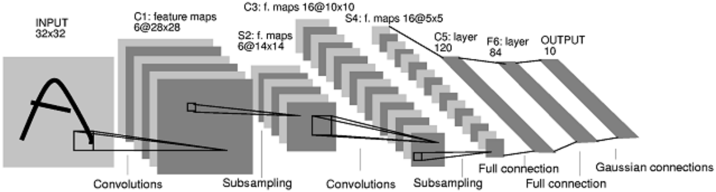
Group Invariance, Stability to Deformations, and Complexity of Deep Convolutional Representations

Alberto Bietti   Julien Mairal

Inria, Grenoble

Journées SMAI-MODE 2018, Autrans

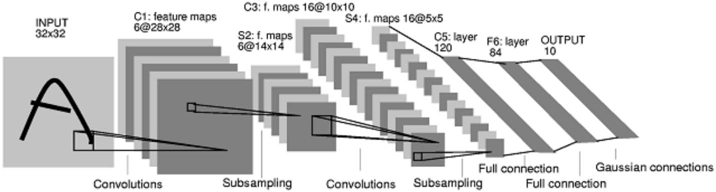
Success of deep convolutional networks



$$\underbrace{\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^{N} \ell(y_i, f(x_i))}_{\text{empirical risk}} + \underbrace{\lambda \Omega(f)}_{\text{regularization}}$$

- Neural networks: $f(x) = W_{n+1}\sigma(W_n\sigma(W_{n-1}...\sigma(W_2\sigma(W_1 x))...))$
- **Convolutional** structure in the linear operations W_k

Success of deep convolutional networks



Convolutional Neural Networks (CNNs):

- Capture **multi-scale** and **compositional** structure in natural signals
- Provide some **invariance**
- Model **local stationarity**
- **State-of-the-art** in many applications

Understanding deep convolutional representations

- Are they **stable to deformations**?
- How can we achieve **invariance to transformation groups**?
- Do they **preserve signal information**?
- How can we measure **model complexity**?

A kernel perspective

Kernels?

- Map data x to high-dimensional space, $\Phi(x) \in \mathcal{H}$ ($\mathcal{H}$: "RKHS")
- Non-linear function $f \in \mathcal{H}$ becomes linear: $f(x) = \langle f, \Phi(x) \rangle$
- Learning with a positive definite kernel $K(x, x') = \langle \Phi(x), \Phi(x') \rangle$

A kernel perspective

Kernels?

- Map data x to high-dimensional space, $\Phi(x) \in \mathcal{H}$ ($\mathcal{H}$: "RKHS")
- Non-linear function $f \in \mathcal{H}$ becomes linear: $f(x) = \langle f, \Phi(x) \rangle$
- Learning with a positive definite kernel $K(x, x') = \langle \Phi(x), \Phi(x') \rangle$
- CKN kernels (Mairal, 2016) satisfy:

$$f(x) = W_{n+1}\sigma(W_n\sigma(W_{n-1}...\sigma(W_2\sigma(W_1 x))...)) = \langle f, \Phi(x) \rangle$$

A kernel perspective

Why? Separate learning from representation: $f(x) = \langle f, \Phi(x) \rangle$

- $\Phi(x)$: CNN **architecture** (stability, invariance, signal preservation)
- f: CNN **model**, learning, generalization through RKHS norm $\|f\|$

$$|f(x) - f(x')| \leq \|f\| \cdot \|\Phi(x) - \Phi(x')\|$$

- $\|f\|$ **controls both stability and generalization**!
  $\rightarrow$ discriminating small deformations requires large $\|f\|$
  $\rightarrow$ learning stable functions is "easier"

Outline

1. Construction of the Convolutional Representation
2. Invariance and Stability
3. Model Complexity and Generalization

A generic deep convolutional representation

- $x_0 : \Omega \rightarrow \mathcal{H}_0$: initial (**continuous**) signal
  - $u \in \Omega = \mathbb{R}^d$: location ($d = 2$ for images)
  - $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)

A generic deep convolutional representation

- $x_0 : \Omega \to \mathcal{H}_0$: initial (**continuous**) signal
  - $u \in \Omega = \mathbb{R}^d$: location ($d = 2$ for images)
  - $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)
- $x_k : \Omega \to \mathcal{H}_k$: *feature map* at layer k

$$P_k x_{k-1}$$

  - P_k: **patch extraction** operator, extract small patch of feature map x_{k-1} around each point u

A generic deep convolutional representation

- $x_0 : \Omega \to \mathcal{H}_0$: initial (**continuous**) signal
  - $u \in \Omega = \mathbb{R}^d$: location ($d = 2$ for images)
  - $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)
- $x_k : \Omega \to \mathcal{H}_k$: *feature map* at layer k

$$M_k P_k x_{k-1}$$

  - P_k: **patch extraction** operator, extract small patch of feature map x_{k-1} around each point u
  - M_k: **non-linear mapping** operator, maps each patch to a new point with a **pointwise** non-linear function $\varphi_k(\cdot)$

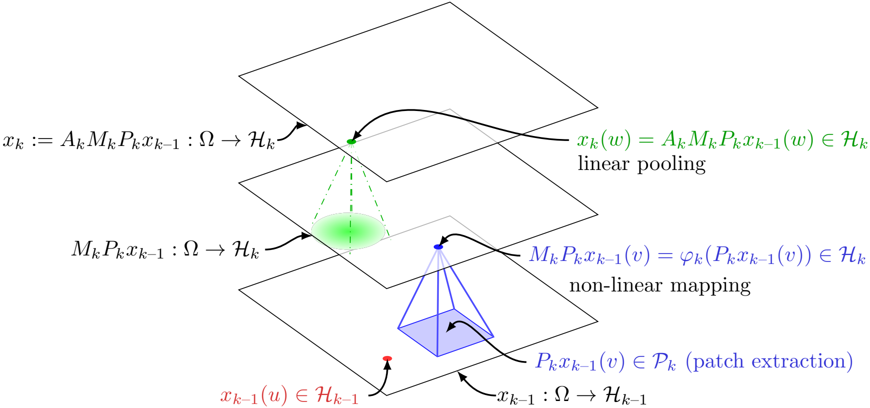
A generic deep convolutional representation

- $x_0 : \Omega \to \mathcal{H}_0$: initial (**continuous**) signal
  - $u \in \Omega = \mathbb{R}^d$: location ($d = 2$ for images)
  - $x_0(u) \in \mathcal{H}_0$: value ($\mathcal{H}_0 = \mathbb{R}^3$ for RGB images)
- $x_k : \Omega \to \mathcal{H}_k$: *feature map* at layer k

$$x_k = A_k M_k P_k x_{k-1}$$

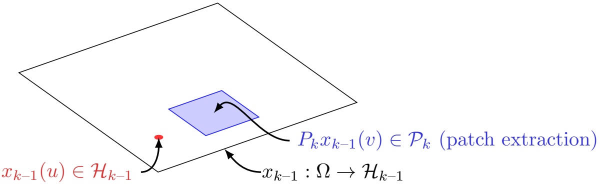
  - P_k: **patch extraction** operator, extract small patch of feature map x_{k-1} around each point u
  - M_k: **non-linear mapping** operator, maps each patch to a new point with a **pointwise** non-linear function $\varphi_k(\cdot)$
  - A_k: (linear, Gaussian) **pooling** operator at scale σ_k

A generic deep convolutional representation



Patch extraction operator P_k

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



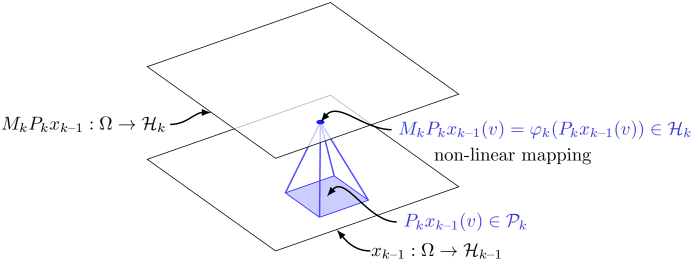
Patch extraction operator P_k

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

- S_k: patch shape, e.g. box
- P_k is **linear**, and **preserves the norm**: $\|P_k x_{k-1}\| = \|x_{k-1}\|$

Non-linear mapping operator M_k

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



Non-linear mapping operator M_k

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

- $\varphi_k : \mathcal{P}_k \to \mathcal{H}_k$ pointwise non-linearity on patches (kernel map)
- We assume **non-expansivity**: for $z, z' \in \mathcal{P}_k$

$$\|\varphi_k(z)\| \leq \|z\| \quad \text{and} \quad \|\varphi_k(z) - \varphi_k(z')\| \leq \|z - z'\|$$

- M_k then satisfies, for $x, x' \in L^2(\Omega, \mathcal{P}_k)$

$$\|M_k x\| \leq \|x\| \quad \text{and} \quad \|M_k x - M_k x'\| \leq \|x - x'\|$$

Non-linear mapping operator M_k

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

- $\varphi_k : \mathcal{P}_k \to \mathcal{H}_k$ pointwise non-linearity on patches
- We assume: for $z, z' \in \mathcal{P}_k$

$$\|\varphi_k(z)\| \leq \rho_k \|z\| \quad \text{and} \quad \|\varphi_k(z) - \varphi_k(z')\| \leq \rho_k \|z - z'\|$$

- M_k then satisfies, for $x, x' \in L^2(\Omega, \mathcal{P}_k)$

$$\|M_k x\| \leq \rho_k \|x\| \quad \text{and} \quad \|M_k x - M_k x'\| \leq \rho_k \|x - x'\|$$

- (can think instead: $\varphi_k(z) = \mathrm{ReLU}(W_k z)$, ρ_k-**Lipschitz** with $\rho_k = \|W_k\|$)

φ_k from kernels

- Kernel mapping of **homogeneous dot-product kernels**:

$$K_k(z, z') = \|z\|\|z'\|\kappa_k\left(\frac{\langle z, z'\rangle}{\|z\|\|z'\|}\right) = \langle \varphi_k(z), \varphi_k(z')\rangle.$$

- $\kappa_k(u) = \sum_{j=0}^{\infty} b_j u^j$ with $b_j \geq 0$, $\kappa_k(1) = 1$
- Commonly used for hierarchical kernels
- $\|\varphi_k(z)\| = K_k(z, z)^{1/2} = \|z\|$
- $\|\varphi_k(z) - \varphi_k(z')\| \leq \|z - z'\|$ if $\kappa_k'(1) \leq 1$
- $\implies$ **non-expansive**

φ_k from kernels

- Kernel mapping of **homogeneous dot-product kernels**:

$$K_k(z, z') = \|z\| \|z'\| \kappa_k\left(\frac{\langle z, z' \rangle}{\|z\| \|z'\|}\right) = \langle \varphi_k(z), \varphi_k(z') \rangle.$$

- $\kappa_k(u) = \sum_{j=0}^{\infty} b_j u^j$ with $b_j \geq 0$, $\kappa_k(1) = 1$
- Commonly used for hierarchical kernels
- $\|\varphi_k(z)\| = K_k(z, z)^{1/2} = \|z\|$
- $\|\varphi_k(z) - \varphi_k(z')\| \leq \|z - z'\|$ if $\kappa_k'(1) \leq 1$
- $\implies$ **non-expansive**
- Examples:
  - $\kappa_{\text{exp}}(\langle z, z' \rangle) = e^{\langle z, z' \rangle - 1}$ (Gaussian kernel on the sphere)
  - $\kappa_{\text{inv-poly}}(\langle z, z' \rangle) = \frac{1}{2 - \langle z, z' \rangle}$

φ_k from kernels: CKNs approximation

Convolutional Kernel Networks approximation (Mairal, 2016):

φ_k from kernels: CKNs approximation

Convolutional Kernel Networks approximation (Mairal, 2016):

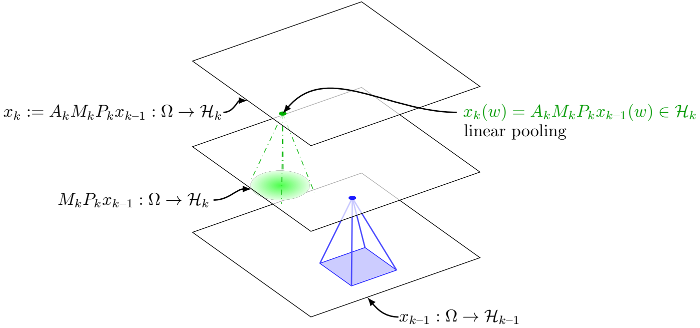
- Approximate $\varphi_k(z)$ by **projection** on $span(\varphi_k(z_1), \dots, \varphi_k(z_p))$ (Nystrom)
- Leads to **tractable**, p-dimensional representation $\psi_k(z)$
- Norm is preserved, and projection is non-expansive:

$$\begin{aligned} \|\psi_k(z) - \psi_k(z')\| &= \|\Pi_k \varphi_k(z) - \Pi_k \varphi_k(z')\| \\ &\leq \|\varphi_k(z) - \varphi_k(z')\| \leq \|z - z'\| \end{aligned}$$

- Anchor points $z_1, \dots, z_p$ ($\approx$ filters) can be **learned from data** (K-means or backprop)

Pooling operator A_k

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



Pooling operator A_k

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

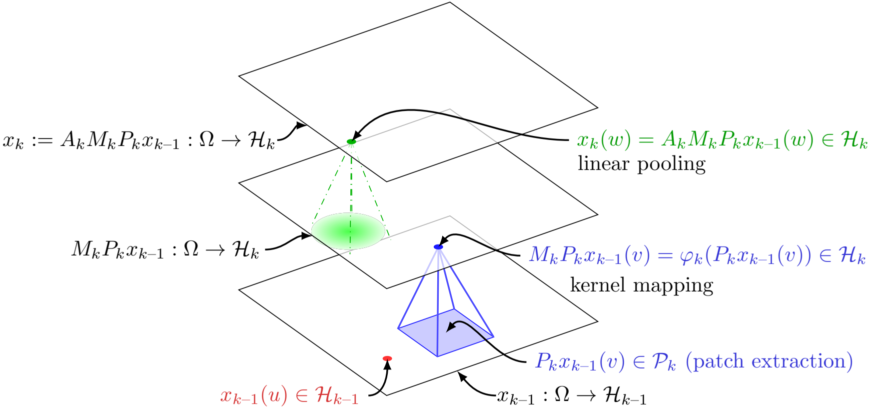
- h_{σ_k}: pooling filter at scale σ_k
- $h_{\sigma_k}(u) := \sigma_k^{-d} h(u/\sigma_k)$ with $h(u)$ **Gaussian**
- **linear, non-expansive operator**: $\|A_k\| \leq 1$

Pooling operator A_k

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

- h_{σ_k}: pooling filter at scale σ_k
- $h_{\sigma_k}(u) := \sigma_k^{-d} h(u/\sigma_k)$ with $h(u)$ **Gaussian**
- **linear, non-expansive operator**: $\|A_k\| \leq 1$
- In practice: **discretization**, sampling at resolution σ_k after pooling
- **Recovery** with kernels possible when **subsampling** $\leq$ **patch size**

Recap: P_k, M_k, A_k



Multilayer construction

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

- S_k, σ_k grow exponentially in practice (i.e. fixed with subsampling)

Multilayer construction

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

- S_k, σ_k grow exponentially in practice (i.e. fixed with subsampling)
- x_0 is typically a **discrete** signal aquired with physical device
  - Natural assumption: $x_0 = A_0 x$, with x the original continuous signal, A_0 local integrator (**anti-aliasing**)

Multilayer construction

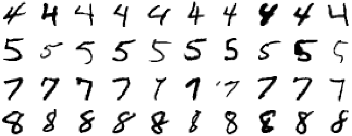
$$x_n := A_n M_n P_n A_{n-1} M_{n-1} P_{n-1} \cdots A_1 M_1 P_1 x_0 \ \in \ L^2(\Omega, \mathcal{H}_n)$$

- S_k, σ_k grow exponentially in practice (i.e. fixed with subsampling)
- x_0 is typically a **discrete** signal aquired with physical device
  - Natural assumption: $x_0 = A_0 x$, with x the original continuous signal, A_0 local integrator (**anti-aliasing**)
- **Prediction layer**: e.g. linear
  - $f(x_0) = \langle w, x_n \rangle$
  - "linear kernel" $\mathcal{K}(x_0, x_0') = \langle x_n, x_n' \rangle = \int_\Omega \langle x_n(u), x_n'(u) \rangle du$

Outline

Stability to deformations: definitions

- $\tau : \Omega \to \Omega$: C^1-diffeomorphism
- $L_\tau x(u) = x(u - \tau(u))$: action operator
- Much richer group of transformations than translations



Stability to deformations: definitions

- Representation $\Phi(\cdot)$ is **stable** (Mallat, 2012) if:

$$\|\Phi(L_\tau x) - \Phi(x)\| \leq (C_1 \|\nabla \tau\|_\infty + C_2 \|\tau\|_\infty) \|x\|$$

- $\|\nabla \tau\|_\infty = \sup_u \|\nabla \tau(u)\|$ controls deformation
- $\|\tau\|_\infty = \sup_u |\tau(u)|$ controls translation
- $C_2 \to 0$: translation invariance

Warmup: translation invariance

- Representation:

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

- Translation: $L_c x(u) = x(u - c)$

Warmup: translation invariance

- Representation:

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

- Translation: $L_c x(u) = x(u - c)$
- *Equivariance* - all operators commute with L_c: $\square L_c = L_c \square$

$$\begin{aligned} \|\Phi(L_c x) - \Phi(x)\| &= \|L_c \Phi(x) - \Phi(x)\| \\ &\leq \|L_c A_n - A_n\| \cdot \|x\| \end{aligned}$$

Warmup: translation invariance

- Representation:

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

- Translation: $L_c x(u) = x(u - c)$
- *Equivariance* - all operators commute with L_c: $\square L_c = L_c \square$

$$\begin{aligned} \|\Phi(L_c x) - \Phi(x)\| &= \|L_c \Phi(x) - \Phi(x)\| \\ &\leq \|L_c A_n - A_n\| \cdot \|x\| \end{aligned}$$

- Mallat (2012): $\|L_\tau A_n - A_n\| \leq \frac{C_2}{\sigma_n} \|\tau\|_\infty$

Warmup: translation invariance

- Representation:

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

- Translation: $L_c x(u) = x(u - c)$
- *Equivariance* - all operators commute with L_c: $\square L_c = L_c \square$

$$\begin{aligned} \|\Phi(L_c x) - \Phi(x)\| &= \|L_c \Phi(x) - \Phi(x)\| \\ &\leq \|L_c A_n - A_n\| \cdot \|x\| \end{aligned}$$

- Mallat (2012): $\|L_c A_n - A_n\| \leq \frac{C_2}{\sigma_n} c$

Warmup: translation invariance

- Representation:

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

- Translation: $L_c x(u) = x(u - c)$
- *Equivariance* - all operators commute with L_c: $\square L_c = L_c \square$

$$\begin{aligned} \|\Phi(L_c x) - \Phi(x)\| &= \|L_c \Phi(x) - \Phi(x)\| \\ &\leq \|L_c A_n - A_n\| \cdot \|x\| \end{aligned}$$

- Mallat (2012): $\|L_c A_n - A_n\| \leq \frac{C_2}{\sigma_n} c$
- **Group invariance** (*e.g.*, rotations) with different construction
  - Group equivariance in P_k, A_k
  - Global pooling on the group at last layer

Stability to deformations

- Representation:

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

- Patch extraction P_k and pooling A_k **do not commute** with L_τ!

Stability to deformations

- Representation:

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

- Patch extraction P_k and pooling A_k **do not commute** with L_τ!
- $\|A_k L_\tau - L_\tau A_k\| \leq C_1 \|\nabla \tau\|_\infty$ (from Mallat, 2012)

Stability to deformations

- Representation:

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

- Patch extraction P_k and pooling A_k **do not commute** with L_τ!
- $\|[A_k, L_\tau]\| \leq C_1 \|\nabla \tau\|_\infty$ (from Mallat, 2012)

Stability to deformations

- Representation:

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

- Patch extraction P_k and pooling A_k **do not commute** with L_τ!
- $\|[A_k, L_\tau]\| \leq C_1 \|\nabla \tau\|_\infty$ (from Mallat, 2012)
- But: $[P_k, L_\tau]$ is **unstable** at high frequencies!

Stability to deformations

- Representation:

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

- Patch extraction P_k and pooling A_k **do not commute** with L_τ!
- $\|[A_k, L_\tau]\| \leq C_1 \|\nabla \tau\|_\infty$ (from Mallat, 2012)
- But: $[P_k, L_\tau]$ is **unstable** at high frequencies!
- Adapt to **current layer resolution**, patch size controlled by σ_{k-1}:

$$\|[P_k A_{k-1}, L_\tau]\| \leq C_{1,\kappa} \|\nabla \tau\|_\infty \qquad \sup_{u \in S_k} |u| \leq \kappa \sigma_{k-1}$$

Stability to deformations

- Representation:

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

- Patch extraction P_k and pooling A_k **do not commute** with L_τ!
- $\|[A_k, L_\tau]\| \leq C_1 \|\nabla \tau\|_\infty$ (from Mallat, 2012)
- But: $[P_k, L_\tau]$ is **unstable** at high frequencies!
- Adapt to **current layer resolution**, patch size controlled by σ_{k-1}:

$$\|[P_k A_{k-1}, L_\tau]\| \leq C_{1,\kappa} \|\nabla \tau\|_\infty \qquad \sup_{u \in S_k} |u| \leq \kappa \sigma_{k-1}$$

- $C_{1,\kappa}$ grows as $\kappa^{d+1} \implies$ more stable with **small patches** (e.g., 3x3, VGG et al.)

Stability to deformations: final result

Theorem

If $\|\nabla\tau\|_\infty \leq 1/2$,

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

- Suggests several layers with small patches and subsampling for stability + signal preservation

Stability to deformations: final result

Theorem

If $\|\nabla\tau\|_\infty \leq 1/2$,

$$\|\Phi_n(L_\tau x) - \Phi_n(x)\| \leq \prod_k \rho_k \left(C_{1,\kappa} \, (n+1) \, \|\nabla\tau\|_\infty + \frac{C_2}{\sigma_n} \|\tau\|_\infty \right) \|x\|$$

- Suggests several layers with small patches and subsampling for stability + signal preservation
- (for generic CNNs, multiply by $\prod_k \rho_k = \prod_k \|W_k\|$)

Outline

RKHS of patch kernels K_k

$$K_k(z, z') = \|z\|\|z'\|\kappa\left(\frac{\langle z, z'\rangle}{\|z\|\|z'\|}\right), \qquad \kappa(u) = \sum_{j=0}^{\infty} b_j u^j$$

- RKHS contains **homogeneous functions**:

$$f : z \mapsto \|z\|\sigma(\langle g, z\rangle / \|z\|)$$

Homogeneous version of (Zhang et al., 2016, 2017)

RKHS of patch kernels K_k

$$K_k(z, z') = \|z\|\|z'\|\kappa\left(\frac{\langle z, z'\rangle}{\|z\|\|z'\|}\right), \qquad \kappa(u) = \sum_{j=0}^{\infty} b_j u^j$$

- RKHS contains **homogeneous functions**:

$$f : z \mapsto \|z\|\sigma(\langle g, z\rangle/\|z\|)$$

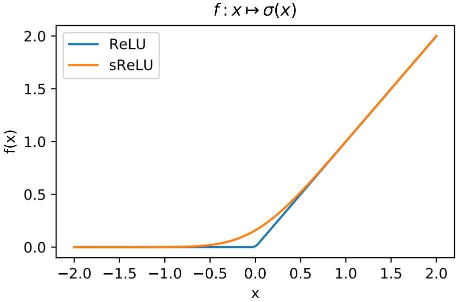
- **Smooth activations**: $\sigma(u) = \sum_{j=0}^{\infty} a_j u^j$
- Norm: $\|f\|_{\mathcal{H}_k}^2 \leq C_\sigma^2(\|g\|^2) = \sum_{j=0}^{\infty} \frac{a_j^2}{b_j}\|g\|^2 < \infty$

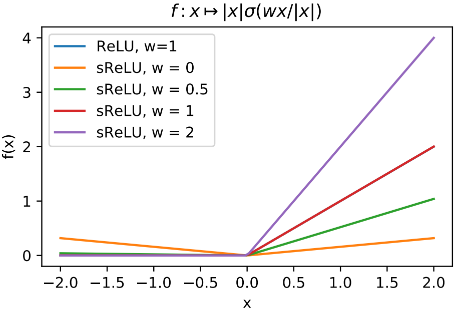
Homogeneous version of (Zhang et al., 2016, 2017)

RKHS of patch kernels K_k

Examples:

- $\sigma(u) = u$ (linear): $C_\sigma^2(\lambda^2) = O(\lambda^2)$
- $\sigma(u) = u^p$ (polynomial): $C_\sigma^2(\lambda^2) = O(\lambda^{2p})$
- $\sigma \approx$ sin, sigmoid, smooth ReLU: $C_\sigma^2(\lambda^2) = O(e^{c\lambda^2})$





Constructing a CNN in the RKHS $\mathcal{H}_\mathcal{K}$

- Consider a CNN with filters $W_k^{ij}(u), u \in S_k$
- "Homogeneous" activations σ
- The CNN can be constructed hierarchically in $\mathcal{H}_\mathcal{K}$
- Norm:

$$\|f_\sigma\|^2 \leq \|W_{n+1}\|_2^2 \; C_\sigma^2(\|W_n\|_2^2 \; C_\sigma^2(\|W_{n-1}\|_2^2 \; C_\sigma^2(\ldots)))$$

Constructing a CNN in the RKHS $\mathcal{H}_{\mathcal{K}}$

- Consider a CNN with filters $W_k^{ij}(u), u \in S_k$
- "Homogeneous" activations σ
- The CNN can be constructed hierarchically in $\mathcal{H}_{\mathcal{K}}$
- Norm (linear layers):

$$\|f_\sigma\|^2 \leq \|W_{n+1}\|_2^2 \cdot \|W_n\|_2^2 \cdot \|W_{n-1}\|_2^2 \dots \|W_1\|_2^2$$

- Linear layers: product of spectral norms

Link with generalization

- Simple bound on Rademacher complexity for linear/kernel methods:

$$\mathcal{F}_B = \{f \in \mathcal{H}_{\mathcal{K}}, \|f\| \leq B\} \implies \mathrm{Rad}_n(\mathcal{F}_B) \leq O\left(\frac{BR}{\sqrt{N}}\right)$$

Link with generalization

- Simple bound on Rademacher complexity for linear/kernel methods:

$$\mathcal{F}_B = \{f \in \mathcal{H}_\mathcal{K}, \|f\| \leq B\} \implies \mathrm{Rad}_n(\mathcal{F}_B) \leq O\left(\frac{BR}{\sqrt{N}}\right)$$

- Leads to margin bound $O(\|\hat{f}_n\| R / \gamma \sqrt{N})$ for a learned CNN $\hat{f}_N$ with margin (confidence) $\gamma > 0$
- Related to recent generalization bounds for neural networks based on **product of spectral norms** (*e.g.*, Bartlett et al., 2017)

Deep convolutional representations: conclusions

Study of generic properties

- Deformation stability with small patches, adapted to resolution
- Signal preservation when subsampling $\leq$ patch size
- Group invariance by changing patch extraction and pooling

Deep convolutional representations: conclusions

Study of generic properties

- Deformation stability with small patches, adapted to resolution
- Signal preservation when subsampling $\leq$ patch size
- Group invariance by changing patch extraction and pooling

Applies to learned models

- Same quantity $\|f\|$ controls stability and generalization:
  - "higher capacity" is needed to discriminate small deformations
  - Learning is "easier" with stable functions
- Questions:
  - Better regularization?
  - How does SGD control capacity in CNNs?

AI Summer School in Grenoble! July 2-6

PAISS - `https://project.inria.fr/paiss/`

References I

P. Bartlett, D. J. Foster, and M. Telgarsky. Spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1706.08498*, 2017.

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

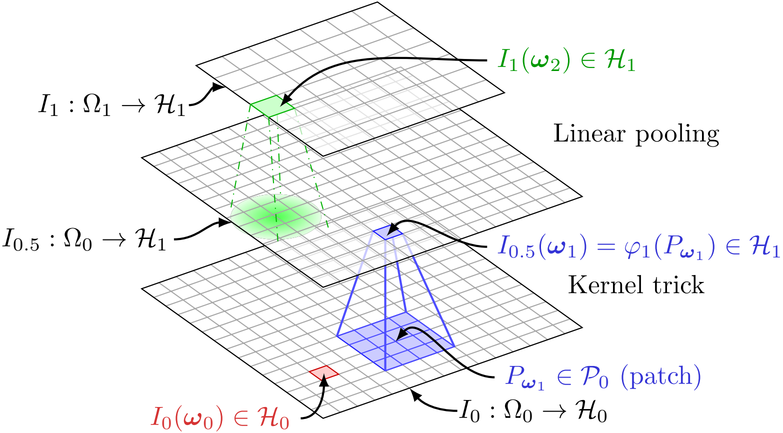
J. Mairal. End-to-End Kernel Learning with Supervised Convolutional Kernel Networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

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

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.

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

Discretization and signal preservation



Discretization and signal preservation

- $\bar{x}_k$: subsampling factor s_k after pooling with scale $\sigma_k \approx s_k$:

$$\bar{x}_k[n] = A_k M_k P_k \bar{x}_{k-1}[n s_k]$$

Discretization and signal preservation

- $\bar{x}_k$: subsampling factor s_k after pooling with scale $\sigma_k \approx s_k$:

$$\bar{x}_k[n] = A_k M_k P_k \bar{x}_{k-1}[n s_k]$$

- **Claim**: We can recover $\bar{x}_{k-1}$ from $\bar{x}_k$ if **subsampling** $s_k \leq$ **patch size**

Discretization and signal preservation

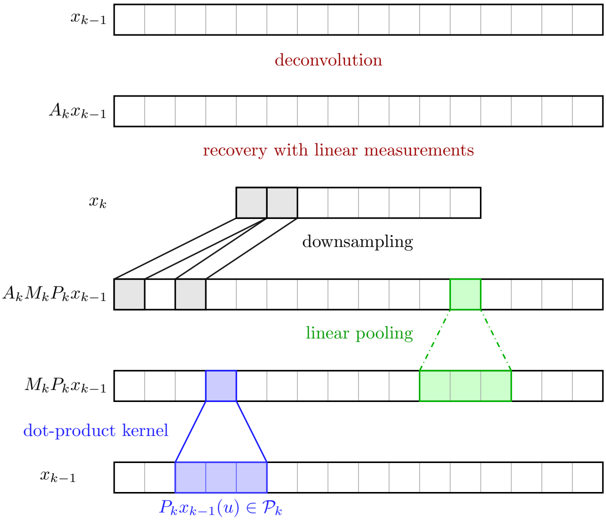
- $\bar{x}_k$: subsampling factor s_k after pooling with scale $\sigma_k \approx s_k$:

$$\bar{x}_k[n] = A_k M_k P_k \bar{x}_{k-1}[n s_k]$$

- **Claim**: We can recover $\bar{x}_{k-1}$ from $\bar{x}_k$ if **subsampling** $s_k \leq$ **patch size**
- **How**? Kernels! Recover patches with **linear functions** (contained in RKHS)

$$\langle f_w, M_k P_k x(u) \rangle = f_w(P_k x(u)) = \langle w, P_k x(u) \rangle$$

Signal recovery: example in 1D



Beyond the translation group

- Global invariance to other groups? (rotations, reflections, roto-translations, ...)
- Group action $L_g x(u) = x(g^{-1}u)$
- **Equivariance** in inner layers + **(global) pooling** in last layer
- Similar construction to (Cohen and Welling, 2016)

G-equivariant layer construction

- Feature maps $x(u)$ defined on $u \in G$ (G: locally compact group)
- **Patch extraction**:

$$Px(u) = (x(uv))_{v \in S}$$

- **Non-linear mapping**: equivariant because pointwise!
- **Pooling** (μ: left-invariant Haar measure):

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

Group invariance and stability

- Stability analysis should work on "compact Lie groups" (similar to Mallat, 2012), e.g., rotations only
- For more complex groups (e.g., roto-translations):
  - Stability only w.r.t. subgroup (translations) is enough?
  - Inner layers: only pool on translation group
  - Last layer: global pooling on rotations
  - Cohen and Welling (2016): rotation pooling in inner layers hurts performance on Rotated MNIST