Meta-Class Features for Large-Scale Object Categorization on a Budget

Speaker: Svebor Karaman

Authors:
Alessandro Bergamo   Lorenzo Torresani

Work presented as a Poster @ CVPR 2012

Object categorization

Two restrictive assumptions of standard benchmarking

- Predefined and fixed set of categories
- Computational costs not much considered

Object class search by example

Accurate real-time search and recognition of arbitrary (i.e. defined on the fly) categories in gigantic image collections.

- Application example: novel object class search

- Approach: learn a *one-vs-all classifier* using the query images as positive examples, then rank the database

Meta-classes overview

Meta-classes history

- "Efficient Object Category Recognition Using Classemes" [TSF10] **Lorenzo Torresani**, Martin Szummer, and Andrew Fitzgibbon @ ECCV 2010.
- "PICODES: Learning a Compact Code for Novel-Category Recognition" [BTF11] Alessandro Bergamo, **Lorenzo Torresani**, Andrew Fitzgibbon @ NIPS 2011.

Key concept

Use the output of rather complex classifiers, learned offline, as a feature vector used as input to learn and test online linear classifiers.

Advantages

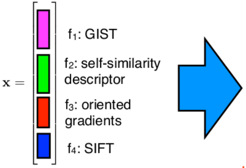
- Each bin is highly informative. Towards breaking the semantic gap?
- New classifiers can be learned efficiently online and give good performances
- Very low memory usage when binarized without much loss of performance

The meta-class features

Classification model h_c of meta-class c

- LP-β classifier [GN09]: linear combination of M non-linear (here, actually approximated by linear kernels using the "lifting" explicit map of [VZ12]) classifiers, one for each low-level feature.
- Two steps learning: parameters $\{\mathbf{w}_{m,c}, b_{m,c}\}$, then, $\beta_c = [\beta_{1,c}, \ldots, \beta_{M,c}]^T$.

- Our meta-class descriptor ***h***(***x***) $\in \mathbb{R}^C$ extracted from an image ***x***, is the output of a set of non-linear binary classifiers $\{h_1, \ldots, h_C\}$ evaluated on ***x***



$$\boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix} h_1(\mathbf{x}) \\ \vdots \\ h_C(\mathbf{x}) \end{bmatrix}$$

meta-class descriptor

Learned once in offline stage

where $h_c(\mathbf{x}) = \sum_{m=1}^{M} \beta_{m,c} \left[\mathbf{w}_{m,c}^T \Psi_m(\mathbf{f}_m(\mathbf{x})) + b_{m,c}\right]$

h_c is a variant of LP-β [12]

explicit feature map [31] to efficiently add non-linearity in the classifier

m-*th* low-level feature

- A linear classifier on this descriptor, effectively implements a non-linear function of the original low-level features:

$$g(\boldsymbol{x}) = \sum_{c=1}^{C} w_c\, h_c(\boldsymbol{x})$$

weighted-sum of the offline non-linear classifiers

Learning the meta-classes

What are meta-classes ?

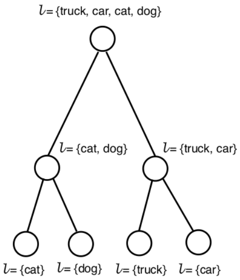
- "Abstract" categories that share common salient visual properties.

Learning the meta-classes

- Inspired by the label-tree learning [BWG10]: hierarchical partitioning of the classes into subsets
  - One meta-class should be easily recognized from the others
  - Rather high confusion within one meta-class
- Whole set of labels l_D. Given a label set l, find two subsets l^L and l^R such as: $l^L \cup l^R = l$ and $l^L \cap l^R = \emptyset$
- Given the symmetrized confusion matrix B, maximize: $E(l^L) = \sum_{i,j \in l^L} B_{ij} + \sum_{p,q \in l - l^L} B_{pq}$
- Solved by spectral clustering. Train a meta-class classifier $h_{(l^L, l^R)}(\mathbf{x})$.
- Applied recursively up to $|l| = 1$.

- <u>Our approach</u>: learn the set of basis classes
  - ✳<u>Goal</u>: each basis class should capture **useful** visual properties **shared by many object classes**
  - ✳<u>Idea</u>: define each basis class as an abstract category (<u>*meta-class*</u>) corresponding to a subset of visually similar training object classes

Label tree: each node is a <u>*meta-class*</u>, i.e. a *subset* of classes easily recognizable



<u>*mc-tree*</u>:
we learn a basis-classifier for each *meta-class*

h$_3$: {truck, car} *vs* {cat, dog}
h$_2$: {cat} *vs* {dog}
h$_1$: {truck} *vs* {car}

D: database with *K* classes (label set is {1... K})

We learn 1-vs-All classifiers over *D* (<u>*mc-classemes*</u>)

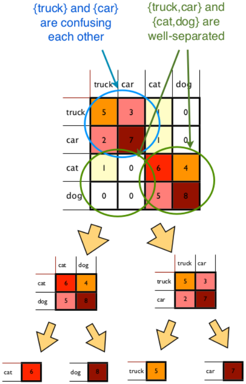
$$\{\hat{h}_1, \ldots, \hat{h}_K\}$$

We create the symmetrized *Confusion Matrix* on a validation set D^{val}

$$B \in \mathbb{R}^{K \times K}$$

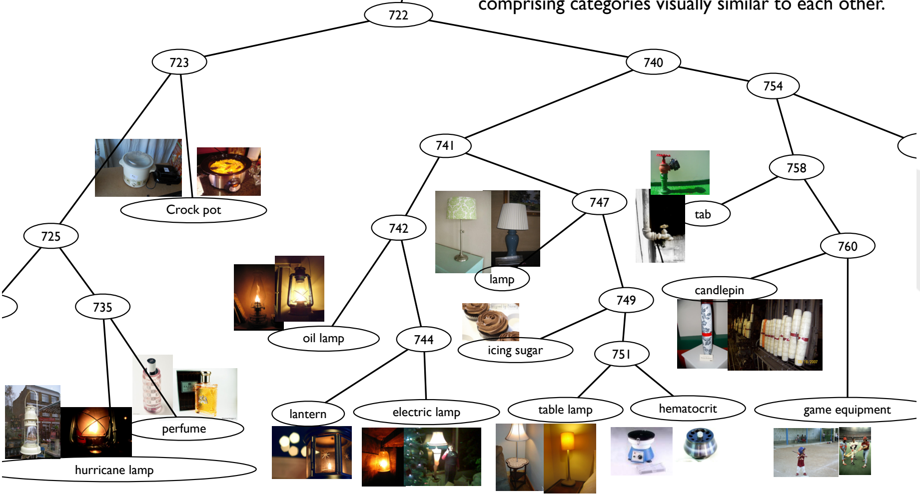
We <u>recursively partition the label set</u> ℓ of each node into two subsets by minimizing (Spectral Clustering [23]):

$$E(\ell^L) = \sum_{i,j \in \ell^L} B_{ij} + \sum_{p,q \in \ell - \ell^L} B_{pq}$$



meta-class tree visualization

We show a small portion of the tree learned by our approach. In the inner nodes, we have meta-classes comprising categories visually similar to each other.



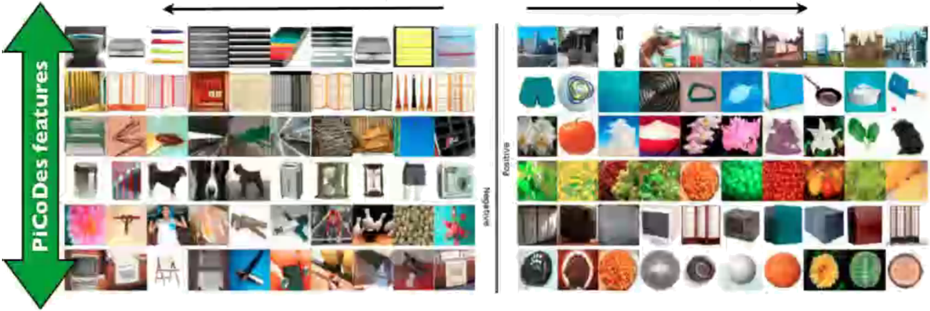
Friday, June 15, 2012

Visualization of PiCoDes

- What kind of information is encoded in PiCoDes?

images with large negative PiCoDes values (before binarization)

images with large positive PiCoDes values (before binarization)



Experiments

- Low-level features: color GIST, oriented and unoriented HOG, SSIM, SIFT + Spatial pyramids of 13 cells
- "Lifting" up the feature dimensionality from $d = 17360$ to $D = 52080$ to approximate the non-linear kernels
- 8000 randomly sampled synsets (classes) from ImageNet *disjoint* from ILSVRC2010 and Caltech256. Label-tree: 7458 internal nodes (meta-classes)
- Feature vector of 15458 dimensions: classemes + meta-classes classifiers
- Probabilistic output by sigmoid normalization through Platt's scaling: $\mathbf{mc}$
- Binarized version $[h_c(x) > 0]$: $\mathbf{mc-bit}$

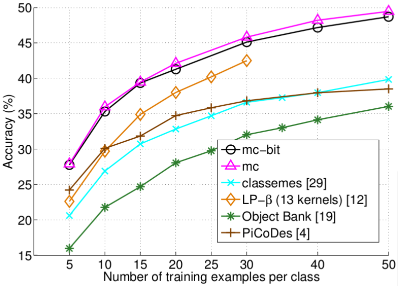


Figure 1: Multiclass object categorization accuracy on Caltech256. Linear SVM applied to mc or mc-bit descriptor outperforms the state-of-the-art LP-β classifier and is orders of magnitude faster to train and test.

Recursive feature elimination

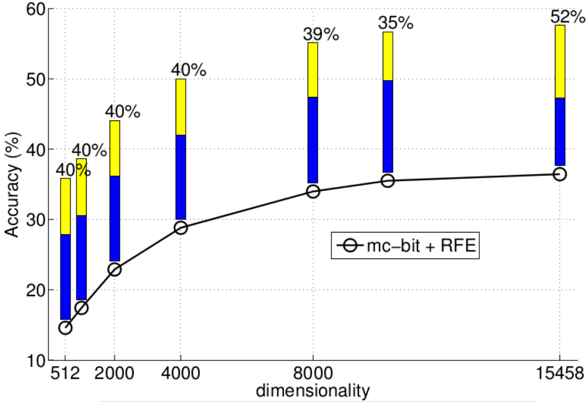
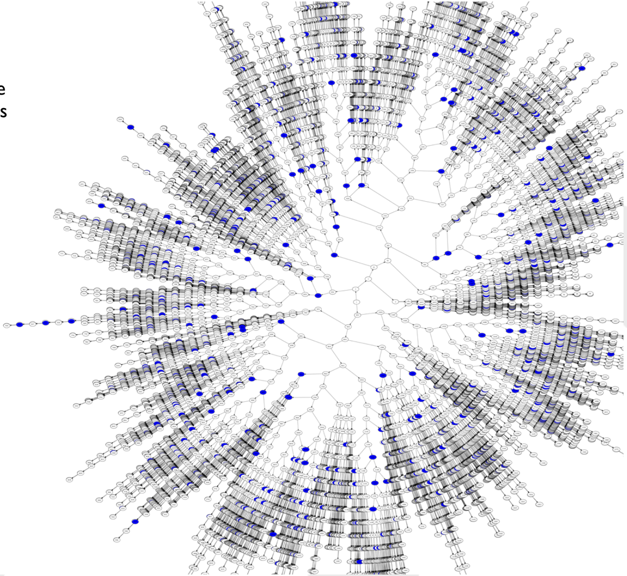


Figure 2: Multiclass recognition accuracy as a function of mc-bit dimensionality on ILSVRC2010. Recursive Feature Elimination to reduce the dimensionality of the mc-bit descriptor. The percentage at each dimensionality indicates the proportion of classeme features retained in the descriptor. Although initially the full descriptor contains more classemes than meta-classes, the majority of features selected at each step are meta-classes.

Recursive feature elimination

mc-bit: tree usage

In this slide we show the label tree learned by our method. The root is at the center of this radial layout; the radial distance of each node indicates its depth in the tree. The blue nodes are the meta-classes selected by Recursive Feature Elimination when the descriptor dimensionality is 2000.



Friday, June 15, 2012

Large scale experiment

- ILSVRC2010 dataset
  - 150000 images
  - 1000 classes
- Training protocol
  - All 150 positives
  - 4995 negatives: 5 images of the 999 other classes
- Ranking of the full dataset for each class
- Very good results, higher than classemes
- Results lower than XRCE and NEC but these methods have much higher memory usage (Fig. 4) and computational costs (Fig. 5)

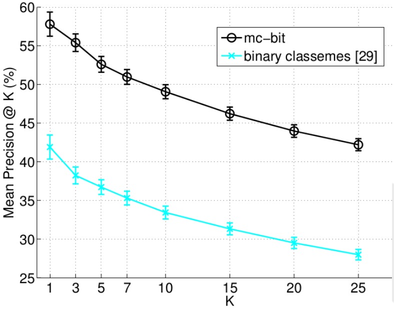


Figure 3: Object-class search on ILSVRC2010: accuracy in retrieving images of a novel class from a dataset of 150,000 photos. For each query class, the true positives are only 0.1% of the database. The mc-bit descriptor significantly outperforms classemes.

Computational costs discussion

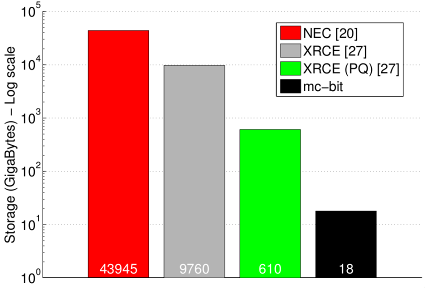


Figure 4: Storage requirements for 10M images with different feature representations (note the log scale). The mc-bit descriptors outperforms the top-method for ILSVRC2010 in terms of scalability to large databases.

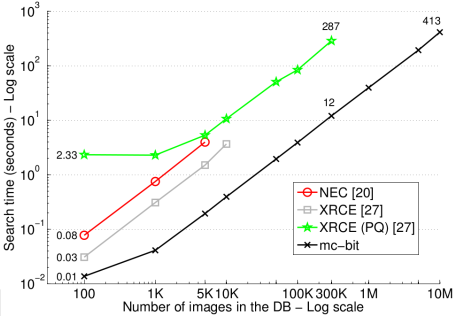


Figure 5: Object-class search time as a function of the number of images in the database, using a machine with 20 GB of memory. The mc-bit approach is significantly faster than the competing approaches and is the only one that allows large databases to be kept in the memory of standard computers.

Implementation

Code available

- The software supports several images types (Jpeg, Png, Tiff, and others) and it is available for Microsoft Windows, GNU/Linux and Mac OSX.
- They exploit explicit features maps approximating the intersection kernel to efficiently evaluate the non-linear kernels used by classemes, resulting in a descriptor extraction that takes about 2 sec per image.

http://vlg.cs.dartmouth.edu/projects/vlg_extractor/vlg_extractor/Home.html

References I

[BTF11] Alessandro Bergamo, Lorenzo Torresani, and Andrew Fitzgibbon, *Picodes: Learning a compact code for novel-category recognition*, Advances in Neural Information Processing Systems 24 (J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, eds.), 2011, pp. 2088–2096.

[BWG10] Samy Bengio, Jason Weston, and David Grangier, *Label embedding trees for large multi-class tasks*, Advances in Neural Information Processing Systems **23** (2010), no. 163-171, 3.

[GN09] Peter Gehler and Sebastian Nowozin, *On feature combination for multiclass object classification*, Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 221–228.

[TSF10] Lorenzo Torresani, Martin Szummer, and Andrew Fitzgibbon, *Efficient object category recognition using classemes*, European Conference on Computer Vision (ECCV), September 2010, pp. 776–789.

[VZ12] Andrea Vedaldi and Andrew Zisserman, *Efficient additive kernels via explicit feature maps*, Pattern Analysis and Machine Intelligence, IEEE Transactions on **34** (2012), no. 3, 480–492.

Meta-Class Features for Large-Scale Object Categorization on a Budget

Speaker: Svebor Karaman

Authors:
Alessandro Bergamo Lorenzo Torresani

Work presented as a Poster @ CVPR 2012