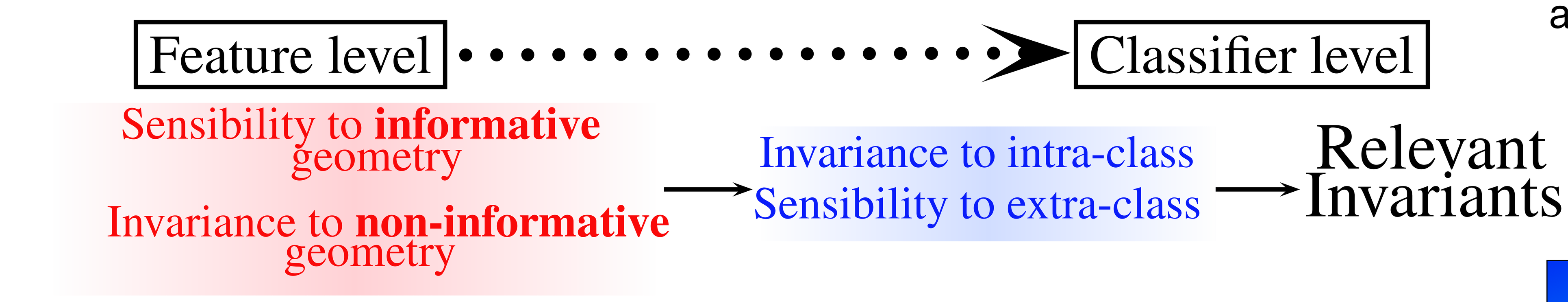
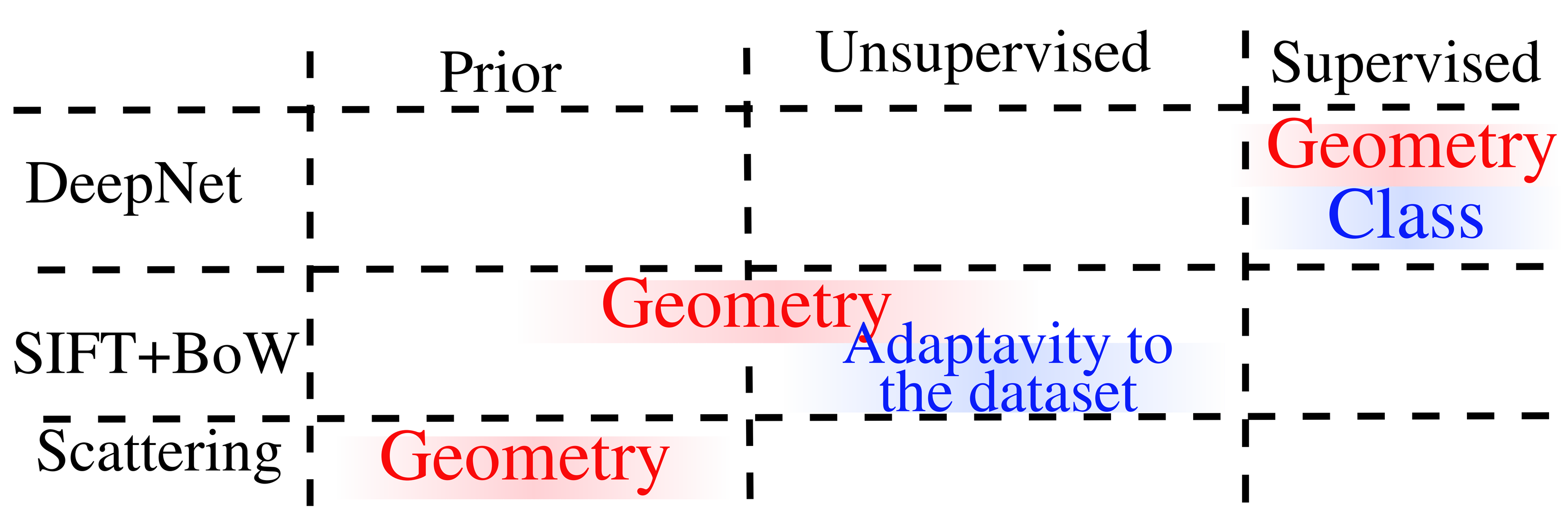
Deep Roto-Translation Scattering for Object Classification

Edouard Oyallon and Stéphane Mallat
DATA, Département Informatique, Ecole Normale Supérieure

DATA

Geometry vs Unsupervised



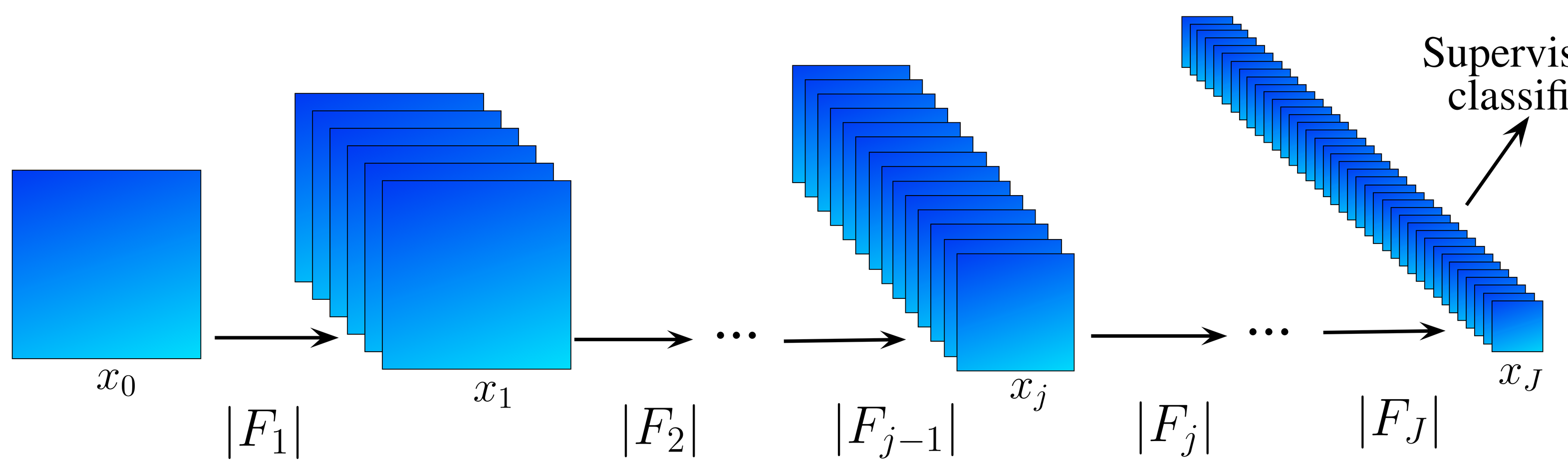


- We build a deep **super-SIFT** generic and geometric descriptor **without learning** competitive with unsupervisedly learned representations.

Deep Scattering representation

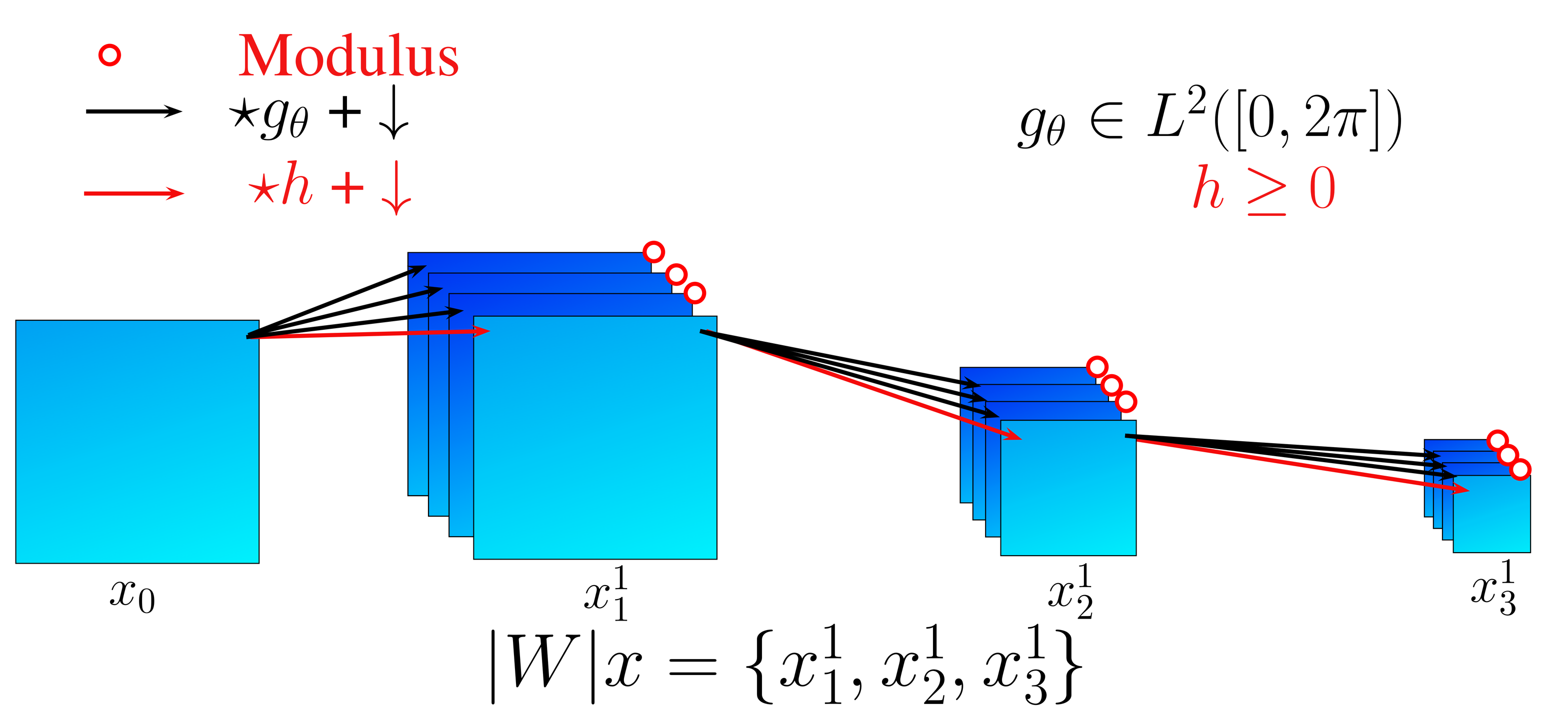
- DeepNet as a deep cascade of filterbanks and nonlinearities:

$$x_{j+1} = |F_{j+1}|x_j$$



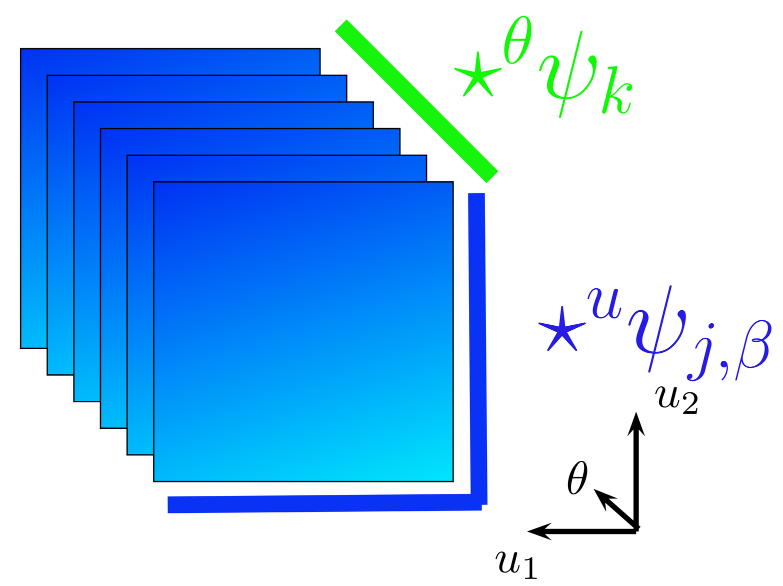
- Modulus of the wavelet transform as a deep cascade of filters and downsampling, followed by a complex modulus:

$$x^1_{j+1}(u,\theta) = |x^1_j(.,0) \star g_\theta|(2u) \qquad x^1_{j+1}(u,0) = (x^1_j(.,0) \star h)(2u)$$



- Then we build a 3D separable wavelet transform along space and angles for a path $q = (\beta, j_1, k, \theta)$:

$$x^2_j(u,q) = |x^1_{j_1} \star^u \psi_{j,\beta} \star^\theta \psi_k|(u,\theta)$$



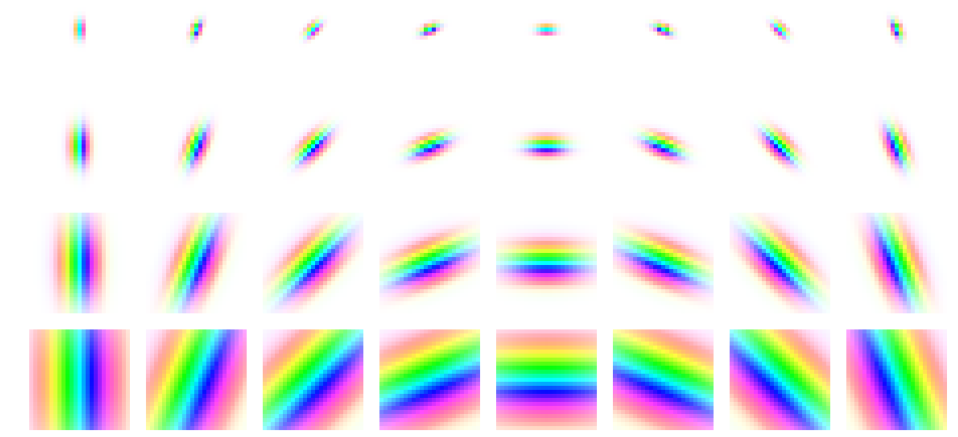
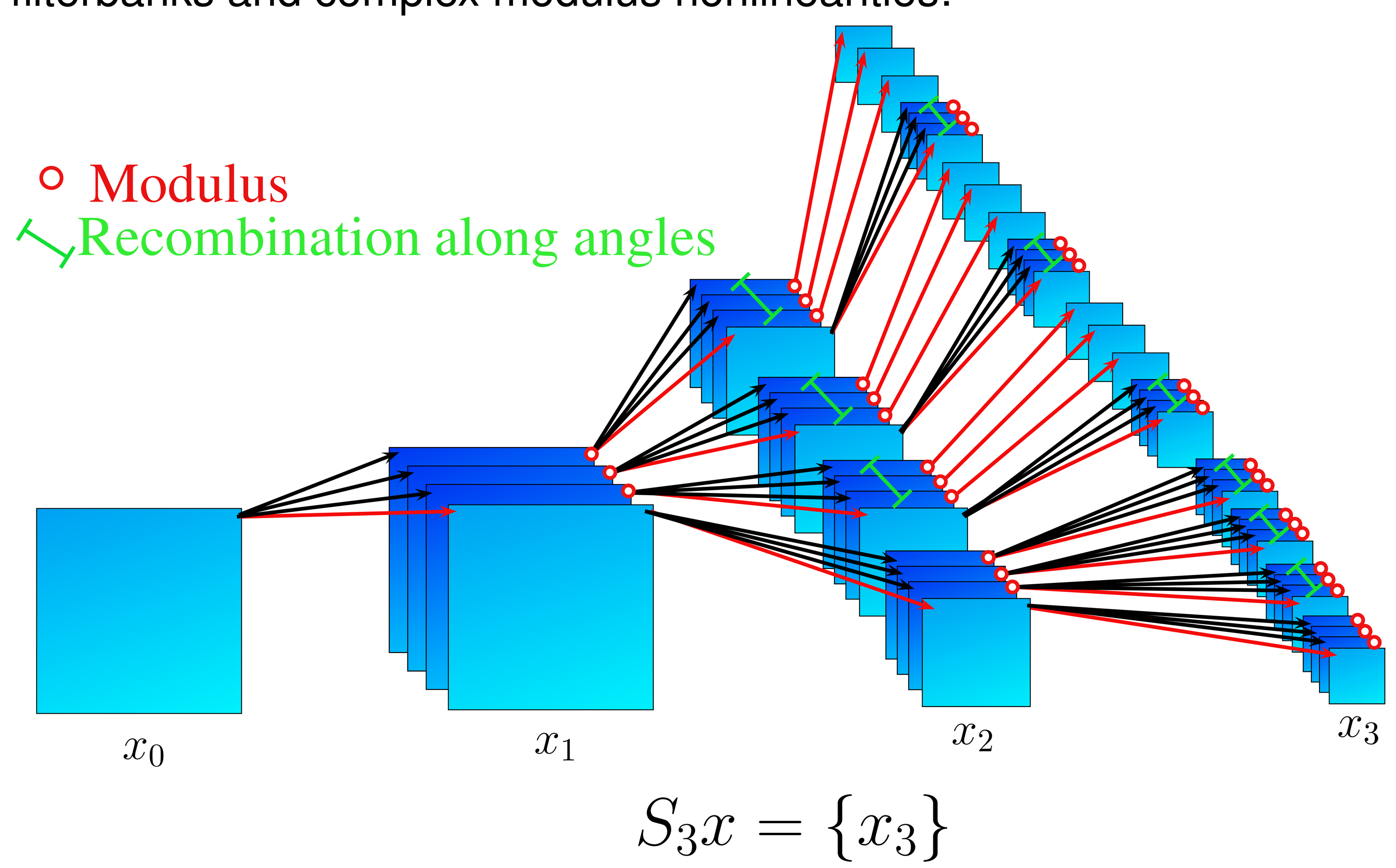


Figure: Morlet wavelets rotated and scaled

- This separable convolution recombines angles, **linearizing the deformations due to rotation**.
- All the coefficients are finally averaged to achieve spatial invariance:

$$S_J x = \{x \star \phi_J, x^1_j \star \phi_J, x^2_j \star \phi_J\}_{j \leq J}$$

- Thus, the **Roto-Translation Scattering Transform** is a deep cascade of filterbanks and complex modulus nonlinearities:



Classification pipeline

x — S_J — $S_J x$ — log — L — Standardization — Gaussian SVM

- S_J is computed on every channels YUV of an image.
- L is a projection supervisedly learned via a forward selection algorithm: **Orthogonal Least Square** (OLS).
- The log linearizes multiplicative luminance variations.
- Standardization: normalization of the mean and variance.
- Coefficients feed a Gaussian kernel SVM with unit variance.

Numerical results

- Image inputs:

Figure: Caltech: 256×256 color images, 30 samples for training

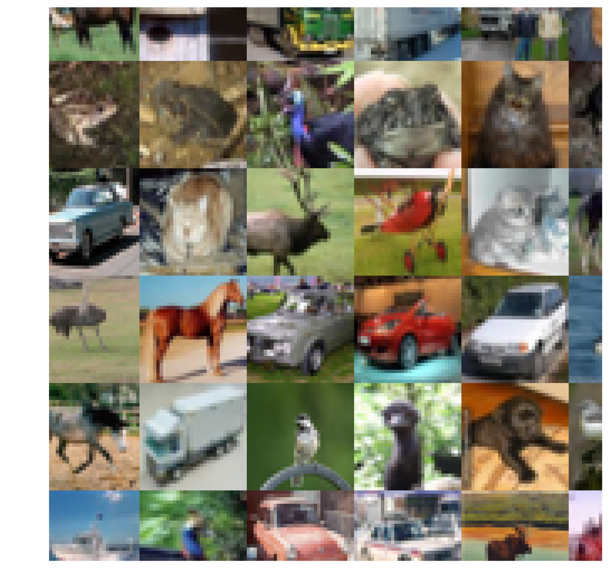


Figure: CIFAR: 32×32 color images, 500/5000 samples for training

- Accuracies on Caltech101/256, CIFAR10/100:

Method	Type	Acc.
ScatNet	Prior	79.9
M-HMP	Unsupervised	82.5
CNN	Supervised	91.4

Table: Results for Caltech101, 101 classes, 1.10^4 samples.

Method	Type	Acc.
ScatNet	Prior	43.6
M-HMP	Unsupervised	50.7
CNN	Supervised	70.6

Table: Results for Caltech256, 256 classes, 3.10^4 samples.

Method	Type	Acc.
ScatNet	Prior	82.3
RFL	Unsupervised	83.1
CNN	Supervised	91.8

Table: Results for CIFAR-10, 10 classes, 6.10^4 samples.

Method	Type	Acc.
ScatNet	Prior	56.8
NOMP	Unsupervised	60.8
CNN	Supervised	65.4

Table: Results for CIFAR-100, 100 classes, 6.10^4 samples.

Method	Caltech-101	CIFAR-10
T first order+SVM	59.8	72.6
T+SVM	70.0	80.3
T+OLS+SVM	75.4	81.6
TR+SVM	74.5	81.5
TR+OLS+SVM	79.9	82.3

Table: "T" and "TR" stands respectively for translation and roto-translation scattering.

Conclusion

- Generic and competitive representation with **few parameters**.
- More supervision could help to improve numerical results: adding more supervision at the top of the network... **input of a DeepNet**?
- ...or unsupervised layers? **Fisher Vectors**? Transform on the affine group?

Contacts:

- Website of the software **ScatNetLight**: **https://github.com/edouardoyallon/ScatNetLight/releases/**
- Website of the team: **http://www.di.ens.fr/data/**
- Edouard Oyallon **edouard.oyallon@ens.fr**