Group Invariance, Stability to Deformations, and Complexity of Deep Convolutional Representations

Alberto Bietti, Julien Mairal

▶ **To cite this version:**

Alberto Bietti, Julien Mairal. Group Invariance, Stability to Deformations, and Complexity of Deep Convolutional Representations. 2017. hal-01536004v2

HAL Id: hal-01536004

https://inria.hal.science/hal-01536004v2

Preprint submitted on 14 Nov 2017 (v2), last revised 10 Oct 2018 (v4)

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

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

Group Invariance, Stability to Deformations, and Complexity of Deep Convolutional Representations*

Alberto Bietti
Inria†
alberto.bietti@inria.fr

Julien Mairal
Inria†
julien.mairal@inria.fr

November 14, 2017

Abstract

In this paper, we study deep signal representations that are invariant to groups of transformations and stable to the action of diffeomorphisms without losing signal information. This is achieved by generalizing the multilayer kernel construction introduced in the context of convolutional kernel networks and by studying the geometry of the corresponding reproducing kernel Hilbert space. We show that the signal representation is stable, and that models from this functional space, such as a large class of convolutional neural networks with homogeneous activation functions, may enjoy the same stability. In particular, we study the norm of such models, which acts as a measure of complexity, controlling both stability and generalization.

1 Introduction

The results achieved by deep neural networks for prediction tasks have been impressive in domains where data is structured and available in large amounts. In particular, convolutional neural networks (CNNs) [19] have shown to model well the local appearance of natural images at multiple scales, while also representing images with some invariance through pooling operations. Yet, the exact nature of this invariance and the characteristics of functional spaces where convolutional neural networks live are poorly understood; overall, these models are sometimes seen as clever engineering black boxes that have been designed with a lot of insight collected since they were introduced.

Understanding the geometry of these functional spaces is nevertheless a fundamental question. In addition to potentially bringing new intuition about the success of deep networks, it may for instance help solving the issue of regularization, by providing ways to control the variations of prediction functions in a principled manner. Small deformations of natural signals often preserve their main characteristics, such as the class label in a classification task (*e.g.*, the same digit with different handwritings may correspond to the same images up to small deformations), and provide a much richer class of transformations than translations. Representations that are stable to small deformations allow more robust models that may exploit these invariances, which may lead to improved sample complexity. The scattering transform [8, 23] is a recent attempt to characterize convolutional multilayer architectures based on wavelets. The theory provides an elegant characterization of invariance and stability properties of signals represented via the scattering operator, through a notion of Lipschitz stability to the action of diffeomorphisms. Nevertheless, these networks do not involve "learning" in the classical sense since the filters of the networks are pre-defined, and the resulting architecture differs significantly from the most used ones.

*This work was supported by a grant from ANR (MACARON project under grant number ANR-14-CE23-0003-01), by the ERC grant number 714381 (SOLARIS project), and from the MSR-Inria joint centre.

†Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France

In this work, we study these theoretical properties for more standard convolutional architectures from the point of view of positive definite kernels [37]. Specifically, we consider a functional space derived from a kernel for multi-dimensional signals, which admits a multilayer and convolutional structure that generalizes the construction of convolutional kernel networks (CKNs) [21, 22]. We show that this functional space contains a large class of CNNs with smooth homogeneous activation functions in addition to CKNs [21], allowing us to obtain theoretical results for both classes of models. While the stability of a CNN from this class depends on its norm in the functional space we consider, which is hard to control in practice, we show that the same stability is naturally obtained for CKNs by controlling the norm of the last prediction layer as done in [21].

The main motivation for introducing a kernel framework is to study separately data representation and predictive models. On the one hand, we study the translation-invariance properties of the kernel representation and its stability to the action of diffeomorphisms, obtaining similar guarantees as the scattering transform [23], while preserving signal information. When the kernel is appropriately designed, we also show how to obtain signal representations that are invariant to the action of any locally compact group of transformations. On the other hand, we show that these stability results can be translated to predictive models (CNNs and CKNs) by controlling their norm in the functional space, or simply the norm of the last layer in the case of CKNs. In particular, the RKHS norm controls both stability and generalization, so that stability may lead to improved sample complexity.

A preliminary version of this paper was published at the NIPS 2017 conference [5].

1.1 Related Work.

Our work relies on image representations introduced in the context of convolutional kernel networks [21, 22], which yield a sequence of spatial maps similar to traditional CNNs, but where each point on the maps is possibly infinite-dimensional and lives in a reproducing kernel Hilbert space (RKHS). The extension to signals with d spatial dimensions is straightforward. Since computing the corresponding Gram matrix as in classical kernel machines is computationally impractical, CKNs provide an approximation scheme consisting of learning finite-dimensional subspaces of each RKHS's layer, where the data is projected, see [21]. The resulting architecture of CKNs resembles traditional CNNs with a subspace learning interpretation and different unsupervised learning principles.

Another major source of inspiration is the study of group-invariance and stability to the action of diffeomorphisms of scattering networks [23], which introduced the main formalism and several proof techniques from harmonic analysis that were keys to our results. Our main effort was to extend them to more general CNN architectures and to the kernel framework. Invariance to groups of transformations was also studied for more classical convolutional neural networks from methodological and empirical points of view [9, 12], and for shallow learned representations [1] or kernel methods [17, 25, 32].

Note also that other techniques combining deep neural networks and kernels have been introduced earlier. Multilayer kernel machines appear for instance in [10, 36]. Shallow kernels for images modelling local regions were also proposed in [35], and a multilayer construction was proposed in [6]. More recently, different models based on kernels have been introduced in [2, 14, 24] to gain some theoretical insight about classical multilayer neural networks, while kernels are used to define convex models for two-layer neural networks in [48]. Theoretical and practical concerns for learning with multilayer kernels have been studied in [13, 14, 43, 47] in addition to CKNs [21, 22]. In particular, [13, 14] study certain classes of dot-product kernels with random feature approximations, [43] consider hierarchical Gaussian kernels with learned weights, and [47] study a convex formulation for learning a certain class of fully connected neural networks using a hierarchical kernel. Finally, we note that the Lipschitz stability of deep predictive models was found to be important to achieve robustness to adversarial examples [11]. Our results show that convolutional kernel networks already enjoy such a property without further modification.

1.2 Notation and Basic Mathematical Tools.

A positive definite kernel K that operates on a set $\mathcal{X}$ implicitly defines a reproducing kernel Hilbert space $\mathcal{H}$ of functions from $\mathcal{X}$ to $\mathbb{R}$, along with a mapping $\varphi : \mathcal{X} \to \mathcal{H}$. A *predictive model* associates to every point z

in $\mathcal{X}$ a label in $\mathbb{R}$; it consists of a linear function f in $\mathcal{H}$ such that $f(z) = \langle f, \varphi(z) \rangle_{\mathcal{H}}$, where $\varphi(z)$ is the *data representation*. Given now two points z, z' in $\mathcal{X}$, Cauchy-Schwarz's inequality allows us to control the variation of the predictive model f according to the geometry induced by the Hilbert norm $\|.\|_{\mathcal{H}}$:

$$|f(z) - f(z')| \leq \|f\|_{\mathcal{H}} \|\varphi(z) - \varphi(z')\|_{\mathcal{H}}. \tag{1}$$

This property implies that two points z and z' that are close to each other according to the RKHS norm should lead to similar predictions, when the model f has reasonably small norm in $\mathcal{H}$.

Then, we consider notation from signal processing similar to [23]. We call a signal x a function in $L^2(\Omega, \mathcal{H})$, where Ω is a subset of $\mathbb{R}^d$ representing spatial coordinates, and $\mathcal{H}$ is a Hilbert space, when $\|x\|_{L^2}^2 := \int_\Omega \|x(u)\|_{\mathcal{H}}^2 du < \infty$, where du is the Lebesgue measure on $\mathbb{R}^d$. Given a linear operator $T : L^2(\Omega, \mathcal{H}) \to L^2(\Omega, \mathcal{H}')$, the operator norm is defined as $\|T\|_{L^2(\Omega,\mathcal{H}) \to L^2(\Omega,\mathcal{H}')} := \sup_{\|x\|_{L^2(\Omega,\mathcal{H})} \leq 1} \|Tx\|_{L^2(\Omega,\mathcal{H}')}$. For the sake of clarity, we drop norm subscripts, from now on, using the notation $\|\cdot\|$ for Hilbert space norms, L^2 norms, and $L^2 \to L^2$ operator norms, while $|\cdot|$ denotes the Euclidean norm on $\mathbb{R}^d$. Some useful mathematical tools are also presented in Appendix A.

1.3 Organization of the Paper.

The rest of the paper is structured as follows:

- In Section 2, we introduce the main object studied in the paper: a multilayer convolutional kernel representation for continuous signals, based on a hierarchy of patch extraction, kernel mapping, and pooling operators. We present useful properties of this representation such as signal preservation, as well as ways to make it practical through discretization and kernel approximations in the context of CKNs.
- In Section 3, we present our main results regarding stability and invariance, namely that the kernel representation introduced in Section 2 is translation-invariant and stable to the action of diffeomorphisms. We then show in Section 3.2 that the same stability results apply in the presence of kernel approximations such as those of CKNs [21], and describe a generic way to modify the multilayer construction in order to guarantee invariance to the action of any locally compact group of transformations in Section 3.4.
- In Section 4, we study the functional spaces induced by our representation, showing that simple neural-network like functions with certain smooth activations are contained in the RKHS at intermediate layers, and that the RKHS of the full kernel induced by our representation contains a class of generic CNNs with smooth and homogeneous activations. We then present upper bounds on the RKHS norm of such CNNs, which serves as a measure of complexity, controlling both generalization and stability.
- Finally, we discuss in Section 5 how the obtained stability results apply to the practical setting of learning prediction functions. In particular, we explain why the regularization used in CKNs [21] provides a natural way to control stability, while a similar control is harder to achieve in generic CNNs.

2 Construction of the Multilayer Convolutional Kernel

We now present the multilayer convolutional kernel, which operates on signals with d spatial dimensions. The construction follows closely that of convolutional kernel networks [21] but is generalized to input signals defined on the continuous domain $\Omega = \mathbb{R}^d$. Dealing with continuous signals is indeed useful to characterize the stability properties of signal representations to small deformations, as done by Mallat [23] in the context of the scattering transform. The issue of discretization where Ω is a discrete grid is addressed in Section 2.1.

In what follows, we consider signals x_0 that live in $L^2(\Omega, \mathcal{H}_0)$, where typically $\mathcal{H}_0 = \mathbb{R}^{p_0}$ (*e.g.*, with $p_0 = 3$ and $d = 2$, the vector $x_0(u)$ in $\mathbb{R}^3$ may represent the RGB pixel value at location u in Ω). Then, we build a sequence of reproducing kernel Hilbert spaces $\mathcal{H}_1, \mathcal{H}_2, \ldots$, and transform x_0 into a sequence of "feature maps", respectively denoted by x_1 in $L^2(\Omega, \mathcal{H}_1)$, x_2 in $L^2(\Omega, \mathcal{H}_2)$, *etc.* As depicted in Figure 1, a new map x_k is built from the previous one x_{k-1} by applying successively three operators that perform patch extraction

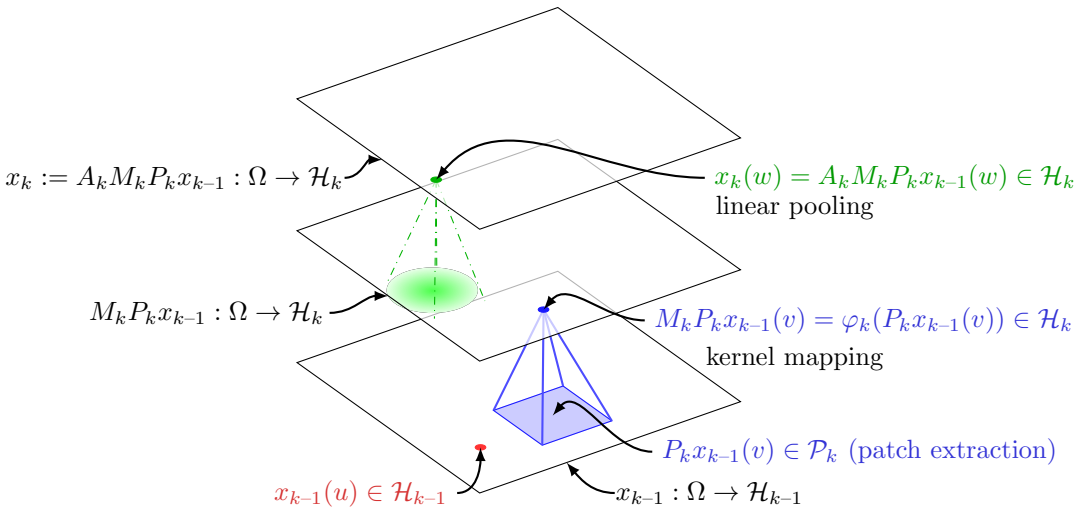

Figure 1: Construction of the k-th signal representation from the k–1-th one. Note that while Ω is depicted as a box in $\mathbb{R}^2$ here, our construction is supported on $\Omega = \mathbb{R}^d$.

(P_k), kernel mapping (M_k) to a new RKHS $\mathcal{H}_k$, and linear pooling (A_k), respectively. When going up in the hierarchy, the points $x_k(u)$ carry information from larger signal neighborhoods centered at u in Ω with more invariance, as we will formally show in Section 3.

Patch extraction operator. Given the layer x_{k-1}, we consider a patch shape S_k, defined as a compact centered subset of Ω, *e.g.*, a box, and we define the Hilbert space $\mathcal{P}_k := L^2(S_k, \mathcal{H}_{k-1})$ equipped with the norm $\|z\|^2 = \int_{S_k} \|z(u)\|^2 d\nu_k(u)$, where $d\nu_k$ is the normalized uniform measure on S_k for every z in $\mathcal{P}_k$. Specifically, we define the (linear) patch extraction operator $P_k : L^2(\Omega, \mathcal{H}_{k-1}) \to L^2(\Omega, \mathcal{P}_k)$ such that for all u in Ω,

$$P_k x_{k-1}(u) = (v \mapsto x_{k-1}(u+v))_{v \in S_k} \in \mathcal{P}_k.$$

Note that by equipping $\mathcal{P}_k$ with a normalized measure, it is easy to show that the operator P_k preserves the norm—that is, $\|P_k x_{k-1}\| = \|x_{k-1}\|$ and hence $P_k x_{k-1}$ is in $L^2(\Omega, \mathcal{P}_k)$.

Kernel mapping operator. In a second stage, we map each patch of x_{k-1} to a RKHS $\mathcal{H}_k$ with a kernel mapping $\varphi_k : \mathcal{P}_k \to \mathcal{H}_k$ associated to a positive definite kernel K_k that operates on patches. It is then possible to define the non-linear pointwise operator M_k such that for all u in Ω,

$$M_k P_k x_{k-1}(u) := \varphi_k(P_k x_{k-1}(u)) \in \mathcal{H}_k.$$

In this paper, we consider homogeneous dot-product kernels K_k operating on $\mathcal{P}_k$, as in [21]: if $z, z' \neq 0$,

$$K_k(z, z') = \|z\| \|z'\| \kappa_k\left(\frac{\langle z, z' \rangle}{\|z\| \|z'\|}\right) \quad \text{with} \quad \kappa_k(1) = 1 \quad \text{and} \quad \kappa_k'(1) = 1, \tag{2}$$

and $K_k(z, z') = 0$ if $z = 0$ or $z' = 0$. The kernel is positive definite if $\kappa_k : [-1, 1] \to \mathbb{R}$ is infinitely differentiable and admits a Maclaurin expansion with only non-negative coefficients [34, 37]—that is, for all u in $[-1, +1]$, $\kappa_k(u) = \sum_{j=0}^{+\infty} b_j u^j < +\infty$ with $b_j \geq 0$ for all j. The condition $\kappa_k(1) = 1$ ensures that the RKHS mapping preserves the norm—that is, $\|\varphi_k(z)\| = K_k(z, z)^{1/2} = \|z\|$, and thus $\|M_k P_k x_{k-1}(u)\| = \|P_k x_{k-1}(u)\|$ for all u in Ω; as a consequence, $M_k P_k x_{k-1}$ is always in $L^2(\Omega, \mathcal{H}_k)$. The technical condition $\kappa_k'(1) = 1$, where κ_k' is the first derivative of κ_k, ensures that the kernel mapping φ_k is non-expansive, according to Lemma 1 below.

Lemma 1 (Non-expansiveness of the kernel mappings)**.** *Consider a positive-definite kernel of the form* (2) *with RKHS mapping $\varphi_k : \mathcal{P}_k \to \mathcal{H}_k$. Then, φ_k is non-expansive—that is, for all z, z' in $\mathcal{P}_k$,*

$$\|\varphi_k(z) - \varphi_k(z')\| \leq \|z - z'\|.$$

Moreover, we remark that the kernel K_k is lower-bounded by the linear one

$$K_k(z, z') \geq \langle z, z' \rangle. \tag{3}$$

From the proof of the lemma, given in Appendix B, one may notice that the assumption $\kappa_k'(1) = 1$ is not critical and in fact, it may be safely replaced by $\kappa_k'(1) \leq 1$. Then, the non-expansiveness property would be preserved. Yet, we have chosen a stronger constraint since it yields a few simplifications in the stability analysis, where we use the relation (3) that requires $\kappa_k'(1) = 1$. In the same manner, the assumption may be replaced by $\kappa_k'(1) \leq C$ with $C > 1$; then, the kernel mapping would become Lipschitz with constant $\sqrt{C}$. Our stability results will hold in such a setting, but with constants that would grow exponentially with the number of layers, which would be reasonable for shallow architectures, but not for very deep ones.

Concrete examples of functions κ_k that satisfy the previous properties (2) are now given in the next table.

exponential kernel	$\kappa_{\exp}(\langle z, z' \rangle) = e^{\langle z, z' \rangle - 1}$
inverse polynomial kernel	$\kappa_{\text{inv-poly}}(\langle z, z' \rangle) = \frac{1}{2 - \langle z, z' \rangle}$
polynomial kernel of degree p	$\kappa_{\text{poly}}(\langle z, z' \rangle) = \frac{1}{(c+1)^p}(c + \langle z, z' \rangle)^p \quad$ with $\quad c = p - 1$
arc-cosine kernel of degree 1	$\kappa_{\text{acos}}(\langle z, z' \rangle) = \frac{1}{\pi}\left(\sin(\theta) + (\pi - \theta)\cos(\theta)\right)$ with $\theta = \arccos(\langle z, z' \rangle)$
Vovk's kernel of degree 3	$\kappa_{\text{vovk}}(\langle z, z' \rangle) = \frac{1}{3}\left(\frac{1 - \langle z, z' \rangle^3}{1 - \langle z, z' \rangle}\right) = \frac{1}{3}\left(1 + \langle z, z' \rangle + \langle z, z' \rangle^2\right)$

We note that the inverse polynomial kernel was used in [48] to build a convex model of a two-layer convolutional neural network, while the arc-cosine kernel appears in early deep kernel machines [10]. Note that the homogeneous exponential kernel reduces to the Gaussian kernel for unit norm vectors. Indeed, for all z, z' such that $\|z\| = \|z'\| = 1$, we have

$$\kappa_{\exp}(\langle z, z' \rangle) = e^{\langle z, z' \rangle - 1} = e^{-\frac{1}{2}\|z - z'\|^2},$$

and thus, we often refer to kernel (2) with the function $\kappa_{\exp}$ as the homogeneous Gaussian kernel. The kernel $\kappa(\langle z, z' \rangle) = e^{\alpha(\langle z, z' \rangle - 1)} = e^{-\frac{\alpha}{2}\|z - z'\|^2}$ with $\alpha \neq 1$ may also be used here, but we choose $\alpha = 1$ for simplicity since $\kappa'(1) = \alpha$ (see discussion above).

Pooling operator. The last step to build the layer x_k is to pool neighboring values to achieve some local shift-invariance. As in [21], we apply a linear convolution operator A_k with a Gaussian filter of scale σ_k, $h_{\sigma_k}(u) := \sigma_k^{-d} h(u/\sigma_k)$, where $h(u) = (2\pi)^{-d/2} \exp(-|u|^2/2)$. Then, for all u in Ω,

$$x_k(u) = A_k M_k P_k x_{k-1}(u) = \int_{\mathbb{R}^d} h_{\sigma_k}(u - v) M_k P_k x_{k-1}(v) dv \in \mathcal{H}_k, \tag{4}$$

where the integral is a Bochner integral (see, [15, 26]). By applying Schur's test to the integral operator A_k (see Appendix A), we obtain that $\|A_k\| \leq 1$. Thus, x_k is in $L^2(\Omega, \mathcal{H}_k)$, with $\|x_k\| \leq \|M_k P_k x_{k-1}\|$. Note that a similar pooling operator is used in the scattering representation [8, 23], though in a different way which does not affect subsequent layers.

Multilayer construction and prediction layer. Finally, we obtain a multilayer representation by composing multiple times the previous operators. In order to increase invariance with each layer, the size of

the patch S_k and pooling scale σ_k typically grow exponentially with k, with σ_k and the patch size $\sup_{c \in S_k} |c|$ of the same order. With n layers, the maps x_n may be written

$$x_n := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x_0 \ \in \ L^2(\Omega, \mathcal{H}_n). \tag{5}$$

It remains to define a kernel that will play the same role as the "fully connected" layer of classical convolutional neural networks. For that purpose, we may consider the Gaussian kernel defined for all x_0, x_0' in $L^2(\Omega, \mathcal{H}_0)$ by using the corresponding feature maps x_n, x_n' in $L^2(\Omega, \mathcal{H}_n)$ given by our multilayer construction (5),

$$\mathcal{K}_n(x_0, x_0') = e^{-\frac{\alpha}{2}\|x_n - x_n'\|^2}, \tag{6}$$

or we may simply use the linear kernel

$$\mathcal{K}_n(x_0, x_0') = \langle x_n, x_n' \rangle = \int_{u \in \Omega} \langle x_n(u), x_n'(u) \rangle du. \tag{7}$$

The Gaussian or the linear kernels are then associated to a RKHS denoted by $\mathcal{H}_{n+1}$, and a kernel mapping $\varphi_{n+1} : L^2(\Omega, \mathcal{H}_n) \to \mathcal{H}_{n+1}$, such that the final representation is given by $\varphi_{n+1}(x_n)$ in $\mathcal{H}_{n+1}$. We note that φ_{n+1} is non-expansive for the Gaussian kernel when $\alpha \leq 1$ (see Section B.1) as well as for the linear kernel (trivially, since in this case φ_{n+1} is an isometric linear mapping). Then, we have the relation $\mathcal{K}_n(x_0, x_0') := \langle \varphi_{n+1}(x_n), \varphi_{n+1}(x_n') \rangle$, for either kernel, and in particular, the RKHS $\mathcal{H}_{\mathcal{K}_n}$ of $\mathcal{K}_n$ contains all functions of the form $f(x_0) = \langle w, \varphi_{n+1}(x_n) \rangle$ with w in $\mathcal{H}_{n+1}$, see Appendix A.

2.1 Signal Preservation and Discretization

In this section, we first show that the multilayer kernel representation preserves all information about the signal at each layer, and besides, each feature map x_k can be sampled on a discrete set with no loss of information. This suggests a natural approach for discretization which will be discussed after the following lemma, whose proof is given in Appendix C.

Lemma 2 (Signal recovery from sampling)**.** *Assume that $\mathcal{H}_k$ contains all linear functions $\langle g, \cdot \rangle$ with g in $\mathcal{P}_k$ (this is true for all kernels K_k described in the previous section, according to Corollary 11 in Section 4.1 later), then the signal x_{k-1} can be recovered from a sampling of x_k at discrete locations as soon as the union of patches centered at these points covers Ω. It follows that x_k can be reconstructed from such a sampling.*

The previous construction defines a kernel representation for general signals in $L^2(\Omega, \mathcal{H}_0)$, which is an abstract object defined for theoretical purposes. In practice, signals are discrete, and it is thus important to discuss the problem of discretization, as done in [21]. For clarity, we limit the presentation to 1-dimensional signals ($\Omega = \mathbb{R}^d$ with $d = 1$), but the arguments can easily be extended to higher dimensions d when using box-shaped patches. Notation from the previous section is preserved, but we add a bar on top of all discrete analogues of their continuous counterparts. *e.g.*, $\bar{x}_k$ is a discrete feature map in $\ell^2(\mathbb{Z}, \bar{\mathcal{H}}_k)$ for some RKHS $\bar{\mathcal{H}}_k$.

Input signals x_0 and $\bar{x}_0$. Discrete signals acquired by a physical device are often seen as local integrators of signals defined on a continuous domain (*e.g.*, sensors from digital cameras integrate the pointwise distribution of photons that hit a sensor in a spatial and temporal window). Let us then consider a signal x_0 in $L^2(\Omega, \mathcal{H}_0)$ and s_0 a sampling interval. By defining $\bar{x}_0$ in $\ell_2(\mathbb{Z}, \mathcal{H}_0)$ such that $\bar{x}_0[n] = x_0(ns_0)$ for all n in $\mathbb{Z}$, it is thus natural to assume that $x_0 = A_0 x$, where A_0 is a pooling operator (local integrator) applied to an original continuous signal x. The role of A_0 is to prevent aliasing and reduce high frequencies; typically, the scale σ_0 of A_0 should be of the same magnitude as s_0, which we choose to be $s_0 = 1$ in the following, without loss of generality. This natural assumption will be kept later in the stability analysis.

Multilayer construction. We now want to build discrete feature maps $\bar{x}_k$ in $\ell^2(\mathbb{Z}, \bar{\mathcal{H}}_k)$ at each layer k involving subsampling with a factor s_k with respect to $\bar{x}_{k-1}$. We now define the discrete analogues of the operators P_k (patch extraction), M_k (kernel mapping), and A_k (pooling) as follows: for $n \in \mathbb{Z}$,

$$\begin{aligned}
\bar{P}_k \bar{x}_{k-1}[n] &:= e_k^{-1/2}(\bar{x}_{k-1}[n], \bar{x}_{k-1}[n+1], \ldots, \bar{x}_{k-1}[n+e_k-1]) \in \bar{\mathcal{P}}_k := \bar{\mathcal{H}}_{k-1}^{e_k} \\
\bar{M}_k \bar{P}_k \bar{x}_{k-1}[n] &:= \bar{\varphi}_k(\bar{P}_k \bar{x}_{k-1}[n]) \in \bar{\mathcal{H}}_k \\
\bar{x}_k[n] = \bar{A}_k \bar{M}_k \bar{P}_k \bar{x}_{k-1}[n] &:= s_k^{1/2} \sum_{m \in \mathbb{Z}} \bar{h}_k[ns_k - m] \bar{M}_k \bar{P}_k \bar{x}_{k-1}[m] = (\bar{h}_k * \bar{M}_k \bar{P}_k \bar{x}_{k-1})[ns_k] \in \bar{\mathcal{H}}_k,
\end{aligned}$$

where (i) $\bar{P}_k$ extracts a patch of size e_k starting at position n in $\bar{x}_{k-1}[n]$ (defining a patch centered at n is also possible), which lives in the Hilbert space $\bar{\mathcal{P}}_k$ defined as the direct sum of e_k times $\bar{\mathcal{H}}_{k-1}$; (ii) $\bar{M}_k$ is a kernel mapping identical to the continuous case, which preserves the norm, like M_k; (iii) $\bar{A}_k$ performs a convolution with a Gaussian filter and a subsampling operation with factor s_k. The next lemma shows that under mild assumptions, this construction preserves the signal information.

Lemma 3 (Signal recovery with subsampling)**.** *Assume that $\bar{\mathcal{H}}_k$ contains the linear functions $\langle w, \cdot \rangle$ for all w in $\bar{\mathcal{P}}_k$ and that $e_k \geq s_k$. Then, $\bar{x}_{k-1}$ can be recovered from $\bar{x}_k$.*

We note that this result relies on recovery by deconvolution of a pooling convolution with filter $\bar{h}_k$, which is stable when its scale parameter, typically of order s_k to prevent anti-aliasing, is small enough. This suggests using small values for e_k, s_k, as in typical recent convolutional architectures [40].

Links between the parameters of the discrete and continuous models. Due to subsampling, the patch size in the continuous and discrete models are related by a multiplicative factor. Specifically, a patch of size e_k with discretization corresponds to a patch S_k of diameter $e_k s_{k-1} s_{k-2} \ldots s_1$ in the continuous case. The same holds true for the scale parameter σ_k of the Gaussian pooling.

2.2 Kernel Approximations and Convolutional Kernel Networks

Besides discretization, two modifications are required to use the image representation we have described in practice. The first one consists of using feature maps with finite spatial support, which introduces border effects that we did not study as [23], but which are negligible when dealing with large realistic images. The second one requires finite-dimensional approximations of the kernel feature maps, leading to the convolutional kernel network model of [21]. Typically, each RKHS's mapping is approximated by performing a projection onto a subspace of finite dimension, a classical approach to make kernel methods work at large scale [16, 41, 45]. If we consider the kernel mapping $\varphi_k : \mathcal{P}_k \to \mathcal{H}_k$ at layer k, then the orthogonal projection onto the finite-dimensional subspace $\mathcal{F}_k = \text{span}(\varphi_k(z_1), \ldots, \varphi_k(z_{p_k})) \subseteq \mathcal{H}_k$, where the z_i's are p_k anchor points in $\mathcal{P}_k$, is given by the linear operator $\Pi_k : \mathcal{H}_k \to \mathcal{F}_k$ defined for f in $\mathcal{H}_k$ by

$$\Pi_k f := \sum_{1 \leq i,j \leq p_k} (K_{ZZ}^{-1})_{ij} \langle \varphi_k(z_i), f \rangle \varphi_k(z_j), \tag{8}$$

where K_{ZZ}^{-1} is the inverse (or pseudo-inverse) of the $p_k \times p_k$ kernel matrix $[K_k(z_i, z_j)]_{ij}$. As an orthogonal projection operator, Π_k is non-expansive, *i.e.*, $\|\Pi_k\| \leq 1$. We can then define the new approximate version $\tilde{M}_k$ of the kernel mapping operator M_k by

$$\tilde{M}_k P_k x_{k-1}(u) := \Pi_k \varphi_k(P_k x_{k-1}(u)) \in \mathcal{F}_k. \tag{9}$$

Note that all points in the feature map $\tilde{M}_k P_k x_{k-1}$ lie in the p_k-dimensional space $\mathcal{F}_k \subseteq \mathcal{H}_k$, which allows us to represent each point $\tilde{M}_k P_k x_{k-1}(u)$ by the finite dimensional vector

$$\psi_k(P_k x_{k-1}(u)) := K_{ZZ}^{-1/2} K_Z(P_k x_{k-1}(u)) \in \mathbb{R}^{p_k}, \tag{10}$$

with $K_Z(z) := (K_k(z_1, z), \dots, K_k(z_{p_k}, z))^\top$; this finite-dimensional representation preserves the Hilbertian inner product and norm in $\mathcal{F}_k$ such that $\|\psi_k(P_k x_{k-1}(u))\|_2^2 = \|\tilde{M}_k P_k x_{k-1}(u)\|_{\mathcal{H}_k}^2$, see [21] for details.

One advantage of such a finite-dimensional mapping is its compatibility with the multilayer construction, which builds layer k by manipulating input points from the previous RKHS $\mathcal{H}_{k-1}$. Here, the kernel approximation provides points in $\mathcal{F}_k \subseteq \mathcal{H}_k$, which remain in $\mathcal{F}_k$ after pooling since $\mathcal{F}_k$ is a linear subspace. Eventually, the sequence of RKHSs $\mathcal{H}_0$, $\mathcal{H}_1$, ..., is not affected by the finite-dimensional approximation. Besides, the stability results we will present next are preserved thanks to the non-expansiveness of the projection. In contrast, other kernel approximations such as random Fourier features [18, 31] do not provide points in the RKHS (see [3]), and their effect on the functional space derived from the multilayer construction is unclear.

It is then possible to derive theoretical results for the CKN model, which appears as a natural implementation of the kernel constructed previously; yet, we will also show in Section 4 that the results apply more broadly to CNNs that are contained in the functional space associated to the kernel. However, the stability of these CNNs depends on their RKHS norm, which is hard to control. In contrast, for CKNs, the studied representation corresponds to the one that is implemented, so that it is much more natural to control stability, typically by controlling the norm of the final prediction layer through regularization.

3 Stability to Deformations and Group Invariance

In this section, we study the translation-invariance and the stability under the action of diffeomorphisms of the kernel representation described in Section 2 for continuous signals. We use a similar characterization of stability to the one introduced by Mallat [23]: for a C^1-diffeomorphism $\tau : \Omega \to \Omega$, let L_τ denote the linear operator defined by $L_\tau x(u) = x(u - \tau(u))$; then, the representation $\Phi(\cdot)$ is *stable* under the action of diffeomorphisms if there exist two non-negative constants C_1 and C_2 such that

$$\|\Phi(L_\tau x) - \Phi(x)\| \leq (C_1 \|\nabla \tau\|_\infty + C_2 \|\tau\|_\infty)\|x\|, \tag{11}$$

where $\nabla\tau$ is the Jacobian of τ, $\|\nabla\tau\|_\infty := \sup_{u\in\Omega} \|\nabla\tau(u)\|$, and $\|\tau\|_\infty := \sup_{u\in\Omega} |\tau(u)|$. As in [23], our results assume the regularity condition $\|\nabla\tau\|_\infty \leq 1/2$. In order to have a translation-invariant representation, we want C_2 to be small (a translation is a diffeomorphism with $\nabla\tau = 0$), and indeed we will show that C_2 is proportional to $1/\sigma_n$, where σ_n is the scale of the last pooling layer, which typically increases exponentially with the number of layers n. Note that as in [23], our kernel representation does not lose signal information.

Additional assumptions. In order to study the stability of the representation (5), we assume that the input signal x_0 may be written as $x_0 = A_0 x$, where A_0 is an initial pooling operator at scale σ_0, which allows us to control the high frequencies of the signal in the first layer. As discussed previously in Section 2.1, this assumption is natural and compatible with any physical acquisition device. Note that σ_0 can be taken arbitrarily small, making the operator A_0 arbitrarily close to the identity, so that this assumption does not limit the generality of our results. Then, we are interested in understanding the stability of the representation

$$\Phi_n(x) := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 A_0 x.$$

We do not consider the prediction layer here for simplicity, but note that if we add a linear of Gaussian kernel prediction layer φ_{n+1} on top of Φ_n, then the stability of the full representation $\varphi_{n+1} \circ \Phi_n$ immediately follows from that of Φ_n thanks to the non-expansiveness of φ_{n+1} (see Section 2 and Section 3.3 below). Then, we make an assumption, that relates the scale of the pooling operator at layer $k-1$ with the size of the patch S_k: we assume that there exists $\kappa > 0$ such that for all $k \geq 1$,

$$\sup_{c\in S_k} |c| \leq \kappa \sigma_{k-1}. \tag{A1}$$

The scales σ_k are typically exponentially increasing with the layers k, and characterize the "resolution" of each feature map. This assumption corresponds to considering patch sizes that are adapted to these intermediate resolutions, which is a sensible way to extract signal information at different scales. Moreover, the stability

bounds we obtain hereafter increase with κ, which leads us to believe that small patch sizes lead to more stable representations, something which matches well the trend of using small, 3x3 convolution filters at each scale in modern deep architectures (*e.g.*, [40]).

Finally, before presenting our stability results, we recall a few properties of the operators involved in the representation Φ_n, which are heavily used in the analysis.

1. **Patch extraction operator**: P_k is linear and preserves the norm;
2. **Kernel mapping operator**: M_k preserves the norm and is non-expansive;
3. **Pooling operator**: A_k is linear and non-expansive $\|A_k\| \leq 1$;

The rest of this section is organized into three parts. We present the main stability results in Section 3.1, explain their compatibility with kernel approximations in Section 3.2, and discuss qualitatively these stability and invariance properties in Section 3.3. In particular, we show that the stability property (11) is non-trivial to achieve in the context of our kernel representation, by discussing the effect of Φ_n on the norm of an input signal, and its non-expansiveness. Finally, we introduce mechanisms to achieve invariance to any group of transformations in Section 3.4.

3.1 Stability Results and Translation Invariance

In this section, we show that our kernel representation Φ_n satisfies the stability property (11), with a constant C_2 inversely proportional to σ_n, thereby achieving near-invariance to translations. This translation invariance will be extended to more general transformation groups in Section 3.4.

General bound for stability. The following result gives an upper bound on the quantity of interest, $\|\Phi_n(L_\tau x) - \Phi_n(x)\|$, in terms of the norm of various linear operators which control how τ affects each layer. The commutator of linear operators A and B is denoted by $[A, B] = AB - BA$.

Proposition 4. *For any x in $L^2(\Omega, \mathcal{H}_0)$, we have*

$$\|\Phi_n(L_\tau x) - \Phi_n(x)\| \leq \left(\sum_{k=1}^{n} \|[P_k A_{k-1}, L_\tau]\| + \|[A_n, L_\tau]\| + \|L_\tau A_n - A_n\| \right) \|x\|. \tag{12}$$

In the case of a translation $L_\tau x(u) = L_c x(u) = x(u-c)$, it is easy to see that pooling and patch extraction operators commute with L_c (this is also known as *covariance* or *equivariance* to translations), so that we are left with the term $\|L_c A_n - A_n\|$, which should control translation invariance. For general diffeomorphisms τ, we no longer have exact covariance, but we show below that commutators are stable to τ, in the sense that $\|[P_k A_{k-1}, L_\tau]\|$ is controlled by $\|\nabla \tau\|_\infty$, while $\|L_\tau A_n - A_n\|$ is controlled by $\|\tau\|_\infty$ and decays with the pooling size σ_n.

Bound on $\|[P_k A_{k-1}, L_\tau]\|$. We begin by noting that $P_k z$ can be identified with $(L_c z)_{c \in S_k}$ isometrically for all z in $L^2(\Omega, \mathcal{H}_{k-1})$, since $\|P_k z\|^2 = \int_{S_k} \|L_c z\|^2 d\nu_k(c)$ by Fubini's theorem. Then,

$$\begin{aligned} \|P_k A_{k-1} L_\tau z - L_\tau P_k A_{k-1} z\|^2 &= \int_{S_k} \|L_c A_{k-1} L_\tau z - L_\tau L_c A_{k-1} z\|^2 d\nu_k(c) \\ &\leq \sup_{c \in S_k} \|L_c A_{k-1} L_\tau x - L_\tau L_c A_{k-1} z\|^2, \end{aligned}$$

so that $\|[P_k A_{k-1}, L_\tau]\| \leq \sup_{c \in S_k} \|[L_c A_{k-1}, L_\tau]\|$. The following result lets us bound $\|[L_c A_{k-1}, L_\tau]\|$ when $|c| \leq \kappa \sigma_{k-1}$, which is satisfied under assumption (A1).

Lemma 5. *Let A_σ be the pooling operator with kernel $h_\sigma(u) = \sigma^{-d}h(u/\sigma)$. If $\|\nabla\tau\|_\infty \leq 1/2$, there exists a constant C_1 such that for any σ and $|c| \leq \kappa\sigma$, we have*

$$\|[L_c A_\sigma, L_\tau]\| \leq C_1 \|\nabla\tau\|_\infty,$$

where C_1 depends only on h and κ.

A similar result is obtained by Mallat [23, Lemma E.1] for commutators of the form $[A_\sigma, L_\tau]$, but we extend it to handle integral operators $L_c A_\sigma$ with a shifted kernel. The proof (given in Appendix C.4) follows closely [23] and relies on the fact that $[L_c A_\sigma, L_\tau]$ is an integral operator in order to bound its norm via Schur's test. Note that κ can be made larger, at the cost of an increase of the constant C_1 of the order κ^{d+1}.

Bound on $\|L_\tau A_n - A_n\|$. We bound the operator norm $\|L_\tau A_n - A_n\|$ in terms of $\|\tau\|_\infty$ using the following result due to Mallat [23, Lemma 2.11], with $\sigma = \sigma_n$:

Lemma 6. *If $\|\nabla\tau\|_\infty \leq 1/2$, we have*

$$\|L_\tau A_\sigma - A_\sigma\| \leq \frac{C_2}{\sigma}\|\tau\|_\infty, \tag{13}$$

with $C_2 = 2^d \cdot \|\nabla h\|_1$.

Combining Proposition 4 with Lemmas 5 and 6, we immediately obtain the following result.

Theorem 7. *Assume* (A1). *If $\|\nabla\tau\|_\infty \leq 1/2$, we have*

$$\|\Phi_n(L_\tau x) - \Phi_n(x)\| \leq \left(C_1\,(1+n)\,\|\nabla\tau\|_\infty + \frac{C_2}{\sigma_n}\|\tau\|_\infty\right)\|x\|. \tag{14}$$

This result matches the desired notion of stability in Eq. (11), with a translation-invariance factor that decays with σ_n. The dependence on a notion of depth (the number of layers n here) also appears in [23], with a factor equal to the maximal length of scattering paths, and with the same condition $\|\nabla\tau\|_\infty \leq 1/2$. However, while the norm of the scattering representation is preserved as the length of these paths goes to infinity, the norm of $\Phi_n(x)$ may decrease with depth due to pooling layers, as discussed previously, making it necessary either to use a Gaussian kernel for the prediction layer (then $\|\varphi_{n+1}(\Phi_n(x))\| = 1$), or to make assumptions about the signal spectrum to ensure that a significant part of the signal norm is preserved. We discuss this in more detail in Section 3.3.

Stability of learned CNNs. In Section 4, we will study how this stability result applies to a broad class of CNNs that are contained in the functional space induced by our multilayer kernel. It is worth noting, however, that the results of this section can also apply directly to generic CNNs with ρ-Lipschitz activations σ satisfying $\sigma(0) = 0$ (this is true with $\rho = 1$ for the ReLU and tanh activations, for instance), at the cost of an additional factor depending on the product of spectral norms of convolutional mappings. Indeed, one can consider a similar construction with patches in $L^2(S_k, \mathbb{R}^{p_{k-1}})$ and non-linear mappings of the form

$$\varphi_k : z \in L^2(S_k, \mathbb{R}^{p_{k-1}}) \mapsto \sigma\left(\int_{S_k} W_k(u) z(u) d\nu_k(u)\right) \in \mathbb{R}^{p_k},$$

where $W_k(u) \in \mathbb{R}^{p_k \times p_{k-1}}$ for $u \in S_k$ and σ is applied element-wise. If we denote by B_k the spectral norm of the linear operator $z \mapsto \int_{S_k} W_k(u) z(u) d\nu_k(u)$, then φ_k satisfies

$$\|\varphi_k(z) - \varphi_k(z')\| \leq \rho B_k \|z - z'\| \quad \text{and} \quad \|\varphi_k(z)\| \leq \rho B_k \|z\|, \tag{15}$$

and the stability bound (14) holds with an additional multiplicative factor $\rho^n \prod_k B_k$.

3.2 Stability with Kernel Approximations

As in the analysis of the scattering transform of [23], we have characterized the stability and shift-invariance of the data representation for continuous signals, in order to give some intuition about the properties of the corresponding discrete representation, which we have described in Section 2.1.

Another approximation performed in the CKN model of [21] consists of adding projection steps on finite-dimensional subspaces of the RKHS's layers, as discusssed in Section 2.2. Interestingly, the stability properties we have obtained previously are compatible with these steps. We may indeed replace the operator M_k with the operator $\tilde{M}_k z(u) = \Pi_k \varphi_k(z(u))$ for any map z in $L^2(\Omega, \mathcal{P}_k)$, instead of $M_k z(u) = \varphi_k(z(u))$; $\Pi_k : \mathcal{H}_k \to \mathcal{F}_k$ is here an orthogonal projection operator onto a linear subspace, given in (8). Then, $\tilde{M}_k$ does not necessarily preserve the norm anymore, but $\|\tilde{M}_k z\| \leq \|z\|$, with a loss of information equal to $\|M_k z - \tilde{M}_k z\|$ corresponding to the quality of approximation of the kernel K_k on the points $z(u)$. On the other hand, the non-expansiveness of M_k is satisfied thanks to the non-expansiveness of the projection. In summary, it is possible to show that the conclusions of Theorem 7 remain valid when adding the CKN projection steps at each layer, but some signal information is lost in the process.

3.3 Discussions

We now discuss a few properties of the kernel representation in terms of norm preservation and non-expansiveness. We consider in this section the full kernel representation, including a prediction layer, which is given by $\Phi(x) = \varphi_{n+1}(\Phi_n(x))$, where φ_{n+1} is the kernel feature map of either a Gaussian kernel (6) with $\alpha = 1$, or a linear kernel (7). In both cases, φ_{n+1} is non-expansive, which yields

$$\|\Phi(L_\tau x) - \Phi(x)\| \leq \|\Phi_n(L_\tau x) - \Phi_n(x)\|,$$

such that the stability result of Theorem 7 also applies to Φ. Then, we will show that

1. the norm of $\|\Phi(x)\|$ is of the same order of magnitude as the norm $\|x\|$;
2. the kernel representation is non-expansive but it is not contractive—that is,

$$\sup_{x, x' \in L^2(\Omega, \mathcal{H}_0)} \frac{\|\Phi(x) - \Phi(x')\|}{\|x - x'\|} = 1. \tag{16}$$

   Otherwise, the proximity of $\Phi(L_\tau x)$ and $\Phi(x)$ would not be surprising; after all, any point $\Phi(x)$ could be close to another one $\Phi(x')$ if the representation Φ was contractive.

In [23], the first point is addressed for the scattering transform by showing that the scattering operator preserves the norm asymptotically. Here, we address this question in the next lemma.

Lemma 8 (Norm of $\Phi(x)$)**.** *For the two choices of prediction layers, $\Phi(x)$ satisfies*

$$\|\Phi(x)\| = 1 \quad \textit{(Gaussian)}, \qquad \|\Phi(x)\| \geq \|A_n A_{n-1} \dots A_0 x\| \quad \textit{(Linear)}.$$

Thus, with a Gaussian prediction layer, the representation always has norm 1, and hence is of the order of $\|x\|$ when the signal x is appropriately normalized. In the case of a linear prediction layer, the operator $A_n A_{n-1} \dots A_0$ is equivalent to a single pooling operator A_σ of scale $\sigma = (\sigma_n^2 + \dots + \sigma_0^2)^{1/2}$, which is of the order of σ_n when σ_k grows exponentially with k. The lower bound in Lemma 8 then approximately corresponds to the amount of energy of the signal x concentrated in frequencies smaller than $1/\sigma$, since the operator A_σ strongly attenuates frequencies higher than $1/\sigma$. Natural signals such as natural images often have high energy in the low-frequency domain (the power spectra of natural images is empirically considered to have a polynomial decay in $1/f^2$, where f is the signal frequency [44]). For such classes of signals, a large part of the signal energy is then preserved by the pooling operator A_σ.

Next, we give a basic result showing that the representation is not contractive.

Lemma 9 (Non-expansive and non-contractive representation)**.** *The representation Φ satisfies (16).*

Note that this result would not be satisfied if we used kernels with contractive mappings φ_k, *i.e.*, ρ-Lipschitz with $\rho < 1$. In particular for a depth-n representation, $\|\Phi(x) - \Phi(x')\|$ would become $O(\rho^n)$ times smaller than $\|x - x'\|$. In the context of CKNs, poor kernel approximations could lead to similar behavior.

3.4 Global Invariance to Group Actions

In Section 3.1, we have seen how the kernel representation of Section 2 creates invariance to translations by commuting with the action of translations at intermediate layers, and how the last pooling layer on the translation group governs the final level of invariance. It is often useful to encode invariances to different groups of transformations, such as rotations or reflections (see, *e.g.*, [12, 23, 32, 39]). Here, we show how this can be achieved by defining adapted patch extraction and pooling operators that commute with the action of a transformation group G (this is known as group covariance or equivariance). We assume that G is locally compact such that we can define a left-invariant Haar measure μ—that is, a measure on G that satisfies $\mu(gS) = \mu(S)$ for any Borel set $S \subseteq G$ and g in G. We assume the initial signal $x(u)$ is defined on G, and we define subsequent feature maps on the same domain. The action of an element g in G is denoted by L_g, where $L_g x(u) = x(g^{-1}u)$. Then, we are interested in defining a layer—that is, a succession of patch extraction, kernel mapping, and pooling operators—that commutes with L_g, in order to achieve equivariance to G.

Patch extraction. We define patch extraction as follows

$$Px(u) = (x(uv))_{v \in S} \quad \text{for all} \quad u \in G,$$

where $S \subset G$ is a patch centered at the identity. P commutes with L_g since

$$PL_g x(u) = (L_g x(uv))_{v \in S} = (x(g^{-1}uv))_{v \in S} = Px(g^{-1}u) = L_g Px(u).$$

Kernel mapping. The pointwise operator M is defined as in Section 2, and thus commutes with L_g.

Pooling. The pooling operator on the group G is defined in a similar fashion as [32] by

$$Ax(u) = \int_G x(uv)h(v)d\mu(v) = \int_G x(v)h(u^{-1}v)d\mu(v),$$

where h is a pooling filter typically localized around the identity element. It is easy to see from the first expression of $Ax(u)$ that $AL_g x(u) = L_g Ax(u)$, making the pooling operator G-equivariant.

In our analysis of stability in Section 3.1, we saw that inner pooling layers are useful to guarantee stability to local deformations, while global invariance is achieved mainly through the last pooling layer. In some cases, one only needs stability to a subgroup of G, while achieving global invariance to the whole group, *e.g.*, in the roto-translation group [30], one might want invariance to a global rotation but stability to local translations. Then, one can perform pooling just on the subgroup to stabilize (*e.g.*, translations) in intermediate layers, while pooling on the entire group at the last layer to achieve the global group invariance.

4 Link with Existing Convolutional Architectures

In this section, we study the functional spaces (RKHS) that arise from our multilayer kernel representation, and examine the connections with more standard convolutional architectures. We begin by considering in Section 4.1 the intermediate kernels K_k, showing that their RKHS contains simple neural-network-like functions defined on patches with smooth activations, while in Section 4.2 we show that a certain class of generic CNNs are contained in the RKHS $\mathcal{H}_{\mathcal{K}_n}$ of the full multilayer kernel $\mathcal{K}_n$ and characterize their RKHS norm. This is achieved by considering particular functions in each intermediate RKHS defined in terms of the convolutional filters of the CNN. A consequence of these results is that our stability and invariance properties from Section 3 are valid for this broad class of CNNs.

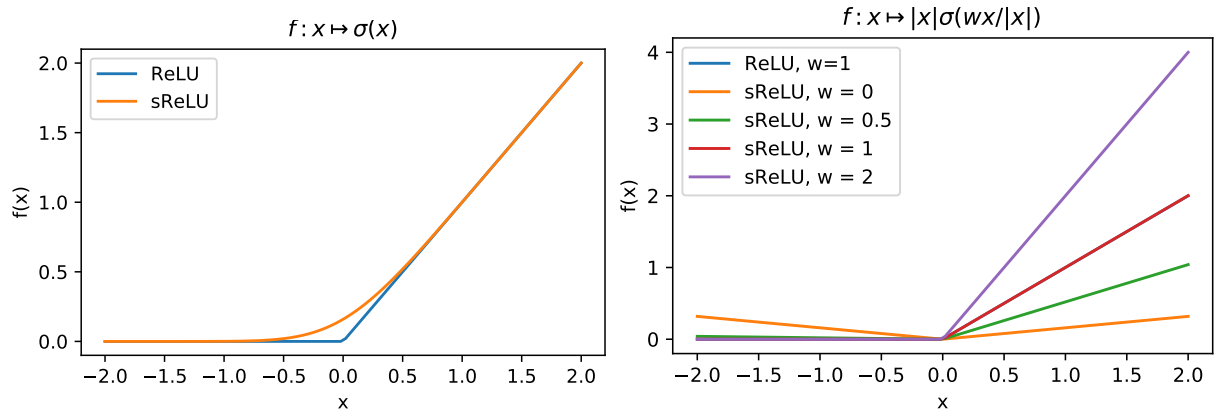

Figure 2: Comparison of one-dimensional functions obtained with relu and smoothed relu (sReLU) activations. (Left) non-homogeneous setting of [47, 48]. (Right) our homogeneous setting, for different values of the parameter w. Note that for $w \geq 0.5$, sReLU and ReLU are indistinguishable.

4.1 Activation Functions and Kernels K_k

Before introducing formal links between our kernel representation and classical convolutional architectures, we study in more details the kernels K_k described in Section 2 and their RKHSs $\mathcal{H}_k$. In particular, we are interested in characterizing which types of functions live in $\mathcal{H}_k$. The next lemma extends some results of [47, 48], originally developed for the inverse polynomial and Gaussian kernels; it shows that the RKHS may contain simple "neural network" functions with activations σ that are smooth enough. This observation will be useful in the sequel to characterize which CNNs are part of the RKHS of the multilayer convolutional kernel.

Lemma 10 (Activation functions and RKHSs $\mathcal{H}_k$)**.** *Let us consider a function $\sigma : [-1, 1] \to \mathbb{R}$ that admits a polynomial expansion $\sigma(u) := \sum_{j=0}^{\infty} a_j u^j$. Consider a kernel K_k from Section 2, given in (2). Let $\kappa_k(u) = \sum_{j=0}^{\infty} b_j u^j$ be the Maclaurin expansion of the function κ_k. Assume further that $a_j = 0$ if $b_j = 0$ for all j, and define the function $C_\sigma^2(\lambda^2) := \sum_{j=0}^{\infty} (a_j^2/b_j)\lambda^{2j}$. Let g in $\mathcal{P}_k$ be such that $C_\sigma^2(\|g\|^2) < \infty$. Then, the RKHS $\mathcal{H}_k$ contains the function*

$$f : z \mapsto \|z\|\sigma(\langle g, z\rangle/\|z\|), \tag{17}$$

and its norm satisfies $\|f\| \leq C_\sigma(\|g\|^2)$.

This result immediately implies the next corollary, which was also found to be useful in our analysis.

Corollary 11 (Linear functions and RKHSs)**.** *The RKHSs $\mathcal{H}_k$ considered in this paper contain all linear functions of the form $z \mapsto \langle g, z\rangle$ with g in $\mathcal{P}_k$.*

The previous lemma shows that for many choices of smooth functions σ, the RKHSs $\mathcal{H}_k$ contains the functions of the form (17). While the non-homogeneous functions $z \mapsto \sigma(\langle g, z\rangle)$ are standard in neural networks, the homogeneous variant is not. Yet, we note that (i) the most successful activation function, namely rectified linear units, is homogeneous—that is, $\text{relu}(\langle g, z\rangle) = \|z\|\text{relu}(\langle g, z\rangle/\|z\|)$; (ii) while relu is nonsmooth and thus not in our RKHSs, there exists a smoothed variant that satisfies the conditions of Lemma 10 for useful kernels. As noticed in [47, 48], this is for instance the case for the inverse polynomial kernel. In Figure 2, we plot and compare these different variants of relu.

4.2 Convolutional Neural Networks and their Complexity

We now study the connection between the kernel representation defined in Section 2 and CNNs. Specifically, we show that the RKHS of the final kernel $\mathcal{K}_n$ obtained from our kernel construction contains a set of CNNs on continuous domains with certain types of smooth homogeneous activations. An important consequence is

that the stability results of previous sections apply to this class of CNNs, although the stability will depend on the RKHS norm, as discussed later in Section 5. The RKHS norm also serves as a measure of model complexity, thus controlling both generalization and stability.

CNN maps construction. We now define a CNN function f_σ that takes as input an image $z_0 = x_0$ in $L^2(\Omega, \mathbb{R}^{p_0})$ with p_0 channels, and build a sequence of feature maps, represented at layer k as a function z_k in $L^2(\Omega, \mathbb{R}^{p_k})$ with p_k channels; the map z_k is obtained from a previous one z_{k-1} by performing linear convolutions with a set of filters $(w_k^i)_{i=1,\dots,p_k}$, followed by a pointwise activation function σ to obtain an intermediate feature map $\tilde{z}_k$, then by applying a linear pooling filter. Note that each w_k^i is in $L^2(S_k, \mathbb{R}^{p_{k-1}})$, with channels denoted by w_k^{ij} in $L^2(S_k, \mathbb{R})$. Formally, the intermediate map $\tilde{z}_k$ in $L^2(\Omega, \mathbb{R}^{p_k})$ is obtained by

$$\tilde{z}_k^i(u) = n_k(u)\sigma\big(\langle w_k^i, P_k z_{k-1}(u)\rangle / n_k(u)\big)\,, \tag{18}$$

where $\tilde{z}_k(u) = (\tilde{z}_k^1(u), \dots, \tilde{z}_k^{p_k}(u))$ is in $\mathbb{R}^{p_k}$, and P_k is a patch extraction operator for finite-dimensional maps. The activation involves a pointwise non-linearity σ along with a quantity $n_k(u) := \|P_k x_{k-1}(u)\|$ in (18), which is due to the homogenization, and which is independent of the filters w_k^i. Finally, the map z_k is obtained by using a pooling operator as in Section 2, with $z_k = A_k \tilde{z}_k$, and $z_0 = x_0$.

Prediction layer. For simplicity, we consider the case of a linear fully connected prediction layer. In this case, the final CNN prediction function f_σ is given by inner products with the feature maps of the last layer:

$$f_\sigma(x_0) = \langle w_{n+1}, z_n \rangle,$$

with parameters w_{n+1} in $L^2(\Omega, \mathbb{R}^{p_n})$. We now show that such a CNN function is contained in the RKHS of the kernel $\mathcal{K}_n$ defined in (7) by considering a linear prediction layer.

Construction in the RKHS. The function f_σ can be constructed recursively by carefully defining functions which lie in the intermediate RKHSs $\mathcal{H}_k$, of the form (17), for appropriate activations σ. Specifically, we define initial quantities f_1^i in $\mathcal{H}_1$ and g_1^i in $\mathcal{P}_1$ for $i = 1, \dots, p_1$ such that

$$\begin{aligned} g_1^i &= w_1^i \in L^2(S_1, \mathbb{R}^{p_0}) = L^2(S_1, \mathcal{H}_0) = \mathcal{P}_1 \\ f_1^i(z) &= \|z\|\sigma(\langle g_i^0, z\rangle / \|z\|) \quad \text{for } z \in \mathcal{P}_1, \end{aligned}$$

and we recursively define, from layer k–1, the quantities f_k^i in $\mathcal{H}_k$ and g_k^i in $\mathcal{P}_k$ for $i = 1, \dots, p_k$:

$$\begin{aligned} g_k^i(v) &= \sum_{j=1}^{p_{k-1}} w_k^{ij}(v) f_{k-1}^j \quad \text{where} \quad w_k^i(v) = (w_k^{ij}(v))_{j=1,\dots,p_{k-1}} \\ f_k^i(z) &= \|z\|\sigma(\langle g_k^i, z\rangle / \|z\|) \quad \text{for } z \in \mathcal{P}_k. \end{aligned}$$

For the linear prediction layer, we define g_σ in $L^2(\Omega, \mathcal{H}_n)$ by:

$$g_\sigma(u) = \sum_{j=1}^{p_n} w_{n+1}^j(u) f_n^j \quad \text{for all } u \in \Omega,$$

so that the function $f : x_0 \mapsto \langle g_\sigma, x_n \rangle$ is in the RKHS of $\mathcal{K}_n$, where x_n is the final representation given in Eq. (5). In Appendix D.2, we show that $f = f_\sigma$, which implies that the CNN function f_σ is in the RKHS. We note that a similar construction for fully connected networks with constraints on weights and inputs was given in [47].

Norm of the CNN f_σ. We now study the RKHS norm of the CNN constructed above. This quantity is important as it controls the stability and invariance of the predictions of a learned model through (1). Additionally, the RKHS norm provides a way to control model complexity, and can lead to generalization bounds, *e.g.*, through Rademacher complexity and margin bounds [7, 38]. In particular, such bounds rely on the following upper bound on the empirical Rademacher complexity of a function class with bounded RKHS norm $\mathcal{F}_\lambda = \{f \in \mathcal{H}_{\mathcal{K}_n} : \|f\| \leq \lambda\}$, for a dataset $\{x^{(1)}, \ldots, x^{(N)}\}$:

$$R_N(\mathcal{F}_\lambda) \leq \frac{\lambda\sqrt{\frac{1}{N}\sum_{i=1}^N \mathcal{K}_n(x^{(i)}, x^{(i)})}}{\sqrt{N}}. \tag{19}$$

The bound remains valid when only considering CNN functions in $\mathcal{F}_\lambda$ of the form f_σ, since such a function class is contained in $\mathcal{F}_\lambda$. We note that various authors have recently considered other norm-based complexity measures to control the generalization of neural networks (see, *e.g.*, [4, 20, 27, 29]). The next proposition (proved in Appendix D.2) characterizes the norm of f_σ in terms of the L^2 norms of the filters w_k^{ij}, and follows from the recursive definition of the intermediate RKHS elements f_k^i.

Proposition 12 (RKHS norm of CNNs)**.** *Assume the activation σ satisfies $C_\sigma(a) < \infty$ for all $a \geq 0$, where C_σ is defined for a given kernel in Lemma 10. Then, the CNN function f_σ defined above is in the RKHS $\mathcal{H}_{\mathcal{K}_n}$, with norm*

$$\|f_\sigma\|^2 \leq p_n \sum_{i=1}^{p_n} \|w_{n+1}^i\|_2^2 B_{n,i},$$

where $B_{n,i}$ is defined recursively by $B_{1,i} = C_\sigma^2(\|w_1^i\|_2^2)$ and $B_{k,i} = C_\sigma^2\left(p_{k-1}\sum_{j=1}^{p_{k-1}} \|w_k^{ij}\|_2^2 B_{k-1,j}\right)$.

Note that this upper bound need not grow exponentially with depth when the filters have small norm and C_σ takes small values around zero. However, the dependency of the bound on the number of feature maps p_k of each layer k may not be satisfactory in situations where the number of parameters is very large, which is common in successful deep learning architectures. The following proposition removes this dependence, relying instead on matrix spectral norms. Similar quantities have been used recently to obtain useful generalization bounds for neural networks [4, 28].

Proposition 13 (RKHS norm of CNNs using spectral norms)**.** *Assume the activation σ satisfies $C_\sigma(a) < \infty$ for all $a \geq 0$, where C_σ is defined for a given kernel in Lemma 10. Then, the CNN function f_σ defined above is in the RKHS $\mathcal{H}_{\mathcal{K}_n}$, with norm*

$$\|f_\sigma\|^2 \leq \|w_{n+1}\|^2\ C_\sigma^2(\|W_n\|_2^2\ C_\sigma^2(\|W_{n-1}\|_2^2 \ldots C_\sigma^2(\|W_2\|_2^2\ C_\sigma^2(\|W_1\|_F^2))\ldots)). \tag{20}$$

The norms are defined as follows:

$$\|W_k\|_2^2 = \int_{S_k} \|W_k(u)\|_2^2 d\nu_k(u), \quad \text{for } k = 2, \ldots, n$$
$$\|W_1\|_F^2 = \int_{S_1} \|W_1(u)\|_F^2 d\nu_1(u),$$

where $W_k(u)$ denotes the matrix $(w_k^{ij}(u))_{ij}$, $\|\cdot\|_2$ the spectral norm, and $\|\cdot\|_F$ the Frobenius norm.

As an example, if we consider $\kappa_1 = \cdots = \kappa_n$ to be one of the kernels introduced in Section 2 and take $\sigma = \kappa_1$ so that $C_\sigma^2 = \kappa_1$, then constraining the norms at each layer to be smaller than 1 ensures $\|f_\sigma\| \leq 1$, since for $\lambda \leq 1$ we have $C_\sigma^2(\lambda^2) \leq C_\sigma^2(1) = \kappa_1(1) = 1$. If we consider linear kernels and $\sigma(u) = u$, we have $C_\sigma^2(\lambda^2) = \lambda^2$ and the bound becomes $\|f_\sigma\| \leq \|w_{n+1}\|\|W_n\|_2 \cdots \|W_2\|_2\|W_1\|_F$. If we ignore the convolutional structure (*i.e.*, only taking 1x1 patches on a 1x1 image), the norm involves a product of spectral norms at each layer (ignoring the first layer), a quantity which also appears in recent generalization bounds [4, 28].

Generalization and stability. The results of this section imply that our study of the geometry of the kernel representations, and in particular the stability and invariance properties of Section 3, apply to the generic CNNs defined above, thanks to the Lipschitz smoothness relation (1). The smoothness is then controlled by the RKHS norm of these functions, which sheds light on the links between generalization and stability. In particular, functions with low RKHS norm (a.k.a. "large margin") are known to generalize better to unseen data (see, *e.g.*, the notion of margin bounds for SVMs [37, 38]). This implies, for instance, that generalization is harder if the task requires classifying two slightly deformed images with different labels, since this requires a function with large RKHS norm according to our stability analysis. In contrast, if a stable function (*i.e.*, with small RKHS norm) is sufficient to do well on a training set, learning becomes "easier" and few samples may be enough for good generalization.

5 Discussion and Concluding Remarks

In this paper, we introduced a multilayer convolutional kernel representation (Section 2), we showed that it is stable to the action of diffeomorphisms and can be made invariant to groups of transformations (Section 3), and finally we explained connections between our representation and generic convolutional networks by showing that certain classes of CNNs with smooth activations are contained in the RKHS of the full multilayer kernel (Section 4). A consequence of this last result is that the stability results of Section 3 apply to any CNN function f from that class, by using the relation

$$|f(L_\tau x) - f(x)| \leq \|f\| \|\Phi_n(L_\tau x) - \Phi_n(x)\|,$$

which follows from (1), assuming a linear prediction layer. However, this stability bound can become much worse as the RKHS norm $\|f\|$ increases, a quantity which is not easy to control in standard CNNs. In fact, our study of the norm of CNNs in Section 4.2 suggests that this norm can vary significantly with different parameters of the model. It has been suggested (see, *e.g.*, [46]) that optimization algorithms may play an important role in controlling the generalization ability of deep networks, and it may be plausible that these impact the RKHS norm of a learned CNN. A better understanding of such implicit regularization behavior would be interesting, but falls beyond the scope of this paper.

In contrast, traditional kernel methods typically control this norm by using it as a regularizer in the learning process, making such a stability guarantee more useful. In order to avoid the scalability issues of such approaches, convolutional kernel networks [21] approximate the full kernel map Φ_n by taking appropriate projections as explained in Section 2.2, leading to a representation $\tilde{\Phi}_n$ that can be represented with a practical representation ψ_n that preserves the Hilbert space structure isometrically (using the finite-dimensional descriptions of points in the RKHS given in (10)). In Section 3.2, we saw that such representations satisfy the same stability and invariance results as the full representation. Then, if we consider a CKN function of the form $f_w(x) = \langle w, \psi_n(x) \rangle$, stability is obtained thanks to the relation

$$|f_w(L_\tau x) - f_w(x)| \leq \|w\| \|\psi_n(L_\tau x) - \psi_n(x)\| = \|w\| \|\tilde{\Phi}_n(L_\tau x) - \tilde{\Phi}_n(x)\|.$$

In particular, learning such a function by controlling the norm of w, *e.g.*, with ℓ_2 regularization as it is done in [21], provides a natural way to explicitly control stability.

A Useful Mathematical Tools

In this section, we present preliminary mathematical tools that are used in our analysis.

Harmonic analysis. We recall a classical result from harmonic analysis (see, *e.g.*, [42]), which was used many times in [23] to prove the stability of the scattering transform to the action of diffeomorphisms.

Lemma A.1 (Schur's test)**.** *Let $\mathcal{H}$ be a Hilbert space and Ω a subset of $\mathbb{R}^d$. Consider T an integral operator with kernel $k : \Omega \times \Omega \to \mathbb{R}$, meaning that for all u in Ω and x in $L^2(\Omega, \mathcal{H})$,*

$$Tx(u) = \int_\Omega k(u, v)x(v)dv, \tag{21}$$

where the integral is a Bochner integral (see, [15, 26]) when $\mathcal{H}$ is infinite-dimensional. If

$$\forall u \in \Omega, \quad \int |k(u, v)|dv \leq C \quad \text{and} \quad \forall v \in \Omega, \quad \int |k(u, v)|du \leq C,$$

for some constant C, then, Tx is always in $L^2(\Omega, \mathcal{H})$ for all x in $L^2(\Omega, \mathcal{H})$ and we have $\|T\| \leq C$.

Note that while the proofs of the lemma above are typically given for real-valued functions in $L^2(\Omega, \mathbb{R})$, the result can easily be extended to Hilbert space-valued functions x in $L^2(\Omega, \mathcal{H})$. In order to prove this, we consider the integral operator $|T|$ with kernel $|k|$ that operates on $L^2(\Omega, \mathbb{R}_+)$, meaning that $|T|$ is defined as in (21) by replacing $k(u, v)$ by the absolute value $|k(u, v)|$. Then, consider x in $L^2(\Omega, \mathcal{H})$ and use the triangle inequality property of Bochner integrals:

$$\|Tx\|^2 = \int_\Omega \|Tx(u)\|^2 du \leq \int_\Omega \left(\int_\Omega |k(u, v)| \|x(v)\| dv \right)^2 du = \||T||x|\|^2,$$

where the function $|x|$ is such that $|x|(u) = \|x(u)\|$ and thus $|x|$ is in $L^2(\Omega, \mathbb{R}_+)$. We may now apply Schur's test to the operator $|T|$ for real-valued functions, which gives $\||T|\| \leq C$. Then, noting that $\||x|\| = \|x\|$, we conclude with the inequality $\|Tx\|^2 \leq \||T||x|\|^2 \leq \||T|\|^2 \|x\|^2 \leq C^2 \|x\|^2$.

The following lemma shows that the pooling operators A_k defined in Section 2 are non-expansive.

Lemma A.2 (Non-expansiveness of pooling operators)**.** *If $h(u) := (2\pi)^{-d/2} \exp(-|u|^2/2)$, then the pooling operator A_σ defined for any $\sigma > 0$ by*

$$A_\sigma x(u) = \int_{\mathbb{R}^d} \sigma^{-d} h\left(\frac{u - v}{\sigma} \right) x(v)dv,$$

has operator norm $\|A_\sigma\| \leq 1$.

Proof. A_σ is an integral operator with kernel $k(u, v) := \sigma^{-d} h((u - v)/\sigma)$. By a change of variables, we have

$$\forall v \in \mathbb{R}^d, \qquad \int_{\mathbb{R}^d} |k(u, v)|du = \int_{\mathbb{R}^d} h(u)du = 1,$$

since h is a standard Gaussian and thus integrates to 1. By symmetry between u and v, we also have $\int_{\mathbb{R}^d} |k(u, v)|dv = 1$ for all u in $\mathbb{R}^d$, and the result follows from Schur's test. □

Kernel methods. We now recall a classical result that characterizes the reproducing kernel Hilbert space (RKHS) of functions defined from explicit Hilbert space mappings (see, *e.g.*, [33, §2.1]).

Theorem A.1. *Let $\psi : \mathcal{X} \to H$ be a feature map to a Hilbert space H, and let $K(z, z') := \langle \psi(z), \psi(z') \rangle_H$ for $z, z' \in \mathcal{X}$. Let $\mathcal{H}$ be the linear subspace defined by*

$$\mathcal{H} := \{ f_w \ ; \ w \in H \} \quad s.t. \quad f_w : z \mapsto \langle w, \psi(z) \rangle_H,$$

and consider the norm

$$\|f_w\|_{\mathcal{H}}^2 := \inf_{w' \in H} \{ \|w'\|_H^2 \quad s.t. \quad f_w = f_{w'} \}.$$

Then $\mathcal{H}$ is the reproducing kernel Hilbert space associated to kernel K.

A consequence of this result is that the RKHS of the kernel $\mathcal{K}_n(x, x') = \langle \Phi(x), \Phi(x') \rangle$, defined from a given final representation $\Phi(x) \in \mathcal{H}_{n+1}$ such as the one introduced in Section 2, contains functions of the form $f : x \mapsto \langle w, \Phi(x) \rangle$ with $w \in \mathcal{H}_{n+1}$, and the RKHS norm of such a function satisfies $\|f\| \leq \|w\|_{\mathcal{H}_{n+1}}$.

B Proofs Related to the Multilayer Kernel Construction

B.1 Proof of Lemma 1 and Non-Expansiveness of the Gaussian Kernel

We begin with the proof of Lemma 1 related to homogeneous dot-product kernels (2).

Proof. In this proof, we drop all indices k since there is no ambiguity.

We will prove the results under the relaxed assumption $\kappa'(u) \leq 1$. Let us consider the Maclaurin expansion $\kappa(u) = \sum_{j=0}^{+\infty} b_j u^j < +\infty$ with $b_j \geq 0$ for all j and all u in $[-1, +1]$. Recall that the condition $b_j \geq 0$ comes from the positive-definiteness of K [34]. Then, it is easy to show that all k-th-order derivatives of κ have the same property. In particular, all of these functions are non-negative, non-decreasing and convex on $[0, 1]$, and they all satisfy the property $|\kappa(u)| \leq \kappa(|u|)$ on $[-1, 1]$, respectively $|\kappa'(u)| \leq \kappa'(|u|)$ on $[-1, 1]$ for κ'.

Consider now the function $f : u \mapsto \kappa(u) - \kappa(1) - \kappa'(1)(u - 1)$; its derivative $f' : u \mapsto \kappa'(u) - \kappa'(1)$ is such that for all u in $[-1, 1]$, $f'(u) = \kappa'(u) - \kappa'(1) \leq \kappa'(|u|) - \kappa'(1) \leq 0$ since κ' is non-decreasing on $[0, 1]$. Therefore f' is non-increasing on $[-1, 1]$ and $f(u) \geq f(1) = 0$ for all u in $[-1, 1]$. Then, if $z, z' \neq 0$,

$$\|\varphi(z) - \varphi(z')\|^2 = K(z, z) + K(z', z') - 2K(z, z') = \|z\|^2 + \|z'\|^2 - 2\|z\|\|z'\|\kappa(u),$$

with $u = \langle z, z' \rangle / (\|z\|\|z'\|)$. Since we have shown that $\kappa(u) \geq \kappa(1) + \kappa'(1)(u - 1)$ for all u in $[-1, 1]$,

$$\begin{aligned}
\|\varphi(z) - \varphi(z')\|^2 &\leq \|z\|^2 + \|z'\|^2 - 2\|z\|\|z'\| \left(1 - \kappa'(1) + \kappa'(1)u\right) \\
&= \left(1 - \kappa'(1)\right) \left(\|z\|^2 + \|z'\|^2 - 2\|z\|\|z'\|\right) + \kappa'(1) \left(\|z\|^2 + \|z'\|^2 - 2\langle z, z' \rangle\right) \\
&= \left(1 - \kappa'(1)\right) \left|\|z\| - \|z'\|\right|^2 + \kappa'(1)\|z - z'\|^2 \\
&\leq \|z - z'\|^2,
\end{aligned}$$

where we used the fact that $0 \leq \kappa'(1) \leq 1$. Note that if we make instead the assumption that $\kappa'(1) > 1$, the same derivation shows that the kernel mapping is Lipschitz with constant $\sqrt{\kappa'(1)}$. Finally, we remark that we have shown the relation $\kappa(u) \geq \kappa(1) - \kappa'(1) + \kappa'(1)u$; when $\kappa'(1) = 1$, this immediately yields (3).

If $z = 0$ or $z' = 0$, the result also holds trivially. For example,

$$\|\varphi(z) - \varphi(0)\|^2 = K(z, z) + K(0, 0) - 2K(z, 0) = \|z\|^2 = \|z - 0\|^2.$$

□

Non-expansiveness of the Gaussian kernel. We now consider the Gaussian (RBF) kernel

$$K(z, z') := e^{-\frac{\alpha}{2}\|z - z'\|^2},$$

with feature map φ. We simply use the convexity inequality $e^u \geq 1 + u$ for all u, and

$$\|\varphi(z) - \varphi(z')\|^2 = K(z, z) + K(z', z') - 2K(z, z') = 2 - 2e^{-\frac{\alpha}{2}\|z - z'\|^2} \leq \alpha\|z - z'\|^2.$$

In particular, φ is non-expansive when $\alpha \leq 1$.

C Proofs of Recovery and Stability Results

C.1 Proof of Lemma 2

Proof. We denote by $\bar{\Omega}$ the discrete set of sampling points considered in this lemma. The assumption on $\bar{\Omega}$ can be written as $\{u + v \ ; \ u \in \bar{\Omega}, v \in S_k\} = \Omega$.

Let B denote an orthonormal basis of the Hilbert space $\mathcal{P}_k = L^2(S_k, \mathcal{H}_{k-1})$, and define the linear function f_w in $\mathcal{H}_k$ such that $f_w : z \mapsto \langle w, z \rangle$ for w in $\mathcal{P}_k$. We thus have

$$\begin{aligned} P_k x_{k-1}(u) &= \sum_{w \in B} \langle w, P_k x_{k-1}(u) \rangle w \\ &= \sum_{w \in B} f_w(P_k x_{k-1}(u)) w \\ &= \sum_{w \in B} \langle f_w, M_k P_k x_{k-1}(u) \rangle w, \end{aligned}$$

using the reproducing property in the RKHS $\mathcal{H}_k$. Applying the pooling operator A_k yields

$$\begin{aligned} A_k P_k x_{k-1}(u) &= \sum_{w \in B} \langle f_w, A_k M_k P_k x_{k-1}(u) \rangle w, \\ &= \sum_{w \in B} \langle f_w, x_k(u) \rangle w. \end{aligned}$$

Noting that $A_k P_k x_{k-1} = A_k (L_v x_{k-1})_{v \in S_k} = (A_k L_v x_{k-1})_{v \in S_k} = (L_v A_k x_{k-1})_{v \in S_k} = P_k A_k x_{k-1}$, with $L_v x_{k-1}(u) := x_{k-1}(u + v)$, we can choose v in S_k and obtain from the previous relations

$$A_k x_{k-1}(u + v) = \sum_{w \in B} \langle f_w, x_k(u) \rangle w(v).$$

Thus, taking all sampling points $u \in \bar{\Omega}$ and all $v \in S_k$, we have a full view of the signal $A_k x_{k-1}$ on all of Ω by our assumption on the set $\bar{\Omega}$.

For $f \in \mathcal{H}_{k-1}$, the signal $\langle f, x_{k-1}(u) \rangle$ can then be recovered by deconvolution as follows:

$$\langle f, x_{k-1}(u) \rangle = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(\langle f, A_k x_{k-1}(\cdot) \rangle)}{\mathcal{F}(h_{\sigma_k})} \right)(u),$$

where $\mathcal{F}$ denotes the Fourier transform. Note that the inverse Fourier transform is well-defined here because the signal $\langle f, A_k x_k(\cdot) \rangle$ is itself a convolution with h_{σ_k}, and $\mathcal{F}(h_{\sigma_k})$ is strictly positive as the Fourier transform of a Gaussian is also a Gaussian.

By considering all elements f in an orthonormal basis of $\mathcal{H}_{k-1}$, we can recover x_{k-1}. The map x_k can then be reconstructed trivially by applying operators P_k, M_k and A_k on x_{k-1}.

□

C.2 Proof of Lemma 3

Proof. In this proof, we drop the bar notation on all quantities for simplicity; there is indeed no ambiguity since all signals are discrete here. First, we recall that $\mathcal{H}_k$ contains all linear functions on $\mathcal{P}_k = \mathcal{H}_{k-1}^{e_k}$; thus, we may consider in particular functions $f_{j,w}(z) := e_k^{1/2} \langle w, z_j \rangle$ for $j \in \{1, \dots, e_k\}$, $w \in \mathcal{H}_{k-1}$, and

$z = (z_1, z_2, \dots, z_{e_k})$ in $\mathcal{P}_k$. Then, we may evaluate

$$\begin{aligned}
\langle f_{j,w}, s_k^{-1/2} x_k[n] \rangle &= \sum_{m \in \mathbb{Z}} h_k[ns_k - m] \langle f_{j,w}, M_k P_k x_{k-1}[m] \rangle \\
&= \sum_{m \in \mathbb{Z}} h_k[ns_k - m] \langle f_{j,w}, \varphi_k(P_k x_{k-1}[m]) \rangle \\
&= \sum_{m \in \mathbb{Z}} h_k[ns_k - m] f_{j,w}(P_k x_{k-1}[m]) \\
&= \sum_{m \in \mathbb{Z}} h_k[ns_k - m] \langle w, x_{k-1}[m + j] \rangle \\
&= \sum_{m \in \mathbb{Z}} h_k[ns_k + j - m] \langle w, x_{k-1}[m] \rangle \\
&= (h_k * \langle w, x_{k-1} \rangle)[ns_k + j],
\end{aligned}$$

where, with an abuse of notation, $\langle w, x_{k-1} \rangle$ is the real-valued discrete signal such that $\langle w, x_{k-1} \rangle[n] = \langle w, x_{k-1}[n] \rangle$. Since integers of the form $(ns_k + j)$ cover all of $\mathbb{Z}$ according to the assumption $e_k \geq s_k$, we have a full view of the signal $(h_k * \langle w, x_{k-1} \rangle)$ on $\mathbb{Z}$. We will now follow the same reasoning as in the proof of Lemma 2 to recover $\langle w, x_{k-1} \rangle$:

$$\langle w, x_{k-1} \rangle = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(h_k * \langle w, x_{k-1} \rangle)}{\mathcal{F}(h_k)} \right),$$

where $\mathcal{F}$ is the Fourier transform. Since the signals involved there are discrete, their Fourier transform are periodic with period 2π, and we note that $\mathcal{F}(h_k)$ is strictly positive and bounded away from zero. The signal x_{k-1} is then recovered exactly as in the proof of Lemma 2 by considering for w the elements of an orthonormal basis of the Hilbert space $\mathcal{H}_{k-1}$. □

C.3 Proof of Proposition 4

Proof. Define $(MPA)_{k:j} := M_k P_k A_{k-1} M_{k-1} P_{k-1} A_{k-2} \cdots M_j P_j A_{j-1}$. Using the fact that $\|A_k\| \leq 1$, $\|P_k\| = 1$ and M_k is non-expansive, we obtain

$$\begin{aligned}
\|\Phi_n(L_\tau x) - \Phi_n(x)\| &= \|A_n (MPA)_{n:2} M_1 P_1 A_0 L_\tau x - A_n (MPA)_{n:2} M_1 P_1 A_0 x\| \\
&\leq \|A_n (MPA)_{n:2} M_1 P_1 A_0 L_\tau x - A_n (MPA)_{n:2} M_1 L_\tau P_1 A_0 x\| \\
&\quad + \|A_n (MPA)_{n:2} M_1 L_\tau P_1 A_0 x - A_n (MPA)_{n:2} M_1 P_1 A_0 x\| \\
&\leq \|[P_1 A_0, L_\tau]\| \|x\| \\
&\quad + \|A_n (MPA)_{n:2} M_1 L_\tau P_1 A_0 x - A_n (MPA)_{n:2} M_1 P_1 A_0 x\|.
\end{aligned}$$

Note that M_1 is defined point-wise, and thus commutes with L_τ:

$$M_1 L_\tau x(u) = \varphi_1(L_\tau x(u)) = \varphi_1(x(u - \tau(u)) = M_1 x(u - \tau(u)) = L_\tau M_1 x(u).$$

By noticing that $\|M_1 P_1 A_0 x\| \leq \|x\|$, we can expand the second term above in the same way. Repeating this by induction yields

$$\begin{aligned}
\|\Phi_n(L_\tau x) - \Phi_n(x)\| &\leq \sum_{k=1}^{n} \|[P_k A_{k-1}, L_\tau]\| \|x\| + \|A_n L_\tau (MPA)_{n:1} x - A_n (MPA)_{n:1} x\| \\
&\leq \sum_{k=1}^{n} \|[P_k A_{k-1}, L_\tau]\| \|x\| + \|A_n L_\tau - A_n\| \|x\|,
\end{aligned}$$

and the result follows by decomposing $A_n L_\tau = [A_n, L_\tau] + L_\tau A_n$ using the triangle inequality. □

C.4 Proof of Lemma 5

Proof. The proof follows in large parts the methodology introduced by Mallat [23] in the analysis of the stability of the scattering transform. More precisely, we will follow in part the proof of Lemma E.1 of [23]. The kernel (in the sense of Lemma A.1) of A_σ is $h_\sigma(z-u) = \sigma^{-d} h(\frac{z-u}{\sigma})$. Throughout the proof, we will use the following bounds on the decay of h for simplicity, as in [23]:[1]

$$|h(u)| \leq \frac{C_h}{(1+|u|)^{d+2}}$$
$$|\nabla h(u)| \leq \frac{C'_h}{(1+|u|)^{d+2}},$$

which are satisfied for the Gaussian function h thanks to its exponential decay.

We now decompose the commutator

$$[L_c A_\sigma, L_\tau] = L_c A_\sigma L_\tau - L_\tau L_c A_\sigma = L_c (A_\sigma - L_c^{-1} L_\tau L_c A_\sigma L_\tau^{-1}) L_\tau = L_c T L_\tau,$$

with $T := A_\sigma - L_c^{-1} L_\tau L_c A_\sigma L_\tau^{-1}$. Hence,

$$\|[L_c A_\sigma, L_\tau]\| \leq \|L_c\| \|L_\tau\| \|T\|.$$

We have $\|L_c\| = 1$ since the translation operator L_c preserves the norm. Note that we have

$$2^{-d} \leq (1 - \|\nabla\tau\|_\infty)^d \leq \det(I - \nabla\tau(u)) \leq (1 + \|\nabla\tau\|_\infty)^d \leq 2^d, \tag{22}$$

for all $u \in \Omega$. Thus, for $f \in L^2(\Omega)$,

$$\begin{aligned}\|L_\tau f\|^2 &= \int_\Omega |f(z-\tau(z))|^2 dz = \int_\Omega |f(u)|^2 \det(I - \nabla\tau(u))^{-1} du \\ &\leq (1 - \|\nabla\tau\|_\infty)^{-d} \|f\|^2,\end{aligned}$$

such that $\|L_\tau\| \leq (1 - \|\nabla\tau\|_\infty)^{-d/2} \leq 2^{d/2}$. This yields

$$\|[L_c A_\sigma, L_\tau]\| \leq 2^{d/2} \|T\|.$$

Kernel of T. We now show that T is an integral operator and describe its kernel. Let $\xi = (I - \tau)^{-1}$, so that $L_\tau^{-1} f(z) = f(\xi(z))$ for any function f in $L^2(\Omega)$. We have

$$\begin{aligned}A_\sigma L_\tau^{-1} f(z) &= \int h_\sigma(z-v) f(\xi(v)) dv \\ &= \int h_\sigma(z - u + \tau(u)) f(u) \det(I - \nabla\tau(u)), du\end{aligned}$$

using the change of variable $v = u - \tau(u)$, giving $\left|\frac{dv}{du}\right| = \det(I - \nabla\tau(u))$. Then note that $L_c^{-1} L_\tau L_c f(z) = L_\tau L_c f(z+c) = L_c f(z + c - \tau(z+c)) = f(z - \tau(z+c))$. This yields the following kernel for the operator T:

$$k(z,u) = h_\sigma(z-u) - h_\sigma(z - \tau(z+c) - u + \tau(u)) \det(I - \nabla\tau(u)). \tag{23}$$

A similar operator appears in Lemma E.1 of [23], whose kernel is identical to (23) when $c = 0$.

As in [23], we decompose $T = T_1 + T_2$, with kernels

$$\begin{aligned}k_1(z,u) &= h_\sigma(z-u) - h_\sigma((I - \nabla\tau(u))(z-u)) \det(I - \nabla\tau(u)) \\ k_2(z,u) &= \det(I - \nabla\tau(u)) \left(h_\sigma((I - \nabla\tau(u))(z-u)) - h_\sigma(z - \tau(z+c) - u + \tau(u))\right).\end{aligned}$$

The kernel $k_1(z,u)$ appears in [23], whereas the kernel $k_2(z,u)$ involves a shift c which is not present in [23]. For completeness, we include the proof of the bound for both operators, even though only dealing with k_2 requires slightly new developments.

[1] Note that a more precise analysis may be obtained by using finer decay bounds.

Bound on $\|T_1\|$. We can write $k_1(z,u) = \sigma^{-d} g(u, (z-u)/\sigma)$ with

$$\begin{aligned} g(u,v) &= h(v) - h((I - \nabla\tau(u))v)\det(I - \nabla\tau(u)) \\ &= (1 - \det(I - \nabla\tau(u)))h((I - \nabla\tau(u))v) + h(v) - h((I - \nabla\tau(u))v). \end{aligned}$$

Using the fundamental theorem of calculus on h, we have

$$h(v) - h((I - \nabla\tau(u))v) = \int_0^1 \langle \nabla h((I + (t-1)\nabla\tau(u))v), \nabla\tau(u)v\rangle dt.$$

Noticing that

$$|(I + (t-1)\nabla\tau(u))v| \geq (1 - \|\nabla\tau\|_\infty)|v| \geq (1/2)|v|,$$

and that $\det(I - \nabla\tau(u))) \geq (1 - \|\nabla\tau\|_\infty)^d \geq 1 - d\|\nabla\tau\|_\infty$, we bound each term as follows

$$\begin{aligned} |(1 - \det(I - \nabla\tau(u)))h((I - \nabla\tau(u))v)| &\leq d\|\nabla\tau\|_\infty \frac{C_h}{(1 + \frac{1}{2}|v|)^{d+2}} \\ \left| \int_0^1 \langle \nabla h((I + (t-1)\nabla\tau(u))v), \nabla\tau(u)v\rangle dt \right| &\leq \|\nabla\tau\|_\infty \frac{C_h'|v|}{(1 + \frac{1}{2}|v|)^{d+2}}. \end{aligned}$$

We thus have

$$|g(u,v)| \leq \|\nabla\tau\|_\infty \frac{C_h d + C_h'|v|}{(1 + \frac{1}{2}|v|)^{d+2}}.$$

Using appropriate changes of variables in order to bound $\int |k_1(z,u)|du$ and $\int |k_1(z,u)|dz$, Schur's test yields

$$\|T_1\| \leq C_1 \|\nabla\tau\|_\infty, \tag{24}$$

with

$$C_1 = \int_\Omega \frac{C_h d + C_h'|v|}{(1 + \frac{1}{2}|v|)^{d+2}} dv$$

Bound on $\|T_2\|$. Let $\alpha(z,u) = \tau(z+c) - \tau(u) - \nabla\tau(u)(z-u)$, and note that we have

$$\begin{aligned} |\alpha(z,u)| &\leq |\tau(z+c) - \tau(u)| + |\nabla\tau(u)(z-u)| \\ &\leq \|\nabla\tau\|_\infty |z + c - u| + \|\nabla\tau\|_\infty |z - u| \\ &\leq \|\nabla\tau\|_\infty (|c| + 2|z-u|). \end{aligned} \tag{25}$$

The fundamental theorem of calculus yields

$$k_2(z,u) = -\det(I - \nabla\tau(u)) \int_0^1 \langle \nabla h_\sigma(z - \tau(z+c) - u + \tau(u) - t\alpha(z,u)), \alpha(z,u)\rangle dt.$$

We note that $|\det(I - \nabla\tau(u))| \leq 2^d$, and $\nabla h_\sigma(v) = \sigma^{-d-1}\nabla h(v/\sigma)$. Using the change of variable $z' = (z-u)/\sigma$, we obtain

$$\int |k_2(z,u)|dz \quad \leq \quad 2^d \int \int_0^1 \left| \nabla h \left(z' + \frac{\tau(u + \sigma z' + c) - \tau(u) - t\alpha(u + \sigma z', u)}{\sigma} \right) \right| \left| \frac{\alpha(u + \sigma z', u)}{\sigma} \right| dt dz'.$$

We can use the upper bound (25), together with our assumption $|c| \leq \kappa\sigma$:

$$\left| \frac{\alpha(u + \sigma z', u)}{\sigma} \right| \leq \|\nabla\tau\|_\infty (\kappa + 2|z'|). \tag{26}$$

Separately, we have $|\nabla h(v(z'))| \leq C'_h/(1+|v(z')|)^{d+2}$, with

$$v(z') := z' + \frac{\tau(u+\sigma z'+c) - \tau(u) - t\alpha(u+\sigma z', u)}{\sigma}.$$

For $|z'| > 2\kappa$, we have

$$\begin{aligned}\left|\frac{\tau(u+\sigma z'+c) - \tau(u) - t\alpha(u+\sigma z', u)}{\sigma}\right| &= \left|t\nabla\tau(u)z' + (1-t)\frac{\tau(u+\sigma z'+c)-\tau(u)}{\sigma}\right| \\ &\leq t\|\nabla\tau\|_\infty|z'| + (1-t)\|\nabla\tau\|_\infty(|z'|+\kappa) \\ &\leq \frac{3}{2}\|\nabla\tau\|_\infty|z'| \leq \frac{3}{4}|z'|,\end{aligned}$$

and hence, using the reverse triangle inequality, $|v(z')| \geq |z'| - \frac{3}{4}|z'| = \frac{1}{4}|z'|$. This yields the upper bound

$$|\nabla h(v(z'))| \leq \begin{cases} C'_h, & \text{if } |z'| \leq 2\kappa \\ \frac{C'_h}{(1+\frac{1}{4}|z'|)^{d+2}}, & \text{if } |z'| > 2\kappa. \end{cases} \tag{27}$$

Combining these two bounds, we obtain

$$\int |k_2(z,u)|dz \leq C_2\|\nabla\tau\|_\infty,$$

with

$$C_2 := 2^d C'_h \left(\int_{|z'|<2\kappa}(\kappa + 2|z'|)dz' + \int_{|z'|>2\kappa}\frac{\kappa+2|z'|}{(1+\frac{1}{4}|z'|)^{d+2}}dz'\right).$$

Note that the dependence of the first integral on κ is of order k^{d+1}. Following the same steps with the change of variable $u' = (z-u)/\sigma$, we obtain the bound $\int |k_2(z,u)|du \leq C_2\|\nabla\tau\|_\infty$. Schur's test then yields

$$\|T_2\| \leq C_2\|\nabla\tau\|_\infty. \tag{28}$$

We have thus proven

$$\|[L_c A_\sigma, L_\tau]\| \leq 2^{d/2}\|T\| \leq 2^{d/2}(C_1+C_2)\|\nabla\tau\|_\infty.$$

□

C.5 Proof of Lemma 8

Proof. The Gaussian case is trivial since the Gaussian kernel mapping φ_{n+1} maps all points to the sphere. In the linear case, we have

$$\begin{aligned}\|\Phi(x)\|^2 = \|\Phi_n(x)\|^2 &= \|A_n M_n P_n x_{n-1}\|^2 \\ &= \int \|A_n M_n P_n x_{n-1}(u)\|^2 du \\ &= \int \langle \int h_{\sigma_n}(u-v) M_n P_n x_{n-1}(v)dv, \int h_{\sigma_n}(u-v') M_n P_n x_{n-1}(v')dv'\rangle du \\ &= \int\int\int h_{\sigma_n}(u-v)h_{\sigma_n}(u-v')\langle\varphi_n(P_n x_{n-1}(v)), \varphi_n(P_n x_{n-1}(v'))\rangle dvdv'du \\ &\geq \int\int\int h_{\sigma_n}(u-v)h_{\sigma_n}(u-v')\langle P_n x_{n-1}(v), P_n x_{n-1}(v')\rangle dvdv'du \\ &= \int \|A_n P_n x_{n-1}(u)\|^2 du = \|A_n P_n x_{n-1}\|^2,\end{aligned}$$

where the inequality follows from $\langle \varphi_n(z), \varphi_n(z') \rangle = K_n(z, z') \geq \langle z, z' \rangle$ (see Lemma 1). Using Fubini's theorem and the fact that A_n commutes with translations, we have

$$\|A_n P_n x_{n-1}\|^2 = \int_{S_n} \|A_n L_v x_{n-1}\|^2 d\nu_n(v) = \int_{S_n} \|L_v A_n x_{k-1}\|^2 d\nu_n(v) = \int_{S_n} \|A_n x_{k-1}\|^2 d\nu_n(v) = \|A_n x_{n-1}\|^2,$$

where we used the fact that translations L_v preserve the norm. Note that we have

$$A_n x_{n-1} = A_n A_{n-1} M_{n-1} P_{n-1} x_{n-2} = A_{n,n-1} M_{n-1} P_{n-1} x_{n-2},$$

where $A_{n,n-1}$ is an integral operator with positive kernel $h_{\sigma_n} * h_{\sigma_{n-1}}$. Repeating the above relation then yields

$$\|\Phi(x)\|^2 \geq \|A_n x_{n-1}\|^2 \geq \|A_n A_{n-1} x_{n-1}\|^2 \geq \ldots \geq \|A_n A_{n-1} \ldots A_0 x\|^2,$$

and the result follows. □

C.6 Proof of Lemma 9

Proof. By our assumptions on φ_{n+1} and on the operators A_k, M_k, P_k, we have that Φ is non-expansive, so that

$$\sup_{x, x' \in L^2(\Omega, \mathcal{H}_0)} \frac{\|\Phi(x) - \Phi(x')\|}{\|x - x'\|} \leq 1.$$

It then suffices to show that one can find x, x' such that the norm ratio $\frac{\|\Phi(x) - \Phi(x')\|}{\|x - x'\|}$ is arbitrarily close to 1. In particular, we begin by showing that for any signal $x \neq 0$ we have

$$\lim_{\lambda \to 1} \frac{\|\Phi(\lambda x) - \Phi(x)\|}{\|\lambda x - x\|} \geq \frac{\|A_\sigma x\|}{\|x\|}, \tag{29}$$

where A_σ is the pooling operator with scale $\sigma = (\sigma_n^2 + \sigma_{n-1}^2 + \ldots + \sigma_1^2)^{1/2}$, and the result will follow by considering appropriate signals x that make this lower bound arbitrarily close to 1.

Note that by homogeneity of the kernels maps φ_k (which follow from the homogeneity of kernels K_k), and by linearity of the operators A_k and P_k, we have $\Phi_n(\lambda x) = \lambda \Phi_n(x)$ for any $\lambda \geq 0$. Taking $\lambda > 0$, we have

$$\|\Phi_n(\lambda x) - \Phi_n(x)\| = (\lambda - 1)\|\Phi_n(x)\| \geq (\lambda - 1)\|A_n A_{n-1} \ldots A_0 x\| = (\lambda - 1)\|A_\sigma x\|,$$

adapting Lemma 8 to the representation Φ_n. Thus,

$$\lim_{\lambda \to 1} \frac{\|\Phi_n(\lambda x) - \Phi_n(x)\|}{\|\lambda x - x\|} \geq \frac{\|A_\sigma x\|}{\|x\|}.$$

When φ_{n+1} is linear, we immediately obtain (29) since $\|\Phi(\lambda x) - \Phi(x)\| = \|\Phi_n(\lambda x) - \Phi_n(x)\|$. For the Gaussian case, we have

$$\begin{aligned}
\|\Phi(\lambda x) - \Phi(x)\|^2 &= 2 - 2e^{-\frac{1}{2}\|\Phi_n(\lambda x) - \Phi_n(x)\|^2} \\
&= 2 - 2e^{-\frac{1}{2}(\lambda - 1)^2 \|\Phi_n(x)\|^2} \\
&= (\lambda - 1)^2 \|\Phi_n(x)\|^2 + o((\lambda - 1)^2) \\
&= \|\Phi_n(\lambda x) - \Phi_n(x)\|^2 + o((\lambda - 1)^2),
\end{aligned}$$

which yields (29).

By considering a Gaussian signal with scale $\tau \gg \sigma$, we can make $\frac{\|A_\sigma x\|}{\|x\|}$ arbitrarily close to 1 by taking an arbitrarily large τ. It follows that

$$\sup_x \lim_{\lambda \to 1} \frac{\|\Phi(\lambda x) - \Phi(x)\|}{\|\lambda x - x\|} = 1,$$

which yields the result. □

D Proofs Related to the Construction of CNNs in the RKHS

D.1 Proof of Lemma 10

Proof. Here, we drop all indices k since there is no ambiguity. We will now characterize the functional space $\mathcal{H}$ by following the same strategy as [47, 48] for the non-homogeneous Gaussian and inverse polynomial kernels on Euclidean spaces. Using the Maclaurin expansion of κ, we can define the following explicit feature map for the dot-product kernel $K_{\text{dp}}(z, z') := \kappa(\langle z, z' \rangle)$, for any z in the unit-ball of $\mathcal{P}$:

$$\begin{aligned} \psi_{\text{dp}}(z) &= \left(\sqrt{b_0}, \sqrt{b_1} z, \sqrt{b_2} z \otimes z, \sqrt{b_3} z \otimes z \otimes z, \dots \right) \\ &= \left(\sqrt{b_j} z^{\otimes j} \right)_{j \in \mathbb{N}}, \end{aligned} \tag{30}$$

where $z^{\otimes j}$ denotes the tensor product of order j of the vector z. Technically, the explicit mapping lives in the Hilbert space $\oplus_{j=0}^{n} \otimes^j \mathcal{P}$, where $\oplus$ denotes the direct sum of Hilbert spaces, and with the abuse of notation that $\otimes^0 \mathcal{P}$ is simply $\mathbb{R}$. Then, we have that $K_{\text{dp}}(z, z') = \langle \psi(z), \psi(z') \rangle$ for all z, z' in the unit ball of $\mathcal{P}$. Similarly, we can construct an explicit feature map for the homogeneous dot-product kernels (2):

$$\begin{aligned} \psi_{\text{hdp}}(z) &= \left(\sqrt{b_0} \|z\|, \sqrt{b_1} z, \sqrt{b_2} \|z\|^{-1} z \otimes z, \sqrt{b_3} \|z\|^{-2} z \otimes z \otimes z, \dots \right) \\ &= \left(\sqrt{b_j} \|z\|^{1-j} z^{\otimes j} \right)_{j \in \mathbb{N}}. \end{aligned} \tag{31}$$

From these mappings, we may now conclude the proof by following the same strategy as [47, 48]. By first considering the restriction of K to unit-norm vectors z,

$$\sigma(\langle w, z \rangle) = \sum_{j=0}^{+\infty} a_j \langle w, z \rangle^j = \sum_{j=0}^{+\infty} a_j \langle w^{\otimes j}, z^{\otimes j} \rangle = \langle \bar{w}, \psi(z) \rangle,$$

where

$$\bar{w} = \left(\frac{a_j}{\sqrt{b_j}} w^{\otimes j} \right)_{j \in \mathbb{N}}.$$

Then, the norm of $\bar{w}$ is

$$\|\bar{w}\|^2 = \sum_{j=0}^{+\infty} \frac{a_j^2}{b_j} \|w^{\otimes j}\|^2 = \sum_{j=0}^{+\infty} \frac{a_j^2}{b_j} \|w\|^{2j} = C_\sigma^2(\|w\|^2) < +\infty.$$

Using Theorem A.1, we conclude that f is in the RKHS of K, with norm $\|f\| \leq C_\sigma(\|w\|^2)$. Finally, we extend the result to non unit-norm vectors z with similar calculations and we obtain the desired result. □

D.2 CNN construction and RKHS norm

In this section, we describe the space of functions (RKHS) $\mathcal{H}_{\mathcal{K}_n}$ associated to the kernel $\mathcal{K}_n(x_0, x_0') = \langle x_n, x_n' \rangle$ defined in (7), where x_n, x_n' are the final representations given by Eq. (5), in particular showing it contains the set of CNNs with activations described in Section 4.1.

D.2.1 Construction of a CNN in the RKHS.

Let us consider the definition of the CNN presented in Section 4. We will show that it can be seen as a point in the RKHS of $\mathcal{K}_n$. According to Lemma 10, we consider $\mathcal{H}_k$ that contains all functions of the form $z \in \mathcal{P}_k \mapsto \|z\| \sigma(\langle w, z \rangle / \|z\|)$, with $w \in \mathcal{P}_k$.

We recall the intermediate quantities introduced in Section 4. That is, we define the initial quantities $f_1^i \in \mathcal{H}_1, g_1^i \in \mathcal{P}_1$ for $i = 1, \dots, p_1$ such that

$$\begin{aligned} g_1^i &= w_1^i \in L^2(S_1, \mathbb{R}^{p_0}) = L^2(S_1, \mathcal{H}_0) = \mathcal{P}_1 \\ f_1^i(z) &= \|z\|\sigma(\langle g_i^0, z\rangle/\|z\|) \quad \text{for } z \in \mathcal{P}_1, \end{aligned}$$

and we recursively define, from layer k–1, the quantities $f_k^i \in \mathcal{H}_k, g_k^i \in \mathcal{P}_k$ for $i = 1, \dots, p_k$:

$$\begin{aligned} g_k^i(v) &= \sum_{j=1}^{p_{k-1}} w_k^{ij}(v) f_{k-1}^j \quad \text{where} \quad w_k^i(v) = (w_k^{ij}(v))_{j=1,\dots,p_{k-1}} \\ f_k^i(z) &= \|z\|\sigma(\langle g_k^i, z\rangle/\|z\|) \quad \text{for } z \in \mathcal{P}_k. \end{aligned}$$

Then, we will show that $\tilde{z}_k^i(u) = f_k^i(P_k x_{k-1}(u)) = \langle f_k^i, M_k P_k x_{k-1}(u)\rangle$, which correspond to feature maps at layer k and index i in a CNN. Indeed, this is easy to see for $k = 1$ by construction with filters $w_1^i(v)$, and for $k \geq 2$, we have

$$\begin{aligned} \tilde{z}_k^i(u) &= n_k(u)\sigma\big(\langle w_k^i, P_k z_{k-1}(u)\rangle/n_k(u)\big) \\ &= n_k(u)\sigma\big(\langle w_k^i, P_k A_{k-1}\tilde{z}_{k-1}(u)\rangle/n_k(u)\big) \\ &= n_k(u)\sigma\left(\frac{1}{n_k(u)}\sum_{j=1}^{p_{k-1}}\int_{S_k} w_k^{ij}(v) A_{k-1}\tilde{z}_{k-1}^j(u+v)d\nu_k(v)\right) \\ &= n_k(u)\sigma\left(\frac{1}{n_k(u)}\sum_{j=1}^{p_{k-1}}\int_{S_k} w_k^{ij}(v)\langle f_{k-1}^j, A_{k-1}M_{k-1}P_{k-1}x_{k-2}(u+v)\rangle d\nu_k(v)\right) \\ &= n_k(u)\sigma\left(\frac{1}{n_k(u)}\int_{S_k}\langle g_k^i(v), A_{k-1}M_{k-1}P_{k-1}x_{k-2}(u+v)\rangle d\nu_k(v)\right) \\ &= n_k(u)\sigma\left(\frac{1}{n_k(u)}\int_{S_k}\langle g_k^i(v), x_{k-1}(u+v)\rangle d\nu_k(v)\right) \\ &= n_k(u)\sigma\left(\frac{1}{n_k(u)}\langle g_k^i(v), P_k x_{k-1}(u)\rangle\right) \\ &= f_k^i(P_k x_{k-1}(u)), \end{aligned}$$

where $n_k(u) := \|P_k x_{k-1}(u)\|$. Note that we have used many times the fact that A_k operates on each channel independently when applied to a finite-dimensional map.

The final prediction function is of the form $f_\sigma(x_0) = \langle w_{n+1}, z_n\rangle$ with w_{n+1} in $L^2(\Omega, \mathbb{R}^{p_n})$. Then, we can define the following function g_σ in $L^2(\Omega, \mathcal{H}_n)$ such that

$$g_\sigma(u) = \sum_{j=1}^{p_n} w_{n+1}^j(u) f_n^j,$$

which yields

$$\begin{aligned}
\langle g_\sigma, x_n \rangle &= \sum_{j=1}^{p_n} \int_\Omega w_{n+1}^j(u) \langle f_n^j, x_n(u) \rangle du \\
&= \sum_{j=1}^{p_n} \int_\Omega w_{n+1}^j(u) \langle f_n^j, A_n M_n P_n x_{n-1}(u) \rangle du \\
&= \sum_{j=1}^{p_n} \int_\Omega w_{n+1}^j(u) A_n \tilde{z}_n^j(u) du \\
&= \sum_{j=1}^{p_n} \int_\Omega w_{n+1}^j(u) z_n^j(u) du \\
&= \sum_{j=1}^{p_n} \langle w_{n+1}^j, z_n^j \rangle = f_\sigma(x_0),
\end{aligned}$$

which corresponds to a linear layer after pooling. Since the RKHS of $\mathcal{K}_n$ in the linear case (7) contains all functions of the form $f(x_0) = \langle g, x_n \rangle$, for g in $L^2(\Omega, \mathcal{H}_n)$, we have that f_σ is in the RKHS.

D.2.2 Proof of Proposition 12

Proof. As shown in Lemma 10, the RKHS norm of a function $f : z \in \mathcal{P}_k \mapsto \|z\| \sigma(\langle w, z \rangle / \|z\|)$ in $\mathcal{H}_k$ is bounded by $C_\sigma(\|w\|^2)$, where C_σ depends on the activation σ. We then have

$$\begin{aligned}
&\|f_1^i\|^2 \leq C_\sigma^2(\|w_1^i\|_2^2) \quad \text{where} \quad \|w_1^i\|_2^2 = \int_{S_1} \|w_1^i(v)\|^2 d\nu_1(v) \\
&\|f_k^i\|^2 \leq C_\sigma^2(\|g_k^i\|^2) \\
&\|g_k^i\|^2 = \int_{S_k} \| \sum_{j=1}^{p_{k-1}} w_k^{ij}(v) f_{k-1}^j \|^2 d\nu_k(v) \\
&\qquad \leq p_{k-1} \sum_{j=1}^{p_{k-1}} \left(\int_{S_k} |w_k^{ij}(v)|^2 d\nu_k(v) \right) \|f_{k-1}^j\|^2 \\
&\qquad = p_{k-1} \sum_{j=1}^{p_{k-1}} \|w_k^{ij}\|_2^2 \|f_{k-1}^j\|^2,
\end{aligned}$$

where in the last inequality we use $\|a_1 + \ldots + a_n\|^2 \leq n(\|a_1\|^2 + \ldots + \|a_n\|^2)$. Since C_σ^2 is monotonically increasing (typically exponentially in its argument), we have for $k = 1, \ldots, n-1$ the recursive relation

$$\|f_k^i\|^2 \leq C_\sigma^2 \left(p_{k-1} \sum_{j=1}^{p_{k-1}} \|w_k^{ij}\|_2^2 \|f_{k-1}^j\|^2 \right).$$

The norm of the final prediction function $f \in L^2(\Omega, \mathcal{H}_n)$ is bounded as follows, using similar arguments as well as Theorem A.1:

$$\|f_\sigma\|^2 \leq \|g_\sigma\|^2 \leq p_n \sum_{j=1}^{p_n} \left(\int_\Omega |w_{n+1}^j(u)|^2 du \right) \|f_n^j\|^2.$$

This yields the desired result. □

D.2.3 Proof of Proposition 13

Proof. Define

$$F_k = (f_k^1, \dots, f_k^{p_k}) \in \mathcal{H}_k^{p_k}$$
$$G_k = (g_k^1, \dots, g_k^{p_k}) \in \mathcal{P}_k^{p_k}$$
$$W_k(u) = (w_k^{ij}(u))_{ij} \in \mathbb{R}^{p_k \times p_{k-1}} \quad \text{for } u \in S_k.$$

We will write, by abuse of notation, $G_k(u) = (g_k^1(u), \dots, g_k^{p_k}(u))$ for $u \in S_k$, so that we can write $G_k(u) = W_k(u)F_{k-1}$. In particular, we have $\|G_k(u)\| \leq \|W_k(u)\|_2 \|F_{k-1}\|$. This can be seen by considering an orthonormal basis B of $\mathcal{H}_k$, and defining real-valued vectors $F_k^w = (\langle w, f_k^1 \rangle, \dots, \langle w, f_k^{p_k} \rangle)$, $G_k^w(u) = (\langle w, g_k^1(u) \rangle, \dots, \langle w, g_k^{p_k}(u) \rangle)$ for $w \in B$. Indeed, we have $G_k^w(u) = W_k(u)F_{k-1}^w$ and hence $\|G_k^w(u)\| \leq \|W_k(u)\|_2 \|F_{k-1}^w\|$ for all $w \in B$, and we conclude using

$$\|G_k(u)\|^2 = \sum_{w \in B} \|G_k^w(u)\|^2 \leq \|W_k(u)\|_2^2 \sum_{w \in B} \|F_{k-1}^w\|^2 = \|W_k(u)\|_2^2 \, \|F_{k-1}\|^2.$$

Then, we have

$$\|G_k\|^2 = \sum_i \|g_k^i\|^2 = \sum_i \int_{S_k} \|g_k^i(u)\|^2 d\nu_k(u) = \int_{S_k} \|G_k(u)\|^2 d\nu_k(u)$$
$$\leq \int_{S_k} \|W_k(u)\|_2^2 \, \|F_{k-1}\|^2 \nu_k(u) = \|W_k\|_2^2 \, \|F_{k-1}\|^2.$$

Separately, we notice that C_σ^2 is super-additive, *i.e.*,

$$C_\sigma^2(\lambda_1^2 + \ldots + \lambda_n^2) \geq C_\sigma^2(\lambda_1^2) + \ldots + C_\sigma^2(\lambda_n^2).$$

Indeed, this follows from the definition of C_σ^2, noting that polynomials with non-negative coefficients are super-additive on non-negative numbers. Thus, we have

$$\|F_1\|^2 = \sum_{i=1}^{p_1} \|f_1^i\|^2 \leq \sum_{i=1}^{p_1} C_\sigma^2(\|w_1^i\|^2) \leq C_\sigma^2(\|W_1\|_F^2)$$
$$\|F_k\|^2 \leq \sum_{i=1}^{p_k} C_\sigma^2(\|g_k^i\|^2) \leq C_\sigma^2(\|G_k\|^2), \quad \text{for } k = 2, \dots, n.$$

Finally, note that

$$\|g_\sigma(u)\|^2 \leq \left(\sum_{j=1}^{p_n} |w_{n+1}^j(u)| \|f_n^j\| \right)^2 \leq \|w_{n+1}(u)\|^2 \|F_n\|^2,$$

by using Cauchy-Schwarz, so that $\|g_\sigma\|^2 \leq \|w_{n+1}\|^2 \|F_n\|^2$. Thus, combining the previous relations yields

$$\|f_\sigma\|^2 \leq \|g_\sigma\|^2 \leq \|w_{n+1}\|^2 \, C_\sigma^2(\|W_n\|_2^2 \, C_\sigma^2(\|W_{n-1}\|_2^2 \dots C_\sigma^2(\|W_1\|_F^2) \dots)),$$

which is the desired result. □

References

[1] F. Anselmi, L. Rosasco, and T. Poggio. On invariance and selectivity in representation learning. *Information and Inference*, 5(2):134–158, 2016.

[2] F. Anselmi, L. Rosasco, C. Tan, and T. Poggio. Deep convolutional networks are hierarchical kernel machines. *preprint arXiv:1508.01084*, 2015.

[3] F. Bach. On the equivalence between kernel quadrature rules and random feature expansions. *Journal of Machine Learning Research (JMLR)*, 18:1–38, 2017.

[4] P. Bartlett, D. J. Foster, and M. Telgarsky. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[5] A. Bietti and J. Mairal. Invariance and stability of deep convolutional representations. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[6] L. Bo, K. Lai, X. Ren, and D. Fox. Object recognition with hierarchical kernel descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[7] S. Boucheron, O. Bousquet, and G. Lugosi. Theory of classification: A survey of some recent advances. *ESAIM: probability and statistics*, 9:323–375, 2005.

[8] J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Transactions on pattern analysis and machine intelligence (PAMI)*, 35(8):1872–1886, 2013.

[9] J. Bruna, A. Szlam, and Y. LeCun. Learning stable group invariant representations with convolutional networks. *preprint arXiv:1301.3537*, 2013.

[10] Y. Cho and L. K. Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.

[11] M. Cisse, P. Bojanowski, E. Grave, Y. Dauphin, and N. Usunier. Parseval networks: Improving robustness to adversarial examples. In *International Conference on Machine Learning (ICML)*, 2017.

[12] T. Cohen and M. Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning (ICML)*, 2016.

[13] A. Daniely, R. Frostig, V. Gupta, and Y. Singer. Random features for compositional kernels. *preprint arXiv:1703.07872*, 2017.

[14] A. Daniely, R. Frostig, and Y. Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[15] J. Diestel and J. J. Uhl. *Vector Measures*. American Mathematical Society, 1977.

[16] S. Fine and K. Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research (JMLR)*, 2:243–264, 2001.

[17] B. Haasdonk and H. Burkhardt. Invariant kernel functions for pattern analysis and machine learning. *Machine learning*, 68(1):35–61, 2007.

[18] Q. Le, T. Sarlós, and A. Smola. Fastfood–approximating kernel expansions in loglinear time. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2013.

[19] 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.

[20] T. Liang, T. Poggio, A. Rakhlin, and J. Stokes. Fisher-rao metric, geometry, and complexity of neural networks. *preprint arXiv:1711.01530*, 2017.

[21] J. Mairal. End-to-End Kernel Learning with Supervised Convolutional Kernel Networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[22] J. Mairal, P. Koniusz, Z. Harchaoui, and C. Schmid. Convolutional kernel networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[23] S. Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012.

[24] G. Montavon, M. L. Braun, and K.-R. Müller. Kernel analysis of deep networks. *Journal of Machine Learning Research (JMLR)*, 12:2563–2581, 2011.

[25] Y. Mroueh, S. Voinea, and T. A. Poggio. Learning with group invariant features: A kernel perspective. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[26] K. Muandet, K. Fukumizu, B. Sriperumbudur, B. Schölkopf, et al. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2):1–141, 2017.

[27] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[28] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *preprint arXiv:1707.09564*, 2017.

[29] B. Neyshabur, R. Tomioka, and N. Srebro. Norm-based capacity control in neural networks. In *Conference on Learning Theory*, 2015.

[30] E. Oyallon and S. Mallat. Deep roto-translation scattering for object classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[31] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems (NIPS)*, 2007.

[32] A. Raj, A. Kumar, Y. Mroueh, T. Fletcher, and B. Schoelkopf. Local group invariant representations via orbit embeddings. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

[33] S. Saitoh. *Integral transforms, reproducing kernels and their applications*, volume 369. CRC Press, 1997.

[34] I. J. Schoenberg. Positive definite functions on spheres. *Duke Mathematical Journal*, 9(1):96–108, 1942.

[35] B. Schölkopf. *Support Vector Learning*. PhD thesis, Technischen Universität Berlin, 1997.

[36] B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

[37] B. Schölkopf and A. J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. 2001.

[38] S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[39] L. Sifre and S. Mallat. Rotation, scaling and deformation invariant scattering for texture discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2013.

[40] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2014.

[41] A. J. Smola and B. Schölkopf. Sparse greedy matrix approximation for machine learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2000.

[42] E. M. Stein. *Harmonic Analysis: Real-variable Methods, Orthogonality, and Oscillatory Integrals*. Princeton University Press, 1993.

[43] I. Steinwart, P. Thomann, and N. Schmid. Learning with hierarchical gaussian kernels. *preprint arXiv:1612.00824*, 2016.

[44] A. Torralba and A. Oliva. Statistics of natural image categories. *Network: computation in neural systems*, 14(3):391–412, 2003.

[45] C. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems (NIPS)*, 2001.

[46] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017.

[47] Y. Zhang, J. D. Lee, and M. I. Jordan. ℓ_1-regularized neural networks are improperly learnable in polynomial time. In *International Conference on Machine Learning (ICML)*, 2016.

[48] Y. Zhang, P. Liang, and M. J. Wainwright. Convexified convolutional neural networks. In *International Conference on Machine Learning (ICML)*, 2017.