform is one of the few descriptors relying on multi-scale separation in the literature. Hence we focus on this descriptor for multi-scale methods. We focus on the SOAP descriptor for local methods as it has been successfully applied to regress molecules' [De et al., 2016] and solids' [Szlachta et al., 2014, Dragoni et al., 2018, Fujikake et al., 2018] energies.

Solid Harmonic wavelet scattering transform is based on the generation of a continuous image that we call density. This continuous density has to be sampled for numerical compu-

tations. Sampling errors cancel the crucial rotation and translation invariance properties. We study these errors and show how to control them using Fourier analysis tools in the first part of this chapter.

In the second part of this chapter, we compare the Solid Harmonic Scattering transform and local SOAP descriptor's predictive performance [Bartók et al., 2013]. We focus on two energy regression benchmarks. The first one is the QM9 database benchmark of small organic molecules [Ramakrishnan et al., 2014]. In classical chemistry, leading terms in the energy of molecules tend to be in the chemical bonds [Flowers et al., 2018], hence in local interactions. However, QM9 database molecules' energies are computed using a theory derived from quantum mechanics, and the resulting energy is not supposed to be localized. Energy regression results show that Solid harmonic scattering and local methods achieve comparable mean absolute errors (MAE) in the order of 0.5 kcal/mol. Local descriptors like HDAD [Faber et al., 2017] and SOAP [Bartók et al., 2013, De Clercq et al., 2016] descriptors obtain comparable performances, suggesting that QM9 energies are indeed local. We introduce a second benchmark consists of graphene solids (Carbon atoms) with long-range Van-der-Waals interactions [Chung, 2002, Novoselov et al., 2004, Bolotin et al., 2008]. Since energies have a long-range component, one can assume that local descriptors can not capture such systems' total energy. On this system, we compare two machine learning potentials, each relying on a different form of separation:

1. A multi-scale Solid Harmonic Scattering Transform descriptor with linear regression. This technique relies on a multi-scale separation.

2. A combination local SOAP descriptor [Bartók et al., 2013] with kernel regression plus a long-range pair potential. This technique relies on spatial separation.

Numerical results show that these two techniques perform similarly. Simple pairwise potentials capture efficiently long-range interactions. The short-range part of the energy is efficiently captured with the combination of two SOAP local descriptors. This demonstrates that a simple spatial separation in short-range and long-range energies is sufficient to regress long-range Van-der-Waals energies. Despite the atoms' interactions, one can capture complex interactions with a local environment description despite the multi-scale nature of the atoms' interactions.

The present chapter is organized as follows. We first present the continuous Solid Harmonic Scattering Transform and derives its invariance properties. We study the errors due to the continuous Solid Harmonic Scattering Transform sampling and show how to control them. Then we present and discuss the numerical experiments on the QM9 database. We detail the construction of the Graphene database for long-range energy regression. We present the regressions technique based on the SOAP descriptor. We then compare and discuss the performances of Solid Harmonic Scattering and SOAP descriptors on this database.

The first part of this chapter has been published in Eickenberg, Exarchakis, Hirn, Mallat, and Thiry [2018]. My personal contributions concerned the derivation of the invariance properties, the whole study of sampling error effects, and the open-source Pytorch implementation of the Solid Harmonic Scattering transform[1].

[1] https://github.com/louity/pyscatharm

The second part is based on unpublished work. It results from a collaboration with the Theoretical Chemical Physics group at the University of Luxembourg led by A. Tkatchenko and Professor Gábor Csányi's team in the Cambridge University Engineering Department. My personal contributions concerned the regression experiments protocol, the numerical experiments with the Solid Harmonic Scattering transform, and the numerical results comparison and discussion.

2.1 Continuous Solid Harmonic Scattering Transform

2.1.1 Gaussian density representation

Continuous density function

The first step in the computation of the Solid Harmonic Scattering transform is to map the atomic positions to a continuous density function. For this purpose, we use isotropic and normalized Gaussian functions g_σ

$$g_\sigma(r) = \frac{1}{(2\pi)^{3/2}\sigma^3} e^{-|r|^2/(2\sigma^2)}$$

This function is separable in Cartesian coordinates $r = (x, y, z)$

$$\begin{aligned} g_\sigma(x, y, z) &= h(x)h(y)h(z) \\ h(t) &= \frac{1}{\sqrt{2\pi}\sigma} e^{-t^2/(2\sigma^2)} \end{aligned}$$

From the persepective of Heisenberg's uncertainty principle, Gaussian functions have an optimal joint spatial-frequency localization. This is beneficial to control sampling errors.

Given the positions of the atoms $(r_1, \dots, r_{N_a})$, the continuous density ρ representing the atoms is a sum of Gaussian functions located at the atomic position

$$\rho(r) = \sum_{i=1}^{N_a} c_i g_\sigma(r - r_i). \tag{2.1}$$

c_i describes the properties of the atom i. It is typically a three-dimensional vector with the nuclear charge, the number of valence electrons, and the number of core electrons of the atom i. If there is a single atomic species in the system, c_i is simply equal to 1.

The Fourier transform of ρ is

$$\hat{\rho}(\omega) = \hat{g}_\sigma(\omega) \sum_{i=1}^{N_a} c_i e^{-\mathbf{i} r_i \cdot \omega}. \tag{2.2}$$

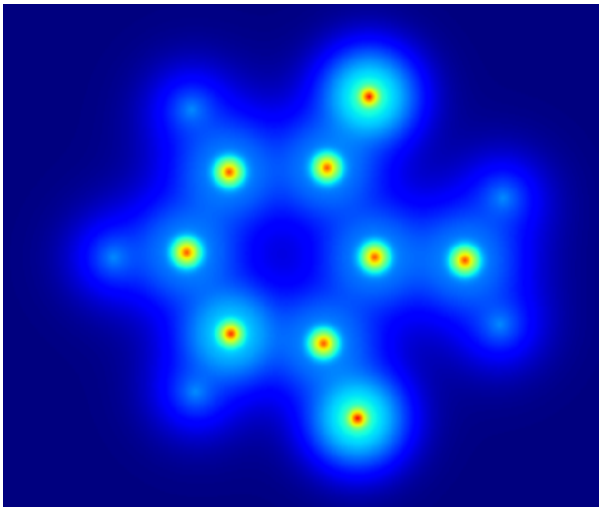


Figure 2.1: 2D slice of the 3D image ρ of the $C_7O_2H_5$ planar molecule. The slice plane corresponds to the molecule's plane.

where $\hat{g}_\sigma(\omega)$ is the Fourier transform of g

$$\hat{g}(u,v,w) = \hat{h}(u)\hat{h}(v)\hat{h}(w)$$
$$\hat{h}(s) = \frac{1}{\sqrt{2\pi}} e^{-s^2\sigma^2/2}$$

Overlapping conditions

We want the Gaussian functions g centered on the atomic positions not to overlap much. Otherwise, we lose the location information. Mathematically, we want to enforce two Gaussian functions located over two different positions to overlap with an amplitude at most equal to the overlapping precision $\epsilon_o < 1$.

Let Δ be the smallest interatomic distance. The overlapping condition yields

$$\forall r \text{ s.t. } |r| = \Delta/2, \quad \frac{|g_\sigma(r)|}{|g_\sigma(0)|} = \epsilon_o$$

After straightforward computations, we have the following condition

$$\Delta = \sigma\sqrt{-8\log(\epsilon_o)}. \tag{2.3}$$

2.1.2 Solid harmonic wavelets

The regular solid harmonic functions are regular solutions of the Laplace equation. The have a primary index $l \in \mathbb{N}$ and a secondary index $m \in [-l, +l]$. In spherical coordinate (u, θ, ϕ),

their expression is

$$R_l^m(u,\theta,\phi) = \sqrt{\frac{4\pi}{2l+1}}\, r^l Y_l^m(\theta,\phi)$$

Y_l^m are the spherical harmonics defined on the sphere $\mathbb{S}^2$

$$Y_l^m(\theta,\phi) = \sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}} P_l^m(cos(\theta))e^{im\phi}$$

P_l^m are the associated Legendre polynomials of indices (l,m).

Definition 2.1.1. *Solid harmonic wavelets are defined by multiplying a solid harmonic function by an isotropic Gaussian function. For $l \in \mathbb{N}$ and $m \in [-l,+l]$, they have the following expression*

$$\psi_{l,m} : (u,\theta,\phi) \mapsto K_l e^{-r^2/2} r^l \; Y_l^m(\theta,\phi)$$

K_l is a normalizing constant :

$$K_l = \begin{cases} \frac{1}{\pi\sqrt{2l+1}(l+1)!!} & \text{if } l \text{ is even.} \\ \\ \frac{1}{2^{\frac{l+1}{2}}\sqrt{2\pi(2l+1)}(\frac{l+1}{2})!} & \text{if } l \text{ is odd.} \end{cases}$$

Proposition 2.1.2. *The Fourier transform $\widehat{\psi}_{l,m}$ of a Solid Harmonic wavelet $\psi_{l,m}$ is also a Solid Harmonic wavelet:*

$$\widehat{\psi}_{l,m} : (\lambda,\alpha,\beta) \mapsto K_l 4\pi(-i)^l e^{\lambda^2/2}(\lambda)^l \; Y_l^m(\alpha,\beta)$$

Proof. See Appendix A. ■

To compute scattering coefficients, we will use solid harmonic wavelets. For the scattering transform, the smallest wavelet will have a width σ_w and we will use $\psi_{l,m,j}$, solid harmonic wavelets of scale j:

$$\psi_{l,m,j}(u,\theta,\phi) = \frac{1}{(2^j\sigma_w)^3}\psi_{l,m,j}\left(\frac{r}{2^j\sigma_w},\theta,\phi\right)$$

The scale j is usually an integer to have dyadic scales 2^j, but it can be a positive real value in general.

2.1.3 Continuous functional operator U

Definition 2.1.3. *For $l \in \mathbb{N}$ and $j > 0$, we define the functional operator $U_{l,j}$*

$$U_{l,j} : \rho \mapsto U_{l,j}[\rho]$$
$$U_{l,j}[\rho](r) = \left(\sum_{m=-l}^{l} |\rho * \psi_{l,m,j}|^2(r) \right)^{1/2}$$

Definition 2.1.4. *For an input function $f : \mathbb{R}^3 \to \mathbb{R}^d$, f_τ is the function f translated by $\tau \in \mathbb{R}^3$:*

$$f_\tau : r \mapsto f(r - \tau)$$

Proposition 2.1.5. *The operator $U_{l,j}$ is covariant to the input function translation.*

$$\forall f : \mathbb{R}^3 \to \mathbb{R}^d, \forall \tau \in \mathbb{R}^3, \quad U_{l,j}[f_\tau] = (U_{l,j}[f])_\tau$$

Proof. We recall that convolution and modulus operators are covariant to the translation of f

$$(f_\tau * \psi)(r) = (f * \psi)_\tau(r)$$
$$|f_\tau|(t) = |f|_\tau(r)$$

. We have thus:

$$\begin{aligned} U_{l,j}[f_\tau](r) &= \left(\sum_{m=-l}^{l} |f_\tau * \psi_{l,m,j}|^2(r) \right)^{1/2} \\ &= \left(\sum_{m=-l}^{l} |f * \psi_{l,m,j}|^2(r - \tau) \right)^{1/2} \\ &= (U_{l,j}[f])_\tau(r) \end{aligned}$$

■

Definition 2.1.6. *For a function f and a rotation R , $R.f$ is the function f rotated by R :*

$$R.f : r \mapsto f(R^{-1}r)$$

Proposition 2.1.7. *The operator $U_{l,j}$ is covariant to rotations :*

$$\forall f : \mathbb{R}^3 \to \mathbb{R}^d, \forall R \in \mathbb{SO}_3, \quad U_{l,j}[R.f] = R.(U_{l,j}[f])$$

Proof. See appendix A. ■

In the following, the density ρ is called the zero-order density. $U_{l,j}[\rho]$ is called a first-order density with indices l, j. $U_{l_2,j_2}[U_{l_1,j_1}[\rho]]$ is called a second-order density with indices l_1, j_1, l_2, j_2.

2.1.4 Examples

For example, if we take ρ to be the density of a single atom, i.e., an isotropic Gaussian function of width σ, we can compute the first-order density $U_{l,j}[\rho]$ analytically (proof in Appendix A.):

$$U_{l,j}[\rho](u,\theta,\phi) = \frac{K_l}{s_j^3} e^{-r^2/(2s_j)^2} \left(\frac{r}{s_j}\right)^l$$
$$s_j = \sqrt{(2^j\sigma_w)^2 + \sigma^2}$$

If we take ρ to be the density of a multi body system:

$$\rho(r) = \sum_k g_\sigma(r - r_k) = \sum_k g_{r_k}(r)$$

we can also compute the first order density analytically (proof in appendix):

$$U_{l,j}[\rho](u,\theta,\phi) = \left(\sum_{m=-l}^{l} \left|\sum_k (\Psi_{l,m,j})_{r_k}(u,\theta,\phi)\right|^2\right)^{1/2}$$

2.1.5 Continuous scattering coefficients

We have defined the functional operator $U_{l,j}$ which is **co**variant to translations and rotations of the density ρ. We compute scattering coefficients by simply integrating the densities over the whole space. Since the integral is **in**variant to translation and rotations of the functions, the resulting coefficients are **in**variant to translation and rotations. The densities are positive. These integrals are hence l^1 norms.

- The zero-order scattering coefficients S^0 are computed with the zero-order density

$$S^0[\rho] = \int_{\mathbb{R}^3} \rho = \|\rho\|_1$$

- The first-order scattering coefficients $S^1_{l,j}$ are computed with the first-order density $U_{l,j}[\rho]]$

$$S^1_{l,j}[\rho] = \int_{\mathbb{R}^3} U_{l,j}[\rho] = \left\|U_{l,j}[\rho]\right\|_1$$

- The second-order scattering coefficients $S^2_{l,j,l',j'}$ are computed with the second-order density $U_{l',j'}[U_{l,j}[\rho]]]$

$$S^2_{l,j,l',j'}[\rho] = \int_{\mathbb{R}^3} U_{l',j'}[U_{l,j}[\rho]] = \left\|U_{l',j'}[U_{l,j}[\rho]]\right\|_1$$

To create new invariants, we raise the densities to a power $q \in \mathbb{N}$ before integration. The power function is a point-wise function. $(U_{l,j}[\rho])^q$ is hence covariant to translations and rotations of ρ. One can see it as l^q norms to the power q. The coefficients

$$\begin{aligned} S^0_q[\rho] &= \int_{\mathbb{R}^3} \rho^q = \|\rho\|_q^q \\ S^1_{l,j,q} &= \int_{\mathbb{R}^3} (U_{l,j}[\rho])^q = \left\|U_{l,j}[\rho]\right\|_q^q \\ S^2_{l,j,l',j',q}[\rho] &= \int_{\mathbb{R}^3} (U_{l',j'}[U_{l,j}[\rho]])^q = \left\|U_{l',j'}[U_{l,j}[\rho]]\right\|_q^q \end{aligned}$$

are hence invariants to translations and rotations of ρ. The concatenation of these coefficients forms the scattering vector $S[\rho]$. $S[\rho]$ is a suitable descriptor of an atoms' system for machine learning potentials.

2.2 Discrete solid harmonic scattering transform

In practice, all computations are discrete :

1. continuous functions are sampled with step size 1, with N_x, N_y, N_z points in direction x, y, z respectively.
2. convolutions are computed as a product in Fourier space, using the Discrete Fourier Transform (DFT) to move from signal space to Fourier space.
3. Since functions are sampled both in signal and Fourier space, our functions are supposed to be $N_x \times N_y \times N_z$ periodic in signal space and $\frac{2\pi}{N_x} \times \frac{2\pi}{N_y} \times \frac{2\pi}{N_z}$ periodic in signal space.
4. Integrals over the (Fourier or signal) space $\mathbb{R}^3$ are replaced by integral over a period.

Scattering coefficients are computed with integrals over the space. To have correct values for the integrals, we need

- the support of integrand ($|\rho|$, $|U_{l,j}[\rho]|$, etc.) to be concentrated in the period over which we intergrate
- to control the aliasing effect in Fourier space due to spatial sampling.

For the first point, we only need to localize the spatial support of the Gaussian functions. For the second point, we need to use Fourier analysis tools.

2.2.1 Zero-order densities with multiple Gaussians

In our case, the density ρ is positive so $|\rho| = \rho$. In order to avoid all subsequent aliasing on the zero-order density, we want the support of the Fourier transform of ρ to be essentially be concentrated in $[-\pi, \pi]^3$, with a precision $\epsilon_a \ll 1$:

$$|\hat{\rho}(\boldsymbol{\omega})| < \epsilon_a, \ \forall \boldsymbol{\omega} \notin [-\pi, \pi]^3$$

Given the Fourier transform of ρ in eq. 2.2, it is equivalent to

$$|\hat{g}_\sigma(\omega)| < \frac{\epsilon_a}{N_a},\ \forall \omega \notin [-\pi, \pi]^3$$

. This gives

$$|\hat{h}(\pi)| = \frac{\epsilon_a}{N_a}$$

and finally

$$\sigma = \frac{1}{\pi}\sqrt{2\log\left(\frac{N_a}{\epsilon_a}\right)}$$

Hence, the width σ of the Gaussian function g is entirely determined by our aliasing condition.

2.2.2 First-order densities with a single Gaussian

If ρ is a single Gaussian, we have:

$$U_{l,j}\left[\rho\right](u,\theta,\phi) = \frac{K_l}{s_j^3} e^{-r^2/(2s_j)^2} \left(\frac{r}{s_j}\right)^l$$
$$s_j = \sqrt{(2^j\sigma_w)^2 + \sigma^2}$$

or in Cartesian coordinates

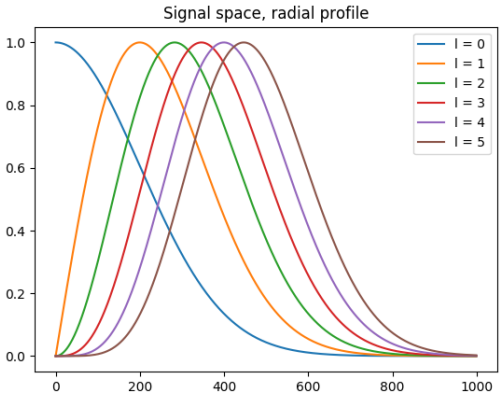
$$U_{l,j}\left[\rho\right](x,y,z) = \frac{K_l}{s_j^3} e^{-(x^2+y^2+z^2)/(2s_j)^2} \left(\frac{\sqrt{x^2+y^2+z^2}}{s_j}\right)^l$$

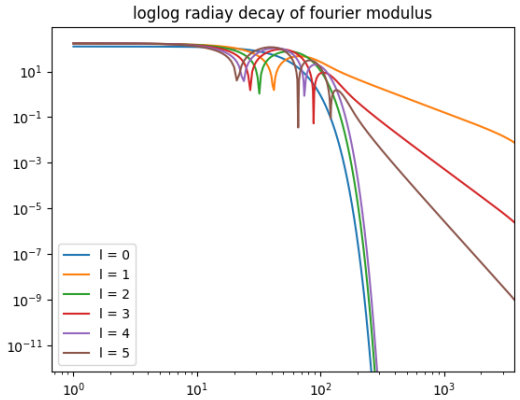
If l is even, this function is C^∞ we can compute the Fourier transform:

$$\widehat{U_{l,j}\left[\rho\right]}(\lambda,\alpha,\beta) = (is_j)^l\, e^{-s_j^2\lambda^2/2} H_l\left(s_j\lambda\right)$$

where H_l is the l^{th} Hermite polynomial.

If l is odd, their is a discontinuity of the l^{th} derivative at point 0, which cause a decay of the Fourier spectrum like $|\lambda|^{-(l+1)}$, which we can see as a line of slope $-(l+1)$ plotting the log of Fourier modulus w.r.t. $\log(\lambda)$. We can numerically see these effects :





So for a single Gaussian density ρ, the first order density $U_{l,j}[\rho]$ is much more likely to be aliased for odd l than for even l.

2.2.3 First-order densities with multiple Gaussians

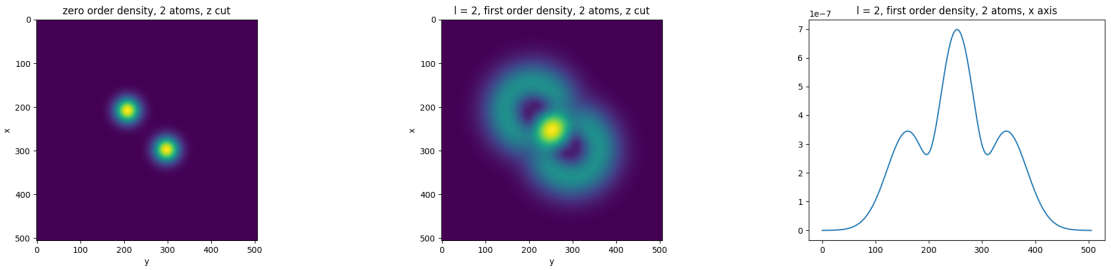
As we saw previously, if ρ is a single Gaussian, we have:

$$U_{l,j}[\rho](u,\theta,\phi) = \left(\sum_{m=-l}^{l} \left| \sum_k (\psi_{l,m,j})_{r_k} \left(\frac{r}{s_j}, \theta, \phi \right) \right|^2 \right)^{1/2}$$

The first order density of multiple Gaussians appears then to be an interference figure between solid harmonic wavelets. It could behave worse than the single Gaussian first-order density in terms of aliasing.

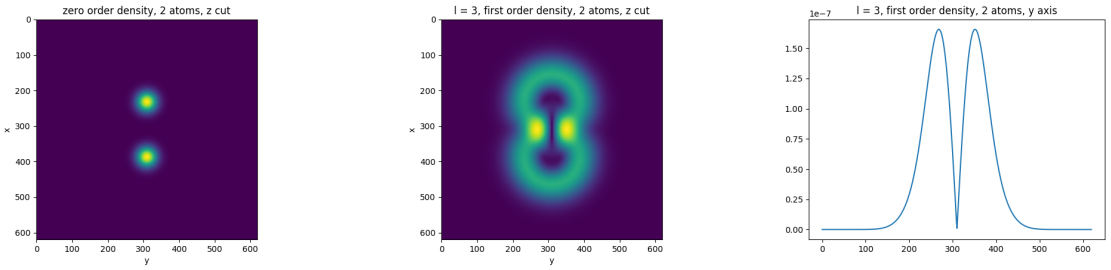
To see what can happen in practice, we can do numerical experiments with two Gaussians, for different values l :

- With $l = 2$, we observe the following interference pattern:



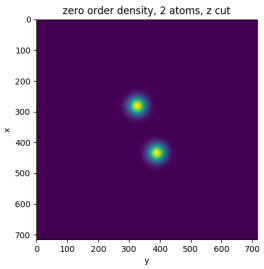
  The interference is constructive, and the function is strictly positive. The square root does not create derivative discontinuity.

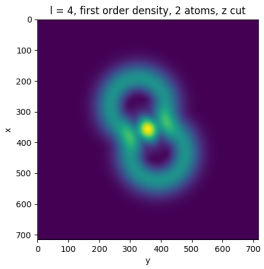
- With $l = 3$, we observe the following interference pattern:

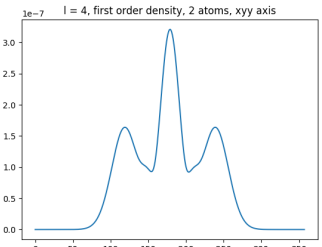


  The interference is destructive. The is zero in the middle of the atom positions. The square root creates derivative discontinuity. This causes slow decay of the Fourier modulus.

- With $l = 4$, we observe the following interference pattern:

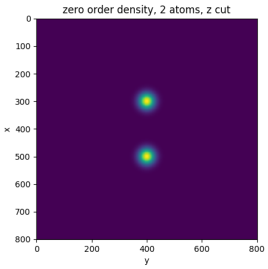


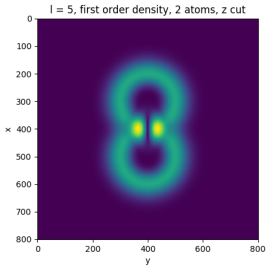


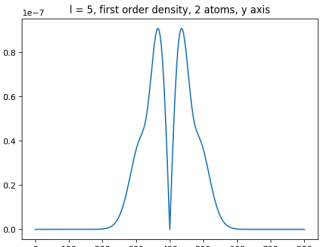


  The interference is constructive and does not create derivative discontinuity.

- With $l = 5$, we observe the following interference pattern

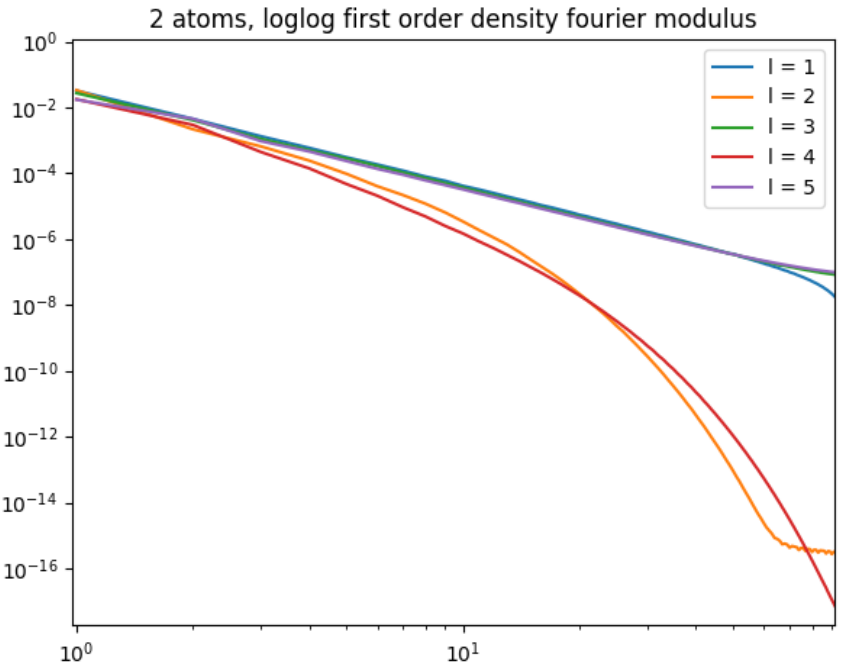






  Once again, we observe a derivative discontinuity in the middle of the two atoms. The Fourier modulus has hence a slow decay.

Plotting the radial decay of the Fourier modulus of these different cases on the same figure gives us an idea of which l behave better in terms of aliasing:



So for a density ρ with two Gaussians, the first-order density $U_{l,j}[\rho]$ is once again much more likely to be aliased for odd l than for even l. It suggests using operator $U_{l,j}[\rho]$ only with even l to use a small sampling grid.

However, with a sufficiently large sampling grid, these aliasing effects can be bounded. We see that the Fourier modulus magnitude is about 10^{-6} at the boundaries with approximately 200 points. One can afford a larger grid and have even lower errors.

Moreover, the nature of the interference patterns is different between even and odd indices l. This might result in different invariants that might be complementary to regress the molecule's energies. We might use odd indices l in general for energy regression if they allow regressing the energy accurately.

2.3 Numerical experiments on the QM9 database.

2.3.1 Solid Harmonic Scattering Transform parameters

For the numerical experiments on the QM9 database [Ramakrishnan et al., 2014], we use Scattering invariant coefficients of order zero, order one, and order two. For each molecule, we compute a density with three channels. These three channels use the number of core electrons, the number of valence electrons, and the total number of electrons of the atoms in the molecule. We use the values $j \in \{0, 1, 2, 3, 4\}$ for the scale parameters and $l \in \{1, 2, 3\}$ for the solid harmonics. We raise the densities to the powers $q \in \{1/2, 1, 2, 3, 4\}$ to create supplementary invariants.

For convenience, we adopt the following notations. We define $x = (r_1, z_1, \ldots, r_{N_a}, z_{N_a})$ the state of a system of atom, which contains the positions and nuclear charges of the atoms. From x we can compute the densities ρ_x and the scattering coefficients. We concatenate these coefficients in a Scattering vector that we denote by $S[\rho_x]$.

2.3.2 Multi-linear regression

We regress the molecules' energy with multi-linear combinations of scattering coefficients in $S[\rho_x]$. A multilinear regression of order r is defined by:

$$F(x) = b + \sum_i \left(\nu_i \prod_{k=1}^{r} (\langle S[\rho_x], w_i^{(k)} \rangle + c_i^{(k)}) \right).$$

For $r = 1$, it gives a linear regression

$$F(x) = b + \langle S[\rho_x], w^{(k)} \rangle$$

For $r = 2$, we have a bilinear regression. It includes products of Scattering coefficients. Hence, it allows to model interactions between different scales j and indices l. The whole regression pipeline is illustrated in figure 2.2.

We optimize the parameters of the multilinear regression by minimizing the quadratic loss over the N training molecules x_i

$$\sum_{i=1}^{N} (E(x_i) - F(x_i))^2,$$

using the Adam algorithm for stochastic gradient descent [Kingma and Ba, 2015].

If $F(x)$ is the total energy of a state x, and x is decomposable into several simpler subsystems, then perturbation theory analysis expands the energy $F(x)$ into an infinite series of higher-order energy terms:

$$F(x) = F_0(x) + F_1(x) + F_2(x) + \cdots,$$

in which F_0 captures the energies of the isolated subsystems, F_1 captures their electrostatic interactions, and F_2 captures induction and dispersion energies, which result from van der Waals interactions. The linear regression similarly expands the total energy into successively higher-order energy terms defined by wavelet scattering coefficients.

Multi-linear regressions were trained and evaluated on five random splits of the dataset, for $r = 1$, i.e., linear regression, and $r = 3$, tri-linear regression. The fit's evaluation criterion is the mean absolute error, which is the most prevalent error measure in the literature. We use 5-fold cross-validation with 107,108 molecules for training and 26,777 for test per fit.

Results are shown in the table 2.1 and compared with other techniques. Tri-linear regression achieves close to state-of-the-art error (0.56 kcal/mol). Notably, a simple linear regression error is much lower than that of Coulomb matrices fit with kernel ridge regression at full sample complexity. Furthermore, while kernel methods with appropriate kernels may decrease the

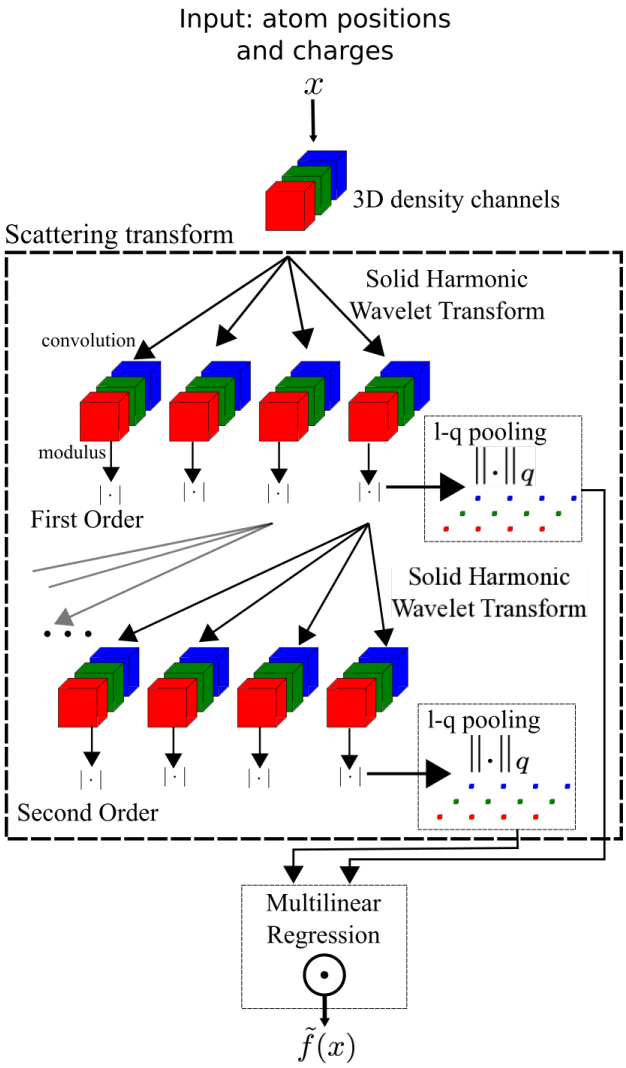


Figure 2.2: **Scattering regression pipeline.** The atomic positions and charges $x = (r_1, z_1 \ldots, r_{N_a}, z_{N_a})$ are mapped to three-dimensional densities. An invariant scattering transform is applied to each density. Invariant scattering coefficients are spatial l^q norms of the resulting zero, first, and second-order densities for several exponents q. A multilinear regression computes an estimation of the energy from these invariant scattering coefficients.

Table 2.1: Test Mean Absolute Error (MAE) on the QM9 molecules' atomization energies in kcal/mol. Scatt., CM, BoB, HDAD, and SOAP stand respectively for Solid Harmonic Scattering Transform, Coulomb Matrix, [Rupp et al., 2012], Bag of Bonds [Hansen et al., 2015], Histogram of distances, angles, and Dihedral angles [Faber et al., 2017] and Smooth Overlapping Atomic Positions [Bartók et al., 2013]. KRR stands for Kernel ridge regression. Results on QM9 are taken from the corresponding publications except for SOAP, where the results are taken from De et al. [2016].

descriptor regression	Scatt. Linear	Scatt. Tri-linear	CM KRR	BoB KRR	HDAD KRR	SOAP KRR
MAE (kcal/mol)	1.89	0.56	2.95	1.53	0.58	0.53

error to 0 given a large or infinite amount of data, their representation sizes increase with the number of samples. Using fixed-size and especially linear regressions, as we do, we can detect their capacity limit and use the relevant prediction depending on the case.

Interestingly, the SOAP kernel achieves a test MAE of 0.5 kcal/mol with a cutoff radius of $3\dot{A}$. The test MAE is worse when using larger cutoff radii of $4\dot{A}$ and $5\dot{A}$ [De et al., 2016]. This demonstrates that the energies of the QM9 database are very local. This database might not be relevant to describe large-scale patterns and scale interactions with a multi-scale descriptor like Scattering Transform. For this reason, we decide to construct a database with long-range energies in the following.

2.4 Graphite database with long-range energies

2.4.1 About graphite

Graphite is a crystal of carbon atoms. It occurs naturally in this form and is the most stable form of carbon under standard conditions. It is used, for example, in pencil leads. It converts to diamond under high pressures. Graphite is a stack of Graphene layers. In Graphene layers, carbon atoms are arranged in a hexagonal structure are (see Figure 2.3). Bonding between layers is via weak Van-der-Waals forces [Chung, 2002]. These forces are long-range forces, which makes the study of graphite interesting for us.

2.4.2 Generating configurations

The atomic systems we consider are 3D cubic periodic cells. They contain from 460 to 512 carbon atoms. These configurations were obtained with molecular dynamics simulations using a Gaussian Approximation potential [Bartók and Csányi, 2015]. Precisely, the method used was a Langevin dynamic at a temperature of $3,000K$, and there are 25 uncorrelated molecular dynamic runs. The volumetric mass density of these systems varies in 1.00, 1.25, 1.50, 1.75, or 2.00 g.cm^{-3}. There are five uncorrelated runs for each of these values, yielding 25 uncorrelated runs in total. Each trajectory is the concatenation of 50 snapshots spaced 2 picoseconds apart[2],

[2] 1 picosecond is 10^{-12} seconds

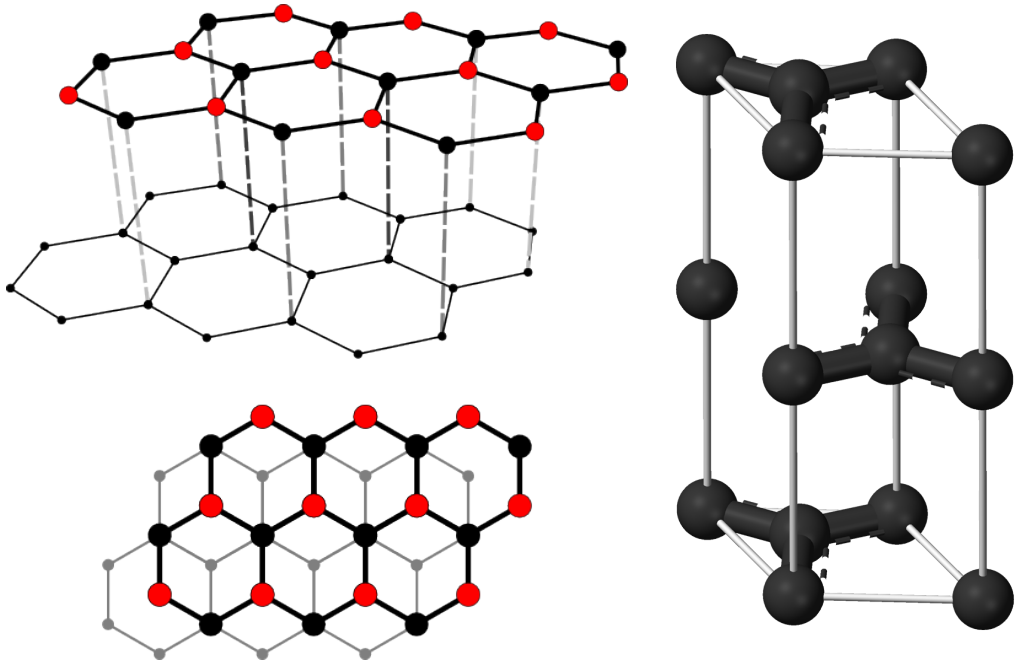


Figure 2.3: (Left) Graphite structure. All atoms are carbon atoms. Atoms represented with black dots are above carbon atoms. Atoms represented with red dots are above the center of hexagons. (right) A unit cell of the graphite crystal. The crystal is obtained by periodic replication of this cell in the directions of the edges.

followed by 50 snapshots spaced 4 ps apart. In total, each trajectory contains a total of 100 snapshots and covers a time-lapse of 300 ps. Volker Deringer from Cambridge University performed the generation of these trajectories.

2.4.3 Energy computations

To compute the energies, we have to use an electronic structure method that effectively models these Van der Waals forces. For this purpose, we use the Many-body dispersion [Tkatchenko et al., 2012, Ambrosetti et al., 2014] electronic structure method. This method uses the coupled Quantum Harmonic Oscillator (QHO) model Hamiltonian. It is coupled with semi-local Density Functional Theory (DFT) functionals by using the range-separated Density functional [Toulouse et al., 2004]. This level of theory can account for Van der Waals forces that can range to 15 Å.

Computations were performed by Martin Stoehr from Luxemburg University using the Many-Body Dispersion software of the Fritz-Haber Institute[3].

The computed energies range between −68.29 eV and −40.19 eV. The mean energy is −54.16 eV, and the standard deviation is 7.06 eV.

[3]http://www.fhi-berlin.mpg.de/ tkatchen/MBD/MBD.tar

2.4.4 Cross-validation folds

For a fair comparison between different methods, we predefine a fivefold cross-validation split:

- The first fold ranges between -67.65 and -45.28 eV with a mean of -54.90 eV and a standard deviation of 6.90 eV.
- The second fold ranges between -67.94 and -41.74 eV with a mean of -54.44 eV and a standard deviation of 6.74 eV.
- The third fold ranges between -68.23 and -40.19 eV with a mean of -53.88 eV and a standard deviation of 7.17 eV.
- The fourth fold ranges between -68.29 and -41.43 eV with a mean of -53.81 eV and a standard deviation of 7.10 eV.
- The fifth fold ranges between -67.84 and -42.51 eV with a mean of -53.75 eV and a standard deviation of 7.32 eV.

2.5 Spatial separation based on local SOAP descriptors

2.5.1 Local SOAP descriptor

The Smooth Overlapping of Atomic Position (SOAP) defines a similarity kernel between atomic environment. The atomic neighborhood $\mathcal{N}_i$ of atom i is represented by a fictitious density function ρ_i. Like in Solid Harmonic Scattering transform, this density is a sum of Gaussian functions g centered on the atomic positions:

$$\rho_i(r) = \sum_{j \in \mathcal{N}_i} g_\sigma(r - r_j)$$

The rotation invariant similarity kernel K takes as input the densities ρ_i and ρ_j representing the two atomic neighborhoods $\mathcal{N}_i$ and $\mathcal{N}_j$. The kernel K is defined as

$$K(\rho_i, \rho_j) = \int_{R \in SO_3} \left[\int_{r \in \mathbb{R}^3} \rho_i(r) R.\rho_j(r) \mathrm{d}r \right]^\zeta \mathrm{d}R$$

where $R.\rho_j$ is the density ρ_j rotated by $R \in SO_3$ and $\zeta \in \mathbb{N}$ is an integer exponent. This similarity measure is then normalised to have the normalized similarity measure K_n

$$K_n(\rho_i, \rho_j) = \frac{K(\rho_i, \rho_j)}{\sqrt{K(\rho_i, \rho_i) K(\rho_j, \rho_j)}}$$

For numerical computations, the densities ρ_i are expanded over radial and Spherical harmonics bases. One can find more details about the SOAP kernel in Bartók and Csányi [2015].

2.5.2 Spatial separation method

Based on this SOAP kernel, we propose a method based on spatial separation. In this setting, the total energy is a sum of three terms that correspond to different spatial components:

1. The first term accounts for long-range interactions. It takes the form of a pair potential. It ranges to 10 $\dot{A}$. We parametrize it using 30 basis functions were used equally spaced on $\left[0\dot{A}, 10\dot{A}\right]$. The bases functions are orthonormalized Gaussian functions length scale of $2\dot{A}$.

2. The second term is a local term. It is a SOAP kernel for each atom, with an exponent ζ=2. It is a quadratic SOAP kernel. This kernel compares atomic neighborhoods of a radius of $3\dot{A}$. The atomic densities are expanded over 10 radial basis functions and 10 angular Spherical harmonics. The Gaussian functions g representing the atoms have a width of $0.5\dot{A}$. To reduce the cost of the Kernel matrix's inversion, 1000 representative configurations out of 2000 available are chosen via CUR decomposition of descriptor matrix.

3. The third term is a local term with a larger range. It is a SOAP kernel for each atom, with an exponent ζ=2. This kernel compares atomic neighborhoods of a radius of $6\dot{A}$. The atomic densities are expanded over 10 radial basis functions and 10 angular Spherical harmonics. The Gaussian functions g representing the atoms have a width of $1\dot{A}$. Similarly, 1000 representative configurations out of 2000 available are chosen via CUR decomposition of descriptor matrix.

The first term is fitted alone, independently of the SOAP kernel. Then the remaining energy is fitted using the two short-range and middle-range SOAP kernels.

2.6 Regression results

2.6.1 Parameters of solid harmonic scattering representation

There is a single atomic species, carbon, in the systems we consider. Hence, we do not use the vector c_i from eq 2.1. We represent the system with a sum of Gaussian functions.

$$\rho(r) = \sum_i g_\sigma(r - r_i)$$

We consider a periodic system with periods d_x, d_y, d_z in the directions x, y, z. The density we construct has to be periodic in a box of dimensions are (d_x, d_y, d_z). We will sample this periodic signal on a discrete grid is of shape (N_x, N_y, N_z). The spatial discretization steps are thus $(\delta_x, \delta_y, \delta_z)$ are

$$(d_x/N_x, d_y/N_y, d_z/N_z).$$

To computed the periodic density, we simply have to compute the Fourier transform of the density

$$\widehat{\rho}(\boldsymbol{\omega}) = e^{-|\boldsymbol{\omega}|^2\sigma^2/2} \sum_n e^{-i\boldsymbol{\omega}\cdot\mathbf{r}_n}$$

on a Fourier domain corresponding to a grid of shape (N_x, N_y, N_z) and of dimensions (d_x, d_y, d_z). This Fourier domain P_f is the following

$$P_f = [-\pi N_x/d_x, \pi N_x/d_x] \times [-\pi N_y/d_y, \pi N_y/d_y] \times [-\pi N_z/d_z, \pi N_x/d_z] \,.$$

In practice, the ratios $d_x/N_x, d_y/N_y$ and d_z/N_z can not be equal in general because N_x, N_y, N_z are integers and d_x, d_y, d_z are arbitrary floats. To have a similar sampling effect in each of the x, y, z dimensions, we try to make them as equal as possible. Another constraint is to have the shapes N_x, N_y, N_z products of powers of 2, 3, and 5 to enable the fast Fourier transform factorization.

We compute a Solid Harmonic Scattering descriptor with coefficients of zero, first and second order. We use the following values for the parameters:

- Integral powers $q = 1, 2$
- Spherical harmonics orders $l = 0, 1, 2, 3$
- Scales $j = 0, 0.5, 1, 1.5, 2, 3, 4$

Using these values, the scattering vector representing a configuration is of dimension 226.

We use a linear ridge regression implemented in the Scikit-Learn package [Pedregosa et al., 2011]. We use 5-fold cross-validation for each value of the volumetric mass density. We train on four runs per volumetric mass density value, which makes 2000 train samples. We test on one run per volumetric mass density value, which makes 500 test samples.

The linear regression's ridge parameter is set with a log-scale grid-search ranging from 10^{-8} to 10^3. The retained value is 10^{-6}.

2.6.2 Results

We obtain a Mean Absolute Error (MAE) of 39.5 meV with the setting described below. The Root Mean Square Error (RMSE) is 49.8 meV.

For each value of the Volumetric Mass (VM) density in $\{1., 1.25, 1.5, 1.75, 2.\text{g/cm}^3\}$, we train a linear regression using a five fold cross validation. The five cross-validation folds are the five uncorrelated runs. We train on four folds and test on the remaining one. Results are summed up in the table below.

VM density (g/cm^3)	1.00	1.25	1.50	1.75	2.00
MAE (meV)	49.6	42.1	52.2	44.5	39.6
RMSE (meV)	64.0	54.0	66.0	57.4	50.3

Scale analysis

To analyze the importance of long-range interaction, we study the evolution of the error w.r.t. to the length of interactions. Solid harmonic scattering coefficients of scale j have a receptive field limited to 2^j. Using the fitted linear regression coefficients, we compute the error using only coefficients of scale smaller than j. So we compute the evolution of the error w.r.t. the length. We plot this quantity in figure 2.4.

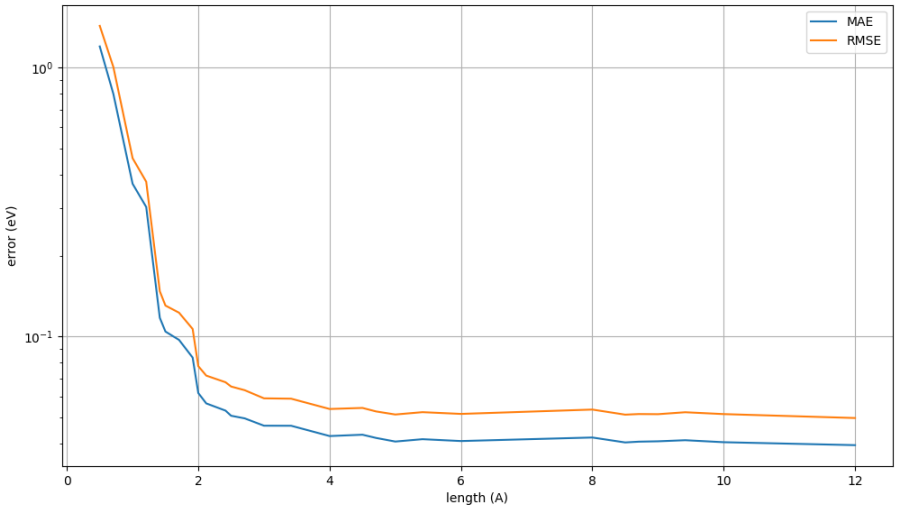


Figure 2.4: Evolution of the error w.r.t. the maximum length captured by scattering coefficients.

2.6.3 Results with two SOAP and a pairwise potential

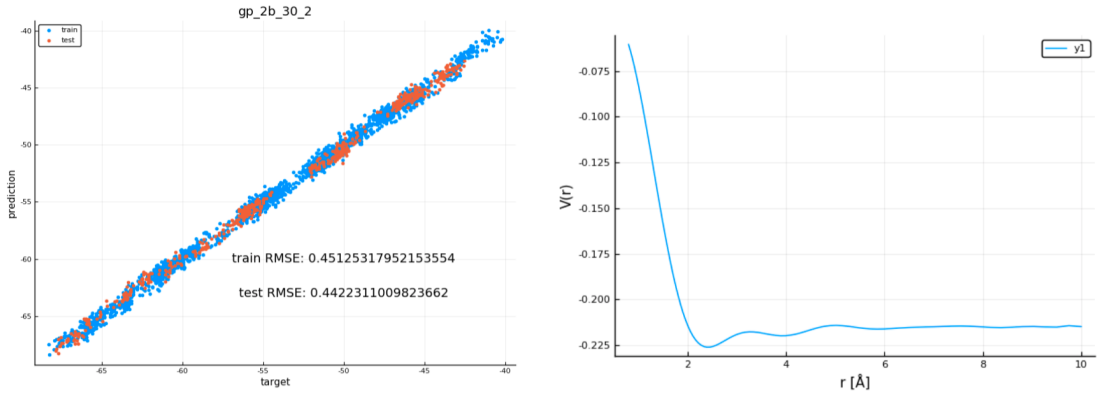


Figure 2.5: (left) Predicted energies using the pair potential w.r.t. true MBD-energies. (right) Learned pair potential w.r.t. the pairwise distance r.

The regression technique is a Bayesian kernel ridge regression or Gaussian process regression. The regularization parameter is chosen to be 5.10^{-5}.

The fitting of the pair potential yields a mean square error of 442 meV. The predictions of this pair potential and the fitted potential are shown in figure 2.5.

The two SOAP kernels that account for short-range and middle-range interactions are fitted on the difference between the pair potential predictions and the true MBD energies. It yields an RMSE of 52.8 meV. The predictions and the errors of the whole model based on this spatial

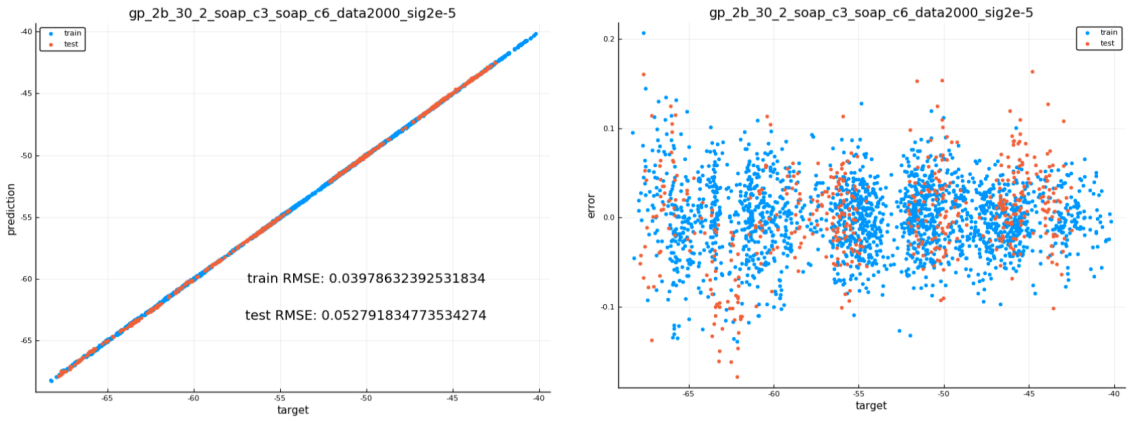


Figure 2.6: (left) Predicted energies and (right) errors using the whole model with 2 SOAP kernels and the pair potential.

separation are shown in figure 2.6.

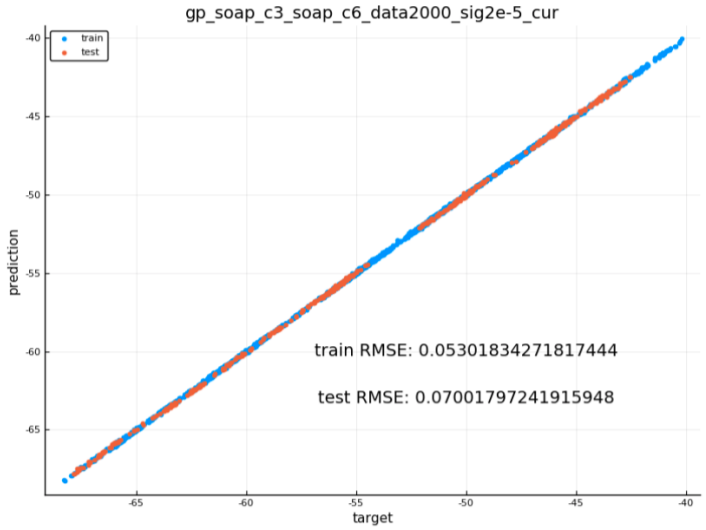


Figure 2.7: (left) Predicted energies a using 2 SOAP kernels without the pair potential.

When fitting the two SOAP kernels directly on the MBD energies, the RMSE is 70.0 meV. This increase of 17 meV is due to the limited range of this model to 6 $\dot{A}$. Predictions of this two-SOAP model are shown in figure 2.7.

2.6.4 Discussion

The Solid Harmonic Scattering transform with a linear regression gives a smaller error, with 49.8 meV compare to 52.2 meV of the model with two SOAP kernels and a pair potential. This difference is relatively small w.r.t. the magnitude of these errors. An error of $50meV$ corresponds to 1.153 kcal/mol, which is approximately the chemical accuracy. Having a lower

error, we might overfit the MBD energies while caring for the true physical energies. So we can consider that these two models perform similarly on the prediction of the MBD energy.

The model with two SOAP kernels and a pair potential based on spatial separation presents advantages in terms of interpretation. The long-range potential gives interesting insights into the nature of long-range interactions. Short-range and middle-range contributions to the energy are separated in the two SOAP kernels. One can analyze them separately.

The Solid Harmonic Scattering Transform does not offer these possibilities directly. It allows solely to analyze the evolution of the error w.r.t. length of interactions as shown in figure 2.4. Physicists would typically prefer a method based on spatial separation that offers more physical interpretability than a systematic multi-scale treatment.

We can notice that when limiting the length to 6 $\dot{A}$, the error is 65 meV, in the same range as the 70 meV error of two-SOAP kernels without pair potential. Not modeling the long-range energies induce an increase of 20 meV in the RMSE. Their importance is thus relatively small. This opens further comparison between multi-scale Solid Harmonic descriptor and descriptors based on spatial separation on new energies databases where long-range interactions count significantly in the total energy.

Chapter 3

A local method for vibrational entropy regression

In the previous chapter, we have seen that despite the multi-scale nature of the ground state energy of molecules or solids, descriptors based on scale separation like Solid Harmonic Scattering Transform do not have a superior predictive power to local ad-hoc descriptors.

However, the ground state energy is not the only thermodynamical property that drives the atoms' evolution in a physical system. At a constant temperature T, the free energy A is the relevant quantity to describe the system evolution. It indicates whether a physical process is likely or not in the system.

The free-energy A is defined as the energy $F(r_1, \dots, r_{N_a})$ minus the temperature T times the entropy $S(r_1, \dots, r_{N_a})$.

$$A(r_1, \dots, r_{N_a}) = F(r_1, \dots, r_{N_a}) - T \times S(r_1, \dots, r_{N_a})$$

The vibrational entropy represents the entropy $S(r_1, \dots, r_{N_a})$ under the harmonic approximation. It is defined only for atomic positions $(r_1^m, \dots, r_N^m)$ at a local minimum of the energy. Computing the vibrational entropy involves the diagonalization of the system's Hessian. In a N_a atoms system, the system's Hessian is a $3N_a \times 3N_a$ symmetric matrix. This matrix' spectrum computational cost scales like $\mathcal{O}(N_a^3)$. It is prohibitive for systems with typically $N_a > 10^4$. Avoiding this computational burden using a regression model is the first motivation for the presented work.

Scattering Transform has shown promising results [Bruna and Mallat, 2019, Zhang and Mallat, 2019] in the context of maximum-entropy stochastic process modeling and sampling. The multi-scale invariant coefficients computed with a cascade of wavelet transforms, non-linearity, and averaging offer appropriate statistics to characterize non-Gaussian processes like turbulence. Moreover, the multi-scale separation of the Scattering Transforms is a key aspect of these statistics. It allows correlating non-linearly phenomena and structures occurring at different scales. The vibrational entropy presented here is a different quantity, but there are connections between these definitions. Is the scale separation appropriate in our context of vibrational entropy regression? How does the Solid Harmonic Scattering Transform descriptor compare with descriptors relying on spatial separation?

We answer these questions in the present chapter. We construct a database of body-centered cubic (BCC) crystals of iron atoms with defects. These configurations are all relaxed to a local minimum of the energy landscape. We compute the spectrum of the Hessian of the system and the exact vibrational entropies of these systems. We then test and compare two regression methods based on two descriptors relying on two types of separation :

1. The first one is a Solid Harmonic wavelet Scattering Transform that we studied in the previous chapter in the context of energy regression.

2. The second one is the Angular Fourier Series (AFS) local descriptor introduced by Bartók et al. [2013]. The AFS descriptor separates the angular and radial information in different channels.

Using linear regression, the AFS descriptor shows a superior predictive power on various large systems of body-centered cubic structures of iron atoms with defects. Moreover, the vibrational entropy's short-range separation property allows our method to extrapolate on very large atom systems. With the AFS descriptor, Vibrational entropies in a range of 250 k_B are predicted with less than 1.6 k_B using training samples in a range of 25 k_B.

This chapter is organized as follows. After having presented the physical background of our problem, we present entropy computations under the harmonic approximation. Then we detail the generation of the configurations' database and the vibrational entropy computations. We present two regression techniques based on two types of separation. Finally, we present and discuss the results.

This work has been published in Lapointe, Swinburne, Thiry, Mallat, Proville, Becquart, and Marinica [2020]. My personal contributions concerned the regression experiments with the Solid Harmonic Scattering Transform and the python open-source implementation of the AFS descriptor [1].

3.1 Physical background

Defects in iron crystals can have extraordinarily diverse morphologies [Marinica et al., 2011, Alexander et al., 2016, Marinica et al., 2012]. The distribution of these defects across the iron crystal exhibits significant variation with temperature.

The complex properties of defect populations are reflected in the underlying defect-free energy landscape. Accurate free energies are essential to model the formation of these defects. At finite temperature T, the energy F must be supplemented with the calculation of vibrational entropy S to give the free energy A

$$A(r_1, \dots, r_{N_a}) = F(r_1, \dots, r_{N_a}) - T \times S(r_1, \dots, r_{N_a})$$

There are now many well-established methods in computational material science to calculate the total energy to varying degrees of accuracy. Due to continuous increases in computational power and parallel software development, sophisticated electronic structure methods

[1] https://github.com/louity/AFS-python

can routinely access thousands of atoms [Dezerald et al., 2016, Domain and Becquart, 2018, Alexander et al., 2016, Olsson et al., 2010]. However, due to the at best $\mathcal{O}(N_a^3)$ scaling of computational effort a single vibrational entropy calculation, exploring free-energy landscape is practically infeasible.

Here, we propose a method to regress the vibrational entropy directly from the relaxed atomic positions. We use a descriptor combined with linear regression and the computational cost scales like $\mathcal{O}(N)$. The method is applied to a wide range of point defects in empirical atomistic body-centered cubic (BCC) iron models. This conceptually simple method we propose exhibits an exceptional degree of transferability, giving the ability to rapidly assess free-energy landscapes at realistic temperatures.

We note that an $\mathcal{O}(N)$ approach has been developed by Huang et al. [2013]. They approximate the probability distribution of the Hessian eigenvalues. They use distribution using a set of Chebyshev polynomials with a random basis set for this approximation. Whilst this stochastic approach is indeed superior to the above $\mathcal{O}(N_a^3)$ treatment for large systems, a converged result requires selecting a suitably large set of polynomials and basis vectors, requiring a substantial computational effort. It is still impractically high for the high-throughput evaluation presented here, motivating the proposed regression method.

3.2 Vibrational entropy in the harmonic approximation

In the harmonic approximation, the entropy is defined only for atomic positions $(r_1^m, \ldots, r_N^m)$ at a local minimum of the energy. This entropy is called vibrational entropy. Since $(r_1^m, \ldots, r_N^m)$ is a minimum, i.e. the gradient ∇F of F is zero and the Hessian $\nabla^2 F$ of F is positive definite

$$\nabla F(r_1^m, \ldots, r_N^m) = 0$$
$$\nabla^2 F(r_1^m, \ldots, r_N^m) \succcurlyeq 0$$

The Hessian $\nabla^2 F(r_1^m, \ldots, r_N^m)$ has $3N$ positive eigenvalues $\omega_{\nu=1\ldots3N}$. The vibrational entropy is defined as follows

$$S(T) = k_B \sum_{\nu=1}^{3N} \left[\ln\left(\frac{k_B T}{\hbar \omega_\nu}\right) + 1 \right] \tag{3.1}$$

where k_B and $\hbar$ are the Boltzmann and Planck constants, respectively. This approximation is valid in the limit of high temperatures such as

$$\forall \nu, \quad \frac{\hbar \omega_\nu}{k_B T} \ll 1$$

For finite crystalline systems containing N_b bulk atoms and $\pm N_d$ point defects, the vibrational formation entropy S_f is defined as

$$S_f(T, N_d) = S_d(T, N_b \pm N_d) - \frac{N_b \pm N_d}{N_b} S_b(T, N_b), \tag{3.2}$$

where the entropies S_b and S_d of the bulk and defective systems are computed at the same volume V. With Hessian eigenvalues $\omega_{\nu_b}^2$ and $\omega_{\nu_d}^2$ for the bulk and defect systems, equation

(3.2) yields a harmonic formation entropy of

$$S_f(T, N_d) = k_B \ln \left(\frac{\prod_{\nu_b} (\hbar\omega_{\nu_b})^{\frac{N_b \pm N_d}{N_b}}}{\prod_{\nu_d} \hbar\omega_{\nu_d}} \right). \tag{3.3}$$

Using Green function formalism, one can show [P. H. Dederichs, 1980, Lapointe et al., 2020] that the total entropy is the exact sum of local entropy contributions:

$$S = \sum_{i=1}^{N} S_i$$

where S_i represents the local entropy contribution of the neighborhood of the atom i. From our perspective, the regression of the vibrational entropy is a separable problem with a local separation. Hence a local method is reasonable to regress the vibrational entropy.

3.3 Configuration database

3.3.1 Generating configurations

Any regression model heavily relies on the database used for training. We used the `ART` method [Barkema and Mousseau, 1996, Malek and Mousseau, 2000, Cancès et al., 2009, Machado-Charry et al., 2011], following the methodology of previous studies by Marinica et al. [2011], to generate a large number of configurations for small vacancy and interstitial clusters in bcc Fe. All clusters contained between 1-4 removed or additional atoms, which we label as V_n and I_n respectively, with $n = 1, 2, 3, 4$. Despite their apparent simplicity, such defect configurations' energy landscape is known to have many thousands of binding configurations [Marinica et al., 2011, 2012, Swinburne and Perez, 2018]. To test the sensitivity of our regression model to the underlying energy model, all calculations were performed in duplicate using two interatomic potentials for bcc Fe, the embedded atom model (EAM) potential of Ackland et al. [2004] (AM04), and the modified embedded atom model (MEAM) potential from Etesami and Asadi [2018].

After an initial period of structure generation, all configurations were pairwise compared to ensure the final database only contained non-equivalent structures. Two configurations are considered as non-equivalent if their energies differ by more than 10^{-2} eV and if the sum of squares of the principal components of inertia tensor of the interstitial atoms are different. Interstitial atoms are localized using the Wigner-Seitz method.

The resulting database is an order of magnitude larger than that obtained by Marinica et al. [2011]. This is partly due to a more aggressive `ART`n parametrization to promote escape from and discovery of deep super basins. In particular, relaxing an earlier restriction that rejected saddle points of more than 2eV above the low energy 'parallel dumbell' configuration [Marinica et al., 2011] allows the discovery of very low energy $C15$ type I_4 clusters [Marinica et al., 2012] that were previously missed. The resulting database is summarised in Table 3.1.

System (N_a, ϵ)	Type of point defects (N_{cf})				Total
	I_2	I_3	I_4	V_4	
1024, $\epsilon = +0\%$	434	1105	1280	1701	4520
1024, $\epsilon = -1\%$	434	1105	1280	1701	4520
1024, $\epsilon = +1\%$	434	1105	1280	1701	4520
1024, $\epsilon = +2\%$	434	1105	1280	1701	4520
1024, $\epsilon = +3\%$	434	1105	1280	1701	4520
2000, $\epsilon = +0\%$	434	1105	1280	1701	4520
3456, $\epsilon = +0\%$	434	1105	1280	1701	4520
Total	3038	7735	8960	11907	31640

Table 3.1: Database used to train the present regression model. N_a is the number of atoms in the perfect system, N_{cf} the number of distinct instances for a point defect class. I_{2-4} and V_4 denotes the interstitial clusters with 2 up to 4 SIAs and the quadri-vacancy, respectively. The size of these systems with defects are $N_a + (2 \dots 4)$ and $N_a - 4$ for I_{2-4} and V_4, respectively. ϵ is an isotropic and homogeneous rate of deformation for the system

3.3.2 Computing vibrational entropies

To compute the harmonic entropy for each configuration, the Hessian was computed from $3N_a$ force evaluations using the standard finite difference formula with a displacement of 10^{-3} Å. Each configuration was tested to be a minimum by counting the number of eigenfrequencies. The relaxation of each configuration was performed using `LAMMPS` [Plimpton, 1995a,b]. The eigenvalues for the vibrational entropies were computed using the `PHONDY` package [Marinica and *et al*, 2007-2019, Marinica and Willaime, 2007, Soulié et al., 2018, Berthier et al., 2019].

3.4 Regression of the vibrational entropy.

3.4.1 Permutation invariance and short-range separation

In our system, we compute a descriptor $\phi(\mathcal{N}_i)$ for each atomic neighborhood $\mathcal{N}_i$. Since the system's energy is invariant w.r.t the atoms' permutation, the descriptor representing our atoms' system has to be invariant w.r.t. to permutation of the indices of our atoms. This is obtained by simply summing the descriptors

$$\Phi(r_1, \dots, r_N) = \sum_i \phi(\mathcal{N}_i)$$

Since we use linear regression, the entropy is simply

$$A = \langle w, \Phi(r_1, \dots, r_N) \rangle = \sum_i \langle w, \phi(\mathcal{N}_i) \rangle$$

On the one hand, this summation yields the extensivity property of the entropy. A system with twice the number of atoms has a twice bigger entropy. On the other hand, given the fact

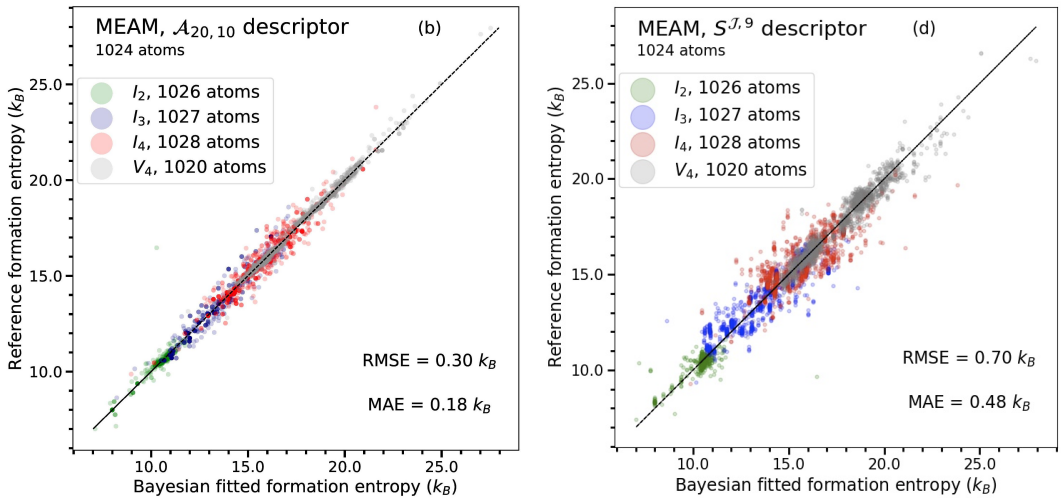


Figure 3.1: Training of different models using on a MEAM dataset of 2-4 interstitial clusters I_{2-4} and quadvacancies V_4, as detailed in the first line of Table 3.1. The descriptors used were (a-b) $\mathcal{A}_{20,10}$, and $S^{\mathcal{J},L}$ with scales $\mathcal{J} = \{0, 0.25, 0.5, 0.75, 1, 2, 3, 4, 5\}$. As is indicated in the root mean square error (RMSE) and mean average error (MAE), the $\mathcal{A}_{20,10}$ model has superior predictability.

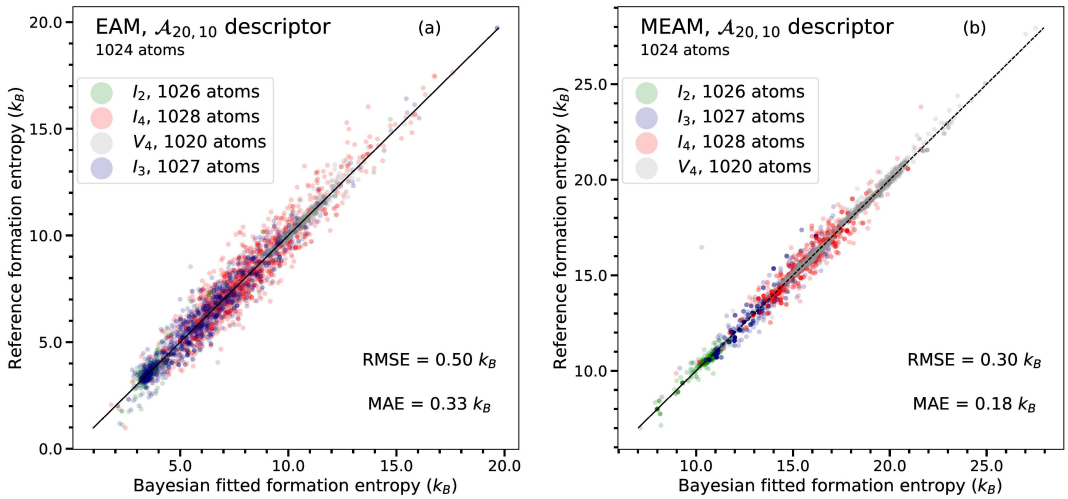


Figure 3.2: Training of models using on (left) an EAM and (right) database dataset of 2-4 interstitial clusters I_{2-4} and quadvacancies V_4, as detailed in the first line of Table 3.1. The descriptors used were AFS descriptors $\mathcal{A}_{20,10}$. Metrics used are the root mean square error (RMSE) and the mean average error (MAE). The analyticity of the MEAM potential gives smoother curvatures, which the linear model can better predict.

that the entropy is a sum of local contributions

$$S = \sum_i S_i$$

we can identify the local entropies S_i with the contribution of the atomic neighborhood $\mathcal{N}_i$ in the linear regression

$$S_i = \langle w, \phi(\mathcal{N}_i) \rangle \tag{3.4}$$

Compared to more sophisticated kernel methods and neural networks [Rasmussen, 2004, Behler, 2011], the linear approach followed here offers many advantages in transferability, overfitting control, and analytic connection to thermodynamic properties.

3.4.2 Local AFS descriptor

The Angular Fourier Series (AFS) descriptor $\mathcal{A}_{n,l}$ combines the radial and angular information of the local atomic environment. The n and l components account for the radial and angular information of the neighborhood $\mathcal{N}_i$ of the atom i. Here $\mathcal{N}_i$ is the ball of center r_i and radius r_{cut}. The AFS descriptor is defined as follows

$$\mathcal{A}^i_{n,l} = \sum_{j,k \in \mathcal{N}_i} g_n(r_{ij}) g_n(r_{ik}) \cos\left(l\theta_{ij,ik}\right) f_i(r_{ij}) f_i(r_{ik}),$$

where r_{ij} is the distance between the atom i and the atom j and $\theta_{ij,ik}$ the angle formed by the triplet of atom i, j, k centred on i. The sum involves the pairs and the triplets of atoms formed by the central i^{th} atom and the neighbouring atoms inside the sphere with the radius r_{cut} around the atom i by definition of $\mathcal{N}_i$. f is a cut-off function and $\forall r \geq r_{cut}, f_i(r) = 0$. The radial functions g_n are obtained from the ortho-normalization of the polynomials $p_n(x) = x^{n+2}, n = 1, \ldots, n_{max}$. The angular functions are the Tchebyshev polynomials with $0 \leq l \leq l_{max}$.

As $\mathcal{A}_{n,l}$ is formed from a product of the radial and angular channels, the descriptor has a total of $n_{max}(l_{max} + 1)$ components. The AFS descriptor enables a wide-ranging level of accuracy on the radial and the angular information by imposing n_{max} and l_{max}.

We used $n_{max} = 20$, and $l_{max} = 10$ and the cut-off distance of 5 in all the experiments performed with AFS. Hence, the total number of components for the AFS descriptor used here is 220. AFS descriptors of the retained configurations were computed using AFS-Python code[2].

3.4.3 Solid harmonic scattering descriptor

The solid harmonic wavelet scattering transform has been presented and studied in the previous chapters. There is a single atomic species, iron, in our systems. Hence, we don't use a vector c_i describing the properties of the atomic species. We use scattering coefficients of order zero and one. We use a single integral power $q = 2$, $L = 9$ angular indices and 9 scales $\mathcal{J} = \{0, 0.25, 0.5, 0.75, 1, 2, 3, 4, 5\}$, yielding a descriptor of size 90.

[2] https://github.com/louity/AFS-python

Solid Harmonic Scattering descriptors of the retained configurations were computed using the `PyScatHarm` package [3].

3.4.4 Linear regression

We use Bayesian linear regressions with a Gaussian prior on the likelihood. In the numerical experiments, we use the Scikit-Learn package [Pedregosa et al., 2011]. The initial value of σ for the Gaussian prior has been set using the default value. We also tested standard ridge regression, which gives the same results.

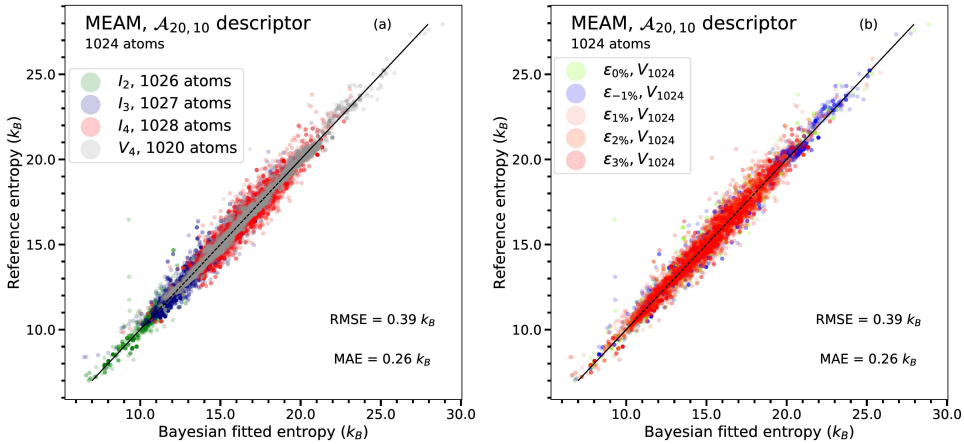


Figure 3.3: Illustration of the performance of the training of the linear model using deformed supercells of I_{2-4}/V_4 clusters (MEAM database) and by using $\mathcal{A}_{20,10}$ descriptor. The initial configurations have a $(8a_0)^3$ volume and have been deformed by applying a homogeneous and isotropic dilatation of the supercell. The deformation rates are from -1% to 3%. The figure illustrates the results of the regression model depending on the type of defect in the supercell (a) or depending on the deformation rate (b).

3.5 Results

3.5.1 Influence of interatomic potential and descriptor set

The linear model was tested on BCC defect systems as described above, initially in a supercell of size $8a_0 \times 8a_0 \times 8a_0$. The bulk lattice contained 1024 atoms before introducing 2-4 interstitial atoms to produce I_{2-4} defects or removal of 4 atoms for the V_4 quadvacancies. No supercell relaxation from the equilibrium bulk was performed.

We first tested the influence of the underlying interatomic potential models by training and testing the linear model on either the EAM or MEAM datasets. The MEAM potential

[3] https://github.com/louity/PyScatHarm

augments the EAM potential form with angular three-body terms. It employs analytic expressions for the pair and embedding functions [Daw and Baskes, 1984, Daw et al., 1993, Baskes, 1992] as opposed to the tabulated cubic splines of EAM potentials.

We compared two sets of descriptors for the linear model, the $\mathcal{A}_{20,10}$ AFS descriptor with $r_{cut} = 5.0$, and the Scattering Transform descriptor $S^{\mathcal{J},L}$. In our tests the $\mathcal{A}_{20,10}$ was around 50% faster to evaluate than $S^{\mathcal{J},9}$.

For both choices of descriptors, the MEAM results have lower RMSE and MAE, a feature we found replicated across other training sets. As the linear model regresses curvature-based entropies to descriptors of atomic structure, the poorer predictive power on the EAM dataset is almost certainly due to curvature irregularities induced by the nonanalytic tabulations used in the EAM formalism. As a result, we use the MEAM potential exclusively [Etesami and Asadi, 2018] in the following.

The results of regressions to the MEAM data with different descriptor functions are shown in Figure 3.1. On formation entropies ranging between $8k_B$ and 28 k_B, the performance in descriptor sets has limited variation, but we find $\mathcal{A}_{20,10}$ consistently outperforms $S^{\mathcal{J},9}$ despite the greater computational efficiency, with an RMSE of resp. $0.8k_B$ and $0.7k_B$ to $0.3k_B$

3.5.2 Modeling datasets with multiple defect species and variable supercell volume

Crystal defects are subject to long-range elastic fields due to external loading conditions and interactions with the wider defect distribution. It is therefore highly desirable to have predictability on the changes in formation entropy under deformations of the simulation supercell, as this can be used as a proxy for changes in the formation entropy under varying microstructural environments.

Since our linear model receives an input vector of fixed dimension independent of system size, it is possible to simultaneously train the model on datasets with a variable number of atoms.

As a first application, we trained the linear model on a large dataset of all I_{2-4} and V_4 configurations found in the `ARTn` searches with the same $8a_0$ cubic supercell as above, where each configuration was additionally copied, subjected to an isotropic dilation of -1% to 3% before a calculation of a new descriptor vector and harmonic entropy, giving a fivefold multiplication in the dataset size. Figure 3.3 illustrates the remarkable accuracy of the linear model using a single weight vector $\underline{w}$ for the entire dataset, with an RSME error of only $0.4k_B$.

3.5.3 Training on combined disordered and crystalline datasets

To test the ability of our linear model with descriptor functions to predict formation entropies beyond largely crystalline structures, we created an additional database of highly disordered structures from an `ARTn` database of I_{2-4} and V_4 configurations in cubic supercells of dimension $8a_0$, $10a_0$ and $12a_0$, as described in Tab. (3.1). Many individual atoms were subject to random displacements for each configuration, creating many Frenkel pairs before relaxation to a highly defective structure containing up to 22 vacancies and 26 interstitials. The set of such structures

will be referred to as the *random* database. The distribution of defects in the *random* database is presented in Figure 3.4; the difference between the number of interstitials and vacancies is conserved before and after the disordering procedure, giving a strong correlation between the effective vacancy and interstitial count. Figure 3.5 presents the results of a linear model trained on this highly diverse dataset. We find that the RSME error is only doubled to $0.8k_B$, which is to be compared to a formation entropy range of approximately $250k_B$. This high value of the formation entropy is attributable to the much higher effective number of defects in the system. As shown in the inset figure, this impressive performance is maintained even with a highly aggressive train/test splitting of 90%.

The presence of vacancies induces local softening of the vibrational modes, whilst interstitial defects both harden and soften. In particular, $\langle 111 \rangle$ interstitials exhibit an extremely soft mode due to an almost free translation of the dumbbell along the $\langle 111 \rangle$ direction [Lucas and Schäublin, 2009, Marinica and Willaime, 2007, Chiesa et al., 2009]. The same phonon mode is highly active in the $\alpha - \gamma$ martensitic transition of Fe and the pair kinks nucleation in the $1/2\langle 111 \rangle$ dislocation [Proville et al., 2012]. The present linear model's ability to mimic the physics of those soft modes is nontrivial, as the characteristic wavelength is far beyond the cutoff radius of the descriptors used to sample the local atomic environment. Despite this, the linear regression in the descriptor space can reconstruct the correlation of high formation entropies to large phonons wavelengths by predicting the right values of entropies at high values.

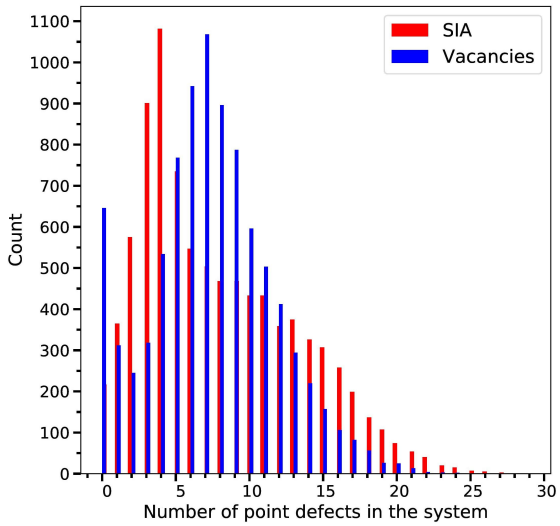


Figure 3.4: The analysis of the distribution of randomly generated point defects in the *random database*. This database is derived from the *ARTn database* only using the supercells of volume $(10a_0)^3$ and $(12a_0)^3$ by random creation of Frenkel pairs. The plot emphasizes the occurrences in the entire *random database* of number of SIAs and vacancies in the same supercell. The *random database* contains 9016 configurations.

3.5.4 Transferability of the crystalline model to disordered structures

In this final example, we artificially tested the transferability of the linear model by training *only* on the ARTn database of defect structures with variable supercell sizes before attempting to predict the formation entropies of the *random* database. As illustrated in Figure 3.5b), despite training on a dataset in an essentially disjoint region of the energy landscape, with a training formation entropy range of less than $25k_B$, the model achieved a remarkable predictive accuracy with an RSME error of only $1.53k_B$.

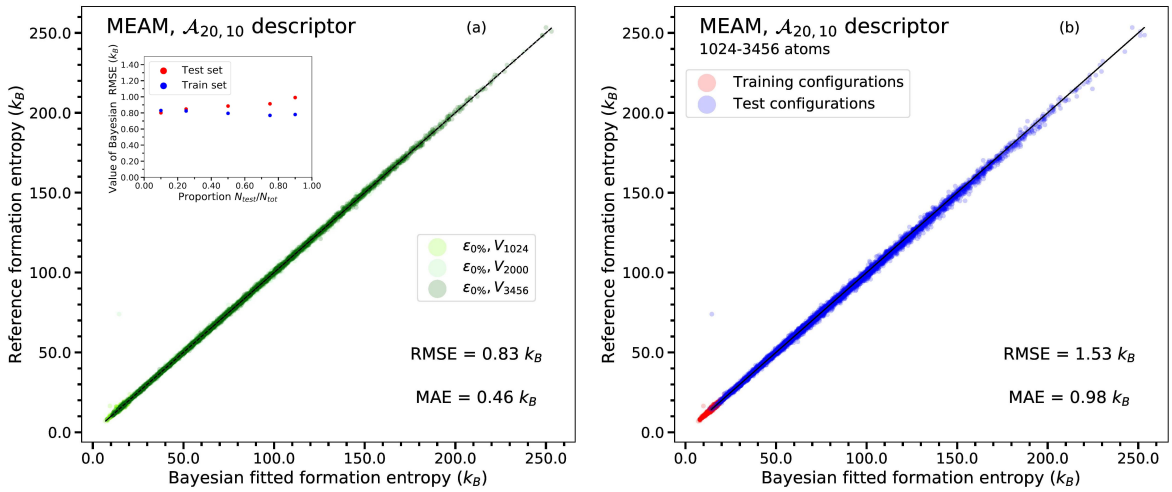


Figure 3.5: The robustness, and the transferability of our linear model is tested by (a) crossing validation using several splitting proportions between train and tested configurations of the joined *ARTn database* and the *random database* (the entropies are computed using MEAM potential [Etesami and Asadi, 2018] and the linear model employs $\mathcal{A}_{20,10}$ descriptor). The error metrics we use are the RMSE And MAE. The RMSE remains stable even for a low learning proportion.
(b) The model's predictive power trained on the *ARTn database* and validated on the *random database*. The statistical indicators, RMSE and MAE, are computed for the *random database*. The order of magnitude of the statistical indicator is the same as for (a), while the model is in an extrapolation regime.

3.6 Discussion

The presented regression method offers several advantages. It allows computing the defects' vibrational entropy in crystalline solids directly from the Cartesian coordinates. The atomic environment descriptor functions are calculated for each atom in a relaxed configuration, then summed across all N atoms, giving a model input space of dimension d independent of the system size N. The extensivity claimed by classical thermodynamics results from the use of

a linear model on top of this summed descriptor. These physical foundations of the presented model ensure robustness and remarkable extrapolation capabilities.

It demonstrates a good transferability from supercells containing only one defect cluster to complex configurations having more defects and clusters. The encouraging low error in predictions opens many perspectives. For example, the defects can be trained separately in small cells, whilst, complicated structures as those in the radiation damage can be accurately predicted.

The local Angular Fourier Series shows a superior predictivity among the descriptors used. The Solid Harmonic Scattering descriptor based on scale separation achieves a twice bigger error. This example demonstrates the efficiency of local methods for free energy regression. Wavelet Scattering coefficients offer appropriate statistics for maximum entropy processes modeling, but not well suited for the direct regression of Vibrational entropy of iron crystal.

Chapter 4

Structured patch-based convolutional neural network for image classification

The different studies we have performed in the field energy regression showed that local methods perform well on various regression tasks. Descriptors of molecules' and solids' small neighborhoods are very efficient in regressing energies and vibrational entropies. Although we know that energy results of multi-scale interactions, high-performing energy regression does not necessarily require a descriptor relying on scale separation.

Concomitantly to our work in energy regression, Brendel and Bethge [2019] have demonstrated that a competitive accuracy (87.5% top5) can be obtained using a CNN encoding small image patches. Local methods seem to perform very well in energy regression and image classification. We would like to understand better the reasons for this success.

The CNN introduced by [Brendel and Bethge, 2019] is based on a 50-layer ResNet architecture, and hence hard to analyze in terms of learning and classification mechanisms. Can we build a structured convolutional neural network architecture that encodes small image patches and reaches a competitive accuracy?

Following the research line of Mallat [2012], Bruna and Mallat [2013], Mallat [2016], Oyallon et al. [2019], we use the Scattering Transform to build a structured convolutional neural network. Scattering transform is a simplified convolutional neural network with wavelet filters that are not learned [Bruna and Mallat, 2013]. It provides state-of-the-art classification results among predefined representations. It is nearly as efficient as learned deep networks on relatively simple image datasets, such as digits in MNIST, textures, [Bruna and Mallat, 2013] or small CIFAR images [Oyallon and Mallat, 2014]. However, over complex datasets such as ImageNet, a learned deep convolutional network's classification accuracy is much higher than a Scattering Transform or any other predefined representation. Oyallon et al. [2017] proposed a structured deep convolutional neural network based on a Scattering transform, a cascade of 1×1 convolutions, and a huge MLP classifier. It reaches an accuracy of 79.6% on the ImageNet dataset. It is more amenable to analysis of the learning mechanism than usual deep

convolutional networks [LeCun et al., 2015].

In the presented architecture, overlapping image patches of size $\sim 16 \times 16$ are encoded with a Scattering Transform descriptor[1] which linearizes variabilities due to geometric transformations such as translations and small deformations. We concatenate Scattering encoding into a descriptor representing image patches. This descriptor is then encoded with a cascade of 1×1 convolutions and ReLU non-linearities. A global average pooling is performed at the end, followed by a linear classification. This structured architecture relies explicitly on patch-separation. We obtain a top-5 classification accuracy of 84.5% and 78.8% with patch sizes 32×32 and 16×16 respectively. These results are comparable with the 50-layers BagNet [Brendel and Bethge, 2019] with a similar patch-size despite a shallower encoding. These results are higher than the previous structured networks using Scattering Transform [Oyallon et al., 2017], although we have introduced patch separation in the architecture.

This chapter is organized as follows. First, we review the Scattering Transform. We detail the proposed architecture. We describe and analyze the numerical experiments we performed and compared with other methods on ImageNet. Finally, we discuss the results and the next research directions.

This work is the result of personal investigations and has not been published. PyTorch code to reproduce the classification experiments is available[2]. This work served as preliminary work for the publication by Zarka, Thiry, Angles, and Mallat [2019].

[1] The use of the word descriptor is rather unconventional for the Scattering transform. We use it here to insist on the analogy between atomic neighborhood descriptors in physics and image patches descriptors in visual representations.

[2] https://github.com/louity/scattering_patch_cnn

4.1 Scattering Transform descriptor of patches

A scattering transform is a cascade of wavelet transforms and modulus non-linearities. It can be interpreted as a deep convolutional network with predefined wavelet filters [Mallat, 2016]. For images, wavelet filters are calculated from a mother complex wavelet ψ whose average is zero. It is rotated by $r_{-\theta}$ and dilated by 2^j:

$$\psi_{j,\theta}(u) = 2^{-2j}\psi(2^{-j}r_{-\theta}u)$$

We choose a Morlet wavelet as in Bruna and Mallat [2013] to produce a sparse set of non-negligible wavelet coefficients.

Scattering coefficients of order $m = 1$ are computed by averaging rectified wavelet coefficients with a subsampling stride of 2^J:

$$Sx(u, j, \theta) = |x \star \psi_{j,\theta,\alpha}| \star \phi_J(2^{J-o}u)$$

where $o \in \mathbb{N}$ is an oversampling factor, and ϕ_J is a Gaussian dilated by 2^J [Bruna and Mallat, 2013]. Note that for appropriate wavelets, Scattering coefficients of order $m = 1$ are equivalent to SIFT coefficients [Lowe, 2004]. The DAISY approximation [Tola et al., 2009] shows that one can approximate SIFT coefficient with $|x \star \psi_{\lambda_1}| \star \Phi_{2^J}$, where ψ_{λ_1} are partial derivatives of a Gaussian computed at the finest image scale along eight different orientations, and Φ_{2^J} is a Gaussian filter scaled by 2^J.

The averaging by ϕ_J eliminates the variations of $|x \star \psi_{j,\theta}|$ at scales smaller than 2^J. This information is recovered by computing their variations at all scales $2^{j'} < 2^J$, with a second wavelet transform. Scattering coefficients of order two are:

$$Sx(u, j, \theta, j', \theta') = ||x \star \psi_{j,\theta}| \star \psi_{j',\theta'}| \star \phi_J(2^{J-o}u) \text{ for } j' > j$$

One can observe that at a spatial position u, a scattering coefficient $Sx(u)$ corresponds to a descriptor of a patch of the initial image. This patch is a square domain whose bottom left corner is the pixel $x(2^J u)$ has a spatial size $\sim 2^J$. If the oversampling factor o is set to 0, the scattering our coefficients are obtained with downsampling of 2^J, which means the Scattering representation of an image can be interpreted as a concatenation of descriptors of non-overlapping patches. With a non-zero oversampling factor, it can be interpreted as a concatenation of descriptors of overlapping patches.

We use a Scattering descriptor with order 1 coefficients and order 2 coefficients. We choose $J = 4$ and hence 4 scales $0 \leq j < J$. We use 8 angles θ evenly spaces in $[0, \pi[$. We set the oversampling factor o to 1. Scattering coefficients are computed with the software package Kymatio [Andreux et al., 2018]. They preserve the image information, and x can be approximately recovered from Sx [Oyallon et al., 2019].

The scattering transform is Lipschitz continuous to translations and deformations [Mallat, 2012]. Intra-class variabilities due to translations smaller than 2^J and small deformations are linearized. Good classification accuracies are obtained with a linear classifier over scattering coefficients in image datasets where translations and deformations dominate intra-class variabilities. This is the case for digits in MNIST or texture images [Bruna and Mallat, 2013].

However, it does not consider the variabilities of pattern structures and clutter that dominate complex image datasets. Removing this clutter while preserving class separation requires some form of supervised learning. The next section introduces a supervised local encoding of Scattering Transform descriptors, implemented in a convolutional network for this purpose.

4.2 Classification with patch separation

4.2.1 Supervised local encoding of Scattering.

We propose to apply a local encoding to the Scattering transform descriptor. This encoding Φ is learned in a supervised way. We choose Φ to be a neural network composed of a cascade of N fully connected layers of hidden size n_c with batch-normalization and ReLU non-linearity. We apply this encoding Φ identically on $Sx(u)$, concatenation of Scattering descriptors of spatial size 3×3. Due to the overlapping of 8 pixels of image patches described in the Scattering transform, 3×3 Scattering patch represents an image patch of size $\sim 32 \times 32$. It can be implemented as a cascade of convolutional operators. The first convolutional operator has a spatial size 3 and n_c channels. The following ones have a spatial size 1×1 and n_c channels on top of the Scattering Descriptors with spatial index u. Our supervised local encoding is a vector of dimension n_c and can be written $\Phi(Sx(u)) \in \mathbb{R}^{n_c}$.

4.2.2 Patch separation

A global average pooling follows this cascade of 1×1 convolutions and a fully connected linear classifier W. The result of this linear prediction is compared with the image label using the standard cross-entropy loss.

Our classification function is thus

$$F(x) = W \sum_u \Phi\left(Sx(u)\right)$$

Since the classification operator W does not take into account the spatial index u, we can write

$$F(x) = \sum_u W\Phi\left(Sx(u)\right)$$

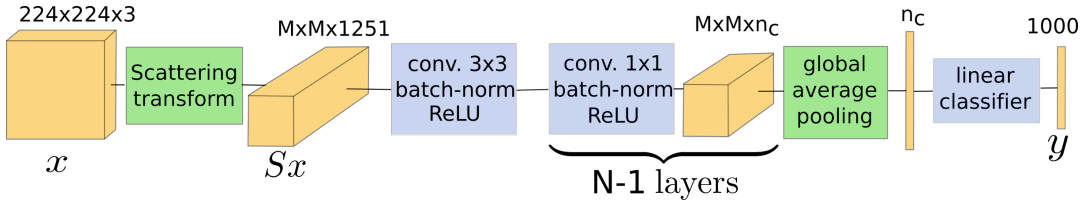
$Sx(u)$ is a descriptor of the image patch. We have here explicitly a separation hypothesis (eq. 1.1) of the classification function with the image patches as sub-variables.

As in the BagNet [Brendel and Bethge, 2019], this patch-separation allows explaining the classification decisions from patch evidence. For each patch encoded in $Sx(u)$, our classification function infers evidences $W\Phi\left(Sx(u)\right)$ for each ImageNet classes. It yields a heatmap of resolution $224/2^{J-o}$ that shows which image patches contribute most to the classification decisions.

4.3 Image Classification on ImageNet

ImageNet 2012 is a challenging color image dataset of 1.2 million training images and 50,000 validation images, divided into 1000 classes. Before convolutional networks, SIFT representa-

Figure 4.1: Architecture of the presented convolutional neural network. Scattering Transform with $J = 4$ and order 2 coefficients is a three-dimensional tensor. It has 1251 channels and a spatial size $M \times M$. When using no oversampling, M is equal to 14, and with an oversampling factor $o = 1$, M is equal to 28. In the main experiment, we used $n_c = 2048$ and $N = 8$ layers for the non-linear encoding part. After the global average pooling, the image representation is simply a vector of dimension n_c. The final vector's dimension is the number of classes, 1000 for ImageNet. When encoding 16×16 patches, the first 3×3 convolution is replaced by a 1×1 convolution.



tions combined with Fisher vector encoding and a linear classifier reached a Top 5 classification accuracy of 74.3% with multiple model averaging [Perronnin et al., 2010]. This technique uses patch separation with a patch size of size 24×24. In their PyTorch implementation, the Top 5 accuracy of AlexNet and BagNet33 is 79.1%[3] and 87.0%[4] respectively.

For the training, we use a stochastic gradient descent algorithm with a momentum of 0.9 and a weight decay of 10^{-5}. The optimization runs 100 epochs, i.e., 100 loops over the dataset. The initial learning rate is 0.1. We decay the learning rate by a factor of 10 every 30 epochs.

We use the standard cross-entropy loss between the network's predictions and the true labels of the images. For data-augmentation, the image is rescaled using a random ratio in $[3/4, 4/3]$. A random crop with a uniform random scale in $[0.8, 1]$ is performed, and this crop is resized to 224×224. For the validation, the smallest size of the image is resized to 256, and we select the center crop of size 224×224.

With an oversampling of 1, the Scattering Transform Sx at a scale $2^J = 16$ of an ImageNet color image is a three-dimensional tensor with spatial dimensions 30×30 and a channel dimension equal to 1251. This representation, with a linear classifier, achieves an accuracy of 41.6% top5. Using the proposed non-linear local encoding with $N = 8$ layers of width 2048, we obtain an accuracy of 84.5% top5. The non-linear local encoding brings an improvement of 42.9% and reaches an accuracy in the range of the BagNets with a similar patch size (see table 4.1).

To identify the relevant parameters for the performance, we perform an ablation study for the proposed architecture. First, we remove the 3×3 concatenation of Scattering descriptors. We simply encode Scattering descriptors with a sequence of 1×1 convolutions. It corresponds to an encoding of $\sim 16 \times 16$ patches. The performance drops by 5.7 %, yielding a 78.8 % top5 accuracy. This performance drop is consistent with the 5.8 % performance drop of the Bagnets between patch sizes 32×32 and 16×16. Now, we vary the parameters of the two blocks: the

[3]Accuracy taken from https://pytorch.org/docs/master/torchvision/models.html
[4]Using the authors' implementation: https://github.com/wielandbrendel/bag-of-local-features-models

Table 4.1: Top 1 and Top 5 accuracy on ImageNet of Fisher Vectors [Perronnin and Larlus, 2015], AlexNet [Krizhevsky et al., 2012], BagNet17 and BagNet33 [Brendel and Bethge, 2019], Scattering with linear classifier (ours) and Scattering with supervised local encoding (ours). Accuracy of AlexNet and BagNets are obtained from Pytorch implementations.

	Fisher Vectors	AlexNet	BagNet17	BagNet33	Scattering + linear	Scattering + non-linear enc.
Top1	-	56.5	58.8	66.7	23.4	63.8
Top5	74.3	79.1	81.2	87.0	41.6	84.5

Scattering transform and the non-linear encoding.

Scattering When removing the coefficients of order 2 in the Scattering transform descriptor, the performance drops of 2.3%, yielding an accuracy of 82.2%. As explained before, second-order coefficients are meant to recover the information eliminated by the averaging by ϕ_J of $|x \star \psi_{j,\theta}|$. The relatively small performance drop suggests that this lost information does not significantly affect the classification. In the following ablation experiments, we keep removing the order 2 Scattering coefficients as they significantly speed up the computations. Hence, the reference accuracy is 82.2% in the following.

When using no oversampling in the Scattering Transform, we observe a performance drop of 2.7%, yielding an accuracy of 79.8%. This performance is comparable with the CNN AlexNet [Krizhevsky et al., 2012]. The oversampling increases the number of encoded patches. The spatial size of the Scattering Transform of an image of size 224×224 increases from 14×14 to 28×28 when using an oversampling factor $o = 1$. We suppose that increasing this spatial resolution reduces the final vector's variance after the global average pooling while keeping the class separated. Hence, the classification performance increases.

Non-linear encoding After the Scattering transform, our pipeline's second block is the non-linear encoding composed of 1×1 convolutions. We perform an ablation study on the parameters of this encoding, i.e., the number of layers N and the width of the layer n_c. As in standard CNNs, we expect the performance to drop when reducing the width and the number of layers.

Using a layer of width 1024 instead of 2048, we observe a performance drop of 1.7%, yielding an accuracy of 80.5%. When using 4 layers of 1×1 convolutions instead of 8, we observe a performance drop of 3.0% yielding an accuracy of 79.2%. Our model follows the general trend of convolutional neural networks: the deeper, the better, the wider, the better.

4.4 Discussion

We demonstrated that learning a non-linear encoding of image patches on top of a predefined scattering representation allows reaching competitive accuracy on ImageNet. The resulting deep convolutional network is a scattering transform followed by a supervised local encoder. The classification function relies explicitly on a patch separation. Classification decisions can thus be explained in terms of patch evidence.

Compared to the BagNets [Brendel and Bethge, 2019], we have presented here a structured convolutional neural network based on patch separation. Using a predefined Scattering transform based on wavelets, we encode 32×32 images in a non-linear way. This relatively shallow encoding of 10 layers is sufficient to obtain competitive accuracy, compared to the 50 layers of the ResNet backbone used by Brendel and Bethge [2019].

This result opens, among others, two main research directions:

- The first one is to propose a model for the supervised local encoding achieved by the sequence of 1×1 convolutions. This was done by Zarka, Thiry, Angles, and Mallat [2019]. They proposed to model this encoding as a ℓ^1 sparse code in a dictionary matrix learned for classification.

- The second one is to study the properties of the image patches for image classification. The first block of the proposed architecture, the Scattering Transform, is meant to create an invariant (or stable) representation of image patches w.r.t. geometric deformations. Can we remove this encoding and work with raw image patches? Is there already regularity and structure in the raw image patches? How does the performance change when we replace the learned encoding with K-nearest neighbors encoding? We address these questions in the next chapter.

Chapter 5

Image classification with patches K-nearest-neighbors

In the previous chapters, we have observed the efficiency of local methods for energy regression and image classification. Atomic neighborhoods and small image patches contain most of the information for energy regression and image classification on usual benchmarks. For energy regression, the SOAP kernel proposed by Bartók et al. [2013] ensures translation and rotation invariance, stability to deformations and locality on the energy. For image classification, the Scattering Transform [Mallat, 2012] is also a local descriptor that ensures invariance properties for image classification. But we have seen that it is not sufficient to have state-of-the-art performances on complex datasets. One needs to learn a non-linear encoding on top of Scattering Transform to have good image classification performance. This learned representation is hard to understand and interpret. Hence, we have no insights into the reasons for the success of patch-based methods.

To understand this success, we first look at the performances of conceptually simple image classification baseline based on patches. The simplest baseline we can think of is a K-nearest-neighbor classifier. K-nearest-neighbors do not incorporate prior invariance information, they simply use the underlying structure in the data. What is the performance of a patch K-nearest-neighbor-based classifier? Is Alexei Efros' motto *"Brain-dead lookup, a.k.a. nearest-neighbors, often works surprisingly well"*[1] true even in the context of image classification ? Do we get better performance when applying this K-nearest-neighbors strategy at the patch level rather than at the image level? How does it compare with predefined invariance-based representations like Scattering Transform?

For the whole image, K-nearest-neighbor classifiers give non-trivial but poor results. On CIFAR 10, a K-nearest neighbor classifier on the raw image yields a classification accuracy of 58.3%[2]. On the Imagenet 2010 results table[3], Georges Quenot from *Laboratoire d'informatique de Grenoble* obtains a 30.5% top5 accuracy using a *"K-Nearest-Neighbors with color histogram and Gabor texture, optimized for the flat measure"*. We hypothesize that the images are too

[1] See for example these slides or this talk.

[2] https://gist.github.com/louity/c6b0c91810c9957f57c56c952323b29e

[3] http://image-net.org/challenges/LSVRC/2010/results

large and too high-dimensional to classify them using K-nearest-neighbors successfully.

To our knowledge, there are no published results of K-nearest-neighbor techniques based on image patches that outperform these results with K-nearest-neighbors at the image level. Indeed, designing a K-nearest-neighbor classifier based on image patches is not straightforward. Assigning the class of the image to all the patches of an image might be a bit rash. Only a small fraction of the patches of an image are informative of the class [Brendel and Bethge, 2019, Geirhos et al., 2019]. Then, one has to aggregate small evidence at the patch level into a global classification decision via a voting system.

In this chapter, we present a classification method based on raw image patches K-nearest-neighbors. We use the Mahanalobis Euclidean distance, [Chandra et al., 1936] which is the usual ℓ^2 Euclidean distance after a linear whitening operation. We encode the natural image patches K-nearest neighbors in a dictionary of randomly selected patches. This representation combined with a linear classifier outperforms by a very large margin K-nearest-neighbors at the image level. We obtain 88.5 % accuracy on CIFAR-10, which is in the range of sophisticated convolutional kernel methods. On ImageNet, such a simple approach exceeds all existing non-learned representation methods by a substantial margin. The presented method shall hence serve as a new baseline for image classification without representation learning.

This chapter is organized as follows. First, we draw connections between the presented method and Convolutional Kernel Methods. Then, we explain precisely how our visual representation is built. We present classification results on the vision datasets CIFAR-10 and the large-scale ImageNet. Finally, we discuss these results and analyze the low-dimensional properties of image patches for classification.

This work has been published in Thiry, Arbel, Belilovsky, and Oyallon [2021]. My personal contributions concerned the proposition of the K-nearest neighbors encoding of image patches, the PyTorch implementation of the method[4] and the achievement of all classification experiments on CIFAR-10 and ImageNet.

[4]https://github.com/louity/patches

5.1 Convolutional Kernel Methods

One can analyze the presented method through the lens of finite-dimensional convolutional kernel methods. In general, convolutional kernel methods can achieve reasonable performances on the CIFAR-10 dataset. Due to their computational cost, it remains open to what extent they achieve similar performances on more complex datasets such as ImageNet. Data-driven convolutional kernels compute a similarity measure between two images x and y using statistics from the training set of images $\mathcal{X}$. In particular, we focus on similarities K that are obtained by first standardizing a representation Φ of the input images and then feeding it to a predefined kernel k:

$$\mathcal{K}_{k,\Phi,\mathcal{X}}(x, y) = k(L\Phi x, L\Phi y)\,, \tag{5.1}$$

where a rescaling and shift is (potentially) performed by a diagonal affine operator $L = L(\Phi, \mathcal{X})$ and is mainly necessary for the optimization step Jin et al. [2009]: it is typically a standardization. The kernel $\mathcal{K}(x, y)$ is said to be *data-driven* if Φ depends on training set $\mathcal{X}$, and *data-independent* otherwise. The convolutional structure of the kernel $\mathcal{K}$ can come either from the choice of the representation Φ (convolutions with a dictionary of patches [Coates et al., 2011]) or by the design of the predefined kernel k [Shankar et al., 2020], or a combination of both [Li et al., 2019, Mairal, 2016].

Our methodology is based on ablation experiments: we would like to measure the effect of incorporating data while reducing other side effects related to the design of Φ, such as the depth of Φ or the implicit bias of a potential optimization procedure. Consequently, we focus on 1-hidden layer neural networks of any widths, which have favorable properties, like the ability to be a universal approximator under non-restrictive conditions. The output linear layer shall be optimized for a classification task, and we consider the first layers, which are predefined and kept fixed, similarly to Coates et al. [2011]. We will see below that simply initializing the weights of the first layer with whitened patches leads to a significant improvement of performances, compared to a random initialization, a wavelet initialization, or even a learning procedure. This patch initialization is used by several works [Li et al., 2019, Mairal, 2016] and is implicitly responsible for their good performances. Other works rely on a whitening step followed by very deep kernels [Shankar et al., 2020], yet we noticed that this was not sufficient in our context. Here, we also try to understand why incorporating whitened patches is helpful for classification. Informally, this method can be thought of as one of the simplest possible in the context of deep convolutional kernel methods. We show that the depth or the non-linearities of such kernels play a minor role compared to patches. In our work, we decompose and analyze each step of our feature design on gold-standard datasets and find that a method based solely on patches and simple non-linearities is actually a strong baseline for image classification.

We investigate the effect of patch-based pre-processing for image classification through a simple baseline representation that does not involve learning (up to a linear classifier) on both CIFAR-10 and ImageNet datasets: the path from CIFAR-10 to ImageNet had never been explored until now in this context. Thus, we believe our baseline to be of high interest for understanding ImageNet's convolutional kernel methods, which almost systematically rely on

a patch (or descriptor of a patch) encoding step. Indeed, this method is straightforward and involves limited ad-hoc feature engineering compared to the deep learning approach: here, contrary to [Mairal, 2016, Coates et al., 2011, Recht et al., 2019, Shankar et al., 2020, Li et al., 2019] we employ modern techniques that are necessary for scalability (from thousands to millions of samples) but can still be understood through the lens of kernel methods (e.g., convolutional classifier, data augmentation, ...). Our method allows understanding the relative improvement of such encoding step. We show that our method is a challenging baseline for classification on Imagenet: we outperform by a large margin the classification accuracy of former attempts to get rid of representation learning on the large-scale ImageNet dataset.

One of our major contributions is introducing a representation that does not involve learning (up to a linear classifier). To our knowledge, it outperforms by a large margin the classification accuracy of former attempts to get rid of representation learning on the large-scale ImageNet dataset. This baseline is of high interest to understand non-deep learning methods on ImageNet that almost systematically rely on a patch (or descriptor of a patch) encoding step: we show that patches solely are a challenging baseline. The presented method allows understanding the relative improvement of the encoding step.

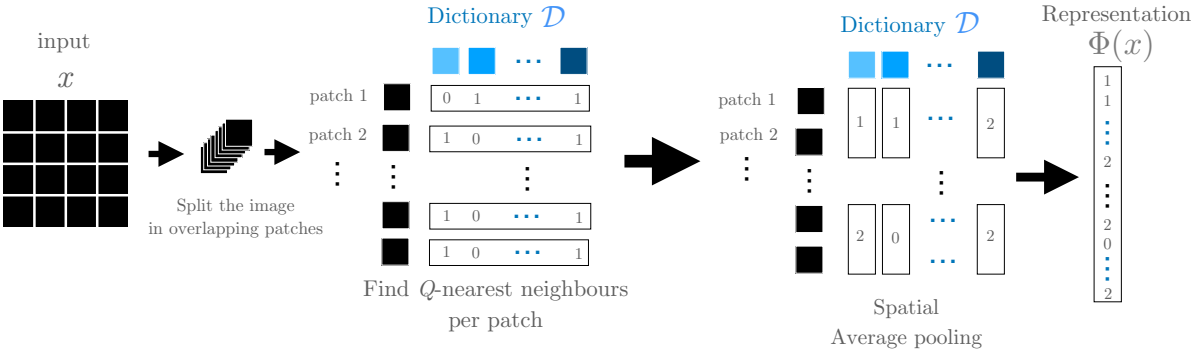
5.2 Method

We first introduce our preliminary notations to describe an image. A patch p of size P of a larger image x is a restriction of that image to a squared domain of surface P^2. We denote by N^2 the size of the natural image x and require that $P \leq N$. Hence, for a spatial index i of the image, $p_{i,x}$ represents the patch of image x located at i. We further introduce the collection of all overlapping patches of that image, denoted by: $\mathcal{P}_x = \{p_{i,x}, i \in \mathcal{I}\}$ where $\mathcal{I}$ is a spatial index set such that $|\mathcal{I}| = (N - P + 1)^2$. Fig. 5.1 corresponds to an overview of our classification pipeline that consists of 3 steps: an initial whitening step of a dictionary $\mathcal{D}$ of random patches, followed by a nearest-neighbor quantization of images patches via $\mathcal{D}$ that are finally spatially averaged.

Whitening We describe the single pre-processing step that we used on our image data, namely a whitening procedure on patches. Here, we view natural image patches of size P^2 as samples from a random vector of mean μ and covariance Σ. We then consider whitening operators which act at the level of each image patch by first subtracting the mean μ then applying the linear transformation $W = (\lambda \mathbf{I} + \Sigma)^{-1/2}$ to the centered patch. The additional whitening regularization with parameter λ was used to avoid ill-conditioning effects.

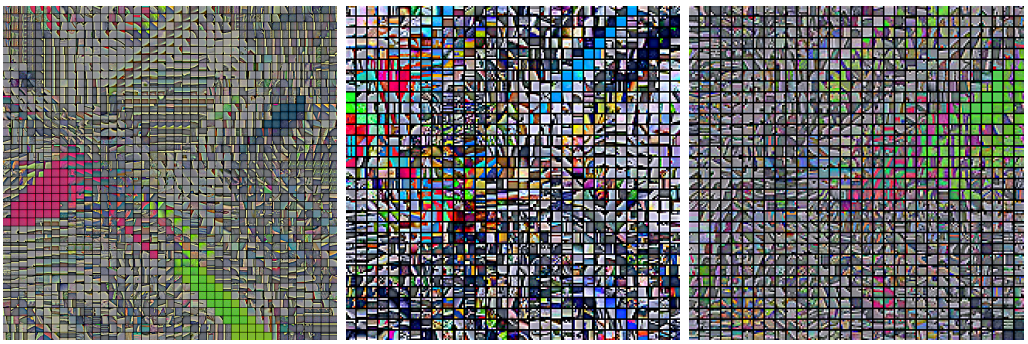
The whitening operation is defined up to an isometry. The Euclidean distance between whitened patches (i.e., the Mahanobolis distance [Chandra et al., 1936]) does not depend upon the choice of isometry leading to PCA, ZCA, etc. This point is detailed in Appendix B. In practice, the mean and covariance are estimated empirically from the training set to construct the whitening operators. For the sake of simplicity, we only consider whitened patches. Unless explicitly stated, we assume that each patch p is already whitened, which holds in particular for the collection of patches in $\mathcal{P}_x$ of any image x. Once this whitening step is performed, the

Figure 5.1: Our classification pipeline described synthetically to explain how we build the representation $\Phi(x)$ of an input image x.



Euclidean distance over patches is approximatively isotropic and is used in the next section to represent our signals.

Figure 5.2: An example of whitened dictionary $\mathcal{D}$ with patch size $P = 6$ from ImageNet-128 (Left), ImageNet-64 (Middle), CIFAR-10 (Right). The atoms have been reordered via a topographic algorithm from Montobbio et al. [2019] and contrast adjusted.



K-Nearest Neighbors on patches This algorithm's basic idea is to compare the distances between each patch of an image and a fixed dictionary of patches $\mathcal{D}$, with size $|\mathcal{D}|$ that is the number of patches extracted. Note that we also propose a variant where we use a soft-assignment operator. For a fixed dataset, this dictionary $\mathcal{D}$ is obtained by uniformly sampling patches from images over the whole training set. We augment $\mathcal{D}$ into $\cup_{d\in\mathcal{D}}\{d, -d\}$ because it allows the dictionary of patches to be contrast invariant and we observe it leads to better classification accuracies; we still refer to it as $\mathcal{D}$. An illustration is given by Fig. 5.2. Once the dictionary $\mathcal{D}$ is fixed, for each patch $p_{i,x}$ we consider the set $\mathcal{C}_{i,x}$ of pairwise distances $\mathcal{C}_{i,x} = \{\|p_{i,x} - d\|, d \in \mathcal{D}\}$. For each whitened patch we encode the K-Nearest Neighbors of $p_{i,x}$ from the set $\mathcal{D}$, for some $K \in \mathbb{N}$. More formally, we consider $\tau_{i,x}$ the K-th smallest element of

$\mathcal{C}_{i,x}$, and we define the K-Nearest Neighbors binary encoding as follow, for $(d, i) \in \mathcal{D} \times \mathcal{I}$:

$$\phi(x)_{d,i} = \begin{cases} 1, & \text{if } \|p_{i,x} - d\| \leq \tau_{i,x} \\ 0, & \text{otherwise.} \end{cases} \tag{5.2}$$

Eq. 5.2 can be viewed as a Vector Quantization (VQ) step with hard-assignment [Coates and Ng, 2011]. The representation ϕ encodes the patch neighborhood in a subset of randomly selected patches and can be seen as a crude description of the image patches' topological geometry. Moreover, it allows viewing the distance between two images x, y as a Hamming distance between the patches neighborhood encoding as:

$$\|\phi(x) - \phi(y)\|^2 = \sum_{i,d} \mathbf{1}_{\phi(x)_{d,i} \neq \phi(y)_{d,i}} \,. \tag{5.3}$$

To reduce the computational burden of our method, we perform an intermediary average-pooling step. Indeed, we subdivide $\mathcal{I}$ in squared overlapping regions $\mathcal{I}_j \subset \mathcal{I}$, leading to the representation Φ defined, for $d \in \mathcal{D}, j$ by:

$$\Phi(x)_{d,j} = \sum_{i \in \mathcal{I}_j} \phi(x)_{d,i} \,. \tag{5.4}$$

Hence, the resulting kernel is simply given by $\mathcal{K}(x, y) = \langle \Phi(x), \Phi(y) \rangle$. One can find implementation details in Appendix B. The next section describes our classification pipeline, as we feed our representation Φ to a linear classifier on challenging datasets.

5.3 Experiments

We train shallow classifiers, i.e., linear classifier and 1-hidden layer CNN (*1-layer*) on top of our representation Φ on two major image classification datasets, CIFAR-10 and ImageNet, which consist respectively of $50k$ small and $1.2M$ large color images divided, respectively into 10 and $1k$ classes. We systematically used mini-batch SGD with a momentum of 0.9 and no weight decay. We used the standard cross-entropy loss.

Classifier parametrization In each experiments, the spatial subdivisions $\mathcal{I}_j$ are implemented as an average pooling with kernel size k_1 and stride s_1. We then apply a 2D batch-normalization [Ioffe and Szegedy, 2015] to standardize our features on the fly before feeding them to a linear classifier. To reduce the linear classifier's size (following the same line of idea of a bottleneck [He et al., 2016]), we factorize it into two convolutional operators. The first one with kernel size k_2 and stride 1 reduces the number of channels from $\mathcal{D}$ to c_2. The second one with kernel size k_3 and stride 1 outputs N_C channels, N_C being the number of image classes. Then we apply a global average pooling. For the 1-hidden layer experiment, we simply add a ReLU non-linearity between the first and the second convolutional layer.

Table 5.1: Classification accuracies on CIFAR-10. VQ indicates whether vector quantization with hard-assignment is applied on the first layer. One layer patch-based classification accuracies on CIFAR-10. Amongst methods relying on random patches, ours is the only approach operating online (and therefore allowing for scalable training).

Method	$\lvert\mathcal{D}\rvert$	**VQ**	**Online**	P	**Acc.**
Coates et al. [2011]	1,000	✓	×	6	68.6
Wavelets [Oyallon and Mallat, 2015]	-	×	×	8	82.2
Recht et al. [2019]	256,000	×	×	6	85.6
SimplePatch (Ours)	2,000	✓	✓	6	82.5
SimplePatch (Ours)	16,000	✓	✓	6	85.6
SimplePatch (Ours)	64,000	✓	✓	6	86.7
SimplePatch (Ours)	64,000	×	✓	6	**86.9**

Table 5.2: Supervised accuracies on CIFAR-10 with comparable shallow supervised classifiers. Here, e2e stands for end-to-end optimized classifier and 1-layer for a 1-layer classifier, and kernel for kernel classifier.

Method	**VQ**	**Depth**	**Classifier**	**Acc.**
SimplePatch (Ours)	✓	2	1-layer	88.5
AlexNet [Krizhevsky et al., 2012]	×	5	e2e	89.1
NK [Shankar et al., 2020]	×	5	kernel	89.8
CKN [Mairal, 2016]	×	9	e2e	89.8

5.3.1 CIFAR-10

Implementation details Our data augmentation consists of horizontal random flips and random crops of size 32^2 after reflect-padding with 4 pixels. For the dictionary, we choose a patch size of $P = 6$ and tested various sizes of the dictionary $|\mathcal{D}|$ and whitening regularization $\lambda = 0.001$. In all cases, we used $K = 0.4|\mathcal{D}|$. The classifier is trained for 175 epoch with a learning rate decay of 0.1 at epochs 100 and 150. The initial learning rate is 0.003 for $|\mathcal{D}| = 2k$ and 0.001 for larger $|\mathcal{D}|$.

Single layer experiments For the linear classification experiments, we used an average pooling of size $k_1 = 5$ and stride $s_1 = 3$, $k_2 = 1$ and $c_2 = 128$ for the first convolutional operator and $k_3 = 6$ for the second one. Our results are reported and compared in Table 5.1. First, note that contrary to experiments done by Coates et al. [2011], our method has surprisingly good accuracy despite the hard-assignment due to VQ. Sparse coding, soft-thresholding, and orthogonal matching-pursuit-based representations used by Coates and Ng [2011], Recht

Table 5.3: Accuracies on CIFAR-10 with Handcrafted Kernels classifiers with and without data-driven reprensentations. For SimplePatch, we replace patches with random Gaussian noise. D-D stands for Data-Driven and D-I for Data-Independent. Whit., Acc., and Improv. stand respectively for whitening, accuracy, and improvement.

Method	VQ	Online	Depth	D-I Acc. (D-D Improv.)	Data used
NK [Shankar et al., 2020]	×	×	5	77.7 (8.1)	ZCA Whit.
Simple Patch (Ours)	✓	✓	1	78.6 (8.1)	Patches + Whit.
CKN [Mairal, 2016]	×	×	2	81.1 (5.1)[5]	Patches + Whit.
NTK [Li et al., 2019]	×	×	8	82.2 (6.7)	Patches + Whit.

et al. [2019] can be seen as soft-assignment VQ and yield comparable classification accuracy (resp. 81.5% with 6.10^3 patches and 85.6% with 2.10^5 patches). However, these representations contain much more information than hard-assignment VQ as they allow to reconstruct a large part of the signal. We get better accuracy with only coarse topological information on the image patches, suggesting that this information is highly relevant for classification. To obtain comparable accuracies with a linear classifier, we use a single binary encoding step compared to, Mairal [2016] and we need a much smaller number of patches than Recht et al. [2019], Coates and Ng [2011]. Moreover, Recht et al. [2019] is the only work in the literature, besides the one presented here, that achieves good performance using a linear model solely with depth one. To test the VQ importance, we replace the hard-assignment VQ implemented with a binary non-linearity $\mathbf{1}_{\|p_{i,x}-d\|\leq\tau_{i,x}}$ (see Eq. 5.2) by a soft-assignment VQ with a sigmoid function $(1+e^{\|p_{i,x}-d\|-\tau_{i,x}})^{-1}$. The accuracy increases by 0.2%, showing that the use of soft-assignment in VQ, which is crucial for performance in Coates and Ng [2011], does not affect our representation's performances.

Importance of data-driven representations As we see in Table5.3, the data-driven representation is crucial for good performances of handcrafted kernel classifiers. We remind that a data-independent kernel is built without using the dataset, which is, for instance, the case with a neural network randomly initialized. The accuracies from Shankar et al. [2020] correspond to Myrtle5 (CNN and kernel) because the authors only report an accuracy without ZCA for this model. As a sanity check, we consider $\mathcal{D}$ whose atoms are sampled from a Gaussian white noise: this step leads to a drop of 8.1%. This is aligned with the finding of each work we compared to: performances drop if no ZCA is applied or if patches are not extracted. Using a dictionary of size $|\mathcal{D}| = 2048$, the same model trained end-to-end (including the learning of $\mathcal{D}$) yields the same accuracy (- 0.1 %), showing that here, sampling patches is as efficient as optimizing them.

Non-linear classification experiments To test the discriminative power of our features, we use a 1-hidden layer classifier with ReLU non-linearity and an average pooling of size $k_1 = 3$

and stride $s_1 = 2$, $k_2 = 3$, $c_2 = 2048$ and $k_3 = 7$. Our results are reported and compared with other non-linear classification methods in Table5.2. Using a shallow non-linear classifier, our method is competitive with end-to-end trained methods [Li et al., 2019, Shankar et al., 2020, Krizhevsky et al., 2012]. This further indicates the relevance of patches neighborhood information for the classification task.

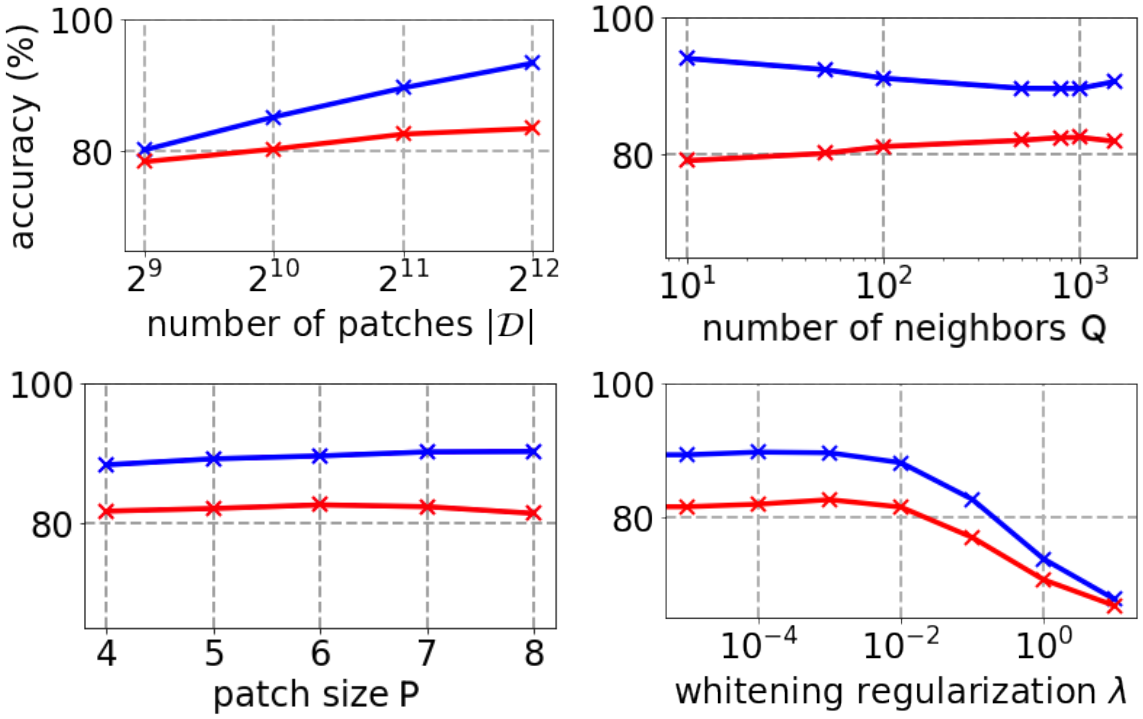
Hyper-parameter analysis CIFAR-10 is a relatively small dataset that allows fast benchmarking. Thus we conducted several ablation experiments to understand the relative improvement due to each hyper-parameter of our pipeline. We thus vary the size of the dictionary $|\mathcal{D}|$, the patch size P, the number of nearest neighbors K and the whitening regularization λ, which are the hyper-parameters of Φ. Results are shown in Fig. 5.3. Note that even a relatively small number of patches is competitive with much more complicated representations, such as Oyallon and Mallat [2015]. While it is possible to slightly optimize the performances according to P or K, the fluctuations remain minor compared to other factors, indicating that our method's performances are relatively stable w.r.t. this set of hyper-parameters. The whitening regularization behaves similarly to a thresholding operator on the eigenvalues of $\Sigma^{1/2}$, as it penalizes larger eigenvalues. Interestingly, we note that this hyper-parameter does almost not affect the classification performances under a certain threshold. This goes in hand with both a fast eigenvalue decay and stability to noise discussed further in the section 5.3.3.

Classifier factorization To test our classifier's inductive bias, we replace it with a simple fully-connected layer, with 6 times more parameters than ours: the train and test accuracies are 93.0% and 81.6% compared to 88.9% and 82.5%. The use of this factorized classifier significantly reduces overfitting while reducing the number of computations. We note that using convolutions is well motivated by the structure of natural images whose class is relatively invariant to translation.

5.3.2 ImageNet

Implementation details To reduce the computational overhead of our method on ImageNet, we followed the same approach as Chrabaszcz et al. [2017]: we reduce the resolution to 64^2, instead of the standard 224^2 length. They observed that this does not alter much the top-performances of standard models (5% to 10% drop of accuracy on average). We believe it introduces a useful dimensionality reduction, as it removes unstable high-frequency parts of images [Mallat, 1999]. We set the patch size to $P = 6$ and the whitening regularization to $\lambda = 10^{-2}$. Since ImageNet is a much larger than CIFAR-10, we restricted to $|\mathcal{D}| = 2048$ patches. As for CIFAR-10, we set $K = 0.4|\mathcal{D}|$. The parameters of the linear convolutional classifier are chosen to be: $k_1 = 10, s_1 = 6, k_2 = 1, c_2 = 256, k_3 = 7$. For the 1-hidden layer experiment, we used kernel size of $k_2 = 3$ for the first convolution. Our models are trained during 60 epochs with an initial learning rate of 0.003 decayed by a factor 10 at epochs 40 and 50. ly to Chrabaszcz et al. [2017] use random flip during training, and we select random crops of size 64, after a reflect-padding of size 8. At testing time, we resize the image to 64.

Figure 5.3: CIFAR-10 ablation experiments, train accuracies in blue, test accuracies in red.



This procedure differs slightly from the usual procedure, consisting of resizing images while maintaining ratios before random cropping.

Classification experiments Table5.4 reports the accuracy of our method, as well as the accuracy of comparable methods. Despite a smaller image resolution, our method outperforms by a large margin ($\sim 10\%$ Top5) the Scattering Transform [Mallat, 2012], which was the previous state-of-the-art-method in the context of no-representation learning. It also outperforms randomly initialized neural networks [Arandjelovic et al., 2017]. Note that our representation uses only 2.10^3 randomly selected patches, a tiny fraction of the billions of ImageNet patches.

Sánchez et al. [2013] obtain 72.0% top-5 accuracy with a patch representation computed in three steps (SIFT, Fisher kernel, power-normalization/compression). This patch representation lives in dimension 6.10^4 in the best setting. Using a smaller patch representation of dimension 4.10^3, they obtain 64.1% top-5 accuracy. The performance of our method tested on lower resolution images (128^2) using a representation of dimension 2.10^3 ($= |\mathcal{D}|$) is relatively close to the performance of the 4.10^3 dimensional Fisher Vectors, but further large scale experiments would be needed to confirm if this holds for higher dimensions. This visual representation involves the learning of a Gaussian mixture model and several PCA dimensionality reductions. These engineering choices are crucial for performance [Perronnin et al., 2010]. Still, a major difference between our representation and both Scattering Transform and Fisher Vector is the hard-assignment VQ that discards signal information. The image can be fairly reconstructed using Scattering coefficients or SIFT descriptors starting from random noise and optimizing

the pixels by gradient descent [Oyallon et al., 2017, Weinzaepfel et al., 2011]. We can not use this technique since the hard-assignment VQ zeros all the gradients.

In Table5.5, we compare our performances with supervised models trained end-to-end using convolutions with small receptive fields. Here, $\mathcal{D} = 2k$. BagNets [Brendel and Bethge, 2019] have shown that competitive classification accuracies can be obtained with patch-encoding that consists of 50 layers. Our shallow experiment's performance with a 1-hidden layer classifier is competitive with a BagNet with a similar patch-size. It suggests once again that hard-assignment VQ does not degrade much of the classification information. We also note that our approach with a linear classifier outperforms supervised shallow baselines that consist of 1 or 2 hidden-layers CNN [Belilovsky et al., 2018]. This indicates that a patch-based representation is a non-trivial baseline for shallow CNNs.

To measure the resolution's importance on the performances, we run a linear classification experiment on ImageNet images with twice bigger resolution ($N = 128^2$, $P = 12, k_1 = 20, s_1 = 12$). We observe that it improves classification performances. Note that the patches used are in $432-$ dimensional space, which is a very high dimension. This improvement is surprising since distances to nearest-neighbors are known to be meaningless in high-dimension [Beyer et al., 1999]. This shows a form of low-dimensionality in the natural image patches that we study in the next section.

Random filters and whitening On Imagenet64, removing the whitening step leads to an accuracy of 18% top-1, i.e., a drop of about 16%. As for CIFAR-10, this step is crucial for performance.

Table 5.4: Handcrafted accuracies on ImageNet, via a linear classifier. No other weights are explicitely optimized. Res. stands for resolution.

Method	$\lvert\mathcal{D}\rvert$	**VQ**	P	**Depth**	**Res.**	**Top1**	**Top5**
Random [Arandjelovic et al., 2017]	-	×	-	9	224	18.9	N.C.
Wavelets [Zarka et al., 2019]	-	×	32	2	224	26.1	44.7
SimplePatch (Ours)	2,000	✓	6	1	64	33.2	54.3
SimplePatch (Ours)	2,000	✓	12	1	128	35.9	57.4
SimplePatch (Ours)	2,000	×	12	1	128	36.0	**57.6**

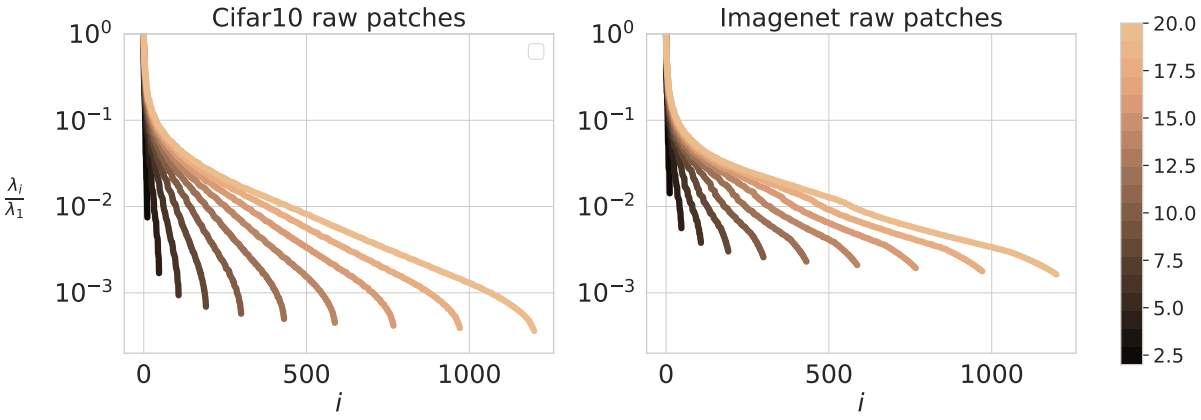
5.3.3 Dictionary structure

The performance obtained with the surprisingly simple classifier hints at a low dimensional structure in the classification problem that is exploited by the patch-based classifier we proposed. This motivates us further to analyze the structure of the patches' dictionary to uncover a lower-dimensional structure and investigate how the whitening, which highly affects performance, relates to such lower-dimensional structure.

Table 5.5: Supervised accuracies on ImageNet, for which our model uses $|\mathcal{D}| = 2048$ patches. Res., e2e, 1-layer, N.C. respectively stand for resolutions, end-to-end, 1-hidden layer classifier, not communicated

Method	VQ	P	Depth	Res.	Classifier	Top1 / Top5
Belilovsky et al. [2018]	×	-	1	224	e2e	N.C. / 26
Belilovsky et al. [2018]	×	-	2	224	e2e	N.C. / 44
SimplePatch (Ours)	✓	6	2	64	1-layer	39.4 / 62.1
BagNet [Brendel and Bethge, 2019]	×	9	50	224	e2e	N.C. / 70.0
AlexNet[Krizhevsky et al., 2012]	-	×	10	224	e2e	56.5 / 79.1

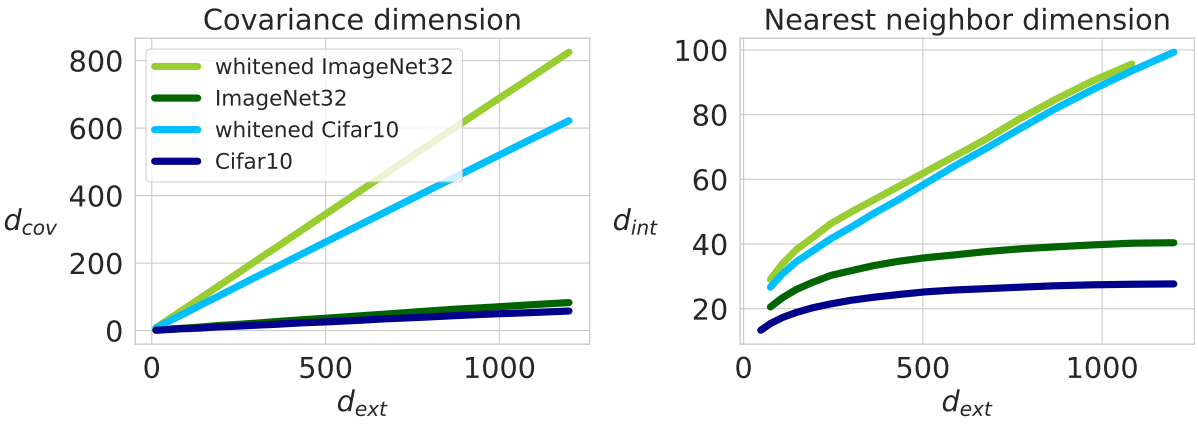
Figure 5.4: Spectrum of $\Sigma^{1/2}$ on CIFAR-10 (left) and ImageNet-64 (right) using small patch sizes in dark-brown to larger patch sizes in light-brown.



Spectrum of $\mathcal{D}$ As a preliminary analysis, we propose to analyse the singular values (spectrum) of $\Sigma^{1/2}$ sorted by a decreasing order as $\lambda_1 \geq ... \geq \lambda_{d_{ext}}$ with $d_{\text{ext}} = 3P^2$ being the extrinsic dimension (number of colored pixels in each patch). From this spectrum, it is straightforward to compute the covariance dimension d_{cov} of the patches defined as the smallest number of dimensions needed to explain 95% of the total variance. In other words, d_{cov} is the smallest index such that $\sum_{i=1}^{d_{cov}} \lambda_i \geq 0.95 \sum_{i=1}^{d_{ext}} \lambda_i$. Fig. 5.4 shows the spectrum for several values of P, normalized by λ_1 on CIFAR-10 and ImageNet-32. The first observation is that patches from the ImageNet-32 dataset tend to be better conditioned than those from CIFAR-10 with a conditioning ratio of 10^2 for ImageNet vs 10^3 for CIFAR-10. This is probably due to the use of more diverse images than on CIFAR-10. Second, note that the spectrum tends to decay exponentially (linear rate in semi-logarithmic scale). This rate decreases as the patch's size increases (from dark brown to light brown) suggesting an increased covariance dimension for larger patches. This is further confirmed in Fig. 5.5(left) which shows the covariance dimension d_{cov} as a function of the extrinsic dimension d_{ext}, with and without whitening. Before whitening, this linear dimension is much smaller than the ambient dimension: whitening the patches increases the patches' linear dimensionality, which still increases at a linear growth as

a function of P^2.

Figure 5.5: Covariance dimension (left) and nearest neighbor dimension (right) as a function of the extrinsic dimension of the patches.



Intrinsic dimension of $\mathcal{D}$ We propose to refine our measure of linear dimensionality to a non-linear measure of the intrinsic dimension. Under the assumption of a non-linear manifold, the linear dimensionality is only an upper bound of image patches' true dimensionality. To get more accurate non-linear estimates, we propose to use the notion of intrinsic dimension d_{int} introduced in [Levina and Bickel, 2004]. It relies on a local estimate of the dimension around a patch point p, obtained by finding the k-Nearest Neighbors to this patch in the whole dataset and estimating how much the Euclidean distance $\tau_k(p)$ between the k-Nearest Neighbor and patch p varies as k increases up to $K \in \mathbb{N}$:

$$d_{\text{int}}(p) = \left(\frac{1}{K-1} \sum_{k=1}^{K-1} \log \frac{\tau_K(p)}{\tau_k(p)} \right)^{-1}. \tag{5.5}$$

In high dimensional spaces, it is possible to have many neighbors that are equi-distant to p, thus $\tau_k(p)$ would barely vary as k increases. As a result the estimate $d_{\text{int}}(p)$ will have large values. Similarly, a small dimension means large variations of $\tau_k(p)$ since it is not possible to pack as many equidistant neighbors of p. This results in a smaller value for $d_{\text{int}}(p)$. An overall estimate of the d_{int} is then obtained by averaging the local estimate $d_{\text{int}}(p)$ over all patches, i.e. $d_{\text{int}} = \frac{1}{|\mathcal{D}|} \sum_{p \in \mathcal{D}} d_{\text{int}}(p)$. Fig. 5.5 (right) shows the intrinsic dimension estimated using $K = 4 \cdot 10^3$ and a dictionary of size $|\mathcal{D}| = 16 \cdot 10^3$. In all cases, the estimated intrinsic dimension d_{int} is much smaller than the extrinsic dimension $d_{\text{ext}} = 3P^2$. Moreover, it grows even more slowly than the linear dimension when the patch size P increases. Finally, even after whitening, d_{int} is only about 10% of the total dimension, which is strong evidence that the natural image patches are low dimensional.

5.4 Discussion

We've presented an image classification method that is not based nor inspired by the image class's invariance w.r.t. transformations. The representation used is based solely on Mahanalobis distances between raw image patches.

Interestingly, recently developed convolutional kernel methods for image classification [Recht et al., 2019, Mairal, 2016, Li et al., 2019, Shankar et al., 2020, Lu et al., 2014] share an additional implicit ingredient. Li et al. [2019], Mairal [2016], Mairal et al. [2014], Recht et al. [2019] use of a dictionary of whitened patches and Shankar et al. [2020] use a whitening step of the whole image. Ablation experiments show the crucial importance of the whitening aspect for the performance of these methods . We obtain here a comparable performance with a single whitening step followed by K-nearest-neighbors encoding. Without this whitening step, our performance drops significantly. It proposes hence a line of explanation for the whitening step's importance in convolutional kernel methods. The Mahanalobis distance is much more relevant than the usual Euclidean distance to have discriminative patch's nearest-neighbors.

Our results reveal implicitly that there is a low-dimensional structure on the raw image patches. This low-dimensional structure might explain in part the success of data-driven kernels and CNNs based on patch separation. We used a shallow, predefined visual representation, which is not optimized by gradient descent. Surprisingly, this method is highly competitive with other data-driven kernels.

Due to limited computational resources, we restricted ourselves on ImageNet to small image resolutions and a relatively small number of patches. Conducting proper large-scale experiments on the challenging ImageNet dataset would be of great interest to have further insights on the properties of image patches for classification.

Chapter 6

Creativity in human-machine artistic interaction

Recently developed algorithms have found applications in the field of artistic creation. For example, CNNs used for image classification can be used to perform artistic style transfer on images [Gatys et al., 2015]. Generative adversarial networks (GANs) developed initially for image generation were used to generate "artistic" images [Elgammal et al., 2017]. A so-called "AI art" movement[1] is emerging from the use of these new techniques in the artistic field. They often portray the algorithm as creative, for example, the *Creative adversarial networks* [Elgammal et al., 2017]. But creativity is a vast notion. Can we propose a more precise characterization of the creativity and creative agency of these algorithms?

The power of these new algorithms and the new interaction possibilities they offer to artists are opportunities to understand better and grasp the qualitative difference between the artist's and the algorithm's creativity. This chapter proposes to design interactive creative systems, focusing on a painting on a canvas. We present two modalities of interactions. The first one, *Dialog on a canvas with a machine*, consists of an iterative interplay between an artist and an algorithm generating strokes. We developed it in collaboration with *Tina&Charly*. The second one, *Interactive Neural Style Transfer*, is an interactive painting setting with a painter and its own style using style transfer techniques. We developed it in collaboration with Erwann Kerdreux. Development of these two systems, we realize how enriching the collaboration with artists is when thinking about creativity. Moreover, the experiments conducted with these two systems allow us to characterize some aspects of the algorithm's creativity compared to the artist.

This work has been published in Cabannes, Kerdreux, Thiry, Campana, and Ferrandes [2019] and Kerdreux, Thiry, and Kerdreux [2020]. It is the result of a global collaboration with equal contributions between the authors.

[1] https://aiartists.org/

6.1 Dialog on a canvas with a machine

The first artist-algorithm interaction setting is a pictorial game consisting of interactive rounds of creation between artists and an algorithm. They repetitively paint one after the other. At its rounds, the computer partially completes the drawing using machine learning algorithms and projects its additions directly on the canvas, which the artists are free to insert or modify. Alongside fostering creativity, the process is designed to question the growing interaction between humans and machines.

6.1.1 Creation Process.

With the on-going technological revolution, human-machine interaction is deeply evolving. Hence art creation could benefit from new tools while simultaneously clarifying how these interactions affect humans.

Contrarily to previous works in which the humans are either engineers or curators of an algorithm that produce potential artworks [Tan et al., 2017, Jin et al., 2017, Donahue and McAuley, 2017, Elgammal et al., 2017, Zhao et al., 2018, Chu, 2018, Lin and Su, 2018, Grimm, 2018, Jetchev et al., 2018], We propose a new utilization of the machine, integrating it at the core of a human creative process. The idea is to suggest to humans, while painting, ramifications, and directions of their on-going artwork. In the following, we approach this generic idea under a specific interactive framework.

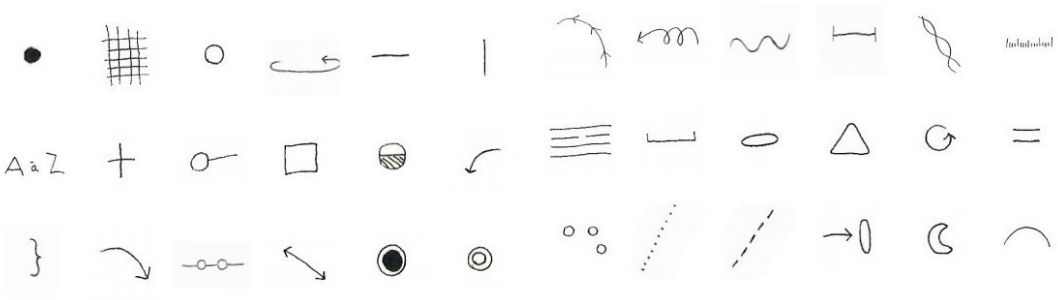


Figure 6.1: *Tina&Charly* glossary, which could be given to a computer as a pictorial vocabulary.

For this purpose, we worked with the artist duo *Tina&Charly*. They have explored interaction using canvas as a media using their own glossary (Fig 6.1). To begin a creation, they choose a theme and symbolize in black on a white canvas. Then a game starts. At each round, using a basis of strokes and symbols that forms their pictorial vocabulary, Charly waits for Tina to schematize her emotions and thoughts in red before answering her in green on the on-going painting. Rounds follow up until a consensus is reached about ending the painting. The whole process takes place in silence, the only dialog being on the canvas.

The goal of this method is to introduce an algorithm as a third player in *Tina&Charly*'s dialog. The AI machine first captures a raw representation of the painting, then analyzes this signal to partially complete the on-going painting. This completion is projected back on the canvas. At this point, the artists are free to incorporate the machine's suggestion in blue,

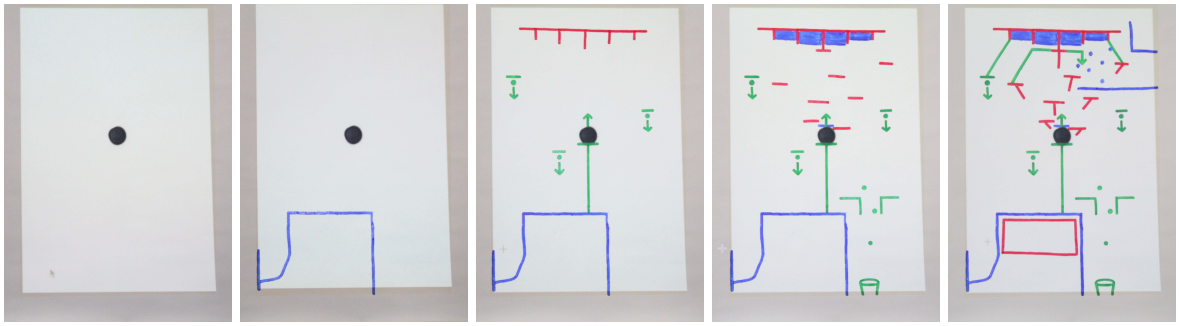


Figure 6.2: Computer captures of the on-going *Monopole* painting. Before taking the picture, the system projects white light on the canvas to better light and ease the following processing steps.

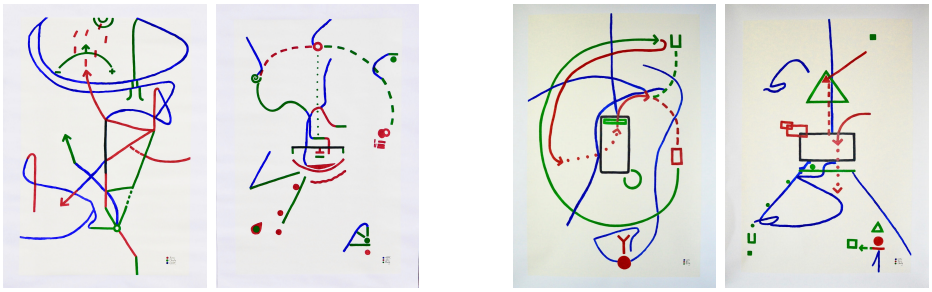


Figure 6.3: Four 110 × 160 cm acrylic on canvas paintings. Two diptychs: *Active, Passive* and *Emitter, Receptor*, from left to right. The blue strokes are the computer suggestions projected on the canvas and interpreted by the artists. In *Emitter* it only completes its own strokes or the black strokes of the canvas, while in its dual *Receptor*, it completes any. It symbolizes a landmark of human-machine interaction when a human starts to systematically send back information that could condition the machine's actions.

a color that has not been assigned to any player. In the end, having used different colors allows analyzing players' contributions. Figure 6.2 shows several creation steps of the painting *Monopole*.

6.1.2 Installation and Specifications.

The engineered system is composed of a camera and a projector connected to a computer on a fixed support (see Figure 6.4). At computer round, the system acquires an image of the painting and analyzes it to recover the exact canvas strokes. This pre-processing was made robust to most luminosity variation for the interaction to be applicable in any studio in a seamless fashion. Those strokes feed a *neural sketcher* that outputs new strokes to add to the painting. Finally, post-processing allows projecting those additions back on the canvas.

The neural sketcher is a recurrent neural network based on the recent powerful improvement [Ha and Eck, 2018] of the seminal work of [Graves, 2013]. It is fed using doodling representation as a sequence of points and a channel encoding for stroke breaks [Graves, 2013, Ha and Eck,

Figure 6.4: Some images of the painting process in Atelier 6B, Saint-Denis, France. (Top left) Artists install canvas while computer scientists install their machine. The machine is made of a camera, a computer and a projector, it is highly portable. (Top middle) The artist draws under the scrutiny of the computer. (Top left) The computer analyzes the on-going painting in order to suggest additions. (Bottom left) Those suggestions are projected on the canvas for the artists to discuss addition. (Bottom middle) Additions are incorporated in blue on the canvas. (Bottom right) At the end, the artists apply a glaze mixture to protect their creations.

2018]. The sketcher then outputs a similar series that we convert back as strokes on the original painting. To train the network, we used the QuickDraw data set [Jongejan et al., 2018], It enables the network to produce human-like strokes. For a smoother integration with *Tina&Charly* style, we further refined the learning using a sketch database from previous paintings of the artists, collected by finding strokes of these and decomposing them into ordered points.

6.1.3 Fostering Creativity.

The artists found the machine strokes surprising and suggestive of moves they would not have done by themselves. Some painters have actually expressed how evocative unintended strokes could be [Deleuze, 1981, Chapter XII]. Our installation, where the machine projects completions without painting, combined with generative network capability, allows us to explore that in a principled way. Furthermore, the ability to change parameters, such as the learning data

set or the amount of completion, adds more degree for the human to control their use of the machine.

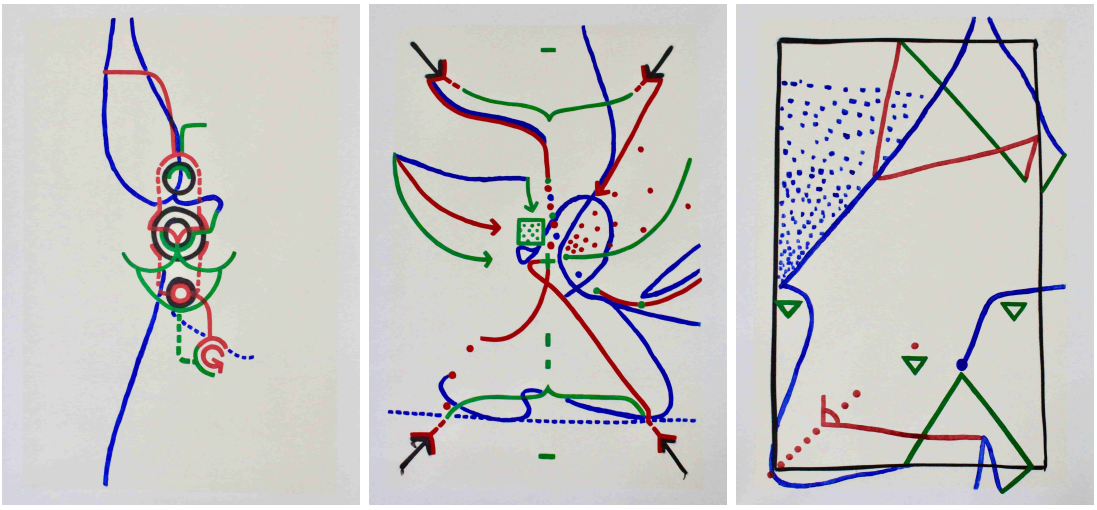


Figure 6.5: Three acrylic on canvas paintings, each 110 × 160 cm. Those are the first three paintings of the series *Influence, Convergence, Contrôle* and *Monopole*.

6.1.4 Human and Machine Interplay.

Our physically interactive installation aims to be used by anybody, hoping to raise awareness and initiate human and machine interplay thoughts. Arguably, it embodies that our technology use is a middle ground where machines are made human-friendly and human drift from their original routines and spaces. Indeed, *Tina&Charly* felt interacting with a full-body system – it has been designed to superficially borrow as much as possible human-like painting behavior. They experienced the machine as sometimes constraining, hard to grasp, and sometimes magical, infusing new dimensions to the painting. Feeling, while in the creative process, that the machine could either be collaborative or muzzling was an unexpected echo to what technologies seem to be in our daily life.

From an outside perspective, the machine distorts their original painting style, both on the short-term artworks resulting from their interaction (see Figure 6.6) and on their long-term body of work as it inspired them on their machine-free paintings. As such, the interaction is not innocuous, even though, contrarily to our daily experience, we have made the machine impact as explicit as possible with its recognizable blue contributions.

6.2 Interactive neural style transfer with artists

We present interactive painting processes in which a painter and various neural style transfer algorithms interact on a real canvas. Understanding what these algorithms' outputs achieve is paramount to describing the creative agency in our interactive experiments. We gather a set of paired painting-pictures images and present a new evaluation methodology based on the

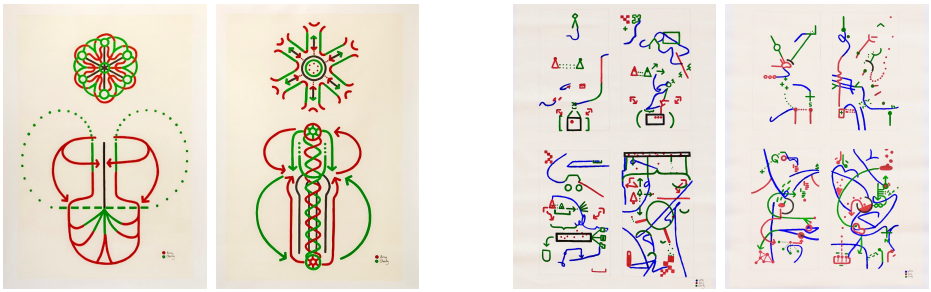


Figure 6.6: Comparing the style of *Tina&Charly*'s diptych without the algorithm (left) and with the algorithm (right). Acrylic on canvas, 110 × 160 cm.

predictivity of neural style transfer algorithms. We point some algorithms' instabilities and show that they can enlarge the diversity and pleasing oddity of the images synthesized by the numerous existing neural style transfer algorithms. This diversity of images was perceived as a source of inspiration for human painters, portraying the machine as a *computational catalyst*.

6.2.1 Introduction

Neural style transfer [Gatys et al., 2015], which seeks at rendering the content of one image using the style of another, provides impressive results as it takes advantage of the rich hierarchical representation of images produced by convolutional neural networks (CNN) to quantify the style and content of images. The many ways to manipulate these complex maps and their increasing ease of implementation have underpinned the development of many successful methods in this area of computational artistic rendering.

Several evaluation techniques exist to compare all different methods. On the one hand, many methods focus on quantifying how much a neural style transfer method attains a numerical objective. These are good engineering indicators, but we highlight that they are not necessarily relevant to measuring the quality of the style transfer algorithm's outputs. On the other hand, qualitative evaluation methods typically collect many subjective impressions on the algorithms' outputs. It provides average scores on the content preservation and style quality of the algorithms' outputs. However, it does not reveal the specificity of an algorithm compared to another.

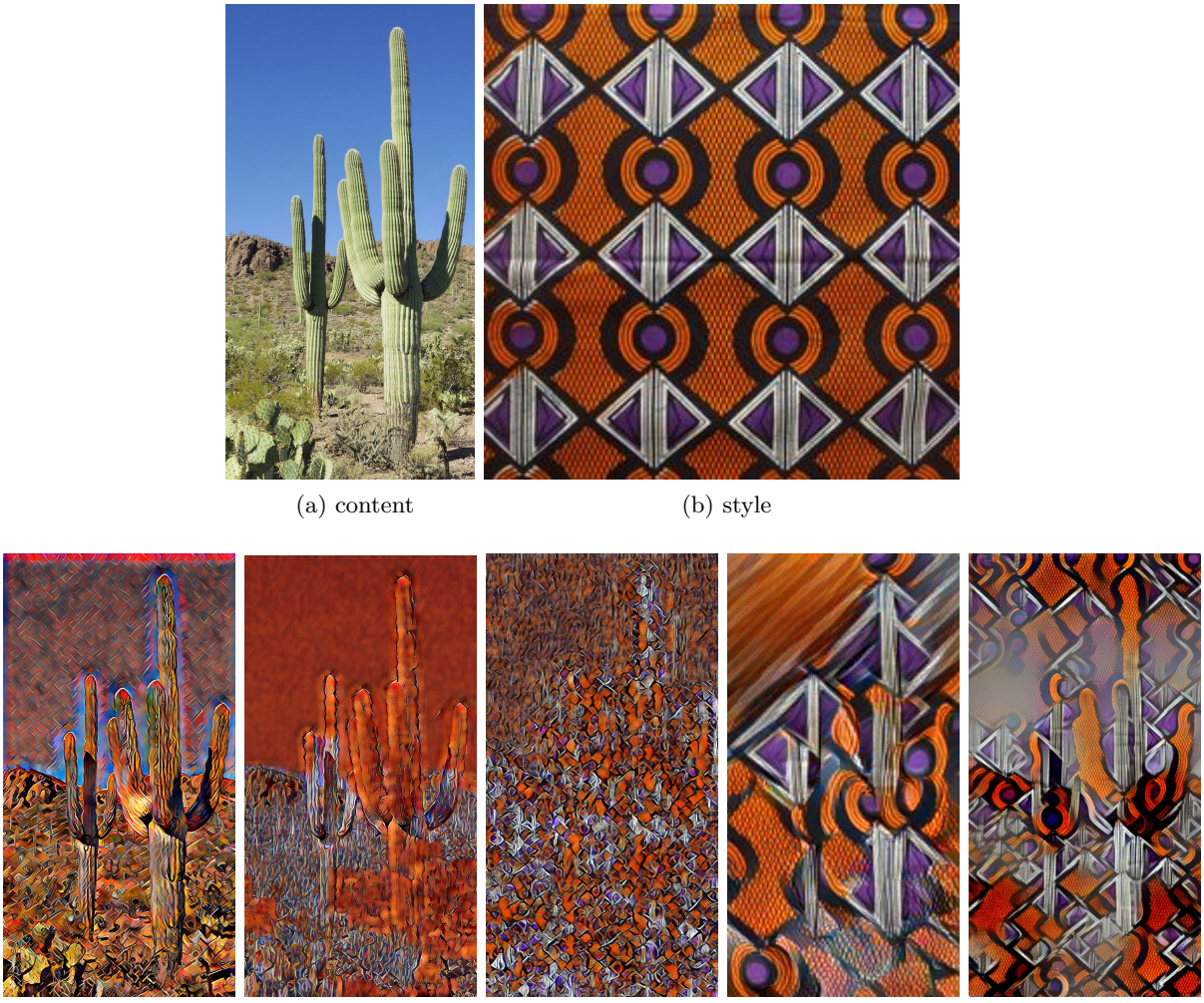
To have a more precise characterization of these algorithms, we introduce a new evaluation methodology based on the *predictivity* of neural style transfer algorithms and gather a set of paired paintings and photographs for this evaluation. The *predictivity* consists of assessing whether or not the algorithms' outputs are close to an existing painting when using this painting as the style image and the associated photograph as the content image. This is also a crucial point in the computational creativity perspective. Some outputs are deemed interesting while bearing not much resemblance with the initial painting, *i.e.*, with what the painter did.

Besides, when showing artists some outputs of style transfer algorithms using their paintings as style images, they often do not observe their practice. However, they sometimes identify inspiring aspects in the various outputs of different algorithms, implicitly acknowledging their computational creativity. This naturally led us to painting processes with artists, who could not

only edit groups of style transfer outputs but use them as basic elements to widen their style. This constructively interlaces the agency attribution of the algorithms part in the creative process.

We further encouraged this complexity by exploring these algorithms in the real world, where the outputs are projected onto a real canvas, the classical space for human painters. Human and machine contributions are then mingled in a single canvas. Interestingly, to help the observer that seeks to untangle each agent's contribution, the canvas can be shown together with the various computational suggestions. The algorithms were experienced as *computational catalysts* to human creativity in such creative processes, a middle ground between creative agents and technical tools.

We first describe the new methodology for the qualitative evaluation of different style transfer algorithms. We indicate that some approaches to neural style transfer do not satisfy a basic property, which leads to an instability behavior that ultimately allows reinforcing the diversity of style transfer outputs. This study helps us understanding and qualifying better what style transfer algorithms achieve. We then present various interactive painting experiments between human and style transfer outputs. This leads us to the notion of *computational catalyst* that helps characterize the algorithms' contribution in our specific settings.



(a) content (b) style

(c) AdaIn (d) MST (e) WCT (f) STROTSS (g) Gatys

Figure 6.7: Outputs of neural style transfer algorithms AdaIn [Huang and Belongie, 2017], MST [Zhang et al., 2019b], WCT [Li et al., 2017], STROTSS [Kolkin et al., 2019] and Gatys [Gatys et al., 2015] on the same {content, style} pair.

6.2.2 Evaluating Neural Style Transfer Methods

Neural methods for style transfer originated with the optimization-based technique of Gatys et al. [2015], which leverages the image features extracted from convolutional neural networks (CNN). To speed up the process as well as to have access to representations of a particular painting style, [Li and Wand, 2016, Ulyanov, 2016, Johnson et al., 2016] then proposed to train a neural network dedicated to a particular style, enabling to do neural style transfer of an image with a single forward pass instead of a full optimization procedure. Later on, universal neural style transfer methods were developed to transfer any style to a content image, again with a single forward pass [Ghiasi et al., 2017, Li et al., 2017, Huang and Belongie, 2017]. These approaches are much faster than the optimization-based approaches, but they suffer from the

well-documented instabilities of neural networks [Szegedy et al., 2013]. We show that a specific instability that, to the best of our knowledge, has not been pointed out yet, can notably be beneficial as it enlarges the creative possibilities of neural style transfer.

Alternatively, to explore other creative opportunities of such algorithms, several user control methods have been developed using, for example, semantic correspondences [Lu et al., 2017, Gatys et al., 2017, Kolkin et al., 2019], allowing to hand-tune the color histograms [Gatys et al., 2017] or the scale of the patterns [Risser et al., 2017]. It is also possible to transfer multiple styles [Mroueh, 2019, Cheng et al., 2019] at once.

Many works are still exploring different neural style transfer approaches, for instance, working with histogram losses [Risser et al., 2017], using various relaxation of optimal transport [Kolkin et al., 2019, Mroueh, 2019, Kotovenko et al., 2019] or trying to match semantic patterns in content and style images [Zhang et al., 2019a]. All these methods achieve impressive plastic results, but it is hard to characterize one w.r.t. the other. They may not yet actually stylize an image in the many ways a human would. We thus study the question of evaluation methods for style transfer.

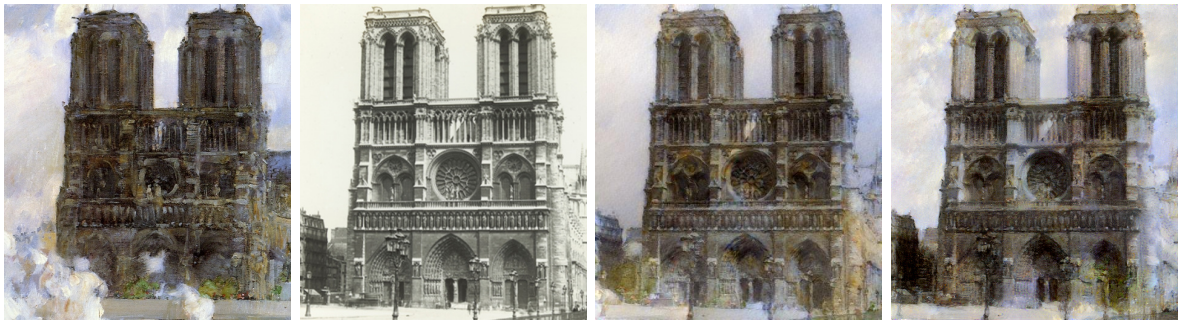
Qualitative evaluation A natural way of evaluating a neural style transfer method is to measure the content preservation and the stylization quality of the outputs. The variety of possible input images for content and style makes this task difficult in general. For example, Gatys successfully transferred Van Gogh's *Starry Night* style, but the examples shown in figures 6.7 and 6.8show notable artifacts. Such an evaluation can still be done by gathering a large number of responses as Kolkin et al. [2019] did for measuring the content or style preservation of their method compared to the others. Results showed that their method (STROTSS) offered, on average, the best trade-off between content and style preservation but do not say in what sense the style and content are better preserved.

To have a systematic and more refined comparison, we propose to study the *predictivity* of style transfer algorithm: does an algorithm stylize the image in a way similar to what a painter would have done? Precisely, when considering a photograph as a content image and a figurative painting of this image as a style image, one can compare the output of the neural style transfer algorithm with the figurative painting. One can further judge whether the style transfer technique succeeds in predicting the painting. If not, one can try to characterize how it differs from it.

Such pairs of photographs and content-preserving paintings are not readily available. Landscapes are constantly changing. Face portraits are rarely faithful to the original, and we rarely possess the model's photograph. Building paintings, however, is a good class of paintings for the proposed study. We thus construct a set of photographic-painting pairs[2], see Figures 6.8-6.9 for instance, focusing on the Series *Notre Dame de Rouen Cathedral* by Claude Monet. It consists of about forty paintings capturing the facade of *Notre Dame de Rouen Cathedral* from nearly the same viewpoint at different times of the day and year and under different meteorological and lighting conditions [Kleiner, 2009, p. 656].

With this set, qualitative evaluation can be done more systematically and less arbitrarily;

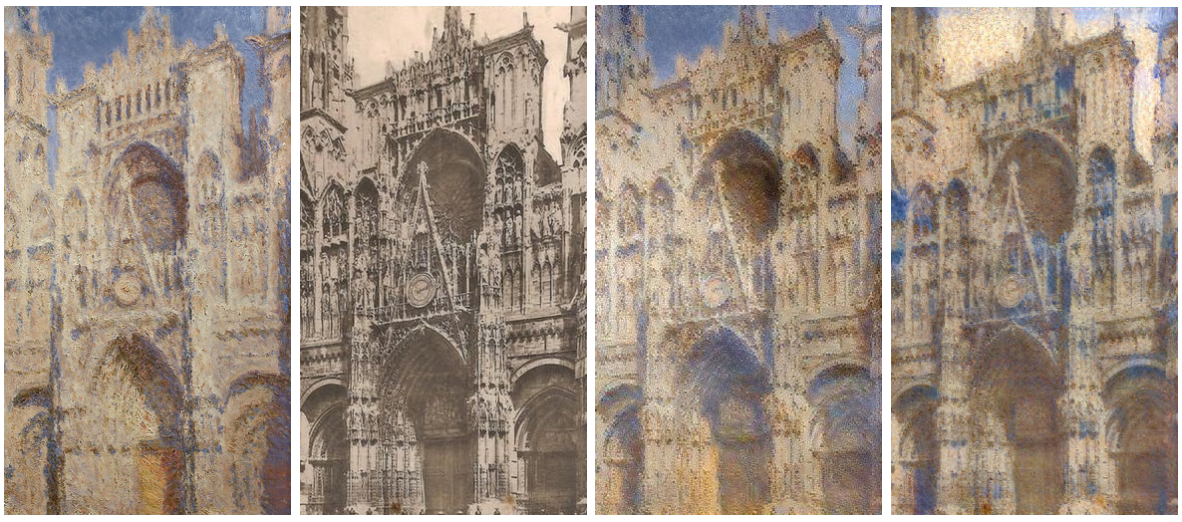
[2]https://www.di.ens.fr/louis.thiry/Monet.zip



(a) Hassam painting (b) photograph (c) STROTSS (d) Gatys

Figure 6.8: Detail of *Quai St Michel*, Childe Hassam, corresponding photo and style transfer outputs.

in the example shown in Figure 6.9, STROTSS output is qualitatively the closest to the Monet painting, especially for the lightening effect on the door and the left of the portal. Gatys and WCT suffer from spatial inconsistency as the blue sky is replaced by a sunlight halo in the first one, and the background is hardly distinguishable in the second one. We release this set and the outputs of the style transfer algorithms to facilitate and systematize the qualitative evaluation of neural style transfer techniques.



(a) Monet painting (b) photograph (c) STROTSS (d) Gatys

Figure 6.9: Detail of *Le Portail de la cathédrale de Rouen au soleil*, Monet, corresponding photograph and style transfer outputs

6.2.3 Quantitative evaluation

Numerical evaluation methods have the benefit of being more systematic and objective. However, we point here that most neural style transfer evaluation methods are specific to certain algorithms and are not always relevant for the output's stylization quality.

In computer vision, perceptual losses are becoming the standard to compare the visual similarity between images [Zhang et al., 2018]. These methods based on CNN features comparison offer state-of-the-art performance on image similarity judgment datasets using different CNN architectures. Since neural style transfer originally consists of optimizing an image to match the CNN features of another style image, the perceptual loss between the outputs and the target style image might be artificially small despite notable perceptual differences.

Other numerical evaluation techniques were proposed; Sanakoyeu et al. [2018] test whether a pre-trained neural network for artist classification on real paintings succeeds in classifying the artist of the style image based on an algorithm's output. Jing et al. [2017] consider comparing saliency maps between images since the saliency maps' spatial integrity and coherence should remain similar after style transfer. Moreover, as neural style transfer relies on a certain quantification of the style based on CNN features, we Jing et al. [2017] propose to evaluate how much the optimization objective is achieved in style transfer. In the following case, we show that improving the optimization objective is not necessarily related to the output's visual quality.

Optimization-based neural style transfer methods consist of optimizing the pixels of an image I to minimize a loss l. This loss l is usually the sum of a content loss $l_c(I, I_c)$ measuring the content similarity between I and the content image I_c and a style loss $l_s(I, I_s)$ measuring the style similarity between I and the style image I_s. In the STROTSS method, Kolkin et al. [2019] define the style loss as the Earth Movers Distance (EMD) between CNN features of the image I and the style image I_s. Given the CNN features $\Phi(I), \Phi(I_s)$ of the images I, I_s, we compute the distance matrix C^{I,I_s} between $\Phi(I)$ and $\Phi(I_s)$ and the EMD is defined as the solution of the following optimization problem

$$\mathrm{EMD}(I, I_s) = \min_{\substack{T \geq 0 \\ \sum_j T_{ij} = 1/m \\ \sum_i T_{ij} = 1/n}} \sum_{ij} T_{ij} C^{I,I_s}_{ij} \ .$$

Exact EMD computations are too expensive for neural style transfer applications, and a relaxed EMD (REMD) is used in STROTSS. It consists of taking the maximum of two simple lower bounds of the EMD, each obtained removing one of the two linear constraints sets $\sum_{i \text{ or } j} T_{ij}$ applied on the transport plan T

$$\mathrm{REMD}(I, I_s) = \max \left(\begin{array}{c} \min\limits_{T \geq 0,\ \sum_{\mathbf{i}} T_{ij} = 1/n} \sum\limits_{ij} T_{ij} C^{I,I_s}_{ij} \ , \\ \min\limits_{T \geq 0,\ \sum_{\mathbf{j}} T_{ij} = 1/m} \sum\limits_{ij} T_{ij} C^{I,I_s}_{ij} \end{array} \right) \ .$$

Despite the use of this loose relaxation, the human evaluation done via Amazon Mechanical Turk (AMT) indicates that STROTSS statistically offers the best style/content trade-off

compared to the other neural style transfer techniques [Kolkin et al., 2019, §4] in the opinion of the AMT workers. Experiments done with artists confirmed this trend as the artists were mostly impressed by results produced by STROTSS.

The authors mentioned that a better approximation might yield better style transfer results. Sinkhorn-distance [Cuturi, 2013] in its log-domain stabilized version [Schmitzer, 2019] is a good candidate for this purpose. We thus replaced the relaxed earth movers distance REMD by the Sinkhorn earth movers distance

$$\text{SEMD}_\epsilon(I, I_s) = \min_{\substack{T \geq 0 \\ \sum_j T_{ij} = 1/m \\ \sum_i T_{ij} = 1/n}} \sum_{ij} T_{ij} C_{ij}^{I,I_s} - \epsilon h(T)$$

where h is the entropy of the transport plan T

$$h(T) = -\sum_{ij} T_{ij} \log T_{ij}$$

and ϵ is the entropic regularization parameter. The corresponding optimization problem is convex and is solved iteratively with a fixed number of iterations N. SEMD_ϵ is an upper-bound of the EMD, and it converges to the exact EMD as ϵ goes to 0. We release a Pytorch [Paszke et al., 2019] implementation[3] of STROTSS, including the SEMD.

Figure 6.10: Left to right, content, style and style transfer outputs using REMD, and SEMD $\epsilon = 1e^{-3}, N = 50$.

Figure 6.10 shows a comparison of experimental results, suggesting that getting much closer to the mathematical quantification of the style does not necessarily lead to more relevant results, and numerical evaluation of how much the mathematical objective is achieved is not essential from a visual perspective.

In the same idea, the instability phenomena commonly assumed to be detrimental in the neural networks literature (e.g., adversarial examples) can qualitatively increase the creative possibilities of neural style transfer.

[3] https://github.com/human-aimachine-art/pytorch-STROTSS-improved

6.2.4 Instability phenomena

Neural style transfer instabilities have been pointed out by Risser et al. [2017] and Gupta et al. [2017] in the case of real-time style transfer for videos. The aim is to identify and remove the time-inconsistent artifacts that create unpleasing effects. Here we outline instabilities stemming from another type of inconsistency and propose to take advantage of them.

A style transfer method is simply a function f that takes as input a style image s and a content image c and outputs a stylized version $f(s, c)$ of c with s. It is reasonable when giving such a method the same image as content and style to expect the image itself, *i.e.*, that f satisfies $f(s, s) \approx s$. Let us now consider the following recursion

$$x_{t+1} = f(x_t, x_t) \ , \tag{6.1}$$

where x_0 is an initial image. Optimization based methods empirically converge to an equilibrium where $f(x, x) = x$ independently of the initialization. On the opposite, feed-forward approaches to style transfer [Ulyanov, 2016, Johnson et al., 2016, Li et al., 2017, Huang and Belongie, 2017] lead to oscillating sequences (x_t) around non-trivial (*i.e.*, not a monochrome image) forms, yet typically bearing absolutely no resemblance with the initial image x_0. Since the pixel values are clamped between 0 and 1, colors end up being either saturated or zero, but not uniformly and still revealing specific patterns in Figure,6.11. Interestingly, when starting from uniform color images x_0, for some f, the sequence would still show the same type of instability in the long-run. This phenomenon happens on this video[4], for instance.

From the perspective of computational creativity, this apparent failure is interesting. In the first iterations, we observe that some methods produce a series of images progressively stylized. Given a style transfer function f, the same effect happens across all sequences we experimented with. For instance, in Figure 6.13 we see a distinct tessellation effect in the images on the first row. We use this technique to produce more diverse and computationally creative style transfer outputs in the interactive painting experiments. See image (f) in Figure 6.16 for instance.

Alternatively, the asymptotical regime of the sequences (x_t) produces surprising animations. The appearing patterns are completely different from one approach to another but are experimentally the same for different initialization images and a given method. Sequences are shown in Figure 6.12, refer to this video[5] or this one[6] for a more lively visualisation.

6.2.5 Interactive Portrait Painting Experiments

In the previous sections, we have questioned the relevance of neural style transfer evaluation. To go beyond comparing techniques, we propose to take advantage of the diversity of the outputs and to use them as a source of inspiration for artists.

Some painters have recently explored interactive processes with machines in the real world, particularly in painting. For instance, Chung [2015] among others, leveraged on the algorithms

[4] https://youtu.be/WCJNLWb-H2M
[5] https://youtu.be/gAq1lvb1G1c
[6] https://youtu.be/s87R-9JITvE

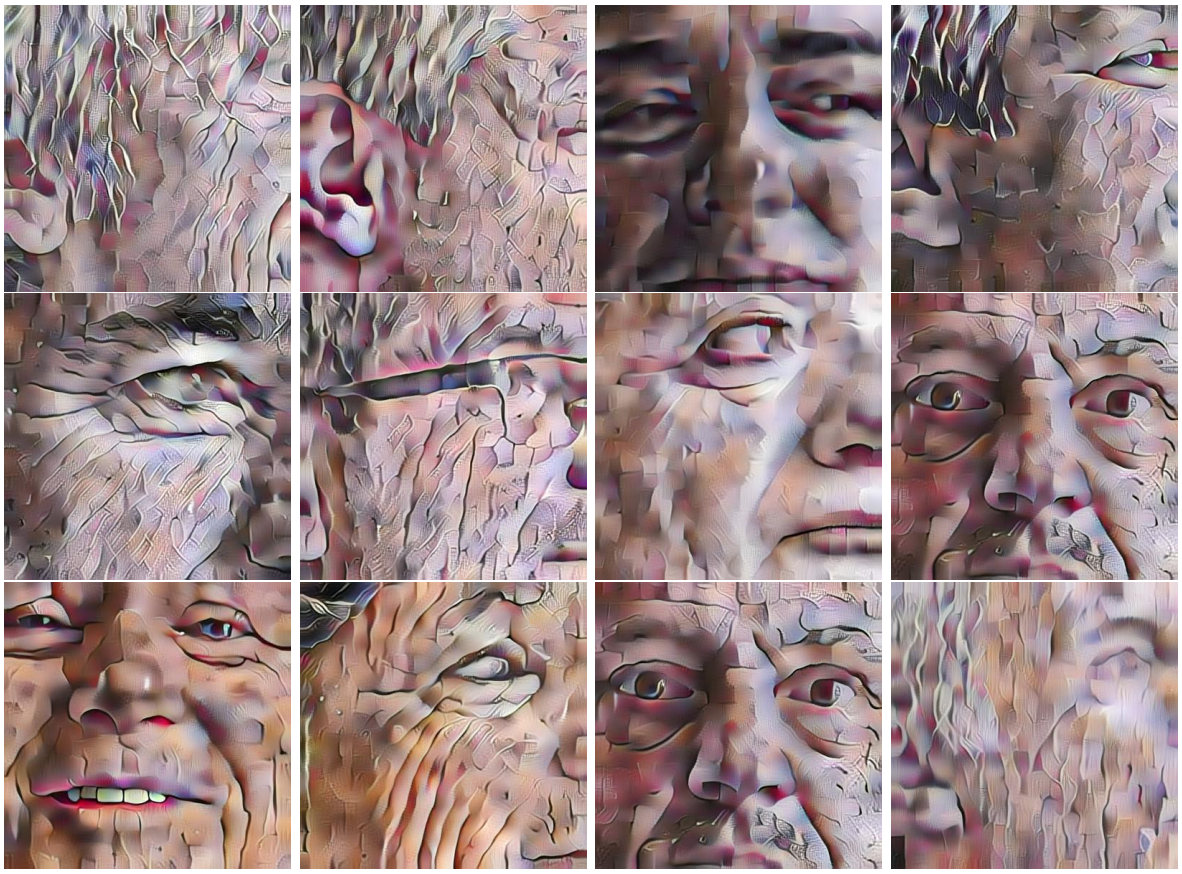


Figure 6.11: Showing the fourth iterates of sequence (x_t) as defined in Equation (6.1) for MST method on various fragments of portraits. The tessellation effect is maintained across all the images.

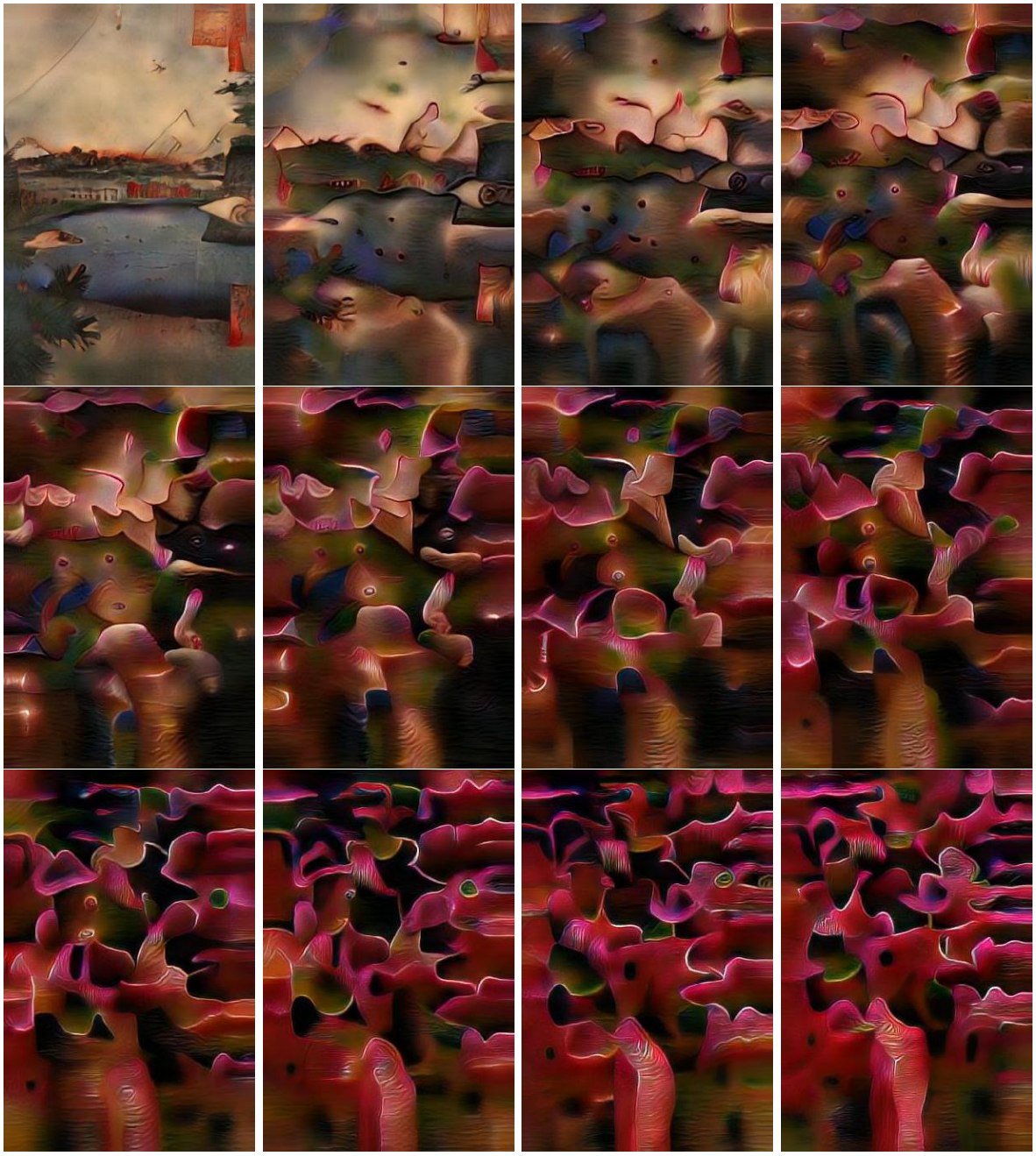


Figure 6.12: We observe the asymptotic regime of Equation (6.1) as the images represents the first $(x_{4k})_{k=0,\dots,11}$ iterations of (6.1) for a given initial image.

from artificial intelligence to paint interactively with humans in the real world, where a machine would act on the real canvas via a robotic arm. **?** also explore such an interaction, where the machine does not act but suggests via projection. However, none of these use style transfer algorithms outputs to paint interactively with an artist on the canvas.

We explore that possibility through various series of interactively painted portraits. We

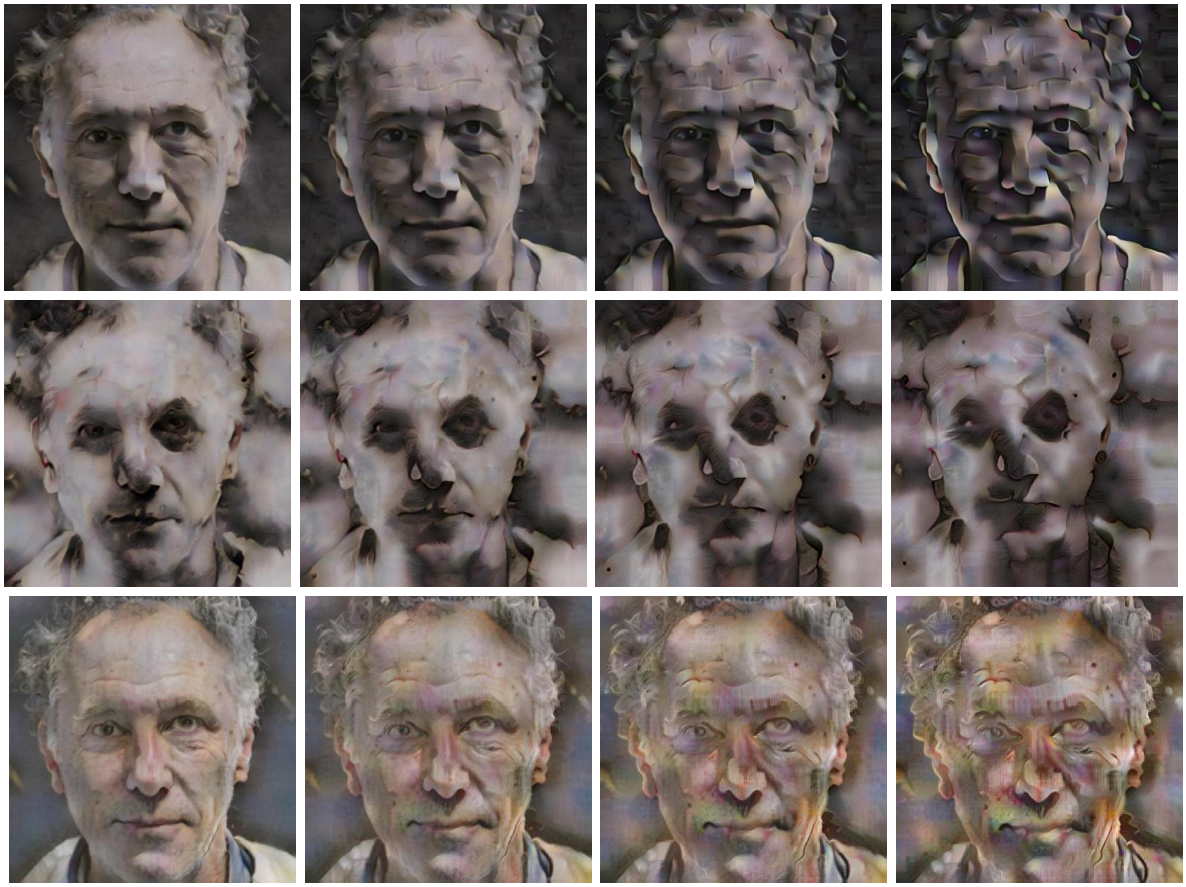


Figure 6.13: Each row represents the first fourth iterate of sequence defined in Equation (6.1) with a different neural style transfer approach. The first row corresponds to MST [Zhang et al., 2019b], the second to WCT [Li et al., 2017] and the third to AdaIn [Huang and Belongie, 2017]. The first method has a clear tessellation effect, the second has a blurring effect, and the third gives an increasingly clownish style to the image. These effects appear to be the same across images, see Figure 6.11.

describe various interactive painting experiments inserting outputs of neural style transfer algorithms during the human painting process. In all cases, the algorithms' creations are projected onto the canvas but never automatically painted, for instance, via a robotic arm or a printer. We first describe the experiments on canvas and then motivate the underlying design choices. We finally show how the notion of *computational catalyst* naturally emerges. Note also that all the paintings revolve around portrait themes.

Editing multiple styles in one portrait. Neural style transfer outputs are diverse from one method to another, as outlined in Figure 6.7 for instance. To edit these outputs' creative content into a single final artwork, we select a person's photography as a content image, and we transfer the style of previous artists' painting into this content image using various algorithms. We then show the stylized images to the artist, but not the original content image, which he

never sees. He then selects some of the outputs that best resonate with his practice. These style transfer outputs are finally alternatively projected on a canvas for a certain amount of time. Figure 6.14is an example of canvas realized according to this process; we complement the final canvas with the various style transfer outputs selected by the artist. We also explore other variations of that idea, for instance, via collage, where the selected outputs are mingled together into a single image projected afterward.

Figure 6.14: Left: two panels of 100×50 cm oil canvas. Right: the style transfer outputs that formed the painter theme and were successively projected on the canvas. For each, the three canvas in Figure 6.17 were used as style images; the content was a photographic portrait.

Pixelizing portrait construction. This creative process's motivation is to artificially create an interactive loop between the painter and the algorithm. The initial image is projected onto or next to the canvas, which is divided into squares. The painter is then asked to paint sequentially on each square of the canvas. Whenever a square is completed, we use it as a new style image to stylize the initial photograph. The stylized image is then projected on the canvas. Images (a)-(d) in Figure 6.15 are some of these projected outputs. The painter only sees an interpretation of the photographic portrait by a style transfer algorithm taking the painter's style in the previous painting. We show one example of a canvas produced in such a way in Figure 6.15.

Note that the style transfer output should be the machine's prediction of what the artist would do, provided that the previous square contains all the painter's style information and that the style transfer method is ideal. This remark was then the basis for a gamification exploration of the painting process, where the artist was asked to attempt to fail the machine prediction as much as possible.

This decomposition of the painting process produces painting artworks that are sequential objects. All of the successive iterations of the painting are of interest and not only the final

canvas. Actually, algorithms in image computational creativity are much less performative at generating images as sequences of brushstrokes than generating images all at once, like with GANs, [Goodfellow et al., 2014] for instance. This is simply because paintings are usually not sequential objects. Indeed we very rarely observe all the steps leading to a painting, apart from large quantities and categories of simple sketches [Eitz et al., 2012, Jongejan et al., 2018]. Alternatively, computationally inferring the steps of a painting from the final canvas [Xie et al., 2013, Ganin et al., 2018, Nakano, 2019] is not yet very successful. This arguably explained why in painting art, compared to other domains such as music, whose artworks are sequential by nature, the computationally creative algorithms are harder to frame in a fully interactive way with humans, hence limiting the painter's ability to interact with machines truly.

Interactive series of portraits. We then considered using neural style transfer outputs for series of portraits. We select a photographic portrait, and we stylize it with a neural style transfer algorithm using a previous artists' painting as a style image. We project the stylized image as inspiration for the painter. When the painter has finished the painting, we stylize the photographic portrait again using the painting that has just been painted. We project the new stylized image as the next inspiration, and we repeat the process, typically two or three times. Figure 6.16-6.17 present two series of canvas in chronological order. In figure 6.16 all canvas stem from the same photographic portrait, while in Figure 6.17 we alternate between two photographic portraits to avoid specialization of the artist to particular content.

We played on the many ways to generate new style images at each iteration. Importantly, at each iteration, the previous image serves as a style image. This allows the artist to interact with a computational version of his past work, a key aspect computational creativity offers.

Also, in Figure 6.16, the photographic portrait is an input in the first canvas. The subsequent machine only uses as content or style images the preceding paintings. The observed divergence is hence an intertwined responsibility between the painter and the algorithms.

6.2.6 Computational catalyst in the interaction

Neural style transfer algorithms are computationally creative because they may produce new images with an aesthetic that can significantly differ from what a painter would do. To turn this creativity into artwork, we have specified various painting experiments on a real canvas between a painter and outputs from these algorithms. We now report how these attempts shed light on a few aspects of the computational creativity of neural style transfer algorithms and cast them, in this specific setting, as *computational catalysts* to human creativity. The interactive painting process itself was designed to embody some questions related to computational creativity and human-machine interplay, which has arguably become a major societal theme.

Computational Creativity and Catalyst. The initial motivation for designing human-neural-style-transfer interactive experiments was to create a single object out of many different style transfer outputs, focusing on a painting instead of a printed version of the numerical output. This echoes other creative works with machines where some artists playfully worded themselves as *editors* of the machine creativity, see, for instance, the rationale surrounding the

last A.I. assisted musical album *Chain Tripper* of the band YACHT [2019]. During our painting experiments, the intertwining between the machine's outputs and painter interpretation was non-trivial since the painter was altering the machine suggestions. The painter felt the outputs gave new style directions, wording them as *computational catalysts* to his own creativity.

In these interactions with algorithms, we exploit the ability of style transfer methods to produce outputs based on the painter's previous works. This is a simple yet powerful idea that allows an artist to interact with computationally influenced versions of its own (past) work. The painter felt this was a semi-extraneous interpretation of his past techniques, allowing him to rediscover some elements of his old practice in a surprising way. Besides our specific framework, this seems to be another major benefit and specificity of computational creativity.

Importantly, in these portrait paintings, the artist could not see the real photographic portrait except in the beginning. We purposely designed it so that the painting practice could embody the fact of perceiving the world only through the machines' lens. This has critical societal echoes; for instance, the issues raised by the so-called *fake news* stems to some extent from generative algorithms capacities, a technical point of view. From a societal point of view, it comes from our increasingly resorting to numerical pieces of information as a way to perceive the world. Hence, we implicitly explore what a painter felt when relying only on machine outputs to see the portraits.

Alternatively, it also gives another perspective on computationally creative algorithms, offering new inspirational spaces to portray. Indeed we may not only explore algorithms outputs through printed versions, pretty much as we do not capture nature only through photography. Computationally creative outputs may hence be thought of as new types of landscapes for painters to capture.

Note also that the *transient* essence of these *computational landscapes* has very different rules than that of Nature. The painting could remain the only imprint of the machine outputs by erasing the content files or algorithms' outputs. This again is a specificity of computational creativity, when framed as a theme creator for artists, that is worth exploring.

Designing Human-Machine painting processes. A major aspect of these human-machine interactive processes is that we engineered the numerical outputs out in the real world, rather than having a painter who interacts with machines on a tablet.

When the painting process materializes in a numerical tablet, it strongly constrains the painter's sensations; he does not feel the brushstrokes' gesture, the canvas is not perceived in the full-dimensional space, etc. Even with interactive experiments on a real-canvas, the painter felt some processes as being too intrusive or constraining, like the experiment reported in Figure,6.16 which forces the artist to follow unusual rules for creating. This highlights that computational creativity, when considered in such a human-machine interplay, is notably conditioned by the current state in the engineering of such interactive systems. For instance, how much projection is less intrusive than a robotic arm?

This level of machine's *intrusion* is inherently linked with how the computational creativity of the algorithms is perceived, notably concerning the creative agency that is attributed to the machine outputs. Part of the discussion around computational creativity may hence be tightly

related to some artists' feelings of losing a share of the creative agency when algorithms become more than a disposable tool.

So it appears that when engineering such systems, there are typically two directions in the interfacing, either the machine goes out of the numerical world, or reversely, the human interacts with the machine in the numerical world. And as we described previously, the interface puts the artists in very different situations. However, this may not only be considered as a limitation as each constraint forces the painter to embody what we may feel in our daily interaction with machines. In particular, each type of interfacing echoes, and may advocate then, a different societal relation humans have with machines; in the era of machines, it is primal to explore many such experiments.

Plastic point of view. These interactive painting experiments were also designed to explore pictorial aspects.

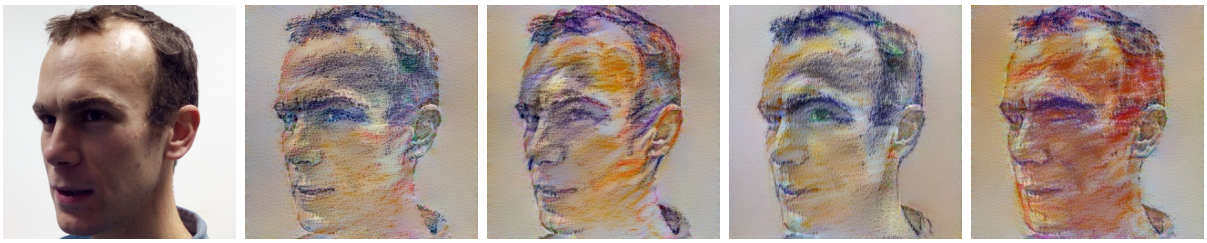
For instance, the photographic portrait that initiates the series in Figure 6.17-6.16 were in black and white. However, the style transfer algorithm and the painter were not constrained in the grey-scale space. The painter could observe in projected outputs of the machine, or conversely initiate in the real canvas, the emergence of the colors. For instance, in Figure 6.17, red appears on the eyebrows, while in Figure 6.16 the colors are intended as variations of shade, which only exists through the machine.

While this is interesting from the creation point of view, it is also from the observer concerned about agency attribution. For a given aspect of the painting, like colors, did the painter simply repeat the machine colorization outputs, re-interpreted it, or even started it? This reinforces the importance, in an exhibition, of algorithms' outputs as testimonies of the final artworks.

6.3 Discussion

The development of these two interactive settings reveals many potential benefits of leveraging computational creativity in this interactive framework. The use of algorithms such as Sketch-RNN and Neural Style Transfer in a real-world interaction also helps to understand what these algorithms achieve. Rather than being creative on its own, the algorithm acts as a computational catalyst to human creativity, offering new sources of inspiration to the artists.

(left) final canvas (right) steps 1, 3, 5, 7.



Original photographic and projections after steps 1, 3, 5, 7.

Figure 6.15: A 50×50 cm oil pastel canvas inspired by an evolving projection. The canvas was divided into 9 squares, and the painter had painted sequentially on each square. After each square was completed, the output of the style transfer algorithm using the original photograph as content and the current canvas as a style image was projected.

(a) First canvas

(b) Second canvas

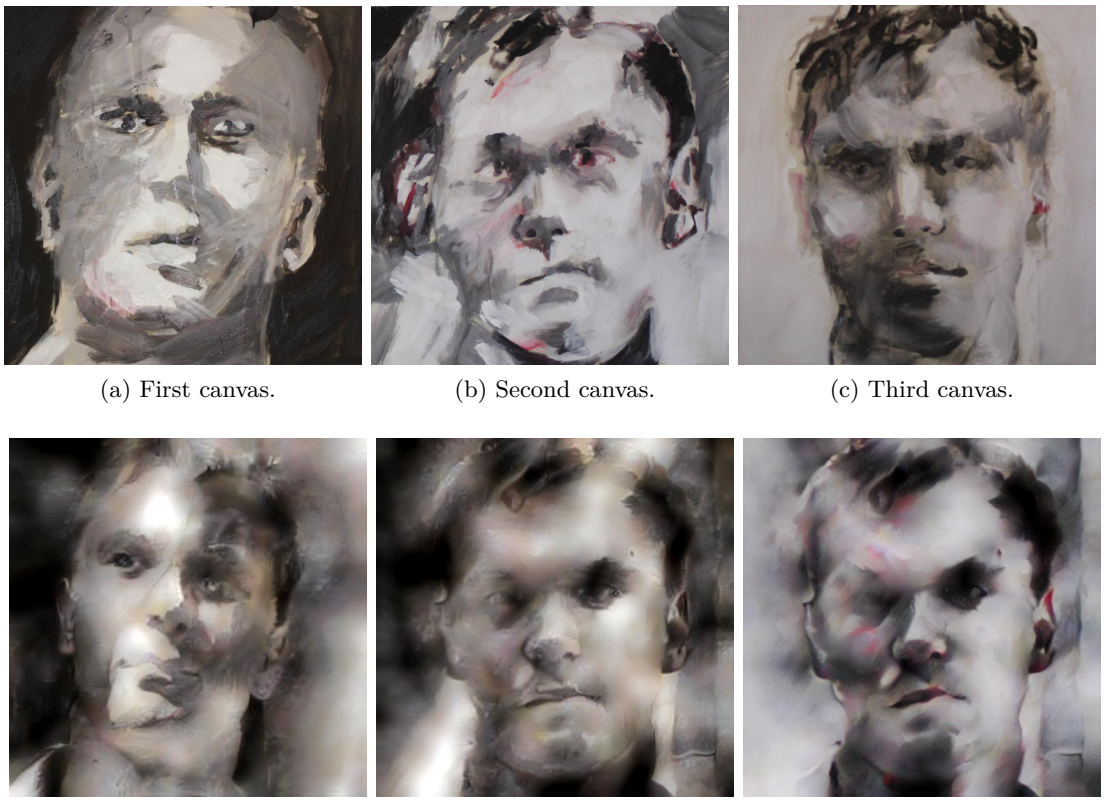
(c) Third canvas

(d) Projection 1

(e) Projection 2

(f) Projection 3

Figure 6.16: Three 50×50 cm oil on canvas of the same face. Iteratively showing style re-interpretation to the painter. Image (a) served as a style image to produce (e); Image (b) served as a style image to produce (f). Note that outputs (f) is the third iterate of Equation (6.1) with the MST style transfer algorithm to produce a slight tessellation effect.



(a) First canvas. (b) Second canvas. (c) Third canvas.

(d) NST output 1 (e) NST output 2 (f) NST output 3

Figure 6.17: The top row collects pictures of the paintings, 50×50 cm oil on canvas. The bottom row gathers the outputs of the neural style transfer (NST) methods which were chosen by the painter as his basis theme. The first is the stylization of the photographic image with a previous painting of the painter. The second one is the stylization of a different photographic image with the first painting as a style image. The third is the stylization of the same previous photographic image with the second painting as a style image. Note that the order of the painting is chronological.

Chapter 7

Conclusion

This dissertation studied local methods for two high-dimensional supervised learning problems: image classification and energy regression in physics. We observed their surprising efficiency despite the multiscale nature of these two learning problems.

7.1 Summary of findings

In the field of regression in physics, we have studied the Solid Harmonic Scattering Transform [Eickenberg et al., 2017, 2018]. This atomic system descriptor differs from other descriptors of the literature. It is inspired by image classification techniques and relies on scale separation.

To study the relevance of scale separation, we have compared this descriptor with descriptors that rely on three different tasks. The first one is the regression of the atomization energy of small organic molecules. The second one is the regression of the energy of Graphite periodic cells. The third one is the regression of the vibrational entropy in Iron periodic cells. On these three problems, methods relying on spatial separation offered similar or better predictive powers than the Solid Harmonic Scattering Transform.

This finding is not intuitive as energy is a complex property of the atomic state. Energy depends on various interactions in the system. These interactions occur at different scales, e.g., ionic and covalent bonds at short range, Van-der-Waals interactions at the mesoscale, and long-range Coulomb interactions. Despite this, local methods capture efficiently even long-range interactions. This might be due to a weak correlation. For example, if typical local patterns result from a certain type of interaction, the regression technique can predict the correct long-range energy from this local information.

Following the natural analogy between image patches and atomic neighborhood, we have studied image classification from the perspective of patch separation. First, we have presented a structured CNN architecture based on Scattering Transform encoding of image patches followed by a supervised local encoding. It outperforms previous attempts by a large margin [Oyallon et al., 2017]. Performances are comparable with the BagNet [Brendel and Bethge, 2019] despite a much shallower encoding. One can explain the classification decision in terms of patch evidence.

Finally, we present a non-learned visual representation based on K-nearest neighbors encoding of image patches using a Euclidean distance. The non-trivial accuracy that we obtain with a linear classifier demonstrates that invariance is not a necessary aspect for predefined visual representation. This method shares a crucial ingredient with high-performing convolutional kernel methods [Li et al., 2019, Recht et al., 2019, Shankar et al., 2020, Mairal, 2016], namely a whitening procedure. It suggests that this ingredient is a key to the performance of these methods. Moreover, the finite-dimensional aspect of our convolutional kernel allows scaling experiments to large datasets like ImageNet. Our method shall serve as a baseline for future investigations of convolutional kernel methods on ImageNet. Finally, we have demonstrated that in the context of classification, *"brain-dead lookup, a.k.a. nearest-neighbors, works surprisingly well"* to quote Alexei Efros[1].

We have revealed the importance of short-range/local interaction in the two studied problems. These short-range/local interactions are the leading first-order term in both energy regression and image classification.

At the end of this dissertation, we studied the impact of algorithms in the field of artistic creation. For this purpose, we presented two interactive systems for human-machine co-creation based on two different deep learning algorithms. In addition to the produced artworks, the experiments and discussion with artists allowed characterizing and refining our perception of creativity from the human-machine interaction perspective.

7.2 Future perspectives

Concerning the relevance of scale separation for energy regression in physics, we do not draw a general conclusion from the three examples we have studied. We wait for physicists and chemists to publish energy databases where the local methods do not perform well.

Given how-well the free-energy regression technique we have presented [Lapointe et al., 2020] works, we consider using it to explore free energy landscapes using, for example, the genetic algorithms proposed by Kaczmarowski et al. [2015].

The major drawback of these energy regression techniques is the lack of generality of the learned functions. A regression model trained on small organic molecules performs terribly badly on Carbon solids. On the opposite, electronic structure methods like Density Functional Theory yield reasonably good results for both molecules and solids apart from specific failure cases [Cohen et al., 2008, Cohen and Mori-Sánchez, 2016]. Supervised learning offers a great opportunity to correct these failures and improve electronic structure methods. And multi-scale invariants might be appropriate descriptors to go beyond local approximations, such as local density approximation [Perdew and Zunger, 1981, Ceperley and Alder, 1980]. Regressing a density functional is a future research direction we consider looking at.

Very recently, Dosovitskiy et al. [2021] adapted the natural language processing "Transformer" architecture [Vaswani et al., 2017] to image classification. They essentially replaced words with image patches. They obtain state-of-the-art results (88.55 % **top-1** accuracy on

[1] See for example these slides or this talk.

ImageNet) when pretraining on very large private datasets. The fact that the same architecture gives state-of-the-art results for natural language processing and image classification lets us believe that there are structural similarities between text and images, between words and image patches. Simple word vector representation like Word2vec [Mikolov et al., 2013] exhibit low dimensional properties related to meaning similarity. In the Word2Vec vector space, the nearest neighbor of the combination $\overrightarrow{\text{King}} - \overrightarrow{\text{Man}} + \overrightarrow{\text{Woman}}$ is $\overrightarrow{\text{Queen}}$[2]. The results obtained with our K-nearest-neighbors-based classifier show that the image patches also have low-dimensional nearest neighbor properties, similarly to vector word representations. Trying to characterize more precisely this low-dimensional structure is one of the next research directions.

The works produced with the artists materialize the paste of the algorithm in a creative interaction. We see them as universal supports to think about the importance of these new technologies in our lives. We have recently worked on creating diptychs that are reinterpretations of an original image according to machine's and human's attentions [Cabannes, Kerdreux, and Thiry, 2020]. We used them to discuss some crucial issues of current task-oriented artificial intelligence and ambiguities on the notion of perception. We believe that using appropriate wording opens the possibility of understanding the impact of machine learning research in our societies. The overuse of anthropomorphic expressions like *artificial intelligence*, *neural network*, or *agent* to describe computer programs tends to confuse or even frighten people outside of the scientific community. We think that these expressions must be used precisely and sparingly. Describing an image classification pipeline as non-linear patch encoding is beneficial to the global understanding as it states the image classification problem from a formal point of view. Presenting an interactive creation system as a computational catalyst to human creativity allows understanding the role a machine can play in artistic creation. We hope the work presented in this dissertation is a step forward in a correct formulation and description of what machines achieve, from image classification algorithms to interactive artistic creation systems.

[2]We denote by $\overrightarrow{\text{Word}}$ the Word2vec representation of the word "Word".

Appendix A

Solid harmonic scattering transform

A.1 Solid harmonic wavelets normalization

The normalizing constant K_l is such that $||W_l||_1 = \int |W_l| = 1$ where W_l is defined by:

$$W_l(u) = \left(\sum_{m=-l}^{l} |\psi_{l,m}(u)|^2 \right)^{1/2}.$$

By definition of the wavelets $\psi_{l,m}$, we have:

$$W_l(u) = K_l \sqrt{\frac{2l+1}{2\pi}} e^{-r^2/2\sigma^2} r^l$$

. so :

$$\begin{aligned} ||W_l||_1 &= K_l \sqrt{\frac{2l+1}{2\pi}} \int_{r=0}^{\infty} \int_{\theta=0}^{2\pi} \int_{\phi=0}^{\pi} e^{-r^2/2\sigma^2} r^l r^2 sin(\phi) dr d\theta d\phi \\ &= K_l \frac{\sqrt{2\pi(2l+1)}}{2\pi} \int_{r=0}^{\infty} e^{-r^2/2} r^{l+2} \end{aligned}$$

Using the formulas below :

$$\begin{aligned} \int_0^{\infty} x^{2n} e^{-\frac{x^2}{a^2}} \, dx &= \sqrt{\pi} \frac{a^{2n+1}(2n-1)!!}{2^{n+1}} \\ \int_0^{\infty} x^{2n+1} e^{-\frac{x^2}{a^2}} \, dx &= \frac{n!}{2} a^{2n+2} \end{aligned}$$

(!! is double factorial), we have :

$$\int_{r=0}^{\infty} r^{l+2} e^{-r^2/2} = \begin{cases} \sqrt{\frac{\pi}{2}}(l+1)!! & \text{if } l \text{ is even.} \\ 2^{\frac{l+1}{2}} (\frac{l+1}{2})! & \text{if } l \text{ is odd.} \end{cases}$$

and thus:

$$K_l = \begin{cases} \frac{1}{\pi\sqrt{2l+1}(l+1)!!} & \text{if } l \text{ is even.} \\ \\ \frac{1}{2^{\frac{l+1}{2}}\sqrt{2\pi(2l+1)}(\frac{l+1}{2})!} & \text{if } l \text{ is odd.} \end{cases} \tag{A.1}$$

A.2 Fourier transform of solid harmonic wavelets

The plane wave $\mathbf{u} \mapsto e^{i\boldsymbol{\omega}\cdot\mathbf{u}}$ can be expanded over spherical harmonics in the following way :

$$e^{i\boldsymbol{\omega}\cdot\mathbf{u}} = 4\pi \sum_{l'=0}^{\infty} \sum_{m'=-l'}^{l'} i^{l'} j_{l'}(|\boldsymbol{\omega}||\mathbf{u}|) Y_{l'}^{m'}(\boldsymbol{\omega}) \overline{Y_{l'}^{m'}}(\mathbf{u})$$

where :

$$Y_l^m(r,\theta,\phi) = \sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}} P_l^m(cos(\theta)) e^{im\phi} \text{ in spherical coordinates}$$

$$j_l(r) = \sqrt{\frac{\pi}{2r}} J_{l+1/2}(r)$$

J_o being the Bessel function of first kind and of order o.

By definition, the Fourier transform of the solid harmonic wavelet $\psi_{l,m}$ is:

$$\begin{aligned}
&\hat{\psi_{l,m}}(\boldsymbol{\omega}) \\
&= \int_{\mathbb{R}^3} \psi_{l,m}(\mathbf{u}) e^{i\boldsymbol{\omega}\cdot\mathbf{u}} du \\
&= K_l \int_{\mathbb{R}^3} e^{-|\mathbf{u}|^2/2} |\mathbf{u}|^l Y_l^m(\mathbf{u}) e^{i\boldsymbol{\omega}\cdot\mathbf{u}} du \\
&= 4\pi K_l \sum_{l'=0}^{\infty} \sum_{m'=-l'}^{l'} i^{l'} Y_{l'}^{m'}(\boldsymbol{\omega}) \int_{\mathbb{R}^3} e^{-|\mathbf{u}|^2/2} |\mathbf{u}|^l j_{l'}(|\boldsymbol{\omega}||\mathbf{u}|) Y_l^m(\mathbf{u}) \overline{Y_{l'}^{m'}}(\mathbf{u}) du \\
&= 4\pi K_l \sum_{l'=0}^{\infty} \sum_{m'=-l'}^{l'} i^{l'} Y_{l'}^{m'}(\boldsymbol{\omega}) \left(\int_{r=0}^{\infty} e^{-r^2/2} r^l j_{l'}(|\boldsymbol{\omega}|r) dr \right) \left(\int_{\theta=0}^{2\pi} \int_{\phi=0}^{\pi} Y_l^m(\theta,\phi) \overline{Y_{l'}^{m'}}(\theta,\phi) d\theta d\phi \right)
\end{aligned}$$

The spherical harmonics are orthonormal functions of $\mathbb{S}^2$, i.e. for $(l_1, m_1) \neq (l_2, m_2)$:

$$\int_{\theta=0}^{2\pi} \int_{\phi=0}^{\pi} Y_{l_1}^{m_1}(\theta,\phi) \overline{Y_{l_2}^{m_2}}(\theta,\phi) d\theta d\phi = 0$$

and:

$$\int_{\theta=0}^{2\pi} \int_{\phi=0}^{\pi} Y_l^m(\theta,\phi) \overline{Y_l^m}(\theta,\phi) d\theta d\phi = 1$$

Thus, in spherical coordinates:

$$\hat{\psi_{l,m}}(\lambda,\alpha,\beta) = 4\pi K_l i^l Y_l^m(\alpha,\beta) \int_{r=0}^{\infty} e^{-r^2/2} r^l j_l(\lambda r) dr$$

A.3 Covariance of the operator U to rotations

Since a rotation R commutes with the Laplacian operator Δ, they share eigenspaces. The eigenspaces of the laplacian with eigenvalue $l(l+1)$ are span $(Y_l^{m'})_{m'=-l...l}$. So $\forall l \in \mathbb{N},\ \forall m \in [-l,l]$:

$$R.Y_l^m \in \text{span } (Y_l^{m'})_{m'=-l...l}$$

so $\forall R \in SO_3(\mathbb{R}),\ \forall l \in \mathbb{N},\ \forall m \in [-l,l],\ \exists \left(C_{l,m,m'}\right)_{m'=-l...l}$:

$$R.Y_l^m = \sum_{m'=-l}^{l} C_{l,m,m'}^R Y_l^{m'} \tag{A.2}$$

The rotations preserves the integral over the sphere $\mathbb{S}^2$:

$$\int_{\mathbb{S}^2} R.f = \int_{\mathbb{S}^2} f$$

thus it preserves the scalar product and the l_2 norm on the sphere $\mathbb{S}^2$ that we denote by $\|.\|_{2,\mathbb{S}^2}$:

$$\begin{aligned} \langle R.f, R.g\rangle_{\mathbb{S}^2} &= \langle f, g\rangle_{\mathbb{S}^2} \\ \|R.f\|_{2,\mathbb{S}^2} &= \|f\|_{2,\mathbb{S}^2} \end{aligned}$$

Since the functions $(Y_l^{m'})_{m'=-l...l}$ are orthonormal, $\forall R \in SO_3(\mathbb{R}),\ \forall l \in \mathbb{N},\ \forall (m, m_1, m_2) \in [-l,l]^3, m_1 \neq m_2$:

$$\begin{aligned} \|R.Y_l^m\|_{2,\mathbb{S}^2} &= \|Y_l^m\|_{2,\mathbb{S}^2} = 1 \\ \langle R.Y_l^{m_1}, R.Y_l^{m_2}\rangle_{\mathbb{S}^2} &= \langle Y_l^{m_1}, Y_l^{m_2}\rangle_{\mathbb{S}^2} = 0 \end{aligned}$$

Using the decomposition (A.2) on the orthonormal functions $(Y_l^{m'})_{m'=-l...l}$, we have :

$$\begin{aligned} \|R.Y_l^m\|_{2,\mathbb{S}^2} &= \sum_{m'=-l}^{l} |C_{l,m,m'}^R|^2 \\ \langle R.Y_l^{m_1}, R.Y_l^{m_2}\rangle_{\mathbb{S}^2} &= \sum_{m'=-l}^{l} \overline{C_{l,m_1,m'}^R} C_{l,m_2,m'}^R \end{aligned}$$

thus

$$\begin{aligned} \sum_{m'=-l}^{l} |C_{l,m,m'}^R|^2 &= 1 \\ \sum_{m'=-l}^{l} \overline{C_{l,m_1,m'}^R} C_{l,m_2,m'}^R &= 0 \end{aligned}$$

and we can define the unitary matrix

$$\Gamma_l^R = \left(C_{l,m,m'}^R\right)_{m,m' \in [-l,l]} \in U_{2l+1}(\mathbb{C}) \tag{A.3}$$

Using the spherical coordinates $r = (u, \theta, \phi)$, we decompose the wavelet $\psi_{l,m,j}$ in angular and radial part:

$$\psi_{l,m,j}(u, \theta, \phi) = F(u) Y_l^m(\theta, \phi)$$

The radial part $F(u)$is not affected a rotation R, thus:

$$\begin{aligned} R.\psi_{l,m,j}(u, \theta, \phi) &= F(r)\ R.Y_l^m(\theta, \phi) \\ &= F(r) \left(\sum_{m'=-l}^{l} C_{l,m,m'}^R Y_l^{m'}(\theta, \phi) \right) \\ &= \sum_{m'=-l}^{l} C_{l,m,m'}^R\ \psi_{l,m',j}(u, \theta, \phi) \end{aligned}$$

Then when convolving with a function f, we have:

$$\begin{aligned} (R.f) * \psi_{l,m,j}(r) &= \int f(R^{-1}r - R^{-1}v)\psi_{l,m,j}(v)\mathrm{d}v \\ &= \int f(R^{-1}r - w)\psi_{l,m,j}(Rw)\mathrm{d}w \text{ by change of variable } v = Rw \\ &= \int f(R^{-1}r - w)(R^{-1}.\psi_{l,m,j})(w)\mathrm{d}w \\ &= f * (R^{-1}.\psi_{l,m,j})(R^{-1}r) \\ &= f * \left(\sum_{m'=-l}^{l} C_{l,m,m'}^{R^{-1}} \psi_{l,m',j} \right) (R^{-1}r) \\ &= \sum_{m'=-l}^{l} C_{l,m,m'}^{R^{-1}}\ f * \psi_{l,m',j}(R^{-1}r) \\ &= \sum_{m'=-l}^{l} C_{l,m,m'}^{R^{-1}}\ R.(f * \psi_{l,m',j})(r) \end{aligned}$$

Then taking the modulus square and summing over m yields:

$$\begin{aligned} (U_{l,j}[R.f](r))^2 &= \sum_{m=-l}^{l} |(R.f) * \psi_{l,m,j}(r)|^2 \\ &= \sum_{m=-l}^{l} \left| \sum_{m'=-l}^{l} C_{l,m,m'}^{R^{-1}}\ R.(f * \psi_{l,m',j})(r) \right|^2 \end{aligned}$$

If we define the vector:

$$\alpha_{l,j}^{R}(r) = \left(R.(f * \psi_{l,m',j})(r)\right)_{m' \in [-l,l]}$$

such that

$$\sum_{m'=-l}^{l} C_{l,m,m'}^{R^{-1}} \; R.(f * \psi_{l,m',j})(r) = \left(\Gamma_l^{R^{-1}} \alpha_{l,j}^{R}\right)_m (r)$$

we have:

$$\begin{aligned}
(U_{l,j}[R.f](r))^2 &= \sum_{m=-l}^{l} |\left(\Gamma_l^{R^{-1}} \alpha_{l,u}^{R}\right)_m|^2(r) \\
&= \|\Gamma_l^{R^{-1}} \alpha_{l,j}^{R}(r)\|_{2,\mathbb{C}^{2l+1}} \\
&= \|\alpha_{l,u}^{R}(r)\|_{2,\mathbb{C}^{2l+1}} \text{ , since } \Gamma_l^{R^{-1}} \text{ is a unitary matrix} \\
&= \sum_{m=-l}^{l} |R.(f * \psi_{l,m,j})|^2(r) \\
&= \sum_{m=-l}^{l} R.|(f * \psi_{l,m,j})|^2(r) \\
&= R.\left(\sum_{m=-l}^{l} |(f * \psi_{l,m,j})|^2\right)(r) \\
&= R.\left(U_{l,j}[f](r)\right)^2
\end{aligned}$$

Then, taking the square root gives us the rotation covariance of $U_{l,j}[f]$.

A.4 Examples of $U_{l,j}\,[\rho]$

A.4.1 Single gaussian

Indeed, if there is only one body in our multi-body system, the density ρ is a single Gaussian function:

$$\begin{aligned}
\rho(u) &= \frac{1}{\sqrt{(2\pi)^3}\sigma^3} e^{-|u|^2/(2\sigma^2)} \\
\hat{\rho}(\omega) &= e^{-|\omega|^2\sigma^2/2}
\end{aligned}$$

The convolutions $\rho * \psi_{l,m,j}$ can be computed analytically :

- Recall that $\psi_{l,m,j}$ Fourier transform is:

$$\hat{\psi}_{l,m,j}(\lambda, \alpha, \beta) \mapsto K_l \frac{4\pi(-i)^l}{\sqrt{2\pi}^3} e^{-\sigma_{w,j}^2 \lambda^2/2} (\sigma_{w,j}\lambda)^l Y_l^m(\alpha, \beta)$$

- The Fourier transform of the convolution is:

$$\widehat{\rho * \psi_{l,m,j}} : (\lambda, \alpha, \beta) \mapsto K_l \frac{4\pi(-i)^l}{\sqrt{2\pi}^3} e^{-(\sigma_{w,j}^2+\sigma^2)\lambda^2/2} (\sigma_{w,j}\lambda)^l Y_l^m(\alpha, \beta)$$

- We introduce the variance $s_j = \sqrt{\sigma_{w,j}^2 + \sigma^2}$ and the ratio $k = \frac{s_j}{\sigma_{w,j}}$ such that:

$$\begin{aligned}\widehat{\rho * \psi_{l,m,j}}(\lambda, \alpha, \beta) &= \frac{K_l}{k^l} \frac{4\pi(-i)^l}{\sqrt{2\pi}^3} e^{\sigma_{w,j}^2 (k\lambda)^2/2} (\sigma_{w,j} k\lambda)^l Y_l^m(\alpha, \beta) \\ &= \frac{K_l}{k^l} \hat{\psi}_{l,m,j}(k\lambda, \alpha, \beta)\end{aligned}$$

- Using the fact that the function $\omega \mapsto \hat{f}(k\omega)$ is the Fourier transform of the function $t \mapsto \frac{1}{k^3} f(t/k)$, we have:

$$\begin{aligned}\rho * \psi_{l,m,j}(r, \theta, \phi) &= \frac{K_l}{k^l} \frac{1}{\sqrt{2\pi}^3 s_j^3} e^{-r^2/(2s_j^2)} \left(\frac{r}{s_j}\right)^l Y_l^m(\theta, \phi) \\ &= \frac{K_l}{k^l} \frac{1}{s_j^3} \psi_{l,m,0}(\frac{r}{s_j}, \theta, \phi)\end{aligned}$$

  We define the function $\Psi_{l,m,j} = \rho * \psi_{l,m,j}$:

$$\Psi_{l,m,j}(r, \theta, \phi) = \frac{K_l}{k^l} \frac{1}{s_j^3} \psi_{l,m,0}(\frac{r}{s_j}, \theta, \phi)$$

Take the modulus square and summing over m and taking the square root gives:

$$\begin{aligned}U_{l,j}[\rho](r, \theta, \phi) &= \frac{1}{k^l} \frac{K_l}{s_j^3} \left(\sum_{m=-l}^{l} |\psi_{l,m,0}(\frac{r}{s_j}, \theta, \phi)|^2 \right)^{1/2} \\ &= \frac{1}{k^l} \frac{K_l}{s_j^3} e^{-r^2/(2s_j^2)} \left(\frac{r}{s_j}\right)^l\end{aligned}$$

A.4.2 Multiple Gaussians

In the general case, we can compute $U_{l,j}[\rho]$. Indeed, since :

$$\rho(u) = \sum_k g(u - r_k) = \sum_k g_{r_k}(u)$$

we have :

$$\begin{aligned}\rho * \psi_{l,m,j}(u) &= \sum_k g_{r_k} * \psi_{l,m,j}(u) \\ &= \sum_k (g * \psi_{l,m,j})_{r_k}(u) \\ &= \sum_k (g * \psi_{l,m,j})_{r_k}(u) \\ &= \sum_k (\Psi_{l,m,j})_{r_k}(u)\end{aligned}$$

So we have:

$$U_{l,j}\left[\rho\right](r,\theta,\phi) = \left(\sum_{m=-l}^{l} \left|\sum_k (\Psi_{l,m,j})_{r_k}(r,\theta,\phi)\right|^2\right)^{1/2}$$

Appendix B

Patches K-nearest-neighbors image classifier

B.1 Mahanalobis distance and whitening

The Mahalanobis distance [Chandra et al., 1936, McLachlan, 1999] between two samples x and x' drawn from a random vector X with covariance Σ is defined as

$$D_M(x, x') = \sqrt{(x - x')^T \Sigma^{-1} (x - x')}$$

If the random vector X has identity covariance, it is simply the usual euclidian distance :

$$D_M(x, x') = \|x - x'\| \ .$$

Using the diagonalization of the coraviance matrix, $\Sigma = P \Lambda P^T$, the affine whitening operators of the random vector $\mathbf{X}$ are the operators

$$w : \mathbf{X} \mapsto O \Lambda^{-1/2} P^T (\mathbf{X} - \mu), \quad \forall O \in O_n(\mathbb{R}) \ . \tag{B.1}$$

For example, the PCA whitening operator is

$$w_{\text{PCA}} : \mathbf{X} \mapsto \Lambda^{-1/2} P^T (\mathbf{X} - \mu)$$

and the ZCA whitening operator is

$$w_{\text{ZCA}} : \mathbf{X} \mapsto P \Lambda^{-1/2} P^T (\mathbf{X} - \mu) \ .$$

For all whitening operator w we have

$$\|w(x) - w(x')\| = D_M(x, x')$$

since

$$\begin{aligned} \|w(x) - w(x')\| &= \|O \Lambda^{-1/2} P^T (x - x')\| \\ &= \sqrt{(x - x')^T P \Lambda^{-1/2} O^T O \Lambda^{-1/2} P^T (x - x')} \\ &= \sqrt{(x - x')^T P \Lambda^{-1} P^T (x - x')} \\ &= D_M(x, x') \ . \end{aligned}$$

B.2 Implementation of the patches K-nearest-neighbors encoding

In this section, we explicitly write the whitened patches with the whitening operator W. Recall that we consider the following set of euclidean pairwise distances:

$$\mathcal{C}_{i,x} = \{\|Wp_{i,x} - Wd\| \, d \in \mathcal{D}\}.$$

For each image patch we encode the K nearest neighbors of $Wp_{i,x}$ in the set $Wd, d \in \mathcal{D}$, for some $K \in 1 \dots |\mathcal{D}|$. We can use the square distance instead of the distance since it doesn't change the K nearest neighbors. We have

$$\|Wp_{i,x} - Wd\|^2 = \|Wp_{i,x}\|^2 - 2\langle p_{i,x}, W^T W d\rangle + \|Wd\|^2$$

The term $\|Wp_{i,x}\|^2$ doesn't affect the K nearest neighbors, so the K nearest neighbors are the K smallest values of

$$\left\{ \frac{\|Wd\|^2}{2} + \langle p_{i,x}, -W^T W d\rangle, \; d \in \mathcal{D} \right\}$$

This can be implemented in a convolution of the image using $-W^T W d$ as filters and $\|Wd\|^2/2$ as bias term, followed by a "vectorwise" non-linearity that binary encodes the K smallest values in the channel dimension. Once this is computed, we can then easily compute

$$\left\{ \frac{\|Wd\|^2}{2} + \langle p_{i,x}, W^T W d\rangle, \; d \in \mathcal{D} \right\}$$

which is the quantity needed to compute the K nearest neighbors in the set of negative patches $\overline{\mathcal{D}}$. This is a computationally efficient way of doubling the number of patches while making the representation invariant to negative transform.

B.3 Intrinsic dimension estimate

The following estimate of the intrinsic dimension d_{int} is introduced in Levina and Bickel [2004] as follows

$$d_{\text{int}}(p) = \left(\frac{1}{K-1} \sum_{k=1}^{K-1} \log \frac{\tau_K(p)}{\tau_k(p)} \right)^{-1}, \tag{B.2}$$

where $\tau_k(p)$ is the euclidean distance between the patch p and it's k-th nearest neighbor int the training set.

Bibliography

G. J Ackland, M. I. Mendelev, D. J. Srolovitz, S. Han, and A. V. Barashev. Development of an interatomic potential for phosphorus impurities in agr-iron. *Journal of Physics: Condensed Matter*, 16:S2629, 2004.

Julien Ah-Pine, Claudio Cifarelli, Stéphane Clinchant, Gabriela Csurka, and J Renders. Xrce's participation to imageclef 2008. 2008.

R Alexander, M-C Marinica, L Proville, F Willaime, K Arakawa, MR Gilbert, and SL Dudarev. Ab initio scaling laws for the formation energy of nanosized interstitial defect clusters in iron, tungsten, and vanadium. *Physical Review B*, 94(2):024103, 2016.

Alberto Ambrosetti, Anthony M Reilly, Robert A DiStasio Jr, and Alexandre Tkatchenko. Long-range correlation energy calculated from coupled atomic response functions. *The Journal of chemical physics*, 140(18):18A508, 2014.

M. Andreux, T. Angles, G. Exarchakis, R. Leonarduzzi, G. Rochette, L. Thiry, J. Zarka, S. Mallat, J. Andén, E. Belilovsky, J. Bruna, V. Lostanlen, M. J. Hirn, E. Oyallon, S. Zhang, C. E. Cella, and M. Eickenberg. Kymatio: Scattering transforms in python. *CoRR*, 2018. URL http://arxiv.org/abs/1812.11214.

Relja Arandjelovic, Andrew Zisserman, and . Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 609–617, 2017.

G. T. Barkema and Normand Mousseau. Event-based relaxation of continuous disordered systems. *Physical Review Letters*, 77:4358–4361, 1996.

James Barker, Johannes Bulin, Jan Hamaekers, and Sonja Mathias. Lc-gap: Localized coulomb descriptors for the gaussian approximation potential. In *Scientific Computing and Algorithms in Industrial Simulations*, pages 25–42. Springer, 2017.

Albert P. Bartók and Gábor Csányi. Gaussian approximation potentials: A brief tutorial introduction. *International Journal of Quantum Chemistry*, 115:1051–1057, 2015.

Albert P. Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. *Physical Review B*, 87:184115, 2013.

Albert P. Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. *Physical Review B*, 87(18):184115(16), May 2013.

M. I. Baskes. Modified embedded-atom potentials for cubic materials and impurities. *Physical Review B*, 46(5):2727, 1992.

Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Physical review letters*, 98(14):146401, 2007.

Jörg Behler. Atom-centered symmetry functions for constructing high-dimensional neural network

potentials. *Journal of Chemical Physics*, 134(7):074106, 2011. ISSN 0021-9606, 1089-7690.

Jörg Behler. Perspective: Machine learning potentials for atomistic simulations. *The Journal of Chemical Physics*, 145(17):170901, 2016. doi: 10.1063/1.4966192. URL https://doi.org/10.1063/1.496692.

Eugene Belilovsky, Michael Eickenberg, and Edouard Oyallon. Greedy layerwise learning can scale to imagenet. *arXiv preprint arXiv:1812.11446*, 2018.

F. Berthier, J. Creuze, T. Gabard, B. Legrand, M.-C. Marinica, and C. Mottet. Order-disorder or phase-separation transition: Analysis of the Au-Pd system by the effective site energy model. *Physical Review B*, 99(1):014108, 2019.

Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? In *International conference on database theory*, pages 217–235. Springer, 1999.

Kirill I Bolotin, K J Sikes, Zd Jiang, M Klima, G Fudenberg, J ea Hone, Ph Kim, and HL Stormer. Ultrahigh electron mobility in suspended graphene. *Solid state communications*, 146(9-10):351–355, 2008.

Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*, 2019.

J. Bruna and S. Mallat. Multiscale sparse microcanonical models. *arXiv:1801.02013*, 2019.

Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886, August 2013.

Vivien Cabannes, Thomas Kerdreux, Louis Thiry, Tina Campana, and Charly Ferrandes. Dialog on a canvas with a machine. In *NeuriIPS 2019 Workshop on Machine Learning for Creativity and Design*, 2019.

Vivien Cabannes, Thomas Kerdreux, and Louis Thiry. Diptychs of human and machine perceptions. *arXiv preprint arXiv:2010.13864*, 2020.

E. Cancès, F. Legoll, M.-C. Marinica, K. Minoukadeh, and F. Willaime. Some improvements of the activation-relaxation technique method for finding transition pathways on potential energy surfaces. *Journal of Chemical Physics*, 130(11):114711, 2009.

David M Ceperley and Berni J Alder. Ground state of the electron gas by a stochastic method. *Physical review letters*, 45(7):566, 1980.

Mahalanobis Prasanta Chandra et al. On the generalised distance in statistics. In *Proceedings of the National Institute of Sciences of India*, volume 2, pages 49–55, 1936.

Ming-Ming Cheng, Xiao-Chang Liu, Jie Wang, Shao-Ping Lu, Yu-Kun Lai, and Paul L Rosin. Structure-preserving neural style transfer. *IEEE Transactions on Image Processing*, 29:909–920, 2019.

S. Chiesa, P. M. Derlet, and S. L. Dudarev. Free energy of a <110> dumbbell interstitial defect in bcc fe: Harmonic and anharmonic contributions. *Physical Review B*, 79:214109, Jun 2009.

Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of imagenet as an alternative to the CIFAR datasets. *CoRR*, abs/1707.08819, 2017. URL http://arxiv.org/abs/1707.08819.

Eric Chu. Artistic influence gan, 2018.

DDL Chung. Review graphite. *Journal of materials science*, 37(8):1475–1489, 2002.

Sougwen Chung. Drawing operations, 2015, 2015.

Adam Coates and Andrew Y Ng. The importance of encoding versus training with sparse coding and vector quantization. 2011.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223, 2011.

Aron J Cohen and Paula Mori-Sánchez. Landscape of an exact energy functional. *Physical Review A*, 93(4):042511, 2016.

Aron J Cohen, Paula Mori-Sánchez, and Weitao Yang. Insights into current limitations of density functional theory. *Science*, 321(5890):792–794, 2008.

Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on image processing*, 13(9):1200–1212, 2004.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

Murray S Daw and Michael I Baskes. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Physical Review B*, 29(12):6443, 1984.

Murray S. Daw, Stephen M. Foiles, and Michael I. Baskes. The embedded-atom method: a review of theory and applications. *Materials Science Reports*, 9(7-8):251, 1993. ISSN 09202307.

Sandip De, Albert P Bartók, Gábor Csányi, and Michele Ceriotti. Comparing molecules and solids across structural and alchemical space. *Physical Chemistry Chemical Physics*, 18(20):13754–13769, 2016.

A De Clercq, S Giorgio, and C Mottet. Pd surface and pt subsurface segregation in pt1- c pd c nanoalloys. *Journal of Physics: Condensed Matter*, 28(6):064006, 2016.

Gilles Deleuze. *Logique de la sensation*. Edition de la Difference, Paris, France, 1981.

René Descartes. *Discours de la méthode:*. manuscript, 1637.

Lucile Dezerald, David Rodney, Emmanuel Clouet, Lisa Ventelon, and Francois Willaime. Plastic anisotropy and dislocation trajectory in BCC metals. *Nature Communications*, 7:11695, 2016. ISSN 2041-1723.

C. Domain and C.S. Becquart. Solute – 111 interstitial loop interaction in -fe: A dft study. *Journal of Nuclear Materials*, 499:582 – 594, 2018. ISSN 0022-3115.

Chris Donahue and Julian McAuley. Disentangled representations of style and content for visual art with generative adversarial networks, 2017.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Daniele Dragoni, Thomas D Daff, Gábor Csányi, and Nicola Marzari. Achieving dft accuracy with a machine-learning interatomic potential: Thermomechanics and defects in bcc ferromagnetic iron. *Physical Review Materials*, 2(1):013808, 2018.

Alexei A Efros and Thomas K Leung. Texture synthesis by non-parametric sampling. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1033–1038. IEEE, 1999.

Michael Eickenberg, Georgios Exarchakis, Matthew Hirn, and Stephane Mallat. Solid harmonic wavelet scattering: Predicting quantum molecular energy from invariant descriptors of 3d electronic densities. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, page 6540–6549. Curran Associates, Inc., 2017.

Michael Eickenberg, Georgios Exarchakis, Matthew Hirn, Stéphane Mallat, and Louis Thiry. Solid harmonic wavelet scattering for predictions of molecule properties. *The Journal of chemical physics*, 148(24):241732, 2018.

Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(4):44:1–44:10, 2012.

Ahmed Elgammal, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone. CAN: Creative Adversarial Networks, Generating "Art" by Learning About Styles and Deviating from Style Norms, 2017.

S. Alireza Etesami and Ebrahim Asadi. Molecular dynamics for near melting temperatures simulations of metals using modified embedded-atom method. *Journal of Physics and Chemistry of Solids*, 112: 61–72, 2018. ISSN 0022-3697.

Felix A Faber, Luke Hutchison, Bing Huang, Justin Gilmer, Samuel S Schoenholz, George E Dahl, Oriol Vinyals, Steven Kearnes, Patrick F Riley, and O Anatole Von Lilienfeld. Prediction errors of molecular machine learning models lower than hybrid dft error. *Journal of chemical theory and computation*, 13(11):5255–5264, 2017.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

Paul Flowers, Klaus Theopold, Richard Langley, William R Robinson, et al. Chemistry: Openstax. 2018.

Vladimir Fock. Näherungsmethode zur lösung des quantenmechanischen mehrkörperproblems. *Zeitschrift für Physik*, 61(1-2):126–148, 1930.

So Fujikake, Volker L Deringer, Tae Hoon Lee, Marcin Krynski, Stephen R Elliott, and Gábor Csányi. Gaussian approximation potential modeling of lithium intercalation in carbon nanostructures. *The Journal of chemical physics*, 148(24):241714, 2018.

Yaroslav Ganin, Tejas Kulkarni, Igor Babuschkin, SM Eslami, and Oriol Vinyals. Synthesizing programs for images using reinforced adversarial learning. *arXiv preprint arXiv:1804.01118*, 2018.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

Leon A Gatys, Alexander S Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Controlling perceptual factors in neural style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3985–3993, 2017.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness, 2019.

Golnaz Ghiasi, Honglak Lee, Manjunath Kudlur, Vincent Dumoulin, and Jonathon Shlens. Ex-

ploring the structure of a real-time, arbitrary neural artistic stylization network. *arXiv preprint arXiv:1705.06830*, 2017.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

Alex Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013. URL http://arxiv.org/abs/1308.0850.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

Holly Grimm. Training on Art Composition Attributes to Influence CycleGAN Art Generation, 2018.

Agrim Gupta, Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Characterizing and improving stability in neural style transfer. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4067–4076, 2017.

David Ha and Douglas Eck. A neural representation of sketch drawings. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=Hy6GHpkCW.

Katja Hansen, Franziska Biegler, Raghunathan Ramakrishnan, Wiktor Pronobis, O. Anatole von Lilienfeld, Klaus-Robert Müller, and Alexandre Tkatchenko. Machine learning predictions of molecular properties: Accurate many-body potentials and nonlocality in chemical space. *The Journal of Physical Chemistry Letters*, 6:2326–2331, 2015.

D. R. Hartree. The wave mechanics of an atom with a non-coulomb central field. part ii. some results and discussion. *Mathematical Proceedings of the Cambridge Philosophical Society*, 24(1):111–132, 1928. doi: 10.1017/S0305004100011920.

Dong-Chen He and Li Wang. Texture unit, texture spectrum, and texture analysis. *IEEE transactions on Geoscience and Remote Sensing*, 28(4):509–512, 1990.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Matthew Hirn, Stéphane Georges Mallat, and Nicolas Poilvert. Wavelet scattering regression of quantum chemical energies. *Multiscale Modeling Simulation*, 15, 2016.

Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. *Physical review*, 136(3B):B864, 1964.

Bing Huang and O Anatole Von Lilienfeld. Communication: Understanding molecular representations in machine learning: The role of uniqueness and target similarity, 2016.

Chen Huang, Arthur F Voter, and Danny Perez. Scalable kernel polynomial method for calculating transition rates. *Physical Review B*, 87(21):214106, 2013.

Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Nikolay Jetchev, Urs Bergmann, and Gokhan Yildirim. Copy the Old or Paint Anew? An Adversarial Framework for (non-) Parametric Image Stylization, 2018.

Ruoming Jin, Yuri Breitbart, and Chibuike Muoh. Data discretization unification. *Knowledge and Information Systems*, 19(1):1, 2009.

Yanghua Jin, Jiakai Zhang, Minjun Li, Yingtao Tian, Huachun Zhu, and Zhihao Fang. Towards the Automatic Anime Characters Creation with Generative Adversarial Networks, 2017.

Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, and Mingli Song. Neural style transfer: A review. *CoRR*, abs/1705.04058, 2017. URL http://arxiv.org/abs/1705.04058.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

John Edward Jones. On the determination of molecular fields.—ii. from the equation of state of a gas. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 106(738):463–477, 1924.

Jonas Jongejan, Henry Rowley, Takashi Kawashima, Jongmin Kim, and Nick Fox-Gieg. The quick, draw!-ai experiment. *Mount View, CA, accessed Feb*, 17, 2018.

Amy Kaczmarowski, Shujiang Yang, Izabela Szlufarska, and Dane Morgan. Genetic algorithm optimization of defect clusters in crystalline materials. *Computational Materials Science*, 98:234–244, 2015.

Thomas Kerdreux, Louis Thiry, and Erwan Kerdreux. Interactive neural style transfer with artists. *arXiv preprint arXiv:2003.06659*, 2020.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations*, San Diego, CA, USA, 2015.

Fred S. Kleiner. Gardner's art through the ages: The western perspective, volume 2. 2009.

Walter Kohn. Density functional and density matrix method scaling linearly with the number of atoms. *Physical Review Letters*, 76(17):3168, 1996.

Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical review*, 140(4A):A1133, 1965.

Nicholas Kolkin, Jason Salavon, and Greg Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity, 2019.

Dmytro Kotovenko, Artsiom Sanakoyeu, Pingchuan Ma, Sabine Lang, and Bjorn Ommer. A content transformation block for image style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10032–10041, 2019.

Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Clovis Lapointe, Thomas D Swinburne, Louis Thiry, Stéphane Mallat, Laurent Proville, Charlotte S Becquart, and Mihai-Cosmin Marinica. Machine learning surrogate models for prediction of point defect vibrational entropy. *Physical Review Materials*, 4(6):063802, 2020.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989. doi: 10.1162/neco.1989.1.4.541. URL https://doi.org/10.1162/neco.1989.1.4.541.

Yann LeCun, Bernhard E Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne E Hubbard, and Lawrence D Jackel. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, pages 396–404, 1990.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Elizaveta Levina and Peter J. Bickel. Maximum likelihood estimation of intrinsic dimension. In *Advances in neural information processing systems 17*, pages 777–784. MIT Press, 2004.

Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European conference on computer vision*, pages 702–716. Springer, 2016.

Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *Advances in neural information processing systems*, pages 386–396, 2017.

Zhiyuan Li, Ruosong Wang, Dingli Yu, Simon S Du, Wei Hu, Ruslan Salakhutdinov, and Sanjeev Arora. Enhanced convolutional neural tangent kernels. *arXiv preprint arXiv:1911.00809*, 2019.

Chih Wen Lin and Ting-Wei Su. Generating images from audio, 2018.

David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

Ming Lu, Hao Zhao, Anbang Yao, Feng Xu, Yurong Chen, and Li Zhang. Decoder network over lightweight reconstructed feature for fast semantic style transfer. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2469–2477, 2017.

Zhiyun Lu, Avner May, Kuan Liu, Alireza Bagheri Garakani, Dong Guo, Aurélien Bellet, Linxi Fan, Michael Collins, Brian Kingsbury, Michael Picheny, et al. How to scale up kernel methods to be as good as deep neural nets. *arXiv preprint arXiv:1411.4000*, 2014.

G. Lucas and R. Schäublin. Vibrational contributions to the stability of point defects in bcc iron: A first-principles study. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 267(18):3009 – 3012, 2009.

Eduardo Machado-Charry, Laurent Karim Béland, Damien Caliste, Luigi Genovese, Thierry Deutsch, Normand Mousseau, and Pascal Pochet. Optimized energy landscape exploration using the ab initio based activation-relaxation technique. *Journal of Chemical Physics*, 135(3):034102, 2011.

Julien Mairal. End-to-end kernel learning with supervised convolutional kernel networks. In *Advances in neural information processing systems*, pages 1399–1407, 2016.

Julien Mairal, Piotr Koniusz, Zaid Harchaoui, and Cordelia Schmid. Convolutional kernel networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

R. Malek and N. Mousseau. Dynamics of lennard-jones clusters: A characterization of the activation-relaxation technique. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 62(6):7723–7728, 2000.

S. Mallat. Understanding deep convolutional networks. *Phil. Trans. of Royal Society A*, 374(2065), 2016.

Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65 (10):1331–1398, 2012.

M. C. Marinica and *et al. Phondy - Phonons Dynamics*. CEA, Saclay, 2007-2019.

M.-C. Marinica and F. Willaime. Orientation of Interstitials in Clusters in α-Fe: A Comparison between Empirical Potentials. *Solid State Phenomena*, 129:67, 2007.

M-C Marinica, F Willaime, and N Mousseau. Energy landscape of small clusters of self-interstitial dumbbells in iron. *Physical Review B*, 83(9):094119, 2011.

M-C Marinica, Francois Willaime, and J-P Crocombette. Irradiation-induced formation of nanocrystallites with c15 laves phase structure in bcc iron. *Physical Review Letters*, 108(2):025501, 2012.

Goeffrey J McLachlan. Mahalanobis distance. *Resonance*, 4(6):20–26, 1999.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Noemi Montobbio, Alessandro Sarti, and Giovanna Citti. A metric model for the functional architecture of the visual cortex. 2019. URL http://arxiv.org/abs/1807.02479.

Youssef Mroueh. Wasserstein style transfer, 2019.

Reiichiro Nakano. Neural painters: A learned differentiable constraint for generating brushstroke paintings. *arXiv preprint arXiv:1904.08410*, 2019.

Kostya S Novoselov, Andre K Geim, Sergei V Morozov, D Jiang, Y_ Zhang, Sergey V Dubonos, Irina V Grigorieva, and Alexandr A Firsov. Electric field effect in atomically thin carbon films. *science*, 306 (5696):666–669, 2004.

P. Olsson, T. P. C. Klaver, and C. Domain. Ab initio study of solute transition-metal interactions with point defects in bcc fe. *Physical Review B*, 81:054102, Feb 2010.

E. Oyallon and S. Mallat. Deep roto-translation scattering for object classification. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2865–2873, 2014.

E. Oyallon, S. Zagoruyko, G. Huang, N. Komodakis, S. Lacoste-Julien, M. Blaschko, and E. Belilovsky. Scattering networks for hybrid representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9):2208–2221, Sep. 2019.

Edouard Oyallon. Building a regular decision boundary with deep networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Edouard Oyallon and Stephane Mallat. Deep roto-translation scattering for object classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Edouard Oyallon, Eugene Belilovsky, and Sergey Zagoruyko. Scaling the scattering transform: Deep hybrid networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5618–5627, 2017.

K. Schroeder P. H. Dederichs, R. Zeller. *Point Defects in Metals II, Dynamical Properties and Diffusion Controlled Reactions*. Springer Tracts in Modern Physics, Berlin, 1980.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

John P Perdew and Alex Zunger. Self-interaction correction to density-functional approximations for many-electron systems. *Physical Review B*, 23(10):5048, 1981.

John P Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Physical review letters*, 77(18):3865, 1996.

Florent Perronnin and Diane Larlus. Fisher vectors meet neural networks: A hybrid classification architecture. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3743–3752, 2015.

Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, pages 143–156. Springer, 2010.

S Plimpton. Fast Parallel Algorithms for Short-Range Molecular Dynamics. *Journal Computational Physics*, 117:1–19, 1995a.

S. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1):1 – 19, 1995b.

Laurent Proville, David Rodney, and Mihai-Cosmin Marinica. Quantum effect on thermally activated glide of dislocations. *Nature Materials*, 11:845–849, 2012.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1, 2014.

C. E. Rasmussen. *Gaussian Processes in Machine Learning*. Springer, Berlin, Heidelberg, 2004.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *arXiv preprint arXiv:1902.10811*, 2019.

Eric Risser, Pierre Wilmot, and Connelly Barnes. Stable and controllable neural texture synthesis and style transfer using histogram losses. *arXiv preprint arXiv:1701.08893*, 2017.

Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O. Anatole von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Physical Review Letters*, 108(5):058301, January 2012.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115 (3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Bjorn Ommer. A style-aware content loss for real-time hd style transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 698–714, 2018.

Jorge Sánchez, Florent Perronnin, Thomas Mensink, and Jakob Verbeek. Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245, 2013.

Bernhard Schmitzer. Stabilized sparse scaling algorithms for entropy regularized transport problems. *SIAM Journal on Scientific Computing*, 41(3):A1443–A1481, 2019.

Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet– a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24): 241722, 2018.

Vaishaal Shankar, Alex Fang, Wenshuo Guo, Sara Fridovich-Keil, Ludwig Schmidt, Jonathan Ragan-Kelley, and Benjamin Recht. Neural kernels without tangents. *arXiv preprint arXiv:2003.02237*, 2020.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Aurélien Soulié, Fabien Bruneval, Mihai-Cosmin Marinica, Samuel Murphy, and Jean-Paul Crocombette. Influence of vibrational entropy on the concentrations of oxygen interstitial clusters and uranium vacancies in nonstoichiometric $\mathrm{U}{\mathrm{O}}_{2}$. *Physical Review Materials*, 2(8):083607, 2018.

Frank H Stillinger and Thomas A Weber. Computer simulation of local order in condensed phases of silicon. *Physical review B*, 31(8):5262, 1985.

Thomas D. Swinburne and Danny Perez. Self-optimized construction of transition rate matrices from accelerated atomistic simulations with bayesian uncertainty quantification. *Physical Review Materials*, 2:053802, May 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Wojciech J Szlachta, Albert P Bartók, and Gábor Csányi. Accuracy and transferability of gaussian approximation potential models for tungsten. *Physical Review B*, 90(10):104108, 2014.

Wei Ren Tan, Chee Seng Chan, Hernán E. Aguirre, and Kiyoshi Tanaka. Artgan: Artwork synthesis with conditional categorical gans. In *2017 IEEE International Conference on Image Processing, ICIP 2017, Beijing, China, September 17-20, 2017*, pages 3760–3764. IEEE, 2017. ISBN 978-1-5090-2175-8. doi: 10.1109/ICIP.2017.8296985. URL https://doi.org/10.1109/ICIP.2017.8296985.

Louis Thiry, Michael Arbel, Eugene Belilovsky, and Edouard Oyallon. The unreasonable effectiveness of patches in deep convolutional kernels methods. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=aYuZO9DIdnn.

Alexandre Tkatchenko, Robert A DiStasio Jr, Roberto Car, and Matthias Scheffler. Accurate and efficient method for many-body van der waals interactions. *Physical review letters*, 108(23):236402, 2012.

Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE transactions on pattern analysis and machine intelligence*, 32(5):815–830, 2009.

Julien Toulouse, Francois Colonna, and Andreas Savin. Long-range–short-range separation of the electron-electron interaction in density-functional theory. *Physical Review A*, 70(6):062505, 2004.

Dmitry Ulyanov. Texture networks: Feed-forward synthesis of textures and stylized images. Association for Computing Machinery, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Gregory K Wallace. The jpeg still picture compression standard. *IEEE transactions on consumer electronics*, 38(1):xviii–xxxiv, 1992.

Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3360–3367. IEEE, 2010.

Philippe Weinzaepfel, Hervé Jégou, and Patrick Pérez. Reconstructing an image from its local descriptors. In *CVPR 2011*, pages 337–344. IEEE, 2011.

Ning Xie, Hirotaka Hachiya, and Masashi Sugiyama. Artist agent: A reinforcement learning approach to automatic stroke generation in oriental ink painting. *IEICE TRANSACTIONS on Information and Systems*, 96(5):1134–1144, 2013.

YACHT. Chain tripping, 2019.

John Zarka, Louis Thiry, Tomás Angles, and Stéphane Mallat. Deep network classification by scattering and homotopy dictionary learning. *arXiv preprint arXiv:1910.03561*, 2019.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018.

S. Zhang and S. Mallat. Wavelet phase harmonic covariance models of stationary processes. *submitted to Jour. of Pure and Applied Harmonic Analysis*, 2019.

Yulun Zhang, Chen Fang, Yilin Wang, Zhaowen Wang, Zhe Lin, Yun Fu, and Jimei Yang. Multimodal style transfer via graph cuts, 2019a.

Yulun Zhang, Chen Fang, Yilin Wang, Zhaowen Wang, Zhe Lin, Yun Fu, and Jimei Yang. Multimodal style transfer via graph cuts, 2019b.

Sicheng Zhao, Xin Zhao, Guiguang Ding, and Kurt Keutzer. Emotiongan: Unsupervised domain adaptation for learning discrete probability distributions of image emotions. In Susanne Boll, Kyoung Mu Lee, Jiebo Luo, Wenwu Zhu, Hyeran Byun, Chang Wen Chen, Rainer Lienhart, and Tao Mei, editors, *2018 ACM Multimedia Conference on Multimedia Conference, MM 2018, Seoul, Republic of Korea, October 22-26, 2018*, pages 1319–1327. ACM, 2018. ISBN 978-1-4503-5665-7. doi: 10.1145/3240508.3240591. URL https://doi.org/10.1145/3240508.3240591.

RÉSUMÉ

MOTS CLÉS

apprentissage profond, apprentissage automatique, classification d'images, surface d'énergie potentielle, regression d'énergie, méthodes locales

ABSTRACT

KEYWORDS

deep learning, machine learning, image classification, potential energy surface, energy regression, local methods, computational creativity, human-machine artistic interactions