Image classification : BoW is beautiful

Journée GdR ISIS

Apprentissage et reconnaissance des formes en signal et images

Matthieu Cord

Nicolas Thome, Hanlin Goh, Sandra Avila, Marc Law
Computer Science dept. (LIP6), UPMC-Paris 6, France

Dec 2012

Outline

Image classification pipeline

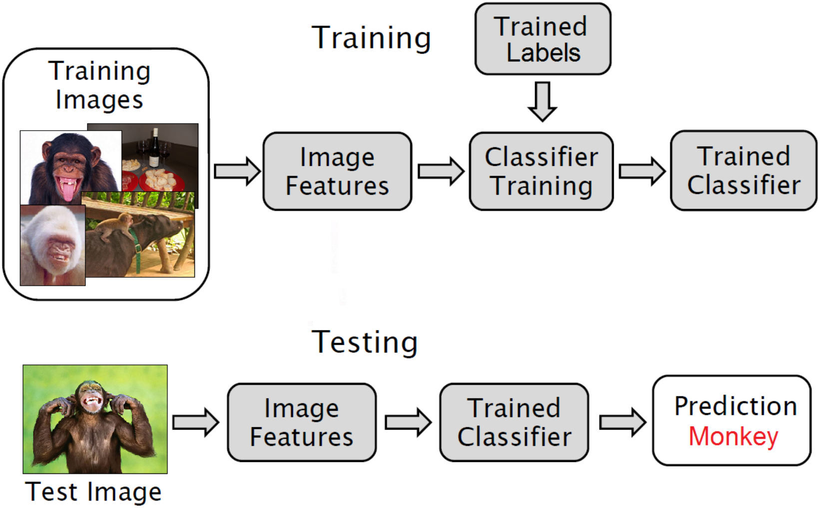
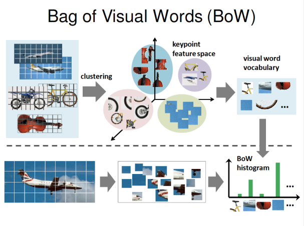


Image classification pipeline

Bag-of-Visual-Words (BoVW) Model



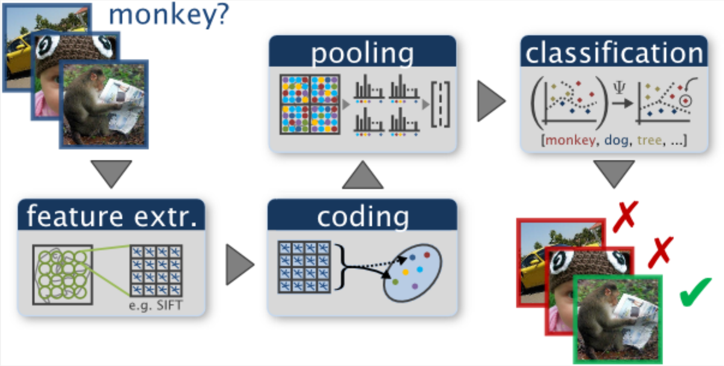
Credit: Prof. Shih-Fu Chang

- 1999 BoW on color [Ma Manjunath]
- 2001 BoW on Gabor [Fournier Cord]
- 2003-4 BoW on SIFT [Csurka]
- Spatial Information [*Lazebnik*06]
- Soft-assignement, sparse coding, max pooling [*Wang*10] [*Boureau*10]

[*Lazebnik*06] P.Lazebnik.S, Schmid.C. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories CVPR2006.
[*Boureau*10]Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition CVPR2010.
[*Wang*10]J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong.Locality-constrained linear coding for image classification CVPR2010.

Image classification: BoW details

Coding/Pooling



BoW Model

$\mathbf{X} = (x_1, \ldots, x_j, \ldots, x_N)$ the set of local descriptors (SIFT) for the image
$\mathbf{C} = (c_1, \ldots, c_m, \ldots, c_M)$ the visual dictionary

$$
\begin{array}{ccc}
 & & \begin{array}{ccccc} \mathbf{x}_1 & & \mathbf{x}_j & & \mathbf{x}_N \end{array} \\
\mathbf{H} \;=\; & \begin{array}{c} \mathbf{c}_1 \\ \\ \mathbf{c}_m \\ \\ \mathbf{c}_M \end{array} & \begin{bmatrix} \alpha_{1,1} & \cdots & \alpha_{1,j} & \cdots & \alpha_{1,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{m,1} & \cdots & \alpha_{m,j} & \cdots & \alpha_{m,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{M,1} & \cdots & \alpha_{M,j} & \cdots & \alpha_{M,N} \end{bmatrix} \Rightarrow g\text{: pooling} \\
 & & \Downarrow \\
 & & f\text{: coding}
\end{array}
$$

BoW Model

$\mathbf{X} = (x_1, \ldots, x_j, \ldots, x_N)$ the set of local descriptors (SIFT) for the image
$\mathbf{C} = (c_1, \ldots, c_m, \ldots, c_M)$ the visual dictionary

$$\mathbf{H} = \begin{array}{cc} & \begin{array}{ccccc} \mathbf{x}_1 & & \mathbf{x}_j & & \mathbf{x}_N \end{array} \\ \begin{array}{c} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_m \\ \vdots \\ \mathbf{c}_M \end{array} & \begin{bmatrix} \alpha_{1,1} & \cdots & \alpha_{1,j} & \cdots & \alpha_{1,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{m,1} & \cdots & \alpha_{m,j} & \cdots & \alpha_{m,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{M,1} & \cdots & \alpha_{M,j} & \cdots & \alpha_{M,N} \end{bmatrix} \end{array} \Rightarrow g\text{: } \textit{pooling}$$

$$\Downarrow$$

$$f\text{: } \textit{coding}$$

Coding: $\mathbf{x}_j \rightarrow f(\mathbf{x}_j) = \{\alpha_{m,j}\}, \quad \alpha_{m,j} = 1 \text{ iff } m = \underset{k \in \{1,\ldots,M\}}{\arg\min} \|\mathbf{x}_j - \mathbf{c}_k\|_2^2$

BoW Model

$\mathbf{X} = (x_1, \ldots, x_j, \ldots, x_N)$ the set of local descriptors (SIFT) for the image
$\mathbf{C} = (c_1, \ldots, c_m, \ldots, c_M)$ the visual dictionary

$$
\mathbf{H} = \begin{array}{c} \\ \mathbf{c}_1 \\ \\ \mathbf{c}_m \\ \\ \mathbf{c}_M \end{array}
\begin{array}{c} \begin{array}{ccccc} \mathbf{x}_1 & & \mathbf{x}_j & & \mathbf{x}_N \end{array} \\
\begin{bmatrix} \alpha_{1,1} & \cdots & \alpha_{1,j} & \cdots & \alpha_{1,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{m,1} & \cdots & \alpha_{m,j} & \cdots & \alpha_{m,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{M,1} & \cdots & \alpha_{M,j} & \cdots & \alpha_{M,N} \end{bmatrix} \end{array}
\Rightarrow g\text{: pooling}
$$

$$\Downarrow$$

$$f\text{: coding}$$

Coding: $\mathbf{x}_j \rightarrow f(\mathbf{x}_j) = \{\alpha_{m,j}\}, \quad \alpha_{m,j} = 1 \text{ iff } m = \underset{k \in \{1,\ldots,M\}}{\arg\min} \|\mathbf{x}_j - \mathbf{c}_k\|_2^2$

Pooling: $g(\{\alpha_j\}) = \mathbf{z} : \forall m, \; z_m = \sum_{j=1}^{N} \alpha_{m,j}$

BoW Model

$\mathbf{X} = (x_1, \ldots, x_j, \ldots, x_N)$ the set of local descriptors (SIFT) for the image
$\mathbf{C} = (c_1, \ldots, c_m, \ldots, c_M)$ the visual dictionary

$$\mathbf{H} \;=\; \begin{array}{c c} & \begin{array}{ccccc} \mathbf{x}_1 & & \mathbf{x}_j & & \mathbf{x}_N \end{array} \\ \begin{array}{c} \mathbf{c}_1 \\ \\ \mathbf{c}_m \\ \\ \mathbf{c}_M \end{array} & \begin{bmatrix} \alpha_{1,1} & \cdots & \alpha_{1,j} & \cdots & \alpha_{1,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{m,1} & \cdots & \alpha_{m,j} & \cdots & \alpha_{m,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{M,1} & \cdots & \alpha_{M,j} & \cdots & \alpha_{M,N} \end{bmatrix} \end{array} \Rightarrow g\text{: pooling}$$

$$\Downarrow$$

$$f\text{: coding}$$

Coding: $\mathbf{x}_j \rightarrow f(\mathbf{x}_j) = \{\alpha_{m,j}\}, \quad \alpha_{m,j} = 1 \text{ iff } m = \underset{k \in \{1,\ldots,M\}}{\arg\min} \|\mathbf{x}_j - \mathbf{c}_k\|_2^2$

Pooling: $g(\{\alpha_j\}) = \mathbf{z} : \; \forall m, \; z_m = \sum_{j=1}^{N} \alpha_{m,j}$

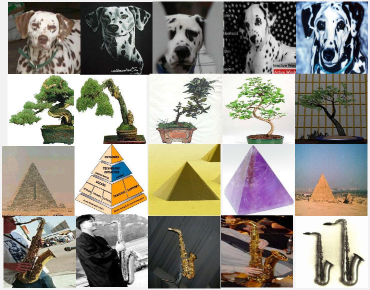
BoW representation: $\mathbf{z} = [z_1, z_2, \cdots, z_M]^T$

Results: Datasets

Many image datasets: Pascal VOC, Scene-15, MirFlickr, ...

Caltech101

- 9,144 images
- 102 categories
- 30 to 800 images per category



- Standard evaluation protocol for baseline comparison:
  - Train with 15-30 images / class
  - Test on the reamaining images
  - Metric: Multi-class Accuracy

Performance evaluation on Caltech101

Average accuracy results

	15 images	30 images
Bow-like architectures		
[Lazebnik&al CVPR06]	56.4	64.6
[Yang&al CVPR09] sparse coding	67.0	73.2

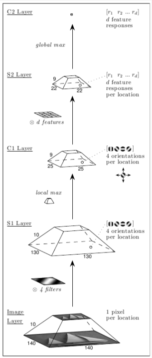
Hierarchical and biologically inspired architectures		
[Mutch&al IJCV08]	51	56
[Ranzato&al CVPR07]	-	54
[Jarrett09&al ICCV09]	-	65.6
[Zeiler&al CVPR2010]	58.6	66.9

BoW extensions:

- Extended coding, pooling
- Parametrization

Image classification: state-of-the-art

Biologically-inspired Methods [*Fidler*08, *Serre*07, *Mutch*08]



- Mimics feedforward properties of primate visual cortex V1 simple cells
- Based on the HMAX model [*Serre*07, *Mutch*08]
  - ⊕ Deep models
  - ⊕ Trainable with real images

[*Fidler*08]S. Fidler, B. Boben, and A. Leonardis. Similarity-based cross-layered hierarchical representation for object categorization CVPR2008.
[*Serre*07] T. Serre, *et.al*, Robust object recognition with cortex-like mechanisms, PAMI, 2007.
[*Mutch*08] Mutch.J and Lowe.D.G, Object class recognition and localization using sparse features with limited receptive fields, IJCV, 2008

Outline

Optimization of the BoW pipeline

with PhD M. Law, N. Thome [ECCVw 2012]

- Parametrization: find the Winner Cocktail
    - SR: Sampling Rate = gap between centers of patches (pixels)
    - Mono/Multi scale SIFT detection
    - Dictionary size
    - Normalization

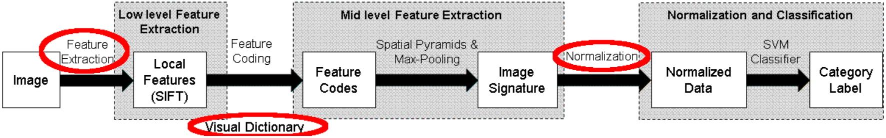


Figure: BoW pipeline for classification

SR	Scaling	Codebook Size	Accuracy (no norm)	Acc. (ℓ_2-norm)
8	mono	800	70.07 ± 0.96	70.46 ± 1.04
6	mono	800	71.64 ± 0.99	72.01 ± 0.96
3	mono	800	72.45 ± 1.05	72.73 ± 0.99
8	mono	1700	71.67 ± 0.93	71.95 ± 0.90
8	mono	3300	72.13 ± 0.99	72.50 ± 0.97
8	multi	800	73.35 ± 0.89	73.83 ± 0.96
8	multi	1700	75.34 ± 0.92	75.97 ± 0.86
8	multi	3300	76.91 ± 0.98	77.02 ± 0.94
3	multi	800	73.81 ± 0.95	73.99 ± 0.86
3	multi	1700	75.72 ± 1.13	76.00 ± 0.94
3	multi	3300	77.23 ± 1.02	77.47 ± 0.99
3	multi	6500	78.00 ± 1.05	78.46 ± 0.95

Table: Classification results on Caltech-101 with 30 training images per class

SR: Sampling Rate = gap between centers of patches (pixels)

Optimization of the BoW pipeline

- Impact of BoW parameters on PASCAL VOC 2007 and Caltech-101 [Chatfield, ...,Zisserman BMVC 2011]

Method		256	600	1500	2000	4000	8000
		codebook size					
(a) FK	Lin	77.78 ± 0.56	–	–	–	–	–
(b) LLC	Lin	–	73.10 ± 1.09	74.84 ± 0.67	75.75 ± 0.71	76.15 ± 0.59	76.95 ± 0.39
(c) LLC	Chi	–	72.30 ± 1.08	74.23 ± 0.62	75.24 ± 0.71	75.95 ± 0.57	76.62 ± 0.61
(d) VQ	Chi	–	72.65 ± 0.77	73.62 ± 0.51	73.93 ± 0.79	74.41 ± 1.04	74.23 ± 0.65
(e) KCB	Chi	–	73.38 ± 0.65	75.24 ± 0.63	75.50 ± 0.65	75.92 ± 0.63	75.93 ± 0.57

Figure: Classification results on the Caltech-101 Dataset [1]. **FK**: Fisher Kernel, **LLC**: Locality-constrained linear Coding [2], **KCB**: Kernel Codebook, **VQ**: baseline method (Lazebnik [3])

- Ultradense sampling ($\sim$ 50,000 features/image)
- Fisher Vector Size : $21 \times 2DK \simeq 21 \times 40,000 \simeq 850,000$ elements
- VLAD [Jégou&al CVPR 2010] and VLAT [Picard&Gosselin ICIP 2011]

Conclusion

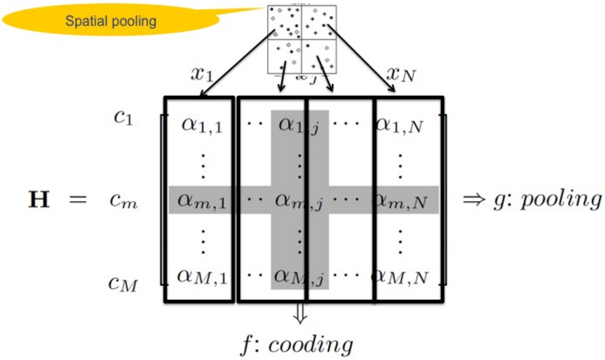
	[Law ECCVw 2012]		[Chatfield BMVC 2011]	
	Cal-101	Sc-15	VOC 07 (LLC)	VOC 07 (BoW)
Sampling Rate	X	X	X	X
Scaling	XXX	X		
Codebook Size	XXX	XXX	XXX	XXX
Normalization	X	X		

Table: Importance of parameters

- Huge performance difference according to the chain parameter tuning
  - the devil is in the (parameter) details ... (Chatfield's title)
- Fair comparisons: implementation details
- Sampling rate more important in mono-scale setup
- BoW has better results than Chatfield's reimplementation of FK on Caltech101 and much more compact

Outline

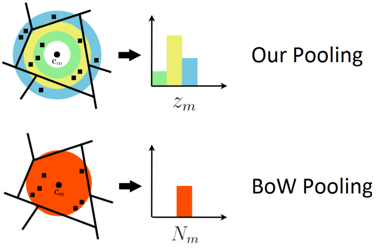
Pooling extension: Spatial Pyramid [Lazebnik et al., 2006]



Clusters for pooling in feature space [Boureau et al. ICCV 2011]

Pooling extension

- Pooling operator: averaging, max, Lp norm
- Learning in spatial pooling: spatial weight learned per visual word [Feng CVPR 2011] => supervised techniques (to learn classifiers and parts of the representation)
- A more information-preserving pooling operation: a distance-to-codeword distribution (BossaNova model)

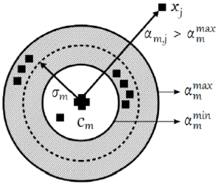


BossaNova Model (PhD's work of Sandra Avila)

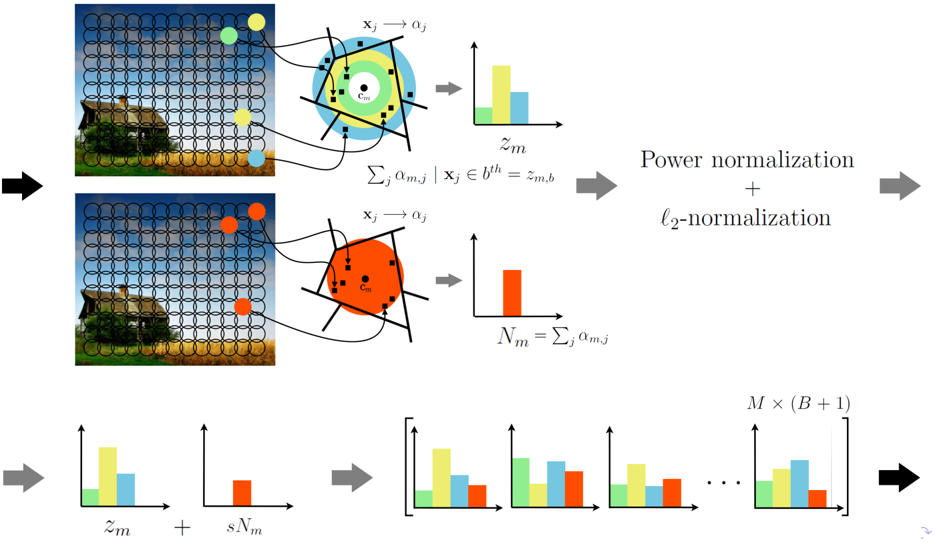
Pooling Formalism

$$
\begin{aligned}
g : \mathbb{R}^N &\longrightarrow \mathbb{R}^B \\
\alpha_{\mathbf{m}} &\longrightarrow g(\alpha_m) = z_m \\
z_{m,b} &= \text{card}\left(\mathbf{x}_j \mid \alpha_{m,j} \in \left[\frac{b}{B}; \frac{b+1}{B}\right]\right) \\
&\quad \frac{b}{B} \geq \alpha_m^{min} \text{ and } \frac{b+1}{B} \leq \alpha_m^{max}
\end{aligned}
$$

B : number of bins of each histogram z_m, and $[\alpha_m^{min}; \alpha_m^{max}]$ distance range



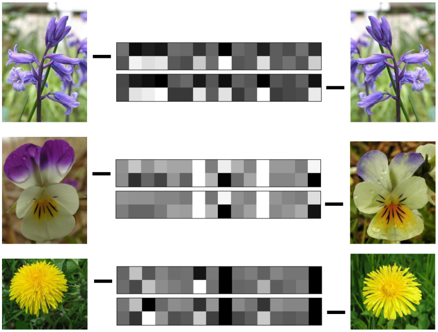
BossaNova Representation

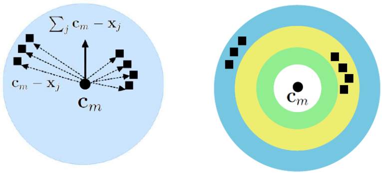


BossaNova Representation

BossaNova (BN) Parameters

- **B** (number of bins): {2, 4, 6, 8, 10}
- α_{min}: {0, 0.6}
- α_{max}: {1.5, 2.0}
- **s** (cross weight): $\{10^{-4};1\}$
- **M** (codebook): {128; 8192}





Fisher Vector / BossaNova

Experimental Results

- Implemented methods: Bag-of-Words (BoW), Fisher Vector (FV), BOSSA, BossaNova (BN), BN + FV
- Datasets: PASCAL VOC 2007, 15-Scenes, MIRFLICKR, ImageCLEF 2011
- MIRFLICKR: 25000 images, manually annotated for 38 concepts.
- ImageCLEF 2011 Photo Annotation: 18000 images, 99 concepts

Experimental Results – PASCAL VOC 2007

	MAP (%)
Implemented methods	
BoW [4]	53.2
BOSSA [5]	54.4
FV [6]	59.5
BN (ours)	58.5
BN + FV (ours)	61.6
Published results	
Krapac et al. [7]	56.7
Wang et al. [8]	59.3
Chatfield et al. [9]	**61.7**

Experimental Results – 15-Scenes

	Accuracy (%)
Implemented methods	
BoW [4]	81.1 ± 0.6
BOSSA [5]	82.9 ± 0.5
FV [6]	88.1 ± 0.2
BN (ours)	85.3 ± 0.4
BN + FV (ours)	$\mathbf{88.9} \pm 0.3$
Published results	
Yang et al. [10]	80.3 ± 0.9
Lazebnik et al. [11]	81.4 ± 0.5
Boureau et al. [12]	85.6 ± 0.2
Krapac et al. [7]	88.2 ± 0.6

Experimental Results – MIRFLICKR

	MAP (%)
Our methods	
BossaNova [Avila et al., 2012]	54.4
BossaNova + FV [Avila et al., 2012]	**56.0**
Implemented methods	
BoW [Sivic and Zisserman, 2003]	51.5
FV [Perronnin et al., 2010]	54.3
Published results	
[Huiskes et al., 2010]	37.5
[Guillaumin et al., 2010]	53.0

- Project Web page with codes available `https://sites.google.com/site/bossanovasite/`
- Publication: CVIU'12 *Pooling in Image Representation: the Visual Codeword Point of View*, S. Avila, N. Thome, M. Cord, E. Valle, A. araujo

Experimental Results – MIRFLICKR

	MAP (%)
Our methods	
BossaNova [Avila et al., 2012]	54.4
BossaNova + FV [Avila et al., 2012]	**56.0**
Implemented methods	
BoW [Sivic and Zisserman, 2003]	51.5
FV [Perronnin et al., 2010]	54.3
Published results	
[Huiskes et al., 2010]	37.5
[Guillaumin et al., 2010]	53.0

- Project Web page with codes available `https://sites.google.com/site/bossanovasite/`
- Publication: CVIU'12 *Pooling in Image Representation: the Visual Codeword Point of View*, S. Avila, N. Thome, M. Cord, E. Valle, A. araujo

Outline

Advanced coding: Localized Soft Coding [Liu ICCV 2011]

LSC principle

$$
\mathbf{H} \;=\; \begin{array}{c} \\ \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_m \\ \vdots \\ \mathbf{c}_M \end{array}
\begin{array}{c} \begin{array}{ccccc} \mathbf{x}_1 & & \mathbf{x}_j & & \mathbf{x}_N \end{array} \\
\begin{bmatrix} \alpha_{1,1} & \cdots & \alpha_{1,j} & \cdots & \alpha_{1,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{m,1} & \cdots & \alpha_{m,j} & \cdots & \alpha_{m,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{M,1} & \cdots & \alpha_{M,j} & \cdots & \alpha_{M,N} \end{bmatrix} \\
\Downarrow \\ f\text{: coding} \end{array} \Rightarrow g\text{: pooling}
$$

$$
\alpha_{m,j} = \frac{e^{-\beta \hat{d}(x_j, c_m)}}{\sum_{l=1}^{M} e^{-\beta \hat{d}(x_j, c_l)}} \qquad \hat{d}(x_j, c_m) = \begin{cases} d(x_j, c_m) & \text{if } c_m \in \mathcal{N}_k(x_j)^a \\ \infty & \text{otherwise} \end{cases}
$$

followed by max pooling, no normalization of the BoW, and Linear SVM

[a] $\mathcal{N}_k(x_i)$ the k-nearest neighbors

Dictionary learning and sparse coding

Principle

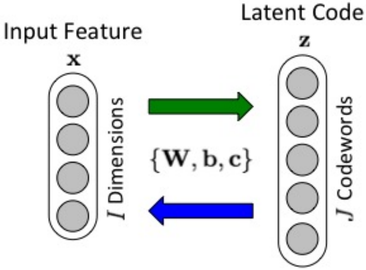
$$
\begin{array}{rl}
 & \begin{array}{ccccc} \mathbf{x}_1 & & \mathbf{x}_j & & \mathbf{x}_N \end{array} \\
\mathbf{H} \; = \begin{array}{c} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_m \\ \vdots \\ \mathbf{c}_M \end{array} & \begin{bmatrix} \alpha_{1,1} & \cdots & \alpha_{1,j} & \cdots & \alpha_{1,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{m,1} & \cdots & \alpha_{m,j} & \cdots & \alpha_{m,N} \\ \vdots & & \vdots & & \vdots \\ \alpha_{M,1} & \cdots & \alpha_{M,j} & \cdots & \alpha_{M,N} \end{bmatrix} \Rightarrow g\text{: } \textit{pooling} \\
 & \qquad\qquad \Downarrow \\
 & \qquad\quad f\text{: } \textit{coding}
\end{array}
$$

- Sparse coding (with matrix C of codewords) for local feature x_j:

$$\alpha_j = Argmin_\alpha L(\alpha, C) = ||x_j - C\alpha||_2^2 + \lambda||\alpha||_1$$

- Dictionary learning: alternate optimization over code α and matrix C over a set of local features (with constraints on vector norms)
- Discussion: one scheme for optimizing C, another for coding (most important) [Coates Ng ICML 2011]
- Sparse Auto encoders [Ranzato 2006, Bengio 2006] and RBM for dictionary learning [Hinton 2006]

Restricted Boltzmann Machine



$$P\left(z_j \mid \mathbf{x}\right) = sigm\left(b_j + \sum_{i=1}^{I} w_{ij} x_i\right)$$

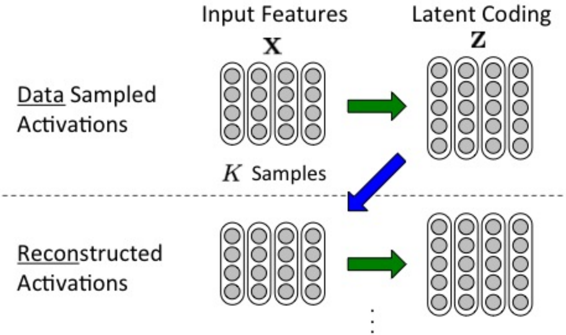
$$P\left(x_i \mid \mathbf{z}\right) = sigm\left(c_i + \sum_{j=1}^{J} w_{ij} z_j\right)$$

$$E\left(\mathbf{x}, \mathbf{z}\right) = -\log P\left(\mathbf{x}, \mathbf{z}\right) = -\sum_{i=1}^{I}\sum_{j=1}^{J} x_i w_{ij} z_j - \sum_{i=1}^{I} c_i x_i - \sum_{j=1}^{J} b_j z_j.$$

Optimization

- Maximum likelihood approximation
- Contrastive divergence learning algorithm

Contrastive Divergence Learning



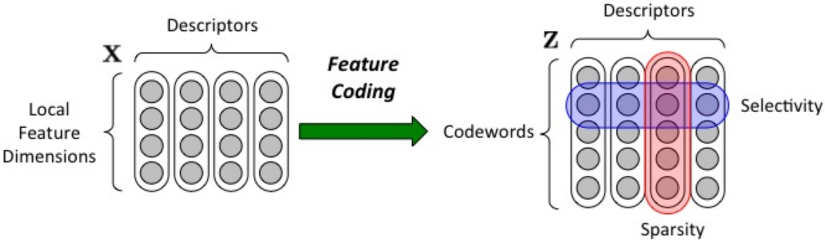
(1) Alternating Gibbs Sampling

- Fix activations of one layer
- Stochastically sample opposite layer

(2) Parameter Updates

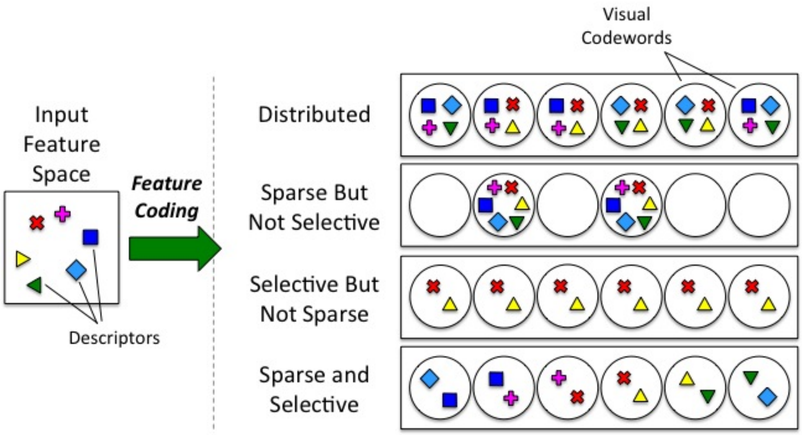
$$\Delta w_{ij} = \varepsilon \left(\langle x_i z_j \rangle_{data} - \langle x_i z_j \rangle_{recon} \right)$$
$$\Delta b_j = \varepsilon \left(\langle z_j \rangle_{data} - \langle z_j \rangle_{recon} \right)$$
$$\Delta c_i = \varepsilon \left(\langle x_i \rangle_{data} - \langle x_i \rangle_{recon} \right)$$

Selective & Sparse Feature Coding



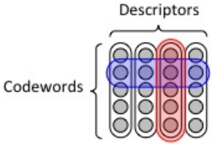
- Selectivity
  - Each codeword should respond only to a small subset of input descriptors
- Sparsity
  - Each input descriptor should only have a small subset of codewords responding to it

Sparse & Selective Coding Schemes

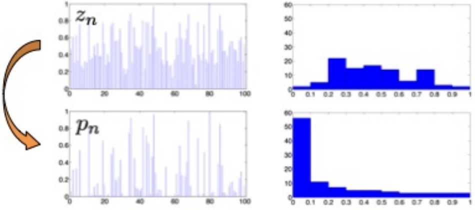


Two-Step Joint RBM Regularization: Hanlin Goh ECCV 2012

STEP 1: Compute Target Codeword Responses



STEP 1A: Map every column
STEP 1B: Remap every row



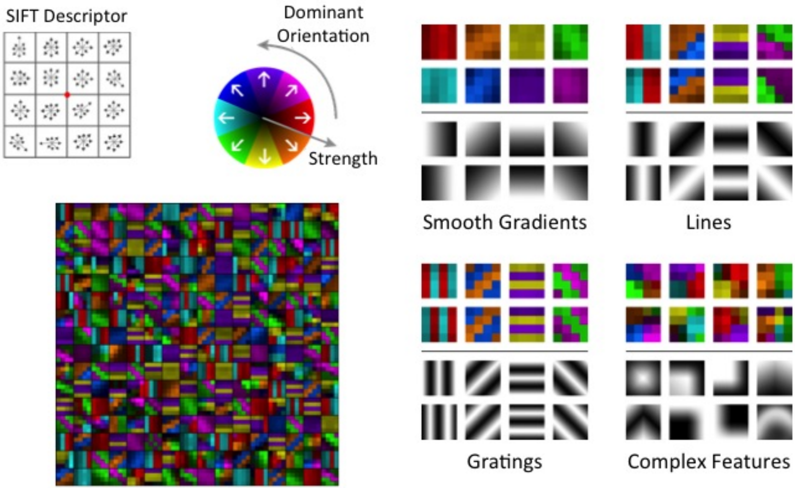
STEP 2: Regularize RBM Learning

$$\arg\min_{\mathbf{W}} \; -\sum_{k=1}^{K} \log \sum_{\mathbf{z}} \Pr(\mathbf{x}_k, \mathbf{z}_k) - \lambda \sum_{j=1}^{J} \sum_{k=1}^{K} p_{jk} \log z_{jk} + (1 - p_{jk}) \log (1 - z_{jk})$$

RBM
(maximum likelihood approximation)

Cross-Entropy Penalty
(per descriptor & codeword)

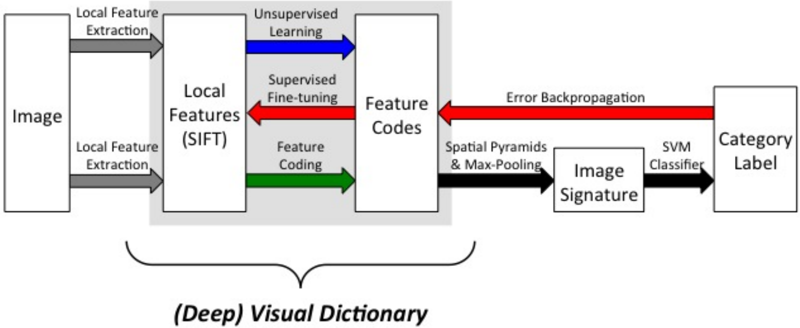
SIFT Visual Codewords



ANR ASAP

Extended Coding: dictionary learning

- Bag of Words (BOW) Model



Shallow & Deep Visual Dictionaries

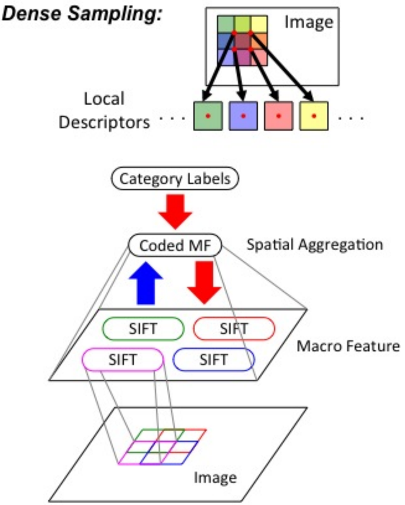


Image Categorization Results

- Achieved high accuracy
- Visual dictionaries are small and concise
- Inference is fast

Architecture	Caltech-101 (30 tr.)	Caltech-256 (60 tr.)	15-Scenes (100 tr.)
Unsupervised Shallow	78.0%	46.1%	85.7%
Supervised Shallow	78.9%	46.0%	86.0%
Unsupervised Deep	72.8%	44.7%	82.5%
Supervised Deep	79.7%	47.2%	86.4%
Bach's Team	*75.7%*	-	*84.3%*

- Deep unsupervised does not do as well as shallow unsupervised.
- Supervision is crucial for deep architectures; less important for shallow architectures.

Deep Transfer Learning

- Learn dictionary from Caltech-101 and evaluate on Caltech-256

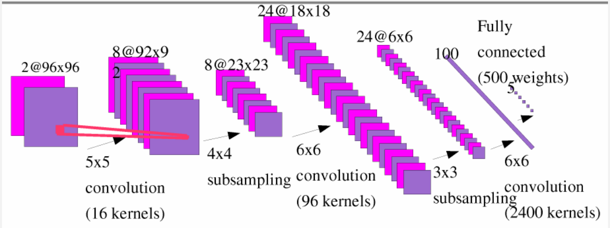
	Architecture	Layer 1 Training Set	Layer 2 Training Set	Unsupervised Results	Supervised Results
Standard Setup	Shallow	Caltech-256	-	46.1%	46.0%
	Deep	Caltech-256	Caltech-256	44.7%	47.2%
Transfer Learning	Shallow	Caltech-101	-	45.8%	45.9%
	Deep	Caltech-101	Caltech-101	41.4%	44.2%
	Deep	Caltech-101	Caltech-256	44.0%	47.0%

- Transfer learning works well on the lower layer
- Performance drops when transferring the higher layer

What's next ?

Comeback of Deep Networks

- Convolutional networks: [LeCun98], improvements [Jarrett09, Lee09]



- Deep Convolutional Neural Networks [Krizhevsky, A., Sutskever, I. and Hinton, G. E. NIPS 2012] for large dataset: ImageNet challenge winner

[*Jarrett*09]K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun.What is the best multi-stage architecture for object recognition? In Proc ICCV2009.
[*Lee*09]H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. ICML2009

People LIP6, Univ. UPMC-PARIS VI

Matthieu Cord, Nicolas Thome, matthieu.cord@lip6.fr

- PhD students: Sandra Avila, Hanlin Goh, Mar Law, Denis Pitzalis
- Post-Docs: Christian Theriault
- Research Inge. J. Guyomard

BossaNova Project Web page with codes available:
`https://sites.google.com/site/bossanovasite/`
JKernelMachines (Java) with D. Picard:
`https://mloss.org/software/view/409/`

http://webia.lip6.fr/~cord/

References I

- K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in the details: an evaluation of recent feature encoding methods, in: BMVC, 2011.
- J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: CVPR, 2010.
- S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: CVPR, 2006.
- J. Sivic, A. Zisserman, Video Google: A text retrieval approach to object matching in videos, in: ICCV, Vol. 2, 2003.
- S. Avila, N. Thome, M. Cord, E. Valle, A. Araújo, BOSSA: extended BoW formalism for image classification, in: ICIP, 2011.
- F. Perronnin, J. Sánchez, T. Mensink, Improving the Fisher Kernel for Large-Scale Image Classification, in: ECCV, 2010.
- J. Krapac, J. Verbeeky, F. Jurie, Modeling spatial layout with fisher vectors for image categorization, in: ICCV, 2011.
- J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: CVPR, 2010.
- K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in the details: an evaluation of recent feature encoding methods, in: BMVC, 2011.
- J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: CVPR, 2009.

References II

S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: CVPR, 2006.

Y. Boureau, F. Bach, Y. LeCun, J. Ponce, Learning mid-level features for recognition, in: CVPR, 2010.