Scattering Transform & Pattern Recognition

(*tutorial*)

Edouard Oyallon

edouard.oyallon@lip6.fr

CNRS, LIP6

Machine Learning & Deep Learning for Information Access

High Dimensional classification

$$(x_i, y_i) \in \mathbb{R}^{224^2} \times \{1, ..., 1000\}, i < 10^6 \longrightarrow \hat{y}(x)?$$

"Rhinos"

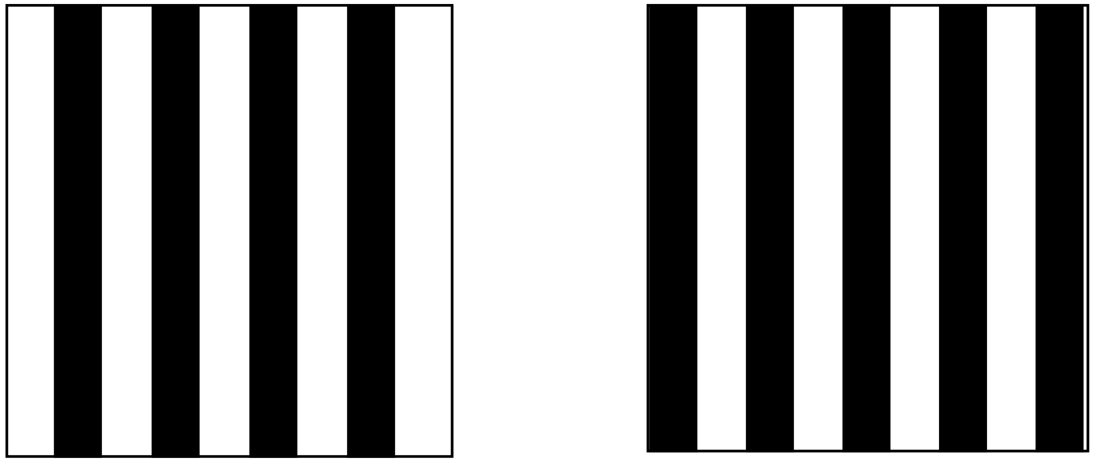
Estimation problem

Training set to predict labels

"Rhino"

Not a "rhino"

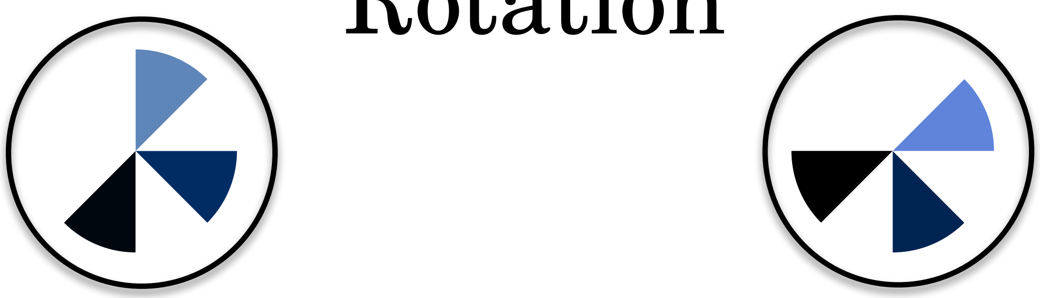
Translation



x y

$$\|x - y\|_2 = 2$$

Rotation



x y

Averaging is the key to get invariants

Averaging makes euclidean distance meaningful in high dimension

Group action

- Consider a signal $\boldsymbol{x}$ and a $\boldsymbol{g}$ from a group G. We typically consider action like:

$$\forall u \in \mathbb{R}^2, g.x(u) \triangleq x(g^{-1}u)$$

- Covariant representation:

$$\Phi(g.x) = g.\Phi(x)$$

- Invariant representation:

$$\Phi(g.x) = \Phi(x)$$

If covariance, invariance is simple to get: $\sum_{g\in G} \Phi(g.x) = \sum_{g\in G} g.\Phi(x)$

Symmetry group hypothesis

Ref.: Understanding deep convolutional networks S Mallat

- To each classification problem corresponds a canonic and unique symmetry group G:

$$\forall x, \forall g \in G, \Phi x = \Phi g.x$$

High dimensional

- We hypothesise there exists Lie groups, which could be progressively linearized:

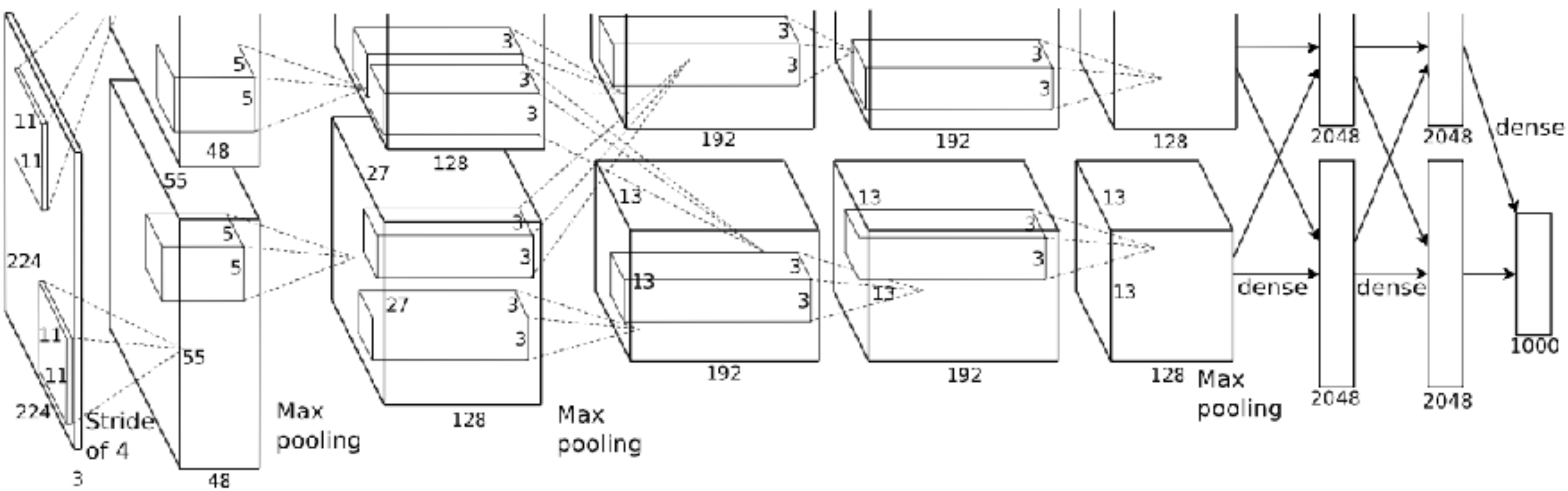
$$G_0 \subset G_1 \subset ... \subset G_J \subset G$$

- Examples are given by the euclidean group:

$$G_0 = \mathbb{R}^2, G_1 = G_0 \ltimes SL_2(\mathbb{R})$$

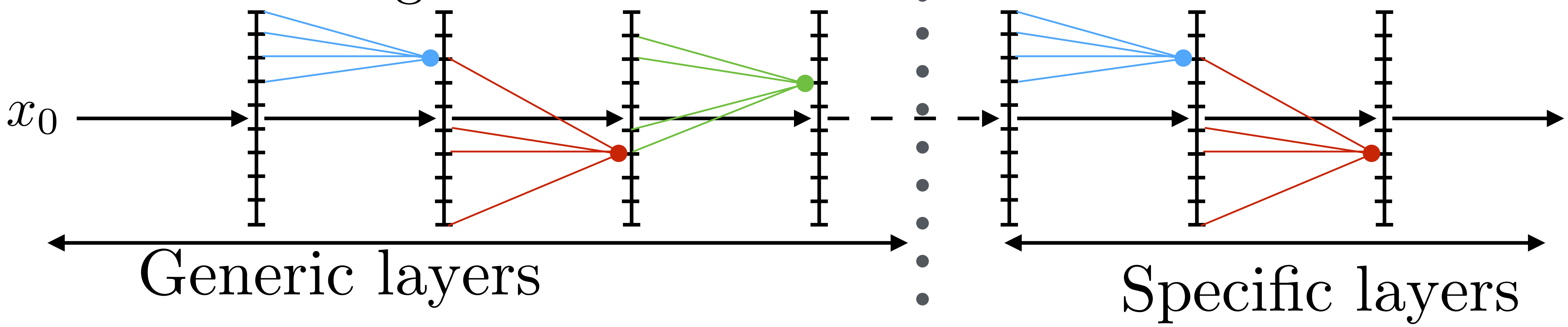
CNNs: state-of-the-art methods

Ref.: ImageNet Classification with Deep Convolutional Network, A Krizhevsky et al.

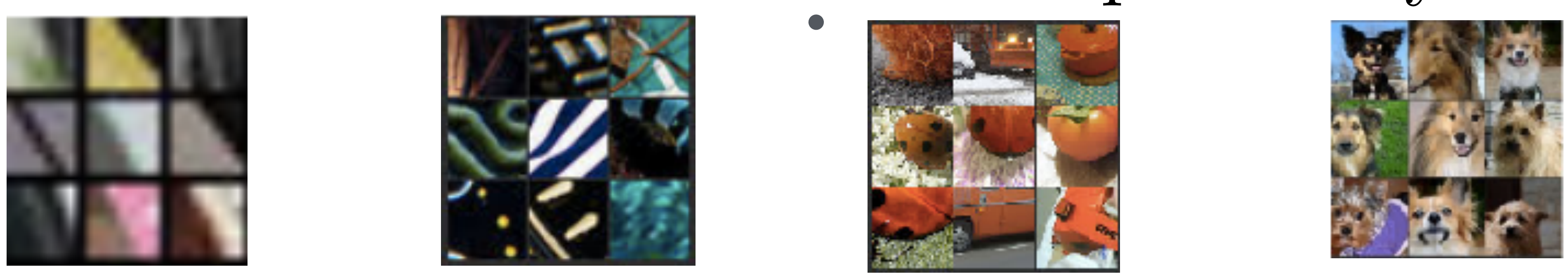


CNNs and generecity

- CNNs are a cascade of supervisedly optimized operators from raw signals:



Ref.: Visualizing and understanding Convolutional Networks Zeiler, et al.



- They necessarily learn physical law, that are generic and relative to the nature of the signals

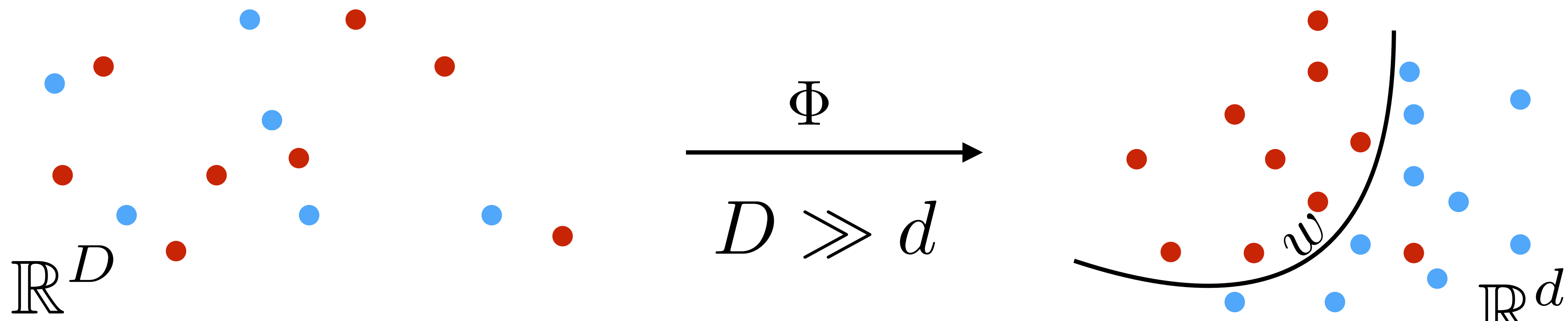
Do we need to learn those laws?

⟵ well structured filters. . .

Ref.: ImageNet Classification with Deep Convolutional Network, A Krizhevsky et al.

Fighting the curse of dimensionality

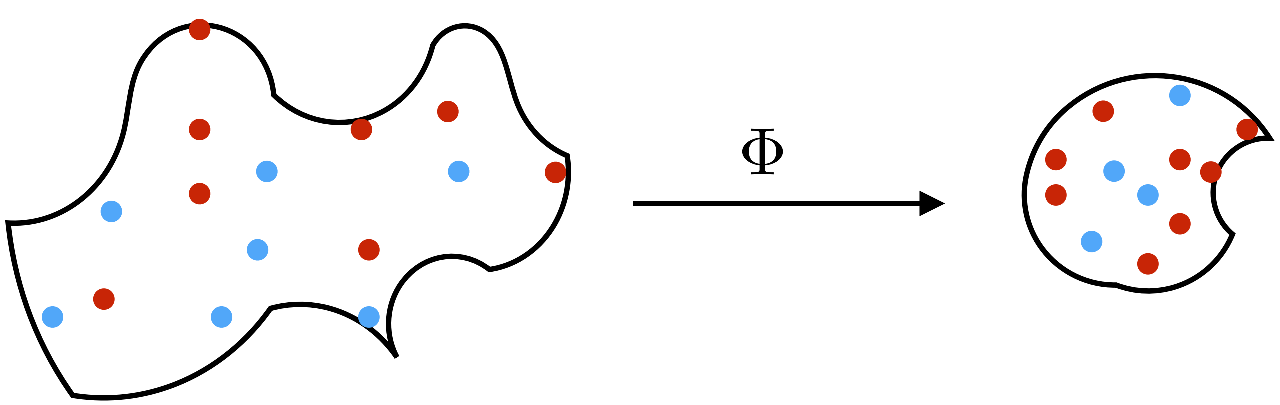
- Objective: building a representation Φx of x such that a simple (say euclidean) classifier $\hat{y}$ can estimate the label y:

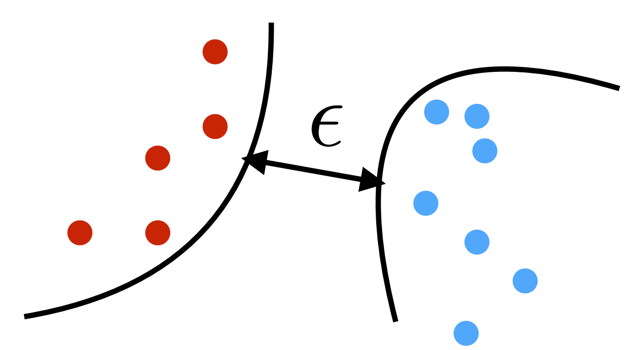


- Designing Φ: must be regular with respect to the class:

$$\|\Phi x - \Phi x'\| \lll 1 \Rightarrow \hat{y}(x) = \hat{y}(x')$$

- Necessary dimensionality reduction and separation to break the curse of dimensionality:

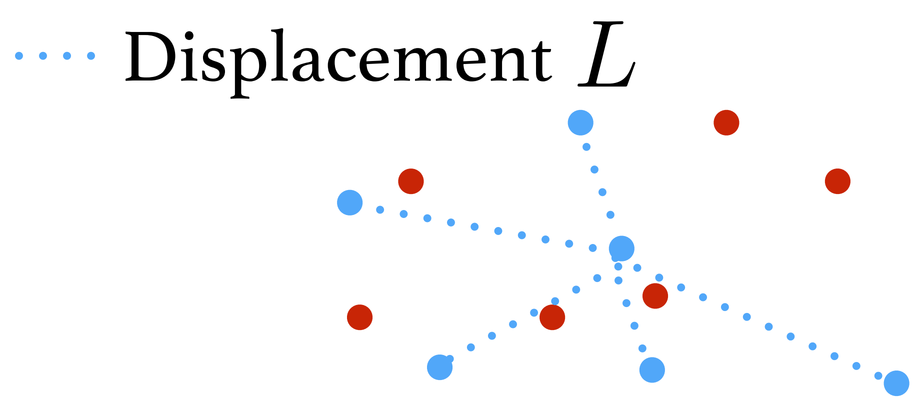




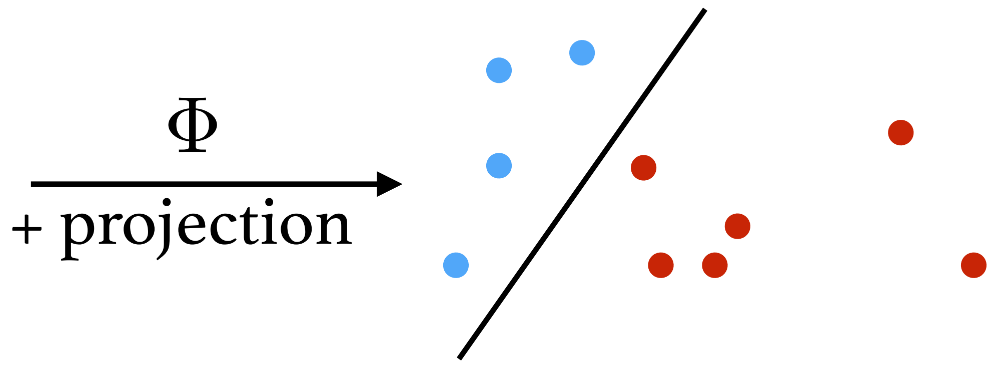
How to tackle the curse of dimensionality?

- Weak differentiability property:

$$\sup_L \frac{\|\Phi L x - \Phi x\|}{\|Lx - x\|} < \infty \Rightarrow \exists \text{ "weak" } \partial_x \Phi$$
$$\Rightarrow \Phi L x \approx \Phi x + \underbrace{\partial_x \Phi L}_{\text{A linear operator}} + o(\|L\|)$$



- A linear projection (to kill L) build an invariant

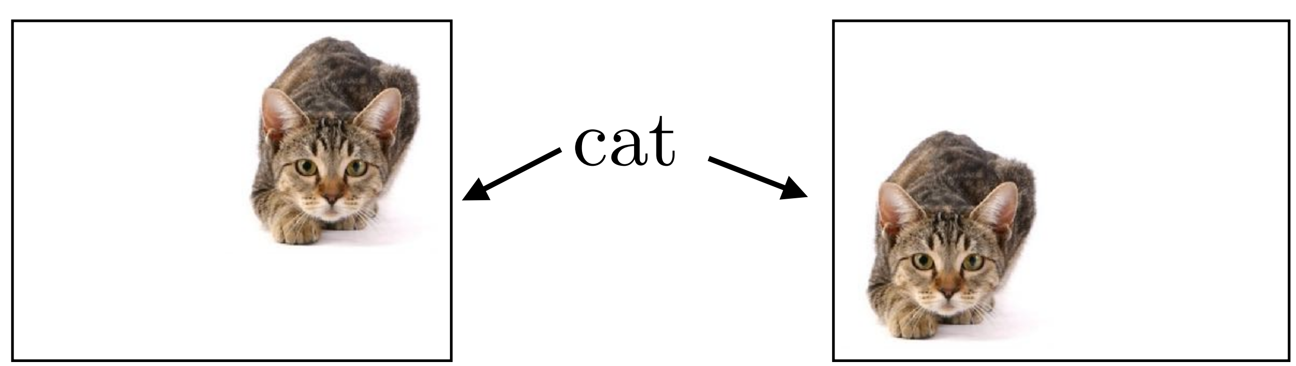


Handcrafted features in classification

- Until 2012, SIFT, HoG, LBPs... combined with an (unsupervised) learning pipeline.

Ref.: Improving the fisher kernel for large-scal
image classification, F Perronnin et al.

- They incorporate invariances w.r.t. to geometric variabilities and discriminate them as well.



- Yet, CNNs removed them and obtain better numerical results.

Why would we want to use them?

Feeding CNN with prior representations: for what?

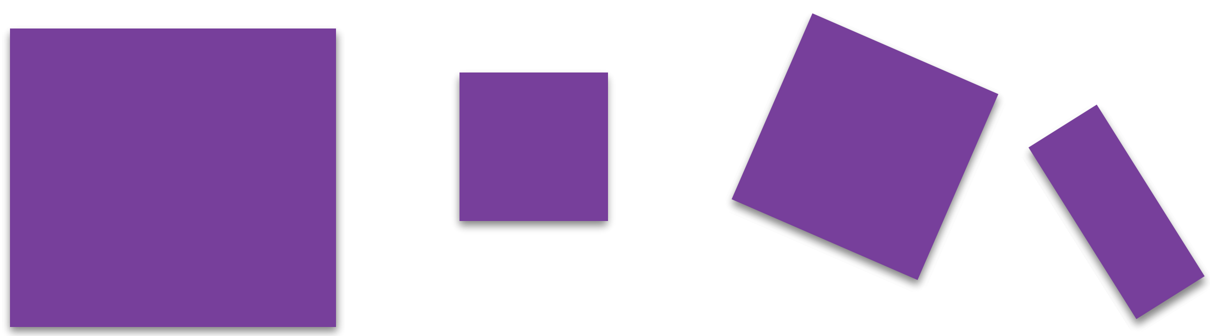
- Features that bring interpretation
- Speeding-up training time & computations
- Speeding-up inferences
- Reducing sample complexity (e.g., reducing overfitting)

Invariances in vision tasks

Before feeding a classifier, removing unnecessary variabilities is necessary:

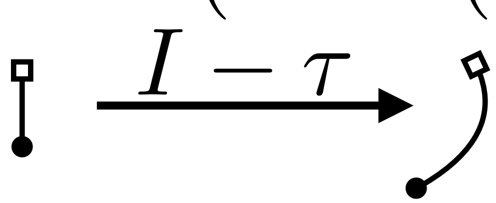
Geometric variability

Groups acting on images:
translation, rotation, scaling



Other sources : luminosity, occlusion, small deformations

$$L_\tau x(u) = x(u - \tau(u)), \tau \in \mathcal{C}^\infty$$



Class variability

Intraclass variability
<u>Not informative</u>

Extraclass variability

Discrete image to continuous image

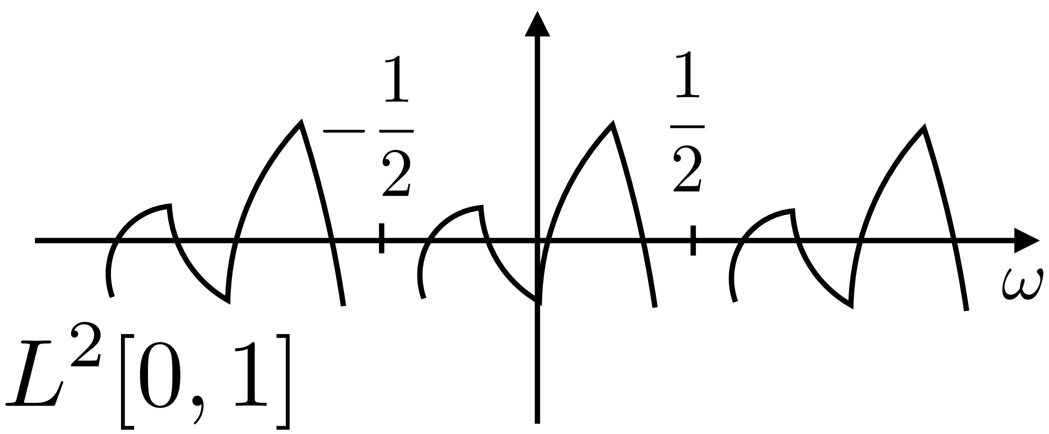
- An image x corresponds to the discretisation of a physical anagogic signal (light!)

- An array of numbers:

$$x[n_1, n_2] \in \mathbb{R}, n_1, n_2 \leq N$$

- One can set

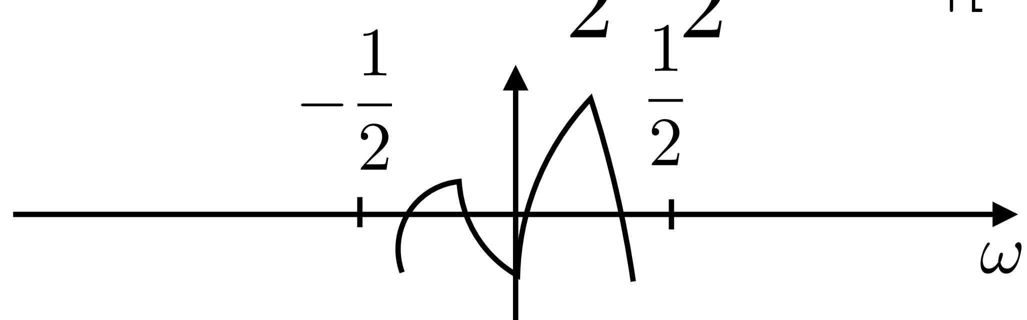
$$x(u) = \sum_{n \in \mathbb{Z}^2} x[n]\delta_n(u)$$



then,
$$\mathcal{F}x(\omega) = \sum_{n \in \mathbb{Z}^2} x[n]e^{-in\omega}, \mathcal{F}x \in L^2[0,1]$$

- Nyquiest-Shannon sampling property:

$$\exists ! \tilde{x} \in \mathbb{L}^2(\mathbb{R}), \text{support}(\mathcal{F}\tilde{x}) \subset [-\frac{1}{2}, \frac{1}{2}], \mathcal{F}\tilde{x}_{|[-\frac{1}{2},\frac{1}{2}]} = \mathcal{F}x$$



Reminder about Fourier

$$\mathcal{F} : \mathbb{L}^2(\mathbb{R}^d) \to \mathbb{L}^2(\mathbb{R}^d)$$

$$\mathcal{F}x(\omega) \triangleq \hat{x}(\omega) \triangleq \int_{\mathbb{R}^d} e^{-2i\pi\omega^T u} x(u)\, du$$

$$x \star y(u) \triangleq \int_{\mathbb{R}^d} x(u-t)y(t)\, dt$$

Isometry: $\|\mathcal{F}x\|_2 = \|x\|_2$

$$x \star y(u) \qquad \hat{x}(\omega)\hat{y}(\omega)$$

$$x(u) \xrightarrow{\mathcal{F}} \hat{x}(\omega)$$

$$\frac{d}{du}x(u) \xrightarrow{\mathcal{F}} i\omega\hat{x}(\omega)$$

$$x_a(u) \triangleq x(u-a) \xrightarrow{\mathcal{F}} e^{2i\pi\omega^T a}\hat{x}(\omega)$$

Let $L : \mathbb{L}^2 \to \mathbb{L}^2$ continuous, if $Lx_a = (Lx)_a$ for any $\boldsymbol{x}$ then

$$\exists k, Lx = k \star x$$

Easy to generalise on compact and commutative groups.

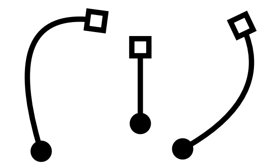
A motivating example

- Translation invariance? Why not:

$$\Phi x(\omega) = |\hat{x}(\omega)|$$

<u>Deformations</u>

$$L_\tau x(u) = x(u - \tau(u))$$



Let $x(u) = e^{i\omega_0 u - \frac{1}{2}u^2}$ and $\tau(u) = su, s > 0$

$$|\Phi x_\tau(\omega) - \Phi x(\omega)| \propto |e^{-(\omega-\omega_0)^2} - \frac{1}{(1-s)} e^{-(\frac{\omega}{(1-s)} - \omega_0)^2}|$$

then:

$$\|\Phi x_\tau - \Phi x\| \sim s\omega_0 = \|\nabla\tau\|\omega_0$$

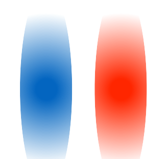
Wavelets

- ψ is a wavelet iff $\int \psi(u)du = 0$ and $\int |\psi|^2(u)du < \infty$
- Typically localised in space and frequency.
- Rotation, dilation of a wavelets:

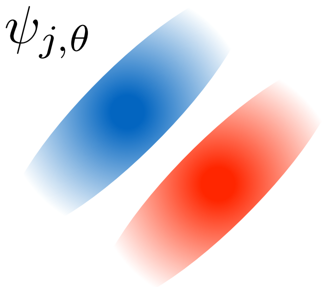
Group action!

$$\psi_{j,\theta} = \frac{1}{2^{2j}} \psi(\frac{r_\theta(u)}{2^j})$$

ψ

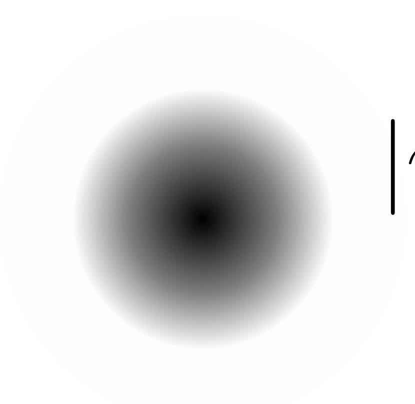


$\psi_{j,\theta}$



- Design wavelets selective to rotation variabilities.

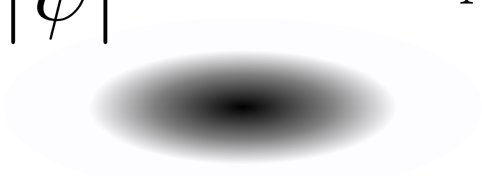
Isotropic



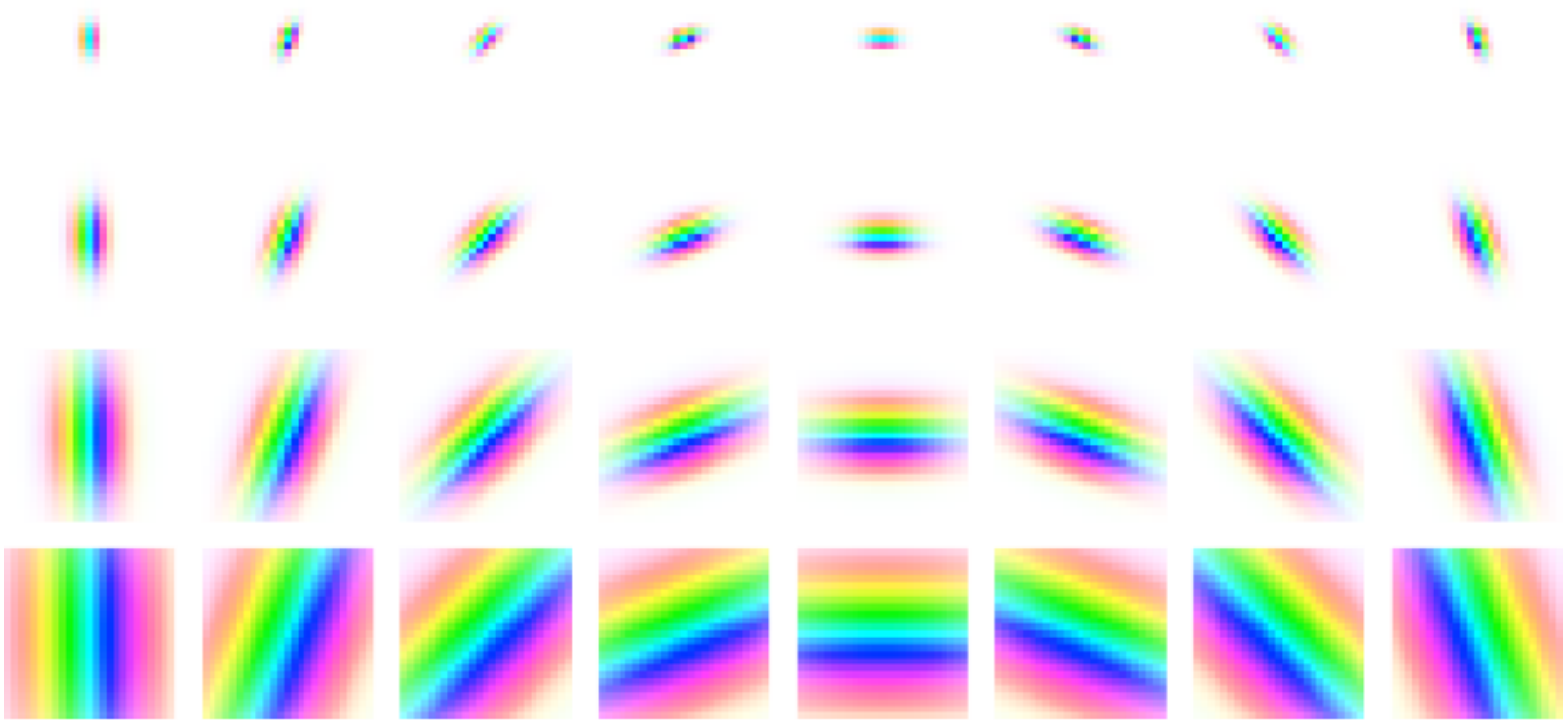
$|\hat{\psi}|$

VS

$|\hat{\psi}|$



Non-Isotropic



$$\psi(u) = \frac{1}{2\pi\sigma} e^{-\frac{\|u\|^2}{2\sigma}} \left(e^{i\xi . u} - \kappa\right)$$

$$\phi(u) = \frac{1}{2\pi\sigma} e^{-\frac{\|u\|^2}{2\sigma}}$$

(for sake of simplicity, formula
are given in the isotropic case)

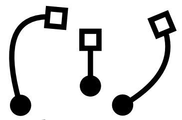
The Gabor wavelet

Invariances via wavelets

Ref.: Group Invariant Scattering, Mallat S

Deformations

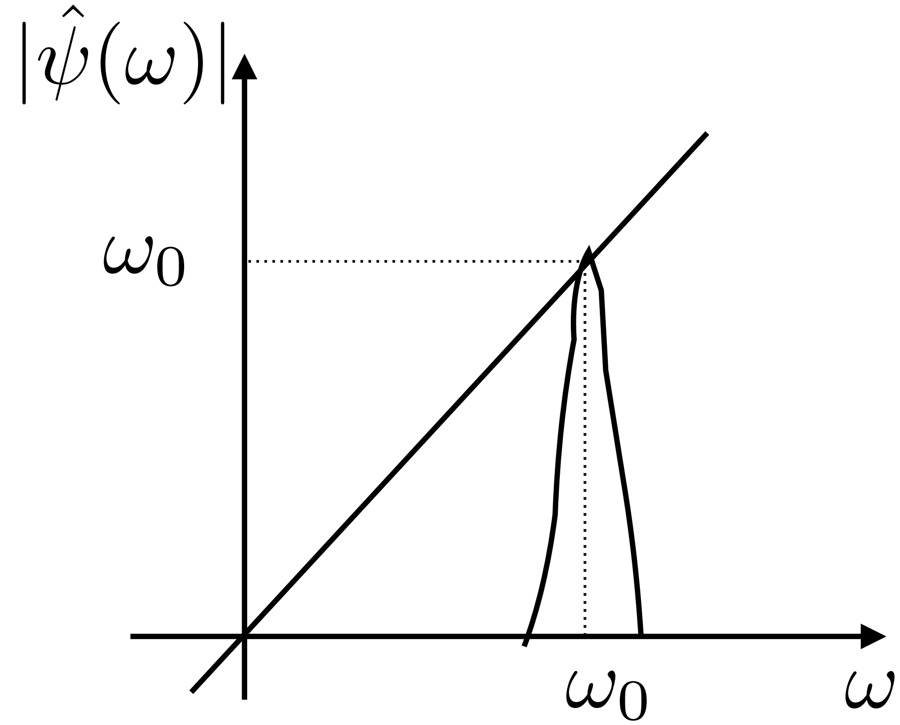
$L_\tau x(u) = x(u - \tau(u))$



- Analytic wavelets permit to build stable invariants to:

  - small translations by a:

$$\widehat{L_a x \star \psi}(\omega) = e^{i\omega^T a}\hat{x}(\omega)\hat{\psi}(\omega)$$
$$= \sum_n \frac{(i\omega^T a)^n}{n!}\hat{x}(\omega)\hat{\psi}(\omega)$$
$$\approx \sum_n \frac{(i\omega_0^T a)^n}{n!}\hat{x}(\omega)\hat{\psi}(\omega)$$
$$= e^{i\omega_0^T a}\widehat{x \star \psi}(\omega)$$



$\omega^T a\hat{\psi}(\omega) \approx \omega_0^T a\hat{\psi}(\omega)$

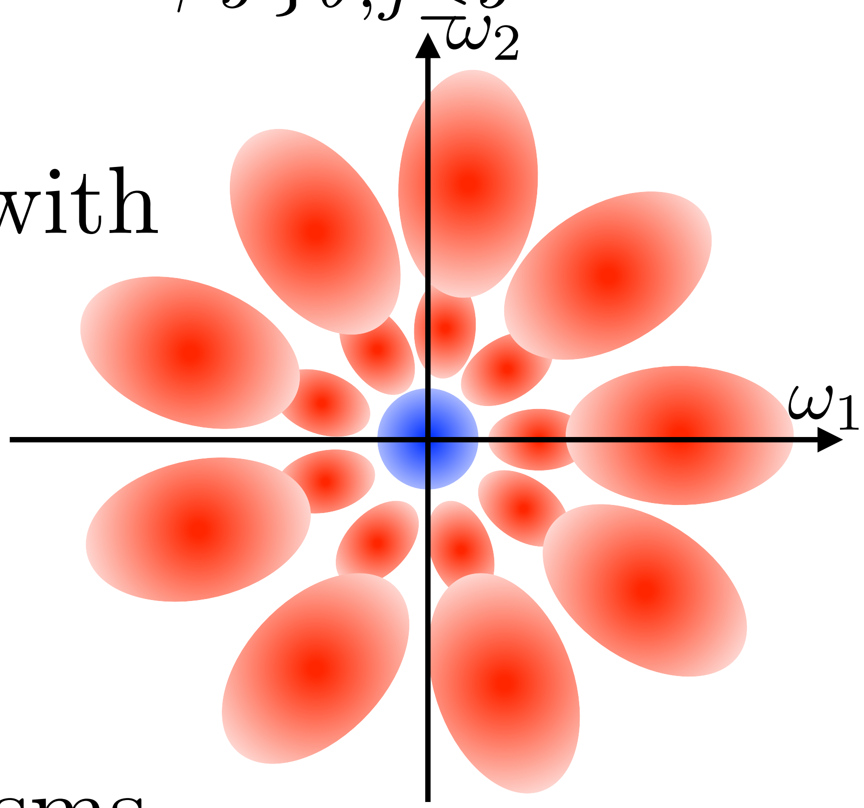
The variability corresponds to a phase!

  - small deformations:

$$\|(L_\tau x) \star \psi - L_\tau(x \star \psi)\| \leq C\nabla\|\tau\|_\infty$$

Wavelet Transform

- Wavelet transform: $Wx = \{x \star \psi_{j,\theta}, x \star \phi_J\}_{\theta, j \leq J}$



- Isometric and linear operator of L^2, with

$$\|Wx\|^2 = \sum_{\theta, j \leq J} \int |x \star \psi_{j,\theta}|^2 + \int x \star \phi_J^2$$

- Covariant with translation L_a:

$$WL_a = L_a W$$

- Nearly commutes with diffeomorphisms

$$\|[W, L_\tau]\| \leq C \|\nabla \tau\|$$

Ref.: Group Invariant Scattering, Mallat S

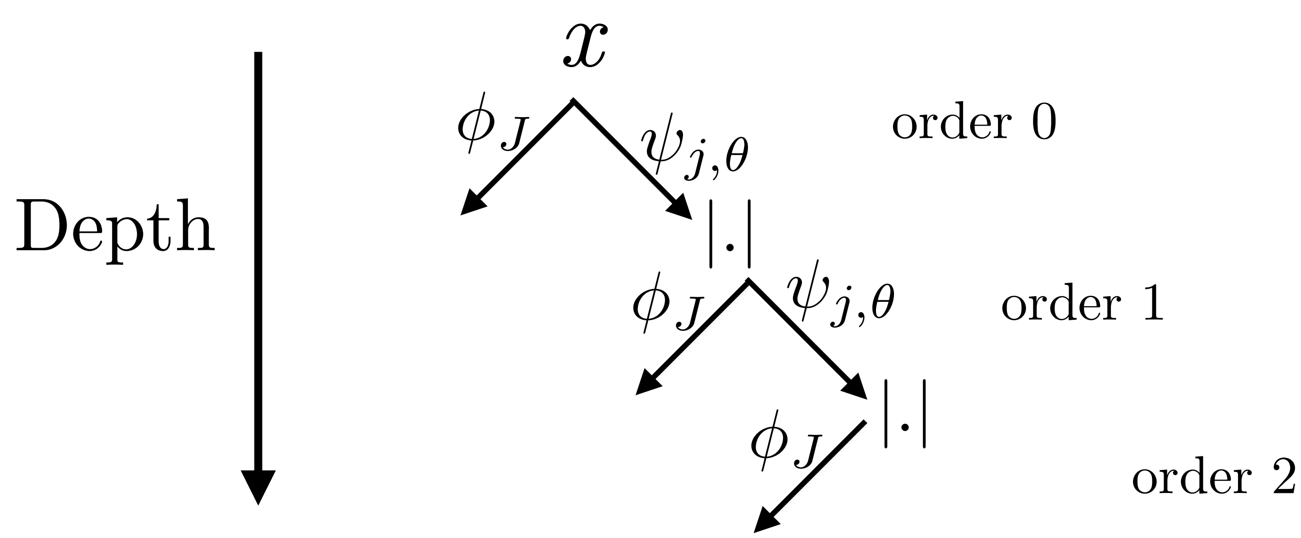
- A good baseline to describe an image!

Scattering Transform

- Scattering transform at scale J is the cascading of complex WT with modulus non-linearity, followed by a low pass-filtering:

Ref.: Group Invariant Scattering, Mallat S

$$S_J x = \{x \star \phi_J,$$
$$|x \star \psi_{j_1,\theta_1}| \star \phi_J,$$
$$||x \star \psi_{j_1,\theta_1}| \star \psi_{j_2,\theta_2}| \star \phi_J\}$$

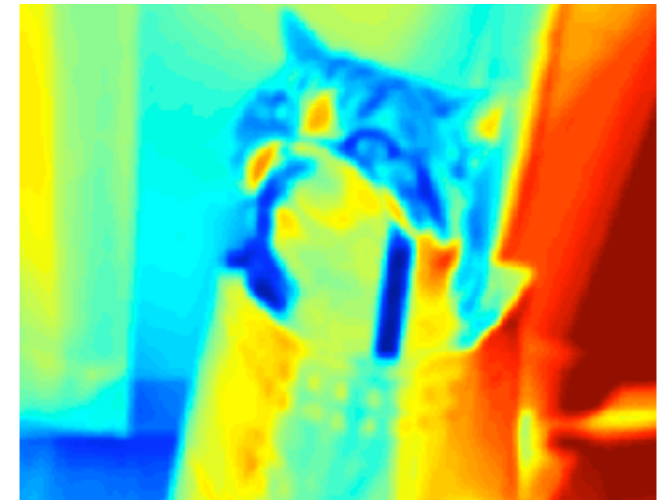


- Mathematically well defined for a large class of wavelets.

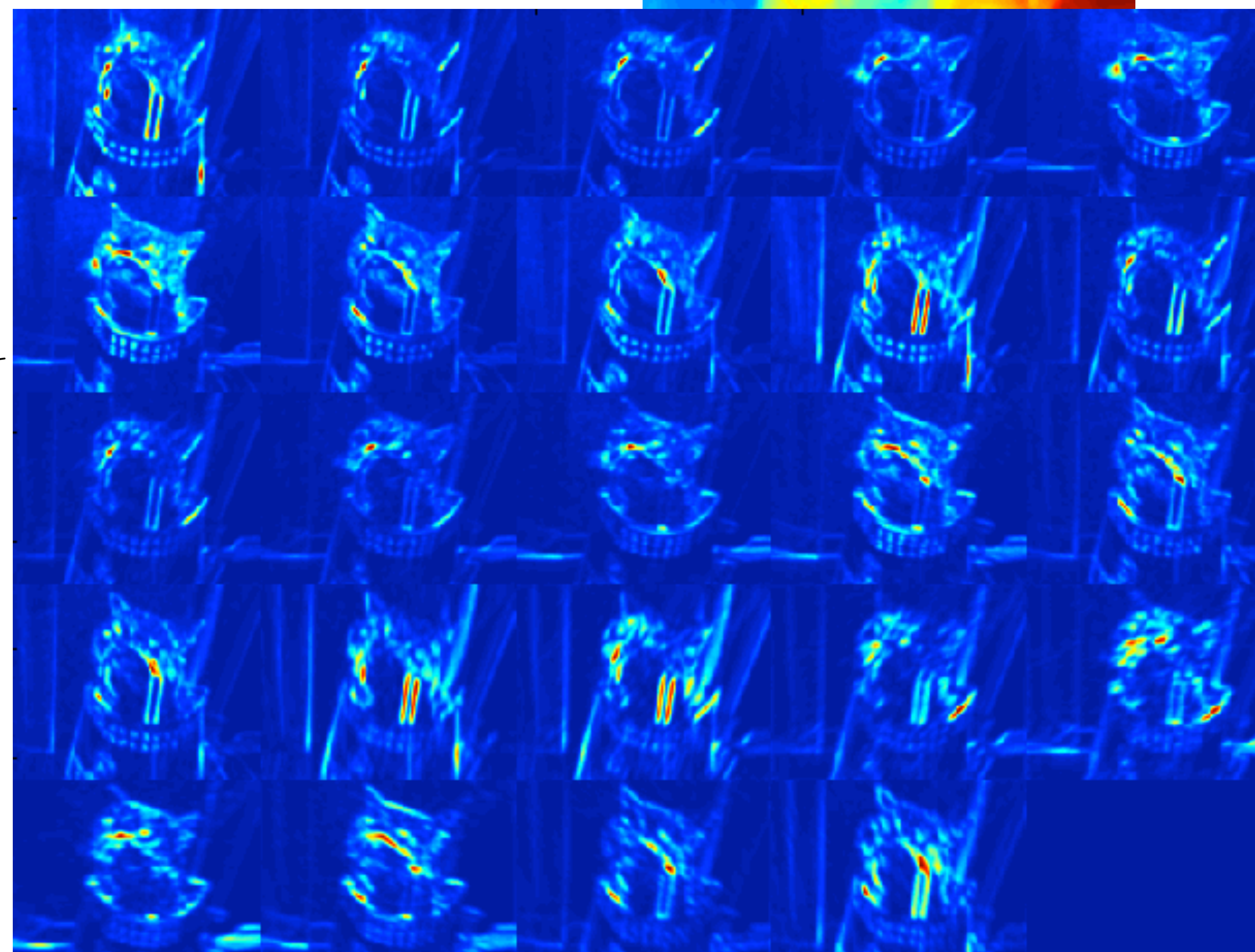
Feature map

x

$x \star \phi$



$|x \star \psi| \star \phi$

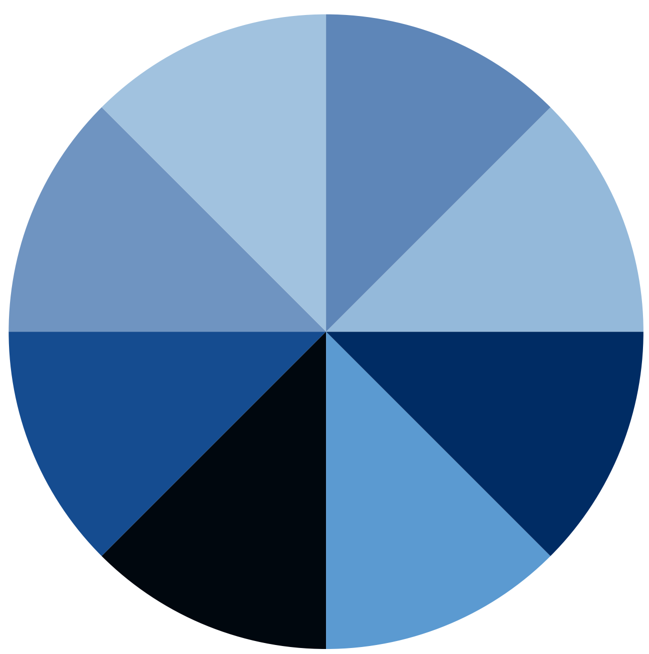


- Several features

1st order coefficients

Example of Scattering coefficients

On SIFT descriptors. . .



SIFT performs a histogram of gradient

$$h(\theta) = \sum_{\angle g \in [\theta, \theta+\eta]} \|g\|$$

Gradient

$$= \sum_g \|1_{\angle g \in [\theta, \theta+\eta]} g\|$$

Quantification. . .

$$\approx |x \star \psi_\theta| \star \phi$$

The averaging leads to a loss of information. . .

SIFT is very similar to an order 1 scattering!

Filter bank implementation of a Fast WT

Ref.: Fast WT, Mallat S, 89

- Assume it is possible to find h and g such that

$$\hat{\psi}_\theta(\omega) = \frac{1}{\sqrt{2}}\hat{g}_\theta(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2}) \quad \textbf{and} \quad \hat{\phi}(\omega) = \frac{1}{\sqrt{2}}\hat{h}(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2})$$

- Set:

$$x_j(u,0) = x \star \phi_j(u) = h \star (x \star \phi_{j-1})(2u) \quad \textbf{and}$$

$$x_j(u,\theta) = x \star \psi_{j,\theta}(u) = g_\theta \star (x \star \phi_{j-1})(2u)$$

- The WT is then given by $Wx = \{x_j(.,\theta), x_J(.,0)\}_{j\leq J,\theta}$

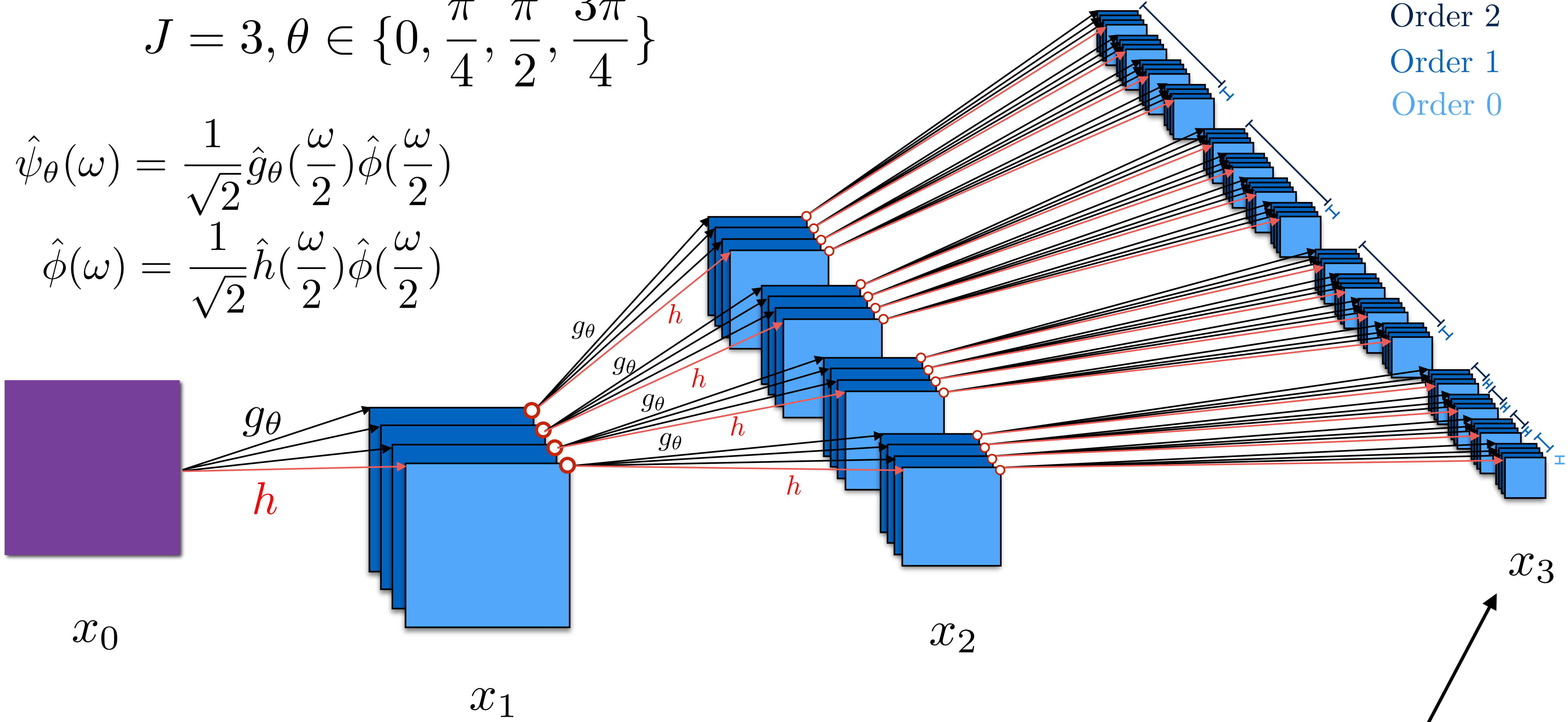
- A WT can be interpreted as a deep cascade of linear operator, which is approximatively verified for the Gabor Wavelets.

Scattering as a CNN

$$J = 3, \theta \in \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$$

$$\hat{\psi}_\theta(\omega) = \frac{1}{\sqrt{2}}\hat{g}_\theta(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2})$$

$$\hat{\phi}(\omega) = \frac{1}{\sqrt{2}}\hat{h}(\frac{\omega}{2})\hat{\phi}(\frac{\omega}{2})$$



○ Modulus

$$h \geq 0$$

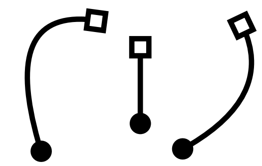
Scattering coefficients are only at the output

Ref.: Deep Roto-Translation Scattering for Object Classification. EO and S Mallat
Recursive Interferometric Representations, S Mallat

Properties of a Scattering Transform

Deformations

$$L_\tau x(u) = x(u - \tau(u))$$



- Scattering is stable:

$$\|S_J x - S_J y\| \leq \|x - y\|$$

- Linearize small deformations:

$$\|S_J L_\tau x - S_J x\| \leq C \|\nabla \tau\| \|x\|$$

- Invariant to local translation:

$$|a| \ll 2^J \Rightarrow S_J L_a x \approx S_J$$

Ref.: Group Invariant Scattering, Mallat S

- For $\lambda, u, S_J x(u, \lambda)$ is covariant with $SO_2(\mathbb{R})$:

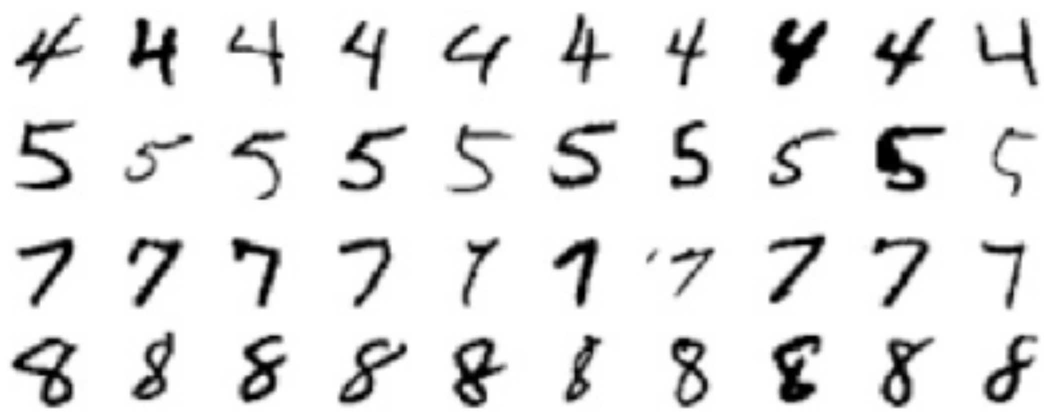
$$\text{if } \forall u \forall g \in SO_2(\mathbb{R}), g.x(u) \triangleq x(g^{-1}u) \text{ then,}$$

$$S_J(g.x)(u, \lambda) = S_J x(g^{-1}u, g^{-1}\lambda) \triangleq g.S_J x(u, \lambda)$$

A successful representation in vision

Ref.: Invariant Convolutional Scattering Network, J. Bruna and S Mallat

- Successfully used in several applications:
  - Digits



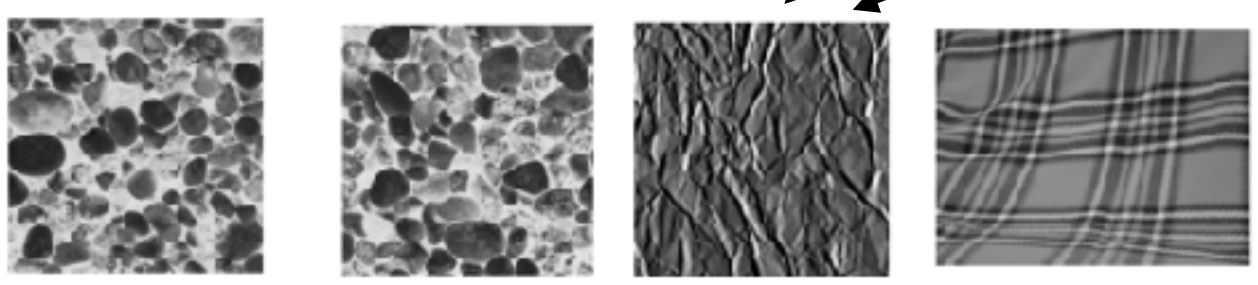
All variabilities are known

Small deformations +Translation

Rotation+Scale

  - Textures

Ref.: Rotation, Scaling and Deformation Invariant Scattering for texture discrimination, Sifre L and Mallat S.



- The design of the scattering transform is guided by the euclidean group

- To which extent can we compete with other architectures on more complex problems (e.g. variabilities are more complex)?

Loss of information?

x

Ref.: Bruna and Mallat

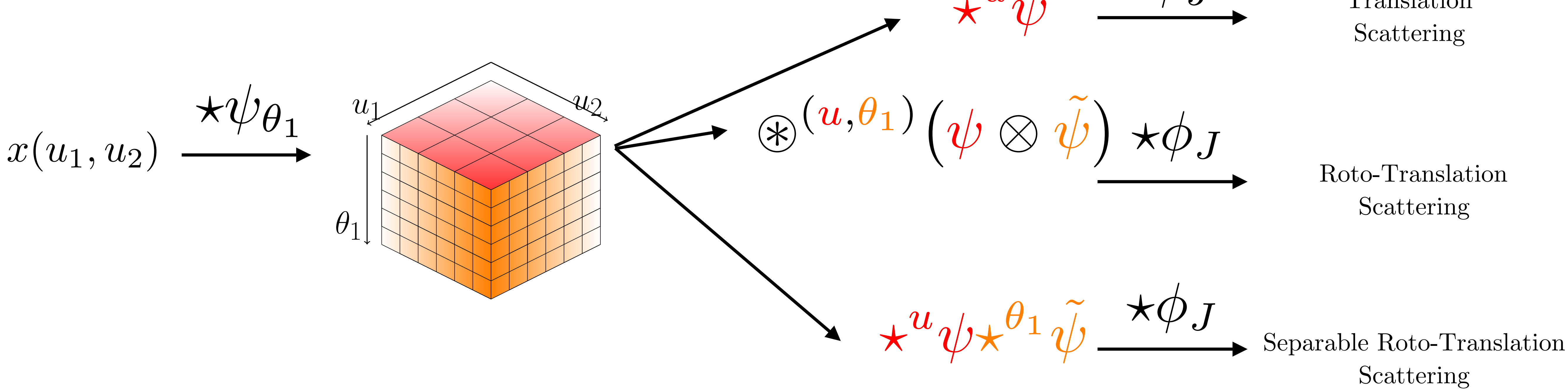
$$\arg\inf_{y} \|S_3 x - S_3 y\|$$

invariance up to 2^3 pixels

$\tilde{y}$

Separable Roto-translation Scattering

$\psi, \tilde{\psi}$ spatial, angular wavelets



- Simplification of the Roto-translation scattering
- Discriminates angular variabilities thanks to a wavelet transform along θ_1 (no averaging!)
- We combine it with Gaussians SVM

Ref.: Deep Roto-Translation Scattering for Object Classification. EO and S Mallat

Scattering on Complex Image Classification

Ref.: Deep Roto-Translation Scattering for Object Classification. EO and S Mallat

Dataset	Type	Accuracy
Caltech101	Scattering	80
	Supervised	*93*
CIFAR100	Scattering	57
	Supervised	*82*

No learning

CALTECH

10^4 images
101 classes
256×256 color images

Can we fill the gap by incorporating supervision?

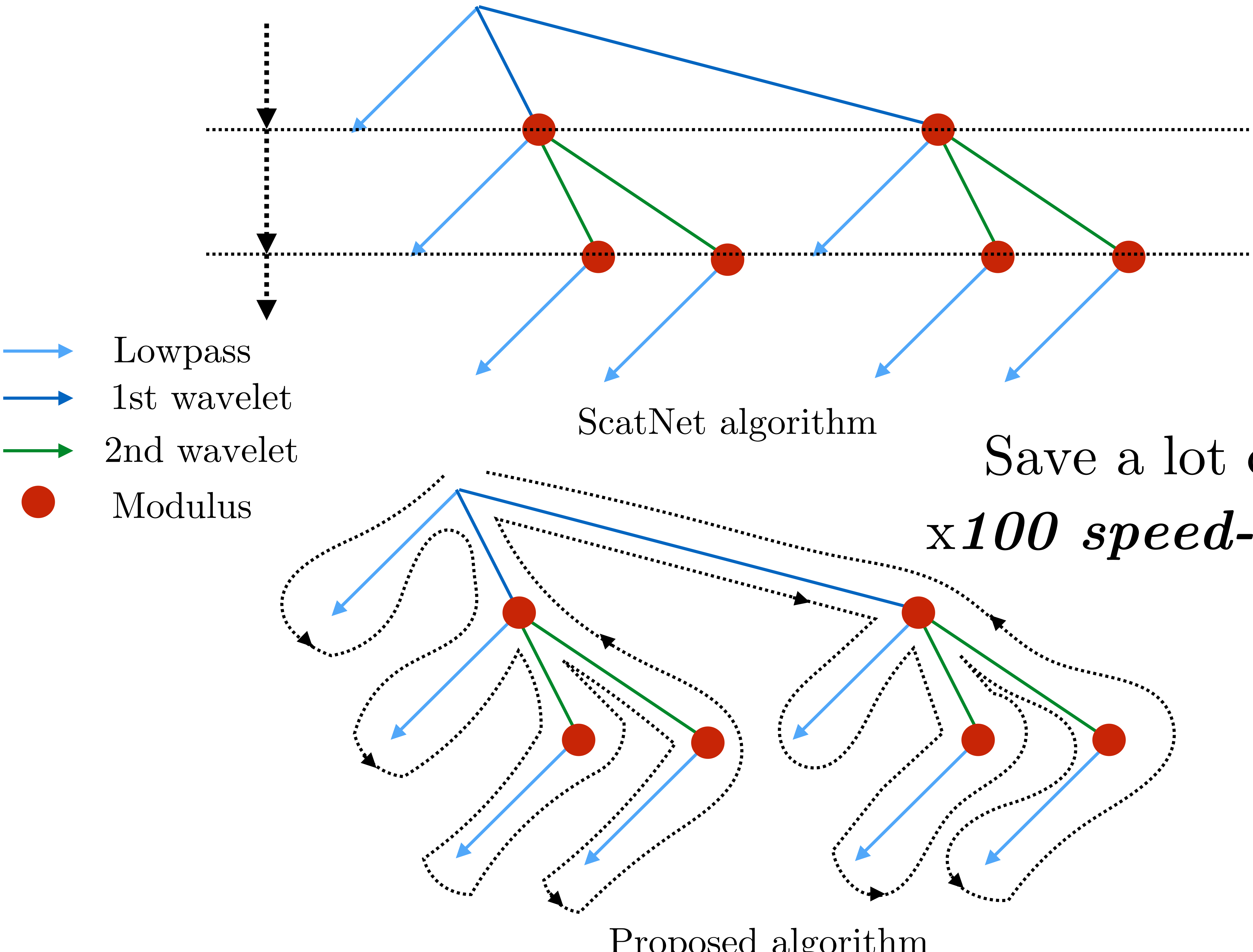
Do we want to learn the filters?

CIFAR

5.10^4 images
100 classes
32×32 color images

Scaling scattering on GPUs

Ref.: Thesis, EO



Save a lot of memory!

x***100 speed-up on GPU***

Computation time

$x \in \mathbb{R}^{3\times N^2}$

$S_J x$

Invariance up to 2^J

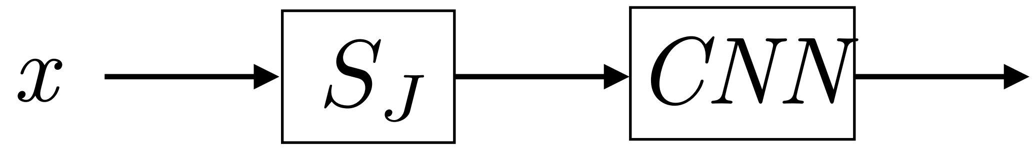
Save a lot of memory too

Ref.: Scaling the Scattering Transform: Deep Hybrid Networks
EO, E Belilovsky, S Zagoruyko

N	J	ScatNetLight MATLAB (CPU)	PyScatWave PyTorch (CUDA)
32	2	19	0.2
32	4	101	1.5
128	2	125	2.0
128	4	406	4.2
256	2	1250	5.5

GPU: GTX 1080

Scattering meets Neural Networks



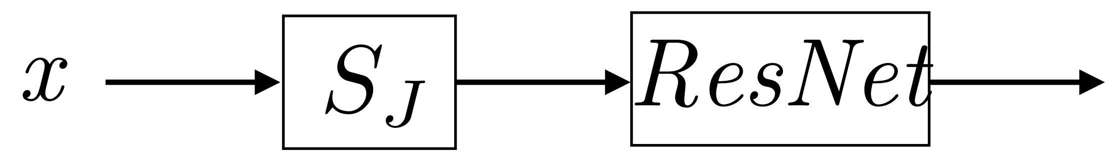
Ref.: Scaling the Scattering Transform: Deep Hybrid Networks
EO, E Belilovsky, S Zagoruyko

- We input raw Scattering coefficients in CNNs.
- All engineering tricks are kept identical: random data augmentation, learning rate schedule, regularization...

Ex.:

- Scattering transform is covariant with the natural symmetries group: structuring $\mathbb{R}^2$ by incorporating $\mathbb{R}^2 \ltimes SL_2$.

ImageNet benchmarking



Ref.: Scaling the Scattering Transform: Deep Hybrid Networks
EO, E Belilovsky, S Zagoruyko

- State-of-the-art result on Imagenet2012:

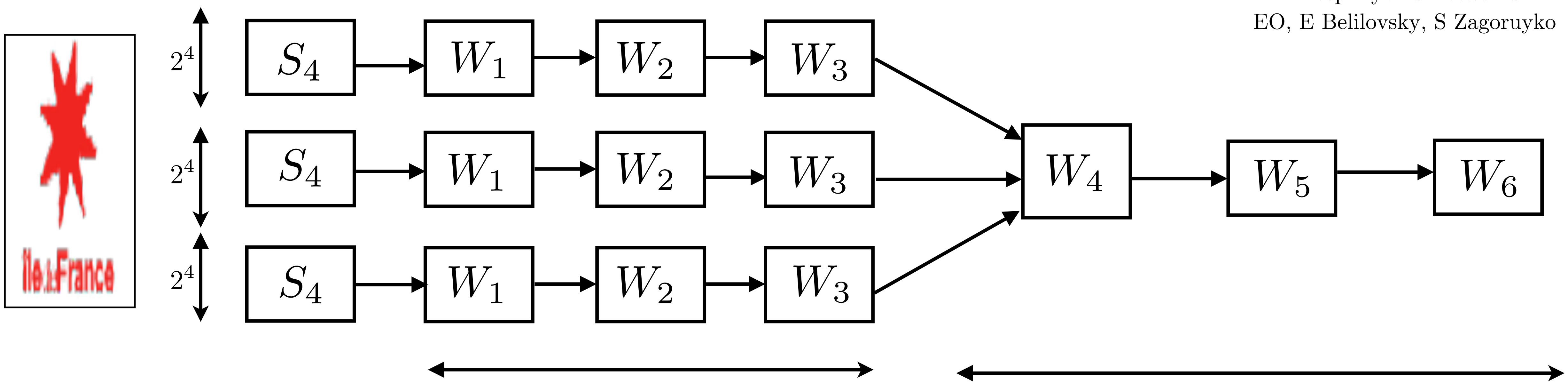
	Top 1	Top 5	#params
Scat+Resnet-10	69	90	12.8M
VGG-16	69	90	138M
ResNet-18	69	89	11.7M
ResNet-200	**79**	**95**	64.7M

- Demonstrates no loss of information + less layers (10 vs 18)

- Scattering + 5-layers perceptron on CIFAR: 85% acc. (SOTA w.r.t. non-convolutional learned representation)

Shared Local Encoder

Ref.: Scaling the Scattering Transform:
Deep Hybrid Networks
EO, E Belilovsky, S Zagoruyko



1×1 convolution (encoder)

Fully Connected Layers (classifier)

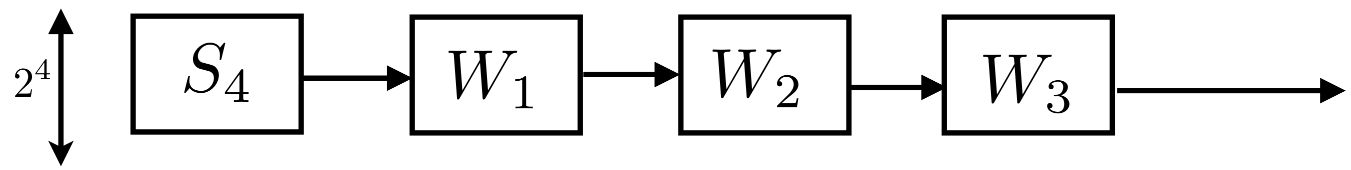
Is it really a classifier?

	Top 1	Top 5
Scat+SLE	**57**	80
FV+FCs	56	79
FV+SVM	54	75
AlexNet	56	**81**

- AlexNet performances with 1x1 conv
- Outperform unsupervised encoders based on SIFT + Fisher Vectors(FV)

A local descriptor for classification

- We analyse the scattering's encoder, which is a descriptor on neighbourhood of size $2^4 \times 2^4$ pixels:



- Good **transfer learning** performance on Caltech101(83%)! Analog to previous reported performance.

Open question: Could this representation generalise to other vision tasks? (e.g. scene matching)

Understanding SLE

- The rotation group SO_2 acts of θ on the scattering coefficient via a translation L_θ, it thus acts on the first layer W_1:

$$Sr_{-\theta}x = L_\theta Sx \Rightarrow W_1 Sr_{-\theta}x = W_1 L_\theta Sx$$

- Atoms' index of W_1 are structured by the order 0, 1, 2 of S_4:

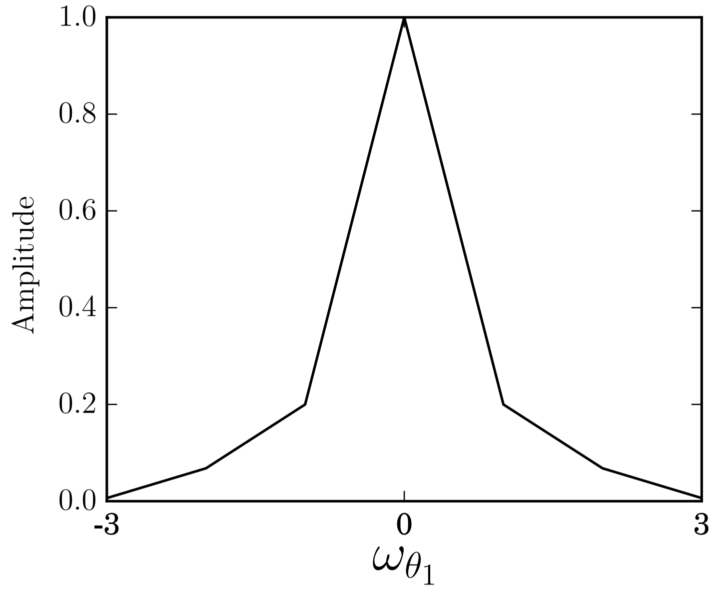
$$(W_1 S_4)_k = w_{0,k}(x \star \phi_j) + \sum_{j_1,\theta_1} w_{(j_1,\theta_1),k}(|x \star \psi_{j_1,\theta_1}| \star \phi_j) + \sum_{j_2,j_1,\theta_1,\theta_2} w_{(j_1,j_2,\theta_1,\theta_2),k}(||x \star \psi_{j_2,\theta_2}| \star \psi_{j_1,\theta_1}| \star \phi_j)$$

→ **What is the nature of the recombination?**

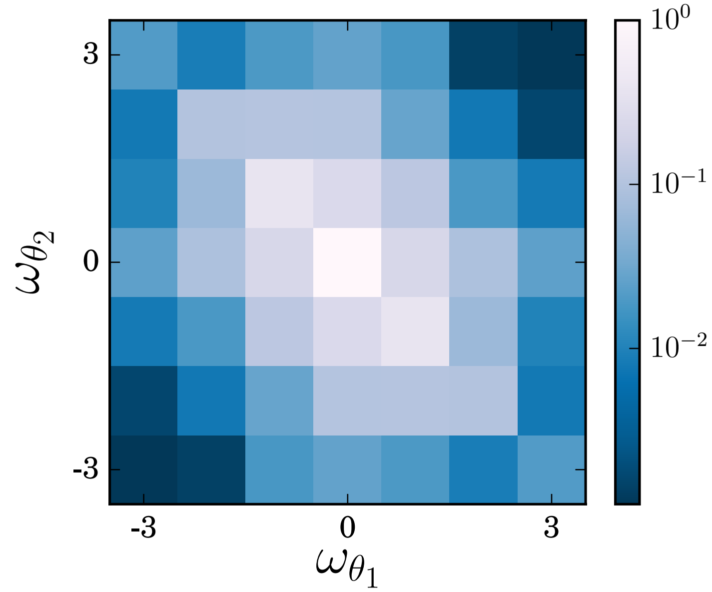
Fourier along θ_1: $\hat{w}_{(j_1,\omega_{\theta_1}),k} = \mathcal{F}^{\theta_1}(w_{(j_1,.),k})(\omega_{\theta_1})$

Fourier along (θ_1, θ_2): $\hat{w}_{(j_1,j_2,\omega_{\theta_1},\omega_{\theta_2}),k} = \mathcal{F}^{(\theta_1,\theta_2)}(w_{(j_1,j_2,.,.),k})(\omega_{\theta_1}, \omega_{\theta_2})$

Explicit invariance to rotation



Ω_1

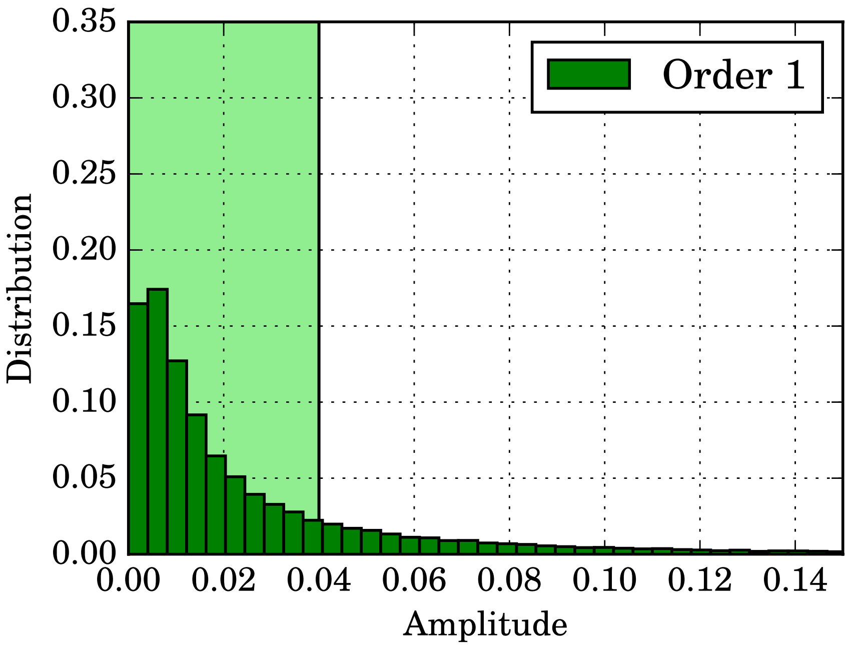


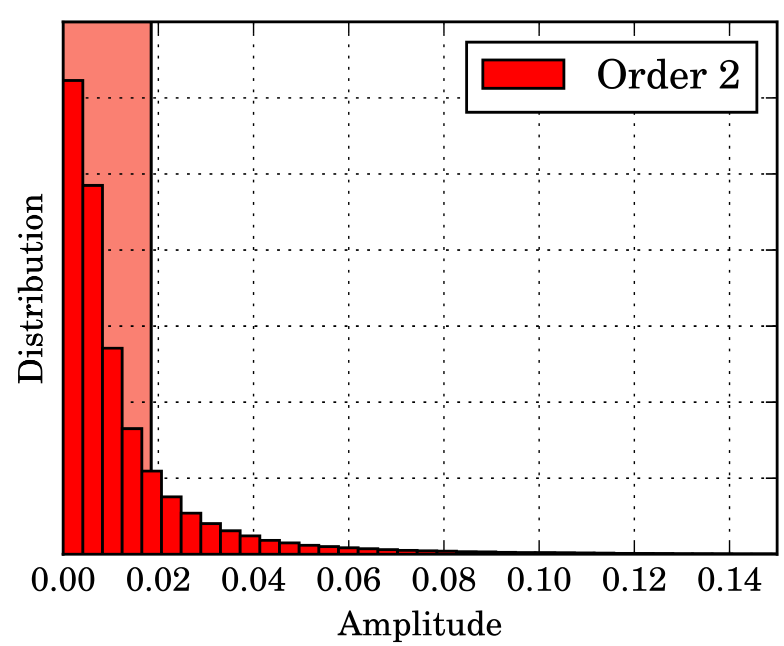
Ω_2

$$\Omega_1(\omega_{\theta_1}) = \sum_{k,j_1} |\hat{w}_{(j_1,\omega_{\theta_1}),k}|^2$$

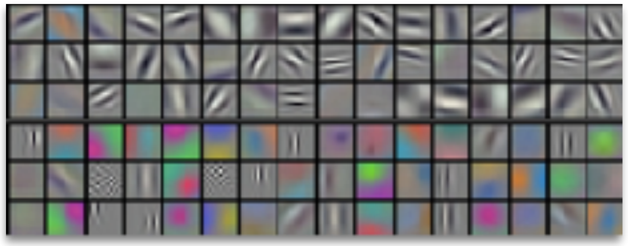
$$\Omega_2(\omega_{\theta_1}, \omega_{\theta_2}) = \sum_{k,j_1,j_2} |\hat{w}_{(j_1,j_2,\omega_{\theta_1},\omega_{\theta_2}),k}|^2$$

- Invariance to rotation is explicitly learned.





method: similar to AlexNet first layer analysis

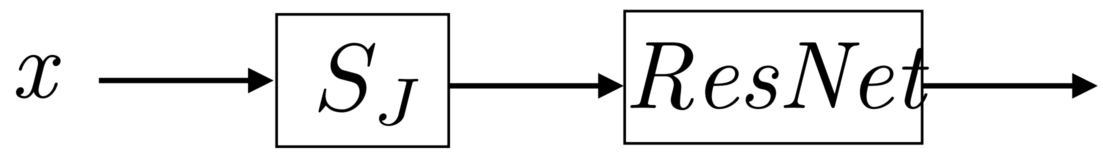


Fourier basis sparsifies the operator!

- Thresholding 80% of the coefficients in Fourier: 2% acc. loss

Open question: Can we find more complex invariance than rotation?

Learning with few samples



Ref.: Scaling the Scattering Transform: Deep Hybrid Networks
EO, E Belilovsky, S Zagoruyko

- We show incorporating **geometrical invariants help learning.**(with limited adaptation)

- State-of-the-art results on STL10 and CIFAR10:

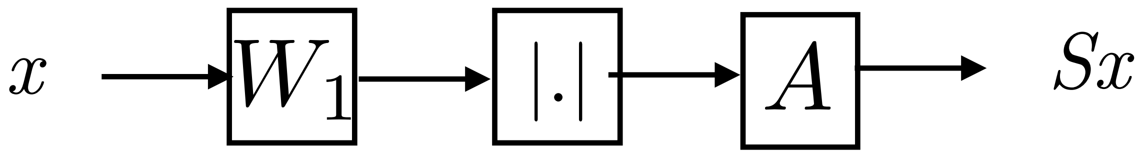
STL10: 5k training, 8k testing, 10 classes
+100k unlabeled(not used!!)

	Acc.
Scat+ResNet	**76**
Supervised	70
Unsupervised	76

Cifar10, 10 classes
keeping 100, 500 and 1000 samples
and testing on 10k

#train	100	500	1000	Full
WRN 16-8	35	47	60	**96**
VGG 16	26	47	56	93
Scat+ResNet	**38**	**55**	**62**	93

Compressing the input for CNNs



$$\frac{\dim Sx}{\dim x} = \frac{1+|\Theta|J}{2^{2J}}$$

Ref.: Compressing the input for CNNs with the first-order Scattering Transform
EO, E Belilovsky, S Zagoruyko, M Valko

- For $J > 3$, one has a compression.

- We noticed that this value was the most efficient for reconstructing natural images.

- Simple model to understand why:

$x_\Sigma(u) \propto e^{-u^T \Sigma u}$ then for ψ Gabor:

$$|x_\Sigma \star \psi|(u) \propto x_\Sigma \star |\psi|(u)$$

smooth

x $J{=}3$ $J{=}4$

Advantage of "compression"

Works for detection + allows to process more images:

Architecture	Classification Models: Speed (64 images)	Classification Models: Max im. ImageNet	Detection Models: Speed (4 images)	Detection Models: Max im. Coco
Order 1 + ScatResNet-50	0.072	175	0.073	9
ResNet-50	0.095	120	0.104	7
ResNet-101	0.158	70	0.182	2

Coco

Architecture	mAP
Faster-RCNN Order 1 + ScatResNet-50	32.2
Faster-RCNN ResNet-50 (ours)	31.0
Faster-RCNN ResNet-101 (ours)	34.5
Faster-RCNN VGG-16 [34]	29.2
Detectron [40]	41.8

Pascal VOC7

Architecture	mAP
Faster-RCNN Order 1 + ScatResNet-50 (ours)	73.3
Faster-RCNN ResNet-50 (ours)	70.5
Faster-RCNN ResNet-101 (ours)	72.5
Faster-RCNN VGG-16 [34]	70.2

Scattering with phase

Phase Harmonic Correlations and Convolutional Neural Networks, Mallat, Zhang and Rochette

Let $\rho(u) = \max(u, 0)$ and $\psi_{j,\theta,\alpha} = \text{Real}(e^{-i\alpha}\psi_{j,\theta}(u))$

- Adding redundant representation which captures interactions between different scales.
- Better reconstruction properties/slight classification improvement: but much bigger representation.

If $(j, \theta) \neq (j', \theta')$ then:

$$\int \rho(x \star \psi_{j,\theta,\alpha})\rho(x \star \psi_{j',\theta',\alpha'})\, du \neq 0$$

$$\int |x \star \psi_{j,\theta}||x \star \psi_{j',\theta'}|\, du \approx 0$$

Scattering on Graph

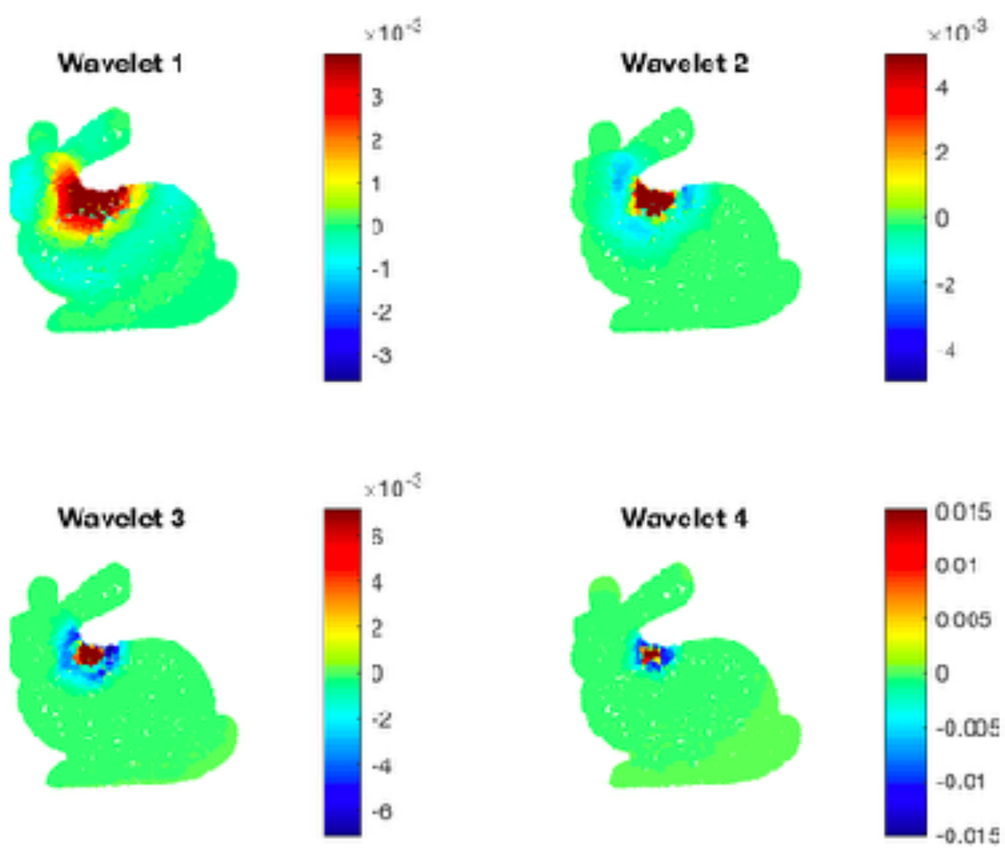
- Similar ideas hold for graphs $(E, \mathcal{G})$.
- Wavelets use require a good notion of duality, which is given by the graph Laplacian.

Ref.: Wavelet Scattering on Graphs, F G, Bruna

- Laplacian in the Euclidean case:

$$\widehat{\Delta x}(\omega) = -\|\omega\|^2 \hat{x}(\omega)$$

We lose the angle…



Can we introduce more structures?

- Convolutions along the spatial angles permit to build more robust invariants along SO_2 :

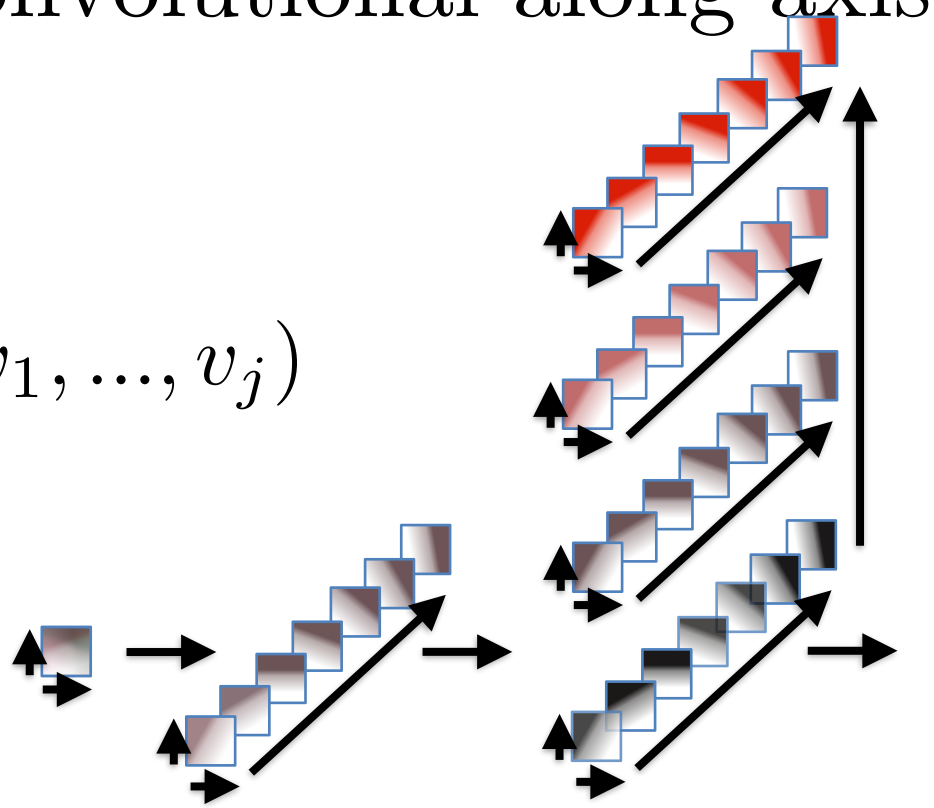
$$|x \star \psi_{j_1,\cdot}| \circledast^{(u,\theta_1)} (\psi_{j_2,\theta_2} \otimes \tilde{\psi}_k)(u,\theta_1)$$

Ref.: Deep Roto-Translation Scattering for Object Classification. EO and S Mallat
Rotation, scaling and deformation invariant scattering for texture discrimination, L Sifre and S Mallat

- Extension: Hierarchical CNN that is convolutional along axis channel, recursively defined via:

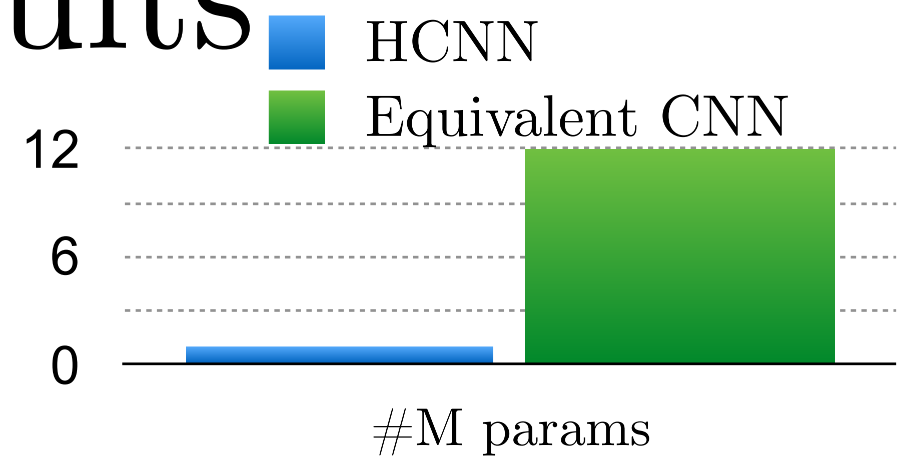
$$x_{j+1}(v_1,...,v_j,v_{j+1}) = \rho_j(x_j \star^{v_1,...,v_j} \psi_{v_{j+1}})(v_1,...,v_j)$$

Ref.: Hierarchical Attribute CNNs, Jacobsen et al.

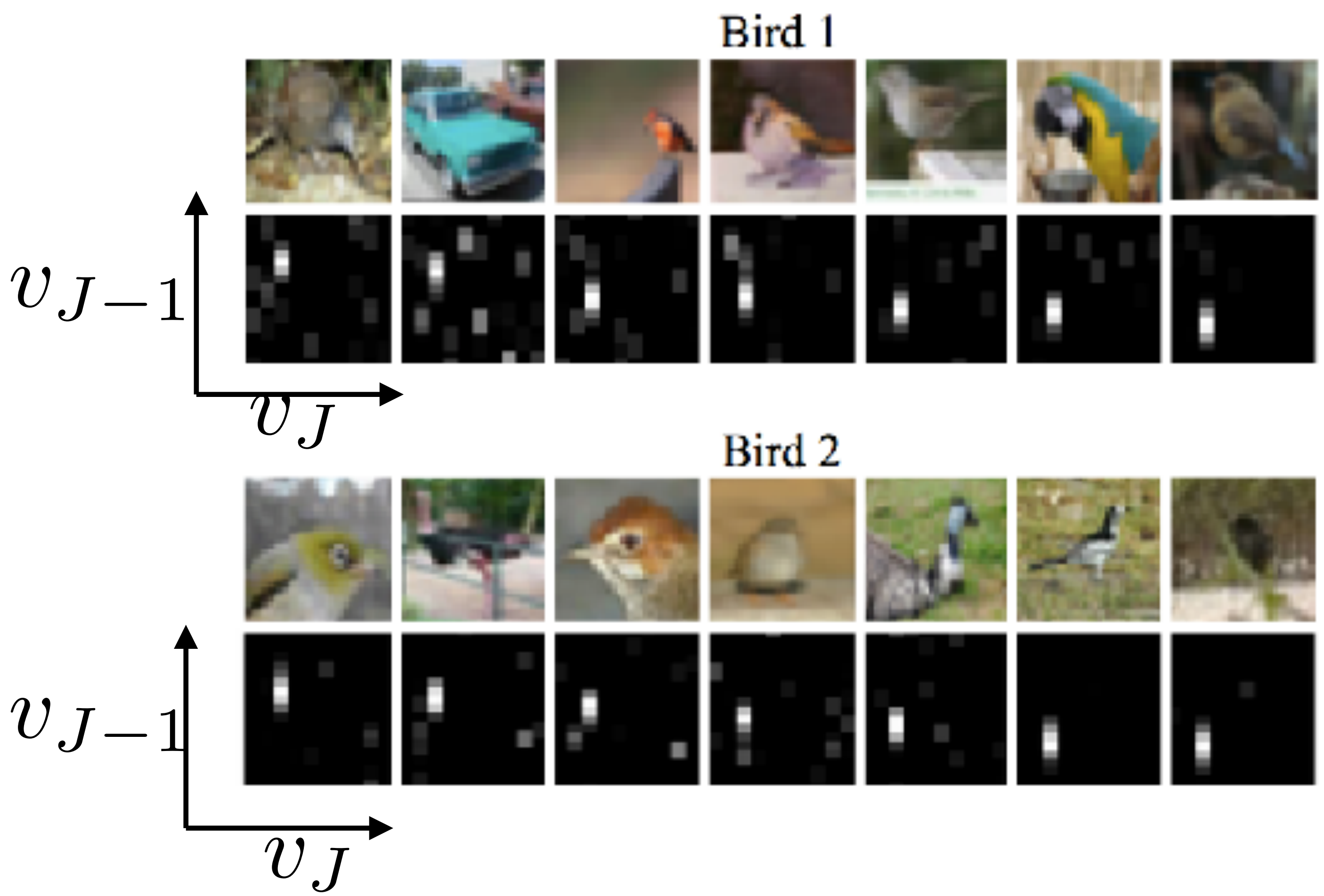


Hiearchical CNN: numerical results

We demonstrate
a reduction in #param
while 91% on CIFAR10



Translations are present in the last layer $x_J(v_{J-1}, v_J)$



But not in the previous layers

Incorporating more structures?

Modelization issue?

Ref.: Multiscale Hierarchical Convolutional Networks
J Jacobsen, EO, S Mallat, AWM Smeulders

kymat.io

- A fast software for TensorFlow, PyTorch and NumPy.
- Lot of examples, work out of the box.
- Differentiable Scattering, multi-GPU. . .
- A significant team of developer is involved!

Conclusion

- Scattering Transform is a strong baseline based purely on geometric priors.
- We propose competitive models, software…

More about my research: https://edouardoyallon.github.io/

Thank you!