Sparse Coding and Dictionary Learning for Image Analysis

Part IV: Recent Advances in Computer Vision and New Models

Francis Bach, Julien Mairal, Jean Ponce and Guillermo Sapiro

CVPR’10 tutorial, San Francisco, 14th June 2010
What this part is about

- Learning dictionaries for discriminative tasks.
- ...and adapted to image classification tasks.
- Structured Sparse Models.
Learning dictionaries with a discriminative cost function

Idea:
Let us consider 2 sets $S_-, S_+$ of signals representing 2 different classes. Each set should admit a dictionary best adapted to its reconstruction.

Classification procedure for a signal $x \in \mathbb{R}^n$:

$$\min (R^*(x, D_-), R^*(x, D_+))$$

where

$$R^*(x, D) = \min_{\alpha \in \mathbb{R}^p} ||x - D\alpha||_2^2 \text{ s.t. } ||\alpha||_0 \leq L.$$ 

“Reconstructive” training

$$\left\{ \begin{array}{l} \min_{D_-} \sum_{i \in S_-} R^*(x_i, D_-) \\ \min_{D_+} \sum_{i \in S_+} R^*(x_i, D_+) \end{array} \right.$$ 

[Grosse et al., 2007], [Huang and Aviyente, 2006], [Sprechmann et al., 2010b] for unsupervised clustering (CVPR ’10)
Learning dictionaries with a discriminative cost function

“Discriminative” training

[Mairal, Bach, Ponce, Sapiro, and Zisserman, 2008a]

$$\min_{D_-, D_+} \sum_i C\left(\lambda z_i (R^*(x_i, D_-) - R^*(x_i, D_+))\right),$$

where $z_i \in \{-1, +1\}$ is the label of $x_i$. 

Logistic regression function
Learning dictionaries with a discriminative cost function

Mixed approach

\[
\min_{D_{-},D_{+}} \sum_{i} C \left( \lambda z_{i} (R^{*}(x_{i}, D_{-}) - R^{*}(x_{i}, D_{+})) \right) + \mu R^{*}(x_{i}, D_{z_{i}}),
\]

where \( z_{i} \in \{-1, +1\} \) is the label of \( x_{i} \).

Keys of the optimization framework

- Alternation of sparse coding and dictionary updates.
- Continuation path with decreasing values of \( \mu \).
- OMP to address the NP-hard sparse coding problem.
- ... or LARS when using \( \ell_{1} \).
- Use softmax instead of logistic regression for \( N > 2 \) classes.
Learning dictionaries with a discriminative cost function

Examples of dictionaries

Top: reconstructive, Bottom: discriminative, Left: Bicycle, Right: Background.
Learning dictionaries with a discriminative cost function
Texture segmentation
Learning dictionaries with a discriminative cost function
Texture segmentation
Learning dictionaries with a discriminative cost function
Pixelwise classification
Learning dictionaries with a discriminative cost function

Multiscale scheme

Signal input → Subsampling → Sparse coding → Classifier 1 → Classifier 2 → Classifier 3 → Linear classifier
Learning dictionaries with a discriminative cost function
weakly-supervised pixel classification

Francis Bach, Julien Mairal, Jean Ponce and Guillermo Sapiro
Application to edge detection and classification

[Mairal, Leordeanu, Bach, Hebert, and Ponce, 2008b]

Good edges

Bad edges
Application to edge detection and classification
Berkeley segmentation benchmark

Raw edge detection on the right
Application to edge detection and classification
Berkeley segmentation benchmark

Raw edge detection on the right
Application to edge detection and classification
Berkeley segmentation benchmark
Application to edge detection and classification
Contour-based classifier: [Leordeanu, Hebert, and Sukthankar, 2007]

Is there a bike, a motorbike, a car or a person on this image?
Application to edge detection and classification

<table>
<thead>
<tr>
<th>Input Contours</th>
<th>Bike Edge Detector</th>
<th>Bottle Edge Detector</th>
<th>People Edge Detector</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Input Contours" /></td>
<td><img src="image2" alt="Bike Edge Detector" /></td>
<td><img src="image3" alt="Bottle Edge Detector" /></td>
<td><img src="image4" alt="People Edge Detector" /></td>
</tr>
</tbody>
</table>
Application to edge detection and classification
Performance gain due to the prefiltering

<table>
<thead>
<tr>
<th>Category</th>
<th>Ours + [Leordeanu '07]</th>
<th>[Leordeanu '07]</th>
<th>[Winn '05]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aeroplane</td>
<td>71.9%</td>
<td>61.9%</td>
<td></td>
</tr>
<tr>
<td>Boat</td>
<td>67.1%</td>
<td>56.4%</td>
<td></td>
</tr>
<tr>
<td>Cat</td>
<td>82.6%</td>
<td>53.4%</td>
<td></td>
</tr>
<tr>
<td>Cow</td>
<td>68.7%</td>
<td>59.2%</td>
<td></td>
</tr>
<tr>
<td>Horse</td>
<td>76.0%</td>
<td>67%</td>
<td></td>
</tr>
<tr>
<td>Motorbike</td>
<td>80.6%</td>
<td>73.6%</td>
<td></td>
</tr>
<tr>
<td>Sheep</td>
<td>72.9%</td>
<td>58.4%</td>
<td></td>
</tr>
<tr>
<td>Tvmonitor</td>
<td>87.7%</td>
<td>83.8%</td>
<td></td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>75.9%</strong></td>
<td><strong>64.2%</strong></td>
<td></td>
</tr>
</tbody>
</table>

Recognition rates for the same experiment as [Winn et al., 2005] on VOC 2005.

Recognition performance at equal error rate for 8 classes on a subset of images from Pascal 07.
Digital Art Authentification
Data Courtesy of Hugues, Graham, and Rockmore [2009]

Authentic

Fake
Digital Art Authentification
Data Courtesy of Hugues, Graham, and Rockmore [2009]

Authentic

Fake

Fake
Digital Art Authentification
Data Courtesy of Hugues, Graham, and Rockmore [2009]

Authentic

Fake

Authentic
Image Half-Toning
Image Half-Toning
Learning Codebooks for Image Classification

**Idea**

Replacing Vector Quantization by Learned Dictionaries!

- unsupervised: [Yang et al., 2009]
- supervised: [Boureau et al., 2010, Yang et al., 2010] (CVPR ’10)
Learning Codebooks for Image Classification

Let an image be represented by a set of low-level descriptors \( x_i \) at \( N \) locations identified with their indices \( i = 1, \ldots, N \).

- **hard-quantization:**
  \[
  x_i \approx D \alpha_i, \quad \alpha_i \in \{0, 1\}^p \quad \text{and} \quad \sum_{j=1}^p \alpha_i[j] = 1
  \]

- **soft-quantization:**
  \[
  \alpha_i[j] = \frac{e^{-\beta \|x_i - d_j\|_2^2}}{\sum_{k=1}^p e^{-\beta \|x_i - d_k\|_2^2}}
  \]

- **sparse coding:**
  \[
  x_i \approx D \alpha_i, \quad \alpha_i = \arg\min_{\alpha} \frac{1}{2} \|x_i - D \alpha\|_2^2 + \lambda \|\alpha\|_1
  \]
Learning Codebooks for Image Classification
Table from Boureau et al. [2010]

<table>
<thead>
<tr>
<th>Method</th>
<th>Caltech-101, 30 training examples</th>
<th>15 Scenes, 100 training examples</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average Pool</td>
<td>Max Pool</td>
</tr>
<tr>
<td></td>
<td>Results with basic features, SIFT extracted each 8 pixels</td>
<td></td>
</tr>
<tr>
<td>Hard quantization, linear kernel</td>
<td>51.4 ± 0.9 [256] 64.3 ± 0.9 [256]</td>
<td>73.9 ± 0.9 [1024] 80.1 ± 0.6 [1024]</td>
</tr>
<tr>
<td>Hard quantization, intersection kernel</td>
<td>64.2 ± 1.0 [256] (1) 64.3 ± 0.9 [256]</td>
<td>80.8 ± 0.4 [256] (1) 80.1 ± 0.6 [1024]</td>
</tr>
<tr>
<td>Soft quantization, linear kernel</td>
<td>57.9 ± 1.5 [1024] 69.0 ± 0.8 [256]</td>
<td>75.6 ± 0.5 [1024] 81.4 ± 0.6 [1024]</td>
</tr>
<tr>
<td>Soft quantization, intersection kernel</td>
<td>66.1 ± 1.2 [512] (2) 70.6 ± 1.0 [1024]</td>
<td>81.2 ± 0.4 [1024] (2) 83.0 ± 0.7 [1024]</td>
</tr>
<tr>
<td>Sparse codes, linear kernel</td>
<td>61.3 ± 1.3 [1024] 71.5 ± 1.1 [1024] (3)</td>
<td>76.9 ± 0.6 [1024] 83.1 ± 0.6 [1024] (3)</td>
</tr>
<tr>
<td>Sparse codes, intersection kernel</td>
<td>70.3 ± 1.3 [1024] 71.8 ± 1.0 [1024] (4)</td>
<td>83.2 ± 0.4 [1024] 84.1 ± 0.5 [1024] (4)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Results with macrofeatures and denser SIFT sampling</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average Pool</td>
</tr>
<tr>
<td>Hard quantization, linear kernel</td>
<td>55.6 ± 1.6 [256] 70.9 ± 1.0 [1024]</td>
</tr>
<tr>
<td>Hard quantization, intersection kernel</td>
<td>68.8 ± 1.4 [512] 70.9 ± 1.0 [1024]</td>
</tr>
<tr>
<td>Soft quantization, linear kernel</td>
<td>61.6 ± 1.6 [1024] 71.5 ± 1.0 [1024]</td>
</tr>
<tr>
<td>Soft quantization, intersection kernel</td>
<td>70.1 ± 1.3 [1024] 73.2 ± 1.0 [1024]</td>
</tr>
<tr>
<td>Sparse codes, linear kernel</td>
<td>65.7 ± 1.4 [1024] 75.1 ± 0.9 [1024]</td>
</tr>
<tr>
<td>Sparse codes, intersection kernel</td>
<td>73.7 ± 1.3 [1024] 75.7 ± 1.1 [1024]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unsup</th>
<th>Discr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>83.6 ± 0.4</td>
</tr>
<tr>
<td>Intersect</td>
<td>84.3 ± 0.5</td>
</tr>
</tbody>
</table>

Yang et al. [2009] have won the PASCAL VOC’09 challenge using this kind of techniques.
Summary so far

- Learned dictionaries are well adapted to model images.
- They can be used to learn dictionaries of SIFT features.
- They are also adapted to discriminative tasks.
Sparse Structured Linear Model

- We focus again on linear models

\[ x \approx D\alpha. \]

- \( x \in \mathbb{R}^m \), vector of \( m \) observations.
- \( D \in \mathbb{R}^{m \times p} \), dictionary or data matrix.
- \( \alpha \in \mathbb{R}^p \), loading vector.

Assumptions:

- \( \alpha \) is \textbf{sparse}, i.e., it has a small support

\[ |\Gamma| \ll p, \quad \Gamma = \{ j \in \{1, \ldots, p\}; \ \alpha_j \neq 0 \}. \]

- The support, or nonzero pattern, \( \Gamma \) is \textbf{structured}:
  - \( \Gamma \) reflects spatial/geometrical/temporal... information.
  - e.g., 2-D grid for features associated to the pixels of an image.
Sparsity-Inducing Norms (1/2)

\[
\min_{\alpha \in \mathbb{R}^p} \left( f(\alpha) + \lambda \psi(\alpha) \right)
\]

**Standard approach to enforce sparsity in learning procedures:**

- Regularizing by a **sparsity-inducing norm** \( \psi \).
- The effect of \( \psi \) is to set some \( \alpha_j \)'s to zero, depending on the regularization parameter \( \lambda \geq 0 \).

**The most popular choice for \( \psi \):**

- The \( \ell_1 \) norm, \( \|\alpha\|_1 = \sum_{j=1}^{p} |\alpha_j| \).
- For the square loss, Lasso [Tibshirani, 1996].
- However, the \( \ell_1 \) norm encodes poor information, just **cardinality**!
Another popular choice for $\psi$:

- The $\ell_1$-$\ell_2$ norm,

$$
\sum_{G \in \mathcal{G}} \|\alpha_G\|_2 = \sum_{G \in \mathcal{G}} \left( \sum_{j \in G} \alpha_j^2 \right)^{1/2}, \text{ with } \mathcal{G} \text{ a partition of } \{1, \ldots, p\}.
$$

- The $\ell_1$-$\ell_2$ norm sets to zero groups of non-overlapping variables (as opposed to single variables for the $\ell_1$ norm).
- For the square loss, group Lasso [Yuan and Lin, 2006].
- However, the $\ell_1$-$\ell_2$ norm encodes fixed/static prior information, requires to know in advance how to group the variables!

Questions:

- What happen if the set of groups $\mathcal{G}$ is not a partition anymore?
- What is the relationship between $\mathcal{G}$ and the sparsifying effect of $\psi$?
Structured Sparsity

[Jenatton et al., 2009]

Case of general overlapping groups.

When penalizing by the \( \ell_1 - \ell_2 \) norm,

\[
\sum_{G \in G} \| \alpha_G \|_2 = \sum_{G \in G} \left( \sum_{j \in G} \alpha_j^2 \right)^{1/2}
\]

- The \( \ell_1 \) norm induces sparsity at the group level:
  - Some \( \alpha_G \)'s are set to zero.
- Inside the groups, the \( \ell_2 \) norm does not promote sparsity.
- Intuitively, variables belonging to the same groups are encouraged to be set to zero together.
- Optimization via reweighted least-squares, proximal methods, etc.
Examples of set of groups $\mathcal{G}$ (1/3)

Selection of contiguous patterns on a sequence, $p = 6$.

$\mathcal{G}$ is the set of blue groups.

Any union of blue groups set to zero leads to the selection of a contiguous pattern.
Examples of set of groups $\mathcal{G}$ (2/3)

Selection of rectangles on a 2-D grids, $p = 25$.

$\mathcal{G}$ is the set of blue/green groups (with their not displayed complements).

Any union of blue/green groups set to zero leads to the selection of a rectangle.
Selection of diamond-shaped patterns on a 2-D grids, $p = 25$.

- It is possible to extent such settings to 3-D space, or more complex topologies.
Overview of other work on structured sparsity

- Specific hierarchical structure [Zhao et al., 2009, Bach, 2008].
- **Union-closed** (as opposed to intersection-closed) family of nonzero patterns [Baraniuk et al., 2010, Jacob et al., 2009].
- Nonconvex penalties based on information-theoretic criteria with greedy optimization [Huang et al., 2009].
- Structure expressed through a Bayesian prior, e.g., [He and Carin, 2009].
Hierarchical Dictionaries

[Jenatton, Mairal, Obozinski, and Bach, 2010]

A node can be active only if its ancestors are active. The selected patterns are rooted subtrees.

Optimization via efficient proximal methods (same cost as $\ell_1$)
Hierarchical Dictionaries

[Jenatton, Mairal, Obozinski, and Bach, 2010]
Group Lasso + $\ell_1 = \text{Collaborative Hierarchical Lasso}$

[Sprechmann, Ramirez, Sapiro, and Eldar, 2010a]

Optimization also via proximal methods
Topographic Dictionaries

“Topographic” dictionaries [Hyvarinen and Hoyer, 2001, Kavukcuoglu et al., 2009] are a specific case of dictionaries learned with a structured sparsity regularization for $\alpha$.

**Figure**: Image obtained from [Kavukcuoglu et al., 2009]


References II


References IV


