Object recognition (part 2)

CSE P 576
Larry Zitnick (larryz@microsoft.com)

Convolutional Nets

Yann Le Cun
The Courant Institute of Mathematical Sciences
New York University
http://yann.lecun.com

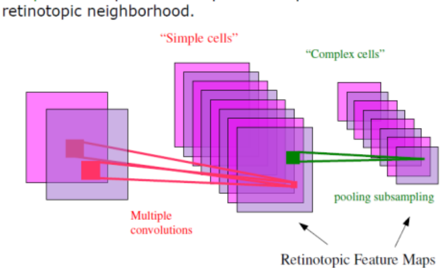
Good Internal Representations are Hierarchical

Feature Transform → - - → Feature Transform → Classifier →

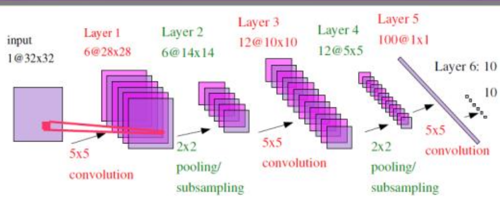
- Low-level features - mid-level features - high-level features - categories
- Representations are increasingly abstract, global, and invariant.
- In Vision: part-whole hierarchy
  - Pixels->Edges->Textons->Parts->Objects->Scenes
- In Language: hierarchy in syntax and semantics
  - Words->Parts of Speech->Sentences->Text
  - Objects,Actions,Attributes...-> Phrases -> Statements -> Stories

An Old Idea for Image Representation with Distortion Invariance

- [Hubel & Wiesel 1962]:
  - simple cells detect local features
  - complex cells "pool" the outputs of simple cells within a retinotopic neighborhood.

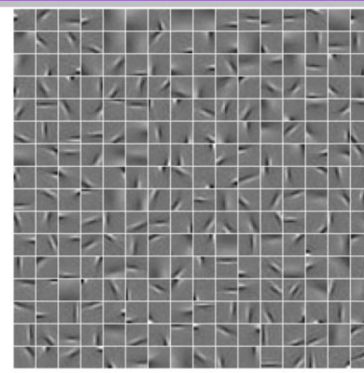


Convolutional Net Architecture

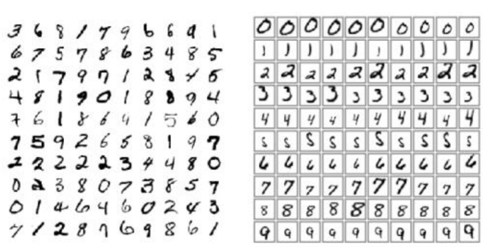


- **Convolutional net for handwriting recognition** (400,000 synapses)
- Convolutional layers (simple cells): all units in a feature plane share the same weights
- Pooling/subsampling layers (complex cells): for invariance to small distortions.
- **Supervised gradient-descent learning using back-propagation**
- **The entire network is trained end-to-end. All the layers are trained simultaneously.**

Learned Features on natural patches: V1-like receptive fields



MNIST Handwritten Digit Dataset



- Handwritten Digit Dataset MNIST: 60,000 training samples, 10,000 test samples

Some Results on MNIST (from raw images: no preprocessing)

CLASSIFIER	DEFORMATION	ERROR	Reference
Knowledge-free methods (a fixed permutation of the pixels would make no difference)			
2-layer NN, 800 HU, CE		1.60	Simard et al., ICDAR 2003
3-layer NN, 500+300 HU, CE, reg		1.53	Hinton, in press, 2005
SVM, Gaussian Kernel		1.40	Cortes 92 + Many others
Convolutional nets			
Convolutional net LeNet-5,		0.80	Ranzato et al. NIPS 2006
Convolutional net LeNet-6,		0.70	Ranzato et al. NIPS 2006
Training set augmented with Affine Distortions			
2-layer NN, 800 HU, CE	Affine	1.10	Simard et al., ICDAR 2003
Virtual SVM deg-9 poly	Affine	0.80	Scholkopf
Convolutional net, CE	Affine	0.60	Simard et al., ICDAR 2003
Training et augmented with Elastic Distortions			
2-layer NN, 800 HU, CE	Elastic	0.70	Simard et al., ICDAR 2003
Convolutional net, CE	Elastic	0.40	Simard et al., ICDAR 2003

Note: some groups have obtained good results with various amounts of preprocessing such as deskewing (e.g. 0.56% using an SVM with smart kernels [deCoste and Schoelkopf]) hand-designed feature representations (e.g. 0.63% with "shape context" and nearest neighbor [Belongie]

Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories

To appear in CVPR 2006

Svetlana Lazebnik (slazebni@uiuc.edu)
Beckman Institute, University of Illinois at Urbana-Champaign

Cordelia Schmid (cordelia.schmid@inrialpes.fr)
INRIA Rhône-Alpes, France

Jean Ponce (ponce@di.ens.fr)
Ecole Normale Supérieure, France

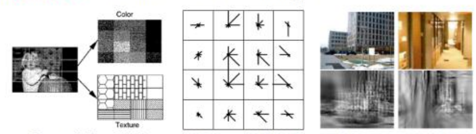
http://www-cvr.ai.uiuc.edu/ponce_grp

1

Overview

- A "pre-attentive" approach: recognize the scene as a whole without examining its constituent objects Biederman (1988), Thorpe et al. (1996), Fei-Fei et al. (2002), Renninger & Malik (2004)
- Inspiration: *locally orderless images* Koenderink & Van Doorn (1999)

- Previous work: "subdivide-and-disorder" strategy

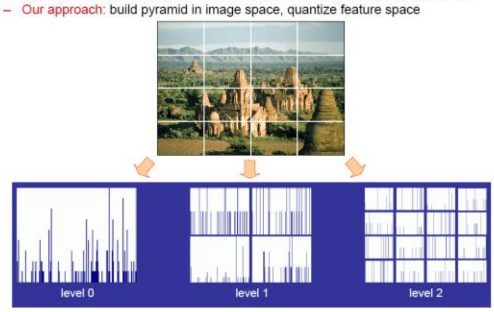


Szummer & Picard (1997) SIFT: Lowe (1999, 2004) Gist: Torralba et al. (2003)

2

Spatial pyramid representation

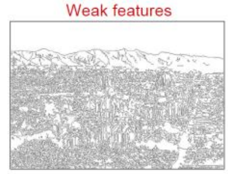
- Extension of a bag of features
- Locally orderless representation at several levels of resolution
- Based on *pyramid match kernels* Grauman & Darrell (2005)
  - Grauman & Darrell: build pyramid in feature space, discard spatial information
  - Our approach: build pyramid in image space, quantize feature space



3

Feature extraction

Weak features



Edge points at 2 scales and 8 orientations (vocabulary size 16)

Strong features

SIFT descriptors of 16x16 patches sampled on a regular grid, quantized to form visual vocabulary (size 200, 400)

5

Support Vector Machines

Modified from the slides by Dr. Andrew W. Moore
http://www.cs.cmu.edu/~awm/tutorials

 Nov 23rd, 2001

Linear Classifiers

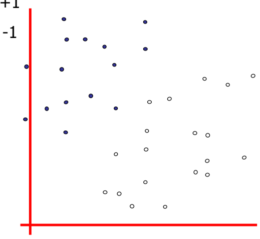
α

$x \rightarrow \boxed{f} \rightarrow y^{est}$

• denotes +1

◦ denotes -1

$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$



How would you classify this data?

 Support Vector Machines: Slide 14

Linear Classifiers

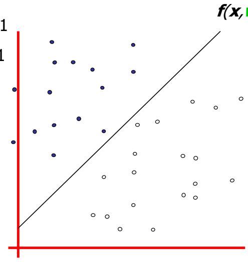
α

$x \rightarrow \boxed{f} \rightarrow y^{est}$

• denotes +1

◦ denotes -1

$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$



How would you classify this data?

 Support Vector Machines: Slide 15

Linear Classifiers

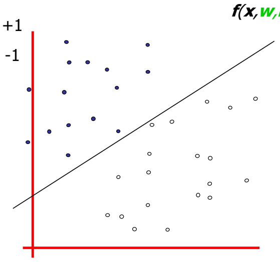
α

$x \rightarrow \boxed{f} \rightarrow y^{est}$

• denotes +1

◦ denotes -1

$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$



How would you classify this data?

 Support Vector Machines: Slide 16

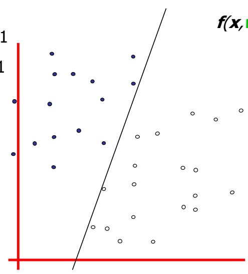
Linear Classifiers

$x \rightarrow \boxed{f} \rightarrow y^{est}$, with α as the input to f

• denotes +1

◦ denotes -1

***f*(*x*,*w*,*b*) = *sign*(*w*. *x* - *b*)**



How would you classify this data?

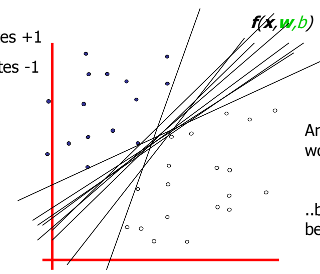
Linear Classifiers

$x \rightarrow \boxed{f} \rightarrow y^{est}$, with α as the input to f

• denotes +1

◦ denotes -1

***f*(*x*,*w*,*b*) = *sign*(*w*. *x* - *b*)**



Any of these would be fine..

..but which is best?

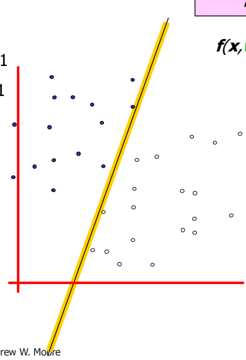
Classifier Margin

$x \rightarrow \boxed{f} \rightarrow y^{est}$, with α as the input to f

• denotes +1

◦ denotes -1

***f*(*x*,*w*,*b*) = *sign*(*w*. *x* - *b*)**



Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

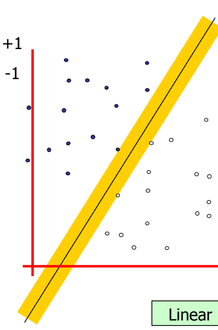
Maximum Margin

$x \rightarrow \boxed{f} \rightarrow y^{est}$, with α as the input to f

• denotes +1

◦ denotes -1

***f*(*x*,*w*,*b*) = *sign*(*w*. *x* - *b*)**

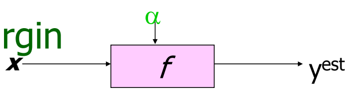


The maximum margin linear classifier is the linear classifier with the, um, maximum margin.

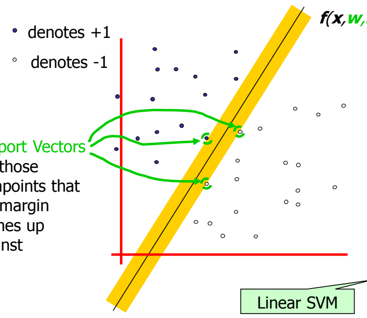
This is the simplest kind of SVM (Called an LSVM)

Linear SVM

Maximum Margin



$f(\boldsymbol{x},\boldsymbol{w},b) = sign(\boldsymbol{w}.\ \boldsymbol{x} - b)$



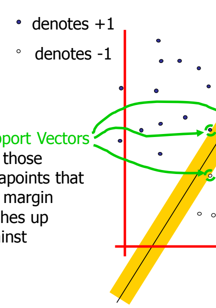
Support Vectors are those datapoints that the margin pushes up against

The maximum margin linear classifier is the linear classifier with the, um, maximum margin.

This is the simplest kind of SVM (Called an LSVM)

Linear SVM

Why Maximum Margin?



Support Vectors are those datapoints that the margin pushes up against

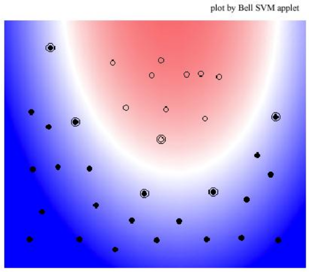
1. Intuitively this feels safest.
2. If we've made a small error in the location of the boundary (it's been jolted in its perpendicular direction) this gives us least chance of causing a misclassification.
3. LOOCV is easy since the model is immune to removal of any non-support-vector datapoints.
4. There's some theory (using VC dimension) that is related to (but not the same as) the proposition that this is a good thing.
5. Empirically it works very very well.

Nonlinear Kernel (I)

Example: SVM with Polynomial of Degree 2

Kernel: $K(\vec{x}_i,\vec{x}_j) = [\vec{x}_i \cdot \vec{x}_j + 1]^2$

plot by Bell SVM applet

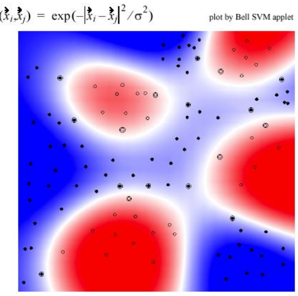


Nonlinear Kernel (II)

Example: SVM with RBF-Kernel

Kernel: $K(\vec{x}_i,\vec{x}_j) = \exp(-|\vec{x}_i - \vec{x}_j|^2/\sigma^2)$

plot by Bell SVM applet



Scene category dataset

Fei-Fei & Perona (2005), Oliva & Torralba (2001)

http://www-cvr.ai.uiuc.edu/ponce_grp/data

Multi-class classification results (100 training images per class)

	Weak features (vocabulary size: 16)		Strong features (vocabulary size: 200)	
Level	Single-level	Pyramid	Single-level	Pyramid
0 (1×1)	45.3 ±0.5		72.2 ±0.6	
1 (2×2)	53.6 ±0.3	56.2 ±0.6	77.9 ±0.6	79.0 ±0.5
2 (4×4)	61.7 ±0.6	64.7 ±0.7	79.4 ±0.3	**81.1** ±0.3
3 (8×8)	63.3 ±0.8	**66.8** ±0.6	77.2 ±0.4	80.7 ±0.3

Fei-Fei & Perona: 65.2%

6

Caltech101 dataset

Fei-Fei et al. (2004)

http://www.vision.caltech.edu/Image_Datasets/Caltech101/Caltech101.html

Multi-class classification results (30 training images per class)

	Weak features (16)		Strong features (200)	
Level	Single-level	Pyramid	Single-level	Pyramid
0	15.5 ±0.9		41.2 ±1.2	
1	31.4 ±1.2	32.8 ±1.3	55.9 ±0.9	57.0 ±0.8
2	47.2 ±1.1	49.3 ±1.4	63.6 ±0.9	**64.6** ±0.8
3	52.2 ±0.8	**54.0** ±1.1	60.3 ±0.9	64.6 ±0.7

9

http://www.vision.caltech.edu/Image_Datasets/Caltech101/

Most Visited Getting Started Latest Headlines Google Apple Yahoo! YouTube Wikipedia News Popular Google Maps Elsevier Edit

COMPUTATIONAL VISION AT CALTECH

Caltech 101

Caltech256

[Description][Download][Discussion [Other Datasets]

Description

Pictures of objects belonging to 101 categories. About 40 to 800 images per category. Most categories have about 50 images. Collected in September 2003 by Fei-Fei Li, Marco Andreetto, and Marc 'Aurelio Ranzato. The size of each image is roughly 300 pixels.

We have carefully clicked outlines of each object in these pictures, these are included under the 'Annotations.tar'. There is also a matlab script to view the annotaitons, 'show_annotations.m'.

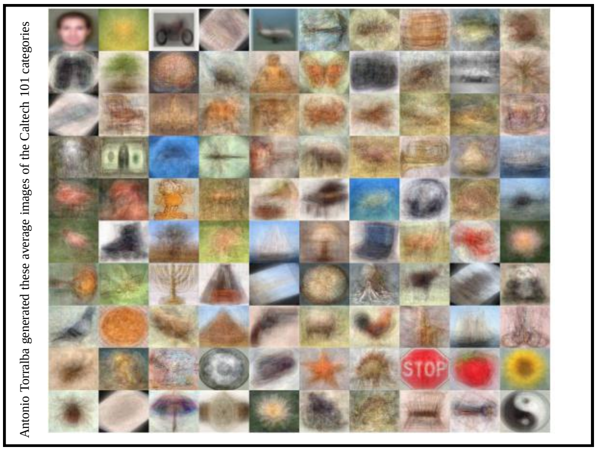
How to use the dataset

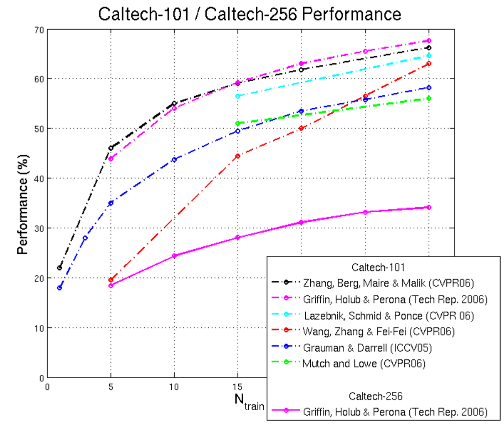
Caltech-101: Drawbacks

- Smallest category size is 31 images: $N_{train} \leq 30$
- Too easy?
  - left-right aligned

  - Rotation artifacts

  - Soon will saturate performance





- Jump to Nicolas Pinto's slides. (page 29)

Papers

- A. Torralba. Contextual priming for object detection. IJCV 2003.
- A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora and S. Belongie. Objects in Context. ICCV 2007

Object Detection

Probabilistic Framework

Object presence at a particular location/scale

$$P(O \mid \mathbf{v}) = \frac{P(\mathbf{v} \mid O)}{P(\mathbf{v})} P(O)$$

(Single Object Likelihood)

Given all image features (local/object and scene/context)

$\mathbf{v} = \mathbf{v}_{Local} + \mathbf{v}_{Contextual}$

Contextual Reasoning

2D Reasoning

2,5D / 3D Reasoning

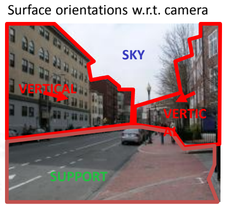
Scene Centered

Object Centered

Surface orientations w.r.t. camera

Contextual priming for object detection

Objects in Context

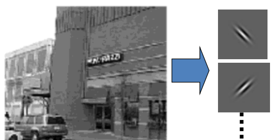


Geometric context from a single image.

Preview: Contextual Priming for Object Detection

Input test image

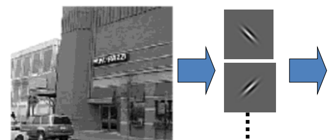
Preview: Contextual Priming for Object Detection



Correlate with many filters

37

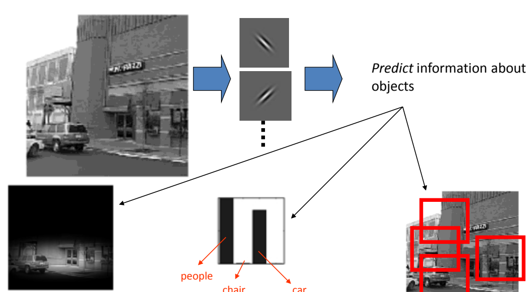
Preview: Contextual Priming for Object Detection



Using previously collected statistics about filter output *predict* information about objects

38

Preview: Contextual Priming for Object Detection



Where I can find the objects easily?

Which objects do I expect to see?

How large objects do I expect to see?

39

Contextual Priming for Object Detection: Probabilistic Framework

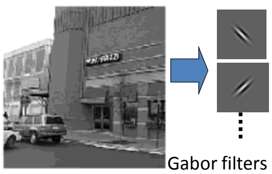
$$P(O \mid \mathbf{v}) = \frac{P(O, \mathbf{v})}{P(\mathbf{v})} = \frac{P(\mathbf{v}_L \mid O, \mathbf{v}_C)}{P(\mathbf{v}_L \mid \mathbf{v}_C)} P(O \mid \mathbf{v}_C)$$

Local measurements
(a lot in the literature)

Contextual features

40

Contextual Priming for Object Detection: Contextual Features



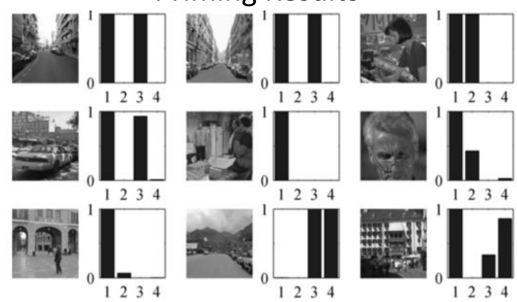
Gabor filters at
4 scales and
6 orientations

Use **PCA** on filter output images to reduce the number of features (< 64)

Use **Mixture of Gaussians** to model the probabilities. (Other alternatives include KNN, parzen window, logistic regression, etc)

41

Contextual Priming for Object Detection: Object Priming Results



(o_1 =people, o_2 =furniture, o_3 =vehicles and o_4 =trees

42

Contextual Priming for Object Detection: Focus of Attention Results

Heads

43

Contextual Priming for Object Detection: **Conclusions**

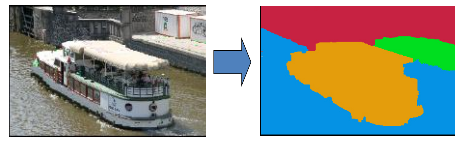
- Proves the relation btw low level features and scene/context
- Can be seen as a computational evidence for the (possible) existence of low-level feature based biological attention mechanisms
- Also a warning: Whether an object recognition system understands the object or works by lots bg features.

44

Preview: Objects in Context

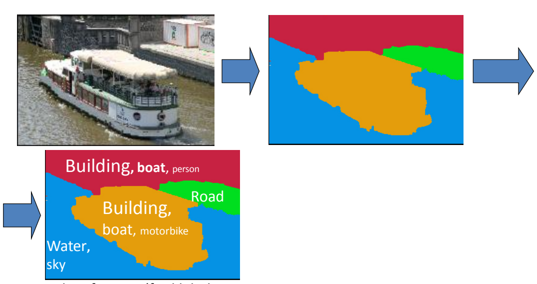
Input test image

Preview: Objects in Context



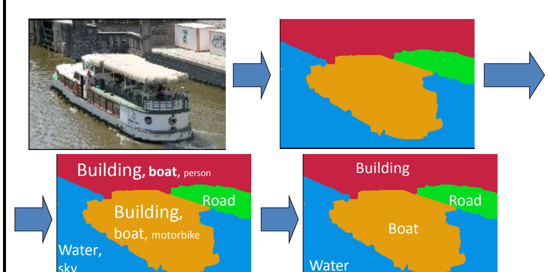
Do segmentation on the image

Preview: Objects in Context



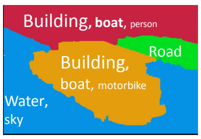
Do classification (find label probabilities) in each segment only with local info

Preview: Objects in Context



Most consistent labeling according to *object co-occurrences* & local label probabilities.

Objects in Context:
Local Categorization



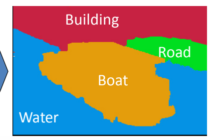
- Extract random patches on zero-padded segments
- Calculate SIFT descriptors

- Use BoF:
  Training:
  - Cluster patches in training (Hier. K-means, K=10x3)
  - Histogram of words in each segment
  - NN classifier (returns a sorted list of categories)

Each segment is classified independently

49

Objects in Context:
Contextual Refinement



Contextual model based on co-occurrences
Try to find the most consistent labeling with **high posterior probability** and **high mean pairwise interaction**.
Use CRF for this purpose.

$$p(c_1 \dots c_k | S_1 \dots S_k) = \frac{B(c_1 \dots c_k) \prod_{i=1}^{k} A(i)}{Z(\phi, S_1 \dots S_k)}$$

$$B(c_1 \dots c_k) = \exp\left(\sum_{i,j=1}^{k} \phi(c_i, c_j)\right)$$

Mean interaction of all label pairs
Φ(i,j) is basically the observed label co-occurrences in training set.

$$A(i) = p(c_i | S_i)$$

Independent segment classification

50

Objects in Context:
Learning Context

Using labeled image datasets (MSRC, PASCAL)

Using labeled text based data (Google Sets): Contains list of related items
→ A large set turns out to be useless! (anything is related)

51

Objects in Context:
Results

	No Context	Google Sets	Using Training
MSRC	45.0%	58.1%	68.4%
PASCAL	61.8%	63.4%	74.2%

Table 1. Average Categorization Accuracy.

52

"Objects in Context" – Limitations: Context modeling

Segmentation

Categorization without context
Local information only

With **co-occurrence** context
Means: **P(person,dog) > P(person, cow)**

(Bonus Q: How did it handle the background?)

53

"Objects in Context" – Limitations: Context modeling

Segmentation

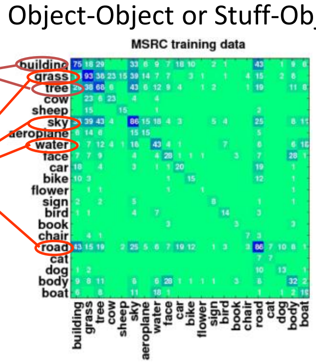
Categorization without context
Local information only

With **co-occurrence** context
P(person,horse) > P(person, dog)

But why? Isn't it only a dataset bias? We have seen in the previous example that P(person,dog) is common too.

54

"Objects in Context"
Object-Object or Stuff-Object ?



Stuff-like

Stuff

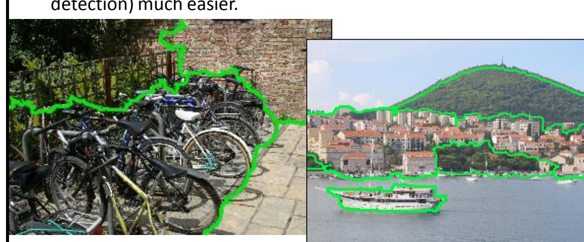
Labels with high co-occurrences with other labels

Looks like "background" stuff – object (such as **water-boat**) does help **rather than** "foreground" object co-occurrences (such as **person-horse**)
[but still car-person-motorbike is useful in PASCAL]

55

"Objects in Context" – Limitations: Segmentation

- Too good: A few or many? How to select a good segmentation in multiple segmentations?
- Can make object recognition & contextual reasoning (due to stuff detection) much easier.



"Objects in Context" - Limitations

- No cue by **unknown objects**
- No **spatial relationship** reasoning
- Object detection part heavily depends on **good segmentations**
- Improvements using object co-occurrences are demonstrated with images where **many labels are already correct**. → How good is the model?

57

Contextual Priming vs. Objects in Context

Scene->Object

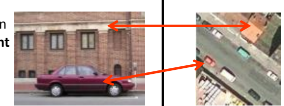
Simpler training data
(only target object's labels are enough)

Scene information is **view-dependent**
(due to gist)

Object **detector independent**

{Object,Stuff} <-> {Object,Stuff}

May need huge amount of **labeled data**



Can be **more generic** than scene->object **with** a very good model

Contextual model is object detector independent, in theory. **But**:

\+ use segmentation → easier to detect **stuff**

\- uses segmentation → can be **unreliable**

58

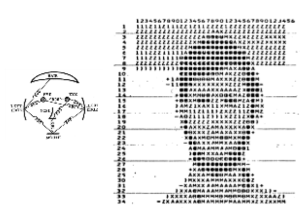
Finding the weakest link in person detectors

Devi Parikh
TTI, Chicago

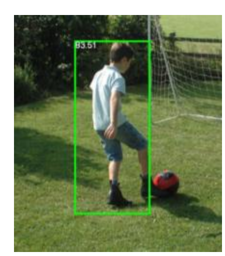
Larry Zitnick
Microsoft Research

Object recognition

We've come a long way…

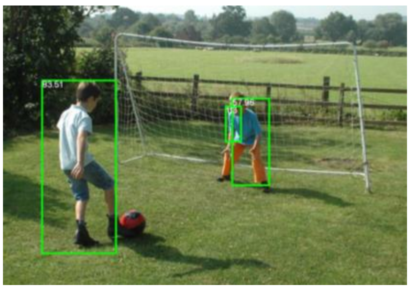


Fischler and Elschlager, 1973

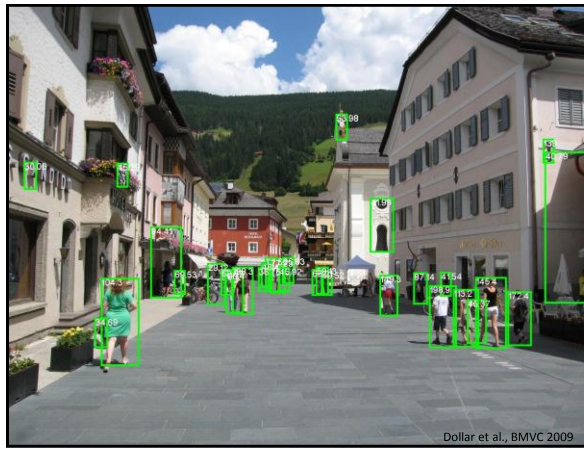


Dollar et al., BMVC 2009

Still a ways to go…

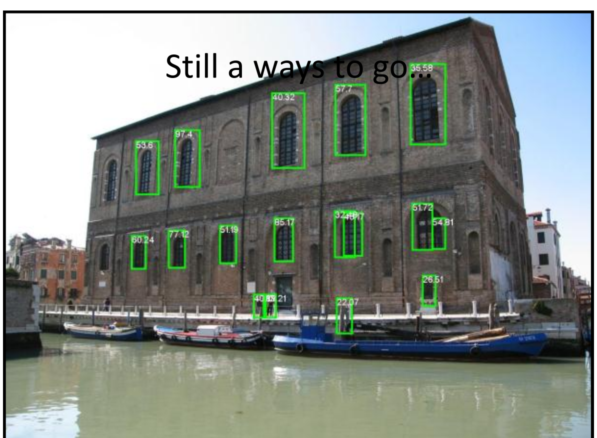


Dollar et al., BMVC 2009



Dollar et al., BMVC 2009

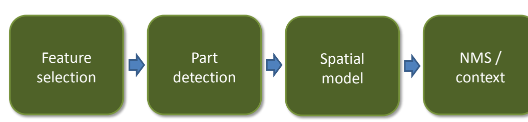
Still a ways to go…



Dollar et al., BMVC 2009

Part-based person detector

4 main components:

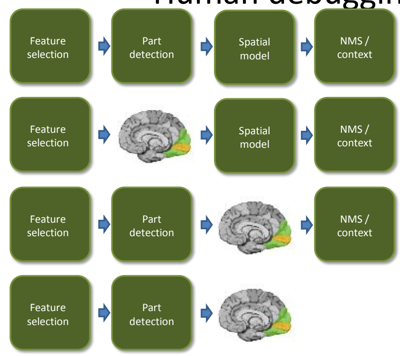


Felzenszwalb et al., 2008

How can we help?

- Humans supply training data…
  - 100,000s labeled images
- We design the algorithms.
  - Going on 40 years.
- Can we use humans to debug?

Human debugging



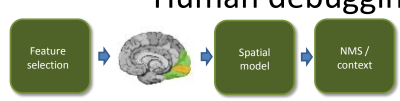

Amazon Mechanical Turk

Human performance

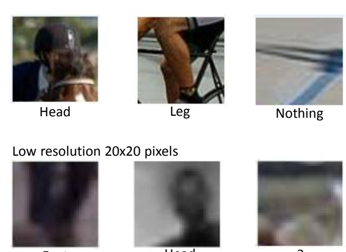
- Humans ~90% average precision
- Machines ~46% average precision


PASCAL VOC dataset

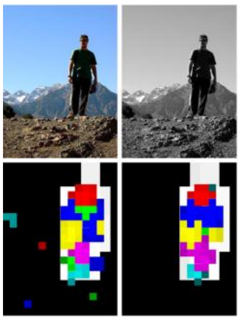
Human debugging



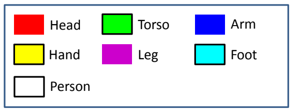
Is it a head, torso, arm, leg, foot, hand, or nothing?



Part detections



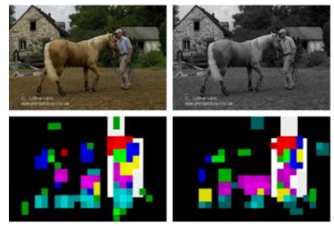
Humans



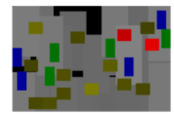
Machine

P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. on PAMI*, 32:1627–1645, 2010.

Part detections

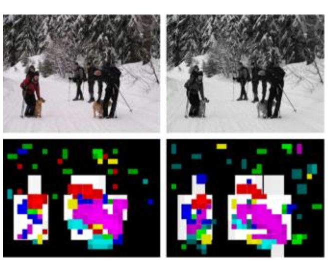


Humans

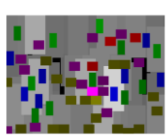


Machine

Part detections

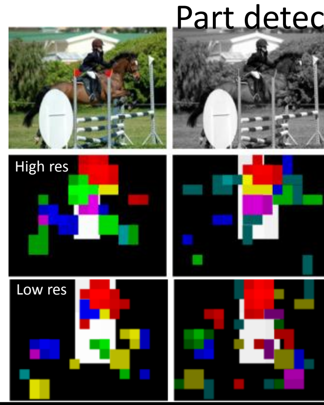


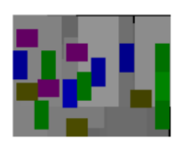
Humans



Machine

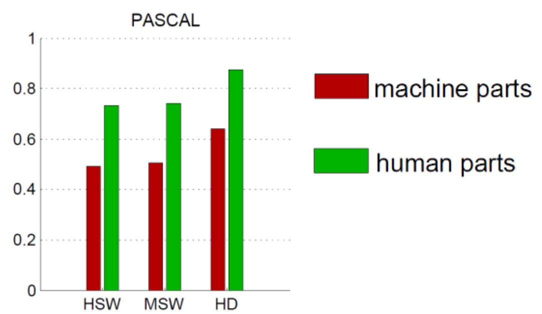
Part detections



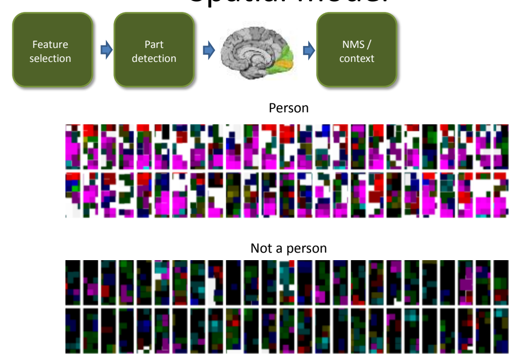


Machine

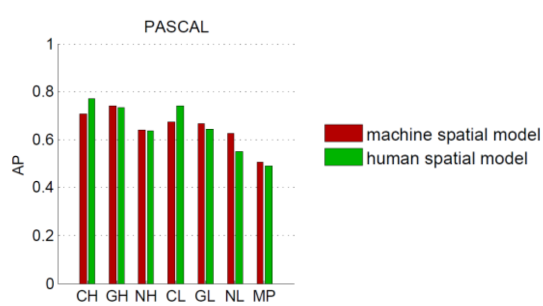
AP results



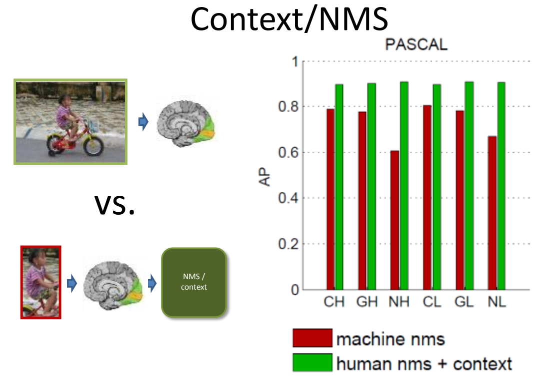
Spatial model



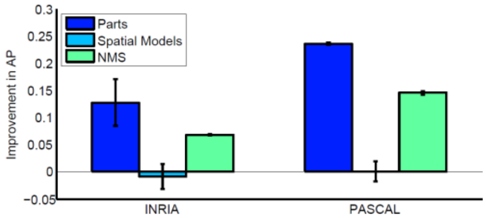
Spatial model



Context/NMS



Conclusion



http://www.ted.com/talks/lang/eng/pawan_sinha_on_how_brains_learn_to_see.html 7:00min

Pawan Sinha on how brains learn to see