Computer Vision

CSE/EE 576
Interest Regions, Recognition, and Matching

Linda Shapiro
Professor of Computer Science & Engineering
Professor of Electrical & Computer Engineering

The Kadir Operator Saliency, Scale and Image Description

Timor Kadir and Michael Brady
University of Oxford

The issues…

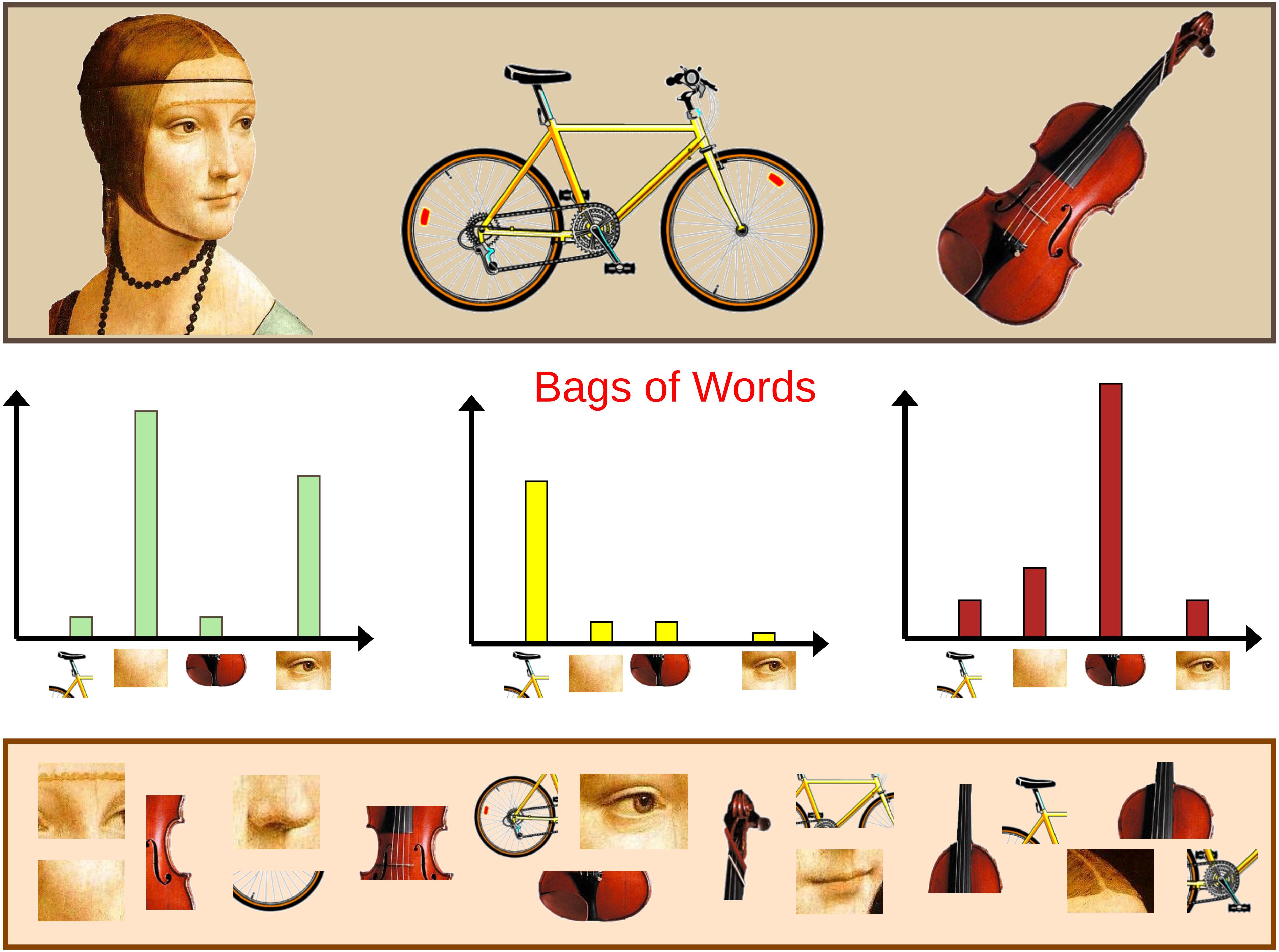
- salient – standing out from the rest, noticeable, conspicous, prominent

- scale – find the best scale for a feature

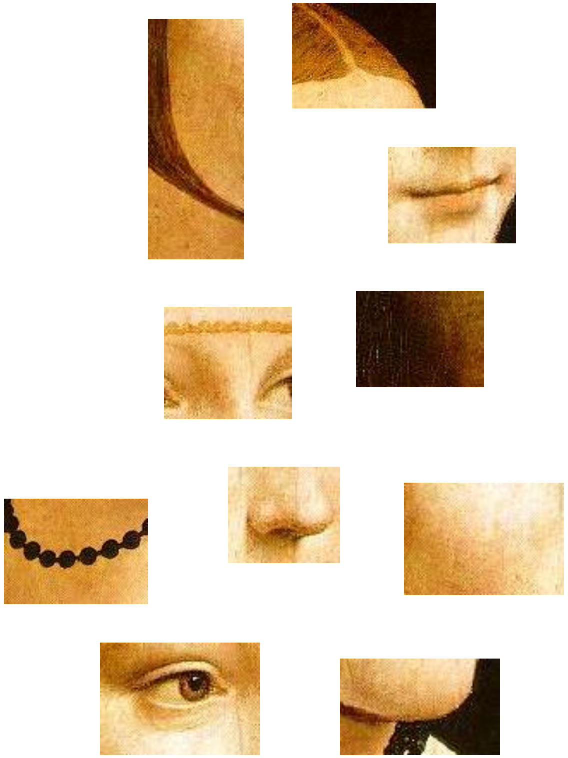
- image description – create a descriptor for use in object recognition

Early Vision Motivation

- pre-attentive stage: features pop out
- attentive stage: relationships between features and grouping


Bags of Words

Detection of Salient Features for an Object Class



How do we do this?

1. fixed size windows (simple approach)
2. Harris detector, Lowe detector, etc.
3. Kadir's approach

Kadir's Approach

- Scale is intimately related to the problem of determining saliency and extracting relevant descriptions.

- Saliency is related to the local image complexity, ie. Shannon entropy.

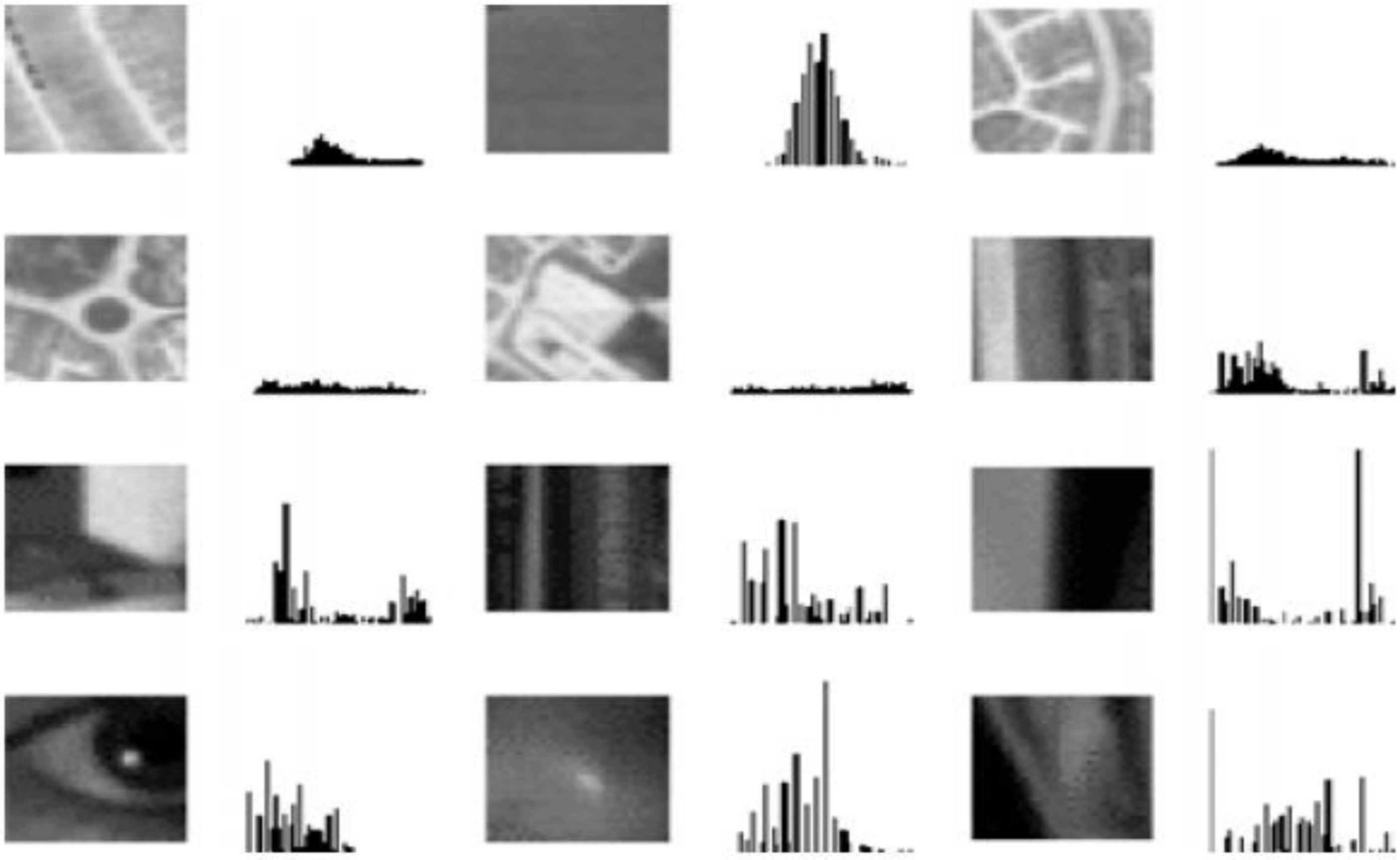
- entropy definition $H = -\sum_{\substack{i \text{ in set} \\ \text{of interest}}} P_i \log_2 P_i$

Specifically

- x is a point on the image
- R_x is its local neighborhood
- D is a descriptor and has values $\{d_1, \ldots d_r\}$.
- $P_{D,Rx}(d_i)$ is the probability of descriptor D taking the value d_i in the local region R_x. (The normalized histogram of the gray tones in a region estimates this probability distribution.)

$$H_{D,R_X} = -\sum_i P_{D,R_X}(d_i)\log_2 P_{D,R_X}(d_i)$$

Local Histograms of Intensity



Neighborhoods with structure have flatter distributions which converts to higher entropy.

Problems Kadir wanted to solve

1. Scale should not be a global, preselected parameter

2. Highly textured regions can score high on entropy, but not be useful

3. The algorithm should not be sensitive to small changes in the image or noise.

Kadir’s Methodology

- use a scale-space approach

- features will exist over multiple scales
  - Berghoml (1986) regarded features (edges) that existed over multiple scales as best.

- Kadir took the opposite approach.
  - He considers these too self-similar.
  - Instead he looks for peaks in (weighted) entropy over the scales.

The Algorithm

1. For each pixel location x
   a. For each scale s between smin and smax
      i. Measure the local descriptor values within a window of scale s
      ii. Estimate the local PDF (use a histogram)
   b. Select scales (set S) for which the entropy is peaked (S may be empty)
   c. Weight the entropy values in S by the sum of absolute difference of the PDFs of the local descriptor around S.

Finding salient points

- the math for saliency discretized

$$Y_D(\mathbf{s},\mathbf{x}) = H_D(\mathbf{s},\mathbf{x})W_D(\mathbf{s},\mathbf{x})$$

$$H_D(\mathbf{s},\mathbf{x}) = -\sum_{d \in D} p_{\mathbf{s},\mathbf{x}}(d)\log_2 p_{\mathbf{s},\mathbf{x}}(d)$$

$$W_D(\mathbf{s},\mathbf{x}) = \frac{s^2}{2s-1}\sum_{d \in D}\left|p_{\mathbf{s},\mathbf{x}}(d) - p_{\mathbf{s}-1,\mathbf{x}}(d)\right|$$

$\mathbf{x}$ = point

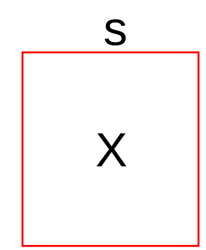
$\mathbf{s} = (\mathrm{s}, \mathrm{r}, \theta) = ($scale, [redacted] $)$

D = low-level feature domain (gray tones)

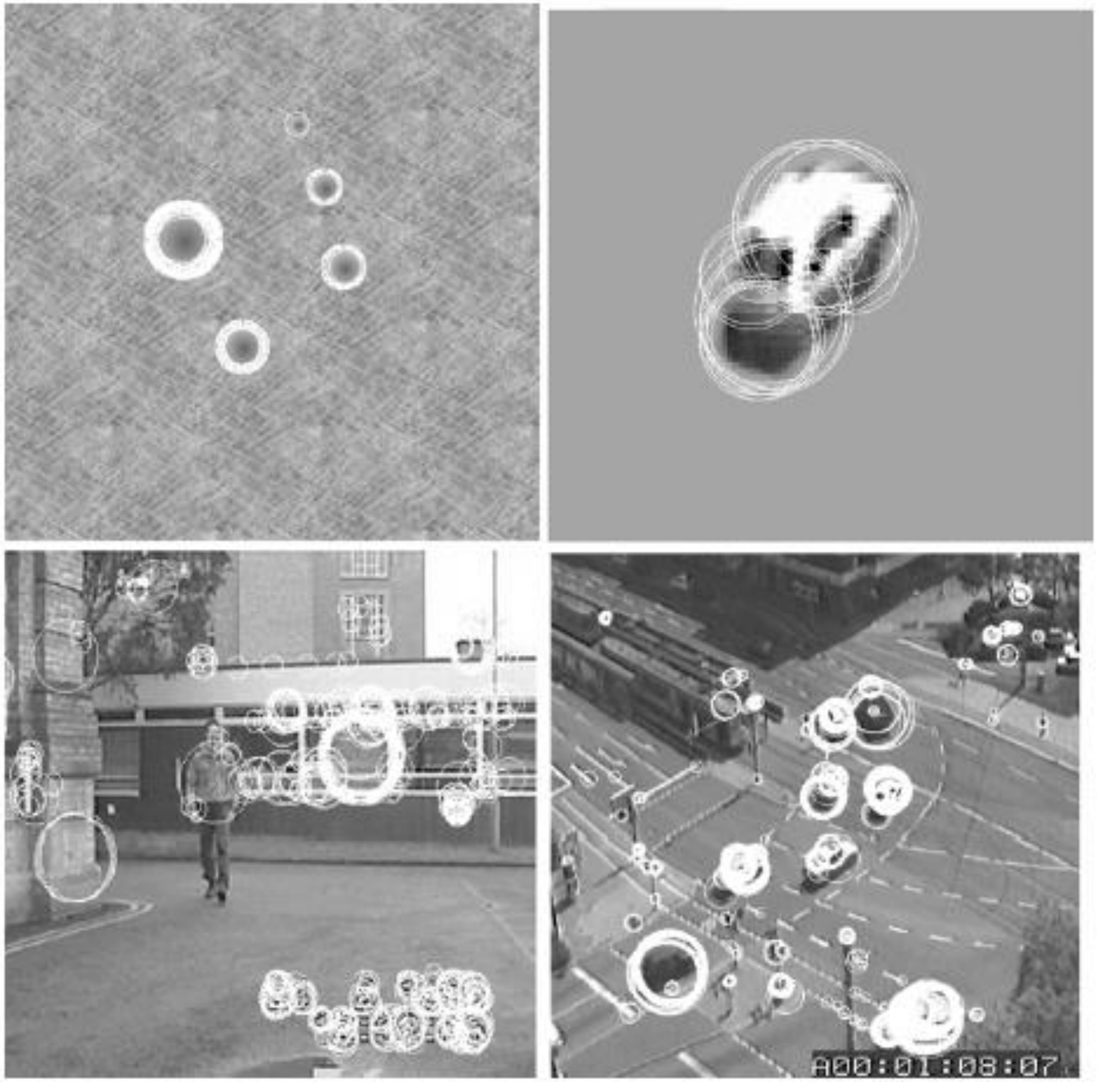
$p_{\mathbf{s},\mathbf{x}}(d)$ = probability of descriptor D taking value d in the region centered at $\mathbf{x}$ with scale $\mathbf{s}$

= normalized histogram count for the bin representing gray tone d.

- saliency
- entropy
- weight based on difference between scales



Picking salient points and their scales



Getting rid of texture

- One goal was to not select highly textured regions such as grass or bushes, which are not the type of objects the Oxford group wanted to recognize
- Such regions are highly salient with just entropy, because they contain a lot of gray tones in roughly equal proportions
- But they are similar at different scales and thus the weights make them go away

Salient Regions

- Instead of just selecting the most salient points (based on weighted entropy), select salient regions (more robust).
- Regions are like volumes in scale space.
- Kadir used clustering to group selected points into regions.
- We found the clustering was a critical step.

Kadir's clustering (VERY ad hoc)

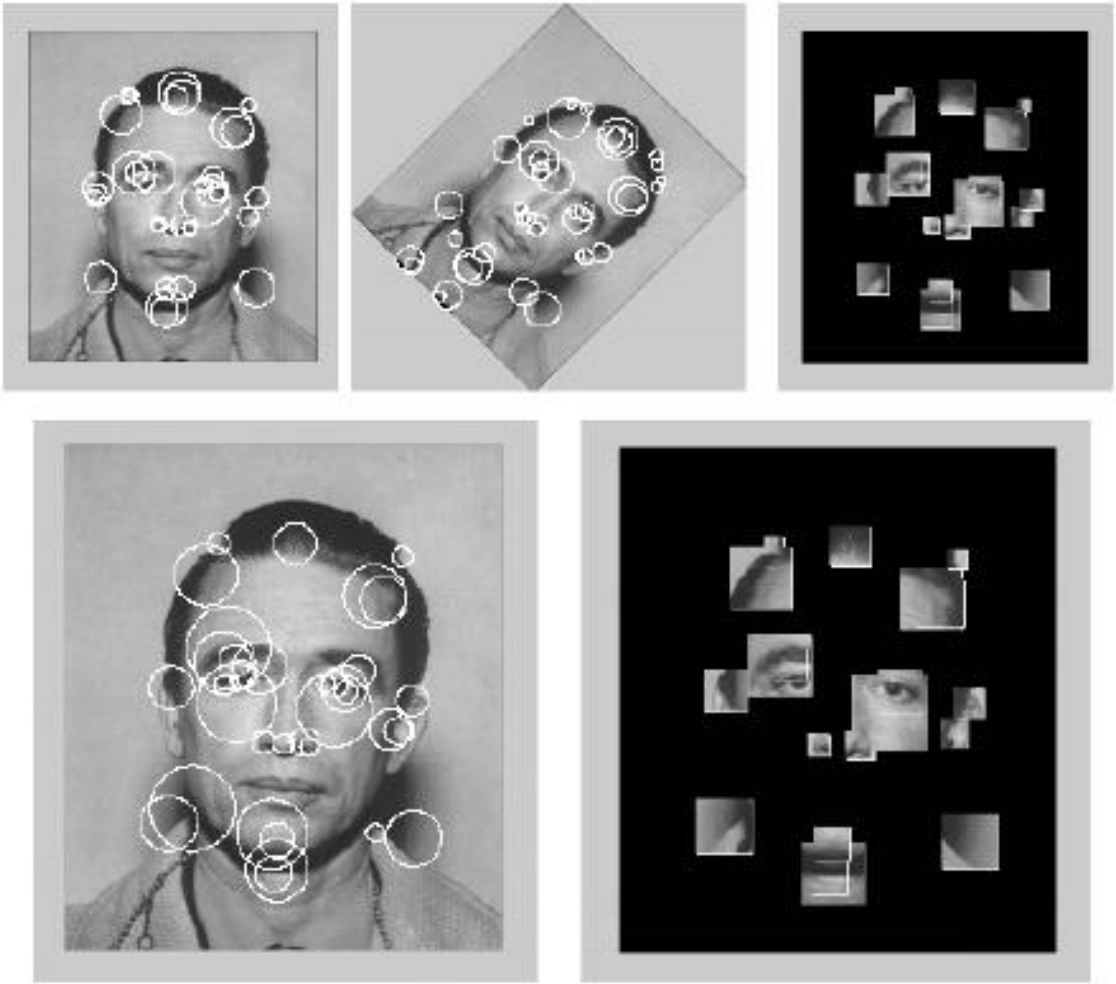
- Apply a global threshold on saliency.
- Choose the highest salient points (50% works well).
- Find the K nearest neighbors (K=8 preset)
- Check variance at center points with these neighbors.
- Accept if far enough away from existant clusters and variance small enough.
- Represent with mean scale and spatial location of the K points
- Repeat with next highest salient point

More examples

Robustness Claims

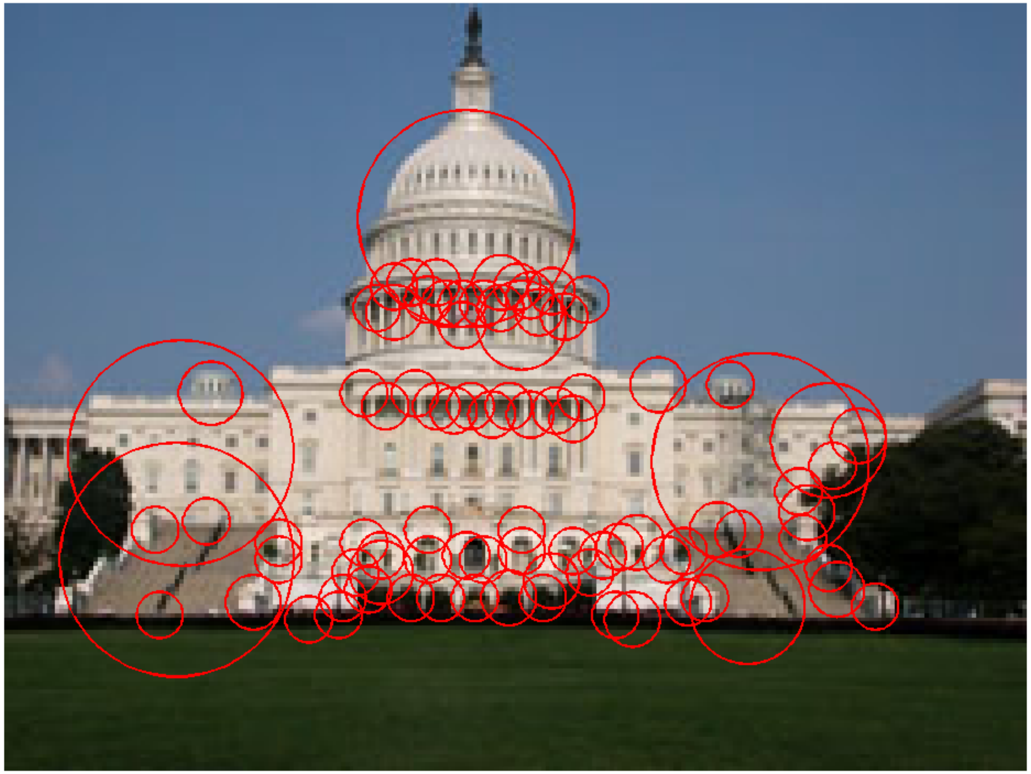
- scale invariant (chooses its scale)
- rotation invariant (uses circular regions and histograms)
- somewhat illumination invariant (why?)
- not affine invariant (able to handle small changes in viewpoint)

More Examples



Temple

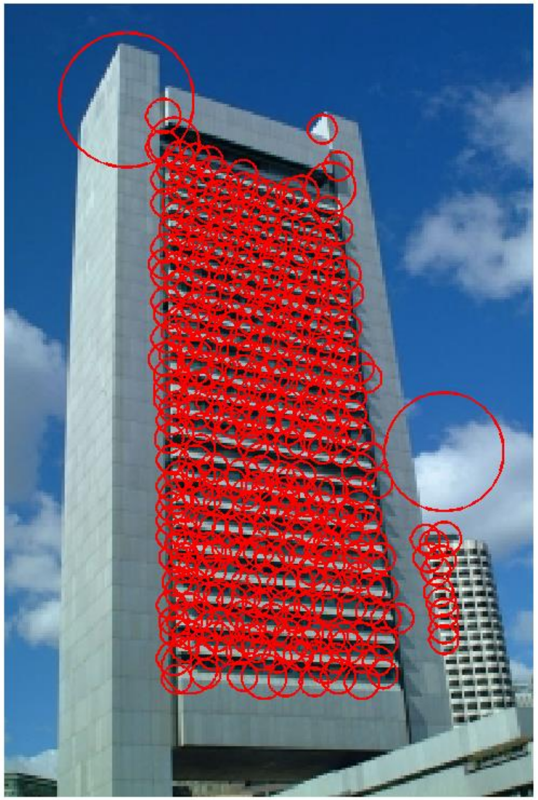
Capitol



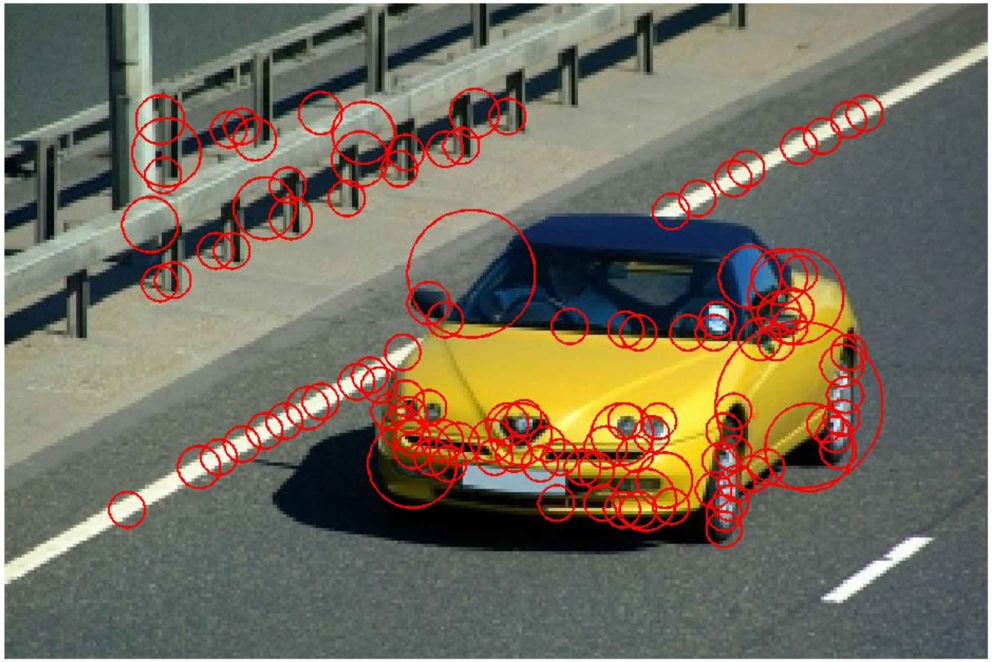
Houses and Boats

Houses and Boats

Sky Scraper

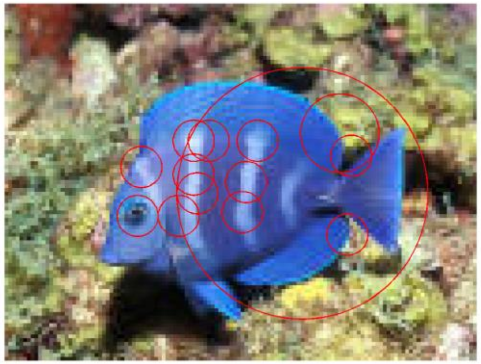


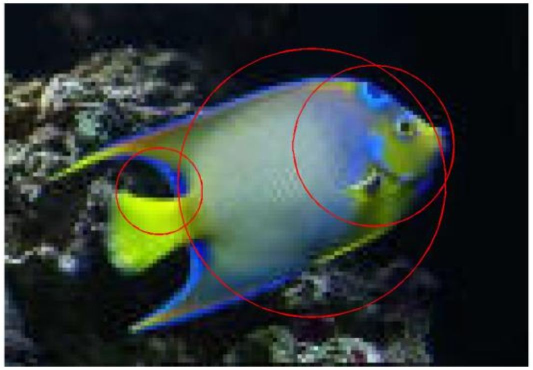
Car



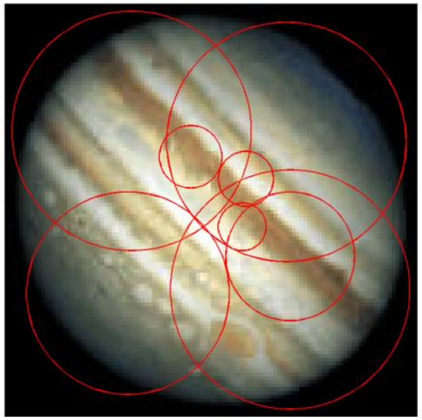
Trucks

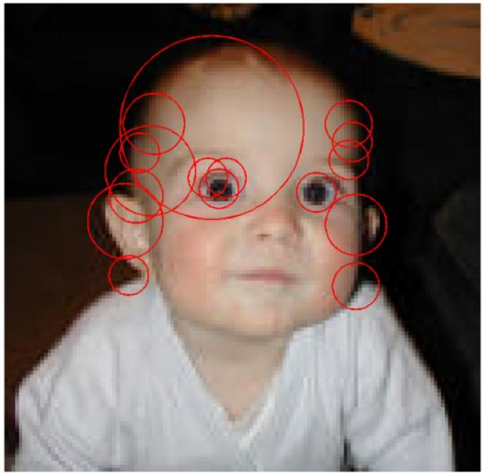
Fish



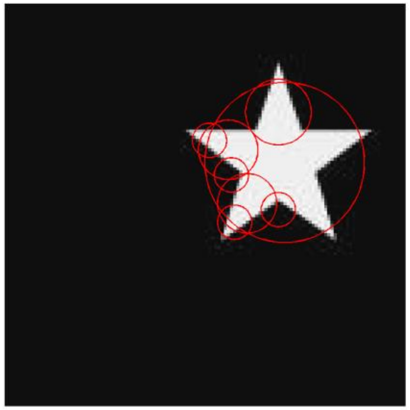


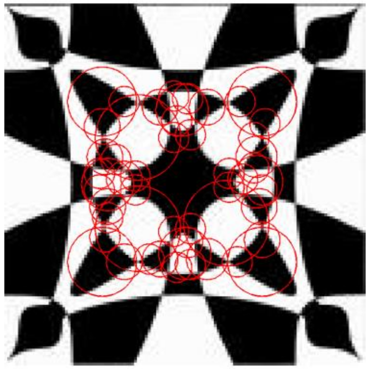
Other

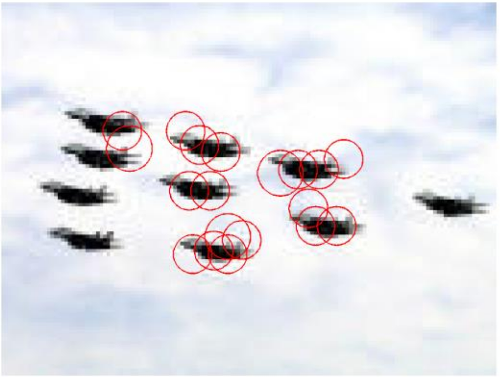




Symmetry and More







Benefits

- General feature: not tied to any specific object
- Can be used to detect rather complex objects that are not all one color
- Location invariant, rotation invariant
- Selects relevant scale, so scale invariant
- What else is good?
- Anything bad?

Object Recognition with Interest Operators

- Object recognition started with line segments.

  - Roberts recognized objects from line segments and junctions.

  - This led to systems that extracted linear features.

  - CAD-model-based vision works well for industrial.

- An “appearance-based approach” was first developed for face recognition and later generalized up to a point.

- The interest operators have led to a new kind of recognition by “parts” that can handle a variety of objects that were previously difficult or impossible.

Object Class Recognition by Unsupervised Scale-Invariant Learning

R. Fergus, P. Perona, and A. Zisserman
Oxford University and Caltech

CVPR 2003

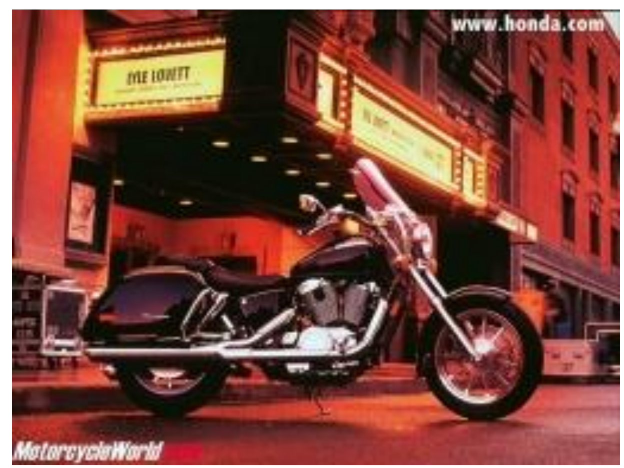
won the best student paper award

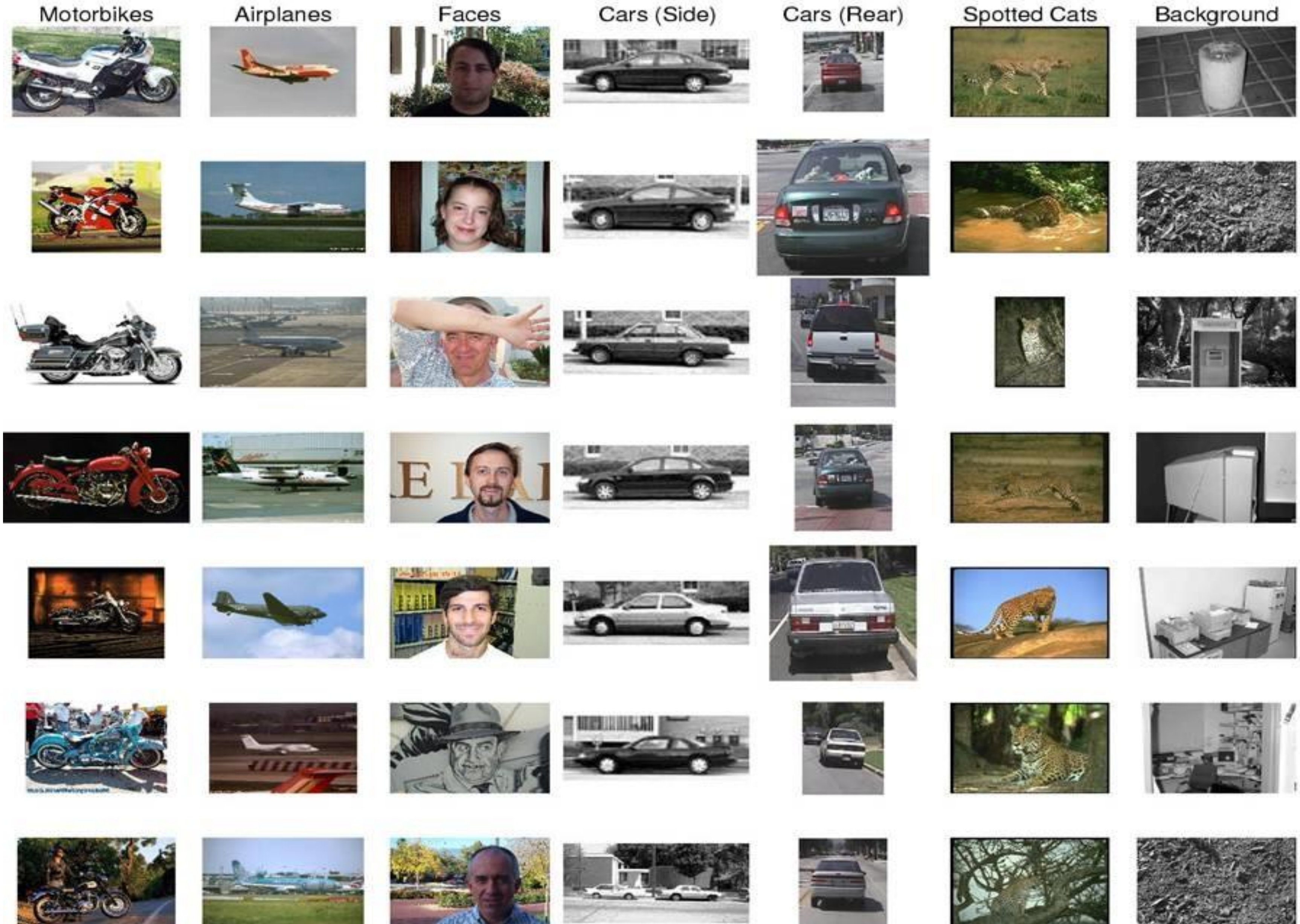
CVPR 2013

won the best 10-year award

Goal:

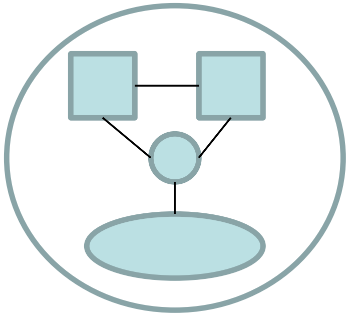
- Enable Computers to Recognize Different Categories of Objects in Images.





Motorbikes
Airplanes
Faces
Cars (Side)
Cars (Rear)
Spotted Cats
Background

Approach

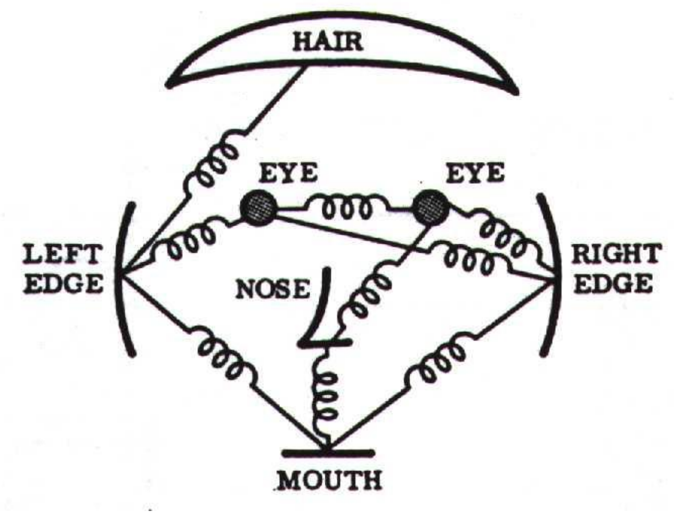


- An object is a constellation of parts (from Burl, Weber and Perona, 1998).
- The parts are detected by an interest operator (Kadir's).
- The parts can be recognized by appearance.
- Objects may vary greatly in scale.
- The constellation of parts for a given object is learned from training images

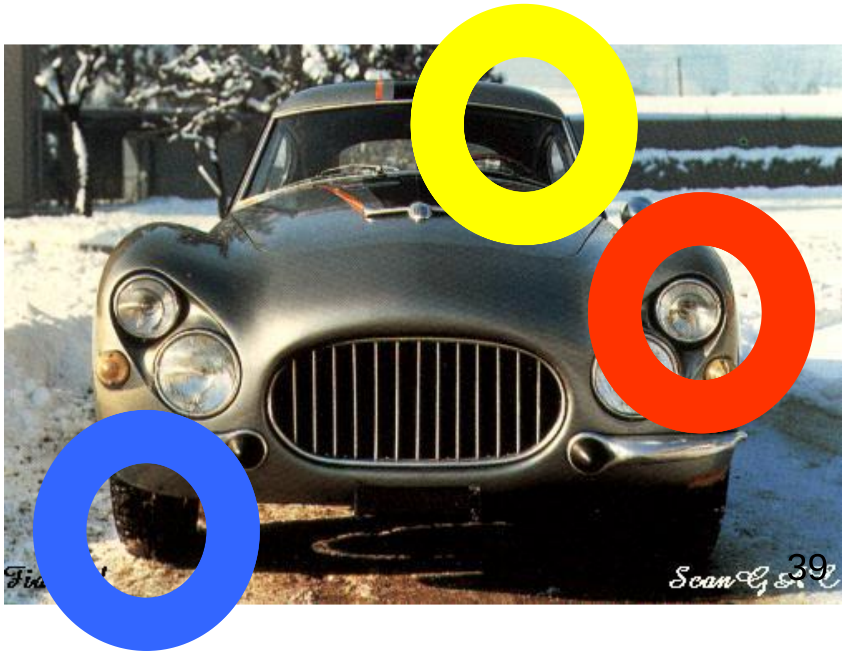
Components

- Model
  - Generative Probabilistic Model including Location, Scale, and Appearance of Parts
- Learning
  - Estimate Parameters Via EM Algorithm
- Recognition
  - Evaluate Image Using Model and Threshold

Model: Constellation Of Parts



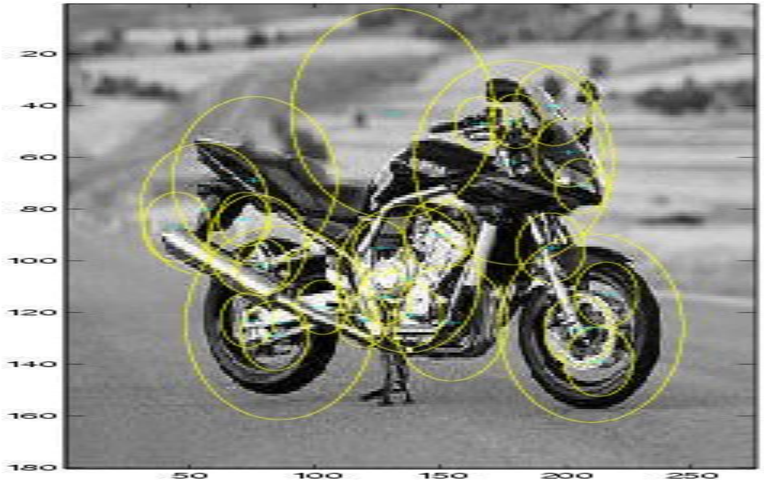
Fischler & Elschlager, 1973

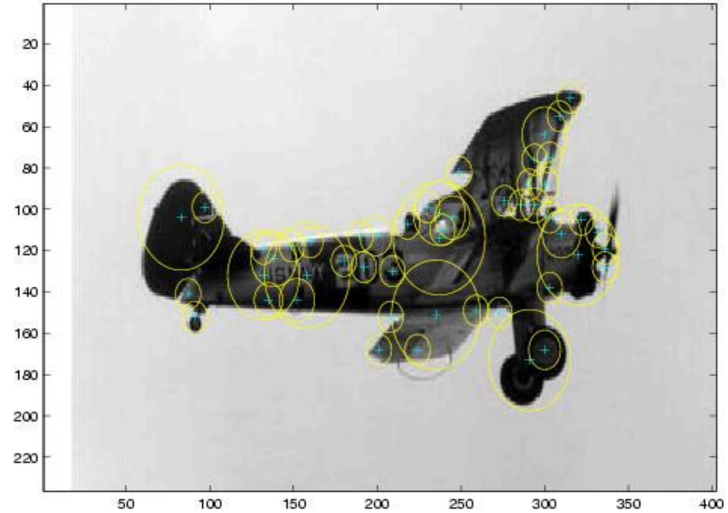


Yuille, 91
Brunelli & Poggio, 93
Lades, v.d. Malsburg et al. 93
Cootes, Lanitis, Taylor et al. 95
Amit & Geman, 95, 99
Perona et al. 95, 96, 98, 00

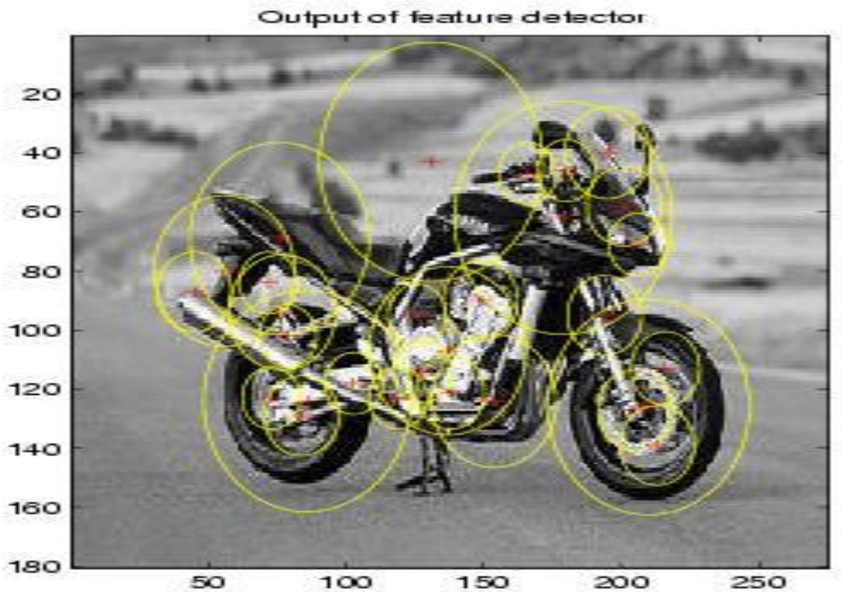
Parts Selected by Interest Operator

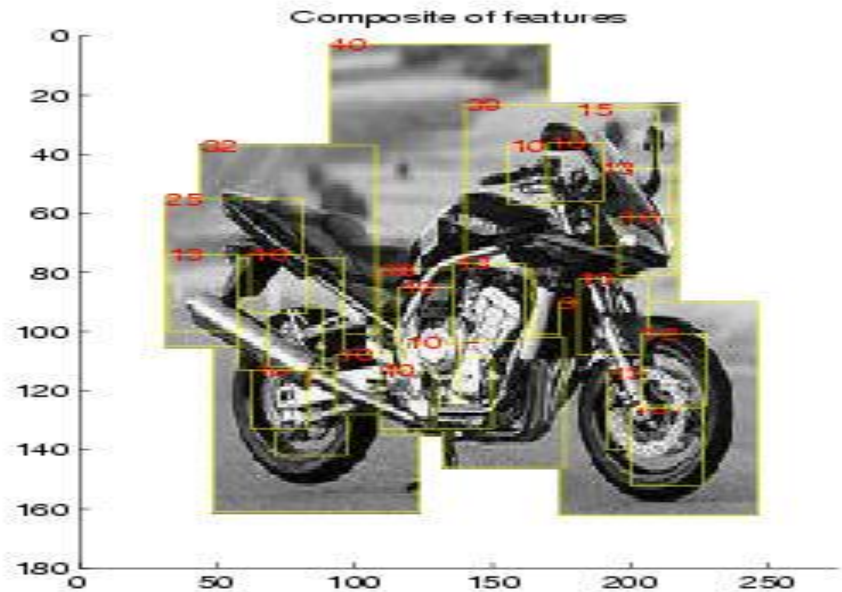
Kadir and Brady's Interest Operator.
Finds Maxima in Entropy Over Scale and Location

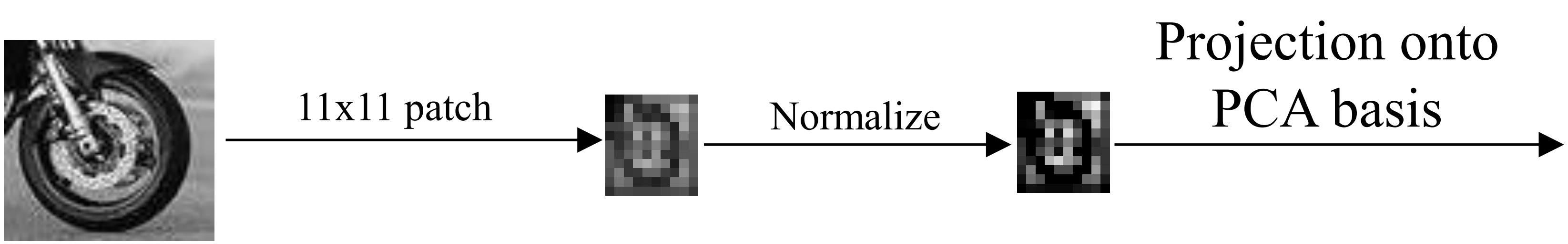


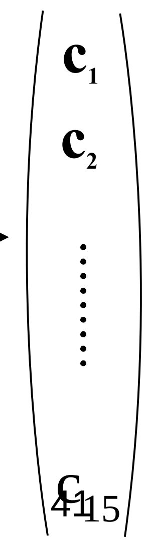


Representation of Appearance









121 dimensions was too big, so they used PCA to reduce to 10-15.

Learning a Model

- An object class is represented by a generative model with P parts and a set of parameters θ.

- Once the model has been learned, a decision procedure must determine if a new image contains an instance of the object class or not.

- Suppose the new image has N interesting features with locations X, scales S and appearances A.

Probabilistic Model

$$p(\mathbf{X}, \mathbf{S}, \mathbf{A} | \theta) = \sum_{\mathbf{h} \in H} p(\mathbf{X}, \mathbf{S}, \mathbf{A}, \mathbf{h} | \theta) =$$

$$\sum_{\mathbf{h} \in H} \underbrace{p(\mathbf{A}|\mathbf{X}, \mathbf{S}, \mathbf{h}, \theta)}_{Appearance} \underbrace{p(\mathbf{X}|\mathbf{S}, \mathbf{h}, \theta)}_{Shape} \underbrace{p(\mathbf{S}|\mathbf{h}, \theta)}_{Rel.\ Scale} \underbrace{p(\mathbf{h}|\theta)}_{Other}$$

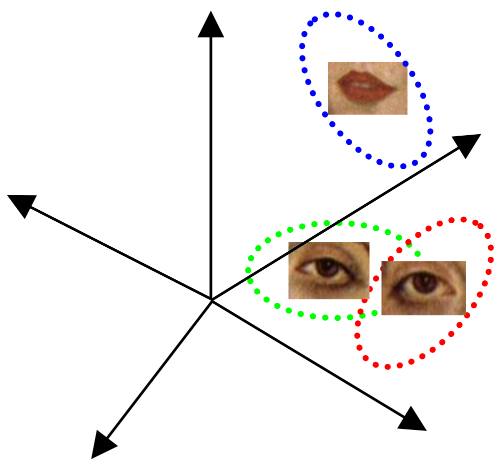
- X is a description of the shape of the object (in terms of locations of parts)
- S is a description of the scale of the object
- A is a description of the appearance of the object
- θ is the (maximum likelihood value of) the parameters of the object
- h is a hypothesis: a set of parts in the image that might be the parts of the object
- H is the set of all possible hypotheses for that object in that image.
- For N features in the image and P parts in the object, its size is $O(N^P)$

Appearance

The appearance (A) of each part p has a Gaussian density with mean c_p and covariance V_P.

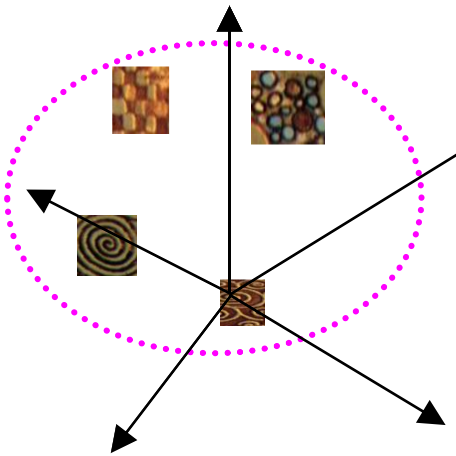
Background model has mean c_{bg} and covariance V_{bg}.

Gaussian Part Appearance PDF



Object

Gausian Appearance PDF

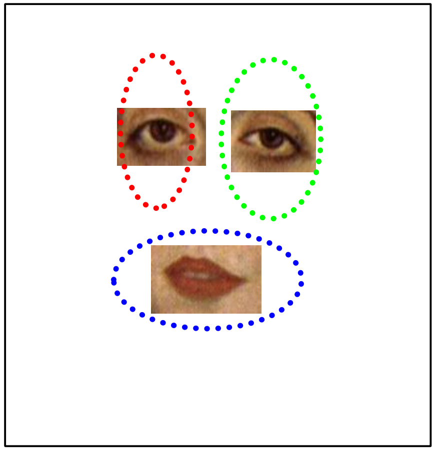


Background

Shape as Location

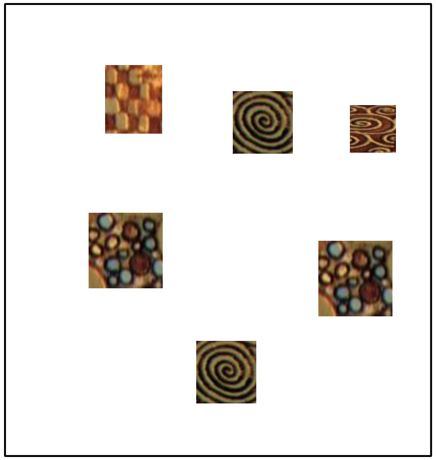
Object shape is represented by a joint Gaussian density of the locations (X) of features within a hypothesis transformed into a scale-invariant space.

Gaussian Shape PDF



Object

Uniform Shape PDF

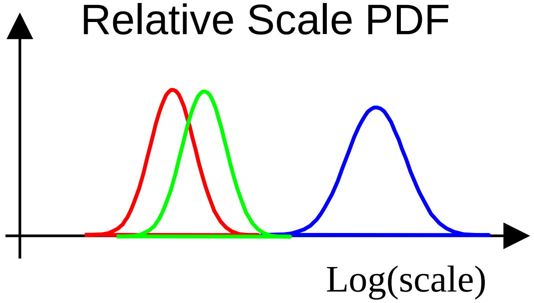


Background

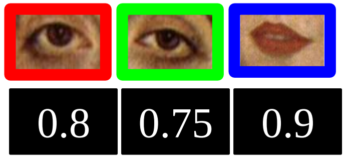
Scale

The relative scale of each part is modeled by a Gaussian density with mean t_p and covariance U_p.

Gaussian
Relative Scale PDF



Prob. of detection



Occlusion and Part Statistics

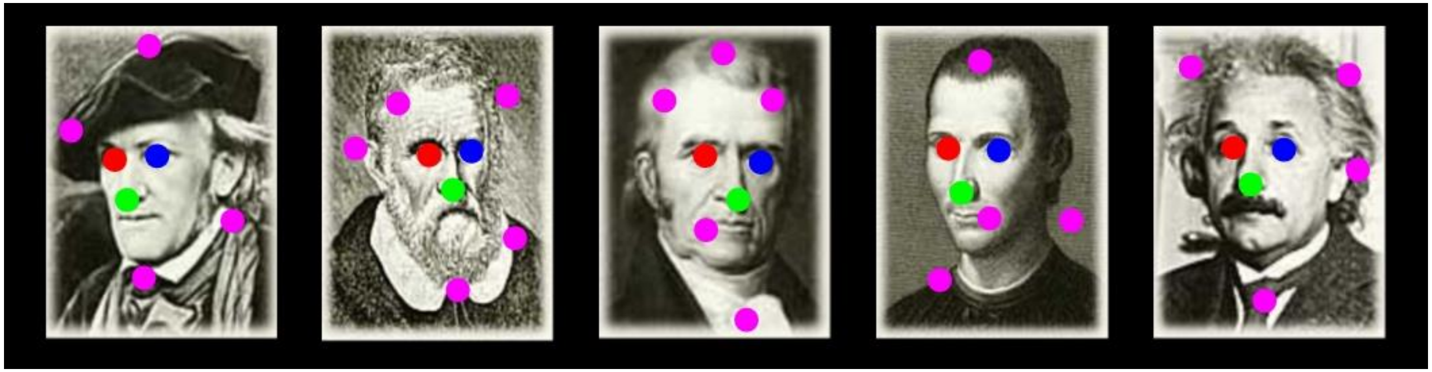
This was very complicated and turned out to not work well and not be necessary, in both Fergus's work and other subsequent works.

Learning

- Train Model Parameters Using EM:
  - Optimize Parameters
  - Optimize Assignments
  - Repeat Until Convergence

$$\theta = \{\underbrace{\mu, \Sigma}_{\text{location}}, \underbrace{\mathbf{c}, V}_{\text{appearance}}, \underbrace{M, p(\mathbf{d}|\theta)}_{\text{occlusion}}, \underbrace{t, U}_{\text{scale}}\}$$

$$\hat{\theta}_{ML} = \underset{\theta}{arg\,max}\; p(\mathbf{X}, \mathbf{S}, \mathbf{A} |\, \theta)$$



Recognition

Make this likelihood ratio:

$$
\begin{aligned}
R &= \frac{p(\text{Object}|\mathbf{X},\mathbf{S},\mathbf{A})}{p(\text{No object}|\mathbf{X},\mathbf{S},\mathbf{A})} \\
&= \frac{p(\mathbf{X},\mathbf{S},\mathbf{A}|\text{Object})\; p(\text{Object})}{p(\mathbf{X},\mathbf{S},\mathbf{A}|\text{No object})\; p(\text{No object})} \\
&\approx \frac{p(\mathbf{X},\mathbf{S},\mathbf{A}|\,\theta)\; p(\text{Object})}{p(\mathbf{X},\mathbf{S},\mathbf{A}|\theta_{bg})\; p(\text{No object})}
\end{aligned}
$$

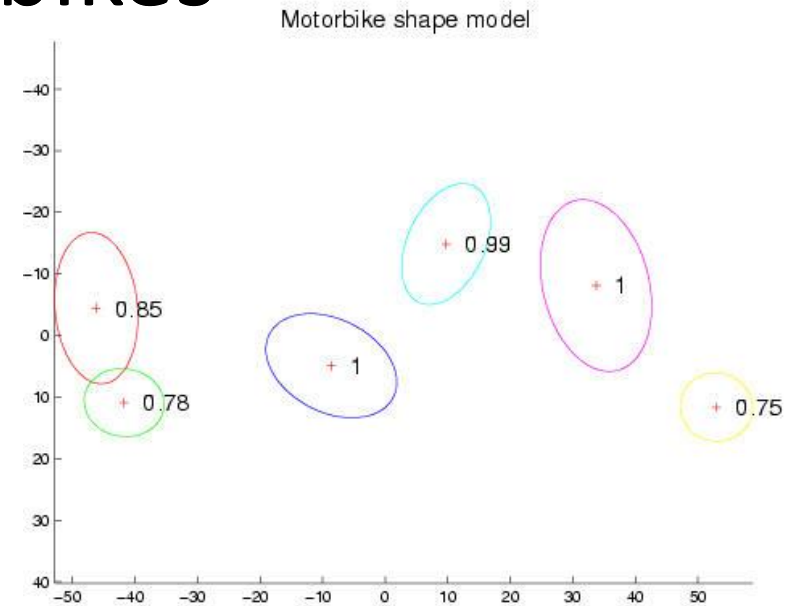
greater than a threshold.

RESULTS

- Initially tested on the Caltech-4 data set
  - motorbikes
  - faces
  - airplanes
  - cars
- Now there is a much bigger data set: the Caltech-101 http://www.vision.caltech.edu/archive.html

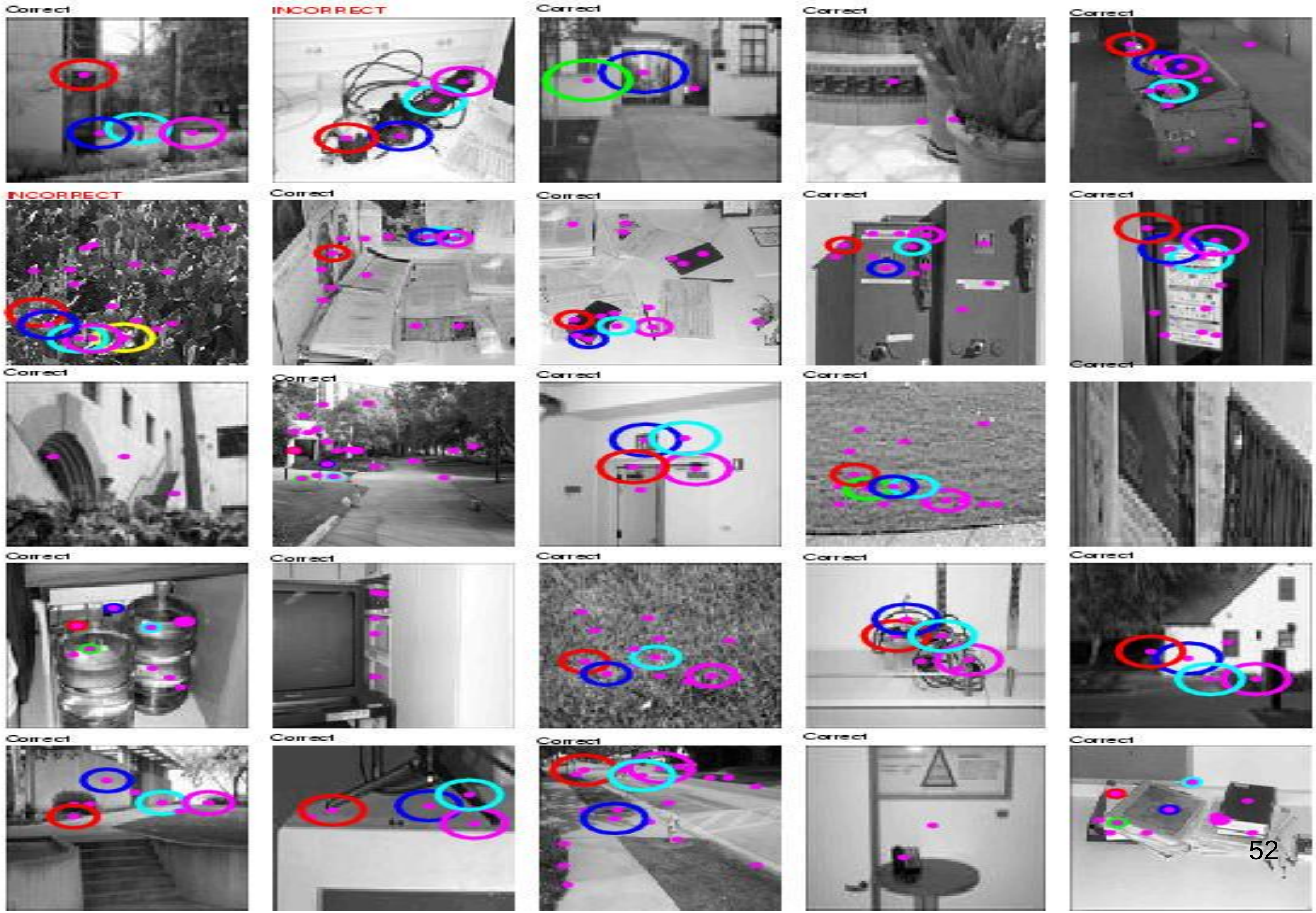
Motorbikes

Equal error rate: 7.5%



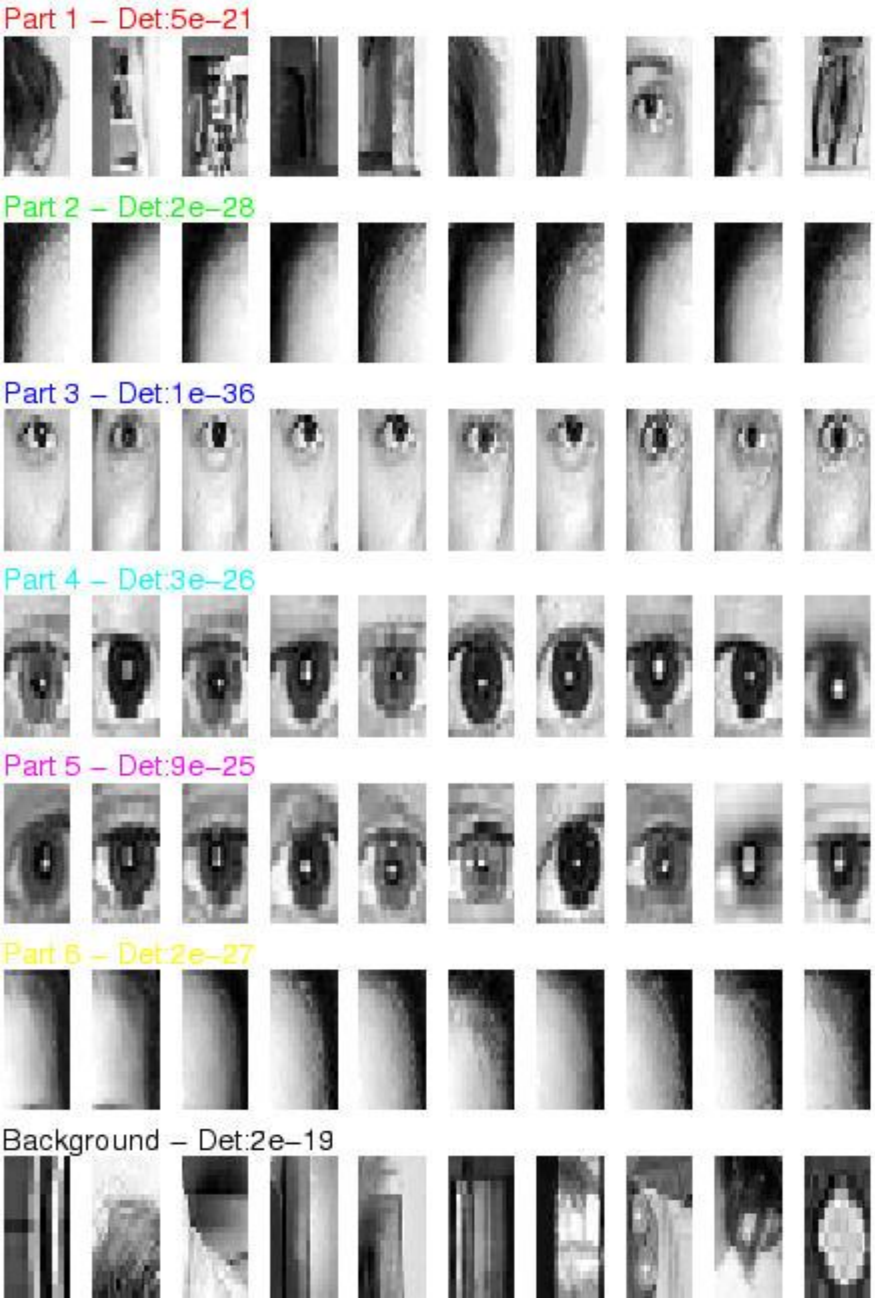
Background Images

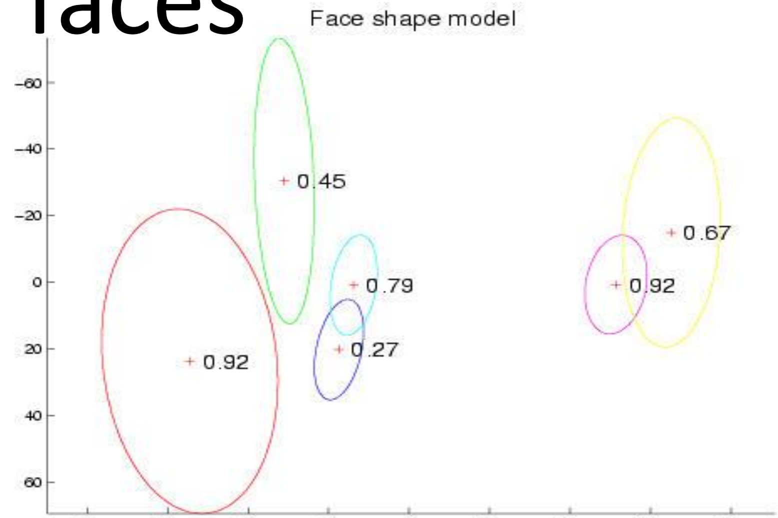
It learns that these are NOT motorbikes.



Frontal faces

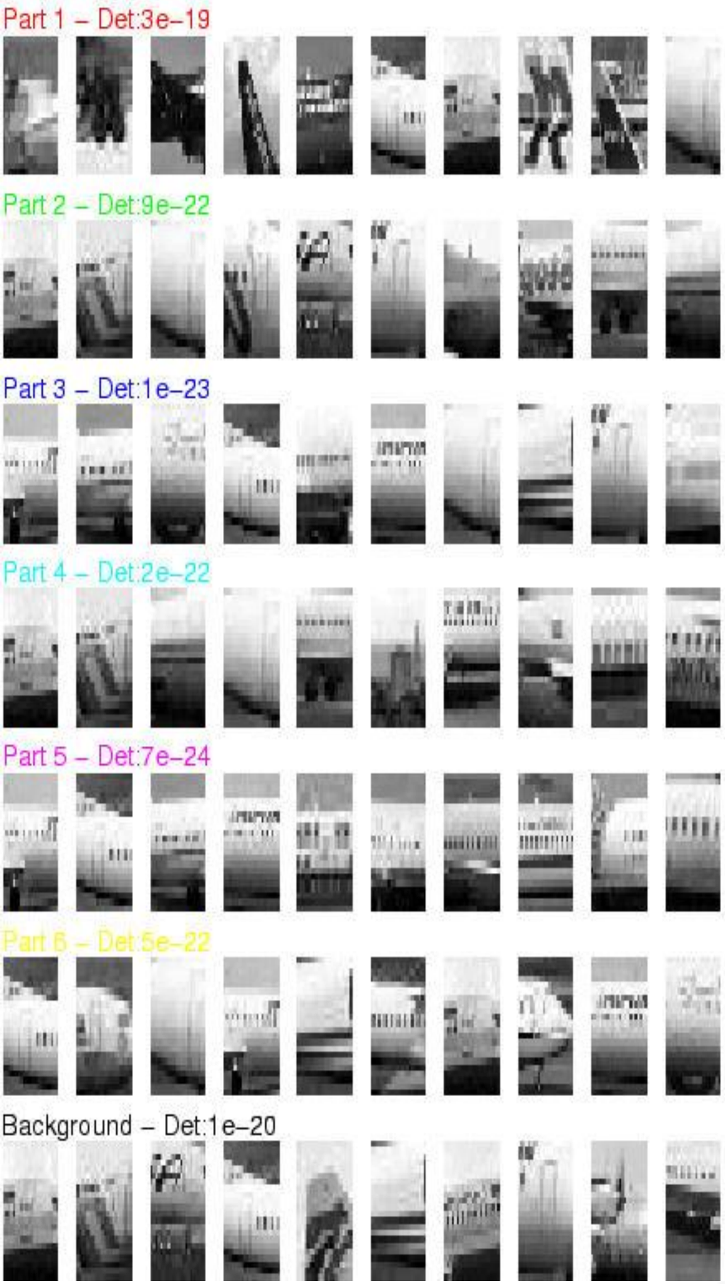
Equal error rate: 4.6%

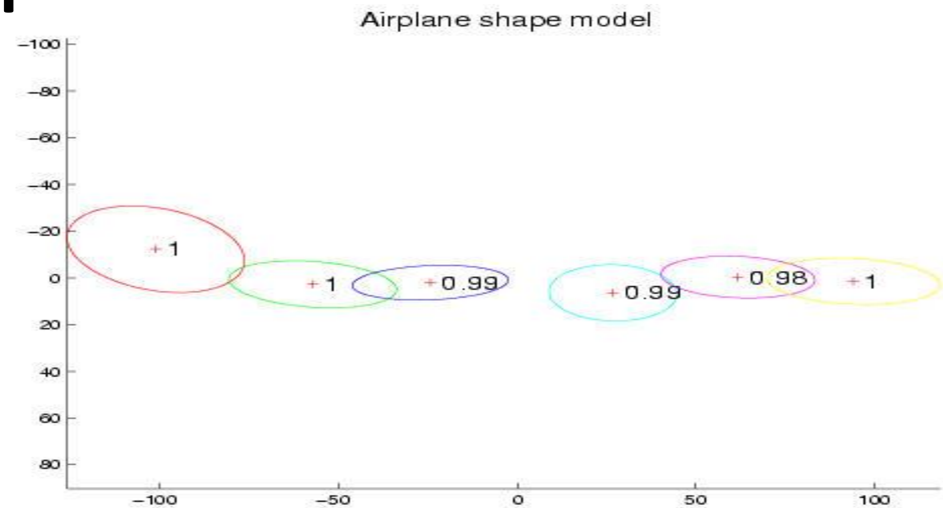


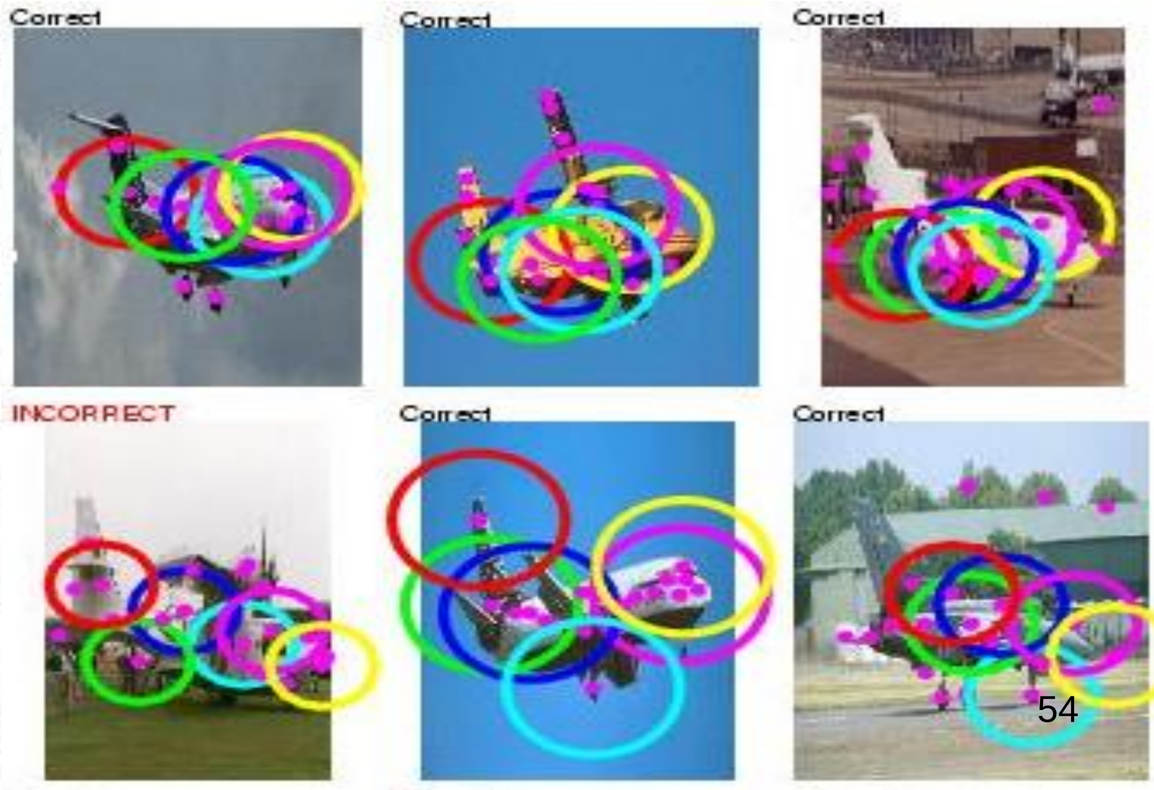


Equal error rate: 9.8%

Airplanes

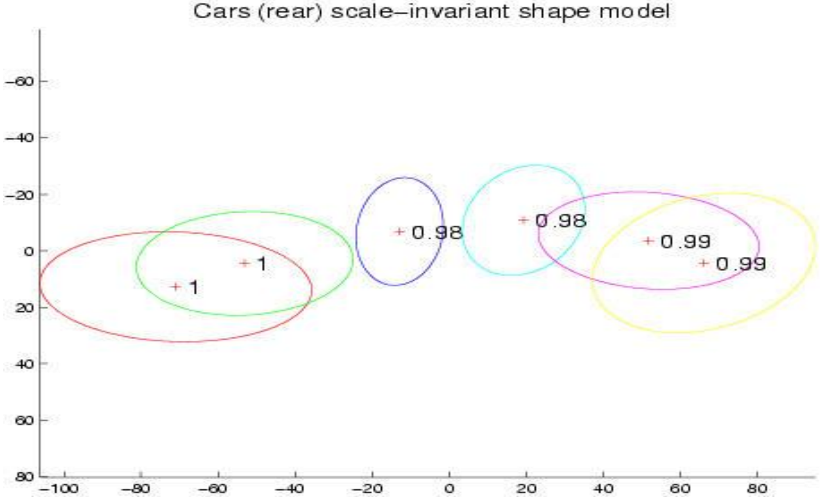






Scale-Invariant Cars

Equal error rate: 9.7%



Accuracy

Initial Pre-Scaled Experiments

Dataset	Ours	Others	Ref.
Motorbikes	92.5	84	[17]
Faces	96.4	94	[19]
Airplanes	90.2	68	[17]
Cars(Side)	88.5	79	[1]

Early Data Set: The CalTech 4

Available Today

- CalTech 101 and Caltech 256
- ImageNet
- Pascal VOC dataset
- CIFAR-10
- MS Coco
- Cityscapes

https://analyticsindiamag.com/10-open-datasets-you-can-use-for-computer-vision-projects/