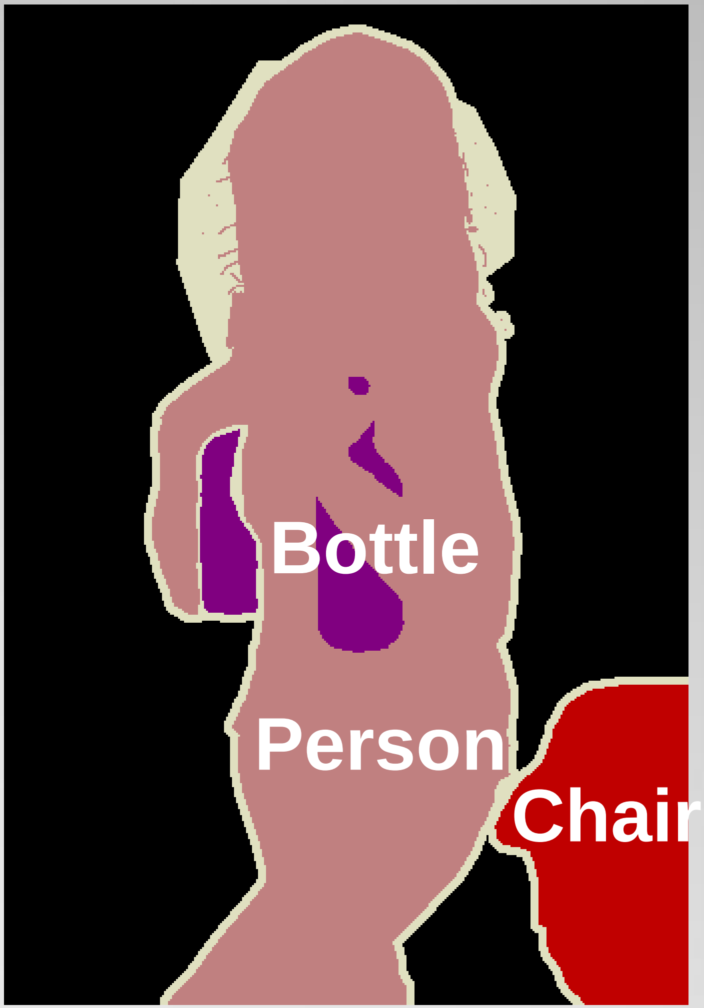
Semantic Segmentation with Second-Order Pooling

João Carreira[1,2], Rui Caseiro[1], Jorge Batista[1], Cristian Sminchisescu[2]

[1] *Institute of Systems and Robotics,* **University of Coimbra**

[2] *Faculty of Mathematics and Natural Science,* **University of Bonn**

Semantic Segmentation



Example from Pascal VOC segmentation dataset

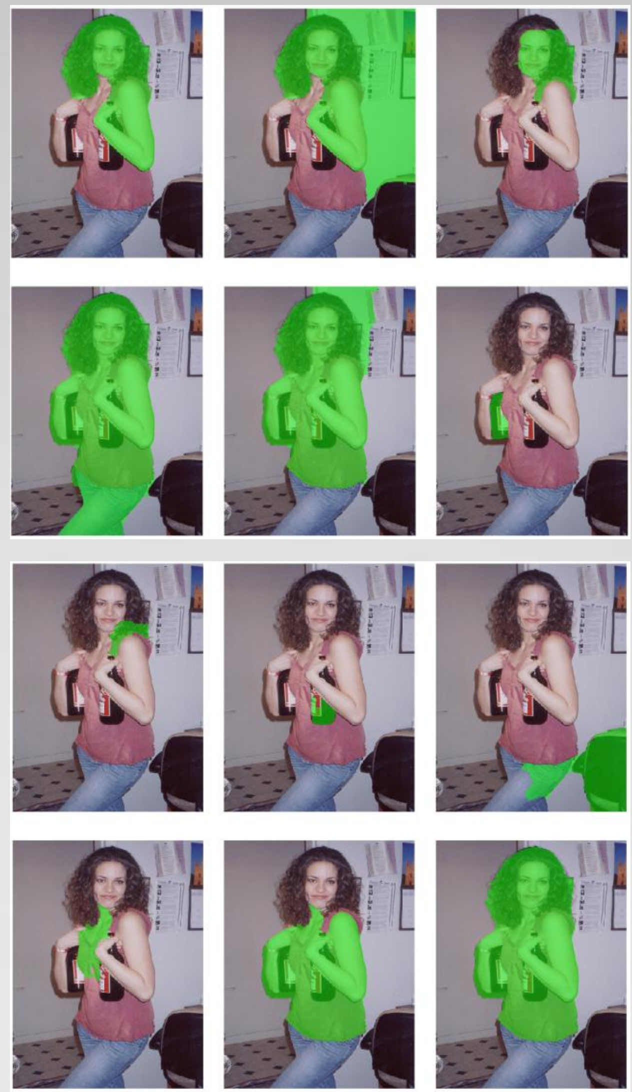
Semantic Segmentation

Our bottom-up pipeline:

Li, Carreira, Sminchisescu, CVPR 2010, IJCV2011

1. Sample candidate object regions (figure-ground)
2. Region description and classification
3. Construct full image labeling from regions

Semantic Segmentation

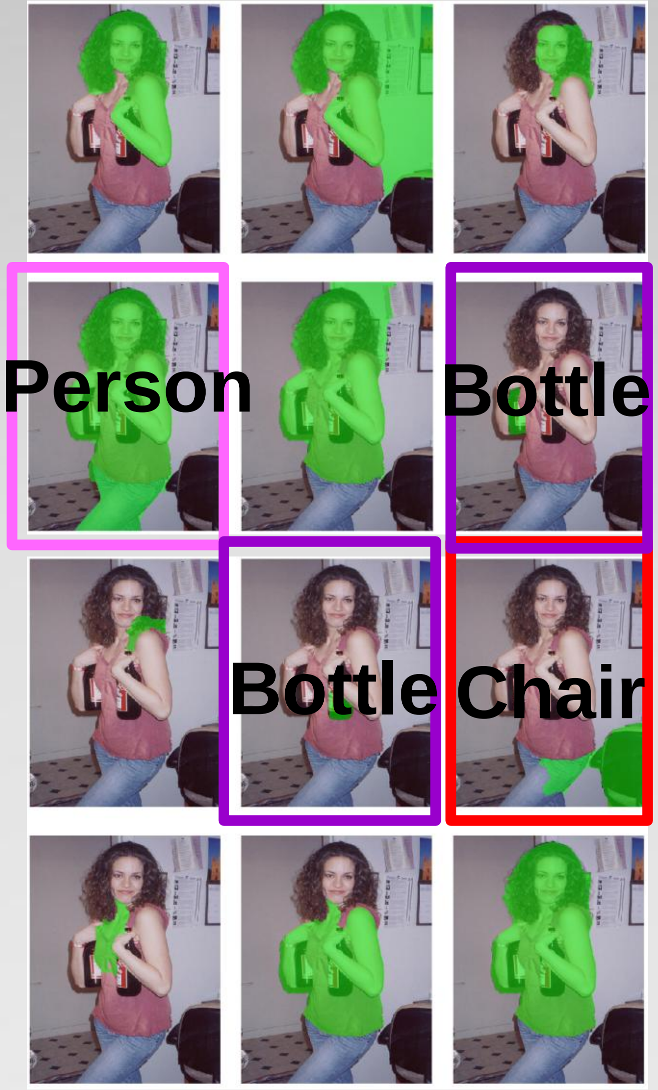


1. Sample candidate object regions (figure-ground)
2. Region description and classification
3. Construct full image labeling from regions

Key: generate good object candidates, not superpixels

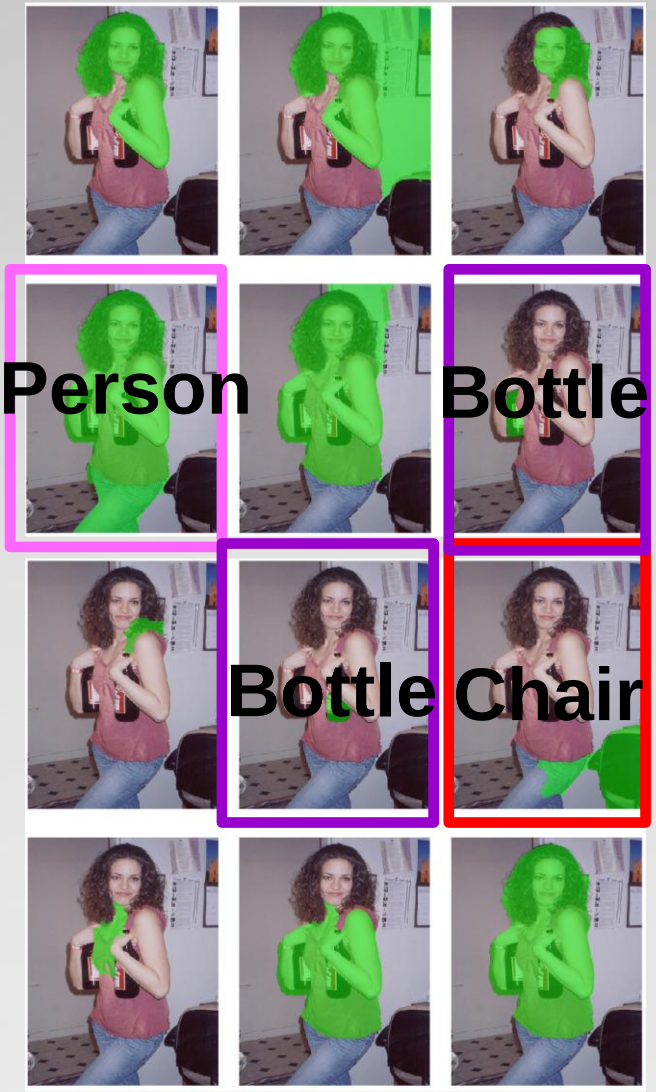
CPMC: Constrained Parametric Min-Cuts for Automatic Object Segmentation,
Carreira and Sminchisescu,
CVPR 2010, PAMI 2012

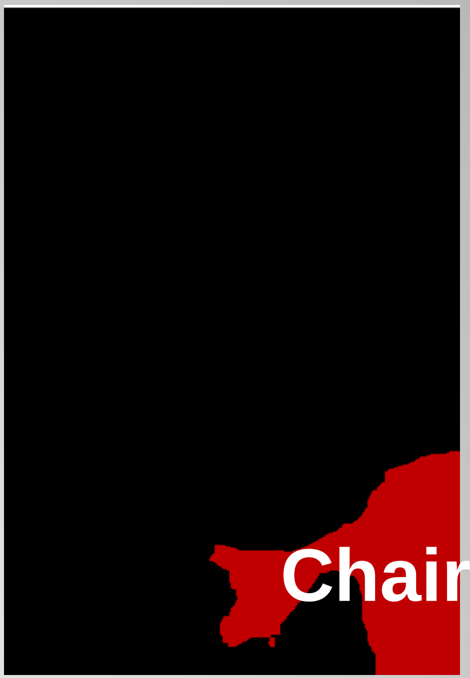
Semantic Segmentation



1. Sample candidate object regions (figure-ground)
2. Region description and classification
3. Construct full image labeling from regions

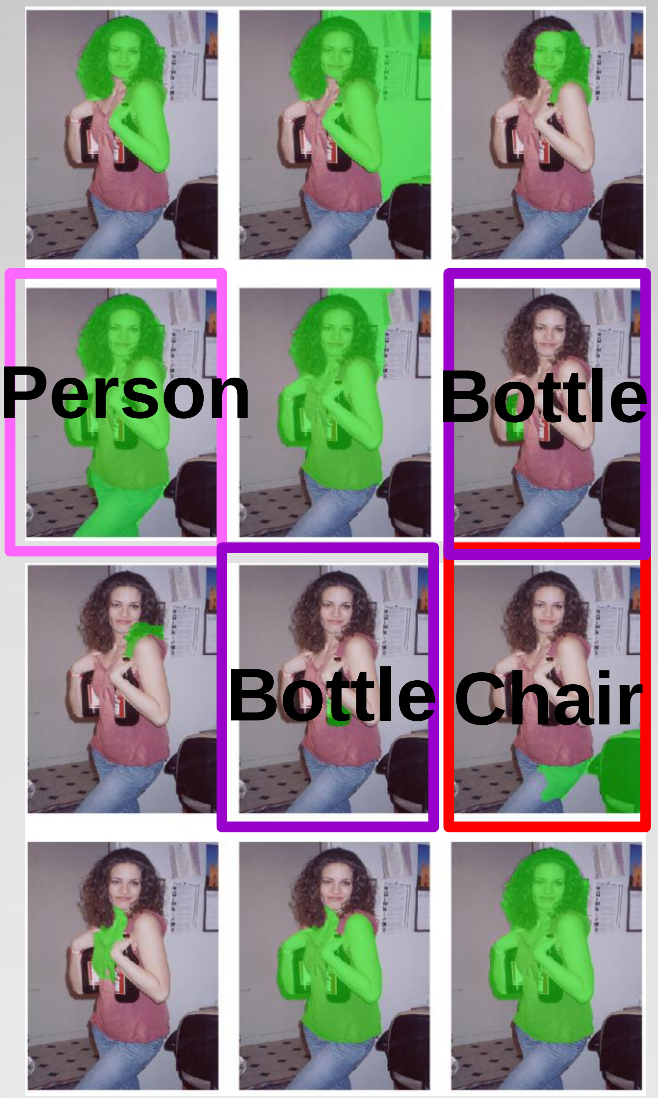
Semantic Segmentation

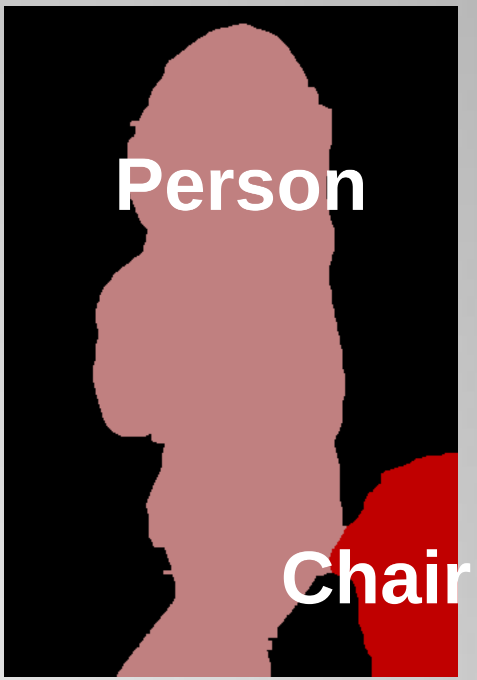




1. Sample candidate object regions (figure-ground)
2. Region description and classification
3. Construct full image labeling from regions

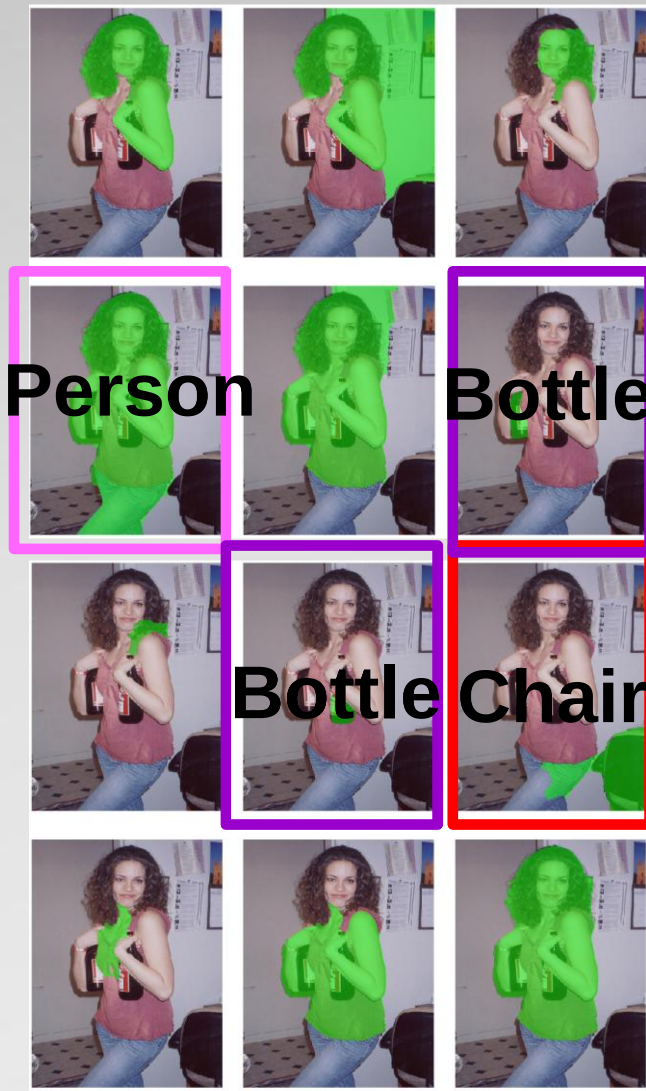
Semantic Segmentation

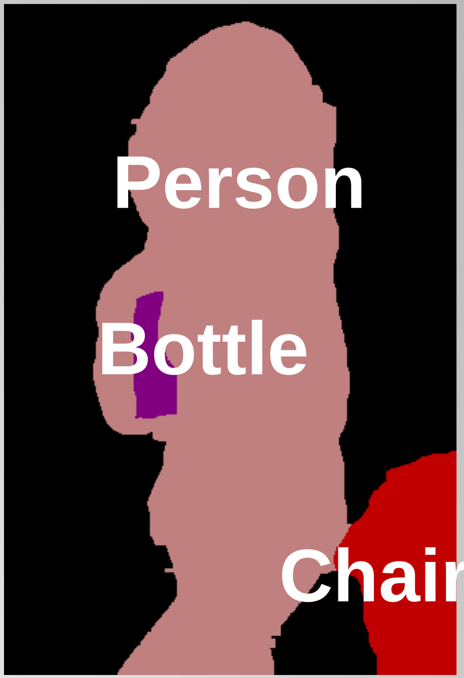




1. Sample candidate object regions (figure-ground)
2. Region description and classification
3. Construct full image labeling from regions

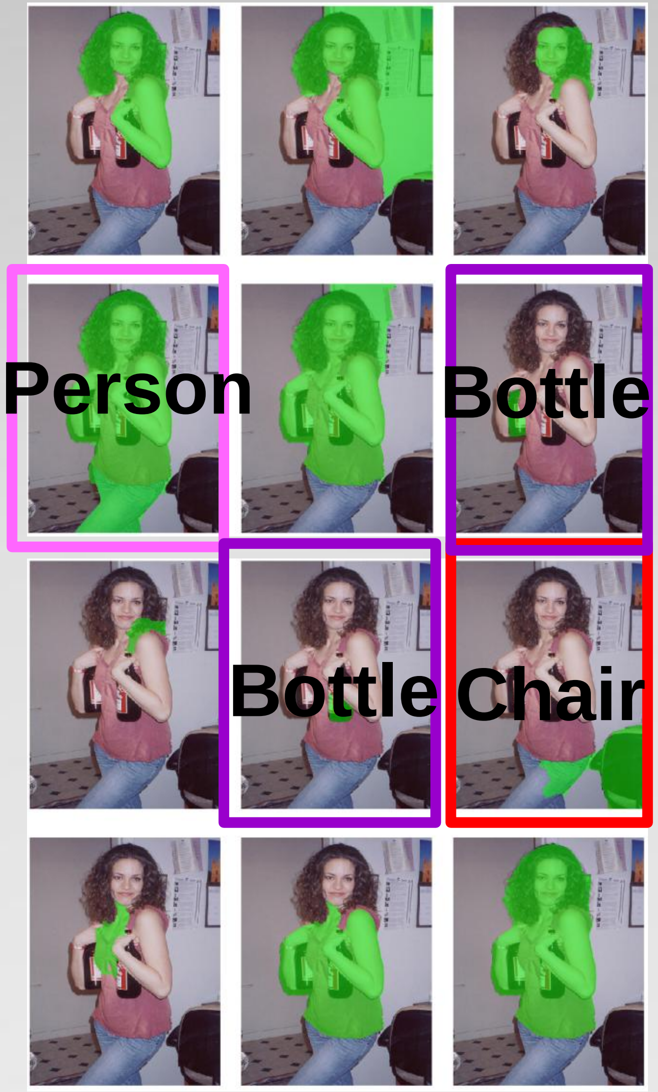
Semantic Segmentation

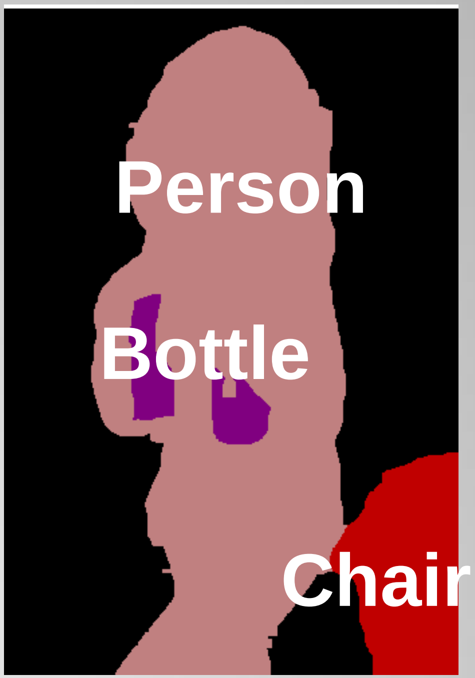




1. Sample candidate object regions (figure-ground)
2. Region description and classification
3. Construct full image labeling from regions

Semantic Segmentation





1. Sample candidate object regions
2. Region description and classification
3. Construct full image labeling from regions

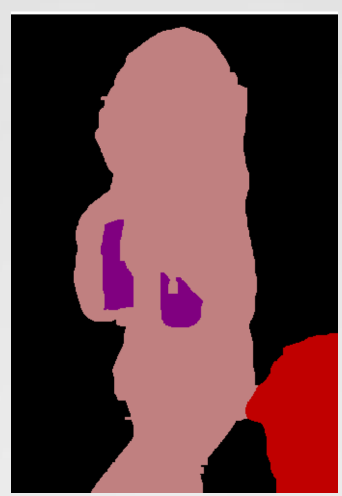
Semantic Segmentation

Bottom-up formulation:

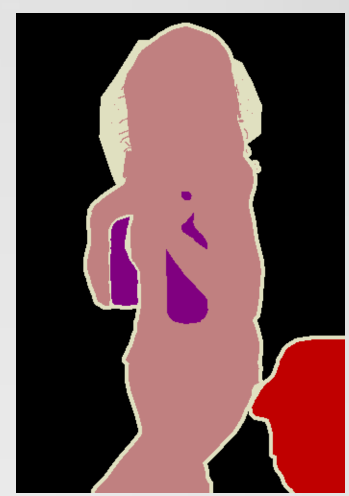
1. Sample candidate object regions (figure-ground)
2. Region description and classification
3. Construct full image labeling from regions

Image

Predicted



Ground Truth



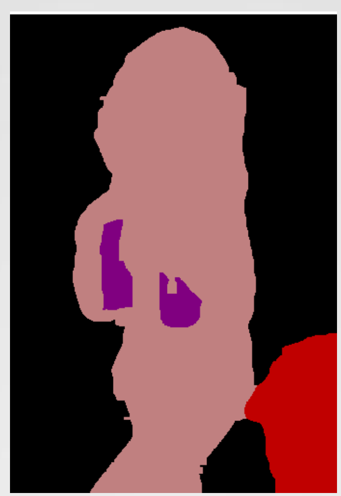
Semantic Segmentation

Bottom-up formulation:

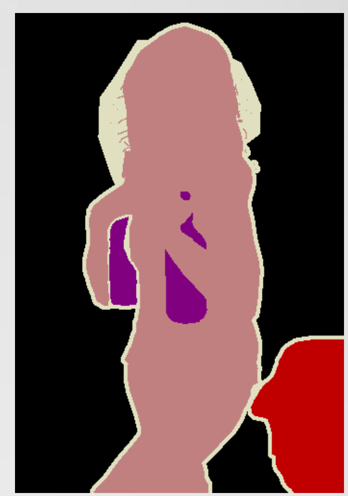
1. Sample candidate object regions (figure-ground)
2. <u>Region description and classification</u> **This work!**
3. Construct full image labeling from regions

Image

Predicted



Ground Truth



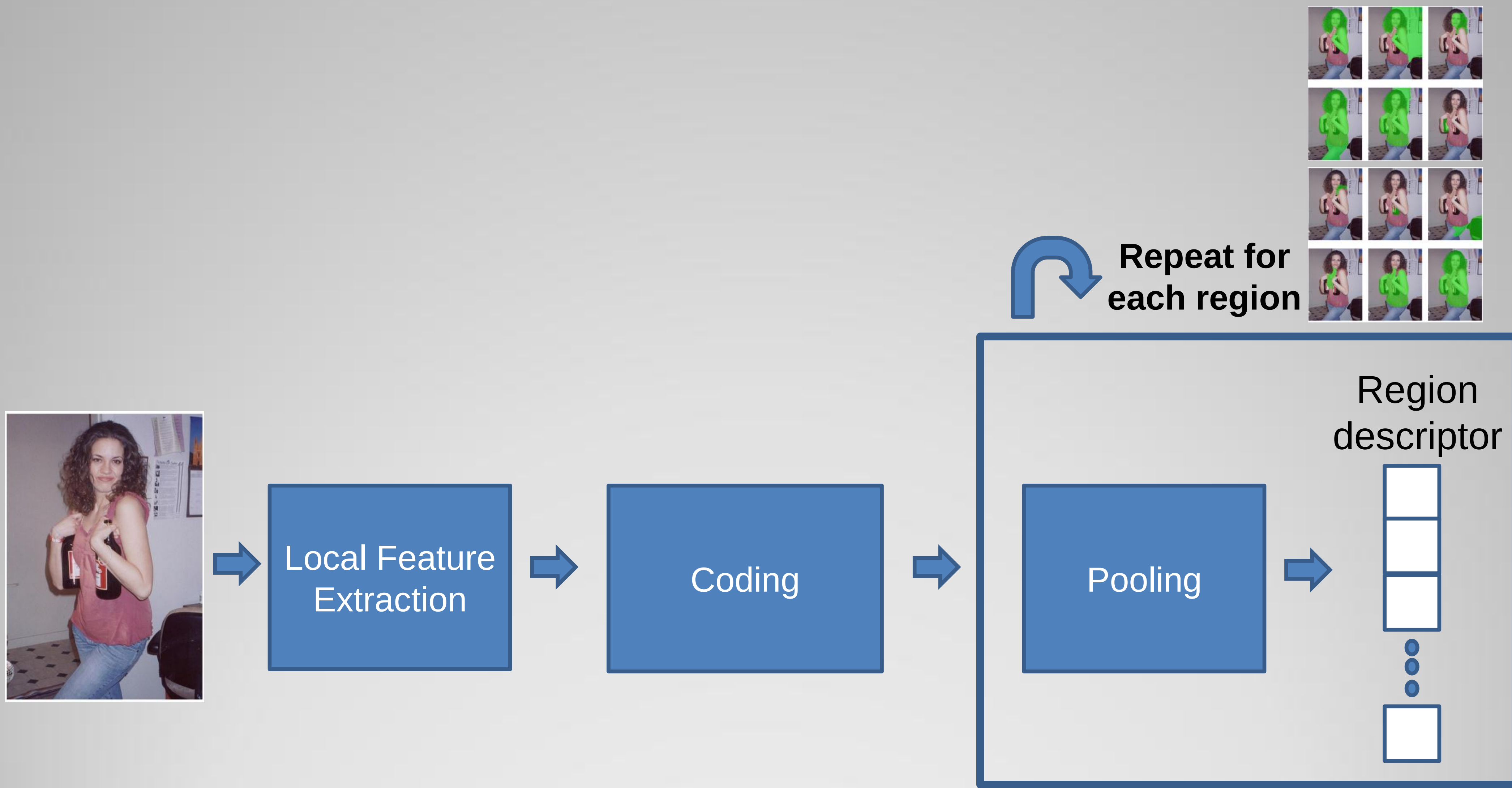
Describing Free-form Regions

Currently, most successful approaches use variations of Bag of Words (BOW) and HOG

- Require expensive classifiers with non-linear kernels
- Used in sliding-window detection and in image classification

Are there descriptors better suited for regions (segments)?

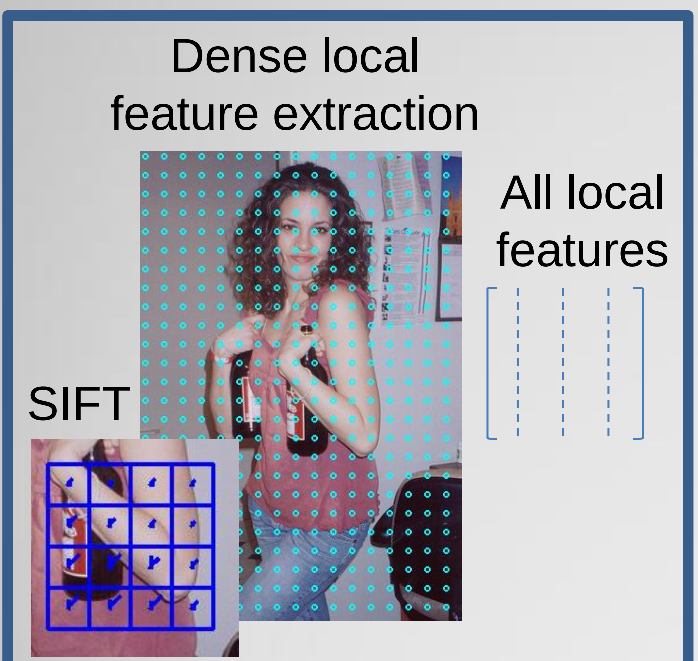
Aggregation-based Descriptors



Aggregation-based Descriptors

Local Feature Extraction → Coding → Pooling

Repeat for each region



Aggregation-based Descriptors

Local Feature Extraction → Coding → Pooling

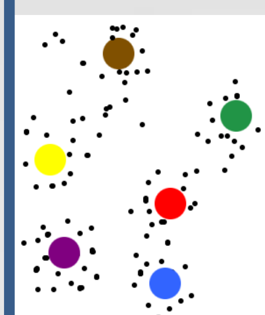
Repeat for each region

Dense local feature extraction

SIFT

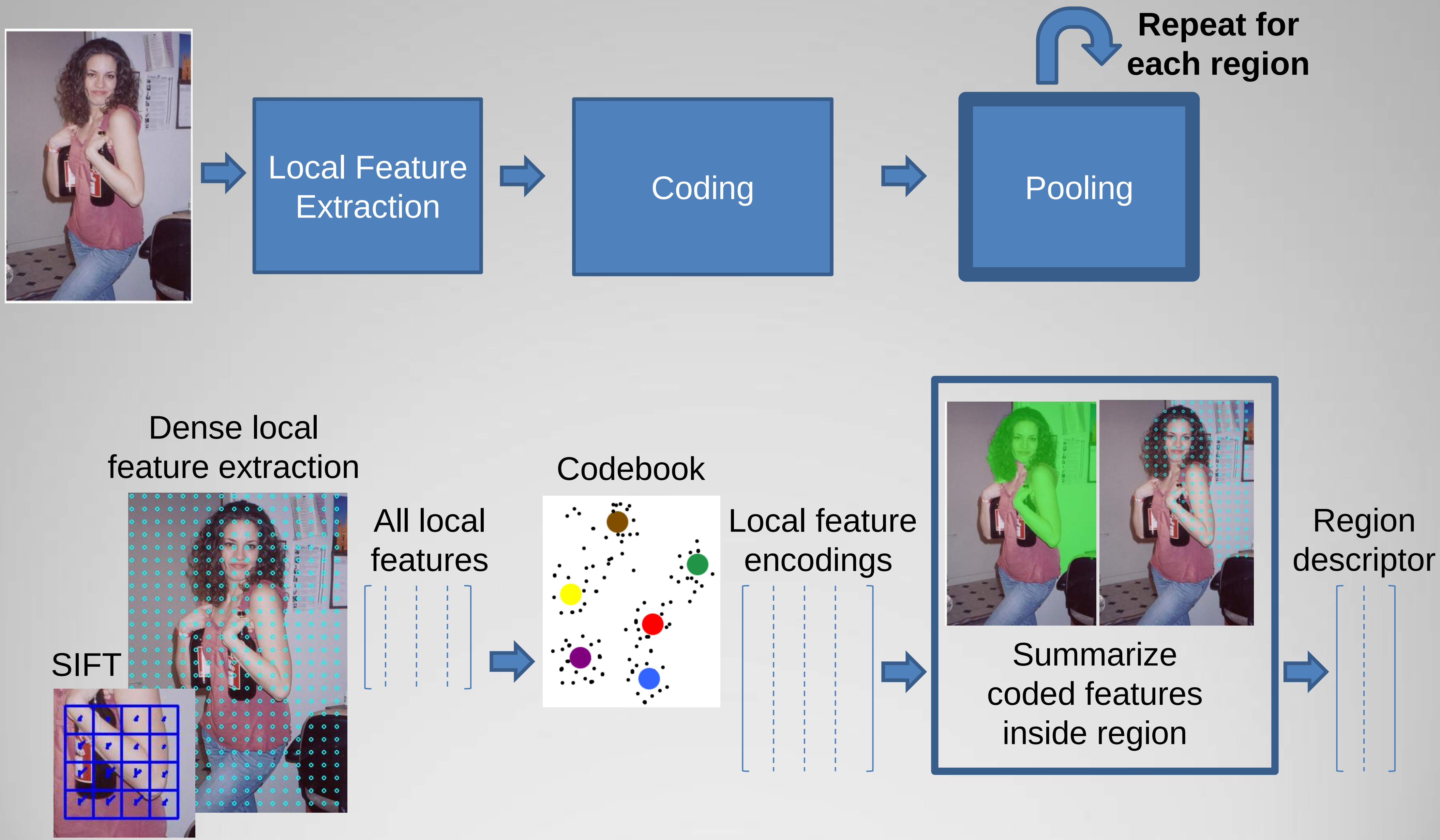
All local features

Codebook

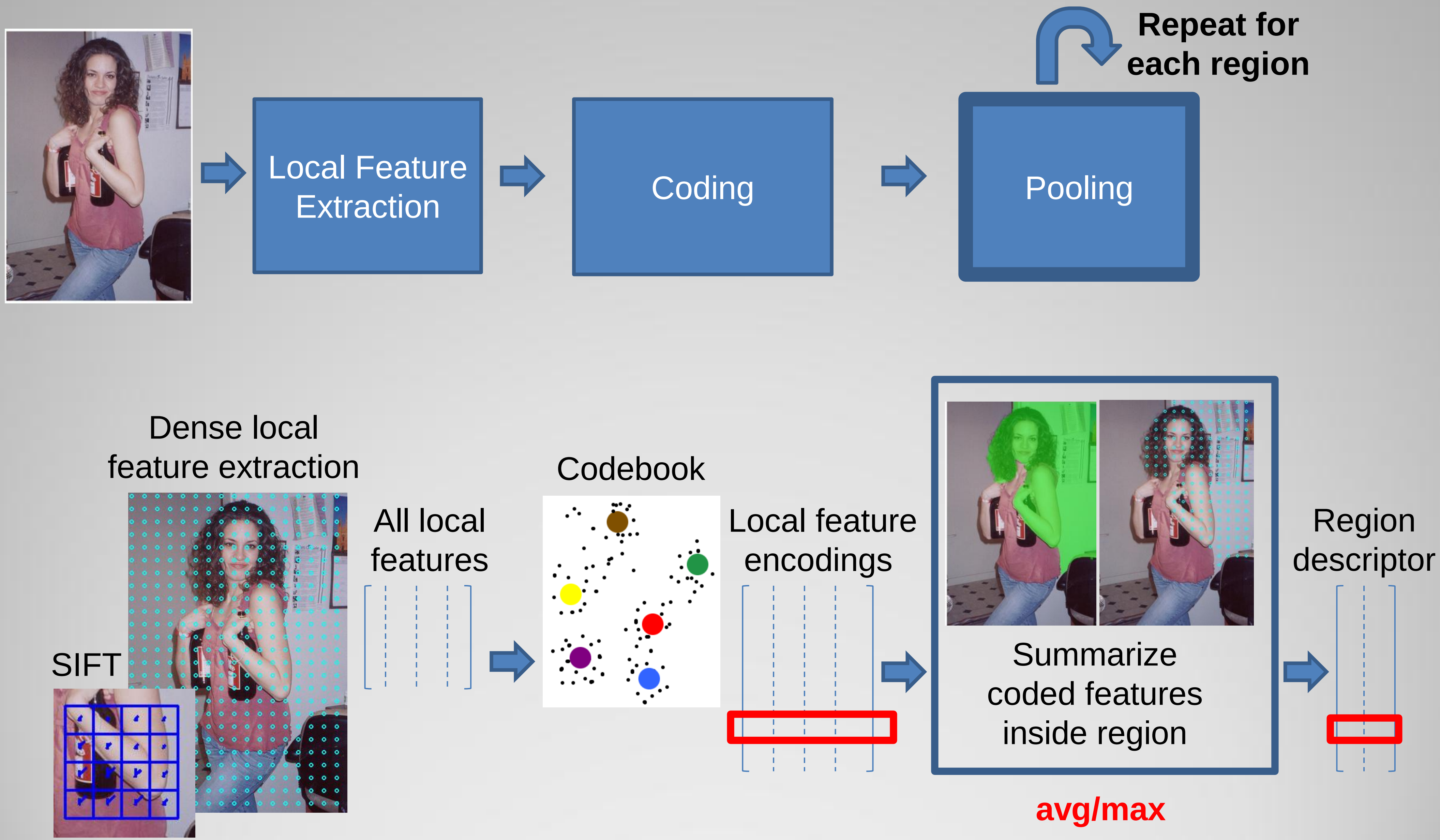


Local feature encodings

Aggregation-based Descriptors



Aggregation-based Descriptors



Aggregation-based Descriptors

	Local Feature Extraction →	Coding →	Pooling →	Descriptor
Bag of Words	Dense SIFT extraction →	Hard Vector Quantization →	Average Pooling →	
Yang et al 09	Dense SIFT extraction →	Sparse Coding →	Max Pooling →	

Aggregation-based Descriptors

Most research so far focused on **coding**

Hard Vector Quantization, Kernel Codebook encoding, Sparse Coding, Fisher encoding, Locality-constrained Linear Coding...

Sivic03, Csurka04, Philbin08, Gemert08 , Yang09, Perronnin10, Wang10, (...)

Pooling has received far less attention

Given N local feature descriptors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ extracted inside region

Max

$$\boldsymbol{g}_{max} = \max_i \boldsymbol{x}_i$$

Average

$$\boldsymbol{g}_{avg} = \frac{1}{N}\sum_i^N \boldsymbol{x}_i$$

Second-Order Pooling

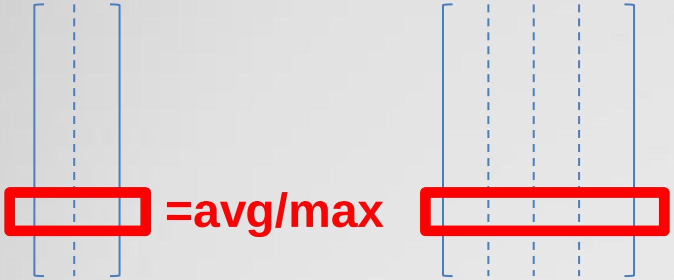
Can we pursue richer statistics for pooling ?

Second-Order Pooling

Can we pursue richer statistics for pooling ?

$$\boldsymbol{g}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i$$

$$\boldsymbol{g}_{max} = \max_{i} \ \boldsymbol{x}_i$$



Given N local feature descriptors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ extracted inside region

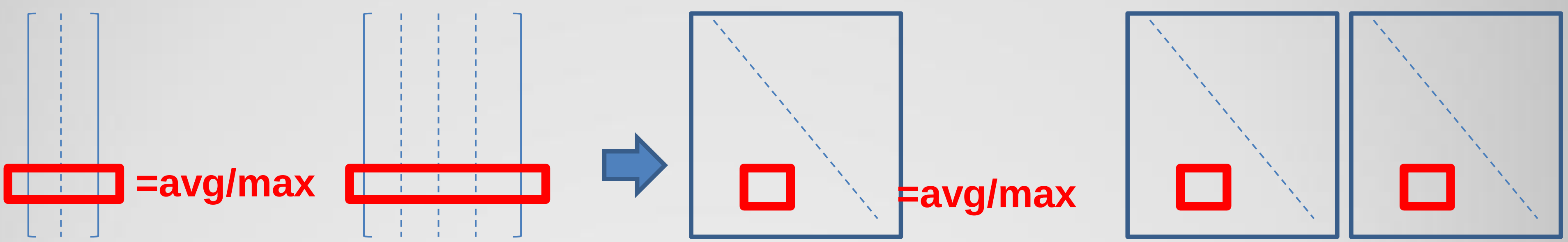
Second-Order Pooling

Can we pursue richer statistics for pooling ?

Capture correlations

$$\boldsymbol{g}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \quad \Longrightarrow \quad \boldsymbol{G}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

$$\boldsymbol{g}_{max} = \max_{i}\ \boldsymbol{x}_i \quad \Longrightarrow \quad \boldsymbol{G}_{max} = \max_{i}\ \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$



Second-Order Pooling

Can we pursue richer statistics for pooling ?

Capture correlations

$$\boldsymbol{g}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \quad \Longrightarrow \quad \boldsymbol{G}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

$$\boldsymbol{g}_{max} = \max_{i} \ \boldsymbol{x}_i \quad \Longrightarrow \quad \boldsymbol{G}_{max} = \max_{i} \ \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

Dimensionality = (local descriptor size)^2

Local Feature Extraction → Coding → Pooling

Second-Order Pooling

Can we pursue richer statistics for pooling ?

Capture correlations

$$\boldsymbol{g}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \quad \Longrightarrow \quad \boldsymbol{G}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

$$\boldsymbol{g}_{max} = \max_{i}\ \boldsymbol{x}_i \quad \Longrightarrow \quad \boldsymbol{G}_{max} = \max_{i}\ \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

Dimensionality = (local descriptor size)^2

Local Feature Extraction $\xrightarrow{\text{Bypass coding}}$ Pooling

Second-Order Pooling

What can we say about these matrices ?

$$\boldsymbol{G}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

$$\boldsymbol{G}_{max} = \max_{i}\ \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

Second-Order Pooling

What can we say about these matrices ?

$$\boldsymbol{G}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$ **Symmetric**

$$\boldsymbol{G}_{max} = \max_{i}\ \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$ **Symmetric**

... so we can simply keep upper triangle

Second-Order Pooling

What can we say about these matrices ?

$$\boldsymbol{G}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

Symmetric Positive Definite (SPD)

$$\boldsymbol{G}_{max} = \max_{i}\ \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

Symmetric

SPD matrices have rich geometry: they form a **Riemannian manifold**

Second-Order Pooling

What can we say about these matrices ?

$$\boldsymbol{G}_{avg} = \frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

Symmetric Positive Definite (SPD)

$$\boldsymbol{G}_{max} = \max_{i}\ \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

Symmetric

SPD matrices have rich geometry: they form a **Riemannian manifold**

- Linear classifiers ignore this additional geometry

Embedding SPD Manifold in Euclidean Space

Usual solution is to flatten the manifold by projecting to local tangent spaces

- Only valid in a local neighborhood, in general

Embedding SPD Manifold in an Euclidean Space

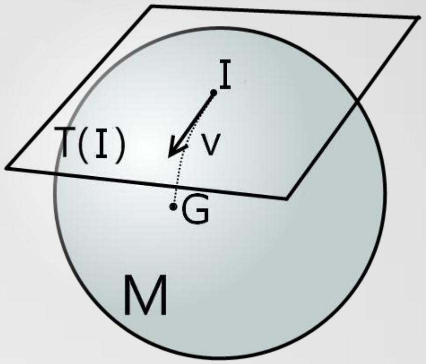
Usual solution is to flatten the manifold by projecting to local tangent spaces

- Only valid in a local neighborhood, in general

By using special, **Log-Euclidean metric**, it is possible to directly embed entire manifold

(Arsigny et al. 07)

$$\boldsymbol{G}_{log} = \log(\boldsymbol{G})$$



Sequence of Operations

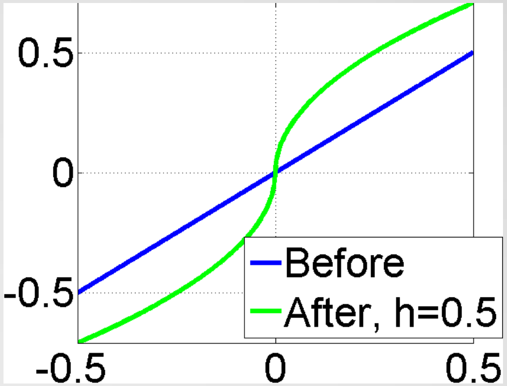
1. Second-Order **Avg** Pooling: $$\log\left(\frac{1}{N}\sum_{i}^{N}\boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}\right)$$

Second-Order **Max** Pooling: $$\max_i \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$$

2. Select upper triangle and convert to vector

3. Power normalize (Perronnin *et al* 2010)

$x = \mathrm{sign}(x) \cdot |x|^h$, with $h \in [0,1]$



Sequence of Operations

1. Second-Order **Avg** Pooling: $\log\left(\frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}\right)$

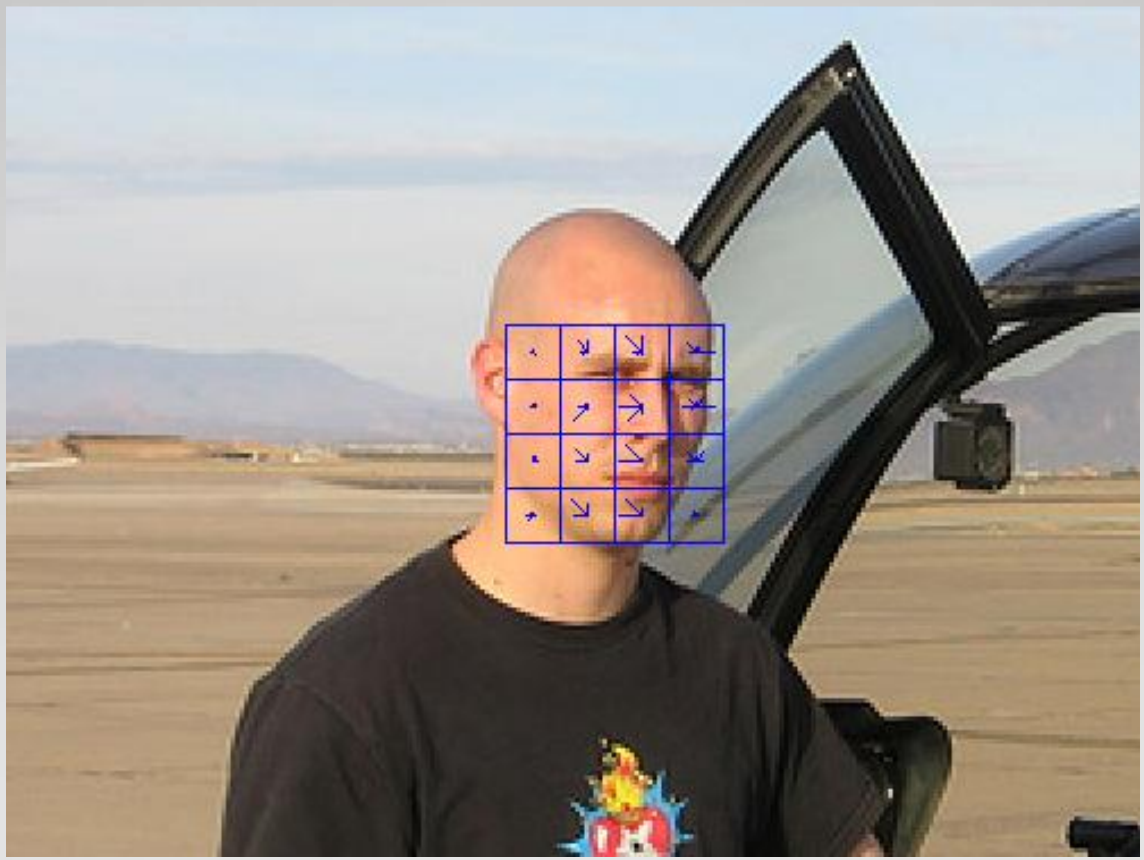
Second-Order **Max** Pooling: $\max_i \; \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}$

2. Select upper triangle and convert to vector

3. Power normalize

Feed resulting descriptor to linear classifier

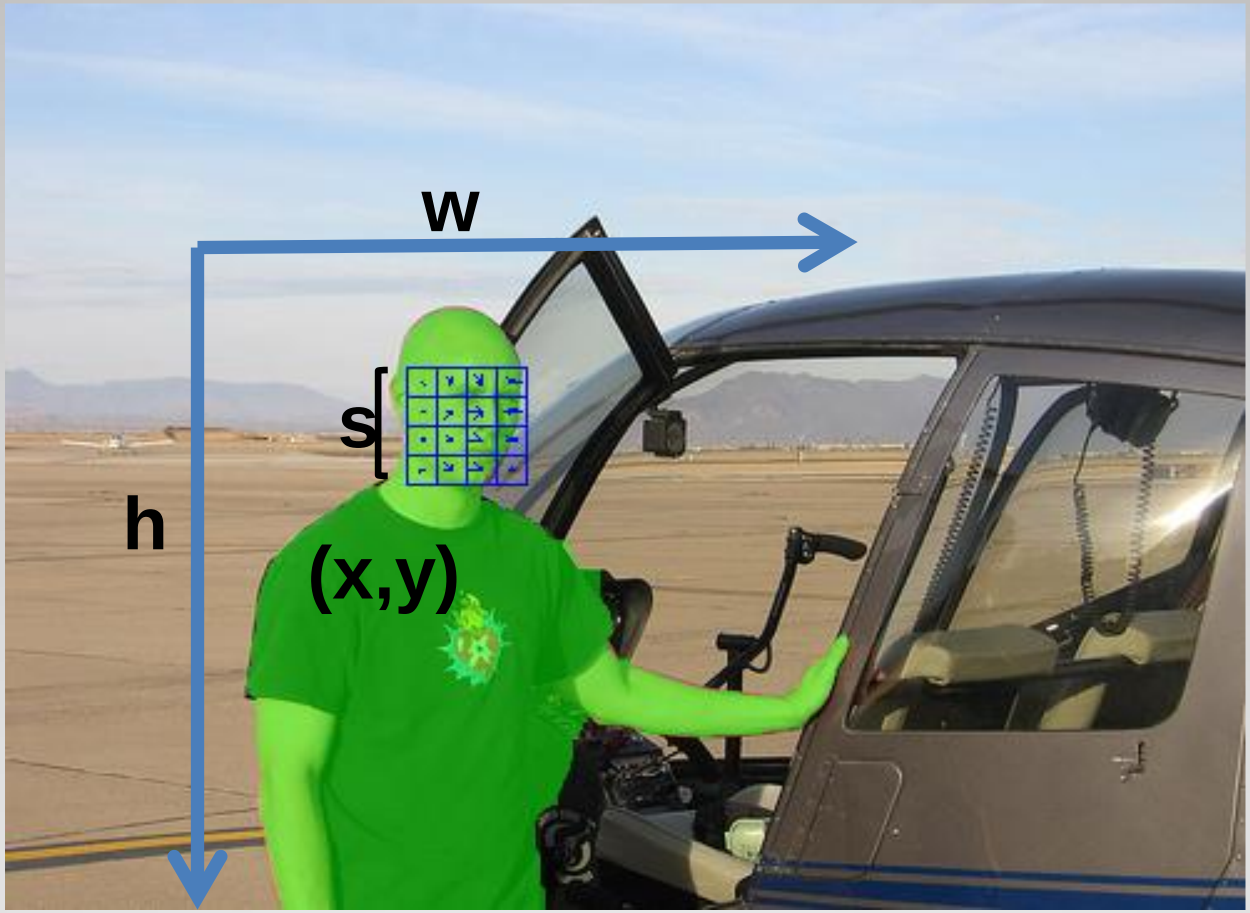
Additionally we use better local descriptors with pooling methods



SIFT

Local Feature Enrichment (1/2)

Relative Position



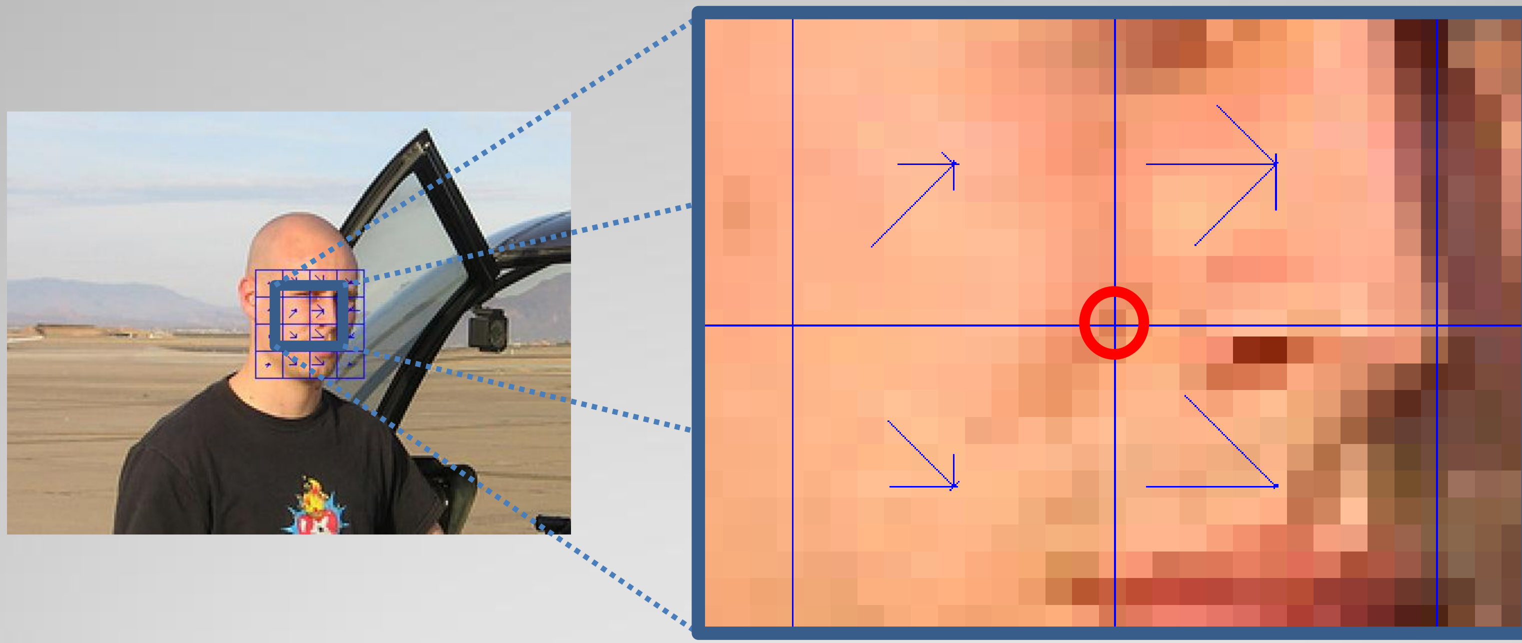
SIFT

Relative position

$\frac{x}{w}$ $\frac{y}{w}$ $\frac{s}{w}$ $\frac{x}{h}$ $\frac{y}{h}$ $\frac{s}{h}$

Local Feature Enrichment (2/2)

Pixel Color

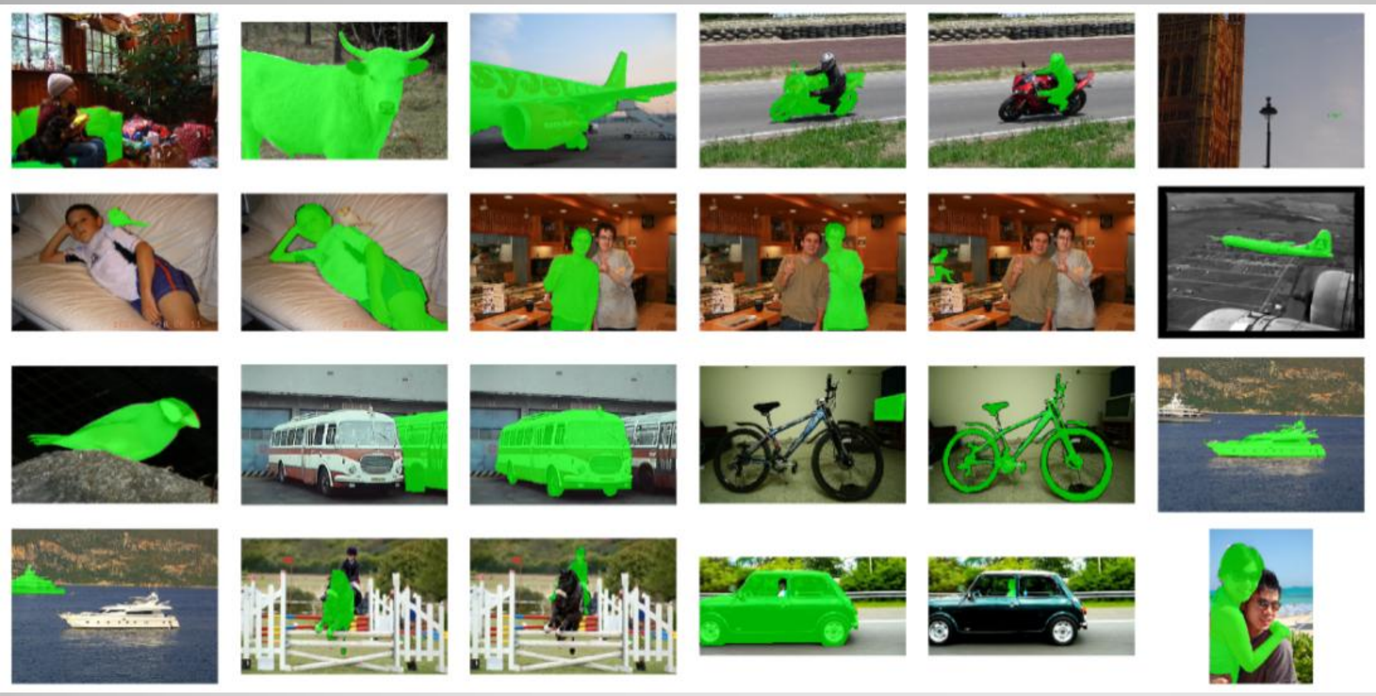


SIFT | Relative position | RGB | LAB | HSV

Concatenate

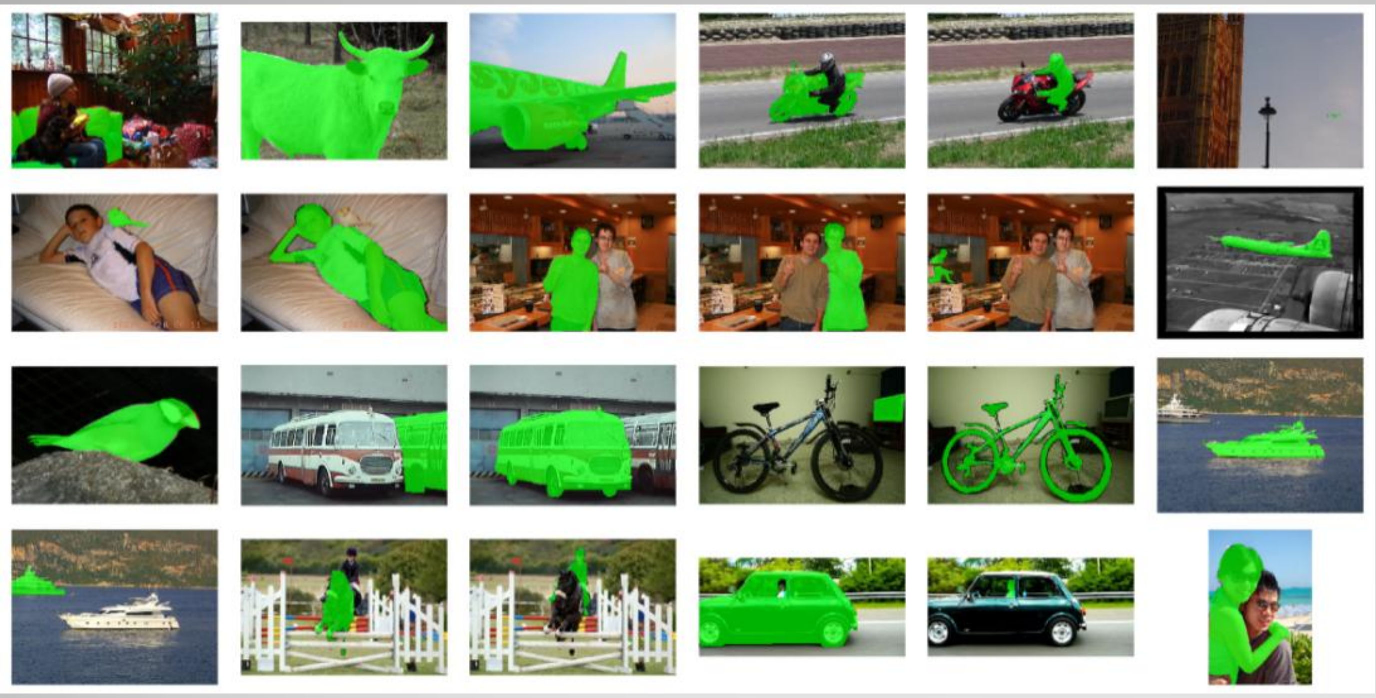
eSIFT

Region Classification VOC 2011



Ground truth regions

Region Classification VOC 2011

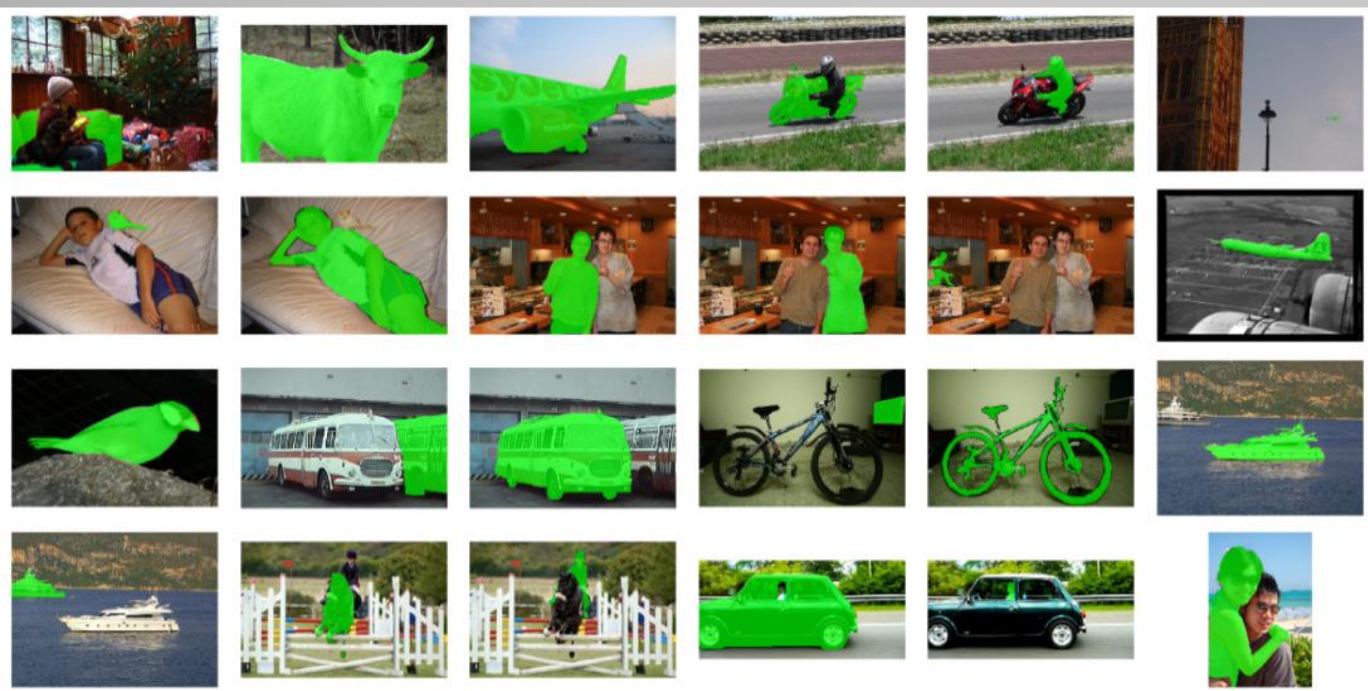


Ground truth regions

Linear classification accuracy

HOG: **41.79%**

Region Classification VOC 2011

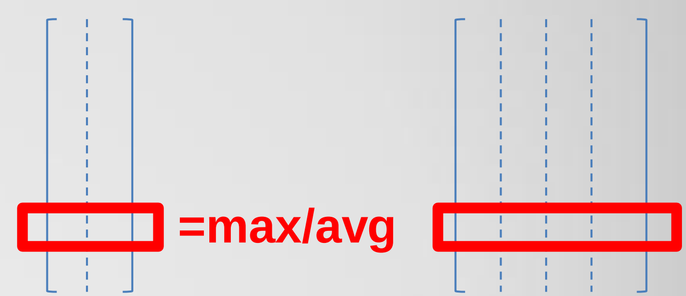


Ground truth regions

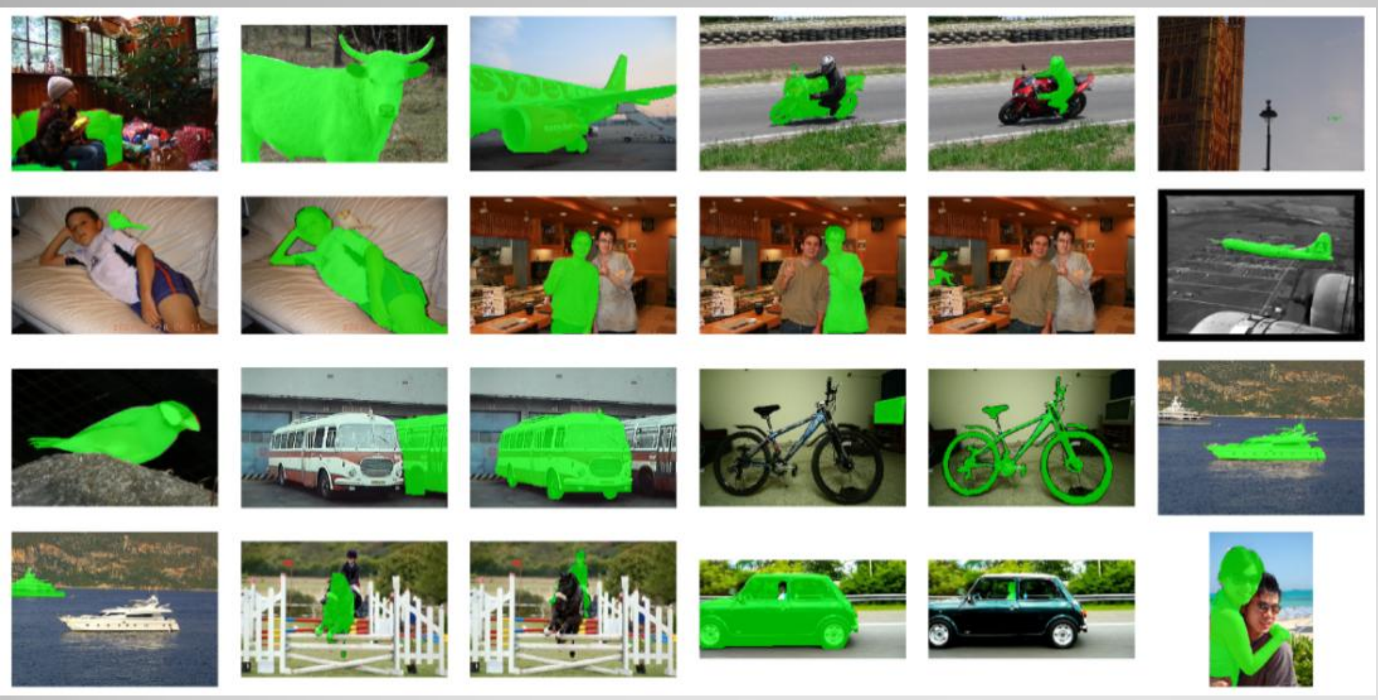
Linear classification accuracy

	1MaxP	1AvgP	2MaxP	2AvgP	Log 2AvgP
SIFT	16.61	33.92			
eSIFT	26.00	**43.33**			

HOG: 41.79%



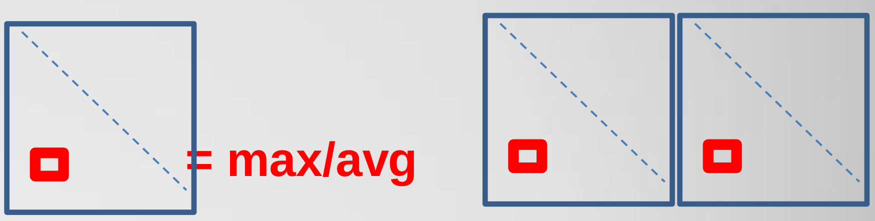
Region Classification VOC2011



Ground truth regions

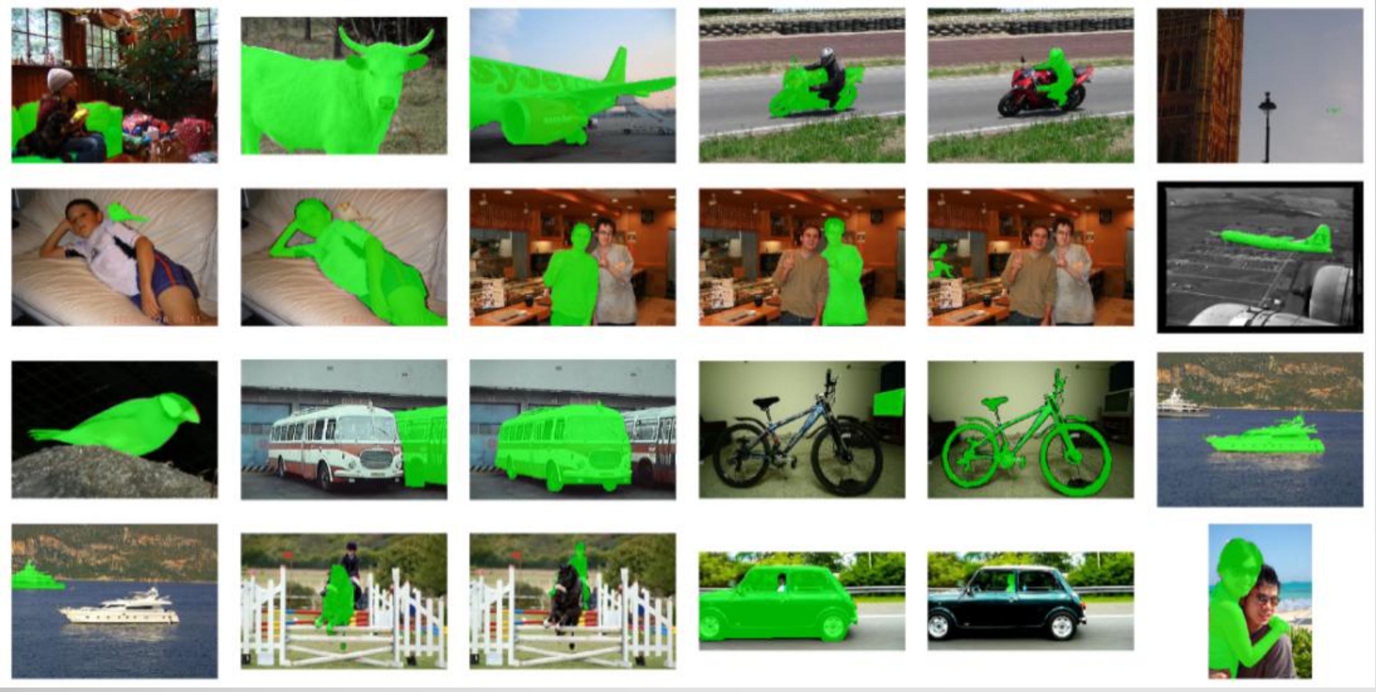
Linear classification accuracy

	1MaxP	1AvgP	2MaxP	2AvgP	Log 2AvgP
SIFT	16.61	33.92	38.74	48.74	
eSIFT	26.00	43.33	50.16	**54.30**	



HOG: 41.79%

Region Classification VOC 2011



Ground truth regions

Linear classification accuracy

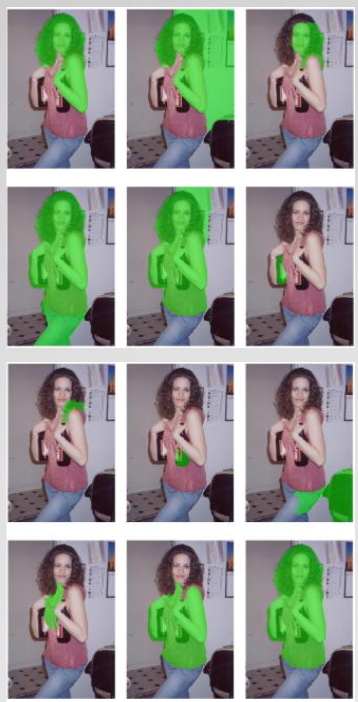
	1MaxP	1AvgP	2MaxP	2AvgP	Log 2AvgP
SIFT	16.61	33.92	38.74	48.74	54.17
eSIFT	26.00	43.33	50.16	54.30	**63.85**

HOG: 41.79%

$$\log\left(\frac{1}{N}\sum_{i}^{N} \boldsymbol{x}_i \cdot \boldsymbol{x}_i^{\mathrm{T}}\right)$$

Semantic Segmentation in the Wild
Pascal VOC 2011

CPMC

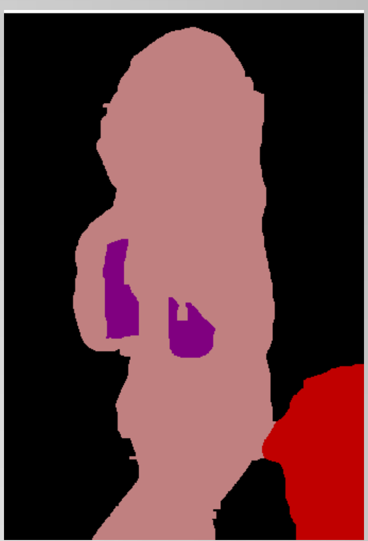


Local Descriptors:
- eSIFT
- Masked eSIFT
- eLBP

Region Descriptors:
- Second-Order Average Pooling

Learning:
- LIBLINEAR

Thresholding + reverse score overlaying



Semantic Segmentation in the Wild
Pascal VOC 2011

This work!!

	comp6		comp5				
	O_2P	**Berkeley**	**BONN-FGT**	**BONN-SVR**	**BROOKES**	**NUS-C**	**NUS-S**
Mean Score	**47.6**	40.8	41.4	43.3	31.3	35.1	37.7
N classes best	**13**	1	2	4	0	0	1

O_2P best on 13 out of 21 categories: background, aeroplane, boat, bus, motorbike, car, train, cat, dog, horse, potted plant, sofa, person

Semantic Segmentation in the Wild
Pascal VOC 2011

	comp6		comp5				
	O_2P	**Berkeley**	**BONN-FGT**	**BONN-SVR**	**BROOKES**	**NUS-C**	**NUS-S**
Mean Score	**47.6**	40.8	41.4	43.3	31.3	35.1	37.7
N classes best	**13**	1	2	4	0	0	1

Linear

Exp-Chi^2 kernels

Semantic Segmentation in the Wild
Pascal VOC 2011

	comp6		comp5				
	O_2P	**Berkeley**	**BONN-FGT**	**BONN-SVR**	**BROOKES**	**NUS-C**	**NUS-S**
Mean Score	**47.6**	40.8	41.4	43.3	31.3	35.1	37.7
N classes best	**13**	1	2	4	0	0	1

Linear

Exp-Chi² kernels

	Feature Extraction	**Prediction**	**Learning**
Exp-Chi²	7.8s / image	87s / image	59h / class
O_2P	4.4s / image	0.004s / image	26m / class
		20,000x faster	**130x faster**

Caltech 101

Important testbed for coding and pooling techniques

Caltech 101

Important testbed for coding and pooling techniques




- No segments, spatial pyramid instead
- Linear classification

Caltech 101

Important testbed for coding and pooling techniques

	SIFT-O_2P	eSIFT-O_2P	SPM[1]	LLC[2]	EMK[3]	MP[4]
Accuracy	**79.2**	**80.8**	64.4	73.4	74.5	77.3

1. *Lazebnik et al. '06*
2. *Wang et al. '10*
3. *Bo & Sminchisescu '10*
4. *Boureau et al. '11*

Conclusions

- Second-order pooling with Log-Euclidean tangent space mappings
- Practical aggregation-based descriptors without unsupervised learning stage (no codebooks)
- High recognition performance on free-form regions using linear classifiers
- Semantic Segmentation on VOC 2011 superior to state-of-the-art with models 20,000x faster

Code available online

Thank you!