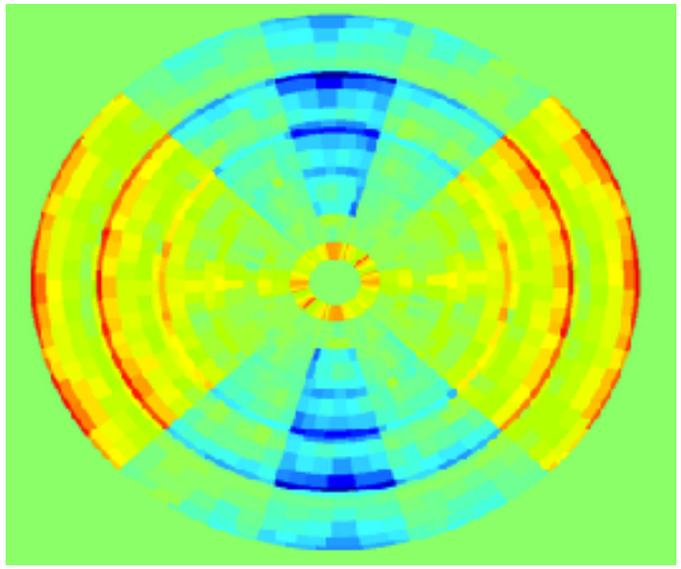
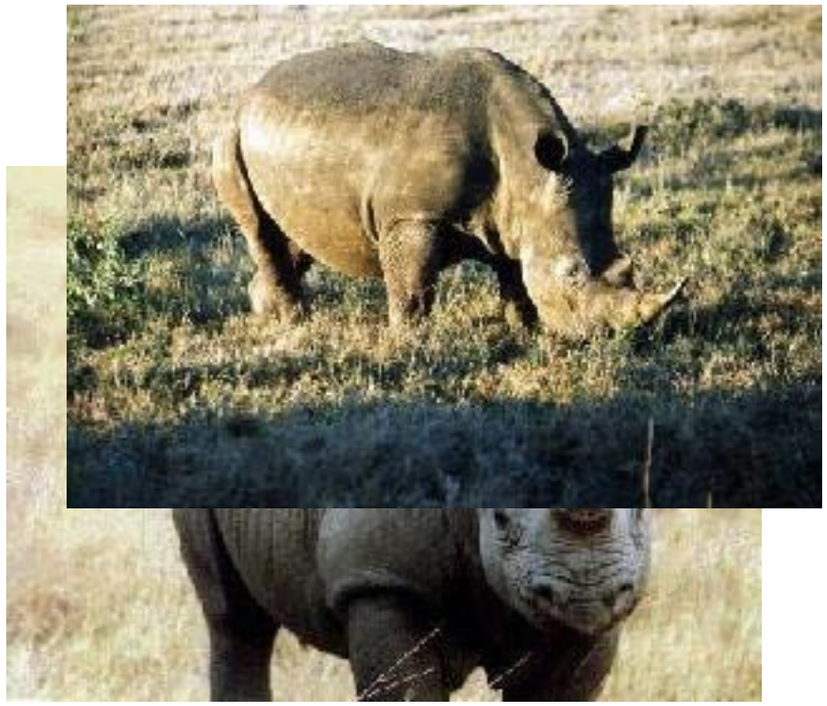
# Analyzing and Introducing Structures in Deep Convolutional Neural Networks

Edouard Oyallon



# 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"

Geometric variability

High variance: how to reduce it?

Groups acting on images:

translation, rotation, scaling

What is the nature of other sources of variability?

# Deep learning: Technical breakthrough

- Deep learning has permitted to solve a large number of task that were considered as extremely challenging for a computer.
  Ex.: Vision, Game of Go, Speech recognition, Artistic style transfer...

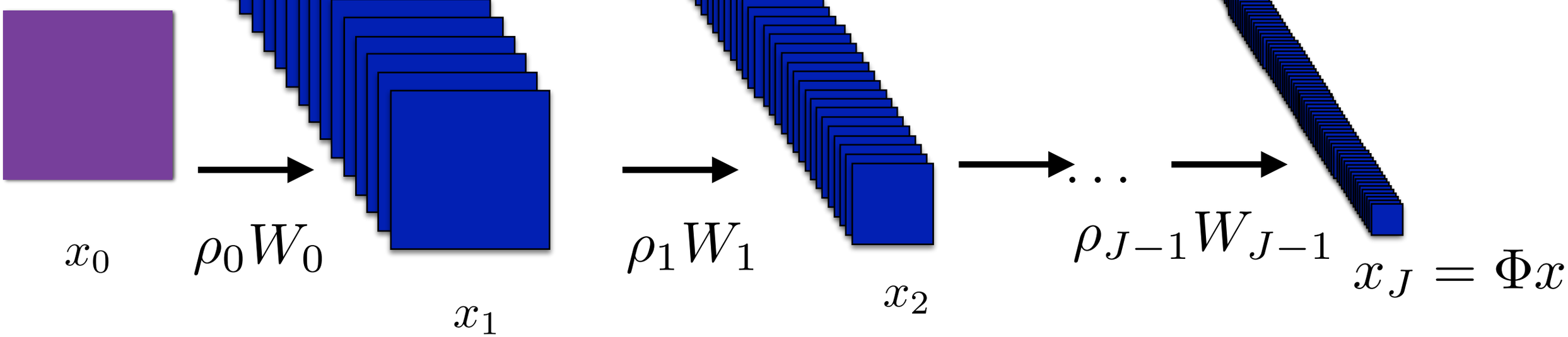
- The technique is generic and its success implies that it reduces large sources of variability.

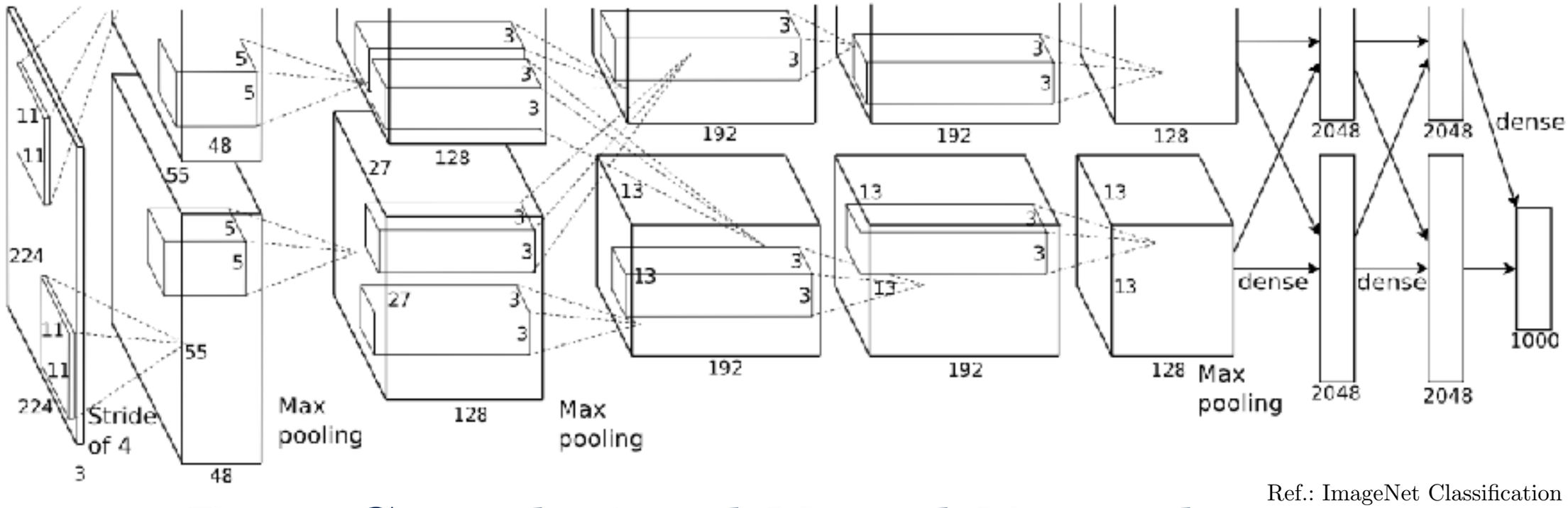
- How, why?

non-linear operator

convolutional operator

$$x_{j+1} = \rho_j W_j x_j$$




# Deep Convolutional Neural Network

Ref.: ImageNet Classification with Deep Convolutional Neural Networks. A Krizhevsky et al.

# Why mathematics about deep learning are important?

- **Pure black box**. Few mathematical results explain the cascade.
  Many rely on a low dimensional "manifold hypothesis":
  Variability is too high in images.

- **No stability results**. It means that "small" variations of the inputs might have a large impact on the system. And this happens.

Ref.: Intriguing properties of neural networks.
C. Szegedy et al.

- **No model of the data.** We do not understand what is the nature of the sources of variabilities that are reduced.

Ref.: Understanding deep learning requires rethinking generalization
C. Zhang et al.

- Shall we learn each layer from scratch? (**geometric priors?**) The deep cascade makes features hard to interpret

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

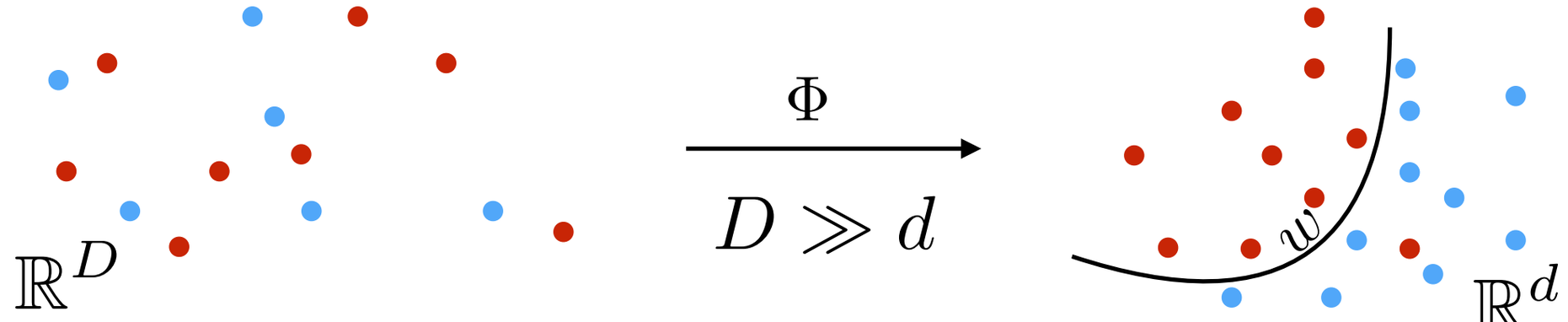
# Overview

Problem: how can we **incorporate** and **analysing structures** into deep networks?

1. Scattering for complex image classification
2. Beyond scattering: analyzing a supervised CNN
3. Best of two worlds: scattering and CNN
4. Beyonds euclidean group: Hiearchical CNN

# Fighting the curse of dimensionality

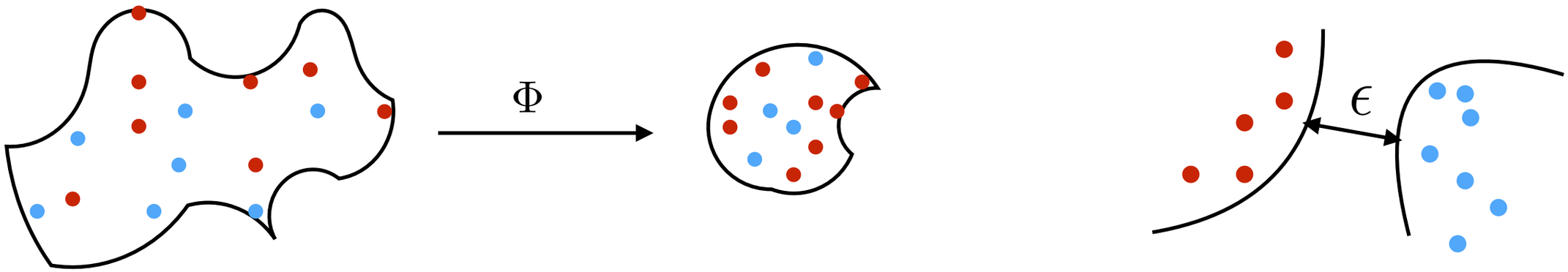
- Objective: building a representation $\Phi x$ of $x$ such that a simple (say euclidean) classifier $\hat{y}$ can estimate the label $y$:




- Designing $\Phi$: must be regular with respect to the class:

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

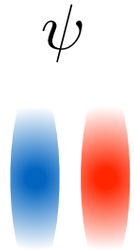
- Necessary dimensionality reduction and separation to break the curse of dimensionality:

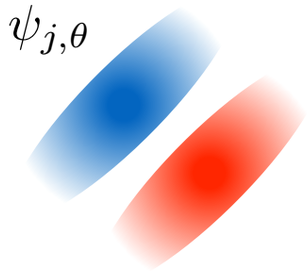


# Wavelets

- $\psi$ 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:

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

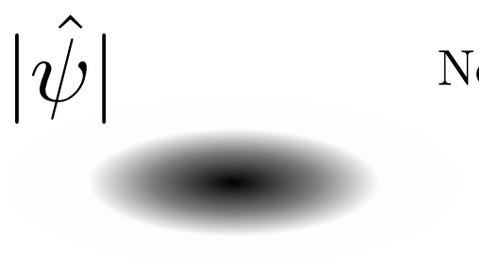



- Design wavelets selective to rotation variabilities.

Isotropic

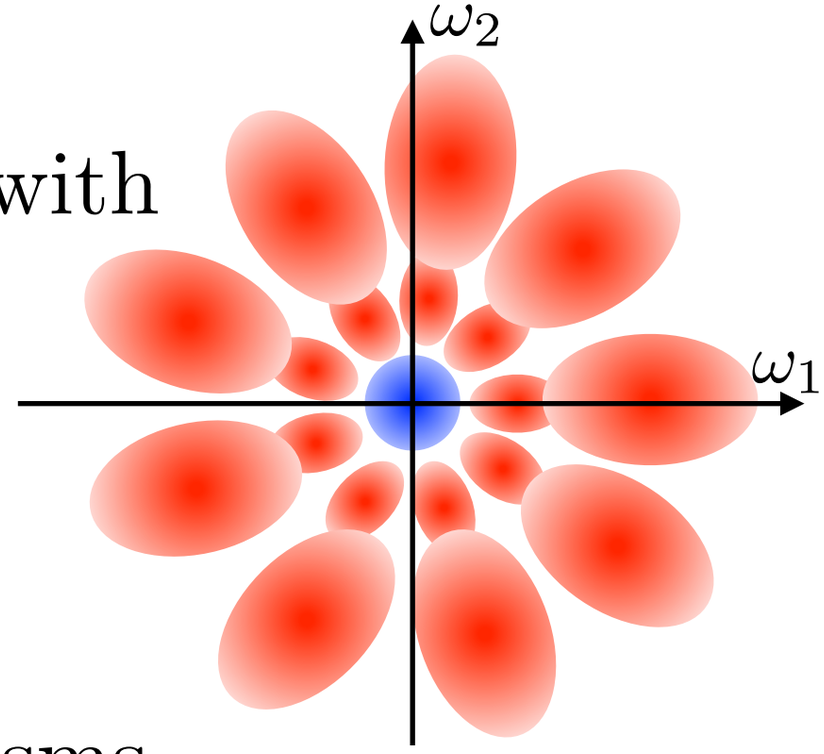

VS




Non-Isotropic

# 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

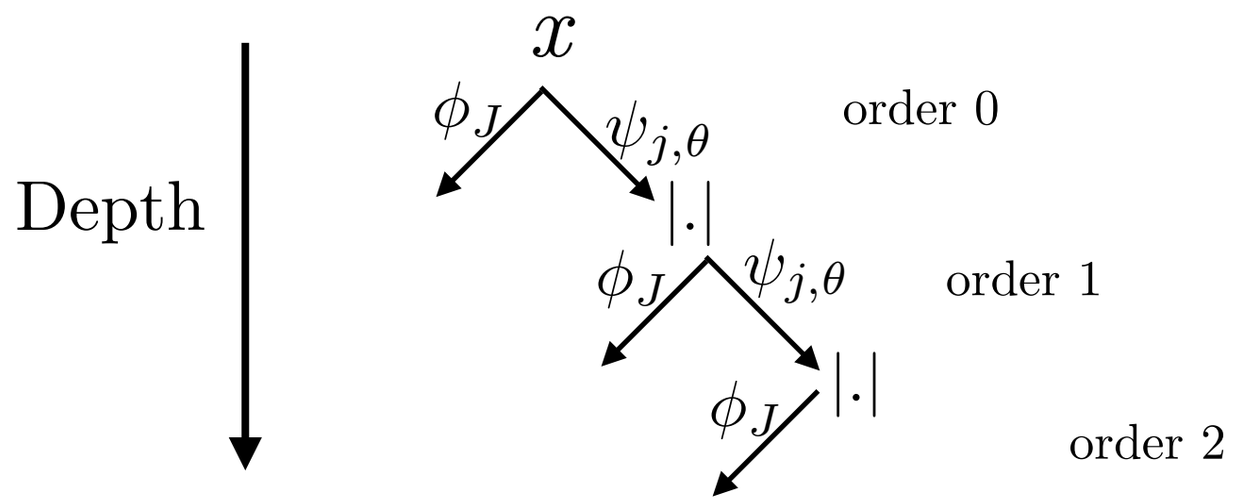
- A good baseline to describe an image!

# 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

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



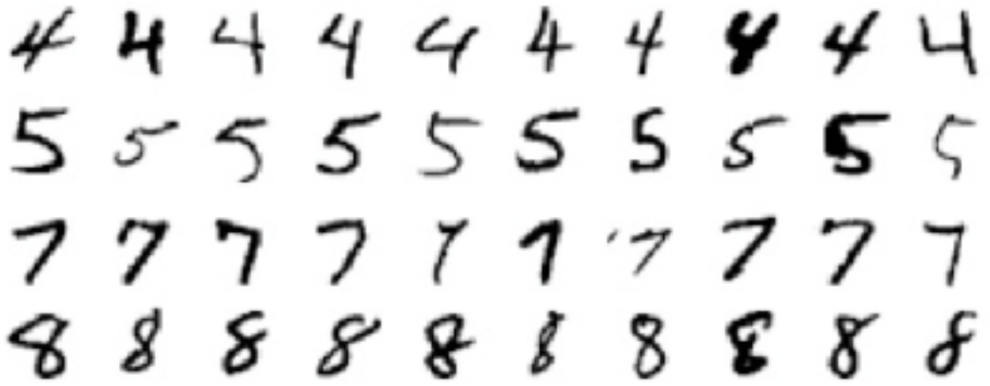

- Mathematically well defined for a large class of wavelets.

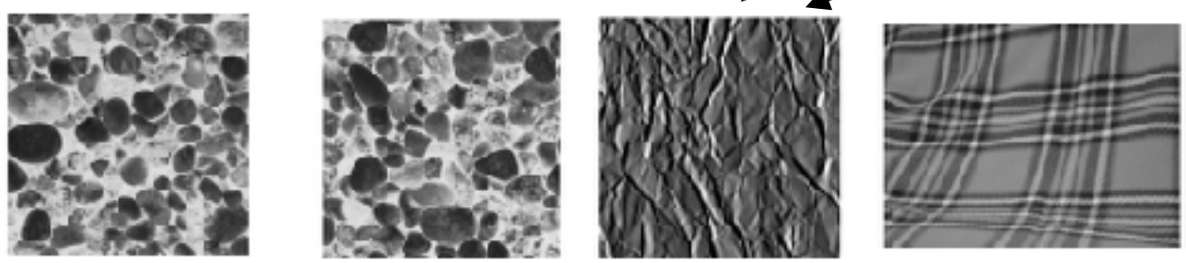
# A successful representation in vision

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

- Successfully used in several applications:
  - Digits
  - Textures

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



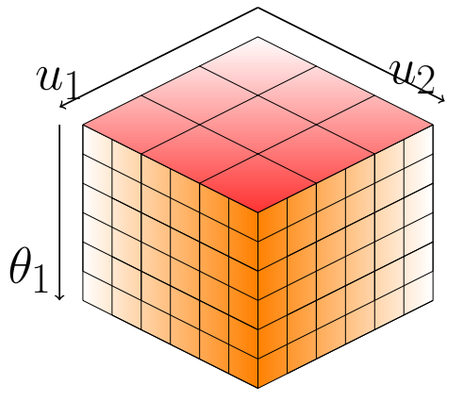



- The design of the scattering transform is guided by the euclidean group
- To which extent can we compete with other architectures on more complex problems (e.g. variabilities are more complex)?

# Separable Roto-translation Scattering

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

$$x(u_1, u_2) \xrightarrow{\star \psi_{\theta_1}}$$




$$\star^{u}\psi \xrightarrow{\star \phi_J}$$ Translation Scattering

$$\circledast^{(u,\theta_1)} \left(\psi \otimes \tilde{\psi}\right) \xrightarrow{\star \phi_J}$$ Roto-Translation Scattering

$$\star^{u}\psi \star^{\theta_1}\tilde{\psi} \xrightarrow{\star \phi_J}$$ Separable Roto-Translation Scattering

- Simplification of the Roto-translation scattering
- Discriminates angular variabilities thanks to a wavelet transform along $\theta_1$ (no averaging!)
- We combine it with Gaussians SVM

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

# How much learning is really required?

Performances are given without intensive data augmentation

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

| Dataset | Type | Accuracy |
|---|---|---|
| Caltech101 | Scattering | 80 |
| | Unsupervised | 77 |
| | Supervised | *93* |
| CIFAR100 | Scattering | 57 |
| | Unsupervised | 61 |
| | Supervised | *82* |

Few adaptation to the dataset

How can we explain the gap with supervised?

**CALTECH**

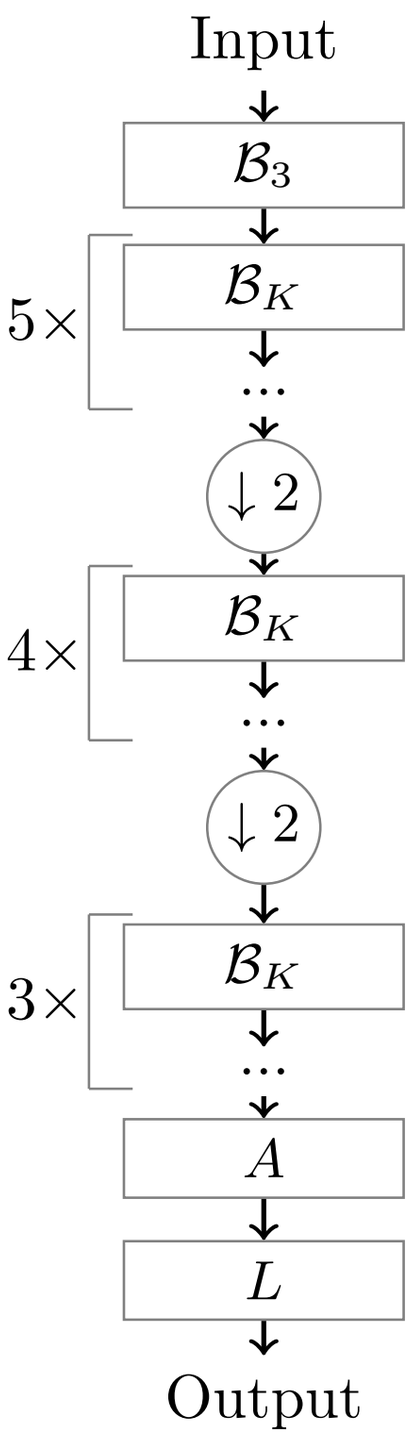
$10^4$ images
101 classes
$256 \times 256$ color images

Scattering is competitive with unsupervised

**CIFAR**

$5.10^4$ images
100 classes
$32 \times 32$ color images

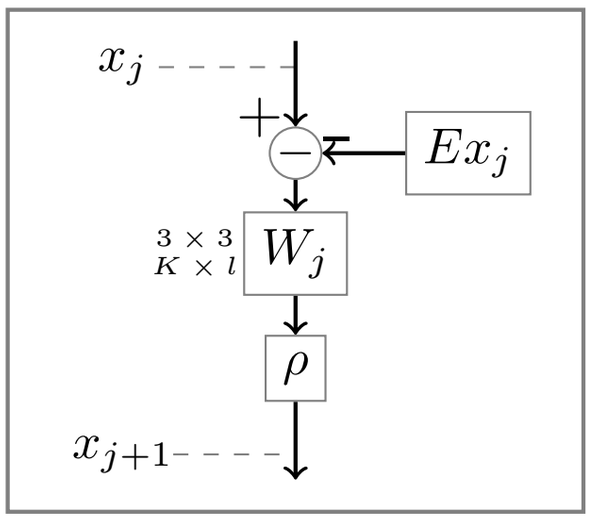
# Studying empirically a CNN




- Which ingredient permits CNNs to outperform the Scattering Transform?

Supervised learning: how? why?

- We introduce a CNN which depends only on its width $K$ and non-linearity $\rho$ in order to study it.
- Good perf and limited engineering: **no max-pooling** or ad-hoc non-linear module, only 13 layers.

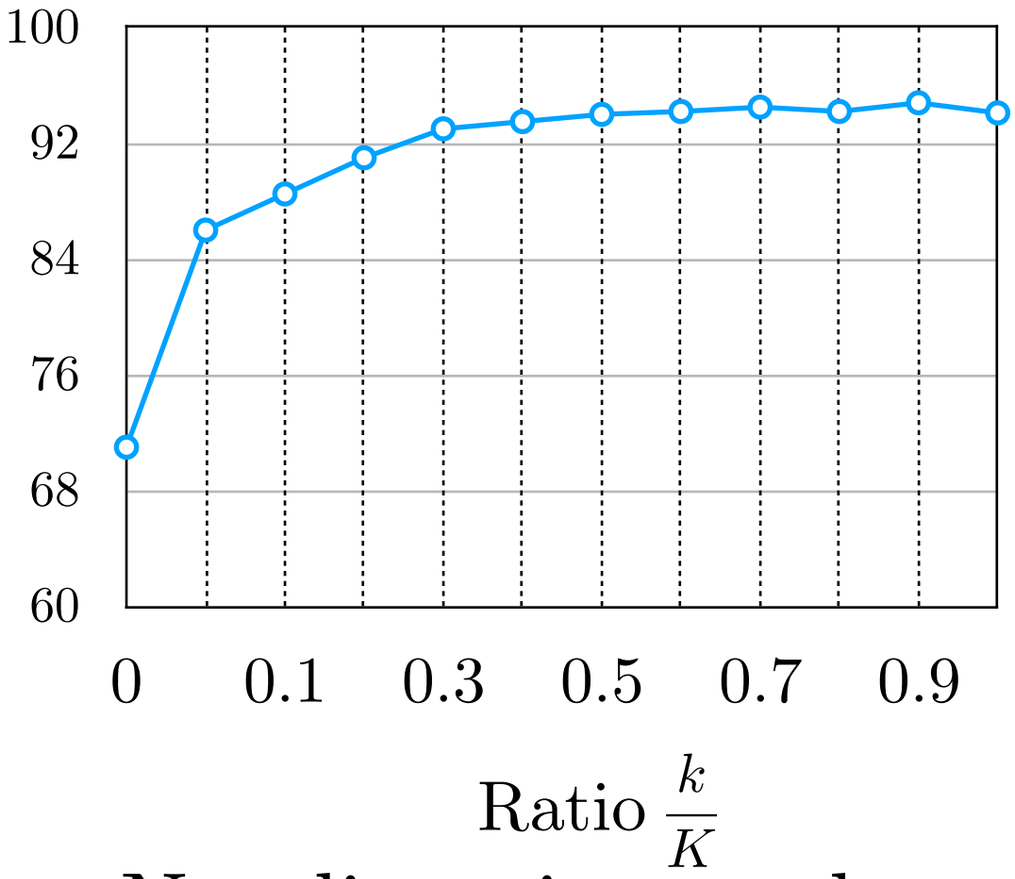



A block $\mathcal{B}_l$

| | Depth | #params | CIFAR10 | CIFAR10 |
|---|---|---|---|---|
| **Ours** | **13** | **28M** | **95.4** | **79.6** |
| SGDR | 28 | 150M | 96.2 | 82.3 |
| WResNet | 28 | 37M | 95.8 | 80.0 |
| All-CNN | 9 | 1.3M | 92.8 | 66.6 |

# Example of sandbox application: on the non-linearity in CNNs

- "More non-linear is better."



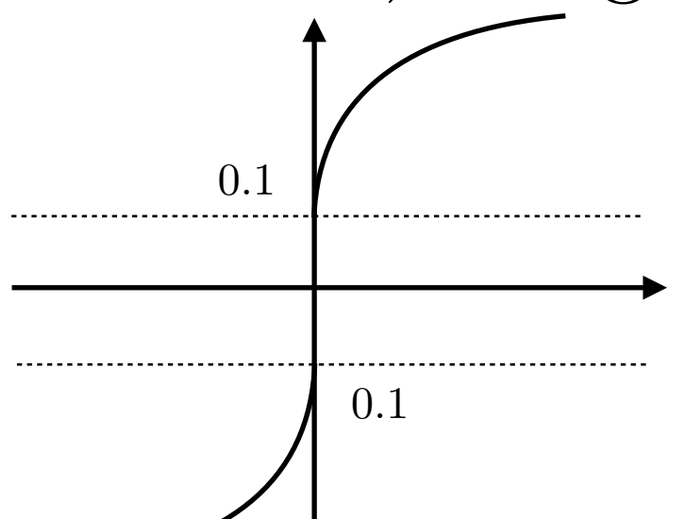

ReLU on a fraction $\frac{k}{K}$ of the coefficients a layer $x$:

$$\mathrm{ReLU}_k^K(x)(.,l) \triangleq \begin{cases} \mathrm{ReLU}(x(.,l)), & \text{if } l \leq k \\ x(.,l), & \text{otherwise} \end{cases}$$

Tradional pointwise non-linearity can be weakened

- Non-linearity needs to contract, being continuous or to remove the phase?

$$\rho(x) = \mathrm{sign}(x)(\sqrt{|x|} + 0.1)$$

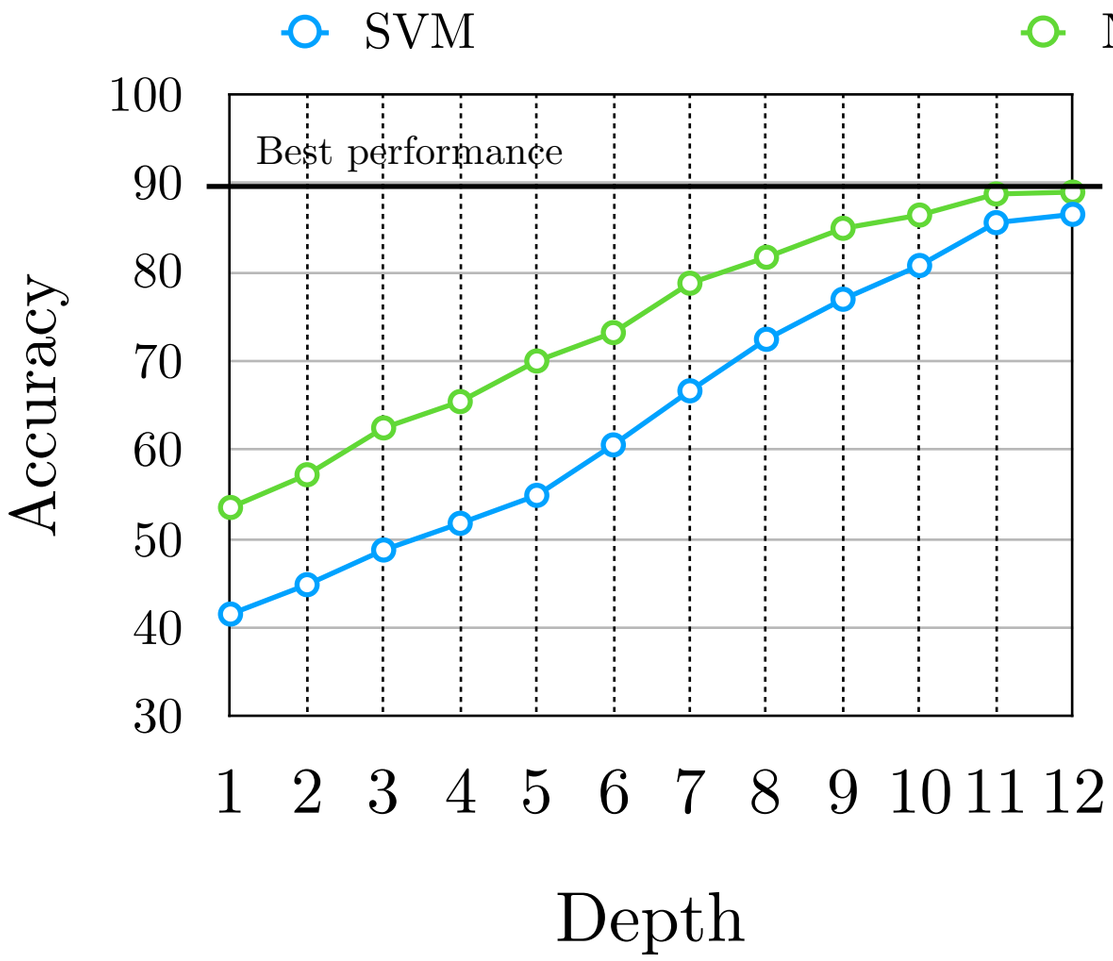



good perf (89% acc.) on CIFAR10

# Progressive properties

- We aim to show the cascade permits a progressive contraction & separation, w.r.t. the depth:


In the following, representations are spatially averaged.



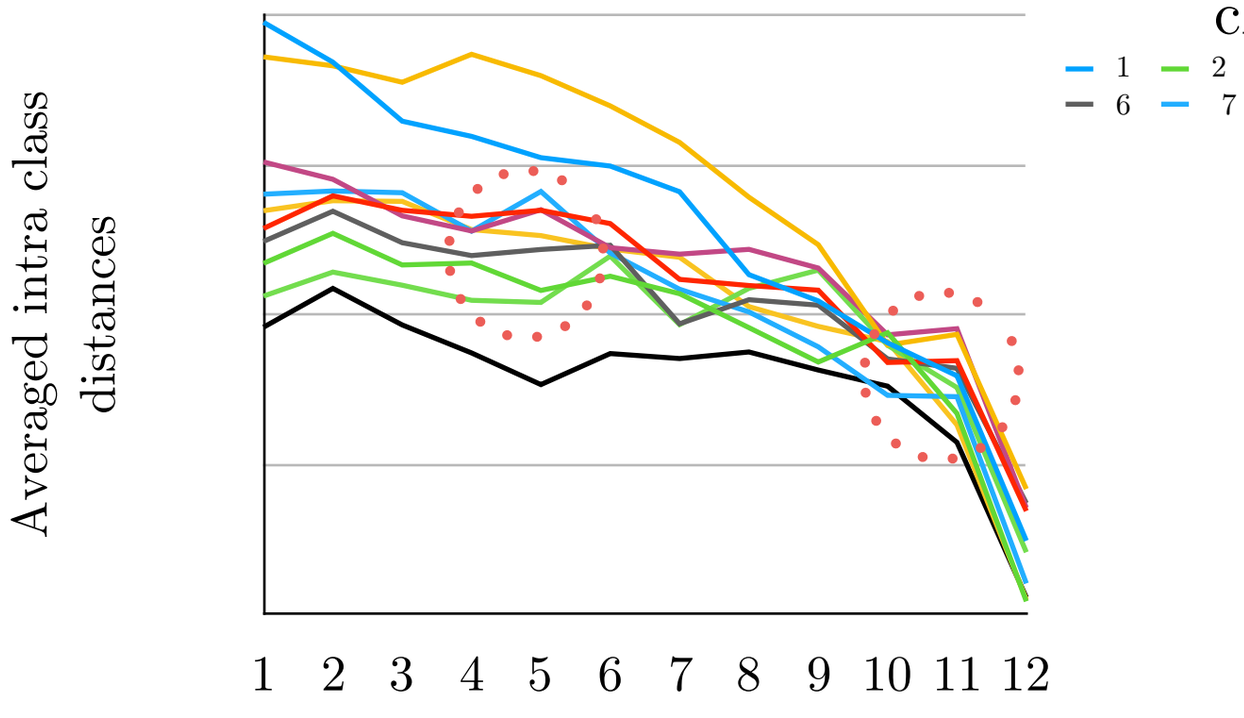




Localised classifiers

- How can we explain it?

# Understanding the progressive improvement

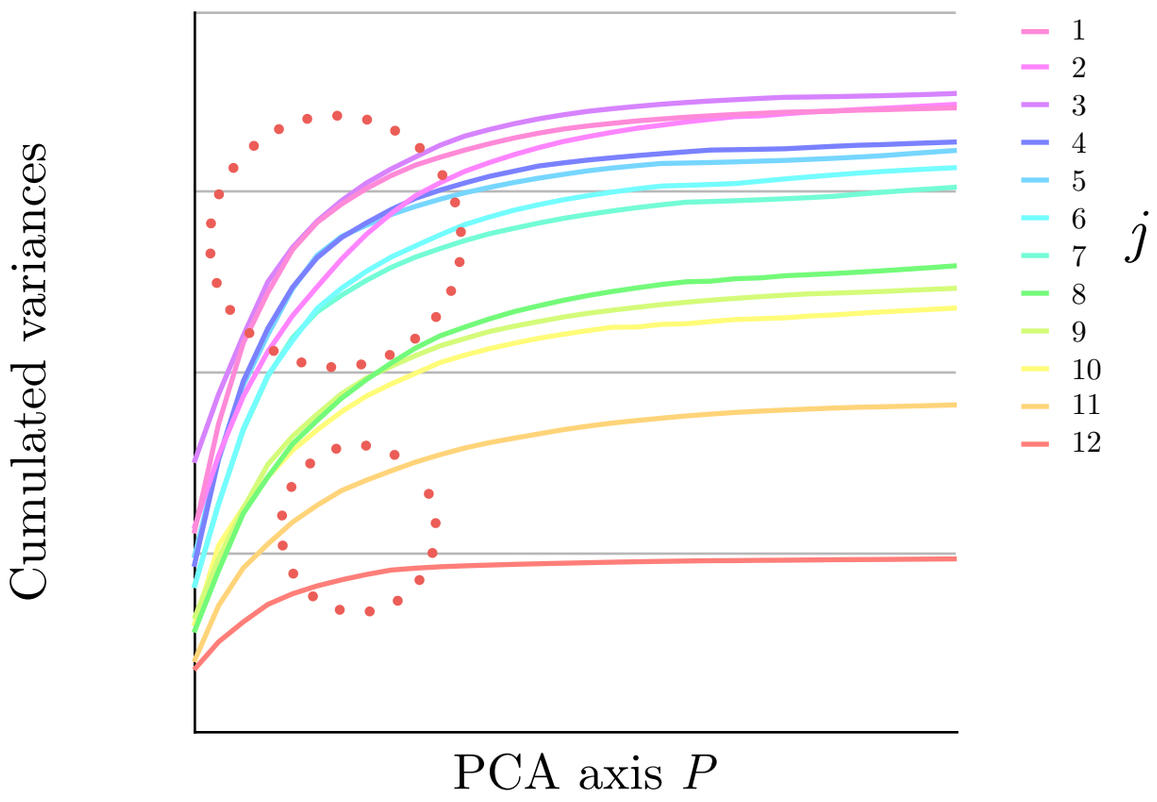



$$\frac{1}{5000^2} \sum_{y(x_j)=y(\tilde{x}_j)=c} \|x_j - \tilde{x}_j\|$$

Intra class distance

**No clear behavior:**

**Refining those measures?**



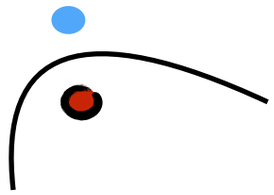

$$\Sigma_P^j = \sum_{p \leq P} \sigma_p^j$$

Cumulative eigenvalues $\sigma_p^j$ for the representation at depth $j$

# Local Support Vectors: exploring regularity

- Estimating the intrinsic dimension of the classification boundary is hard *(curse of dimensionality!)*
- We introduce Local Support Vectors(LSV):

$$\Gamma_j^1 = \{x_j | y(x_j^{(1)}) \neq y(x_j)\}$$



$x_j^{(l)}$ : $l$-th nearest neighbour of feature at depth $j$

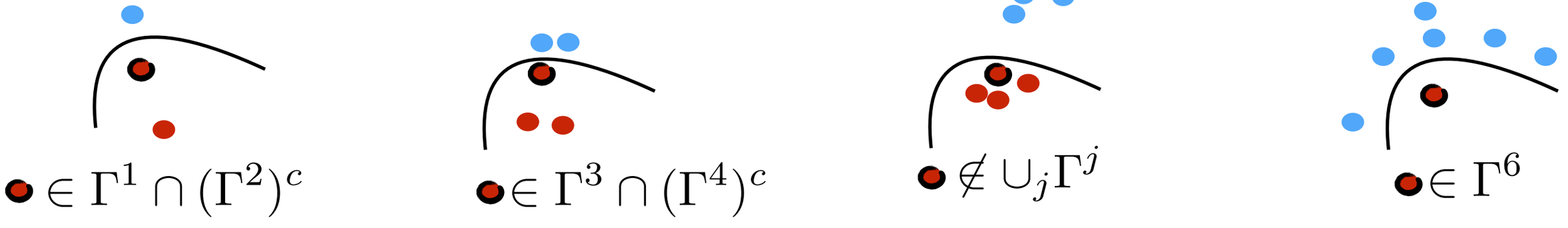
$y(x_j)$ : class of $x$

- We extend this definition recursively to define $k$-LSV:

$$\Gamma_j^{k+1} = \{x_j \in \Gamma_j^k | \text{card}\{y(x_j) \neq y(x_j^{(l)}), l \leq k+1\} > \frac{k+1}{2}\}$$

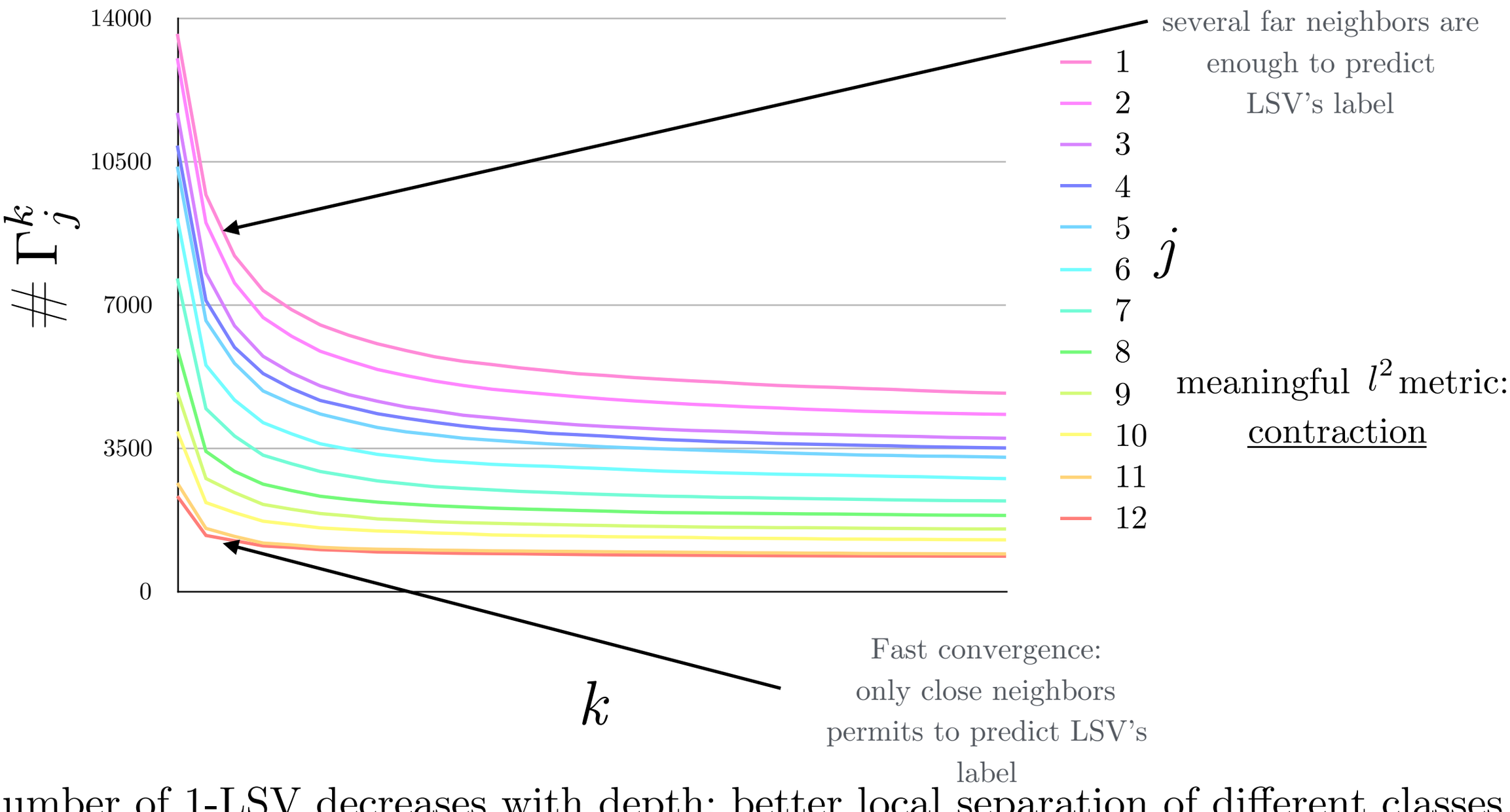
Features that are not classified by any $l$- nearest, $l<k$ ,neighbour at depth $j$

- Regularity measure at depth $j$: $k$-LSV approximatively indicates how local is the euclidean metric:



# Progressive contraction-separation measure

- We compute the # of $k$-LSV at each depth $j$ on CIFAR10:




- The number of 1-LSV decreases with depth: better local separation of different classes

**How does the separation/contraction happen?**

# Scattering + CNN

| | CNN | Scattering |
|---|---|---|
| Good perf | Yes | No |
| Interpretable | No | Yes |

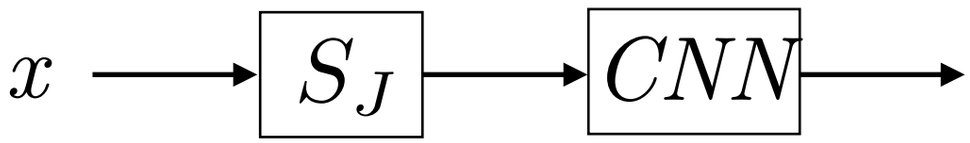
- Can we **combine** the best of both world to understand better CNNs?

- We demonstrate **interpretability** and **no loss** in informative variabilities for training a CNN

- Remark: Scattering is stable, CNN is unstable, thus Scattering+CNN has no reason to be more stable

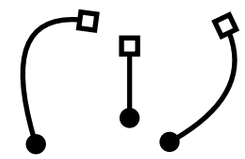
# An ideal input for a modern CNN

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




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

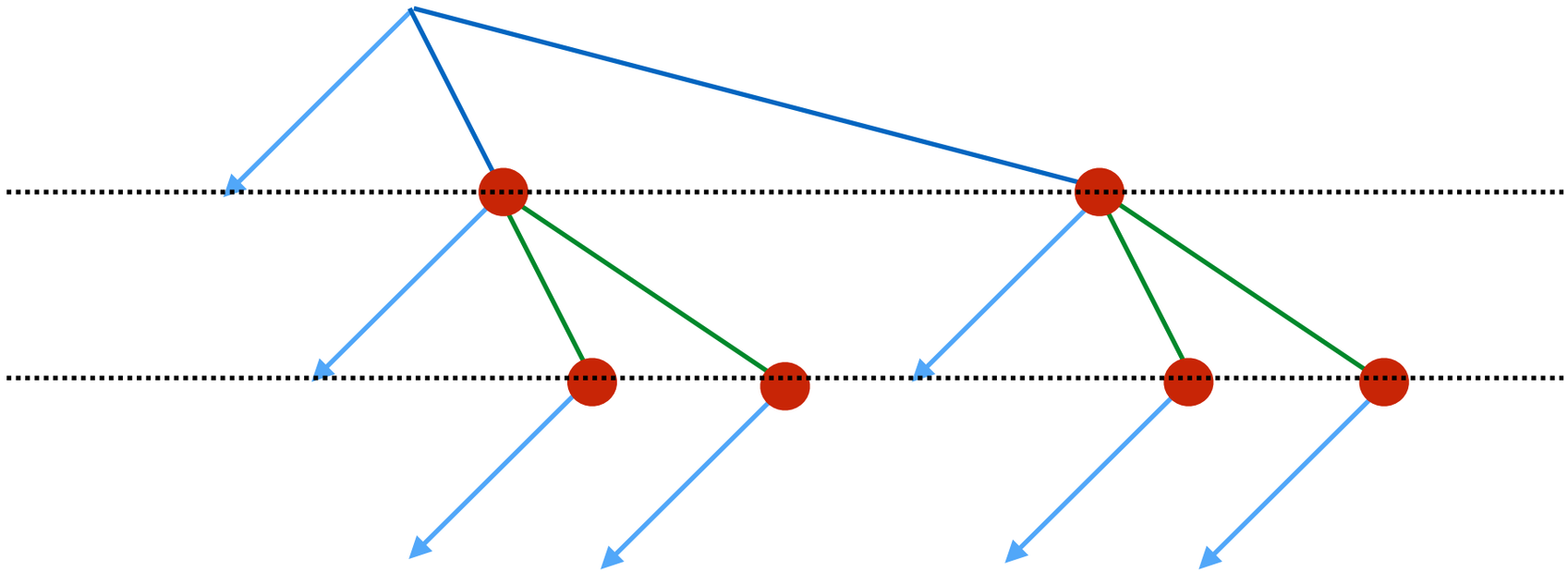
- For $\lambda, u$, $S_J x(u, \lambda)$ is covariant with $SO_2(\mathbb{R})$:

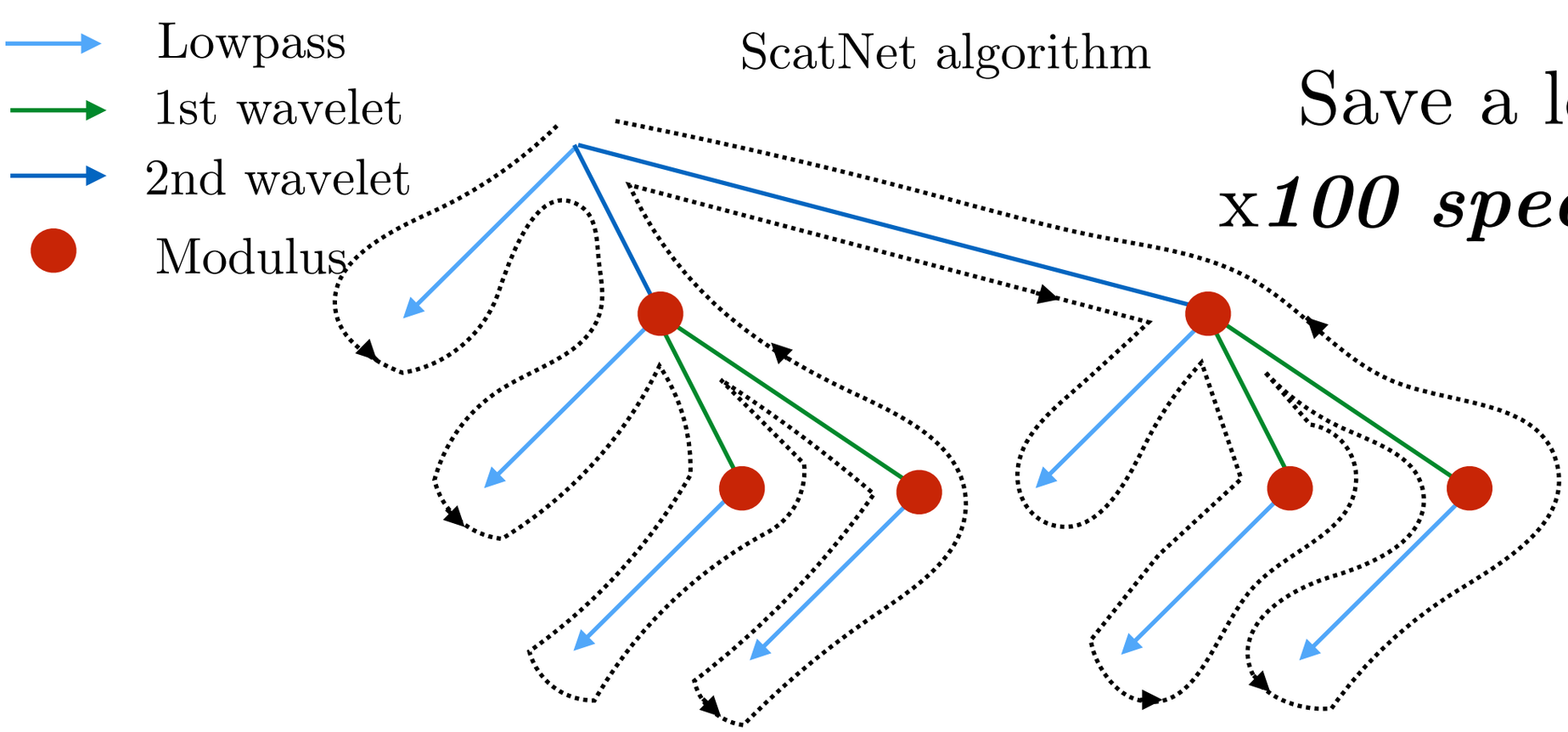
$$\text{if } \forall u \forall g \in SO_2(\mathbb{R}), g.x(u) \triangleq x(g^{-1}u) \text{ then,}$$

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

# Scaling scattering on GPUs

Ref.: Thesis, EO

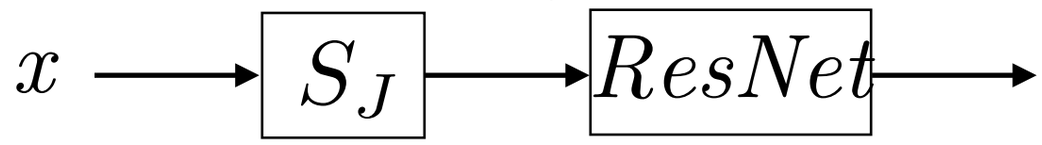





Save a lot of memory!

***x100 speed-up on GPU***

# Imagenet/CIFAR




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

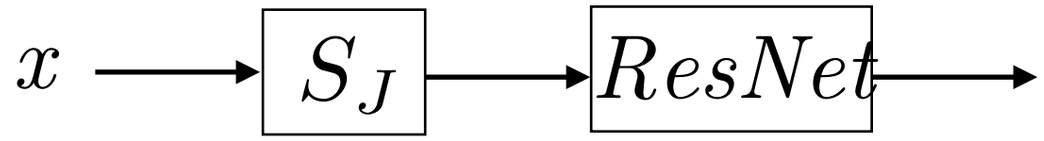
- State-of-the-art result on Imagenet2012:

| | 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

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

# Benchmarking Small data




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!!)

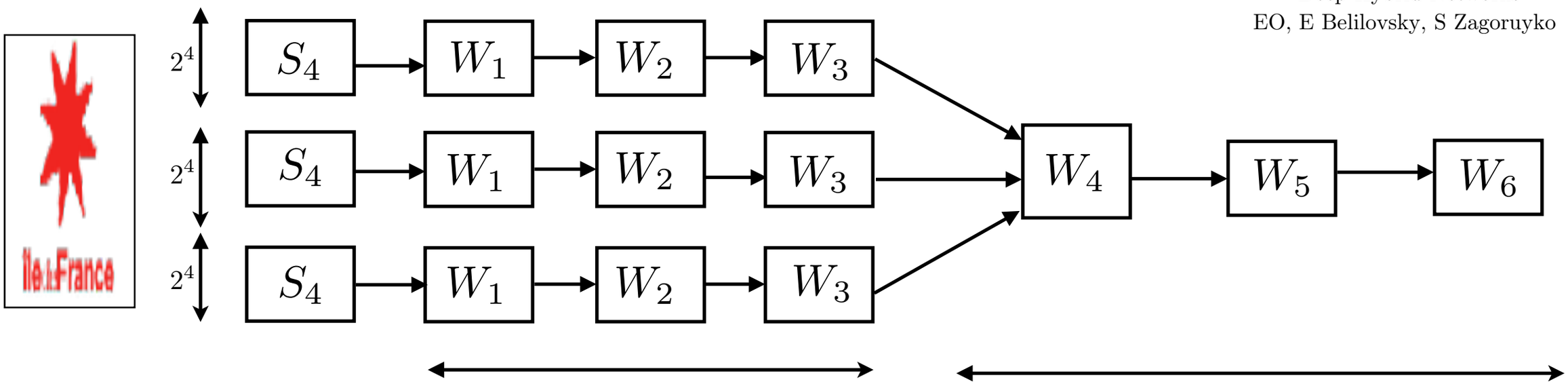
| | Acc. |
|---|---|
| Scat+ResNet | **76** |
| Supervised | 70 |
| Unsupervised | 76 |

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 |

# Shared Local Encoder

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




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

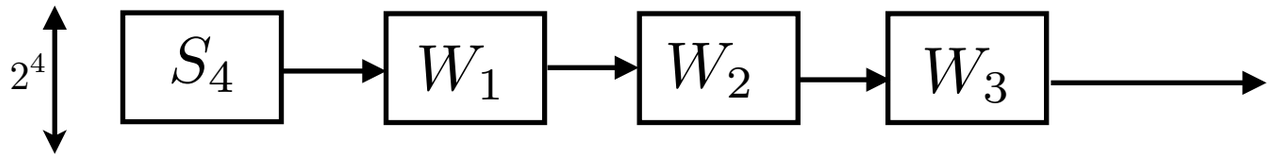
Is it really a classifier?

- 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:

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




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

Ref.: Visualizing and Understanding Convolutional Networks, M Zeiler, R Fergus

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

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

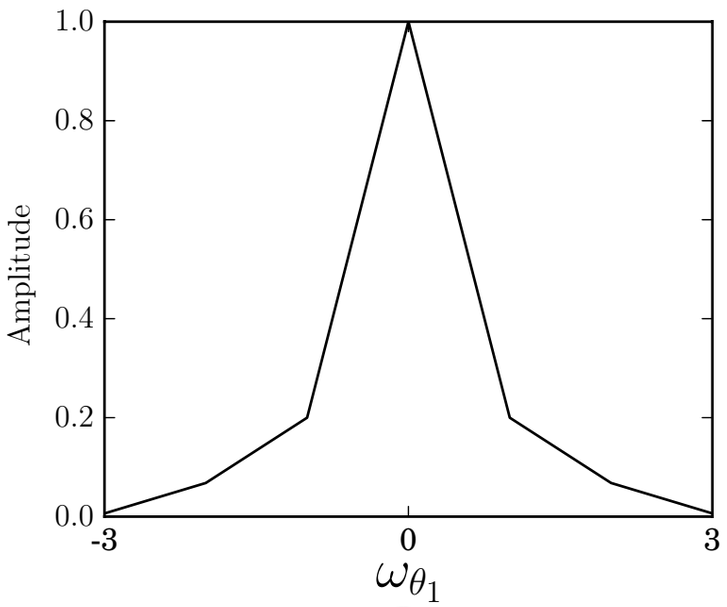
**What is the nature of the recombination?**

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

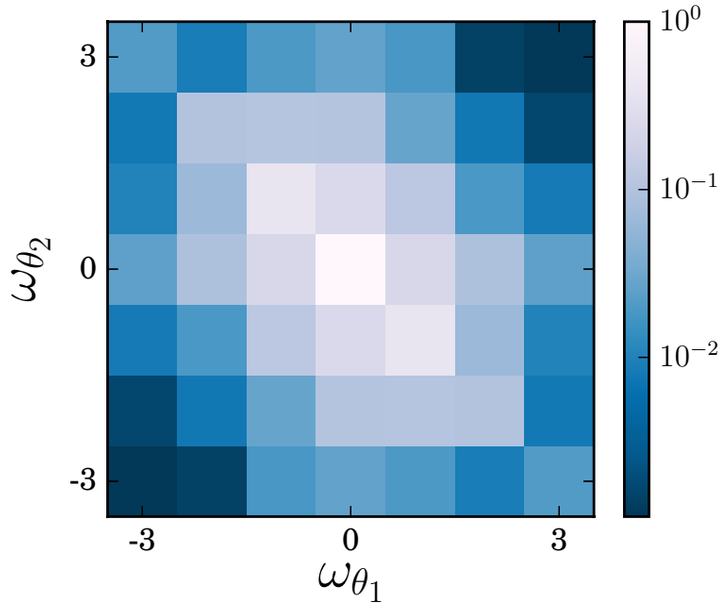
Fourier along $(\theta_1, \theta_2)$: $\hat{w}_{(j_1,j_2,\omega_{\theta_1},\omega_{\theta_2}),k} = \mathcal{F}^{(\theta_1,\theta_2)}(w_{(j_1,j_2,.,.),k})(\omega_{\theta_1},\omega_{\theta_2})$

# Analysis of $w$

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




$\Omega_1$

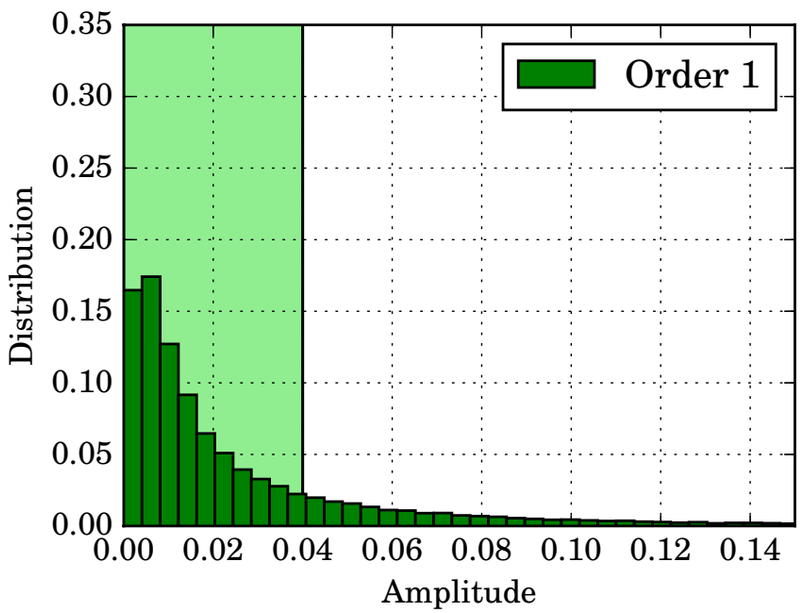


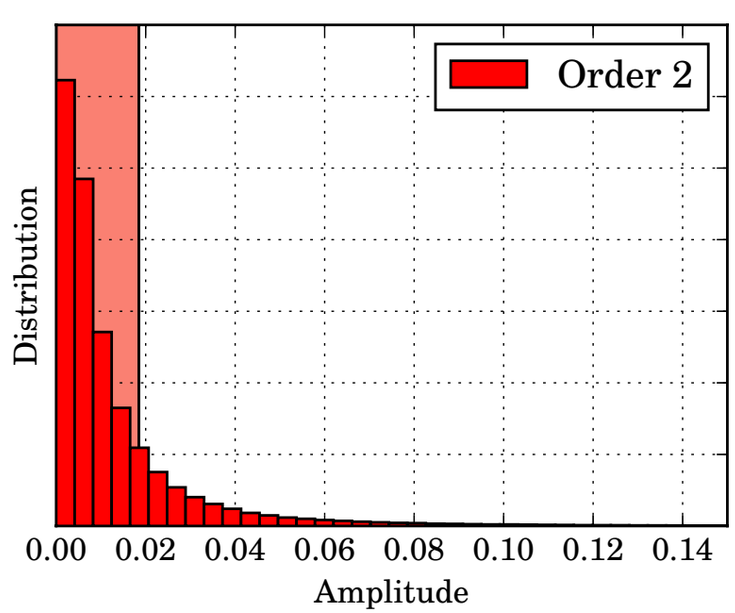

$\Omega_2$

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

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

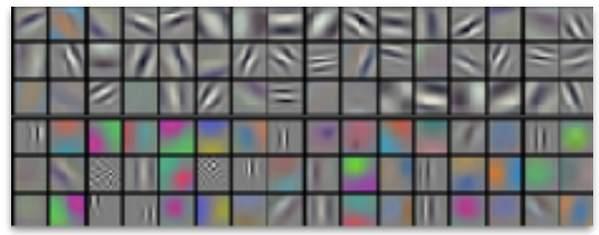
- Invariance to rotation is explicitly learned.




method: similar to AlexNet

first layer analysis



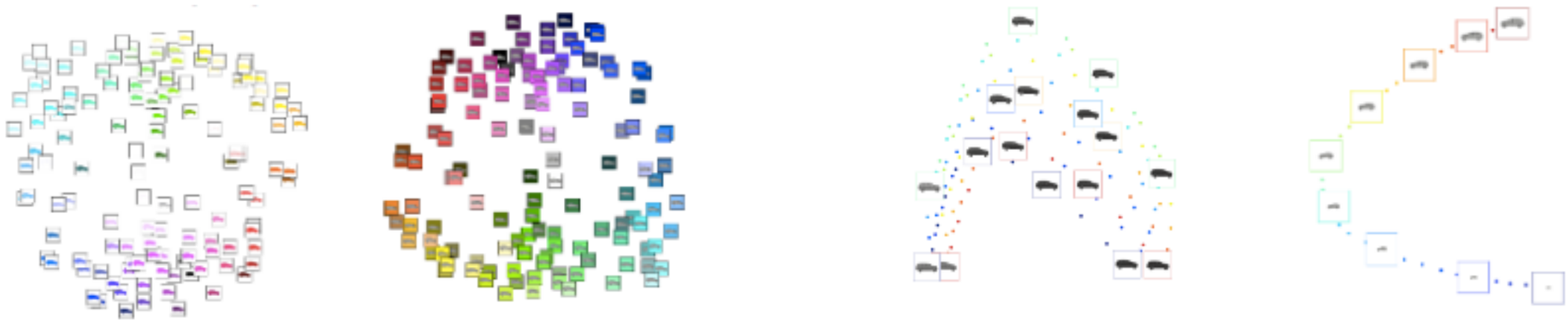
Fourier basis
sparsifies the operator!

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

Can we find more complex invariance than rotation?

# Identifying the variabilities?

- CNNs exhibit some covariance w.r.t. variabilities:



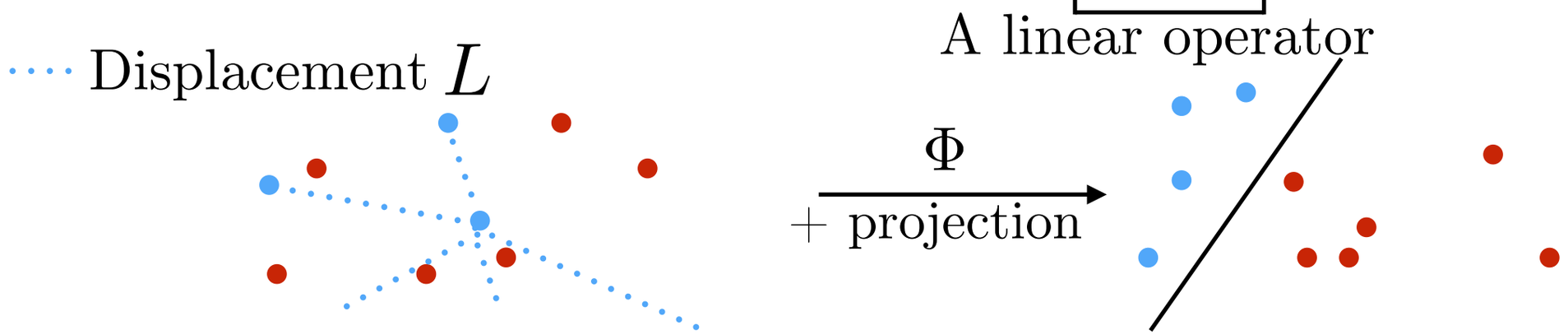
(c) Object color (d) Background color (a) Lighting (b) Scale

Ref.: Understanding deep features with computer-generated imagery, M Aubry, B Russel

- If we understood how this is done, we could use this to engineer CNN with nice properties:

$$\sup_L \frac{\|\Phi L x - \Phi x\|}{\|Lx - x\|} < \infty \Rightarrow \exists \text{ "weak" } \partial_x \Phi$$

$$\Rightarrow \Phi L x \approx \Phi x + \underbrace{\partial_x \Phi L}_{\text{A linear operator}} + o(\|L\|)$$



# Introducing structures to understand better

- SLE learned invariance: invariants to group are explicitly built via convolutions.

- Are all the non-informative variabilities linked to group actions?

- Can we structure the channel axis as we did?

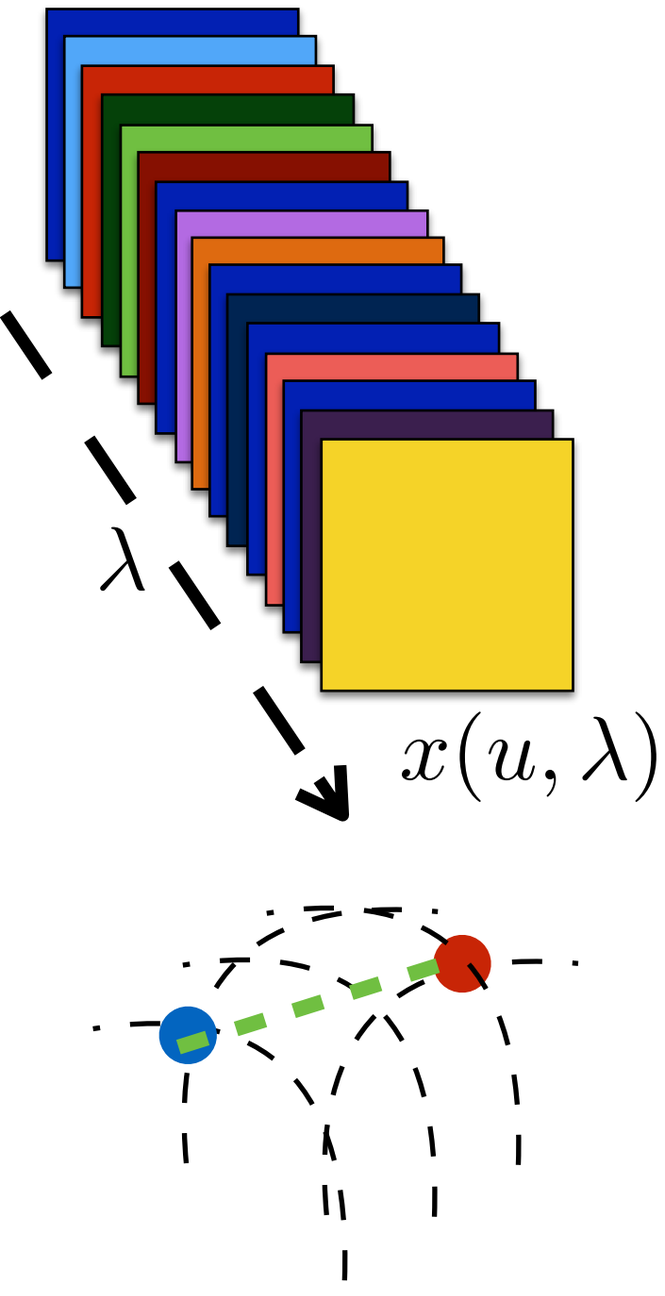
- We will apply $n$-d convolutions, with $n>3$

$$x \star k(u_1, ..., u_n) = \sum_{v_1, ..., v_n} x(u_1 - v_1, ..., u_n - v_n) k(v_1, ..., v_n)$$

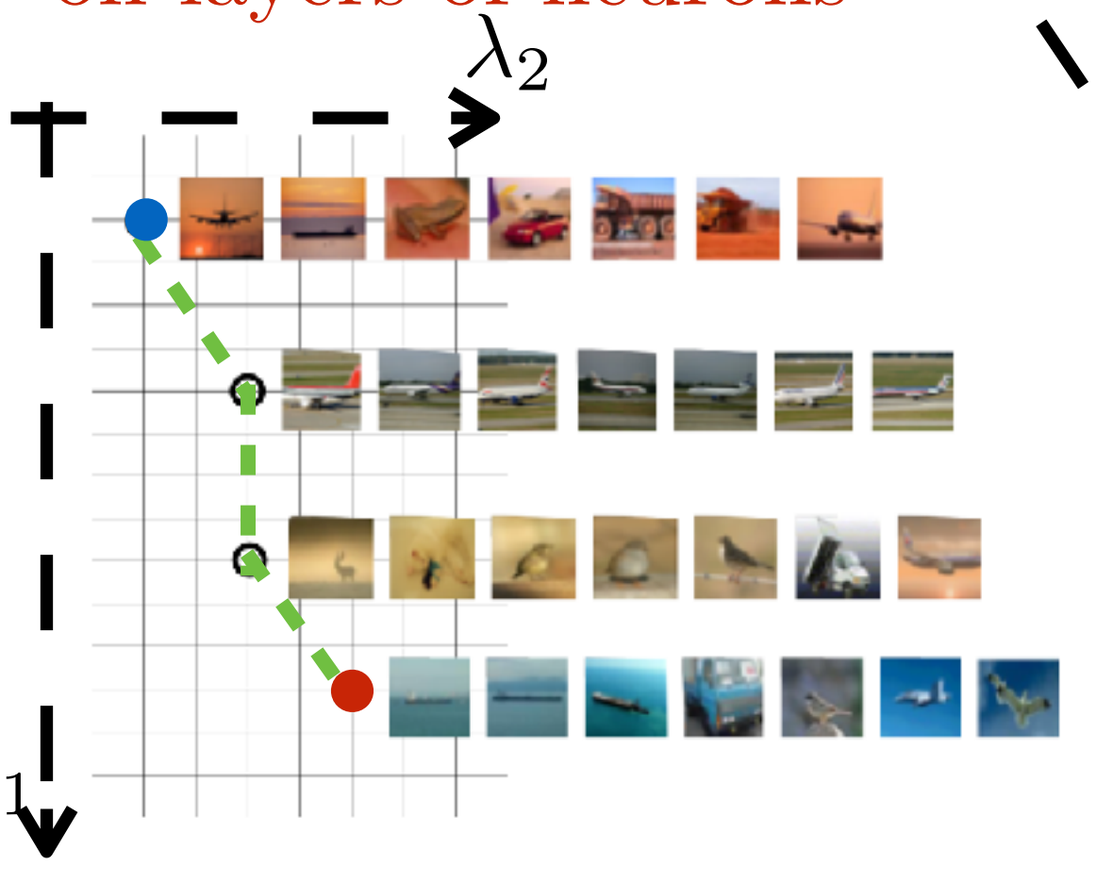
Beyond the order of scattering convolution (i.e. 3)

# Objective: Flattening the variability

Standard CNNs layers

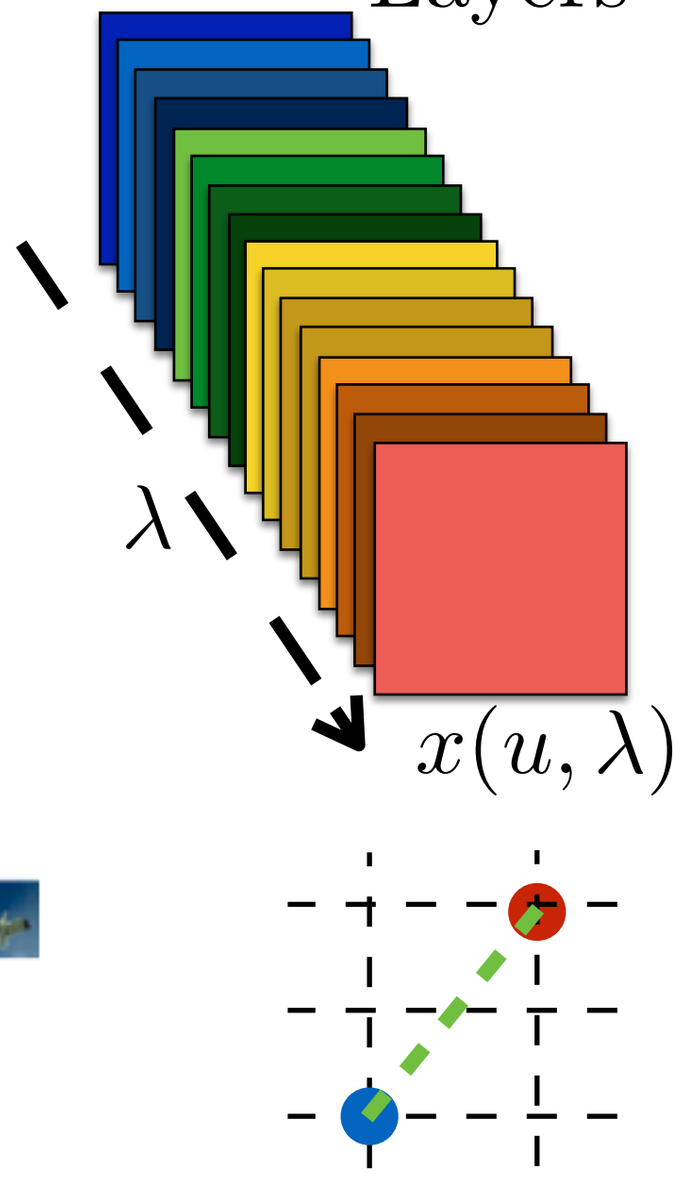



Defining an order
on layers of neurons




$\lambda = (\lambda_1, \lambda_2)$

Organised
Layers



# 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 and CNNs such that:

$$G_0 \subset G_1 \subset ... \subset G_J \subset G$$
$$\forall g_j \in G_j, \phi_j(g_j.x) = \phi_j(x) \text{ where } x_j = \phi_j(x)$$

- Examples are given by the euclidean group:

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

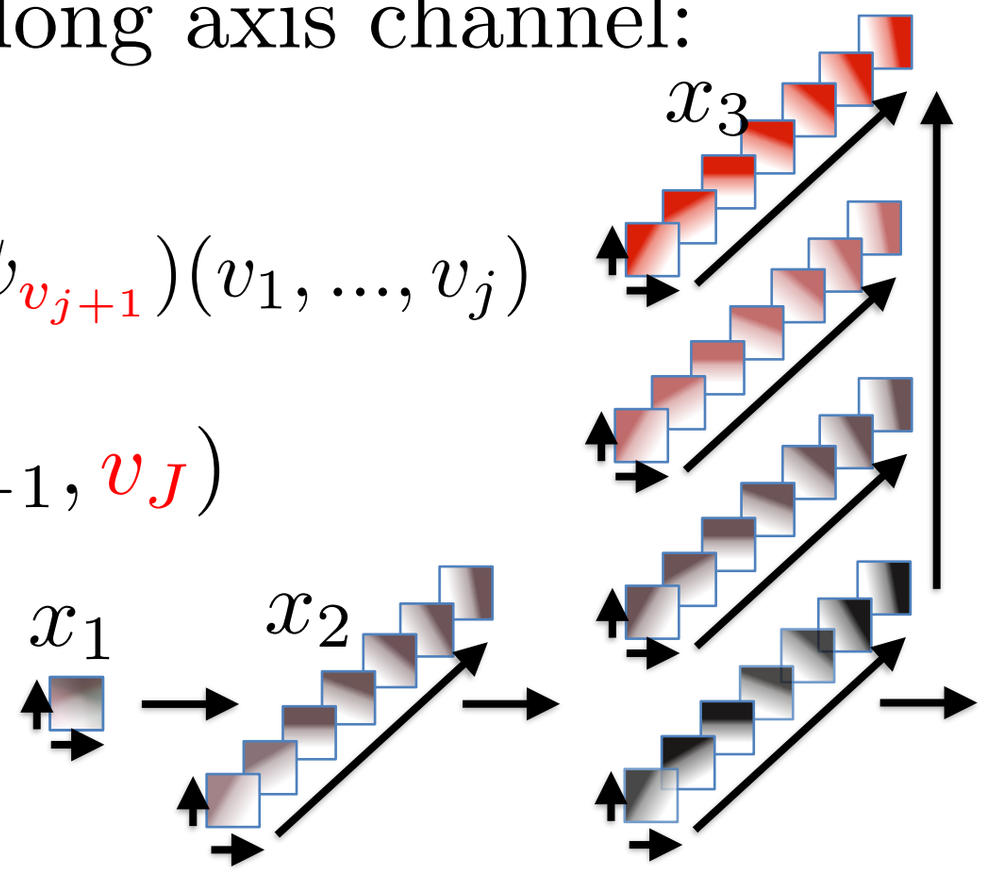
# Multiscale Hiearchical CNN

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

- CNN that is convolutional along axis channel:

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

$$x_J(v_J) = \sum_{v_1,...,v_{J-1}} x_{J-1}(v_1, ..., v_{J-1}, v_J)$$



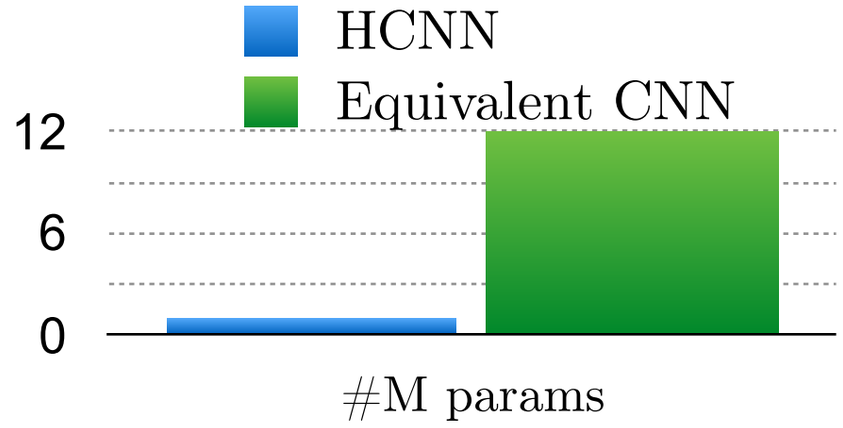

- For $x_j$, we refer to the variable $v_j$ as an attribute that discriminates previously obtained layer.

Understanding deep convolutional networks
S Mallat

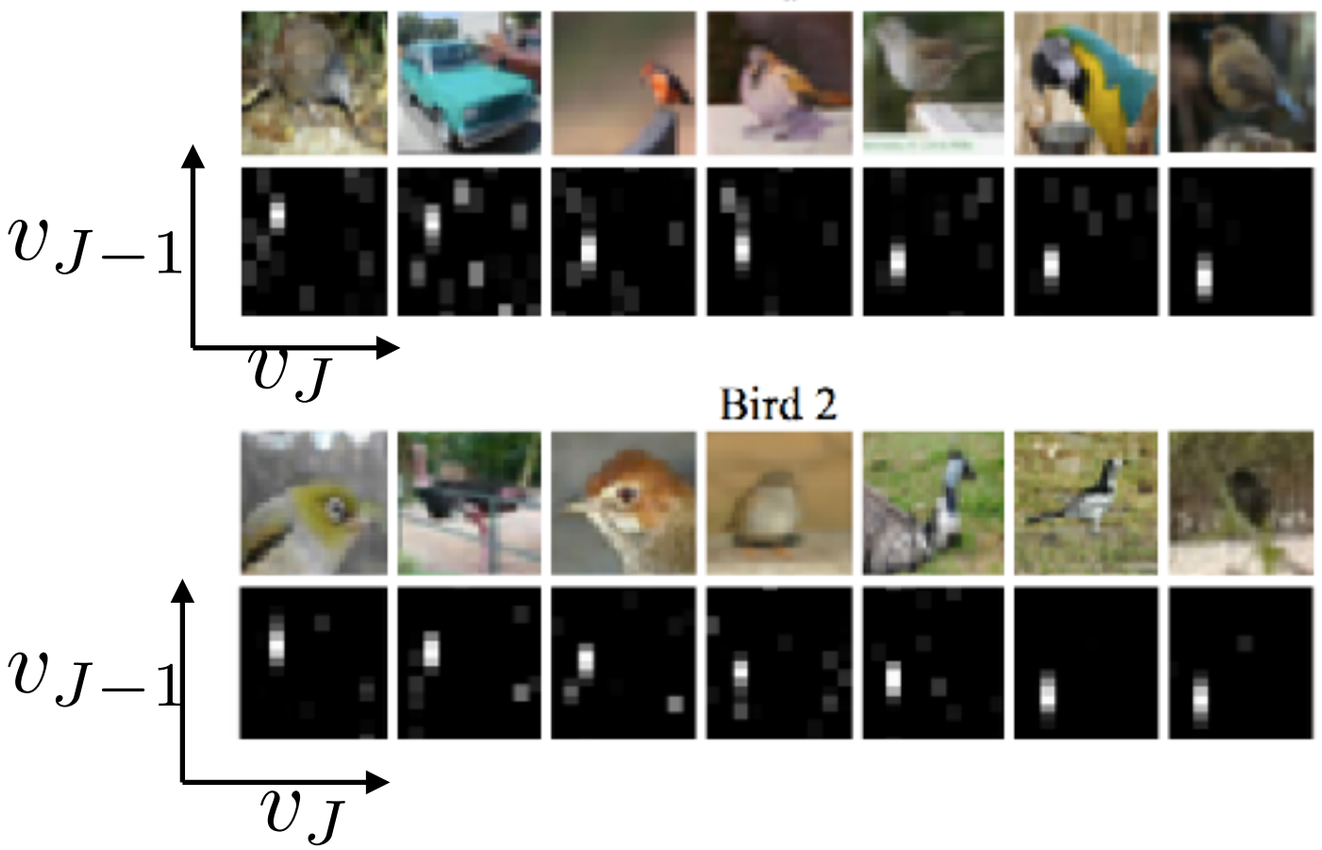
- Representation is finally averaged: invariant along translations by $v$.

# 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

# Conclusion

- The problem was to introduce and analyze structures in CNNs.

- We demonstrate:

  State-of-the-art performance without learning first layer weights

  CNN progressively contract

  Scattering + CNN is a robust baseline

  Implement a new class of CNN

Thank you!