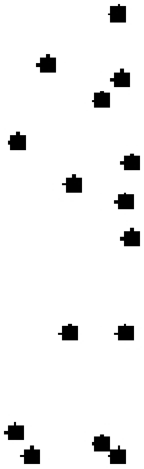
Recognition

CS 554 – Computer Vision

Pinar Duygulu

Bilkent University

What?



Materials

Objects

Actions

Scenes

Adapted from Pietro Perona, Object Recognition Workshop, 2004

Individuals, categories, ...

individual objects

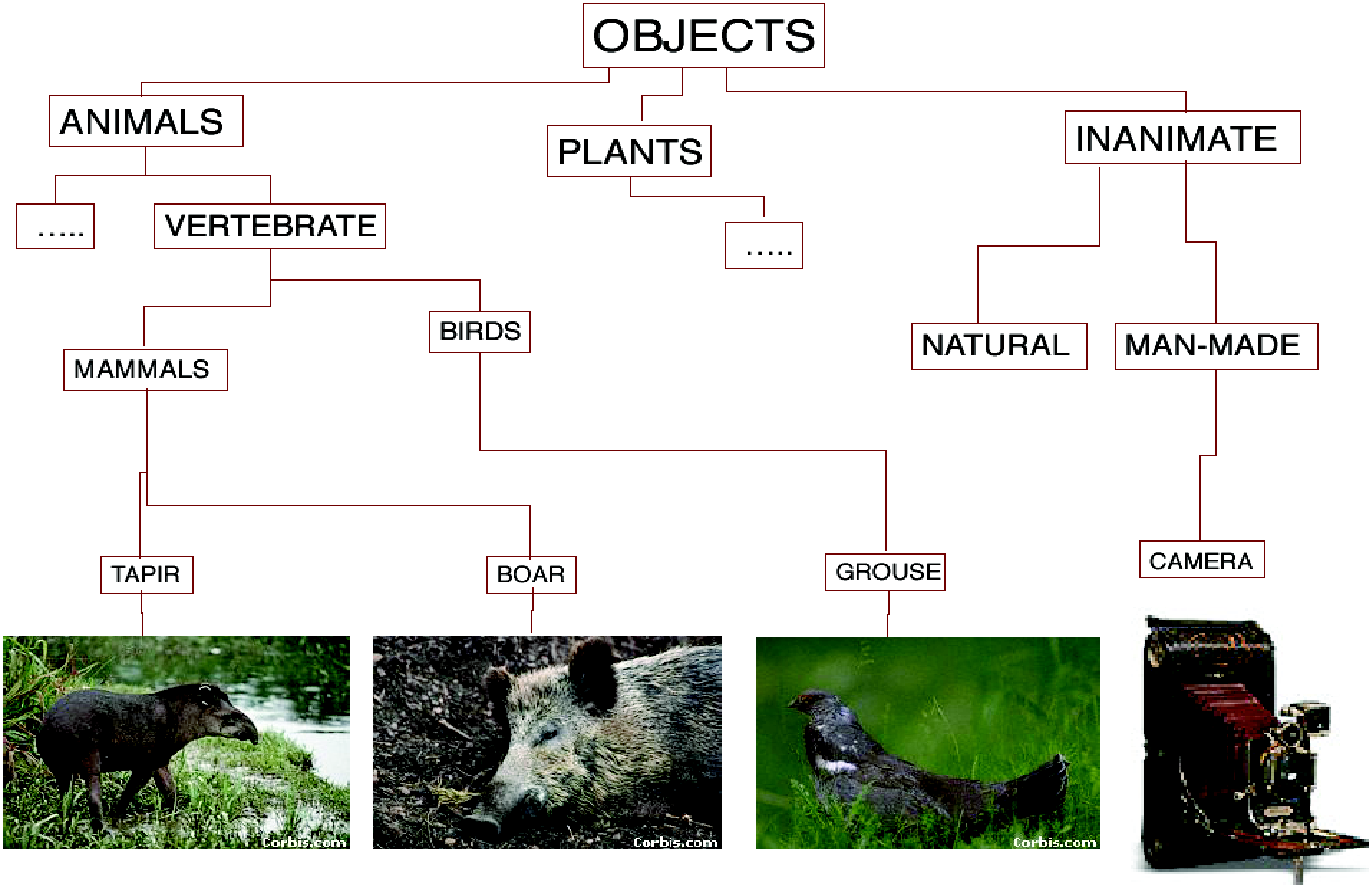
`visual' object classes

`functional' classes?

Adapted from Pietro Perona, Object Recognition Workshop, 2004

How many?

Adapted from Pietro Perona, Object Recognition Workshop, 2004



OBJECTS

ANIMALS

PLANTS

INANIMATE

.....

VERTEBRATE

.....

NATURAL

MAN-MADE

BIRDS

MAMMALS

TAPIR

BOAR

GROUSE

CAMERA

Adapted from Pietro Perona, Object Recognition Workshop, 2004

Tasks

- Verification
- Detection (+Localization)
- Classification / Recognition
- Grouping
- Analogy
- ...

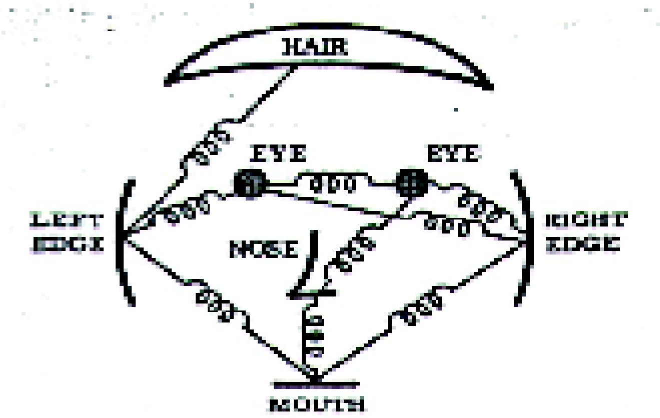
Adapted from Pietro Perona, Object Recognition Workshop, 2004

Issues

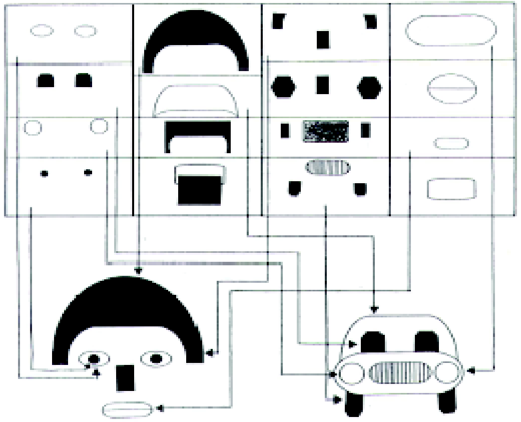
- Representation
- Recognition
- Learning

Adapted from Pietro Perona, Object Recognition Workshop, 2004

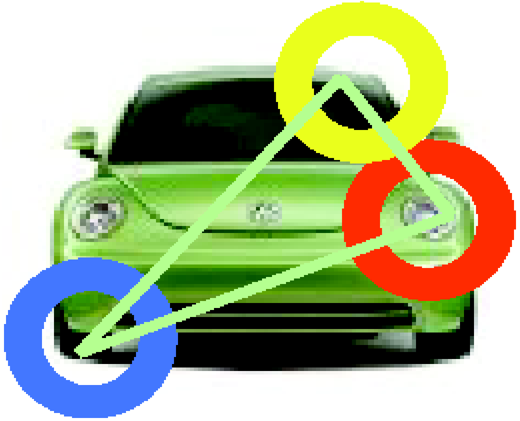
Models: appearance+shape



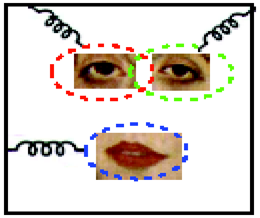
Fischler & Elschlager, 1973



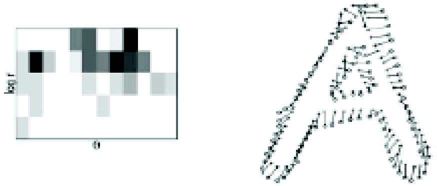
Perrett & Oram, 1993



Perona et al. '95



Schmid '99,
Lowe '99, Moreels '04

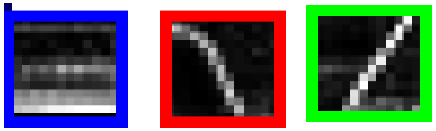


Belongie et al. '02

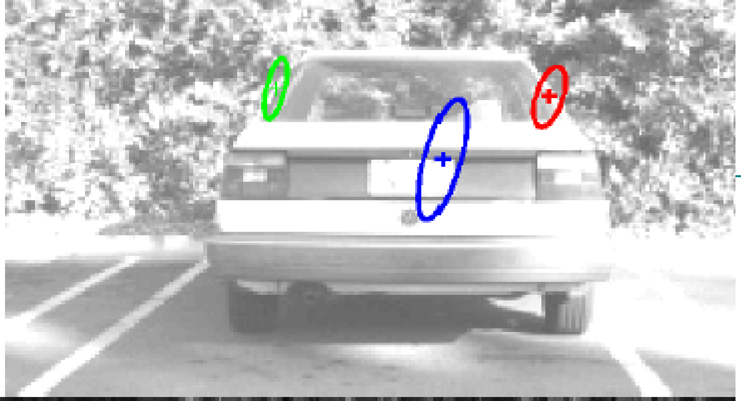
(Interest points)
Local appearance
Shape / deformation
(Clutter)
Correspondence

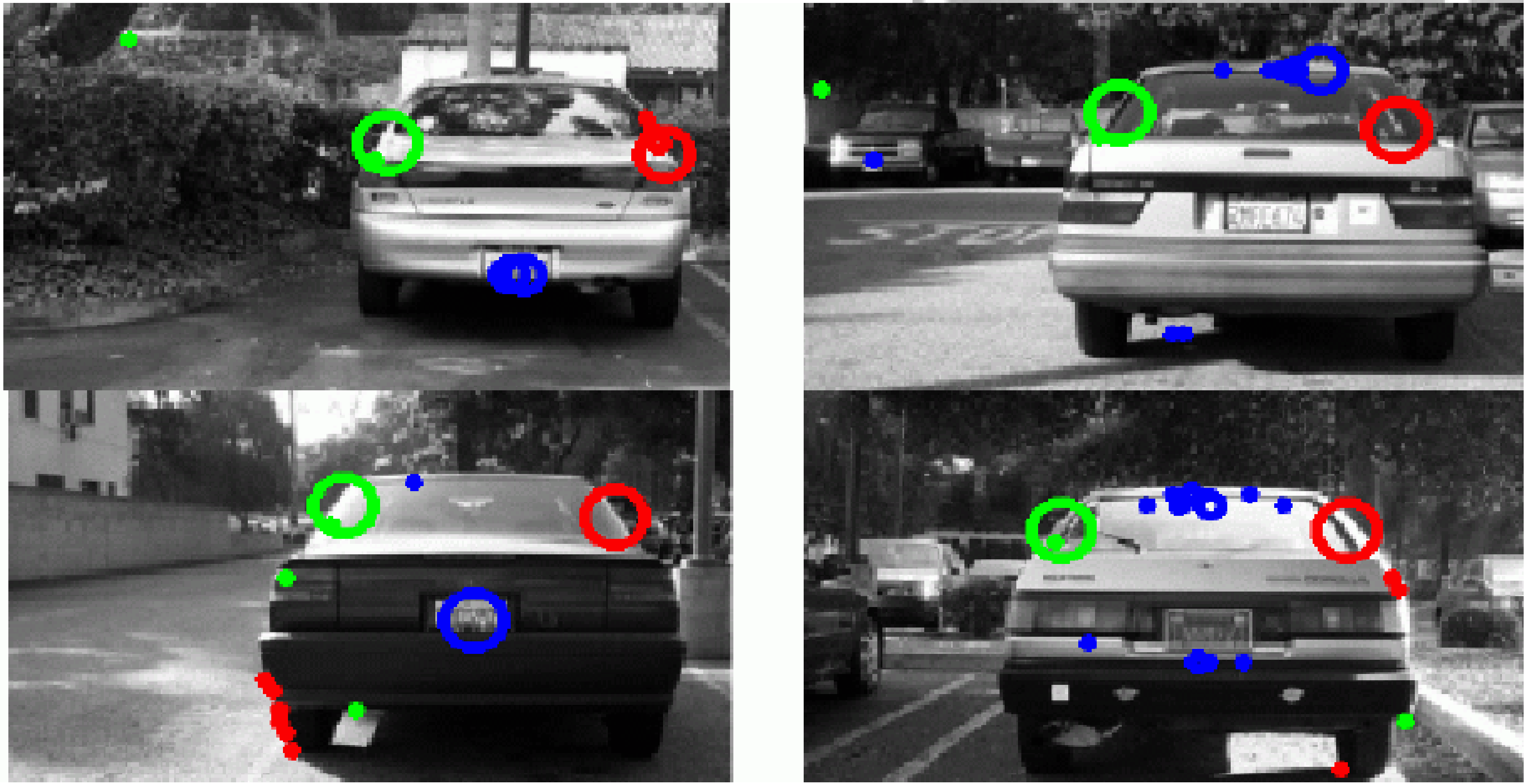
Adapted from Pietro Perona, Object Recognition Workshop, 2004

Correspondence



Weber '00





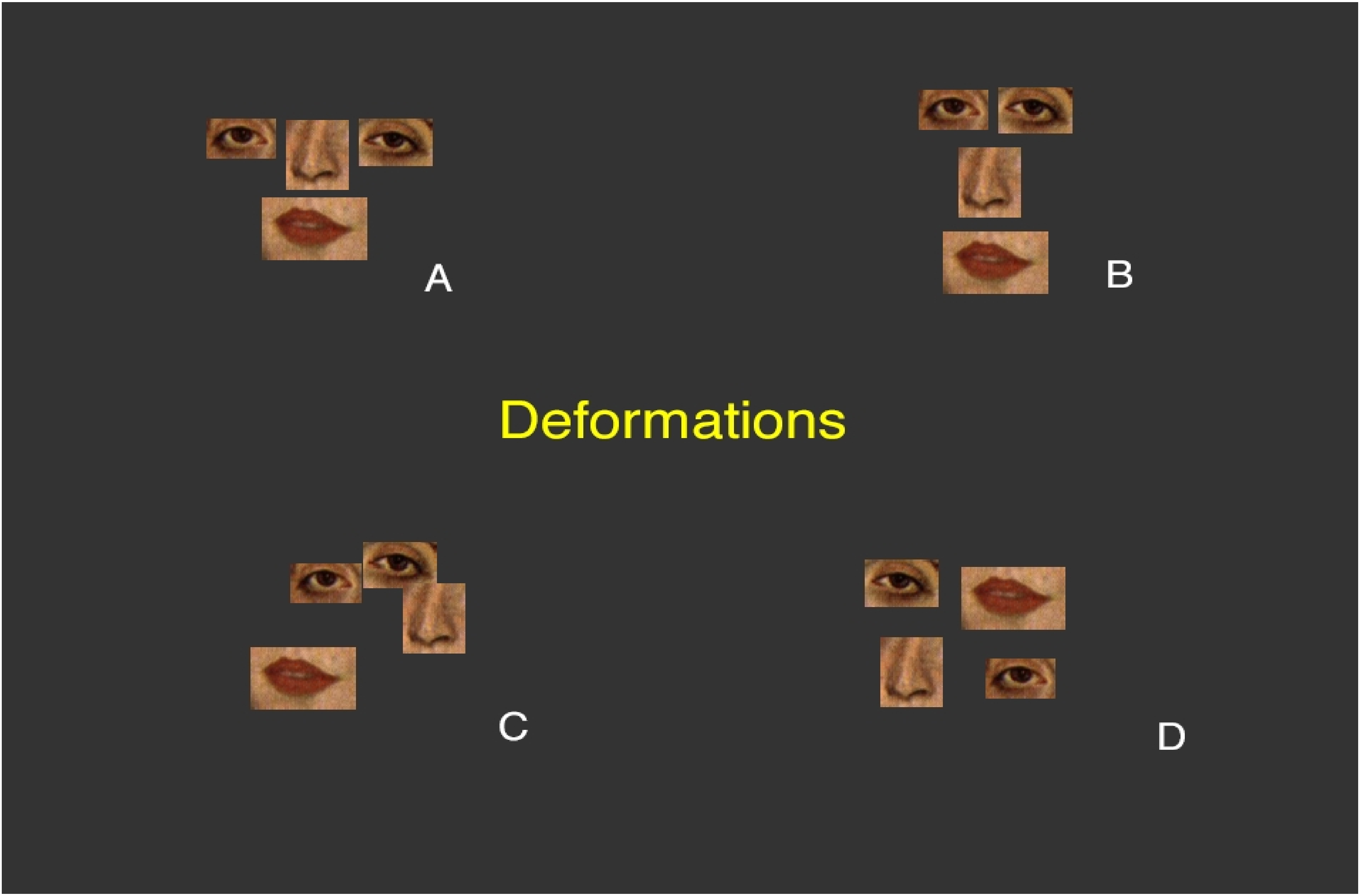
Adapted from Pietro Perona, Object Recognition Workshop, 2004

Occlusion and `unreliable' features

occlusion

Adapted from Pietro Perona, Object Recognition Workshop, 2004

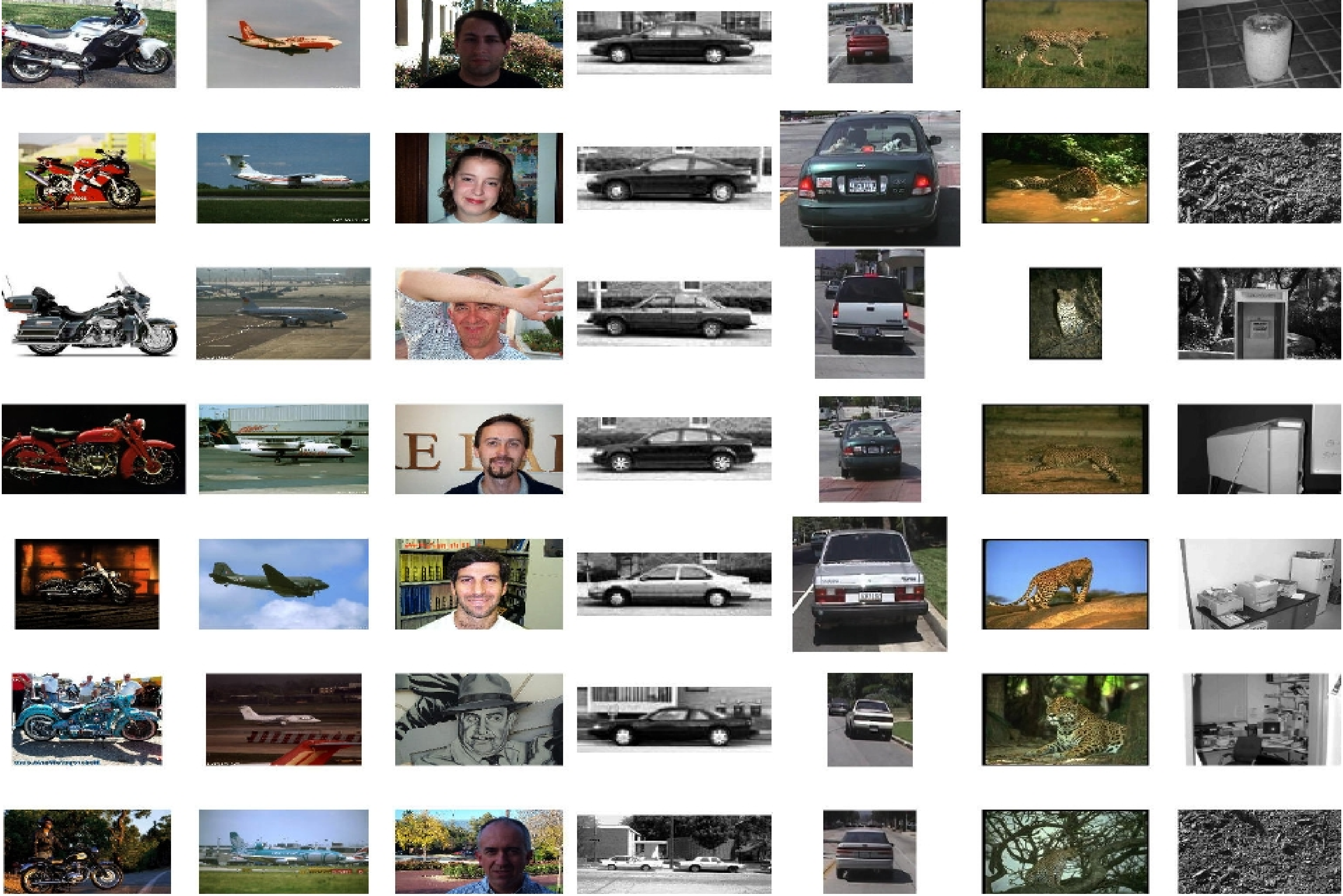
Deformations



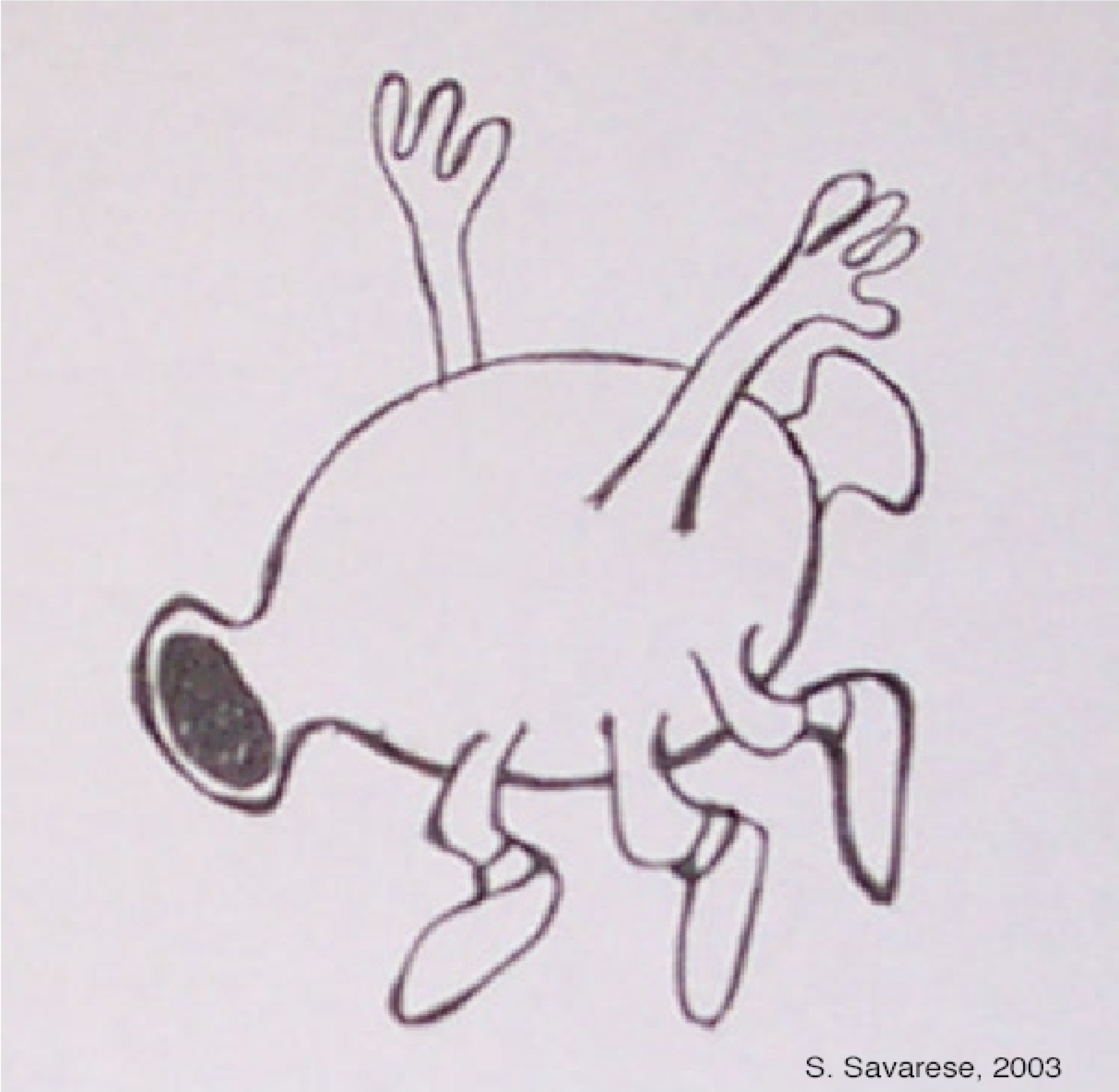
Adapted from Pietro Perona, Object Recognition Workshop, 2004

Clutter

Adapted from Pietro Perona, Object Recognition Workshop, 2004



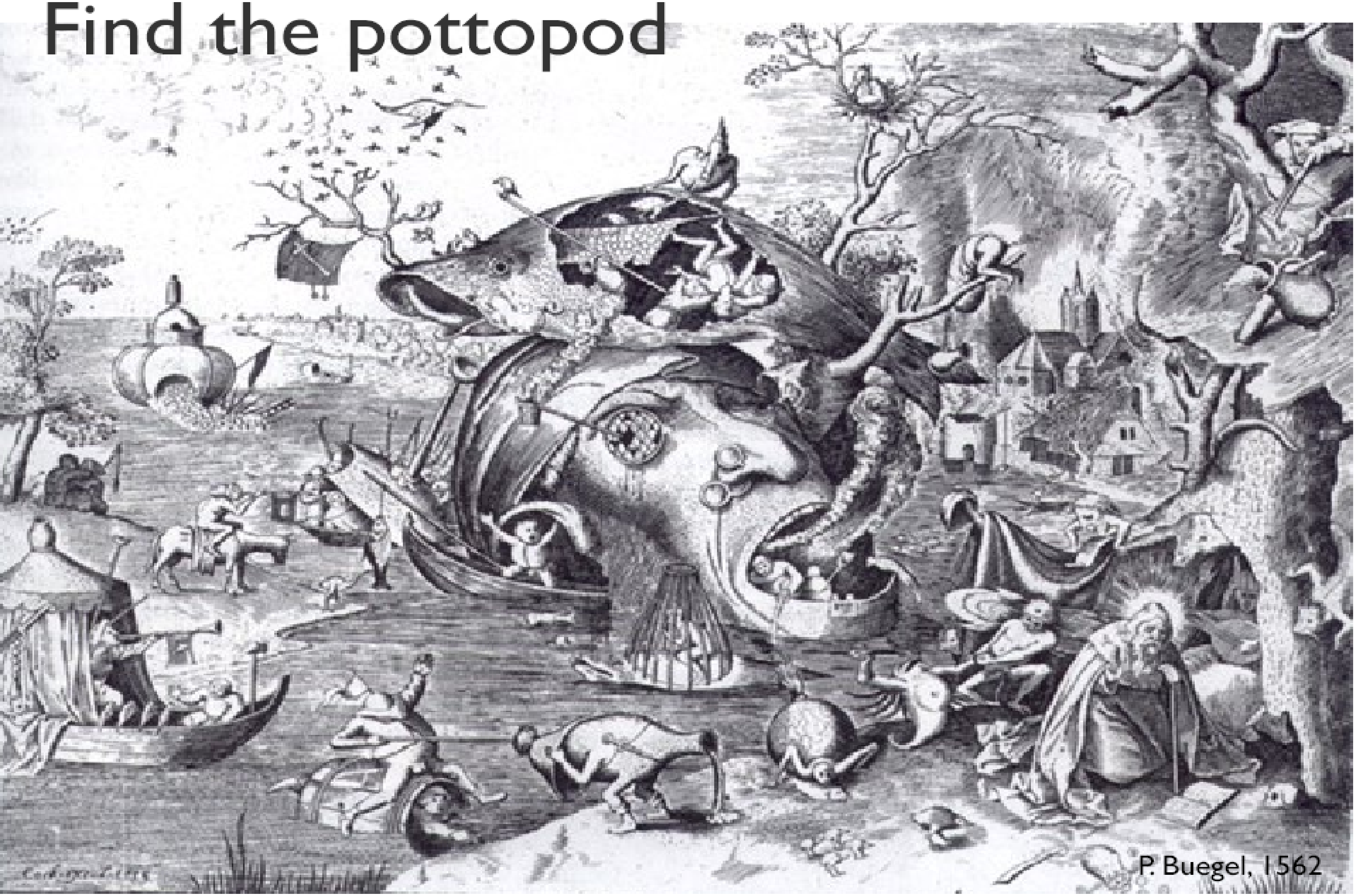
Adapted from Pietro Perona, Object Recognition Workshop, 2004



This is a pottopod

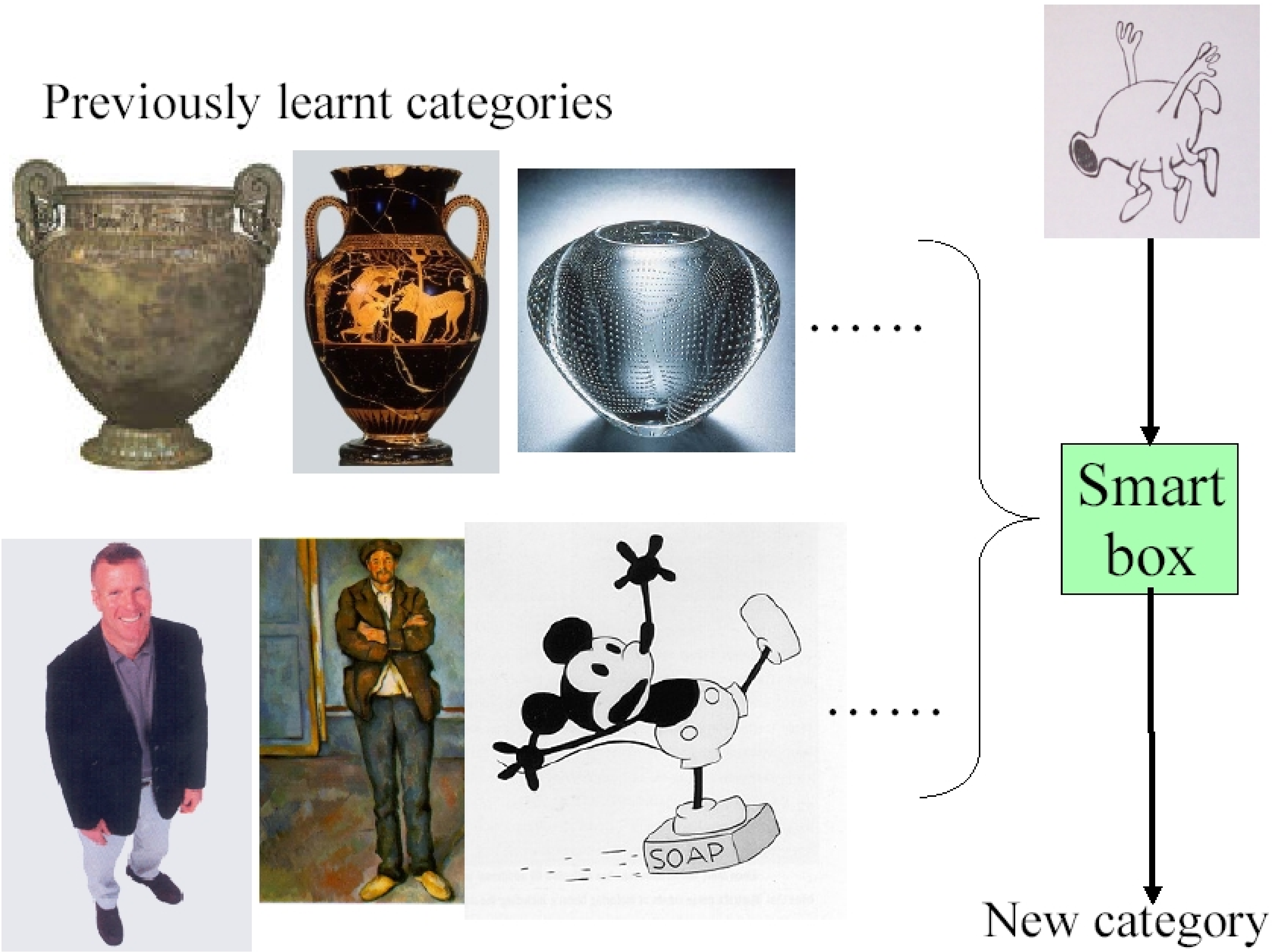
Adapted from Pietro Perona, Object Recognition Workshop, 2004

Find the pottopod



P. Buegel, 1562

Adapted from Pietro Perona, Object Recognition Workshop, 2004



Training example

Previously learnt categories

Smart box

New category

Previously learnt categories

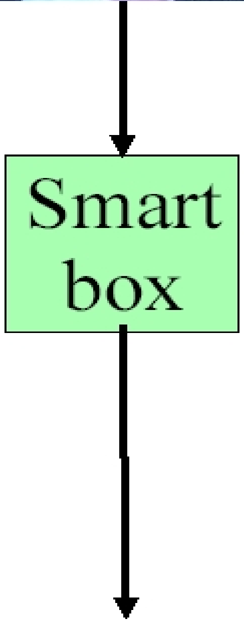
Spotted cats

Airplanes

Motorcycles

FeiFei et al '03

Training example

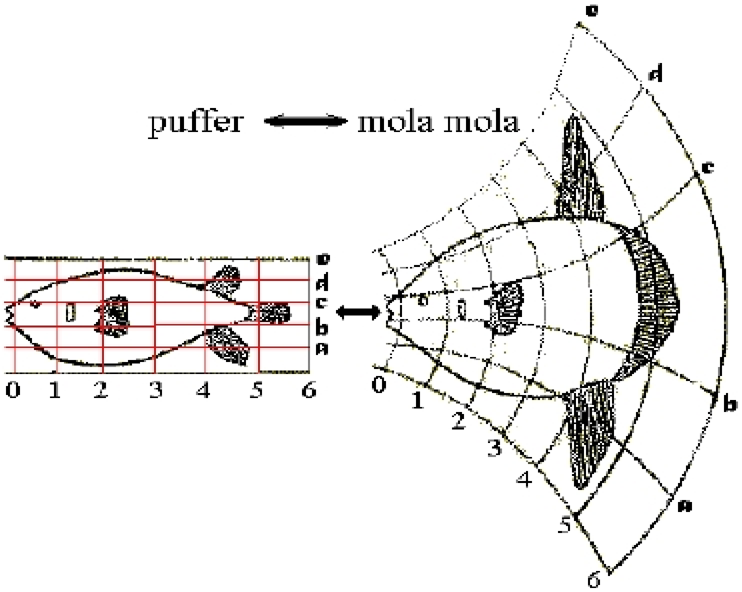


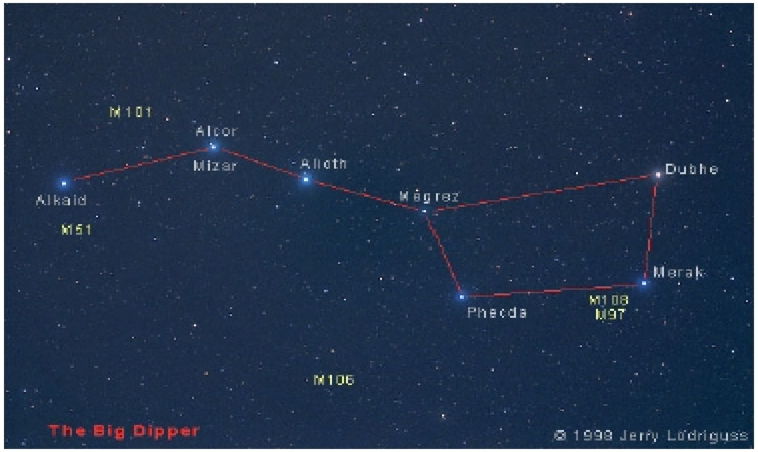
Smart box

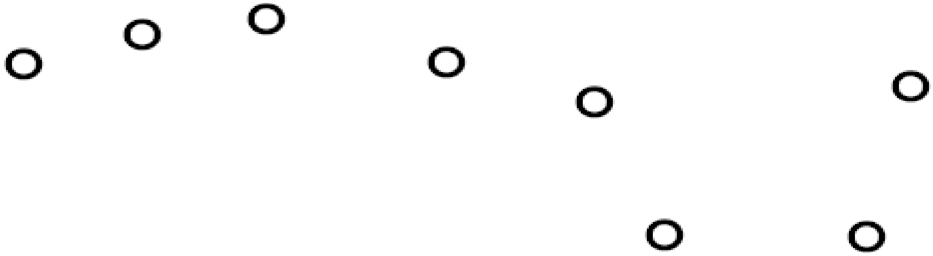
New category

Adapted from Pietro Perona, Object Recognition Workshop, 2004

Priors on geometry

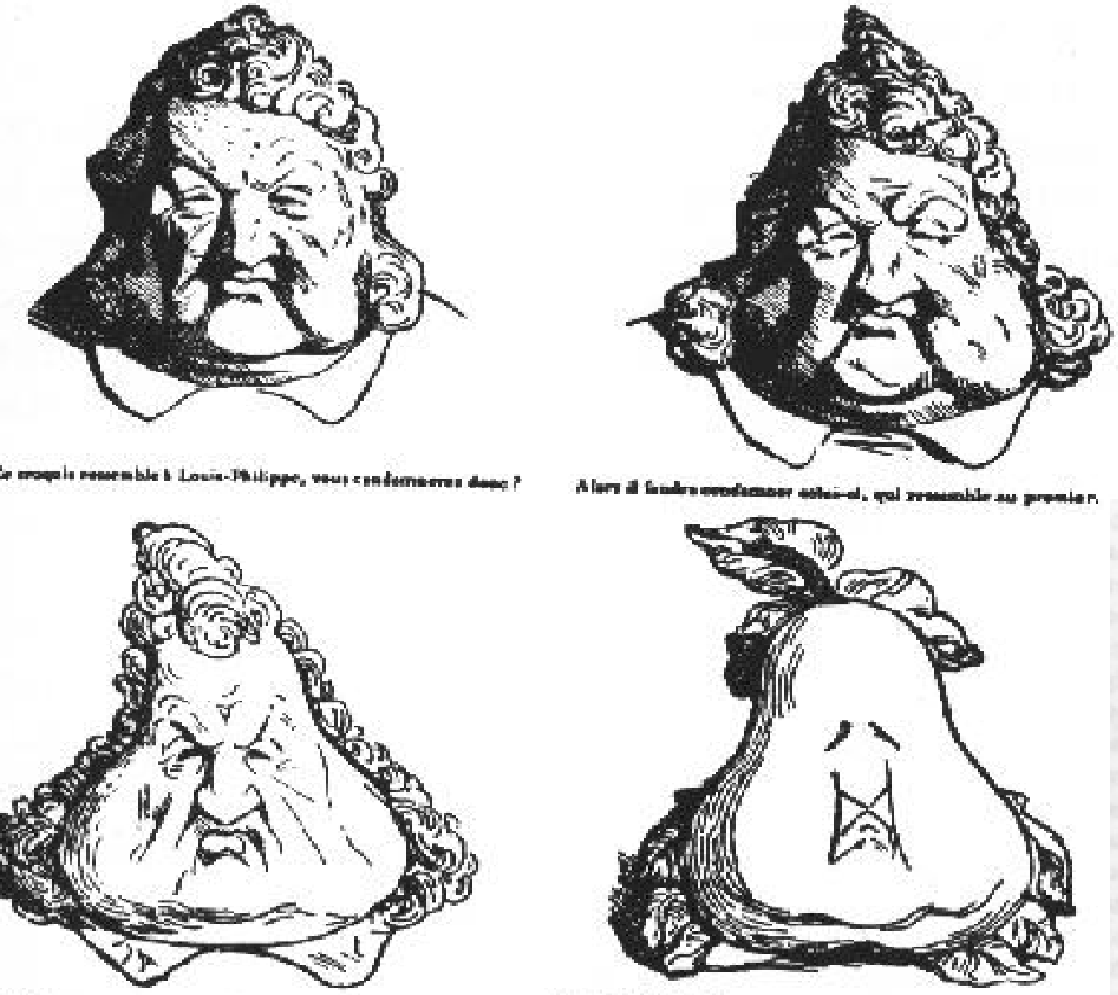






Adapted from Pietro Perona, Object Recognition Workshop, 2004

Similarity metric



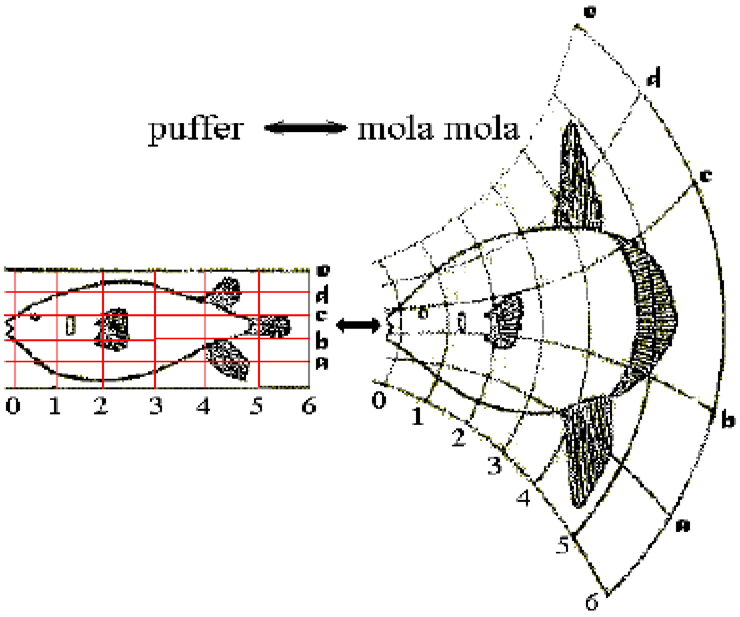
Ce croquis ressemble à Louis-Philippe, vous condamnerez donc?

Alors il faudra condamner celui-ci, qui ressemble au premier.

Puis condamner cet autre, qui ressemble au second.

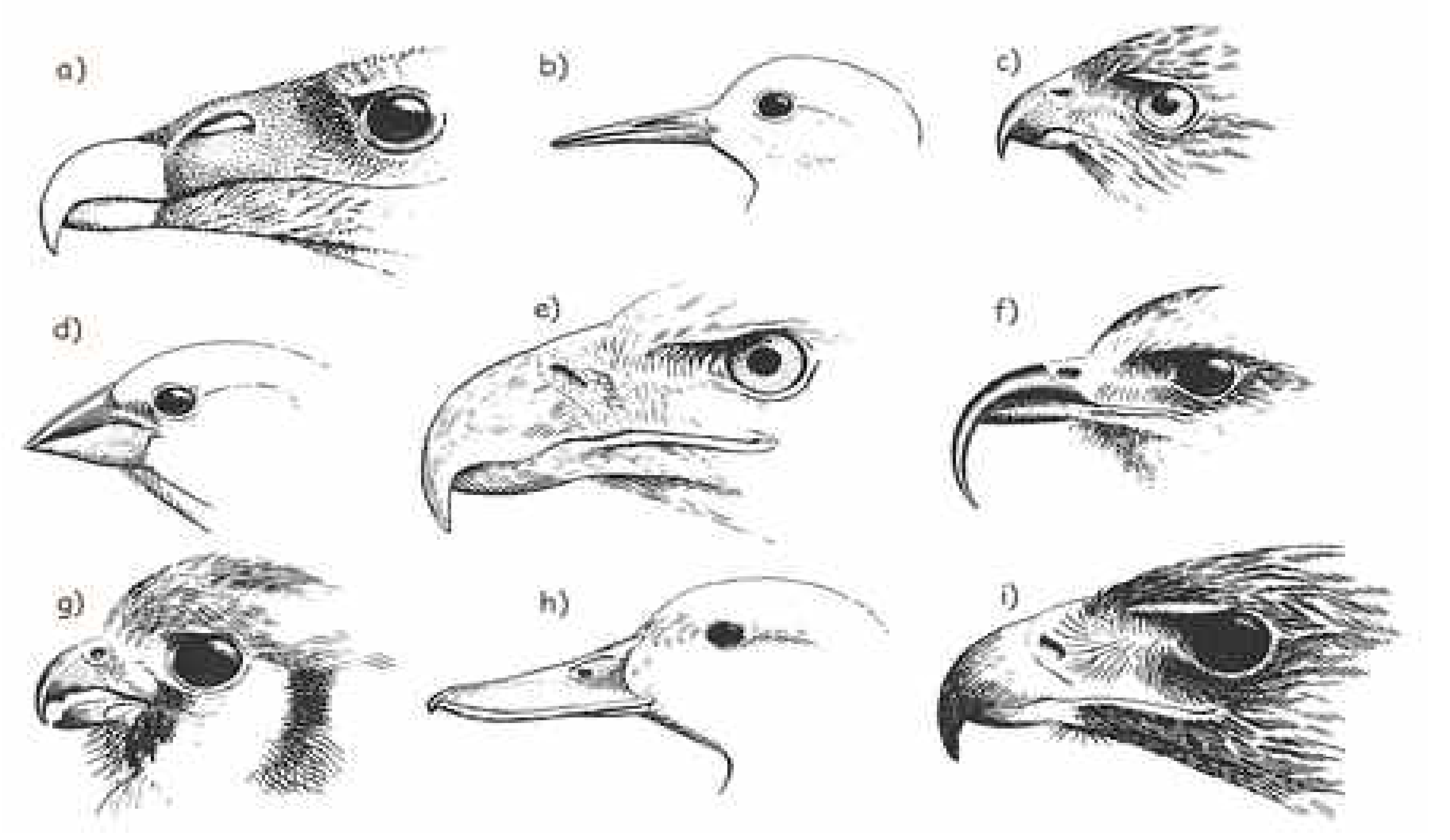
Ainsi, pour une poire, [illegible]

En enfin, si vous êtes conséquents, vous ne sauriez absoudre cette poire, qui ressemble aux croquis précédents.



Adapted from Pietro Perona, Object Recognition Workshop, 2004

Form and function

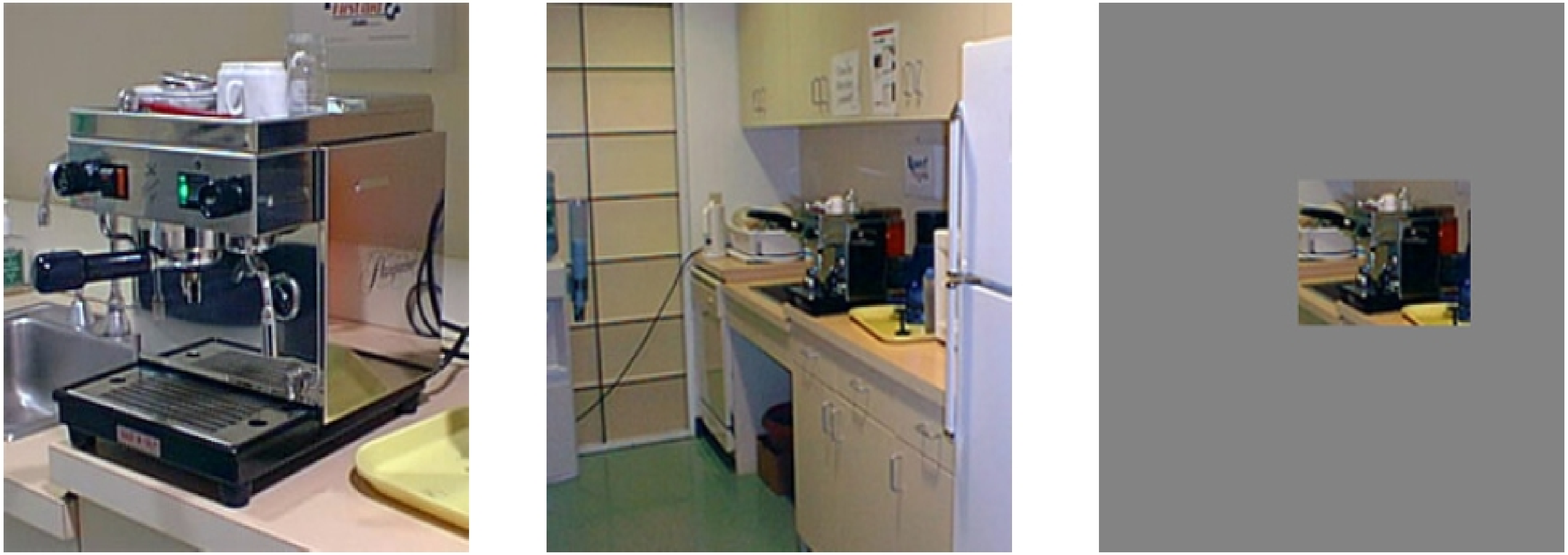


Adapted from Pietro Perona, Object Recognition Workshop, 2004

Context

Murphy et al., ICCV2003

Context

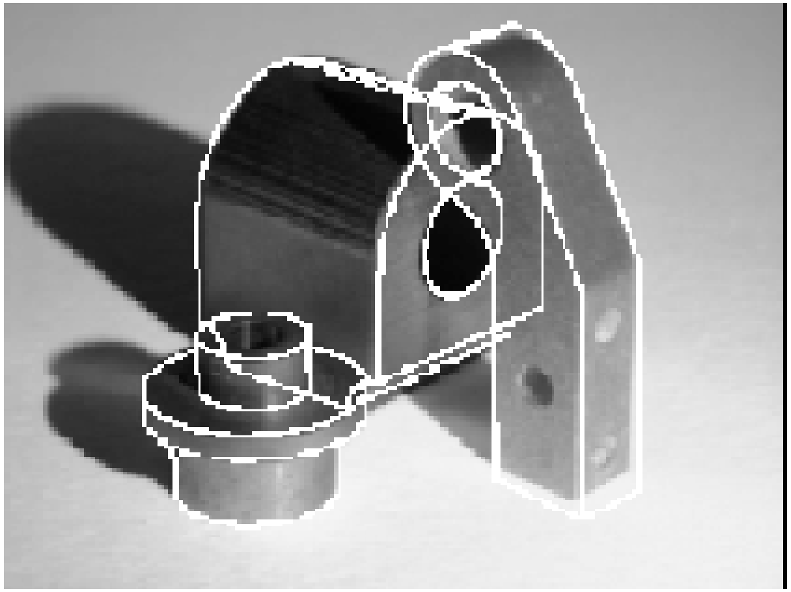


(a) Isolated object (b) Object in context (c) Low-res Object

Murphy et al., ICCV2003

Object Recognition

- Model based vision
- Object Recognition as template matching



Adapted from David Forsyth, UC Berkeley

Model based Vision

–Object recognition as a correspondence problem –

–which image feature corresponds to which feature on which object?

Idea : If we know correspondences for a small set of features it is easy to obtain correspondences for a much larger data set.

Assumption : there is a collection of geometric models of objects that should be recognized.

The collection is called a *modelbase*

Model based Vision

Method : Hypothesize and test

- hypothesize a correspondence between a collection of image features and a collection of object features, then use this to generate a hypothesis about the projection from the object coordinate frame to the image frame

When camera intrinsic parameters are known, the hypothesis is known equivalent to a hypothetical position and orientation - *pose*

- Use this projection hypothesis to generate a rendering of the object – usually known as *backprojection*
- Compare the rendering to the image, and if two are sufficiently similar accept the hypothesis

Pose consistency

- Correspondences between image features and model features are not independent.
- A small number of correspondences yields a camera hypothesis --- the others must be consistent with this.

- Strategy:
  - Generate hypotheses using small numbers of correspondences (e.g. triples of points for a calibrated perspective camera, etc., etc.)
  - Backproject and verify
- Notice that the main issue here is camera calibration
- Appropriate groups are “frame groups”

Adapted from David Forsyth, UC Berkeley

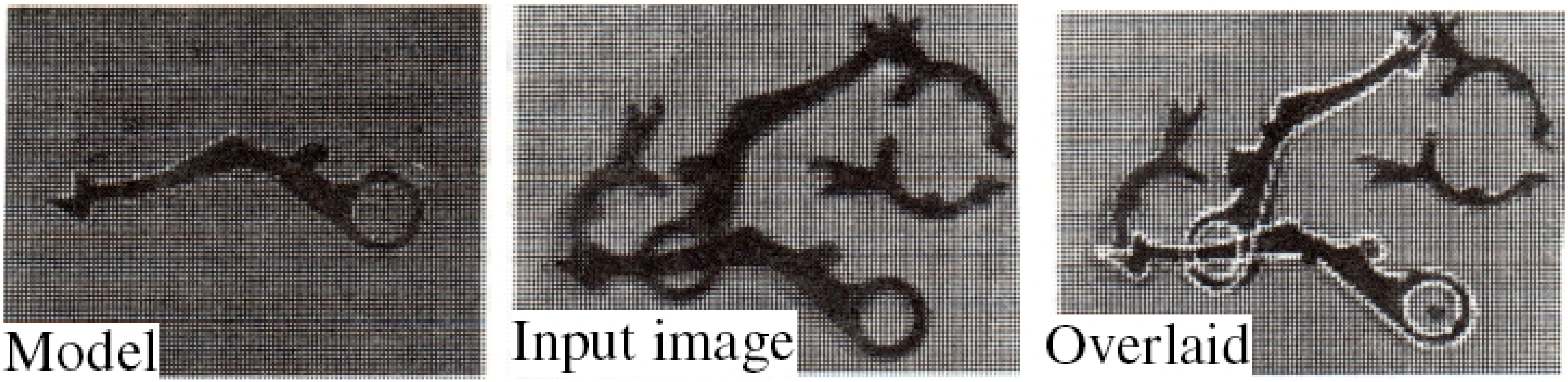


Figure from “Object recognition using alignment,” D.P. Huttenlocher and S. Ullman, Proc. Int. Conf. Computer Vision, 1986, copyright IEEE, 1986

Adapted from David Forsyth, UC Berkeley

Voting on Pose

- Each model leads to many correct sets of correspondences, each of which has the same pose
  - Vote on pose, in an accumulator array
  - This is similar to hough transform
  - Problems:
    - Noise
    - Bucket size

Adapted from David Forsyth, UC Berkeley

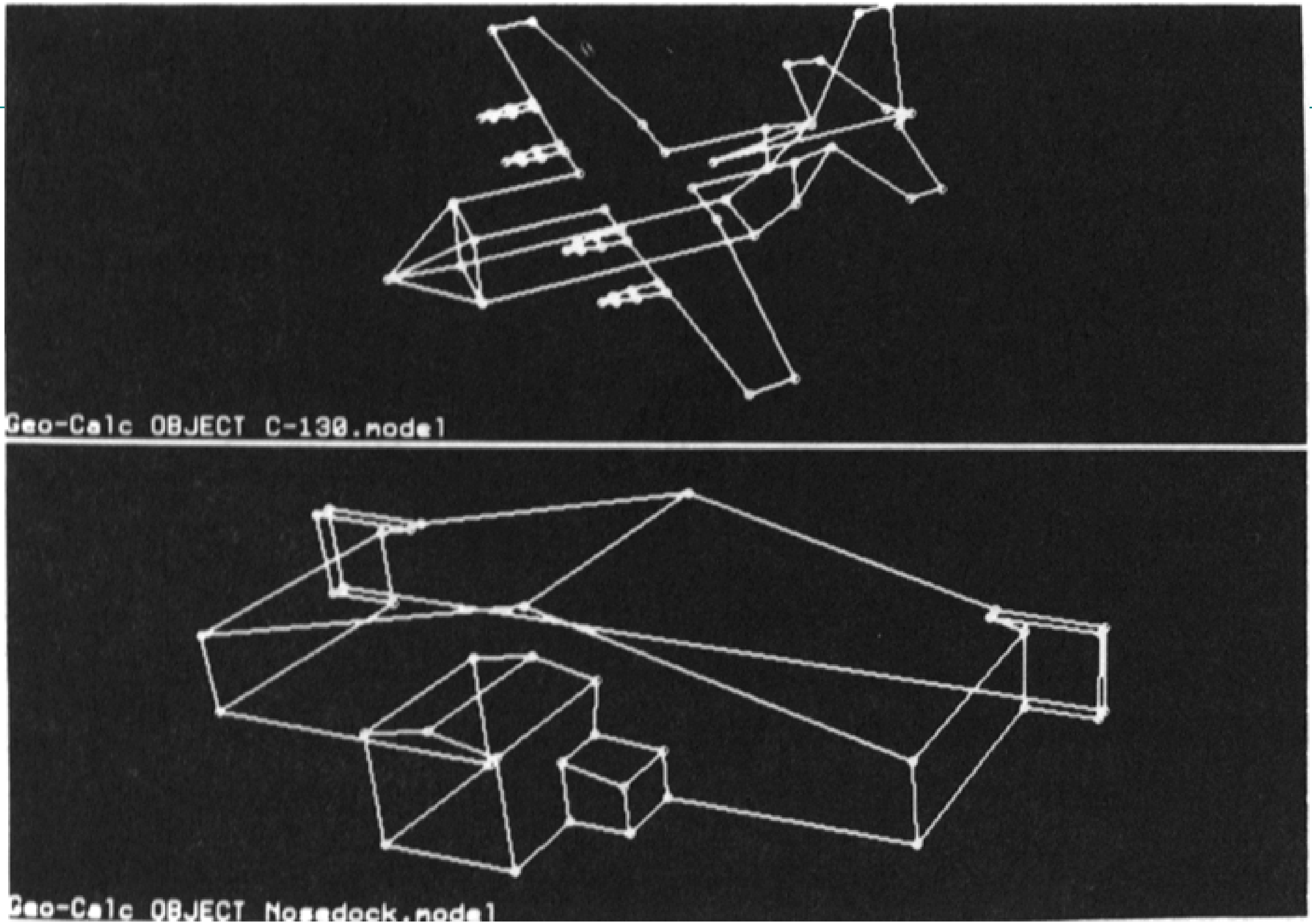


Figure from “The evolution and testing of a model-based object recognition system”, J.L. Mundy and A. Heller, Proc. Int. Conf. Computer Vision, 1990 copyright 1990 IEEE

Adapted from David Forsyth, UC Berkeley

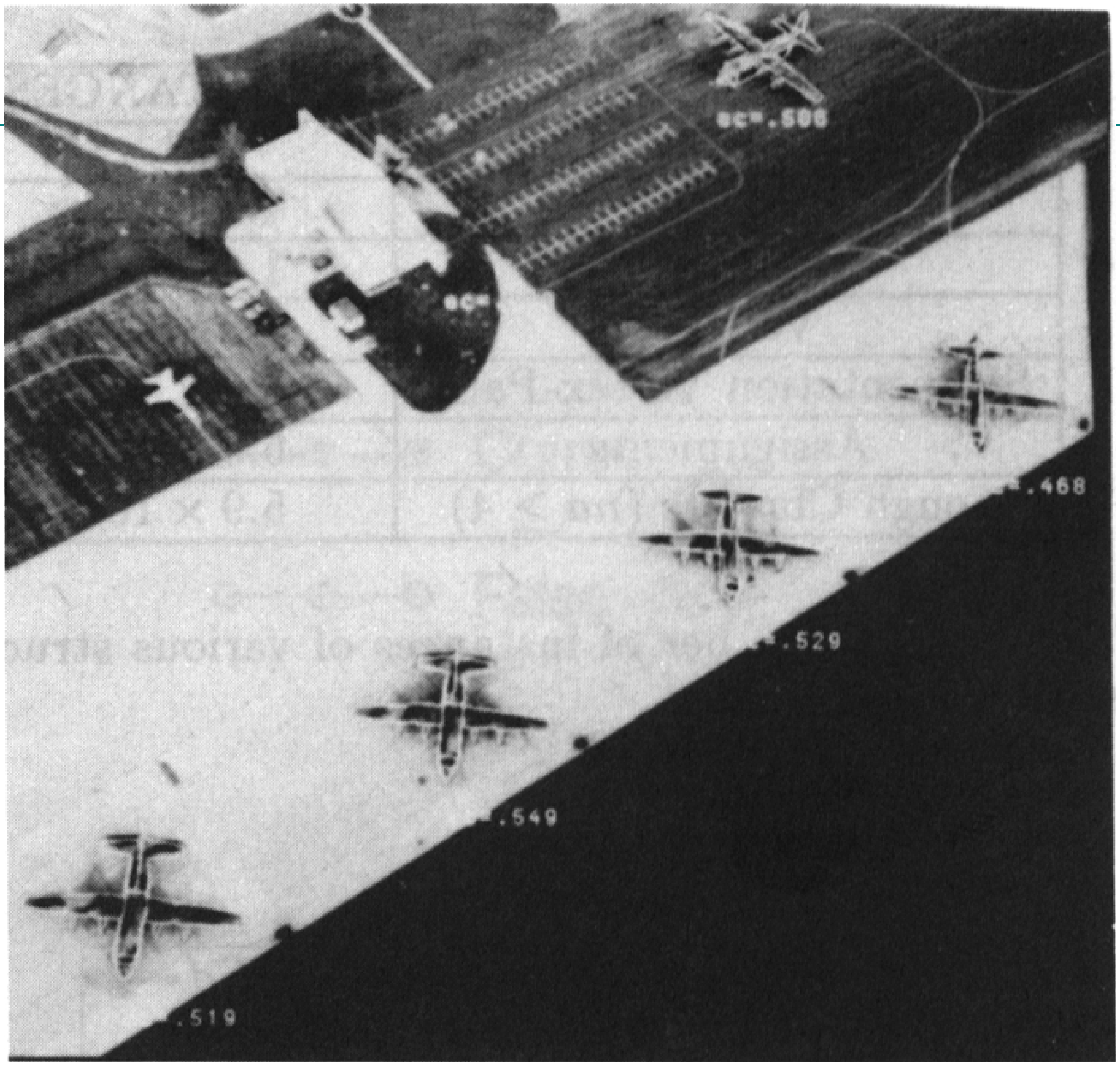


Figure from "The evolution and testing of a model-based object recognition system", J.L. Mundy and A. Heller, Proc. Int. Conf. Computer Vision, 1990 copyright 1990 IEEE

Adapted from David Forsyth, UC Berkeley

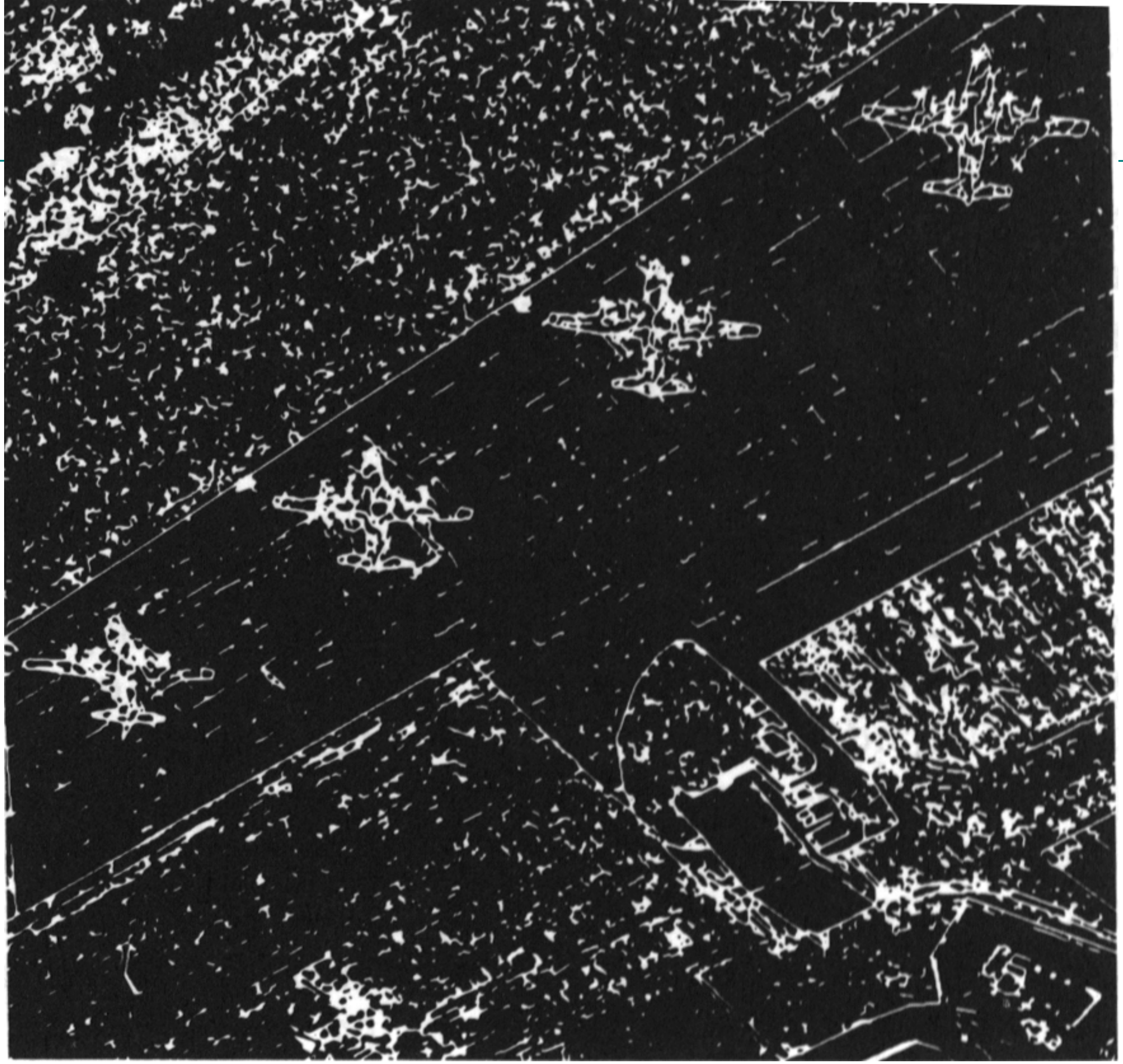


Figure from “The evolution and testing of a model-based object recognition system”, J.L. Mundy and A. Heller, Proc. Int. Conf. Computer Vision, 1990 copyright 1990 IEEE

Adapted from David Forsyth, UC Berkeley

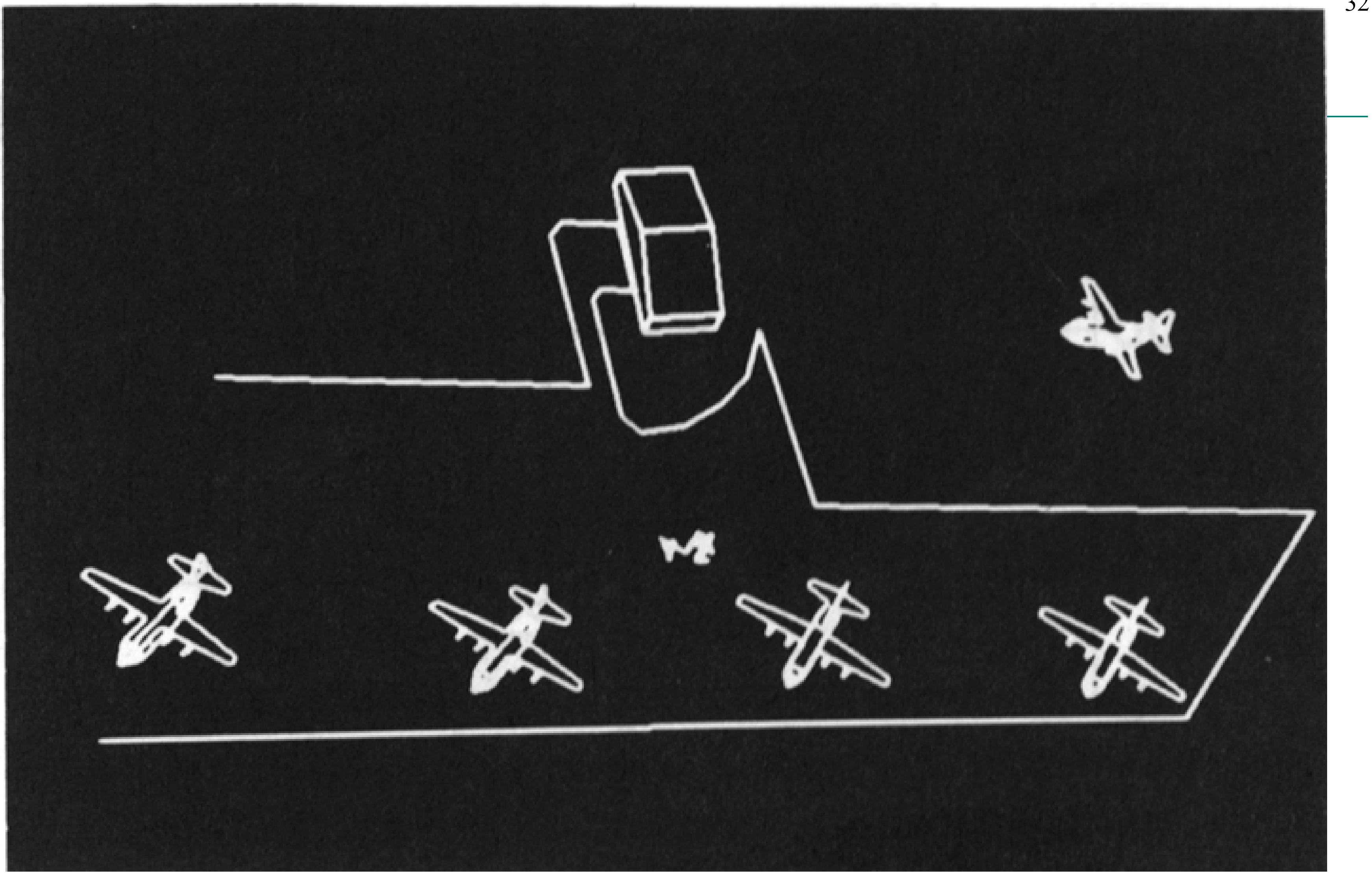
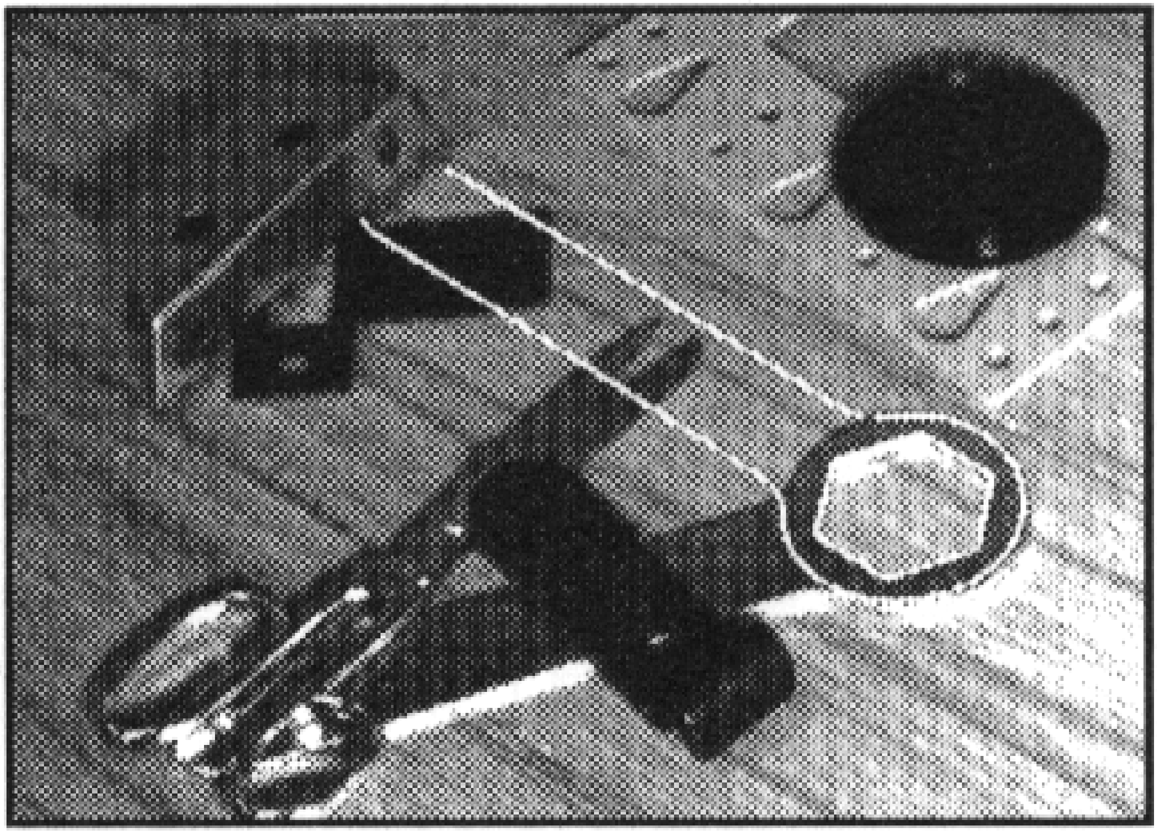


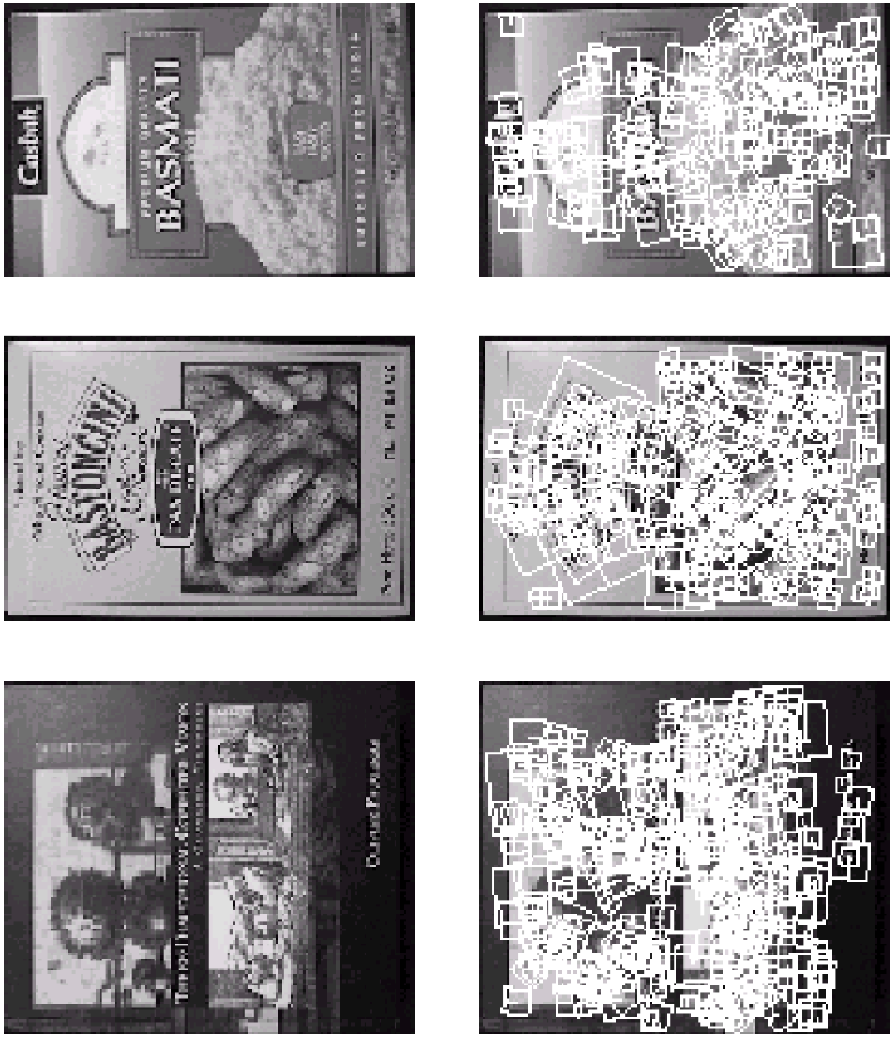
Figure from “The evolution and testing of a model-based object recognition system”, J.L. Mundy and A. Heller, Proc. Int. Conf. Computer Vision, 1990 copyright 1990 IEEE

Adapted from David Forsyth, UC Berkeley



Adapted from David Forsyth, UC Berkeley

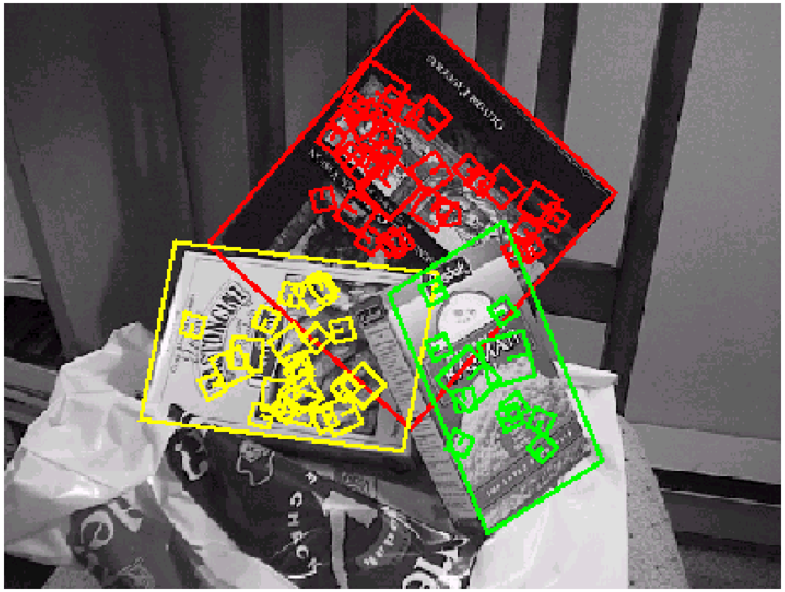
Models for planar surfaces with SIFT keys:



Adapted from Trevor Darrell, MIT

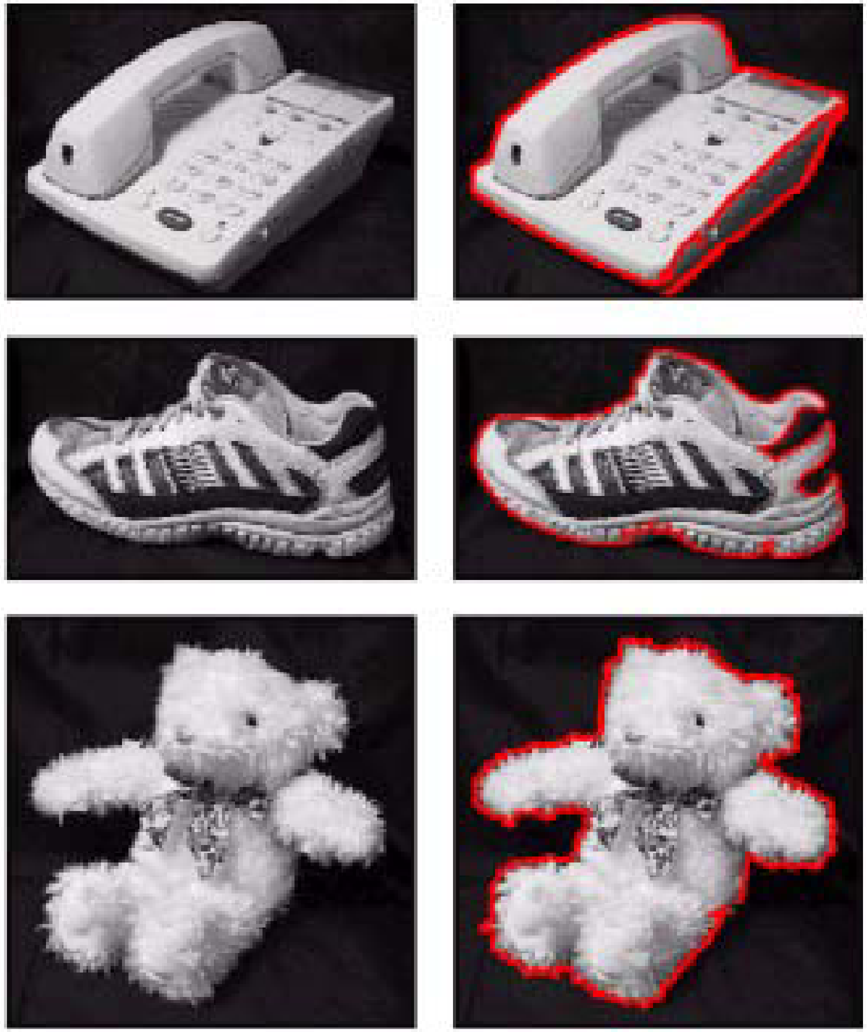
Planar recognition

- Planar surfaces can be reliably recognized at a rotation of 60° away from the camera
- Affine fit approximates perspective projection
- Only 3 points are needed for recognition



Adapted from Trevor Darrell, MIT

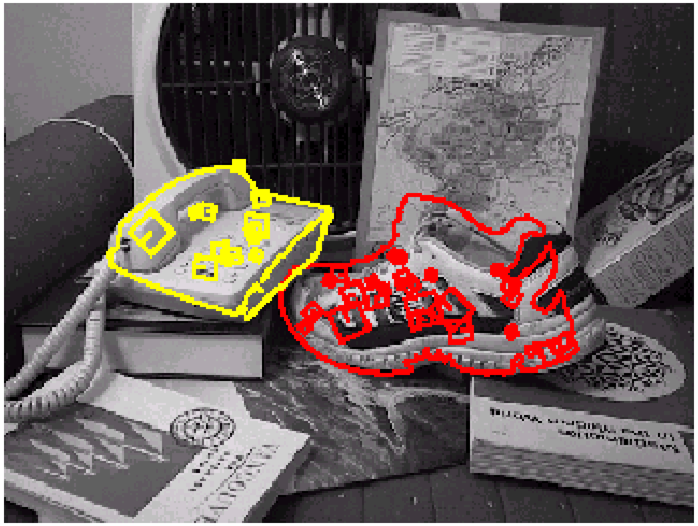
3D Object Recognition



Extract outlines with background subtraction

Adapted from Trevor Darrell, MIT

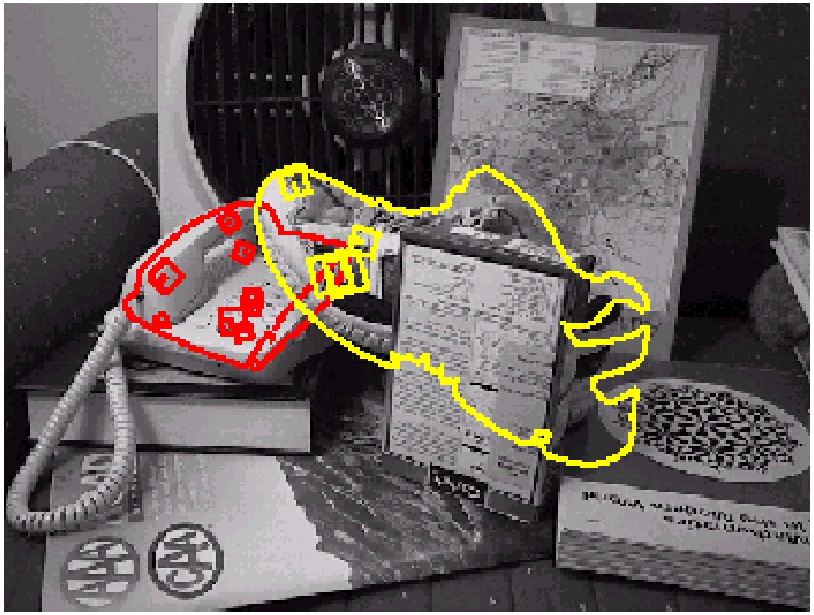
3D Object Recognition

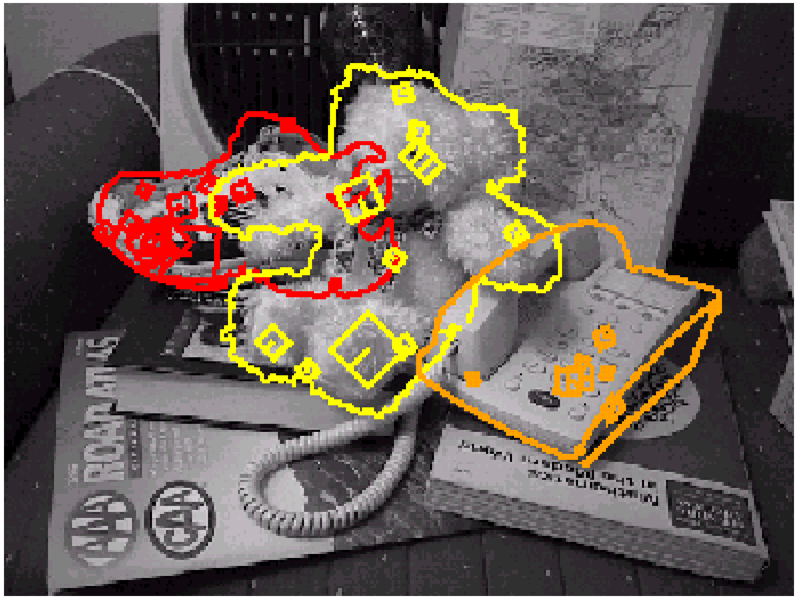


Adapted from Tre

- Only 3 keys are needed for recognition, so extra keys provide robustness
- Affine model is no longer as accurate

Recognition under occlusion





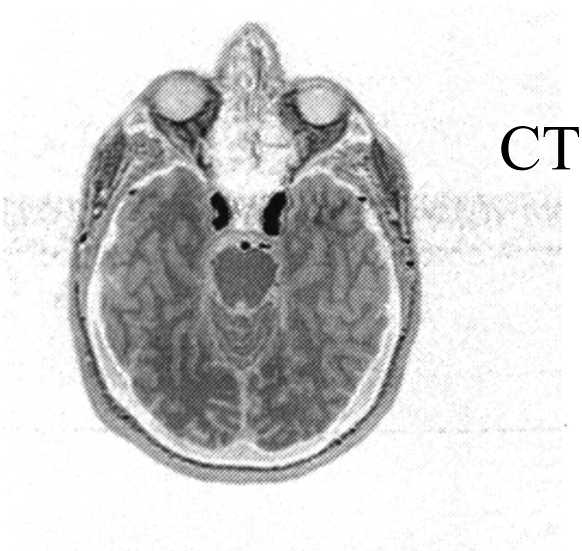
Adapted from Trevor Darrell, MIT

Application: Surgery

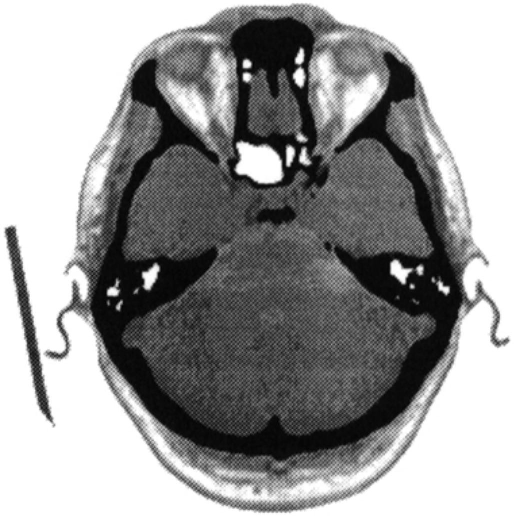
- To minimize damage by operation planning
- To reduce number of operations by planning surgery
- To remove only affected tissue
- Problem
  - ensure that the model with the operations planned on it and the information about the affected tissue lines up with the patient
  - display model information supervised on view of patient
  - **Big Issue**: coordinate alignment, as above

Adapted from David Forsyth, UC Berkeley

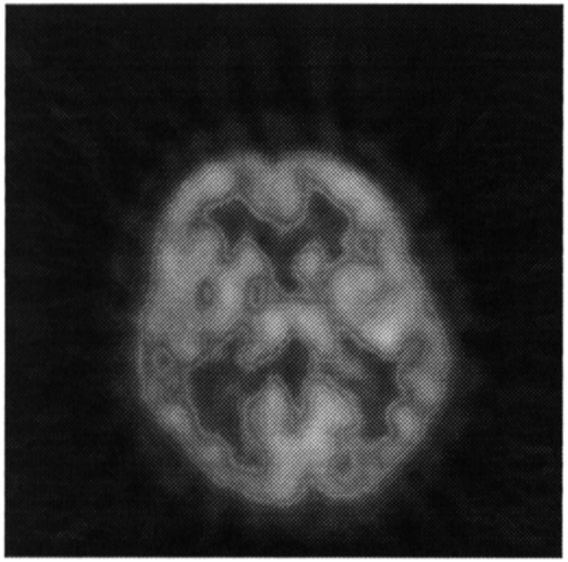
MRI



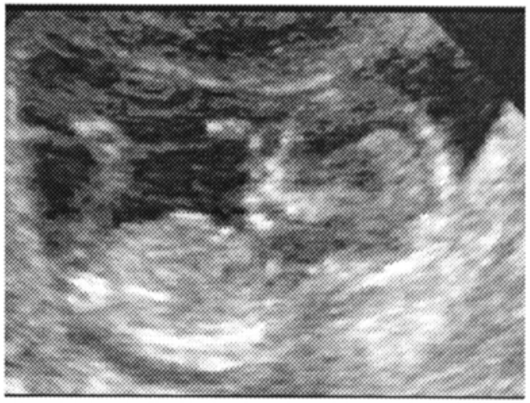
CTI



NMI

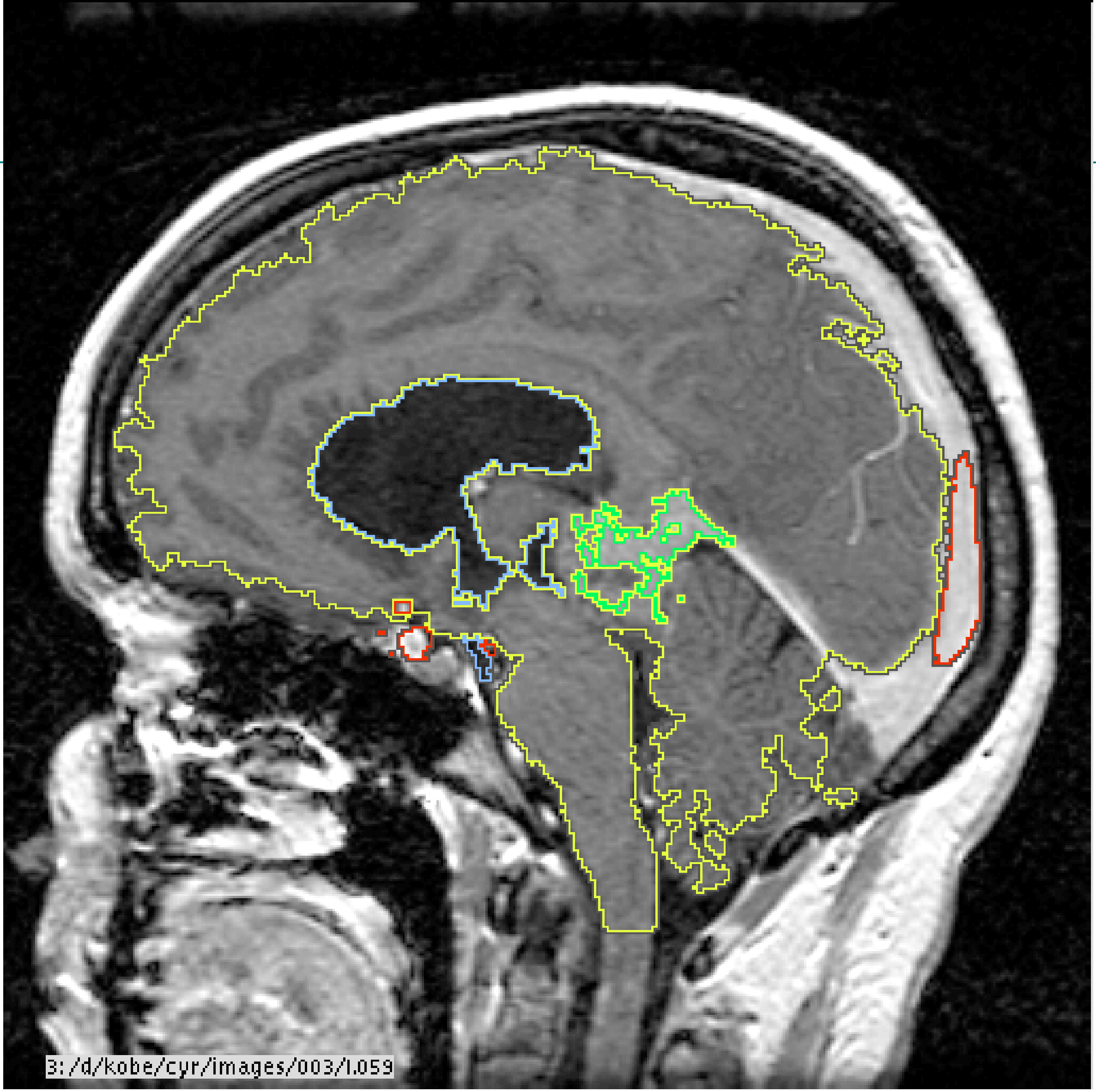


USI



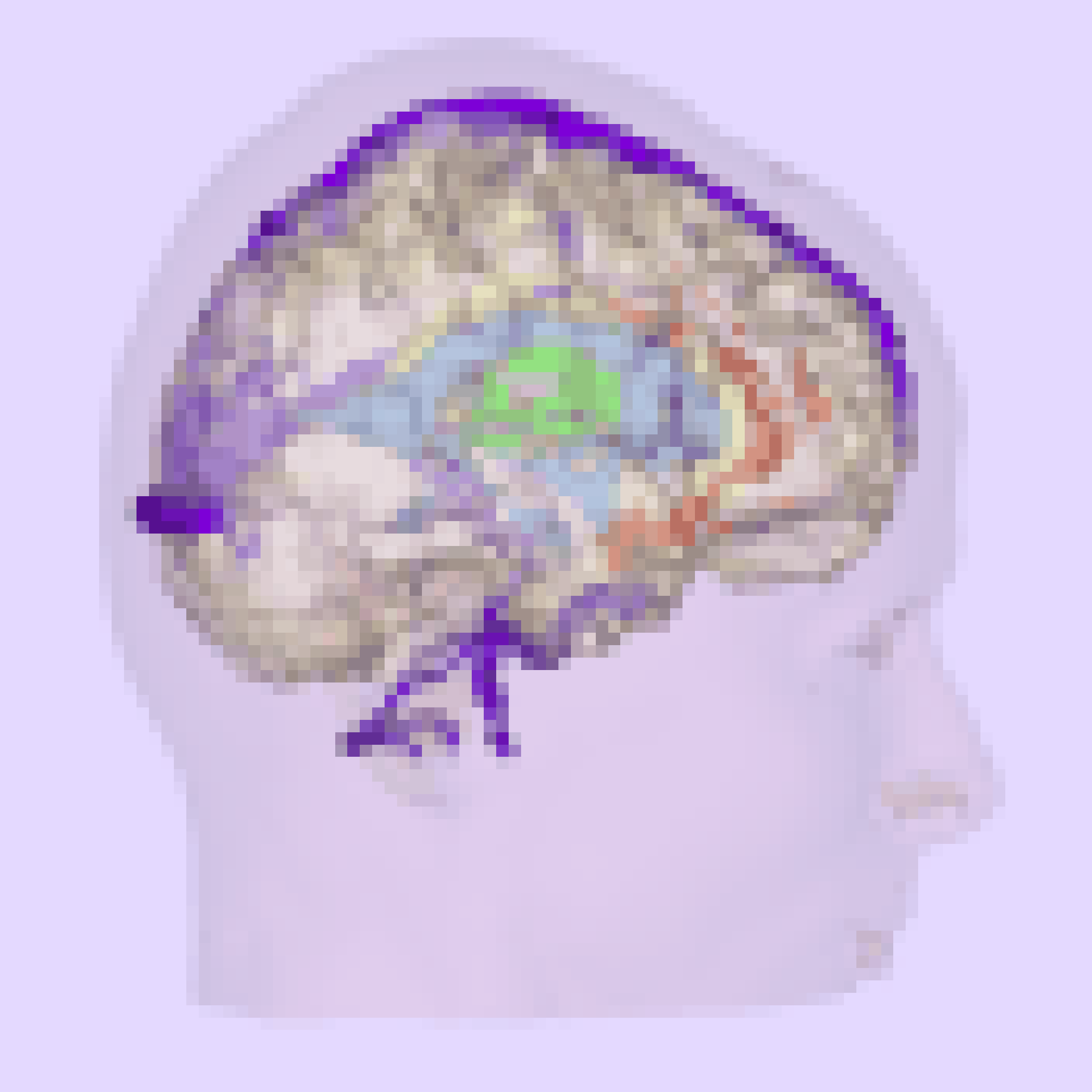
Reprinted from Image and Vision Computing, v. 13, N. Ayache, "Medical computer vision, virtual reality and robotics", Page 296, copyright, (1995), with permission from Elsevier Science

Adapted from David Forsyth, UC Berkeley



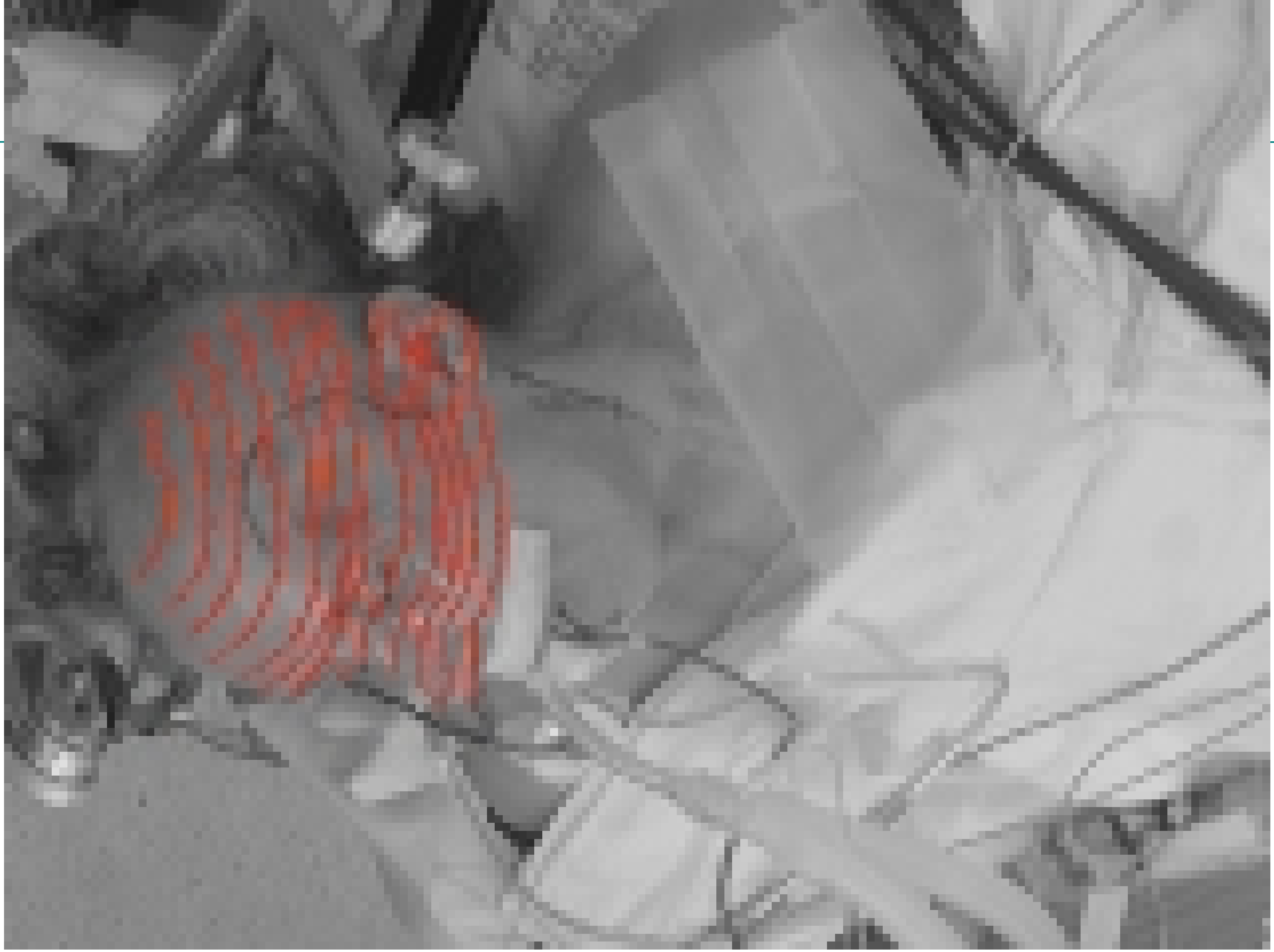
Figures by kind permission of Eric Grimson; further information can be obtained from his web site http://www.ai.mit.edu/people/welg/welg.html.

Adapted from David Forsyth, UC Berkeley



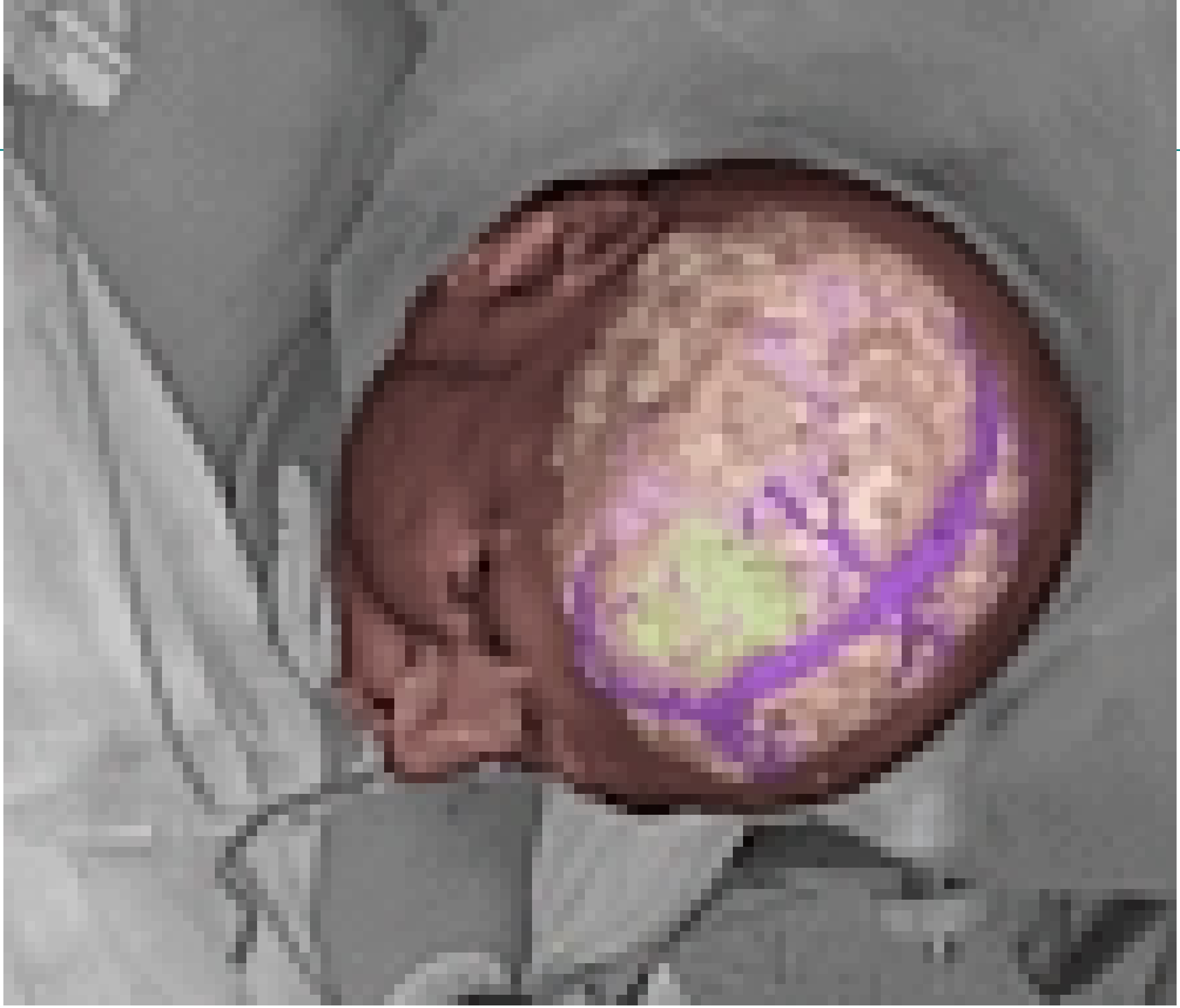
Figures by kind permission of Eric Grimson; further information can be obtained from his web site http://www.ai.mit.edu/people/welg/welg.html.

Adapted from David Forsyth, UC Berkeley



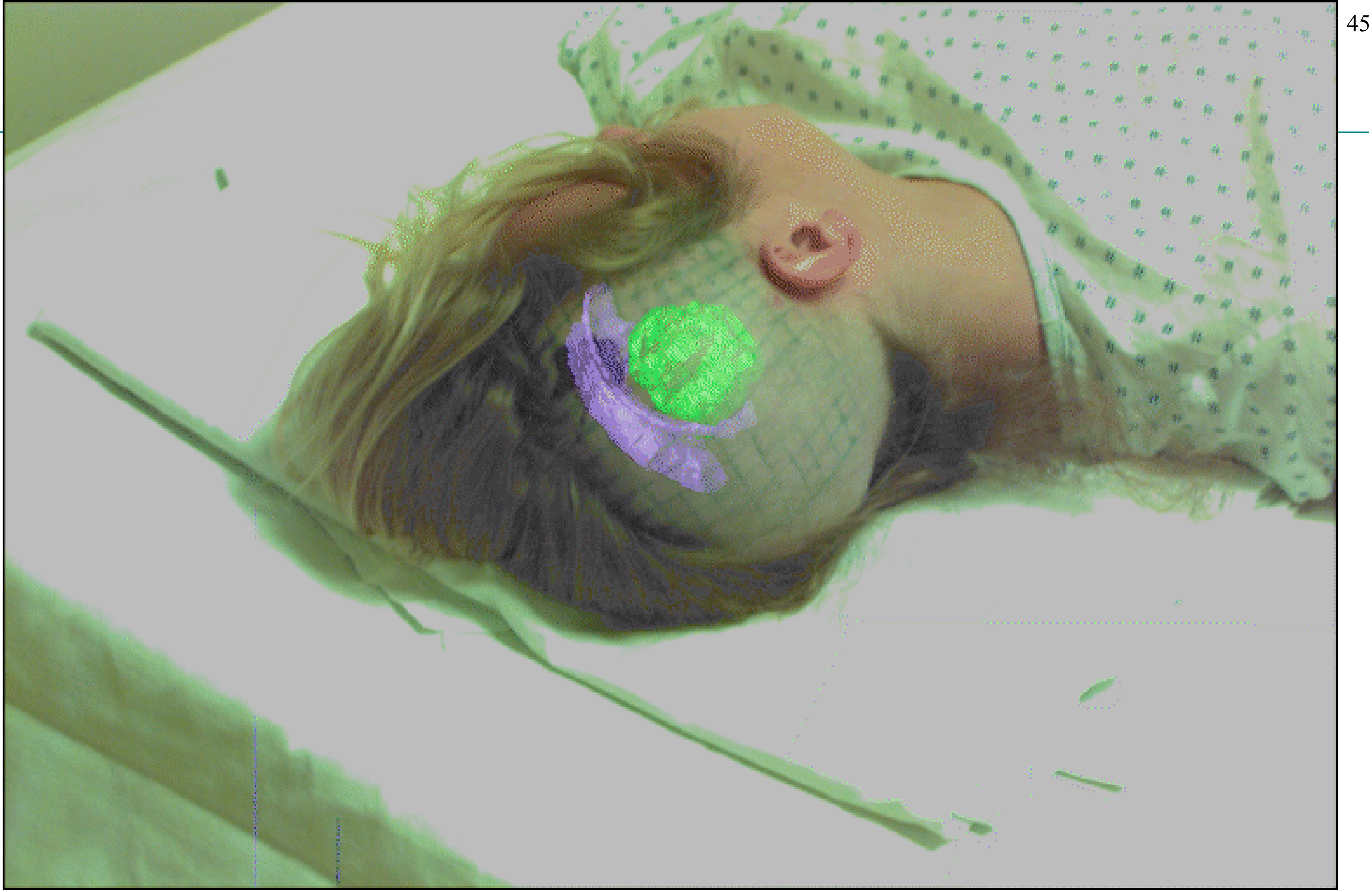
Figures by kind permission of Eric Grimson; further information can be obtained from his web site http://www.ai.mit.edu/people/welg/welg.html.

Adapted from David Forsyth, UC Berkeley



Figures by kind permission of Eric Grimson; further information can be obtained from his web site http://www.ai.mit.edu/people/welg/welg.html.

Adapted from David Forsyth, UC Berkeley

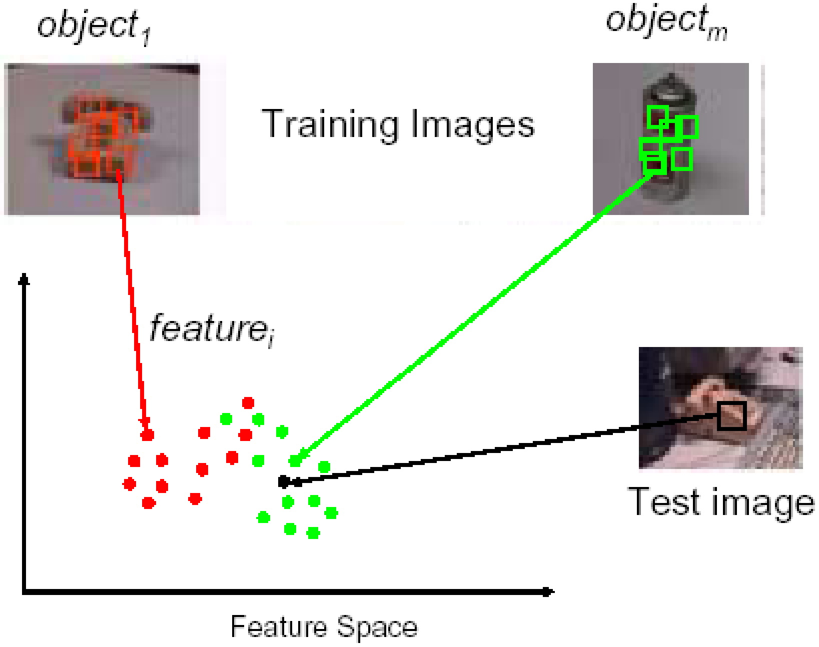


Figures by kind permission of Eric Grimson; further information can be obtained from his web site http://www.ai.mit.edu/people/welg/welg.html.

Adapted from David Forsyth, UC Berkeley

Template based Recognition

- View based, Image-based
- We have seen very simple template matching (under filters)
- Some objects behave like quite simple templates
  - Frontal faces
- Strategy:
  - Find image windows
  - Correct lighting
  - Pass them to a statistical test (a classifier) that accepts faces and rejects non-faces



Adapted from Martial Hebert, CMU

Basic ideas in classifiers

- Loss
  - some errors may be more expensive than others
    - e.g. a fatal disease that is easily cured by a cheap medicine with no side-effects -> false positives in diagnosis are better than false negatives
  - We discuss two class classification: L(1->2) is the loss caused by calling 1 a 2
- Total risk of using classifier s

$$R(s) = Pr\left\{1 \to 2|\text{using } s\right\} L(1 \to 2) + Pr\left\{2 \to 1|\text{using } s\right\} L(2 \to 1)$$

Adapted from David Forsyth, UC Berkeley

Basic ideas in classifiers

- Generally, we should classify as 1 if the expected loss of classifying as 1 is better than for 2
- gives

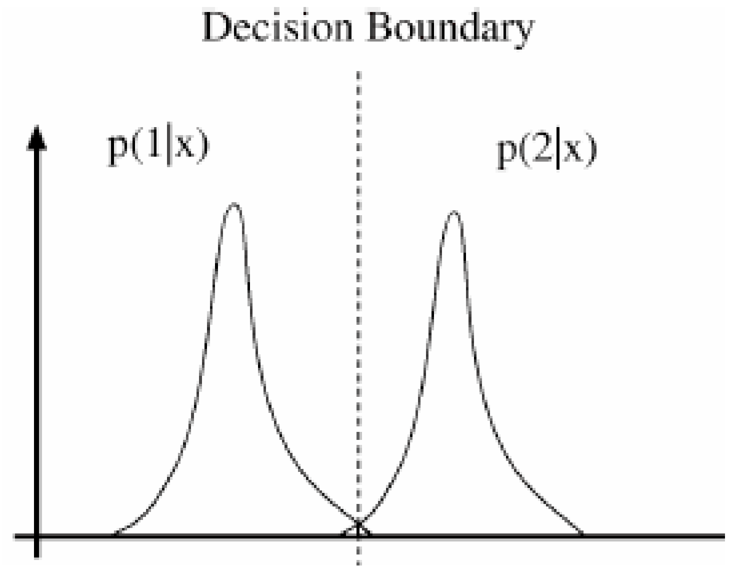
1 if $$p(1|x)L(1 \to 2) > p(2|x)L(2 \to 1)$$

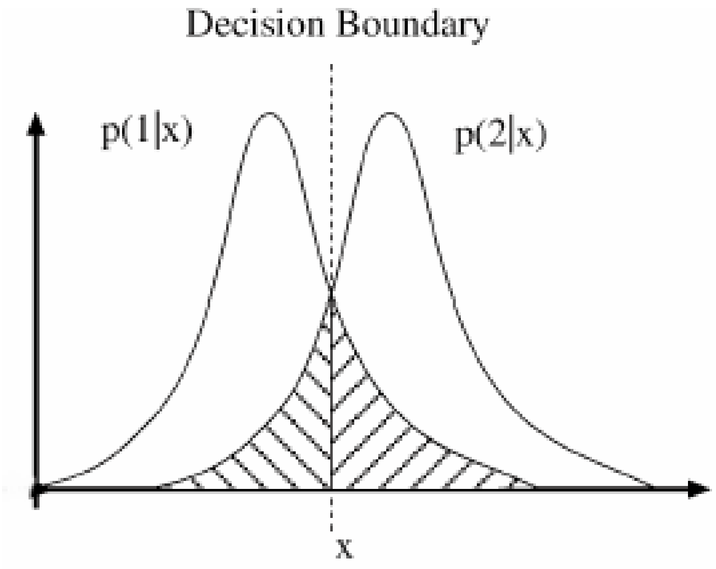
2 if $$p(1|x)L(1 \to 2) < p(2|x)L(2 \to 1)$$

- Crucial notion: Decision boundary
  - points where the loss is the same for either case

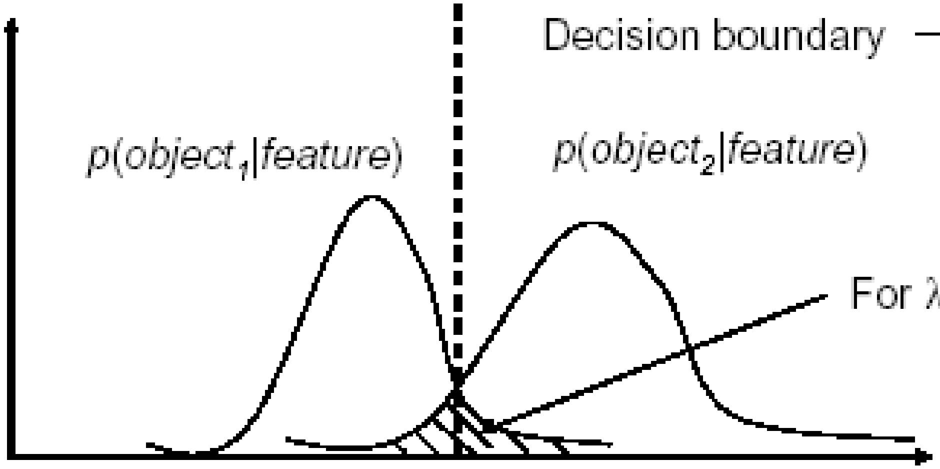
Adapted from David Forsyth, UC Berkeley

Some loss may be inevitable: the minimum risk (shaded area) is called the Bayes risk





Adapted from David Forsyth, UC Berkeley



Decision boundary $\frac{p(object_1|feature)}{p(object_2|feature)} > \lambda$

$p(object_1|feature)$

$p(object_2|feature)$

For λ = 1: Minimizes Bayes risk

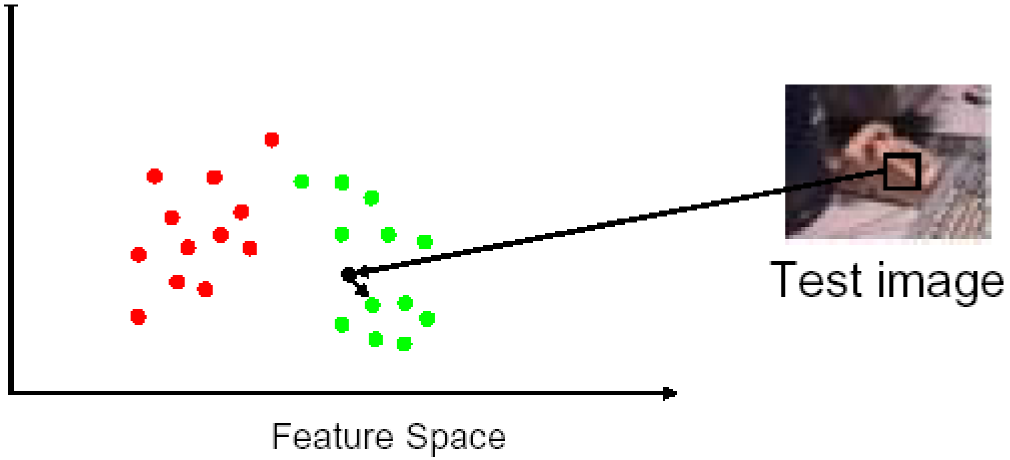
Learned from training data

$p(object_i|feature) \sim p(feature|object_i)\, p(object_i)$

How to represent and learn $p(feature|object_i)$ or decision boundary?
How to approach Bayes risk given small number of samples?
What features to use?
How to reduce the feature space?

Adapted from Martial Hebert, CMU

Nearest Neighbor



- Does not require recovery of distributions or decision surfaces
- Asymptotically twice Bayes risk at most
- Choice of distance metric critical
- Indexing may be difficult

Adapted from Martial Hebert, CMU

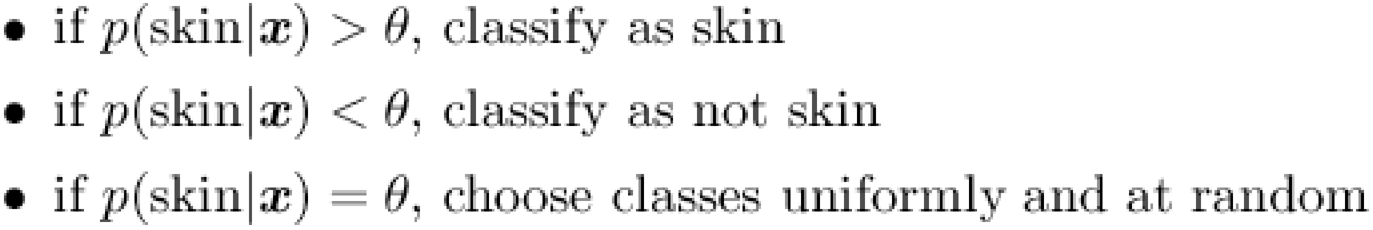
Histogram based classifiers

- Use a histogram to represent the class-conditional densities
  - (i.e. p(x|1), p(x|2), etc)
- Advantage: estimates become quite good with enough data!
- Disadvantage: Histogram becomes big with high dimension
  - but maybe we can assume feature independence?

Adapted from David Forsyth, UC Berkeley

Finding skin

- Skin has a very small range of (intensity independent) colours, and little texture
  - Compute an intensity-independent colour measure, check if colour is in this range, check if there is little texture (median filter)
  - See this as a classifier - we can set up the tests by hand, or learn them.
  - get class conditional densities (histograms), priors from data (counting)
- Classifier is



- if $p(\mathrm{skin}|\boldsymbol{x}) > \theta$, classify as skin
- if $p(\mathrm{skin}|\boldsymbol{x}) < \theta$, classify as not skin
- if $p(\mathrm{skin}|\boldsymbol{x}) = \theta$, choose classes uniformly and at random

Adapted from David Forsyth, UC Berkeley

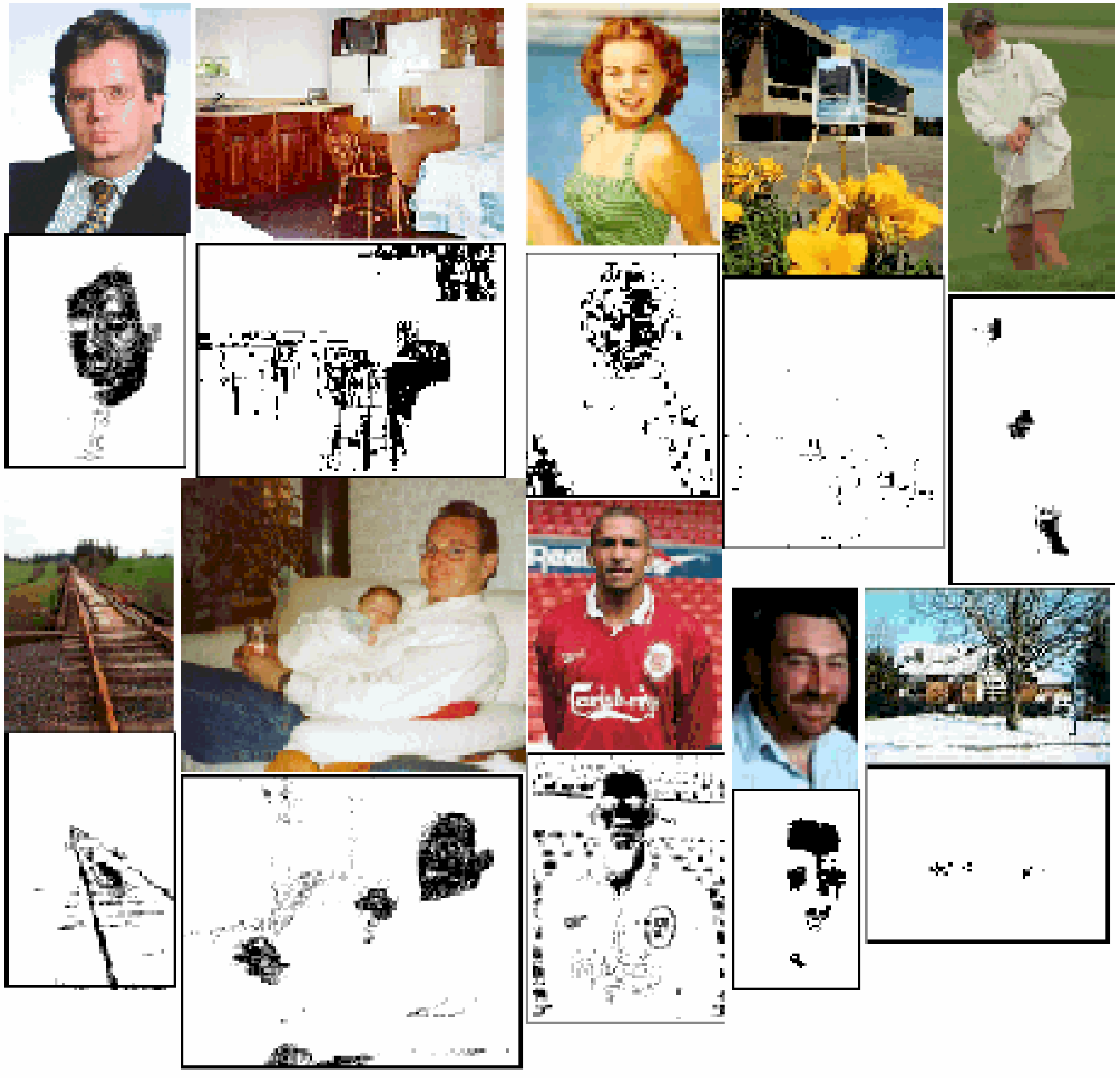


Figure from “Statistical color models with application to skin detection,” M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 copyright 1999, IEEE

Adapted from David Forsyth, UC Berkeley

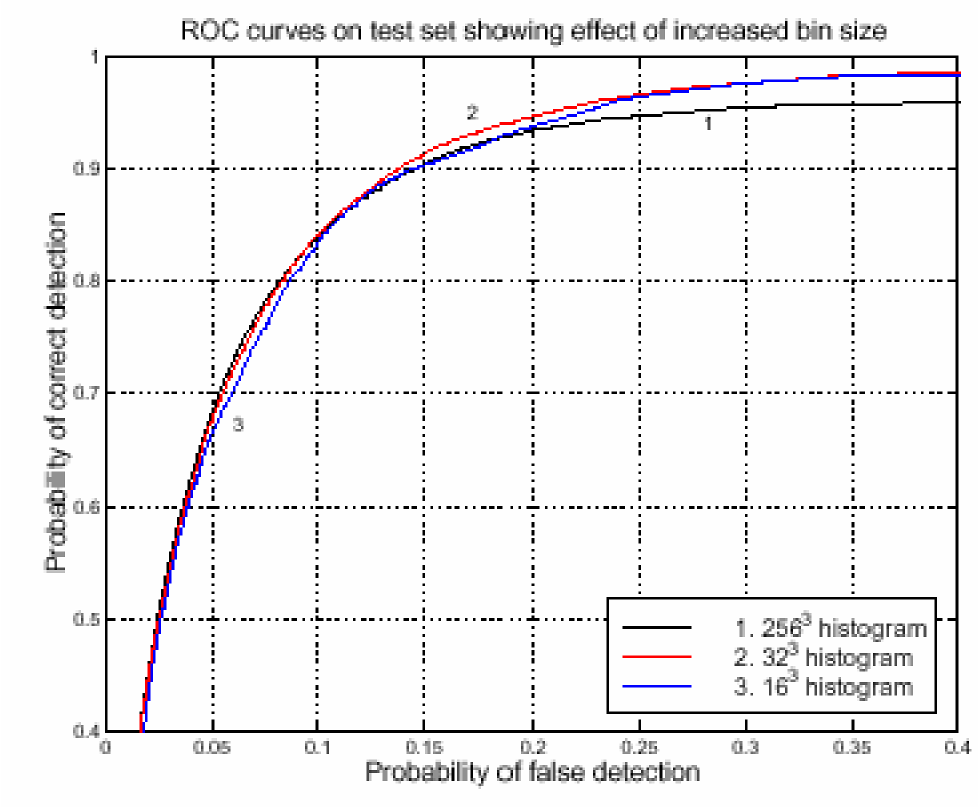
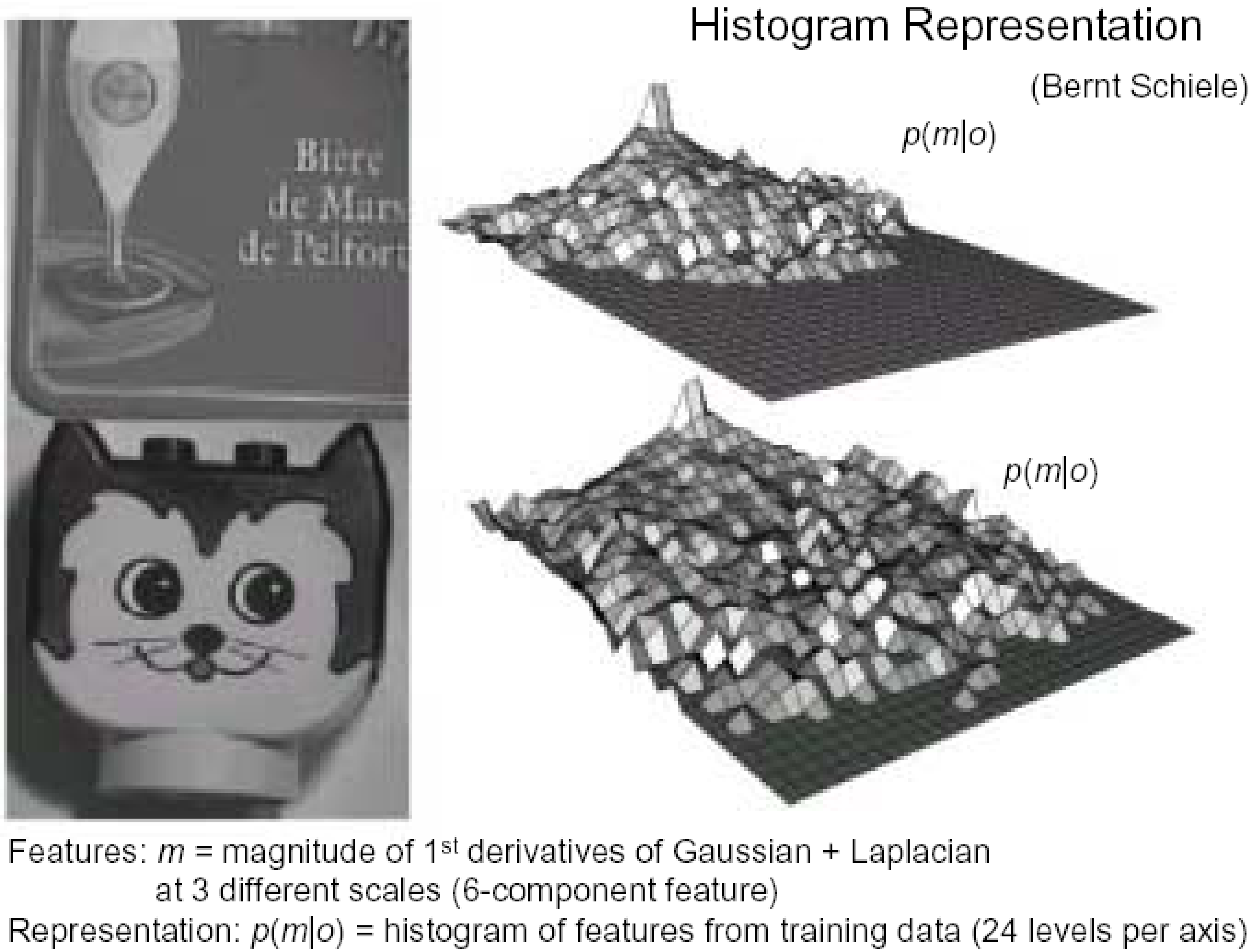


Figure from “Statistical color models with application to skin detection,” M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 copyright 1999, IEEE

Adapted from David Forsyth, UC Berkeley



Histogram Representation

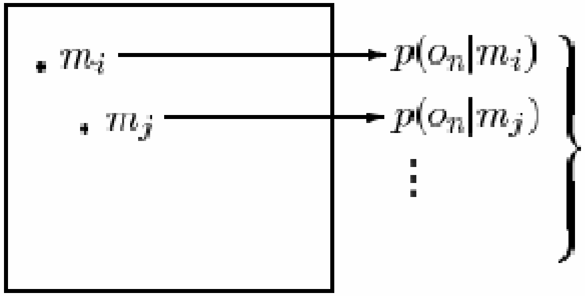
(Bernt Schiele)

Features: m = magnitude of 1st derivatives of Gaussian + Laplacian at 3 different scales (6-component feature)

Representation: $p(m|o)$ = histogram of features from training data (24 levels per axis)

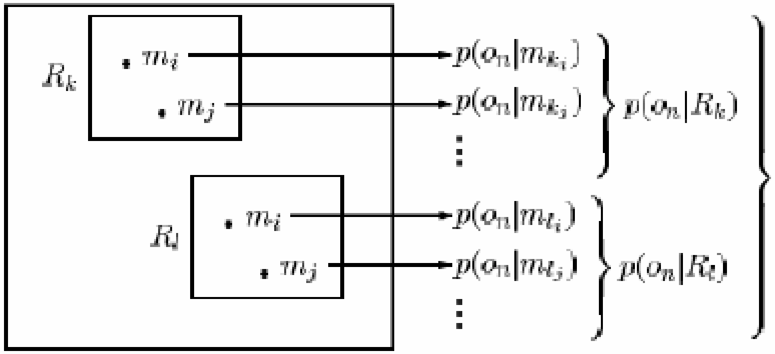
Adapted from Martial Hebert, CMU

Recognition of full-image object:



$$p(o_n|Image) = \frac{\prod_i p(m_i|o_n)p(o_n)}{\sum_j \prod_i p(m_i|o_j)p(o_j)}$$

Recognition of partial-image object:



$$vote(o_n|Image) = \sum_k p(o_n|R_k)$$

Adapted from Martial Hebert, CMU

Adapted from Martial Hebert, CMU

Test image 1

First Match

Second Match

Third Match

Test image 2

First Match

Second Match

Third Match

Test image 3

First Match

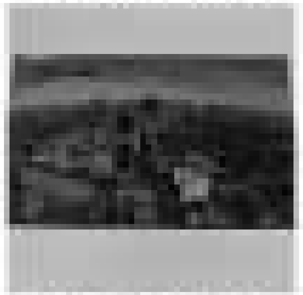
Second Match

Third Match

Test image 4

First Match

Second Match


Third Match

Fourth Match

Finding faces

- Faces “look like” templates (at least when they’re frontal).
- General strategy:
  - search image windows at a range of scales
  - Correct for illumination
  - Present corrected window to classifier

Adapted from David Forsyth, UC Berkeley

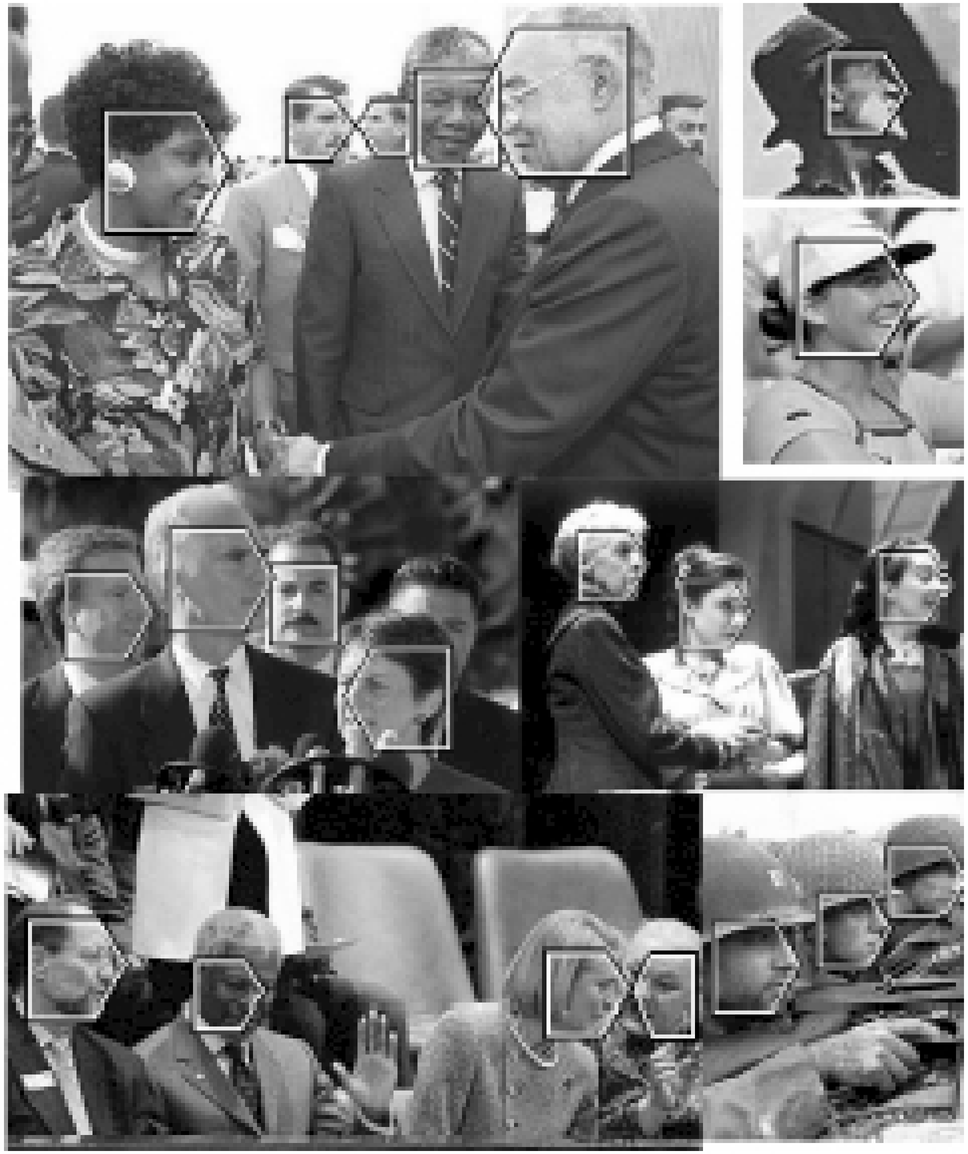


Figure from A Statistical Method for 3D Object Detection Applied to Faces and Cars, H. Schneiderman and T. Kanade, Proc. Computer Vision and Pattern Recognition, 2000, copyright 2000, IEEE

Adapted from David Forsyth, UC Berkeley

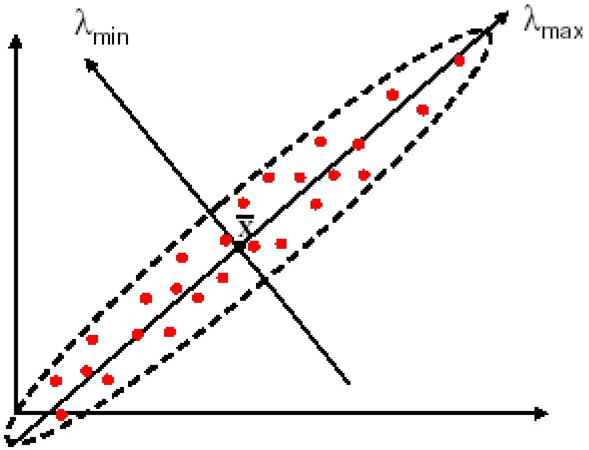
Face Recognition

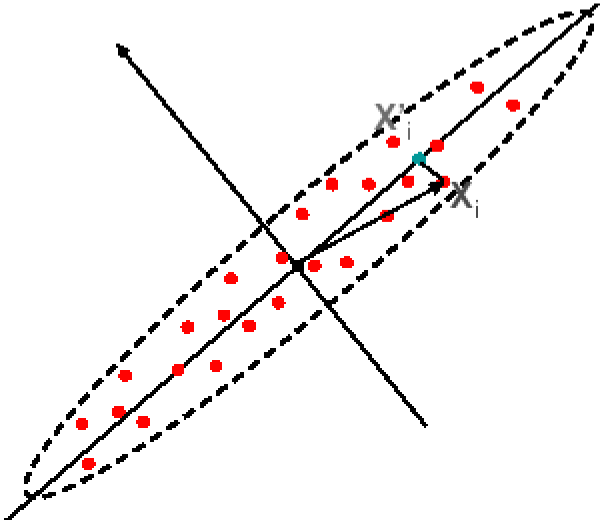
- Whose face is this? (perhaps in a mugshot)
- Issue:
  - What differences are important and what not?
  - Reduce the dimension of the images, while maintaining the "important" differences.
- One strategy:
  - Principal components analysis

- Many face recognition strategies at http://www.cs.rug.nl/users/peterkr/FACE/face.html

Adapted from David Forsyth, UC Berkeley

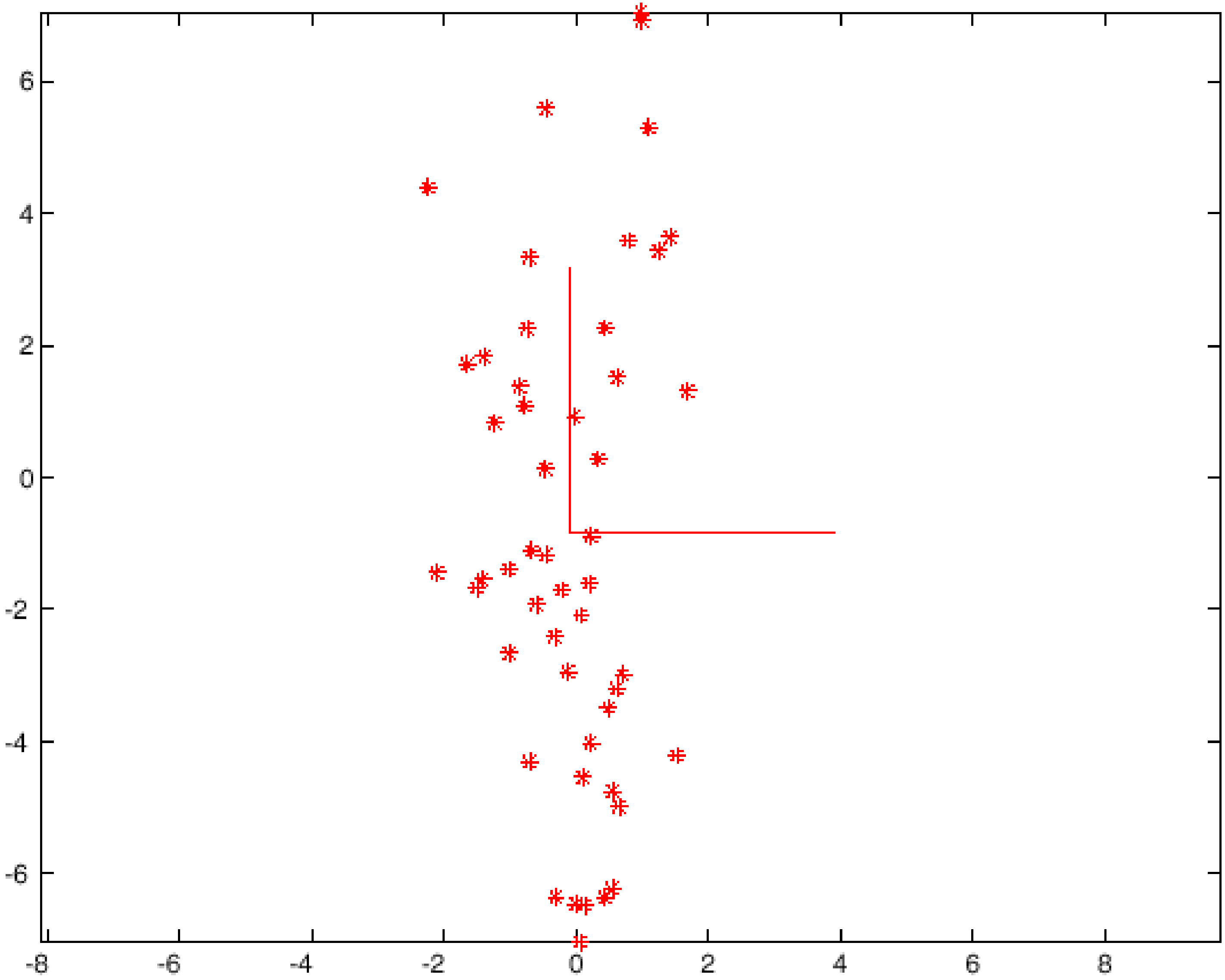
X = feature vector of high dimension
→ Difficult indexing in high-dimensional space
→ Most of the dimensions are probably not useful





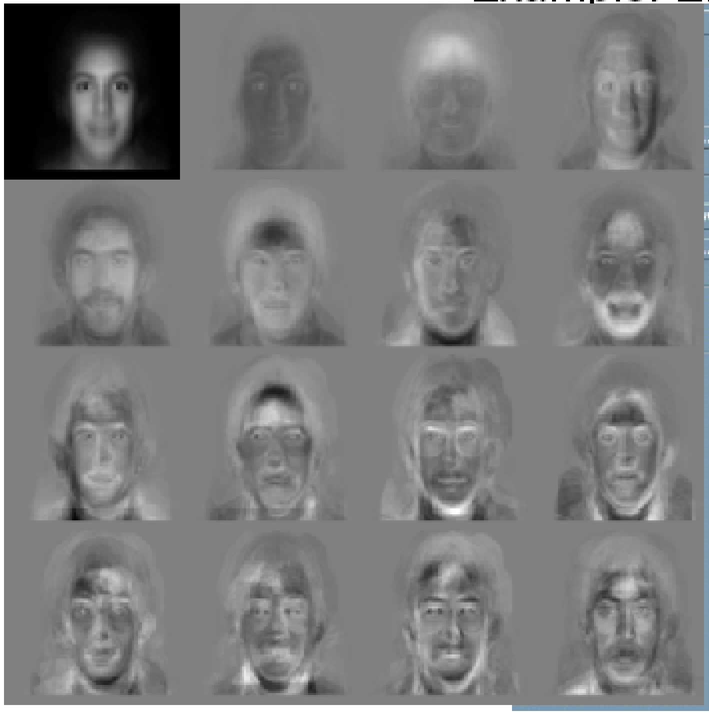
PCA: Project first in the lower-dimensional space spanned by the principal component
→ Indexing in much lower dimensional space
→ Feature selection

Adapted from Martial Hebert, CMU

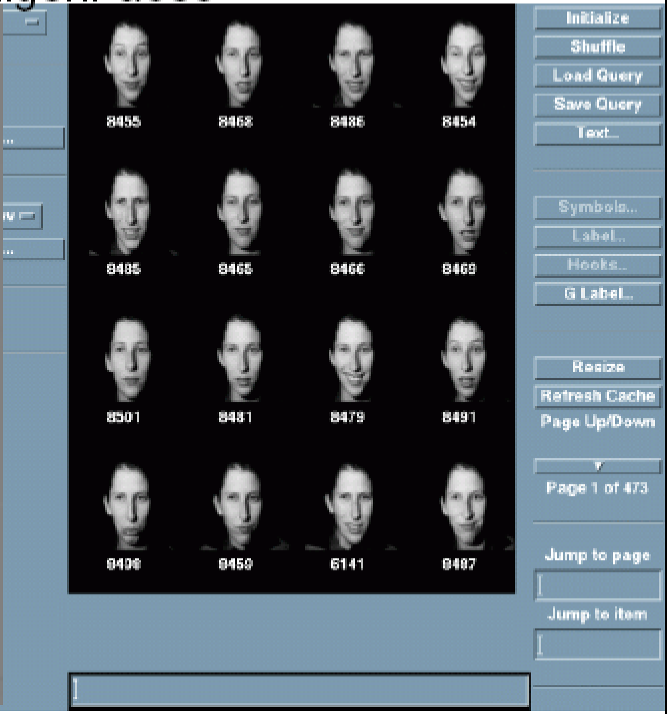


Adapted from David Forsyth, UC Berkeley

Example: EigenFaces



Eigenfaces: Projection on the first 20 eigenvectors from 128 face images



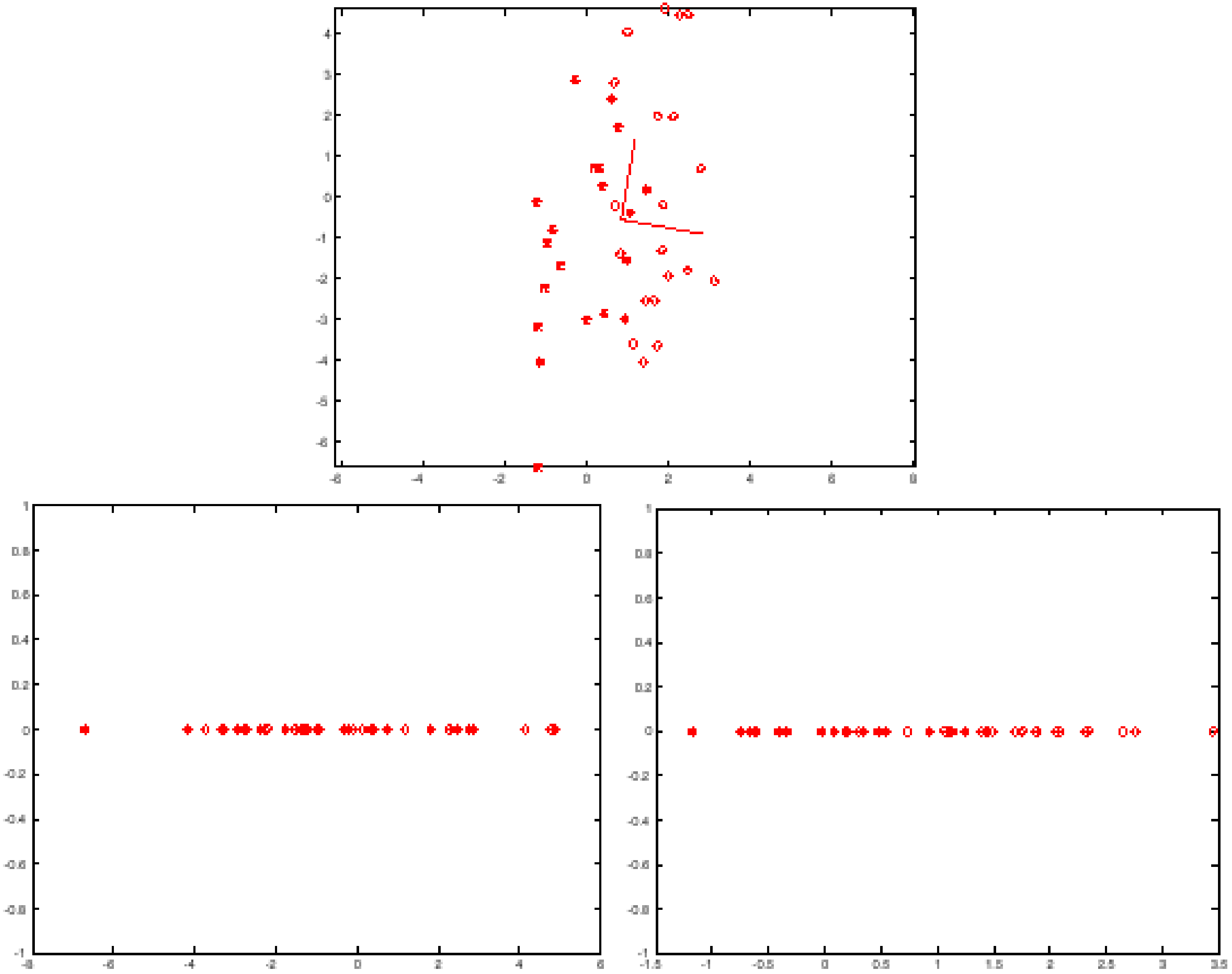
15 most similar faces among 7,562 faces (3000 subjects)

http://www-white.media.mit.edu/vismod/demos/facerec/basic.html

Difficulties with PCA

- Projection may suppress important detail
  - smallest variance directions may not be unimportant
- Method does not take discriminative task into account
  - typically, we wish to compute features that allow good discrimination
  - not the same as largest variance

Adapted from David Forsyth, UC Berkeley

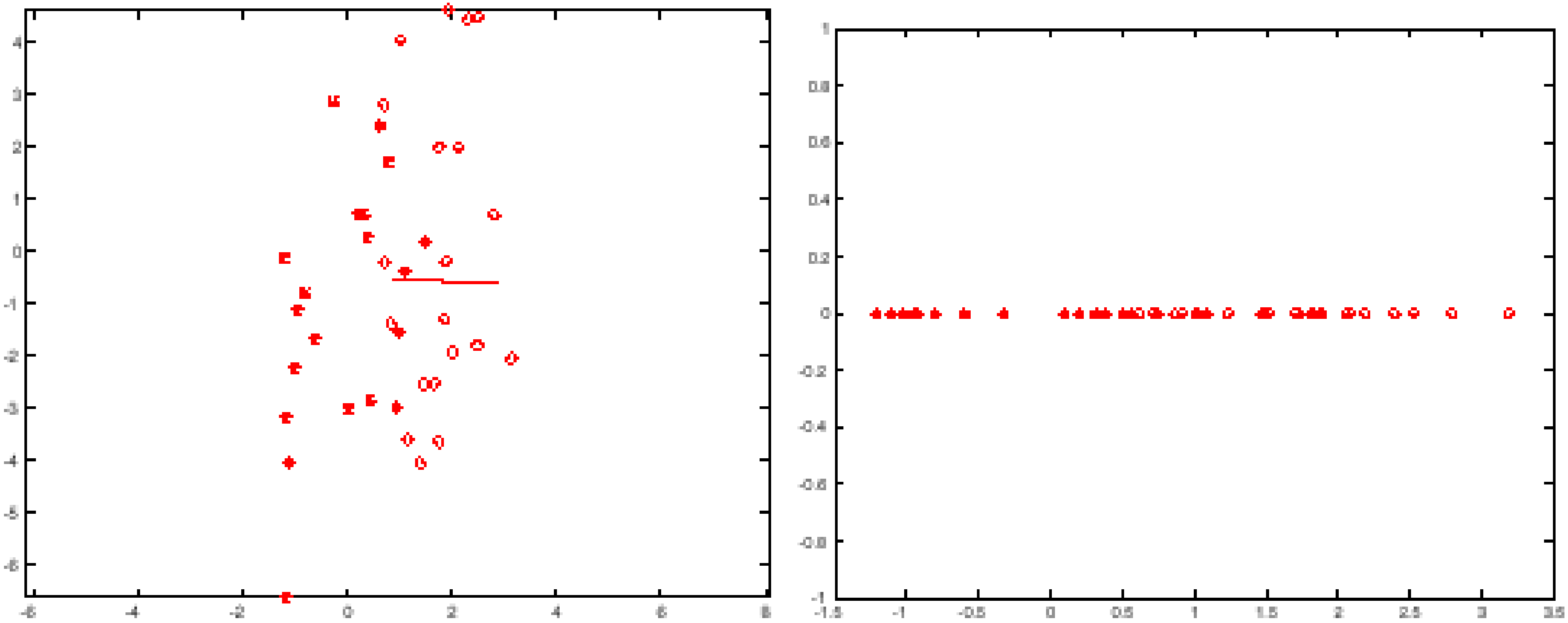


Adapted from David Forsyth, UC Berkeley

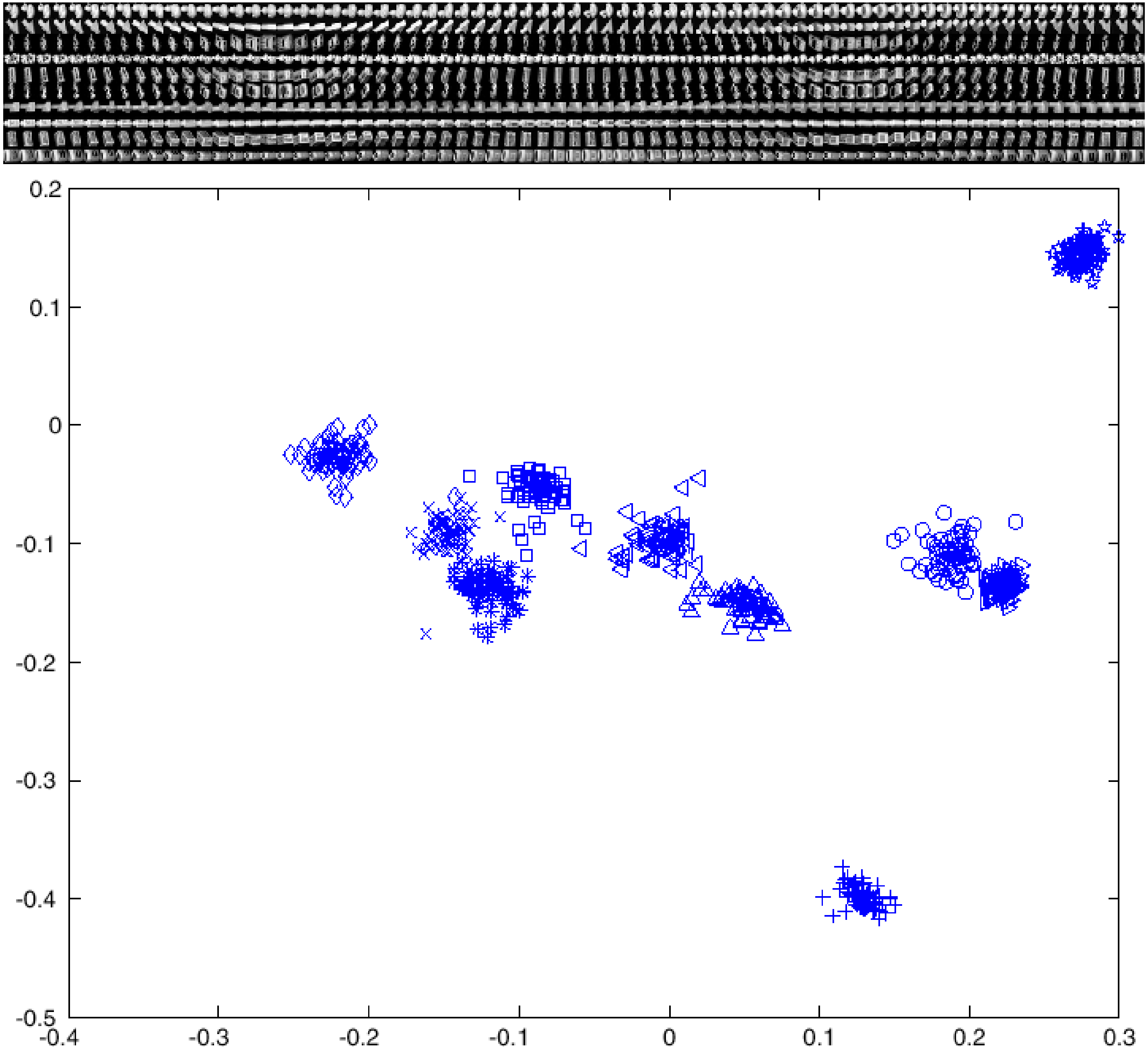
Linear Discriminant Analysis

- We wish to choose linear functions of the features that allow good discrimination.
  - Assume class-conditional covariances are the same
  - Want linear feature that maximises the spread of class means for a fixed within-class variance

Adapted from David Forsyth, UC Berkeley



Adapted from David Forsyth, UC Berkeley

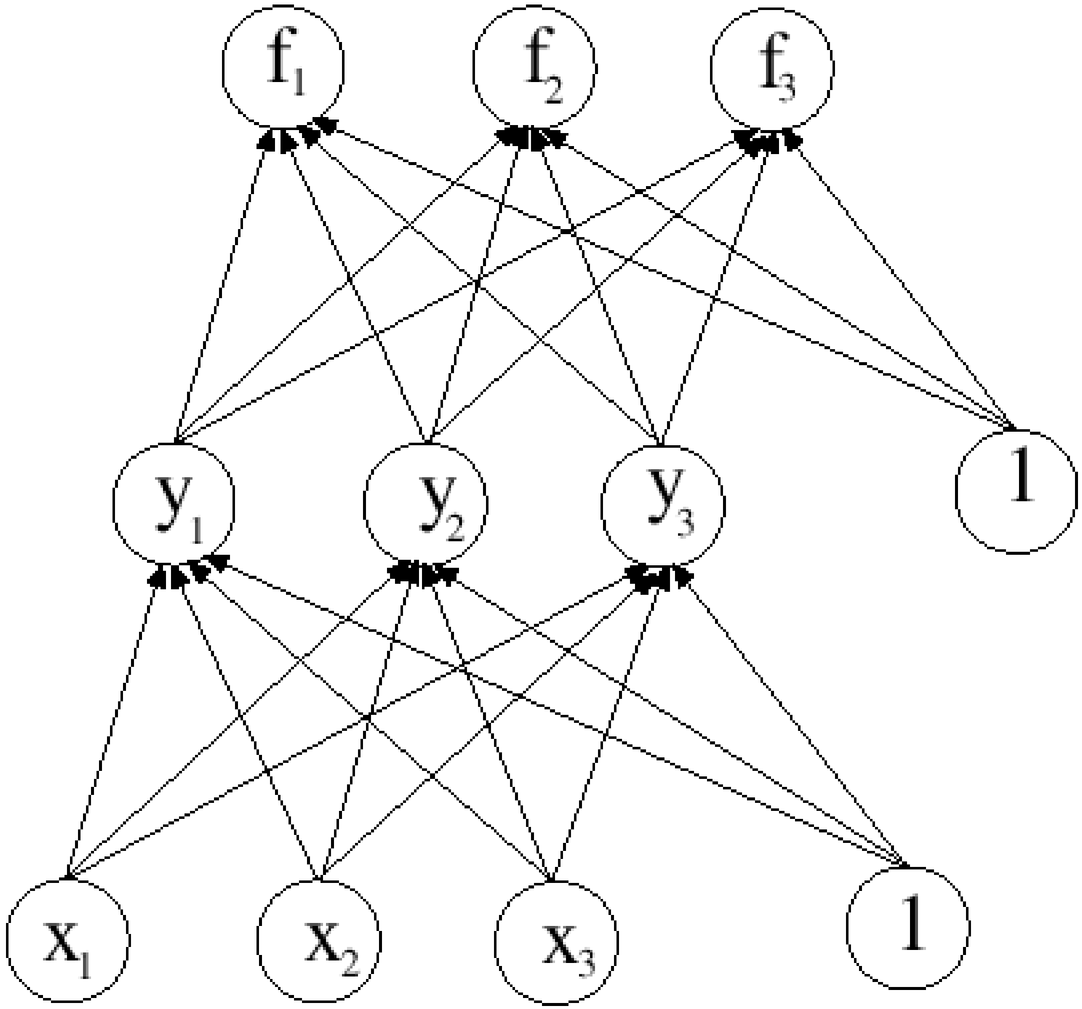


Adapted from David Forsyth, UC Berkeley

Neural networks

- Linear decision boundaries are useful
  - but often not very powerful
  - we seek an easy way to get more complex boundaries
- Compose linear decision boundaries
  - i.e. have several linear classifiers, and apply a classifier to their output
  - a nuisance, because sign(ax+by+cz) etc. isn't differentiable.
  - use a smooth "squashing function" in place of sign.

Adapted from David Forsyth, UC Berkeley



$$\boldsymbol{g}(\boldsymbol{x}) \approx \boldsymbol{f}(\boldsymbol{x}) = [\phi(\boldsymbol{w}_{21} \cdot \boldsymbol{y}), \phi(\boldsymbol{w}_{22} \cdot \boldsymbol{y}), \ldots \phi(\boldsymbol{w}_{2n} \cdot \boldsymbol{y})]$$

$$\boldsymbol{y}(\boldsymbol{z}) = [\phi(\boldsymbol{w}_{11} \cdot \boldsymbol{z}), \phi(\boldsymbol{w}_{12} \cdot \boldsymbol{z}), \ldots \phi(\boldsymbol{w}_{1m} \cdot \boldsymbol{z}), 1]$$

Adapted from David Forsyth, UC Berkeley

$$\boldsymbol{z}(\boldsymbol{x}) = [x_1, x_2, \ldots, x_p, 1]$$

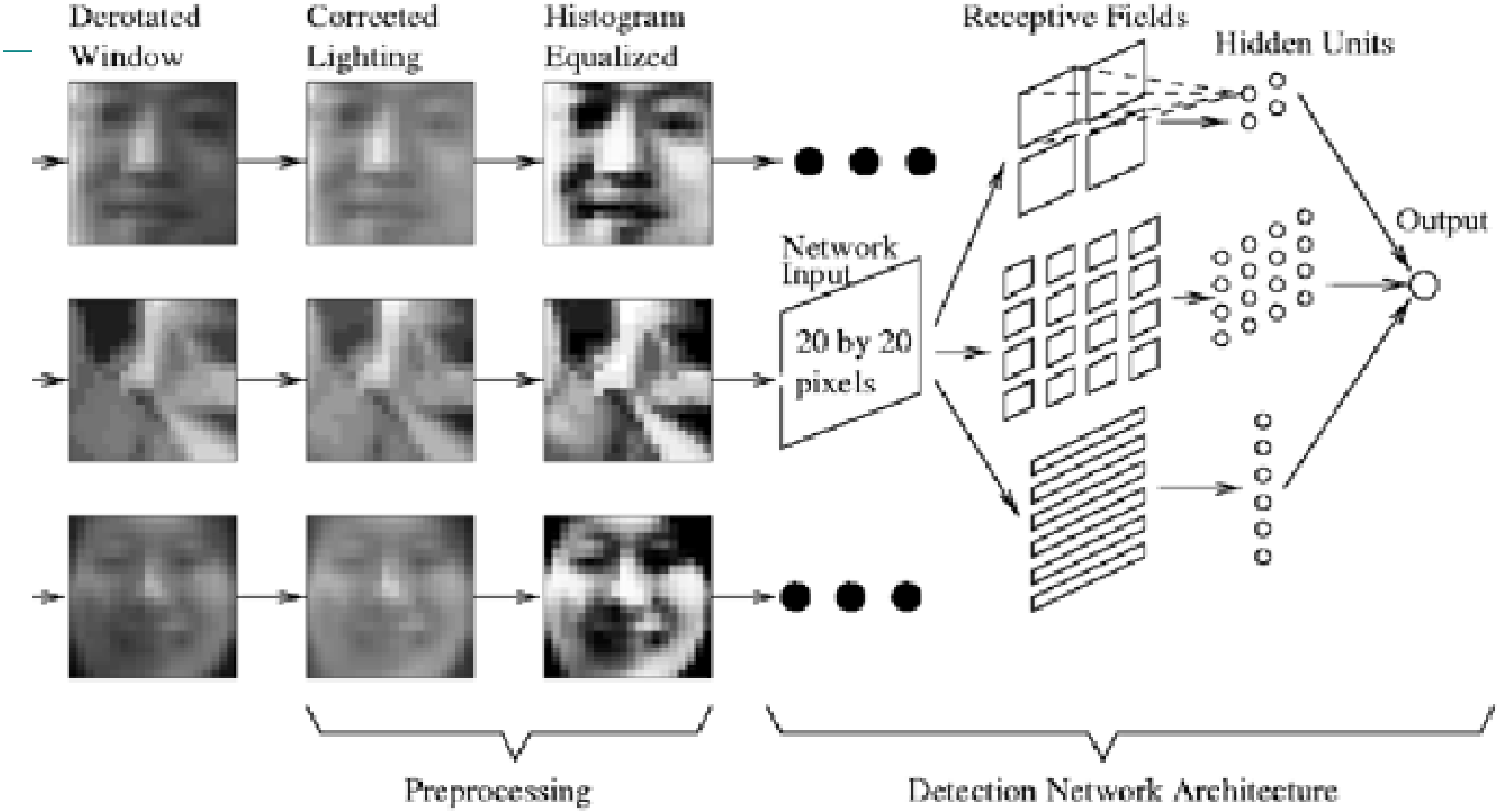
Training

- Choose parameters to minimize error on training set

$$Error(p)=\left(\frac{1}{2}\right)\sum_e\left(n\left(x^e;p\right)-o^e\right)$$

- Stochastic gradient descent, computing gradient using trick (backpropagation, aka the chain rule)
- Stop when error is low, and hasn’t changed much

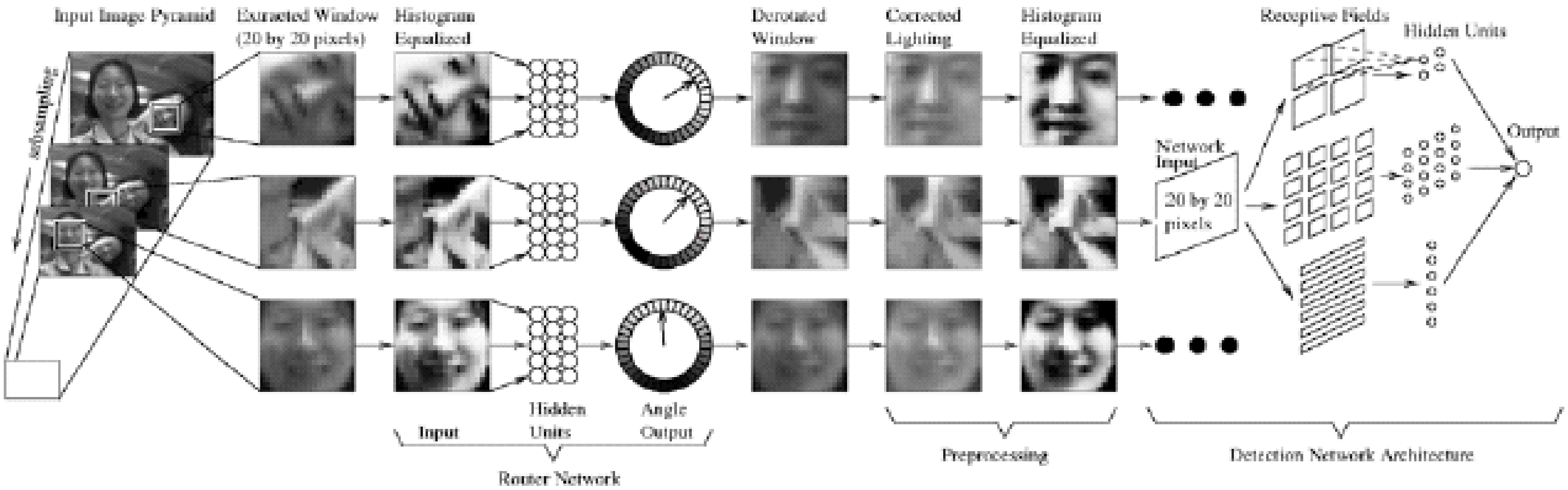
Adapted from David Forsyth, UC Berkeley



The vertical face-finding part of Rowley, Baluja and Kanade's system

Figure from "Rotation invariant neural-network based face detection," H.A. Rowley, S. Baluja and T. Kanade, Proc. Computer Vision and Pattern Recognition, 1998, copyright 1998, IEEE

Adapted from David Forsyth, UC Berkeley



Architecture of the complete system: they use another neural net to estimate orientation of the face, then rectify it. They search over scales to find bigger/smaller faces.

Figure from "Rotation invariant neural-network based face detection," H.A. Rowley, S. Baluja and T. Kanade, Proc. Computer Vision and Pattern Recognition, 1998, copyright 1998, IEEE

Adapted from David Forsyth, UC Berkeley

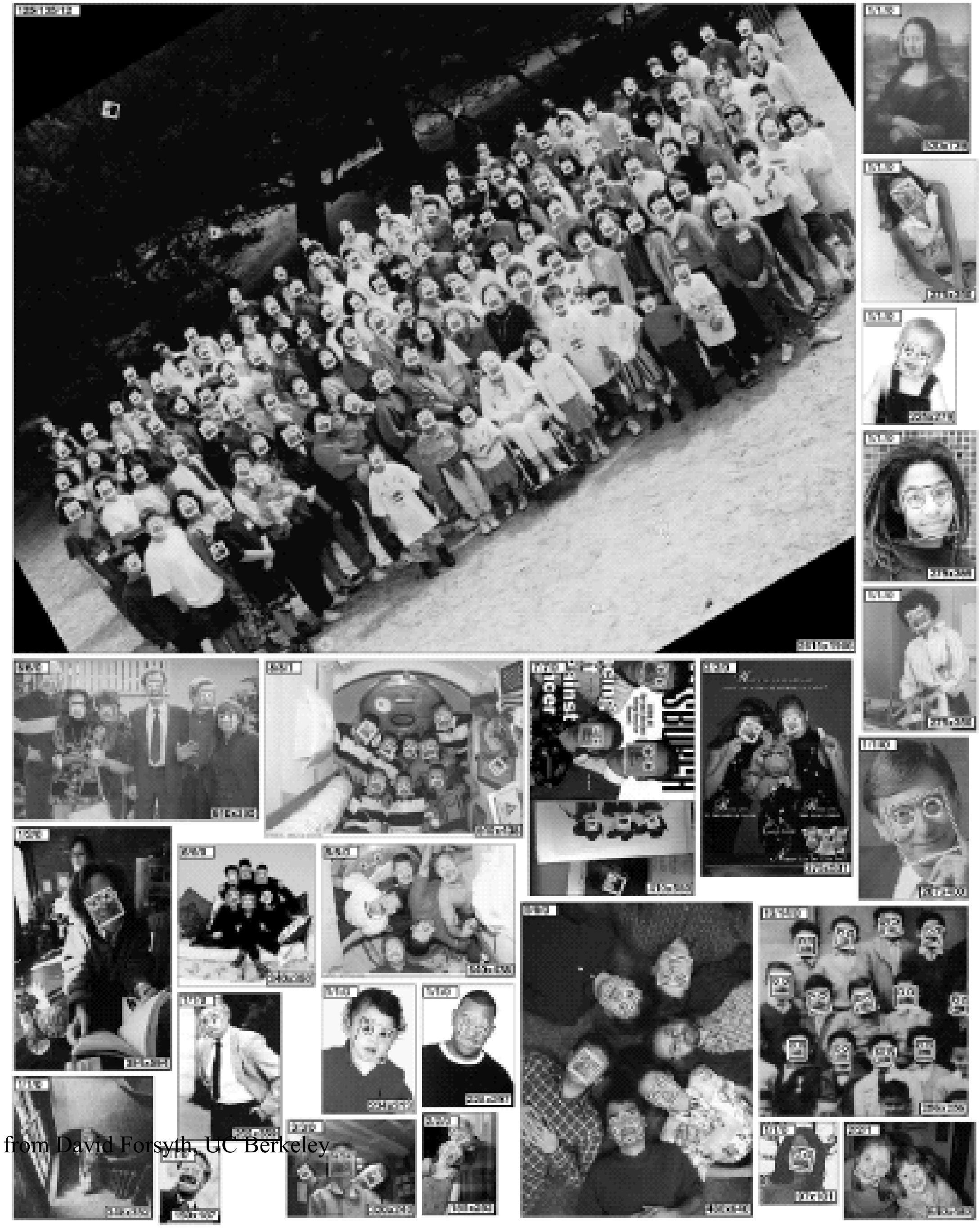


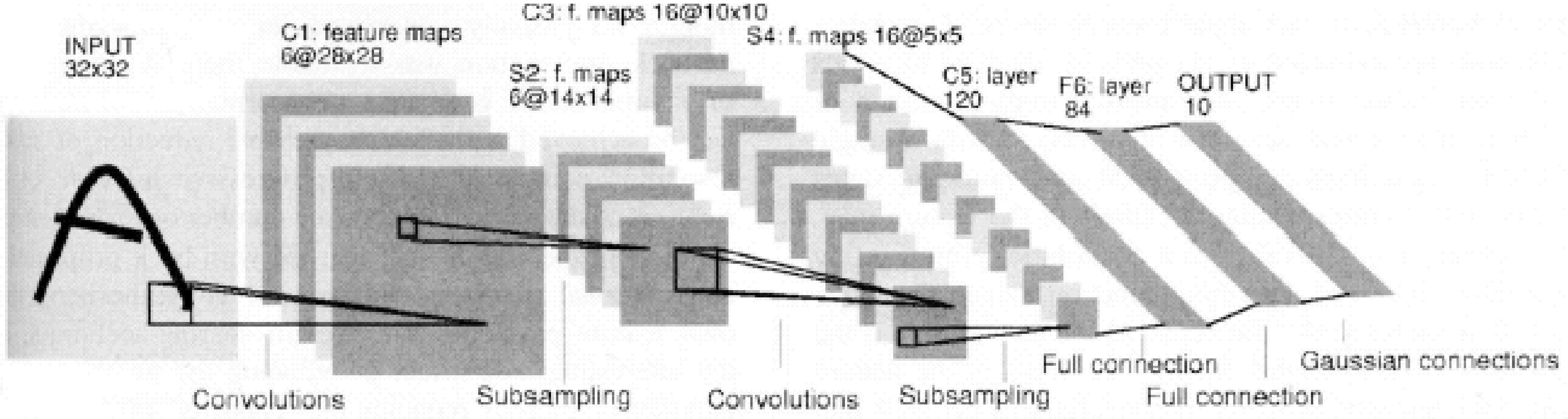
Figure from “Rotation invariant neural-network based face detection,” H.A. Rowley, S. Baluja and T. Kanade, Proc. Computer Vision and Pattern Recognition, 1998, copyright 1998, IEEE

Adapted from David Forsyth, UC Berkeley

Convolutional neural networks

- Template matching using NN classifiers seems to work
- Natural features are filter outputs
  - probably, spots and bars, as in texture
  - but why not learn the filter kernels, too?

Adapted from David Forsyth, UC Berkeley



A convolutional neural network, LeNet; the layers filter, subsample, filter, subsample, and finally classify based on outputs of this process.

Figure from “Gradient-Based Learning Applied to Document Recognition”, Y. Lecun et al Proc. IEEE, 1998 copyright 1998, IEEE

Adapted from David Forsyth, UC Berkeley

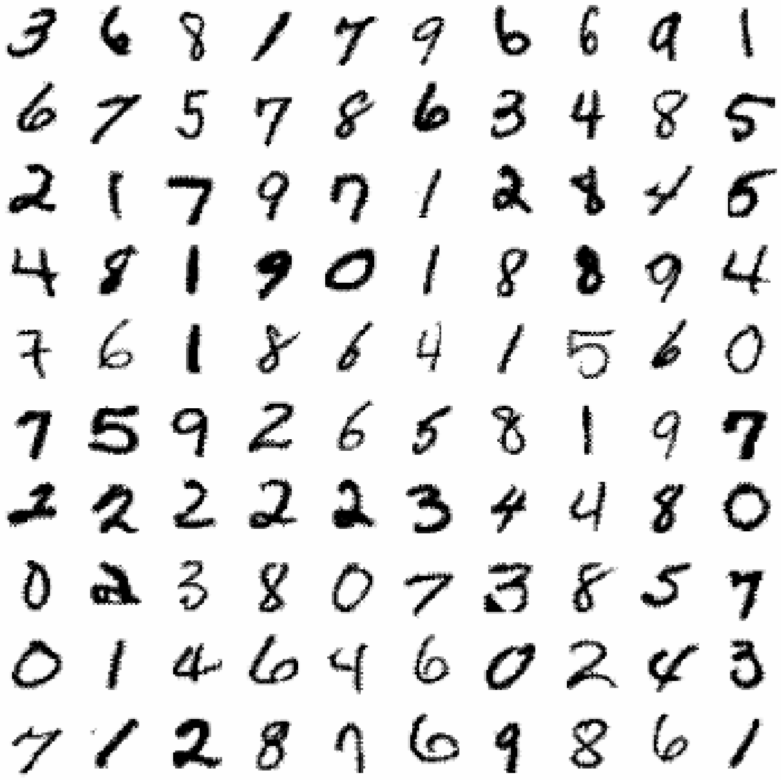


Fig. 4. Size-normalized examples from the MNIST database.

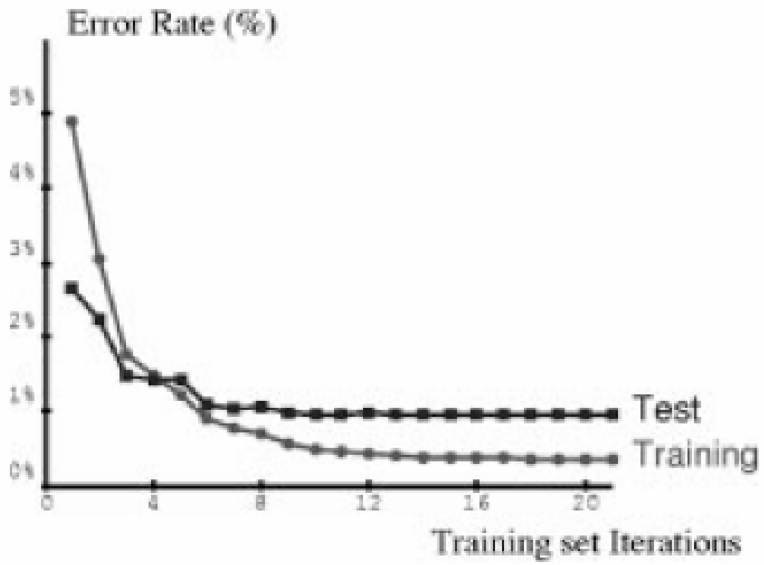


Fig. 5. Training and test error of LeNet-5 as a function of the number of passes through the 60 000 pattern training set (without distortions). The average training error is measured on-the-fly as training proceeds. This explains why the training error appears to be larger than the test error initially. Convergence is attained after 10–12 passes through the training set.

LeNet is used to classify handwritten digits. Notice that the test error rate is not the same as the training error rate, because the test set consists of items not in the training set. Not all classification schemes necessarily have small test error when they have small training error.

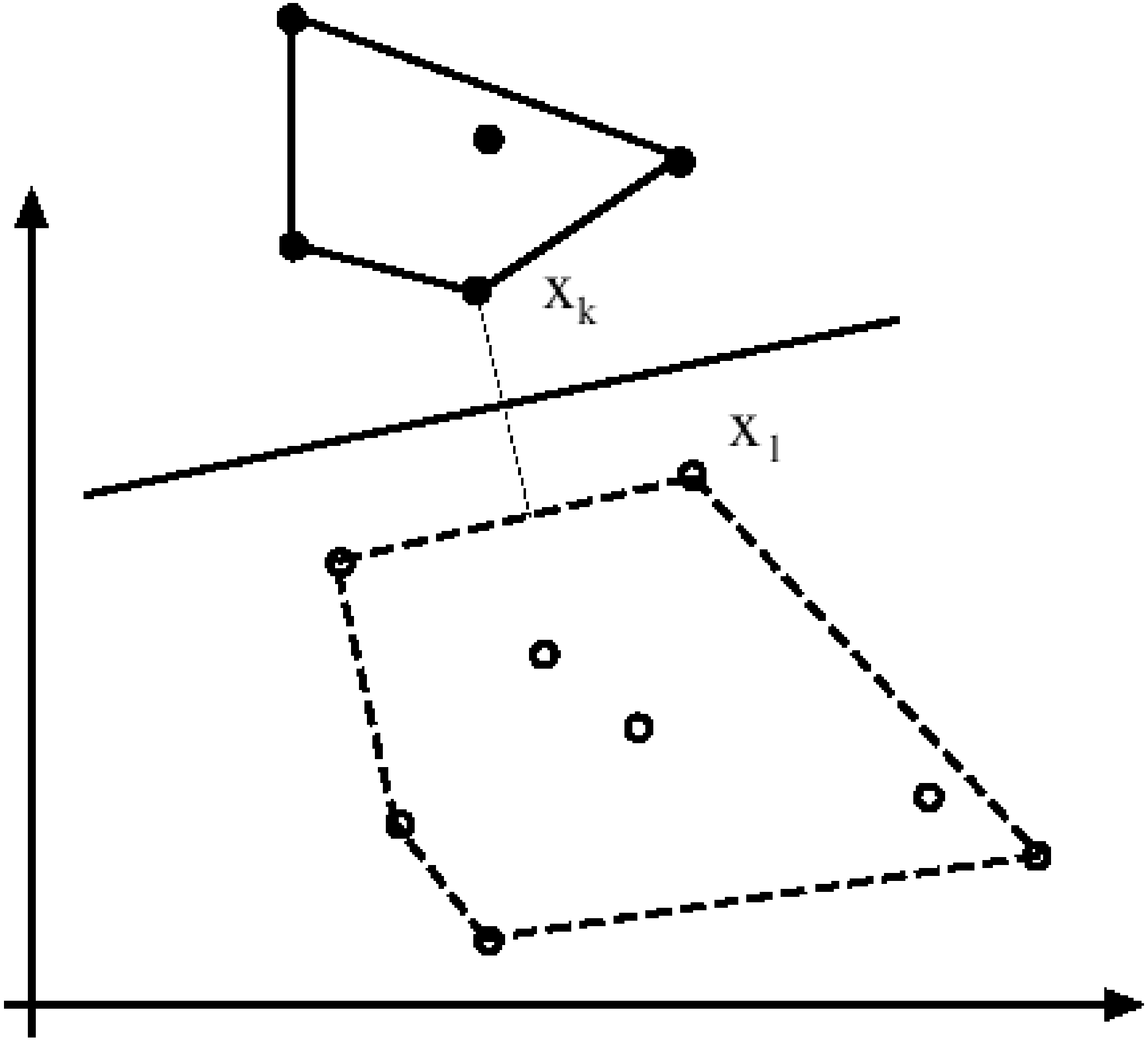
Figure from "Gradient-Based Learning Applied to Document Recognition", Y. Lecun et al Proc. IEEE, 1998 copyright 1998, IEEE

Adapted from David Forsyth, UC Berkeley

Support Vector Machines

- Neural nets try to build a model of the posterior, p(k|x)
- Instead, try to obtain the decision boundary directly
  - potentially easier, because we need to encode only the geometry of the boundary, not any irrelevant wiggles in the posterior.
  - Not all points affect the decision boundary

Adapted from David Forsyth, UC Berkeley



Adapted from David Forsyth, UC Berkeley

Vision applications

- Reliable, simple classifier,
  - use it wherever you need a classifier
- Commonly used for face finding

- Pedestrian finding
  - many pedestrians look like lollipops (hands at sides, torso wider than legs) most of the time
  - classify image regions, searching over scales
  - But what are the features?
  - Compute wavelet coefficients for pedestrian windows, average over pedestrians. If the average is different from zero, probably strongly associated with pedestrian

Adapted from David Forsyth, UC Berkeley

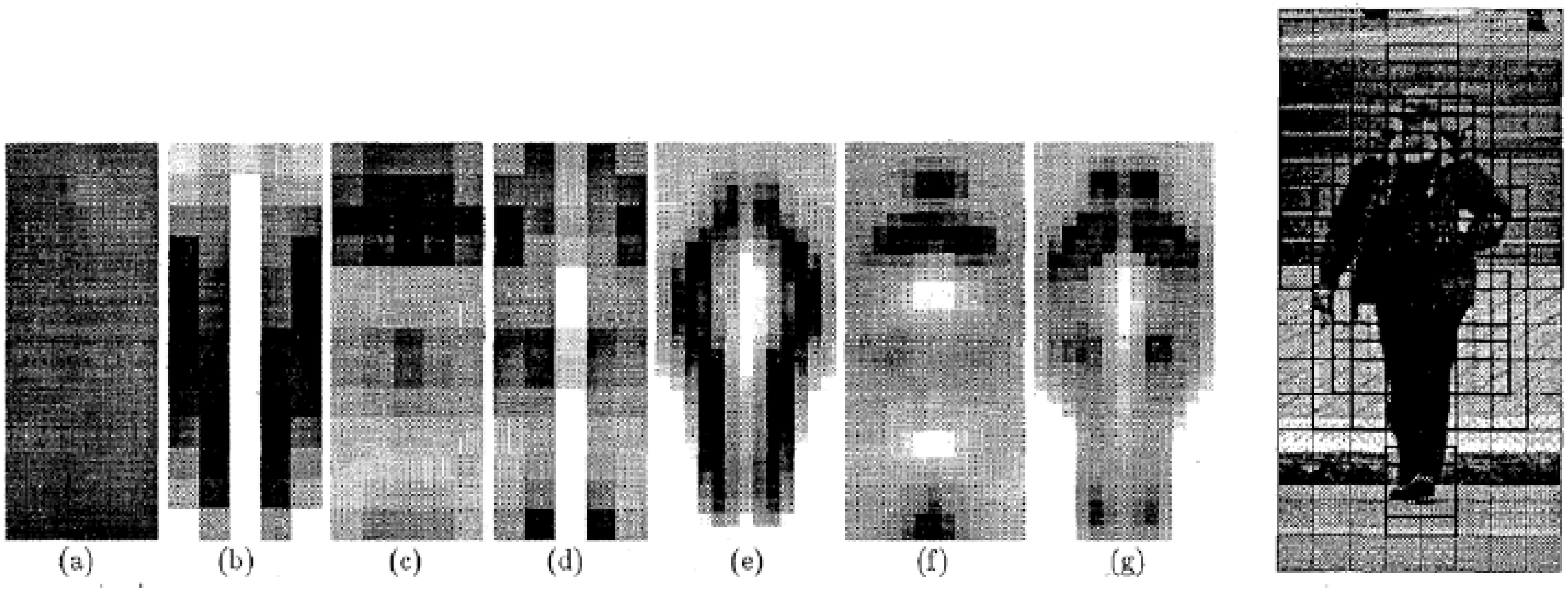


Figure from, “A general framework for object detection,” by C. Papageorgiou, M. Oren and T. Poggio, Proc. Int. Conf. Computer Vision, 1998, copyright 1998, IEEE

Adapted from David Forsyth, UC Berkeley

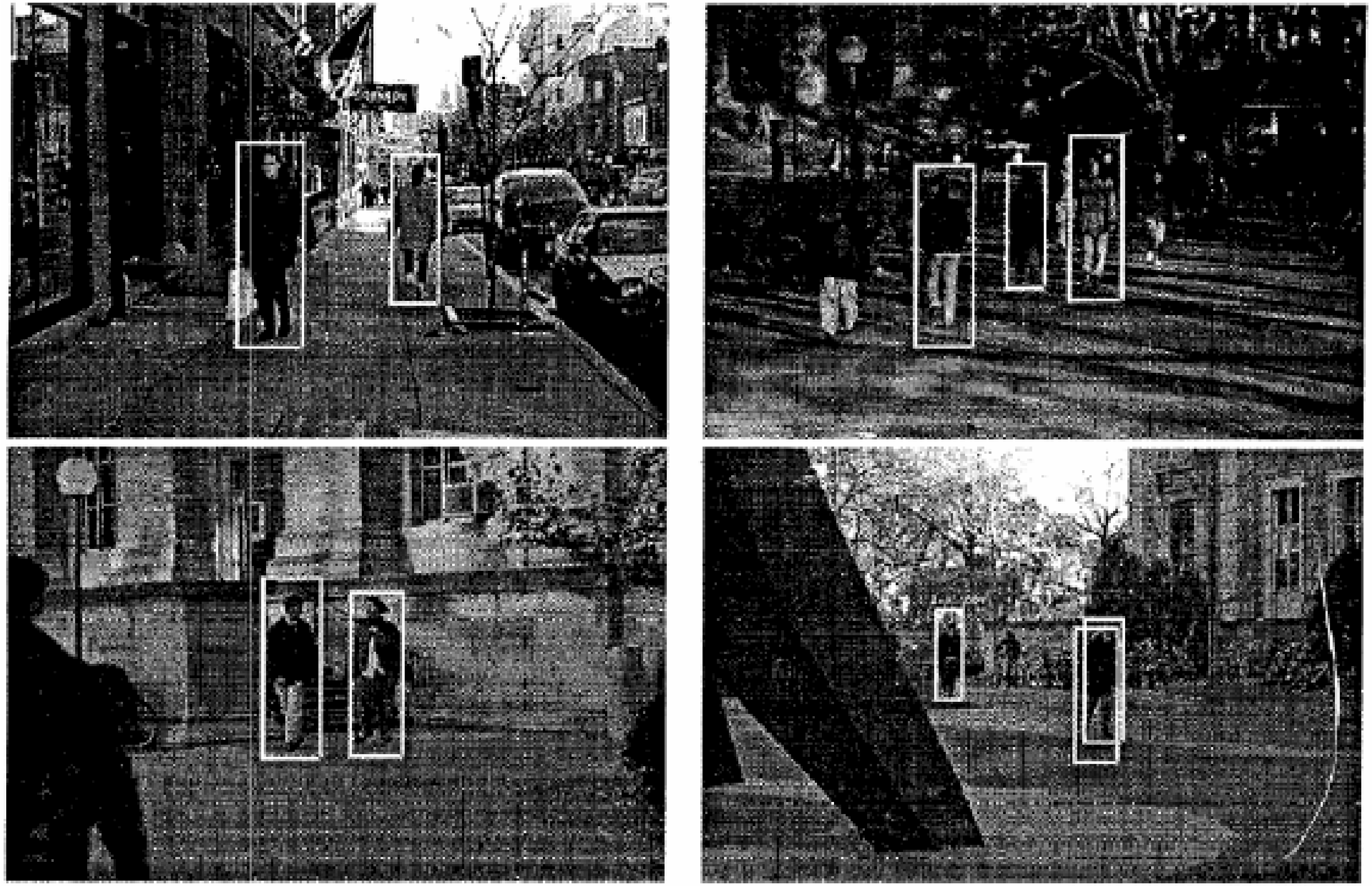


Figure from, “A general framework for object detection,” by C. Papageorgiou, M. Oren and T. Poggio, Proc. Int. Conf. Computer Vision, 1998, copyright 1998, IEEE

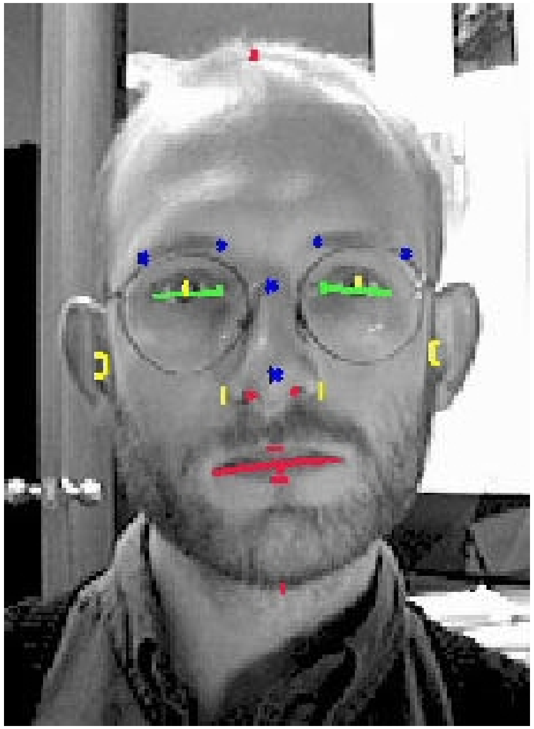
Adapted from David Forsyth, UC Berkeley

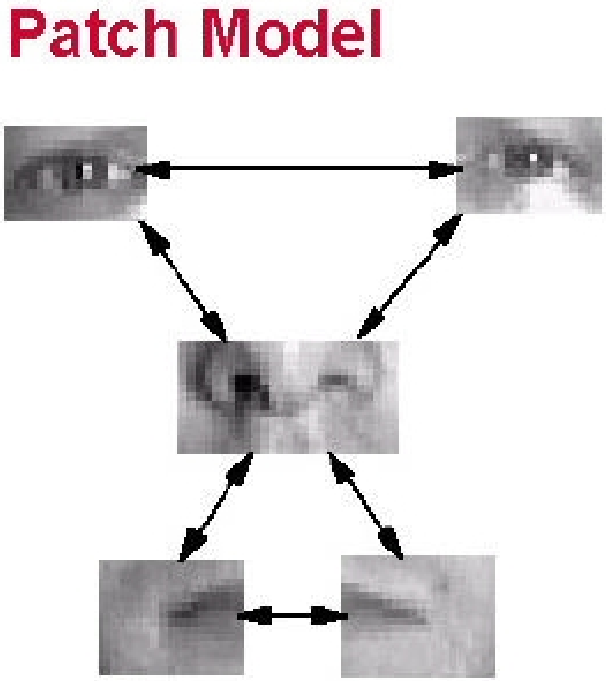
Templates and relations

e.g. find faces by

– finding eyes, nose, mouth

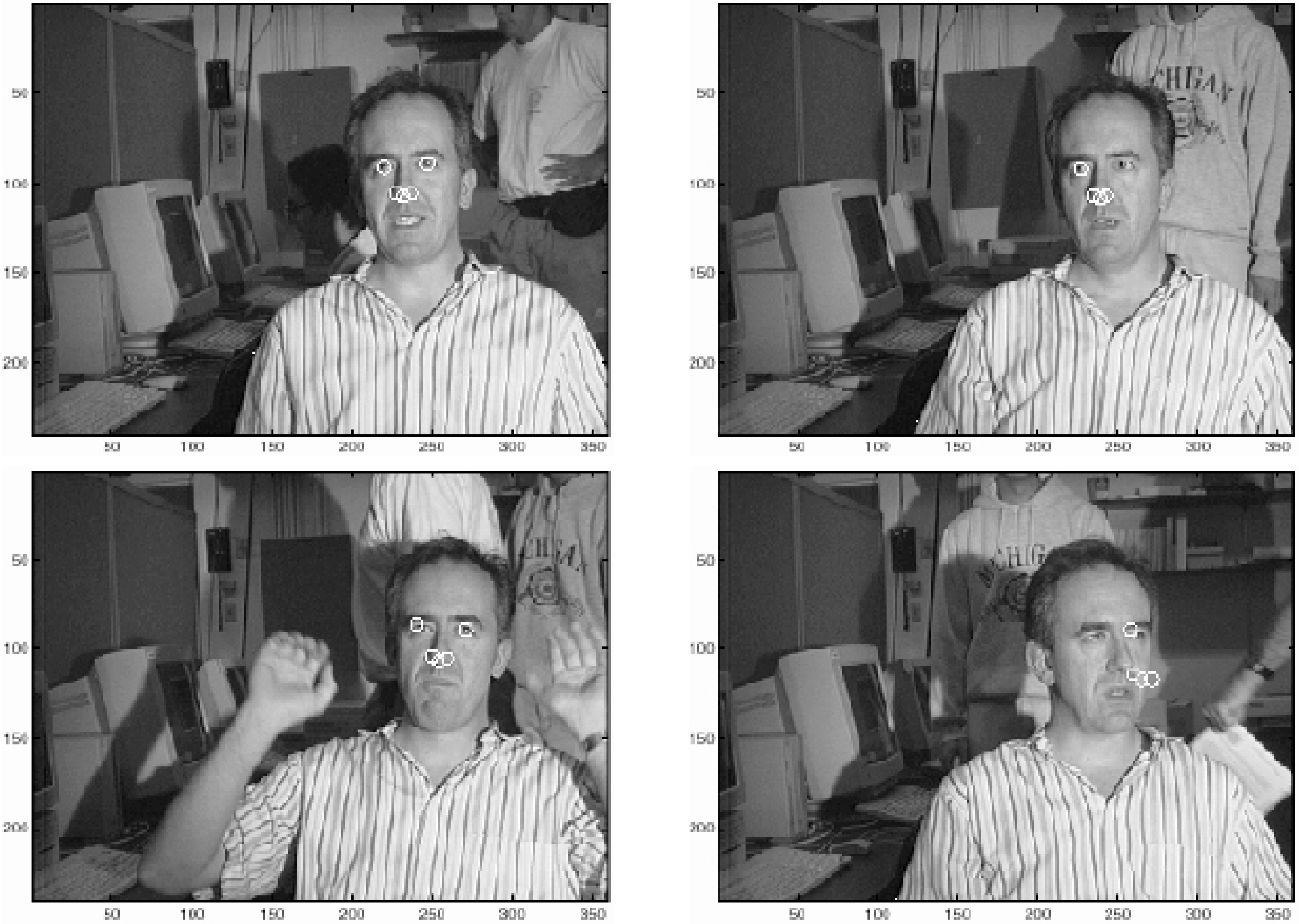
– finding assembly of the three that has the “right” relations





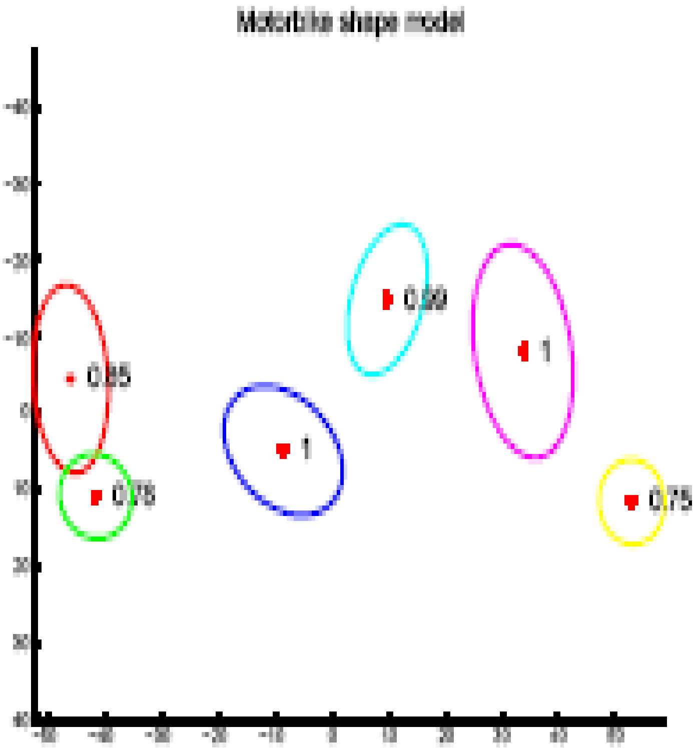
http://www.research.ibm.com/ecvg/biom/facereco.html

adapted from Michael Black, Brown University



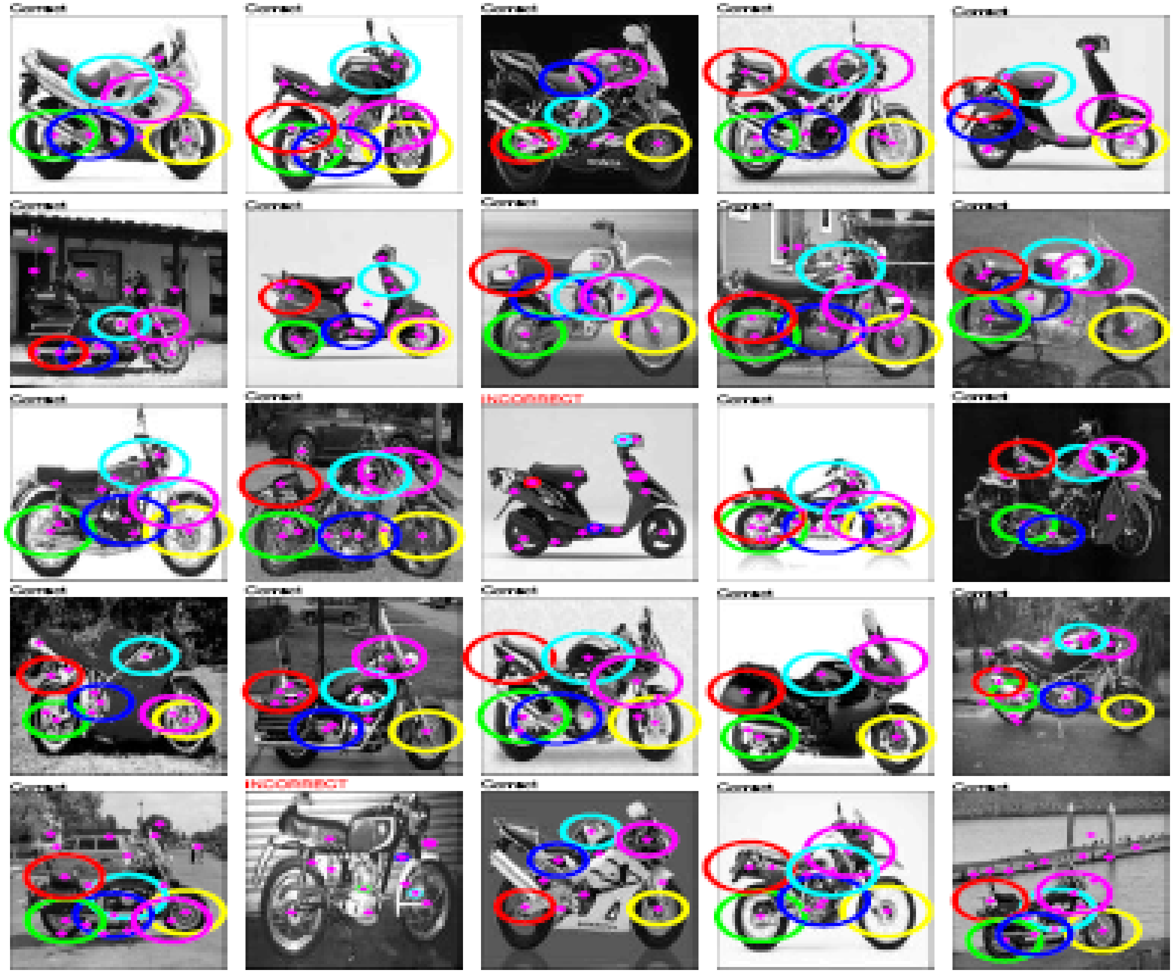
Adapted from David Forsyth, UC Berkeley

Relations between templates



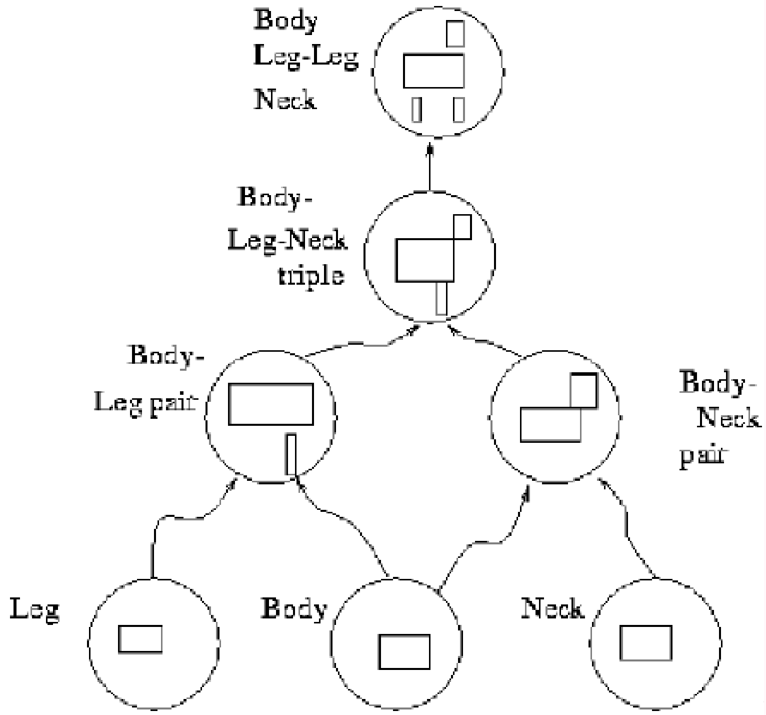
Fergus et al. CVPR 2003

Relations between templates

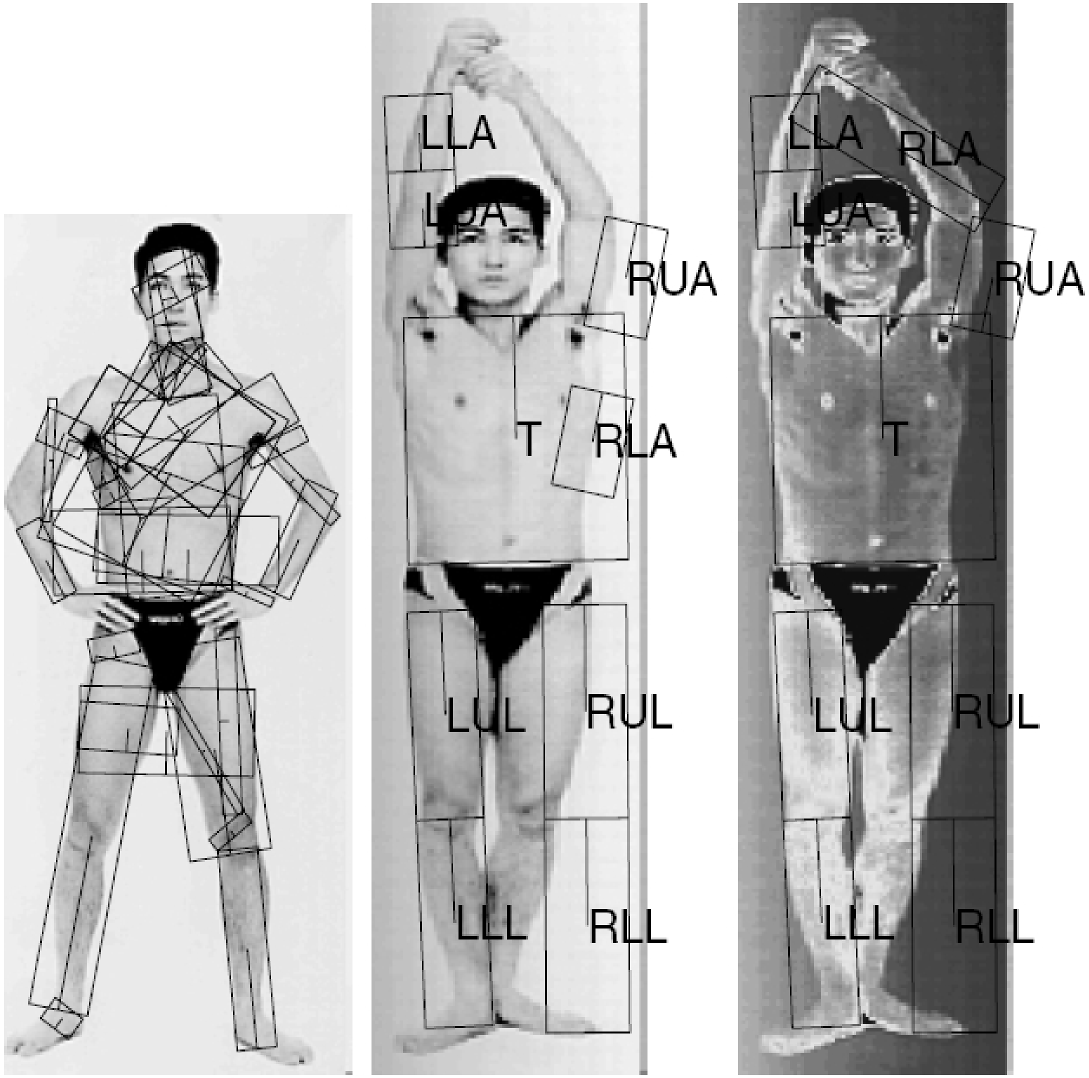


Fergus et al. CVPR 2003

Recognition



adapted from David Forsyth, UC Berkeley



Adapted from David Forsyth, UC Berkeley