Organizing Deep Networks

advisor: Stéphane Mallat
following the works of Laurent Sifre, Joan Bruna, ...

collaborators: Eugene Belilovsky, Sergey Zagoruyko, Bogdan Cirstea, Jörn Jacobsen, ...
Classification of signals

- Let $n > 0$, $(X, Y) \in \mathbb{R}^n \times \mathcal{Y}$ random variables

- **Problem**: Estimate $\hat{y}$ such that $\hat{y} = \arg \inf_{\tilde{y}} \mathbb{E}(|\tilde{y}(X) - Y|)$

- We are given a training set $(x_i, y_i) \in \mathbb{R}^n \times \mathcal{Y}$ to build $\hat{y}$

- Say one can write $\hat{y} = \text{Classifier}(\Phi x)$, Classifier being built with $(\Phi x_i, y_i)$

- 3 ways to build $\Phi$:
  - Supervised $(x_i, y_i)_i$
  - Unsupervised $(x_i)_i$
  - Predefined Geometric priors

\[ \mathcal{Y} = \{ , \} \]
\[ n = 2 \]

Classifier $w$
High Dimensional classification

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

Estimation problem

Training set to predict labels

"Rhino"

Not a "rhino"

"Rhinos"
High-dimensional variabilities

- **Claim:** In $\mathbb{R}^n$, $n \gg 1$, the variance is huge.
  
  Ex.:
  
  \[ X \sim \mathcal{N}(0, I_n) \text{ then } \exists C > 0, \forall n, \mathbb{P}(\|X\| \geq t)) \leq 2e^{-\frac{t^2}{Cn}} \]
  \[ \mathbb{E}(X) = 0 \]

- **Claim:** Small deformations (not parametric) can have huge effects:

  Ex.:
  
  \[ x \in L^2(\mathbb{R}^n), \tau \in \mathcal{C}^\infty \text{ define } L_\tau x(u) = x(u - \tau(u)) \]
  
  \[ \tau(u) = \epsilon, \mathcal{C} \subset \mathbb{R}^2, \|1_C - L_\tau 1_C\| = 2 \]

- The variance is **high**, and the bias is difficult to estimate. There are also few available samples...

How to handle that?

\[ \|x - y\|_2 = 2 \]
Image variabilities

Geometric variability

Groups acting on images:
translation, rotation, scaling

Class variability

Intraclass variability
Not informative

Extraclass variability

Other sources: luminosity, occlusion, small deformations

\[ L_\tau x(u) = x(u - \tau(u)), \tau \in C^\infty \]

High variance: how to reduce it?
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 $\Phi$ consist of building an approximation of a low dimensional space which is regular with respect to the class:

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

- **Necessary** dimensionality reduction
Averaging makes euclidean distance meaningful in high dimension.

Translation

\[ \| x - y \|_2 = 2 \]

Rotation

Averaging is the key to get invariants

Averaging makes euclidean distance meaningful in high dimension
An example: Invariance to translation

- In many cases, one wish to be invariant globally to translation, a simple way is to perform an averaging:

\[ Ax = \int L_\alpha x da = \int x(u) du \quad \text{It's the 0 frequency!} \]

- Even if it can be localized, the averaging keeps the low frequency structures: the invariance brings a loss of information!

- Bias issue! How do we recover the missing information?
Necessary mechanism: Separation - Contraction

• In high dimension, typical distances are huge, thus an appropriate representation must contract the space:
  \[ \| \Phi x - \Phi x' \| \leq \| x - x' \| \]

• While avoiding the different classes to collapse:
  \[ \exists \epsilon > 0, y(x) \neq y(x') \Rightarrow \| \Phi x - \Phi x' \| \geq \epsilon \]
Deep learning: Technical breakthrough

• Deep learning has permitted to **solve** a large number of task that were considered as extremely challenging for a computer.

• The technique that is used is **generic** and its success implies that it reduces those sources of variability.

• Previous properties hold for deep learning.

• How, why?
A Krizhevsky et al.

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 are available. Many rely on a "manifold hypothesis". Clearly wrong:
  Ex: stability to diffeomorphisms

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

- No generalisation result. Rademacher complexity cannot explain the generalization properties.

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

C. Szegedy et al.

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

Ref.: Deep Roto-Translation Scattering for Object Classification.
EO and S Mallat
Organization is a key

- Consider a problem of questionnaires: people answer to 0 or 1 to some question. What does structuration means?

In general, works tackle only one of the aspect

Ref.: Harmonic Analysis of Digital Data Bases
Coifman R. et al.
Organization permits creation of invariance

- As (all) the sources of regularities are obtained, interpolating new points is possible (in statistical terms: generalisation property!)

- In the previous case, one can build a discriminative and invariant representation: Haar wavelets on graphs for example.
Organising the CNN representation: Local Support Vectors

- Let’s consider a CNN of depth $J$.
  
  **Local dimension is intractable!**

- Local Support Vectors of order $k$ at depth $j$: representations at depth $j$ that are well classified by a $k$-NN but not by a $l$-NN for $l<k$

- They give a measure of the separation-contraction via:

\[
\Gamma_{j}^{k+1} = \{ x_{j} \in \Gamma_{j}^{k} | \text{card}\{y(x_{j}^{(l)}) \neq y(x_{j}^{(l)}), l \leq k + 1\} > \frac{k}{2}\}
\]

\[
x_{j}^{(l)}: l\text{-NN at depth } j
\]
Complexity measure

# of k-local support vectors at different depth $n$

Slow decay to stationary regime indicates high complexity (separation)

Small amount indicates contraction
An organisation of the representation

• There is a progressive localisation which explains why a 1-NN (or a Gaussian SVM) works better with depth:

• How do the representation got localized? Necessary variability reduction
Identifying the variabilities?

• Several works showed a Deepnet exhibits some covariance:

  (c) Object color  
  (d) Background color

• Manifold of faces at a certain depth:

• Can we use these?
Linearizing variabilities

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

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

example: Scattering Transform
Symmetry group hypothesis

- To each classification problem corresponds a canonic and unique symmetry group $G$: $\forall x, \forall g \in G, \Phi x = \Phi g.x$

- We hypothesise there exists Lie groups and CNNs such that:

  $G_0 \subset G_1 \subset \ldots \subset G_J \subset G$

  $\forall g_j \in G_j, \phi_j(g_j.x) = \phi_j(x)$ where $x_j = \phi_j(x)$

- Examples are given by the euclidean group:

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

Ref.: Understanding deep convolutional networks
S Mallat
Structuring the input with the Scattering Transform

- Scattering Transform $S_J$ is a local descriptor of neighbourhood of amplitude $2^J$.

- It is a representation built via geometry with limited learning. (≈SIFT)

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

- Successfully used in several applications:
  - Digits
    
    ![Digits Example](image)

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

  - Textures
    
    ![Textures Example](image)
Wavelets

- Wavelets help to describe signal structures. $\psi$ is a wavelet iff
  \[ \psi \in \mathcal{L}^2(\mathbb{R}^2, \mathbb{C}) \text{ and } \int_{\mathbb{R}^2} \psi(u) du = 0 \]
- They are chosen localised in space and frequency.

- Wavelets can be dilated in order to be a **multi-scale** representation of signals, **rotated** to describe rotations.
  \[ \psi_{j,\theta} = \frac{1}{2^j} \psi\left(\frac{r_{\theta}(u)}{2^j}\right) \]
- Design wavelets selective to an **informative** variability.

Isotropic $|\hat{\psi}|$ vs Non-Isotropic $|\hat{\psi}|$
The Gabor wavelet

\[
\psi(u) = \frac{1}{2\pi\sigma} e^{-\frac{\|u\|^2}{2\sigma}} (e^{i\xi \cdot u} - \kappa)
\]

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

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

Heisenberg principle!
Good localisation in space and Fourier
Wavelet Transform

• Wavelet transform: \( Wx = \{ x \ast \psi_{j,\theta}, x \ast \phi_J \}_{\theta,j \leq J} \)

• Isometric and linear operator of \( L^2 \) with

\[
\| Wx \|^2 = \sum_{\theta,j \leq J} \int |x \ast \psi_{j,\theta}|^2 + \int x \ast \phi_J^2
\]

• Covariant with translation \( L_a \):

\[
WL_a = L_a W
\]

• Nearly commutes with diffeomorphisms

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

• A good baseline to describe an image!

Ref.: Group Invariant Scattering, Mallat S
Filter bank implementation of a Fast WT

• Assume it is possible to find \( h \) and \( g \) such that
\[
\hat{\psi}_\theta(\omega) = \frac{1}{\sqrt{2}} \hat{g}_\theta(\omega) \hat{\phi}(\omega / 2) \quad \text{and} \quad \hat{\phi}(\omega) = \frac{1}{\sqrt{2}} \hat{h}_\theta(\omega / 2) \hat{\phi}(\omega / 2)
\]

• Set:
\[
x_j(u, 0) = x \star \phi_j(u) = h \star (x \star \phi_{j-1})(2u) \quad \text{and} \quad x_j(u, \theta) = x \star \psi_{j, \theta}(u) = g_\theta \star (x \star \phi_{j-1})(2u)
\]

• The WT is then given by \( W x = \{x_j(., \theta), x_J(., 0)\}_{j \leq J, \theta} \)

• A WT can be interpreted as a deep cascade of linear operator, which is approximatively verified for the Gabor Wavelets.
\[ \hat{\phi}_j = \frac{1}{\sqrt{2}} \hat{h}(\cdot) \hat{\phi}_{j-1} \]

\[ \hat{\psi}_{j, \theta} = \frac{1}{\sqrt{2}} \hat{g}_\theta(\cdot) \hat{\phi}_{j-1} \]

There is an oversampling

\[ h \geq 0 \]

Implementation of a WT
Scattering Transform

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

$$S_J x = \left\{ x \ast \phi_J, \right. \quad \left. |x \ast \psi_{\lambda_1}| \ast \phi_J, \right. \quad \left. ||x \ast \psi_{\lambda_1}| \ast \psi_{\lambda_2}| \ast \phi_J \right\}$$

with $\lambda_i = \{j_i, \theta_i\}, j_i \leq J$

- Mathematically well defined for a large class of wavelets.
$J = 3, \theta \in \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$

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

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

Scattering coefficients are only at the output

$\hat{h} \geq 0$

Modulus

Ref.: Deep Roto-Translation Scattering for Object Classification. EO and S Mallat
Analytic wavelets and modulus?

- For any translations:
  \[
  L_a x \star \psi(\omega) = e^{i\omega^T a} \hat{x}(\omega) \hat{\psi}(\omega)
  = \sum_n \frac{(i\omega^T a)^n}{n!} \hat{x}(\omega) \hat{\psi}(\omega)
  \approx \sum_n \frac{(i\omega_0^T a)^n}{n!} \hat{x}(\omega) \hat{\psi}(\omega)
  = e^{i\omega_0^T a} x \star \psi(\omega)
  \]

- A modulus removes the phase!

Ref.: Group Invariant Scattering, Mallat S

The infinitesimal generator of translations is the derivative...

Non-linear projection
Information loss

Reconstruction

$$\arg \inf_y \| S_3 x - S_3 y \|$$

invariance up to

$$2^3$$ pixels
Wavelets on Lie group

- Discovering more complex groups is necessary to build more complex invariants:

\[
\mathbb{R}^2 \rightarrow SO_2(\mathbb{R}) \times \mathbb{R}^2 \rightarrow \ldots
\]

Translation Scattering does not see the difference

- A wavelet is defined by \( \psi \in L^2(G) \), \( \hat{\psi}(e) = 0 \) and can be dilated via \( \psi_\lambda = L_\lambda \psi \)

- **Theorem:** Let \( G \) be a compact Lie group, for appropriate mother wavelet \( \psi \) and \( \Lambda \) then

\[
W x = \left\{ \int_G x, x \ast^G \psi_\lambda \right\}_{\lambda \in \Lambda}
\]

is an isometry and covariant with the action of \( G \)

- **Proposition:** \( W \) almost commutes with deformations but is not invariant to translation...

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

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

Ref.: Stein, E. M. Topics in harmonic analysis related to the Littlewood-Paley theory.

Ref.: Group Invariant Scattering, Mallat S
An ideal input for a modern CNN

- 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 by local translation:
  \[ |a| \ll 2^J \Rightarrow S_J L_a x \approx S_J \]
- For \( \lambda, u \), \( S_J x(u, \lambda) \) has a topology that is structured by \( SO_2(\mathbb{R}) \), and this structures the first layer also:

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

  \[
  S_J(g.x)(u, \lambda) = S_J x(g^{-1}u, g^{-1}\lambda) \triangleq g.S_J x(u, \lambda)
  \]
## Data

### How much learning is really required?

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Type</th>
<th>Paper</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caltech101</td>
<td>Scattering</td>
<td>Ask the locals</td>
<td>79.9</td>
</tr>
<tr>
<td></td>
<td>Unsupervised</td>
<td>DeepNet</td>
<td>77.3</td>
</tr>
<tr>
<td></td>
<td>Supervised</td>
<td>DeepNet</td>
<td>91.4</td>
</tr>
<tr>
<td>CIFAR100</td>
<td>Scattering</td>
<td>RFL</td>
<td>54.2</td>
</tr>
<tr>
<td></td>
<td>Unsupervised</td>
<td>DeepNet</td>
<td>65.4</td>
</tr>
<tr>
<td></td>
<td>Supervised</td>
<td>DeepNet</td>
<td></td>
</tr>
</tbody>
</table>

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

---

**Group representations are competitive with representations learned from data without labels**
Benchmarking ImageNet

- Cascading a modern CNN leads to almost state-of-the-art result on ImageNet2012:

<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 5</th>
<th>Params</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlexNet</td>
<td>56.9</td>
<td>80.1</td>
<td>61M</td>
</tr>
<tr>
<td>VGG-16</td>
<td>68.5</td>
<td>88.7</td>
<td>138M</td>
</tr>
<tr>
<td>Scat + Resnet-10 (ours)</td>
<td>68.7</td>
<td>88.6</td>
<td>12.8M</td>
</tr>
<tr>
<td>Resnet-18 (ours)</td>
<td>68.9</td>
<td>88.8</td>
<td>11.7M</td>
</tr>
<tr>
<td>Resnet-200</td>
<td><strong>78.3</strong></td>
<td><strong>94.2</strong></td>
<td>64.7M</td>
</tr>
</tbody>
</table>

- Demonstrates no loss of information + Less layers

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

- It is equivalent to encode the non-overlapping scattering patches: the output of the $1 \times 1$ is a local descriptor of an image that leads to AlexNet performances.

- Good generalization on Caltech101

**Table:**

<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>FV + FC</td>
<td>55.6</td>
<td>78.4</td>
</tr>
<tr>
<td>FV + SVM</td>
<td>54.3</td>
<td>74.3</td>
</tr>
<tr>
<td>AlexNet</td>
<td>56.9</td>
<td>80.1</td>
</tr>
<tr>
<td>Scat + SLE</td>
<td>57.0</td>
<td>79.6</td>
</tr>
</tbody>
</table>

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

- Adding geometric prior regularises the CNN input, in the particular case of limited samples situations, without reducing the number of parameters.

- State-of-the-art results on STL10 and CIFAR10:
  - STL10: 5k training, 8k testing, 10 classes
  - Cifar10, 10 classes
    - Keeping 100, 500 and 1000 samples and testing on 10k

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised methods</td>
<td></td>
</tr>
<tr>
<td>Scat + WRN 19-8</td>
<td>76.0 ± 0.6</td>
</tr>
<tr>
<td>CNN</td>
<td>70.1 ± 0.6</td>
</tr>
<tr>
<td>Unsupervised methods</td>
<td></td>
</tr>
<tr>
<td>Exemplar CNN</td>
<td>75.4 ± 0.3</td>
</tr>
<tr>
<td>Stacked what-where AE</td>
<td>74.33</td>
</tr>
<tr>
<td>Hierarchical Matching Pursuit (HMP)</td>
<td>64.5±1</td>
</tr>
<tr>
<td>Convolutional K-means Network</td>
<td>60.1±1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>100</th>
<th>500</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>WRN 16-8</td>
<td>34.7 ± 0.8</td>
<td>46.5 ± 1.4</td>
<td>60.0 ± 1.8</td>
</tr>
<tr>
<td>Scat + WRN 12-8</td>
<td><strong>38.9 ± 1.2</strong></td>
<td><strong>54.7 ± 0.6</strong></td>
<td><strong>62.0 ± 1.1</strong></td>
</tr>
</tbody>
</table>

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

- We evaluate the angular energy propagated for given frequencies:
  \[ \Omega(\omega_{\theta_1}, \omega_{\theta_2}) = \sum |W_1(., \omega_{\theta_1}, \omega_{\theta_2})|^2 \]

- They are all localised in the low-frequency domain: invariance to rotation is learned. (supports symmetry group hypothesis)

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

- Can we structure the next layers?

- Introduce a CNN that is convolutional along each direction, finally averaged:

\[
x_{j+1} = \rho_j W_j x_j
\]

\[
x_{j+1}(v_1, \ldots, v_j, v_{j+1}) = \rho_j (x_j * v_1, \ldots, v_j \psi v_{j+1})(v_1, \ldots, v_j)
\]

\[
x_J = \sum_{v_j, j \leq J-2} x_{J-1}(v_1, \ldots, v_{J-1})
\]

- For \(x_j\), we refer to the variable \(v_j\) as an attribute that discriminates previously obtained tensor.

- \(W_j\) performs an averaging along \(v_{j-2}\).
Flattening the variability

- An explicit invariant of any translations along \((v_1, \ldots, v_j)\) is built.

- Completely structures the axis of the "channels" via convolutions.

- It aims at mapping the symmetries of \(\Phi x = x_J\) into the translations along \(G_j = \mathbb{R}^j, j \leq J\).

Organizing the channels indexes

Ref.: Multiscale Hierarchical Convolutional Networks, J Jacobsen, EO, S Mallat, Smeulders AWM
Reducing the number of parameters

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

### CIFAR10

<table>
<thead>
<tr>
<th>Model</th>
<th># Parameters</th>
<th>% Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchical CNN</td>
<td>0.098M</td>
<td>91.43</td>
</tr>
<tr>
<td>Hierarchical CNN (+)</td>
<td>0.34M</td>
<td>92.50</td>
</tr>
<tr>
<td>ALL-CNN</td>
<td>1.3M</td>
<td>92.75</td>
</tr>
<tr>
<td>ResNet</td>
<td>0.27M</td>
<td>91.25</td>
</tr>
<tr>
<td>Network in Network</td>
<td>0.98M</td>
<td>91.20</td>
</tr>
<tr>
<td>WRN-student</td>
<td>0.17M</td>
<td>91.23</td>
</tr>
<tr>
<td>FitNet</td>
<td>2.5M</td>
<td>91.61</td>
</tr>
</tbody>
</table>

This implies an effective structuration

### CIFAR100

<table>
<thead>
<tr>
<th>Model</th>
<th># Parameters</th>
<th>% Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchical CNN</td>
<td>0.25M</td>
<td>62.01</td>
</tr>
<tr>
<td>Hierarchical CNN (+)</td>
<td>0.89M</td>
<td>63.19</td>
</tr>
<tr>
<td>ALL-CNN</td>
<td>1.3M</td>
<td>66.29</td>
</tr>
<tr>
<td>Network in Network</td>
<td>0.98M</td>
<td>64.32</td>
</tr>
<tr>
<td>FitNet</td>
<td>2.5M</td>
<td>64.96</td>
</tr>
</tbody>
</table>
We observe that representations at several layers are translated:

**Bird 1**

$x$

$\Phi x$

**Bird 2**

$x$

$\Phi x$
Conclusion

• Structuration should be the topic of future research to improve Deep neural networks

• Check my webpage for softwares and papers: http://www.di.ens.fr/~oyallon/

Thank you!