Learning Representations for Visual Object Class Recognition

Marcin Marszałek Cordelia Schmid
Hedi Harzallah Joost van de Weijer

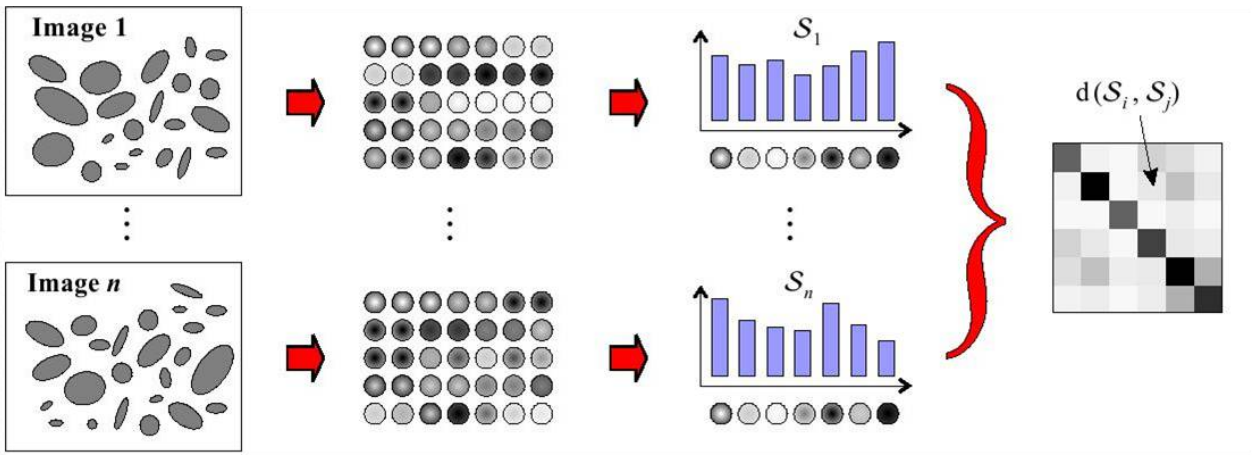
LEAR, INRIA Grenoble, Rhône-Alpes, France

October 15th, 2007

Bag-of-Features

Zhang, Marszałek, Lazebnik and Schmid [IJCV'07]

- Bag-of-Features (BoF) is an orderless distribution of local image features sampled from an image
- The representations are compared using χ^2 distance

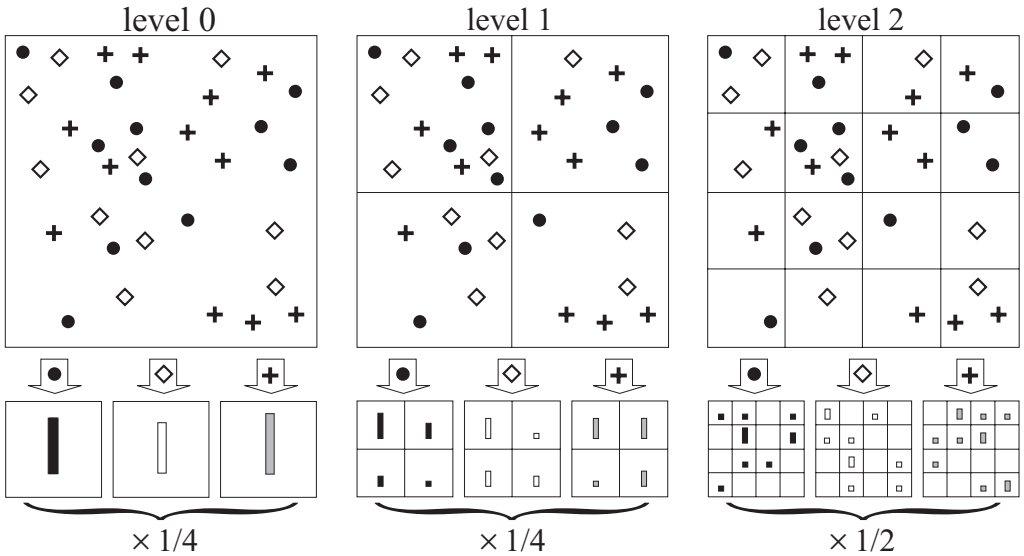


- Channels can be combined to improve the accuracy
- Classification with non-linear Support Vector Machines

Spatial pyramid

Lazebnik, Schmid and Ponce [CVPR'06]

- Spatial grids allow for locally orderless description
- They can be viewed as an extension to Bag-of-Features



- They were shown to work on scene category and object class datasets

Combining kernels

Bosch, Zisserman and Munoz [CIVR'07], Varma and Ray [ICCV'07]

- It was shown that linear kernel combinations can be learned
  - Through extensive search [Bosch'07]
  - By extending the C-SVM objective function [Varma'07]
- We learn linear distance combinations instead
  - Our approach can still be viewed as learning a kernel
  - We exploit the kernel trick (it's more than linear combination of kernels)
  - No kernel parameters are set by hand, everything is learned
  - Optimization task is more difficult

Our approach: large number of channels

- In our approach images are represented with several BoFs, where each BoF is assigned to a cell of a spatial grid
- We combine various methods for sampling the image, describing the local content and organizing BoFs spatially
- With few samplers, descriptors and spatial grids we can generate tens of possible representations that we call “channels”
- Useful channels can be found on per-class basis by running a multi-goal genetic algorithm

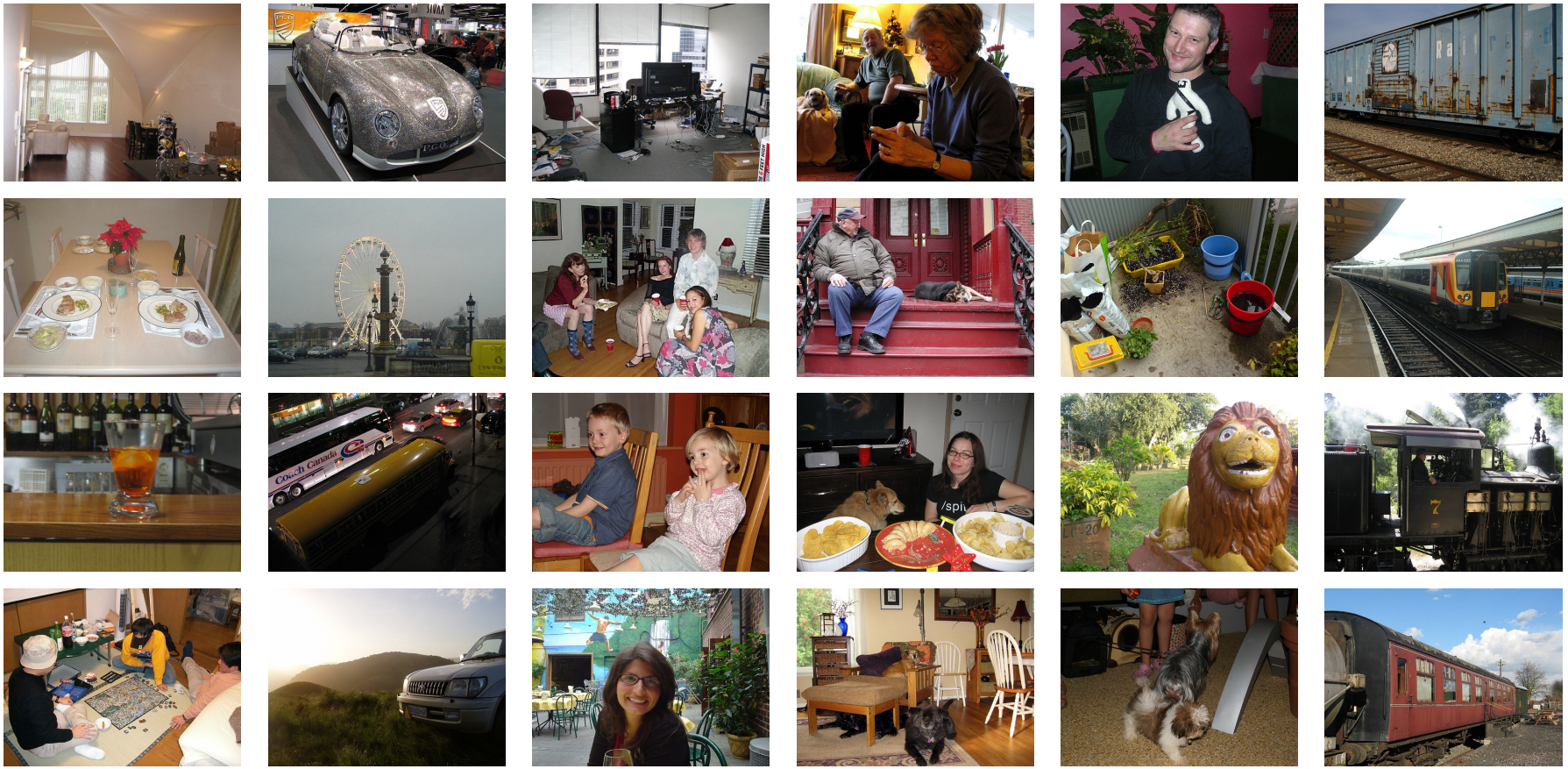
Overview of the processing chain

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- Image is sampled
- Regions are locally described with feature vectors
- Features are quantized (assigned to a vocabulary word) and spatially ordered (assigned to a grid cell)
- Various channels are combined in the kernel
- Image is classified with an SVM

PASCAL VOC 2007 challenge

Image → Sampler × Local descriptor × Spatial grid ⇒ Fusion → Classification



bottle car chair dog plant train

Image sampling

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

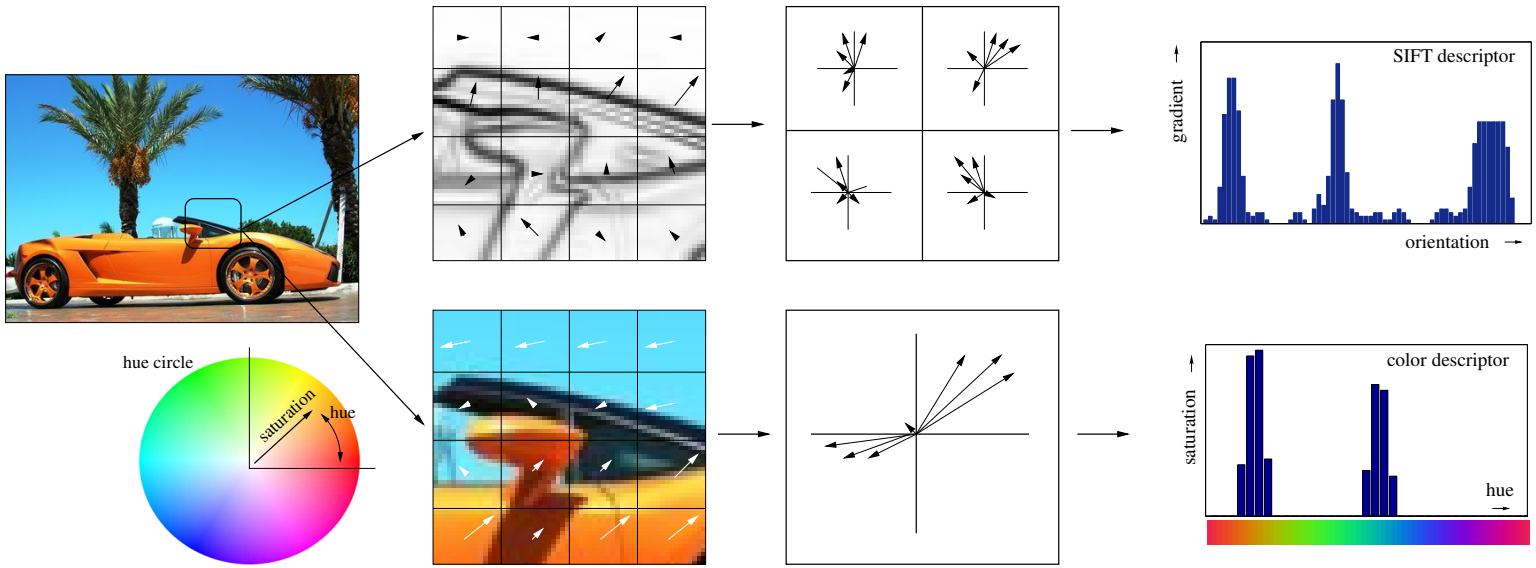
- Interest points detectors
  - Harris-Laplace — detects corners [Mikołajczyk'04]
  - Laplacian — detects blobs [Lindeberg'98]

- Dense sampling
  - Multiscale grid with horizontal/vertical step of 6 pixels (half of the SIFT support area width/height) and scaling factor of 1.2 per scale-level

Local description

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- SIFT — gradient orientation histogram [Lowe'04]
- SIFT+hue — SIFT with color [van de Weijer'06]

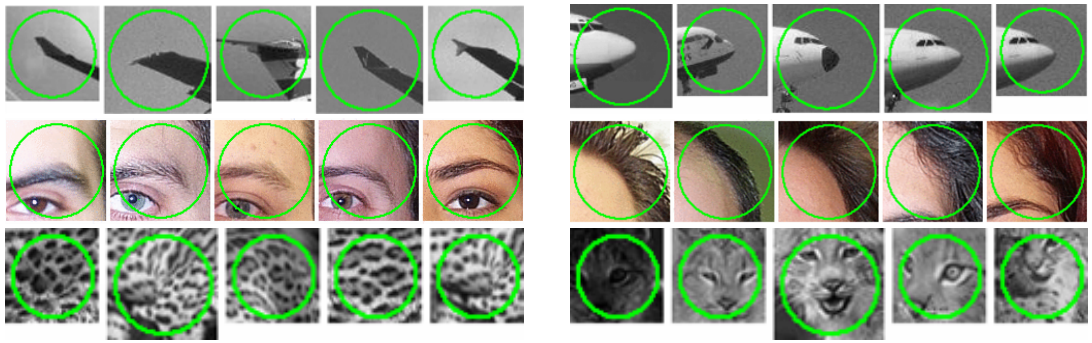


- PAS — edgel histogram [Ferrari'06]

Spatial organization

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- Visual vocabulary is created by clustering the features using k-means ($k = 4000$)



- Spatial grids allow to separately describe the properties of roughly defined image regions
  - 1x1 — standard Bag-of-Features
  - 2x2 — defines four image quarters
  - horizontal 3x1 — defines upper, middle and lower regions

Support Vector Machines

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- We use non-linear Support Vector Machines
- The decision function has the following form

$$g(x) = \sum_i \alpha_i y_i K(x_i, x) - b$$

- We propose a multichannel extended Gaussian kernel

$$K(x_j, x_k) = \exp\Big(- \sum_{ch} \gamma_{ch} D_{ch}(x_j, x_k) \Big)$$

- $D_{ch}(x_j, x_k)$ is a similarity measure (χ^2 distance in our setup) for channel *ch*

Support Vector Machines

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- We use non-linear Support Vector Machines
- The decision function has the following form

$$g(x) = \sum_i \alpha_i y_i K(x_i, x) - b$$

- We propose a multichannel extended Gaussian kernel

$$K(x_j, x_k) = \exp\Big(- \sum_{ch} \gamma_{ch} D_{ch}(x_j, x_k) \Big)$$

- $D_{ch}(x_j, x_k)$ is a similarity measure (χ^2 distance in our setup) for channel *ch*
- Problem: How to set each γ_{ch}?

Weighting the channels

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- If we set γ_{ch} to $1/\overline{D_{ch}}$ we almost obtain (up to channels normalization) the method of Zhang et al.
  - This approach demonstrated remarkable performance in both VOC'05 and VOC'06
  - We submit this approach as the "flat" method
- As γ_{ch} controls the weight of channel *ch* in the sum, it can be used to select the most useful channels
  - We run a genetic algorithm to optimize per-task $\gamma_{ch,t}$ kernel parameters and also C_t SVM parameter
  - The learned channel weights are used for the "genetic" submission

Genetic algorithm to optimize SVM parameters

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- The genoms encode the optimized parameters
- In every iteration (generation)
  1. Random genoms are added to the pool (population)
  2. Cross-validation is used to evaluate the genoms (individuals) simultaneously for each class
  3. The more useful the genom is the more chance it has to be selected and combined with another good genom
  4. Information from combined genoms is randomly mixed (crossed) and forms the next generation
  5. To better avoid local minimas, random genes are altered (mutated)
- Useful genes and gene combinations survive and multiply

Genetic algorithm to optimize SVM parameters

Image $\rightarrow$ Sampler $\times$ Local descriptor $\times$ Spatial grid $\Rightarrow$ Fusion $\rightarrow$ Classification

- The genoms encode the optimized parameters
- In every iteration (generation)
  1. Random genoms are added to the pool (population)
  2. Cross-validation is used to evaluate the genoms (individuals) simultaneously for each class
  3. The more useful the genom is the more chance it has to be selected and combined with another good genom
  4. Information from combined genoms is randomly mixed (crossed) and forms the next generation
  5. To better avoid local minimas, random genes are altered (mutated)
- Useful genes and gene combinations survive and multiply

Multiplying channels

Channels ($\gamma_{ch} = 1/\overline{D_{ch}}$)			#	Average AP
HS,LS	× SIFT	× 1,2x2	4	47.7
HS,LS,DS	× SIFT	× 1,2x2	6	52.6
HS,LS,DS	× SIFT	× 1,2x2,h3x1	9	53.3
HS,LS,DS	× SIFT,SIFT+hue	× 1,2x2,h3x1	18	54.0
HS,LS,DS	× SIFT,SIFT+hue,PAS	× 1,2x2,h3x1	21	54.2
DS	× SIFT,SIFT+hue,PAS	× 1,2x2,h3x1	9	51.8

Table: Class-averaged AP on VOC'07 validation set

- Combination of interest points and dense sampling boosts the performance, color and 3x1 grid are important
- The performance monotonically increases with the number of channels
- Last experiments show, that anything sensible (HoGs, different vocabularies) further helps the performance

PASCAL VOC'07 Challenge results

	INRIA (genetic)	INRIA (flat)	XRCE	TKK	QMUL (lspch)	QMUL (hsls)
aeroplane	0.775	0.748	0.723	0.714	0.716	0.706
bicycle	0.636	0.625	0.575	0.517	0.550	0.548
bird	0.561	0.512	0.532	0.485	0.411	0.357
boat	0.719	0.694	0.689	0.634	0.655	0.645
bottle	0.331	0.292	0.285	0.273	0.272	0.278
bus	0.606	0.604	0.575	0.499	0.511	0.511
car	0.780	0.763	0.754	0.701	0.722	0.714
cat	0.588	0.576	0.503	0.512	0.551	0.540
chair	0.535	0.531	0.522	0.517	0.474	0.466
cow	0.426	0.411	0.390	0.323	0.359	0.366
dining table	0.549	0.540	0.468	0.463	0.374	0.344
dog	0.458	0.428	0.453	0.415	0.415	0.399
horse	0.775	0.765	0.757	0.726	0.715	0.715
motorbike	0.640	0.623	0.585	0.602	0.579	0.554
person	0.859	0.845	0.840	0.822	0.808	0.806
potted plant	0.363	0.353	0.326	0.317	0.156	0.158
sheep	0.447	0.413	0.397	0.301	0.333	0.358
sofa	0.506	0.501	0.509	0.392	0.419	0.415
train	0.792	0.776	0.751	0.711	0.765	0.731
tv/monitor	0.532	0.493	0.495	0.410	0.459	0.455
average	0.594	0.575	0.556	0.517	0.512	0.503

Learning channel weights (VOC'07 results)

	HS	LS	DS	SIFT	+hue	PAS	1	2x2	h3x1	C	flat	genetic
aeroplane	0.7	1.5	2.8	0.04	0.09	0.29	5.7	3.1	4.3	897	0.748	0.775
bicycle	0.7	1.5	2.8	0.04	0.09	0.12	5.7	1.0	4.3	521	0.625	0.636
bird	0.7	1.5	4.3	0.04	0.09	0.12	5.7	1.0	4.3	141	0.512	0.561
boat	0.2	1.5	4.3	0.12	0.03	0.12	5.7	1.0	4.3	897	0.694	0.719
bottle	0.7	1.5	1.5	0.09	0.09	0.12	5.7	3.1	4.3	897	0.292	0.331
bus	0.2	1.5	1.5	0.09	0.03	0.29	1.5	7.3	4.3	6	0.604	0.606
car	0.7	1.5	4.3	0.09	0.09	0.29	5.7	0.1	4.3	521	0.763	0.780
cat	0.7	1.5	2.8	0.04	0.03	0.12	5.7	3.1	4.3	19	0.576	0.588
chair	2.5	1.5	2.8	0.09	0.09	0.29	5.7	3.1	4.3	19	0.531	0.535
cow	0.2	1.5	4.3	0.04	0.09	0.29	5.7	3.1	4.3	897	0.411	0.426
dining table	0.2	1.5	4.3	0.12	0.02	0.29	5.7	3.1	4.3	6	0.540	0.549
dog	0.2	1.5	4.3	0.12	0.09	0.07	5.7	3.1	4.3	6	0.428	0.458
horse	0.7	1.5	4.3	0.09	0.03	0.12	5.7	0.1	4.3	521	0.765	0.775
motorbike	0.7	1.5	4.3	0.04	0.03	0.12	1.5	3.1	4.3	897	0.623	0.640
person	0.2	1.5	7.9	0.12	0.09	0.29	5.7	1.0	4.3	141	0.845	0.859
potted plant	2.5	1.5	4.3	0.04	0.09	0.29	5.7	3.1	4.3	19	0.353	0.363
sheep	0.2	1.5	4.3	0.12	0.03	0.29	5.7	0.1	4.3	6	0.413	0.447
sofa	2.5	0.7	4.3	0.04	0.03	0.05	5.7	3.1	4.3	141	0.501	0.506
train	0.7	1.5	4.3	0.12	0.09	0.29	5.7	0.1	4.3	897	0.776	0.792
tv/monitor	2.5	0.7	4.3	0.04	0.03	0.12	5.7	3.1	4.3	19	0.493	0.532

Summary

- We have shown that using a large number of channels helps recognition due to complementary information
- We have demonstrated how it is possible to generate tens of useful channels
- We have proposed to use a genetic algorithm to discover the most useful channels on per-class basis
- The experimental results show excellent performance

Thank you for your attention

I will be glad to answer your questions