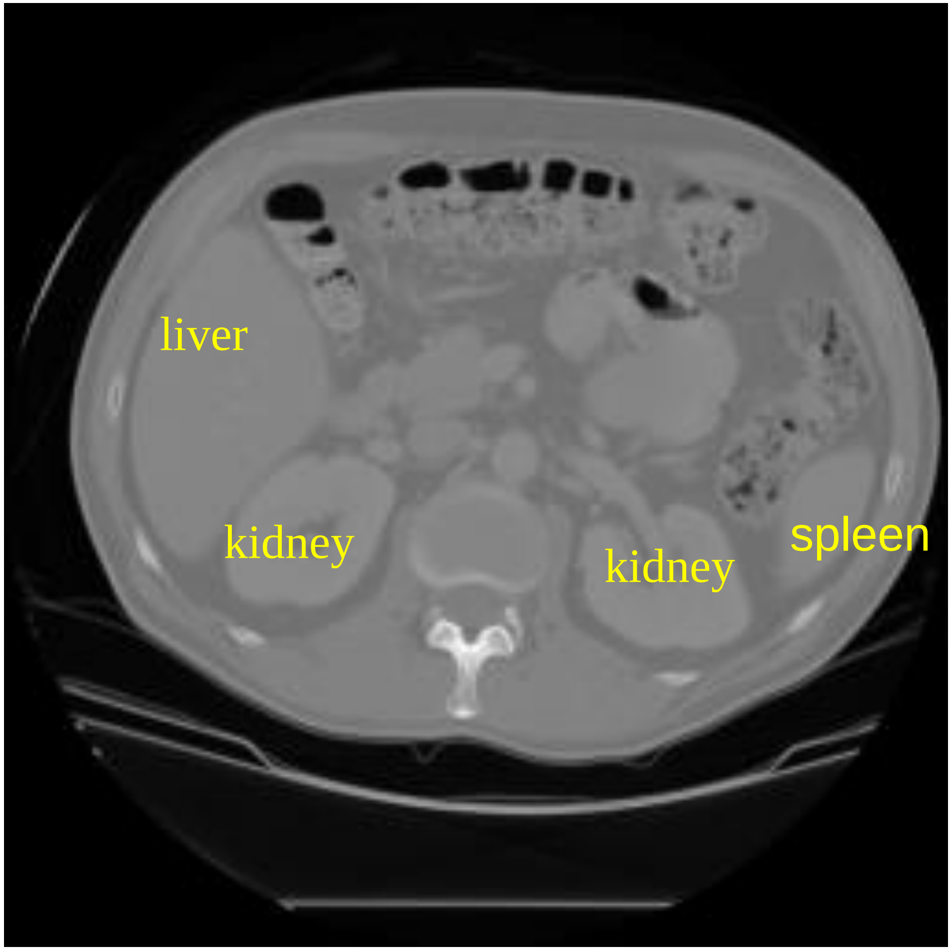
COMPUTER VISION
Introduction

Computer vision is the analysis of digital images by a computer for such applications as:

- Industrial: part localization and inspection, robotics
- Medical: disease classification, screening, planning
- Military: autonomous vehicles, tank recognition
- Intelligence Gathering: face recognition, video analysis
- Security: video analysis
- Science: classification, measurement
- Document Processing: text recognition, diagram conversion

Medical Applications

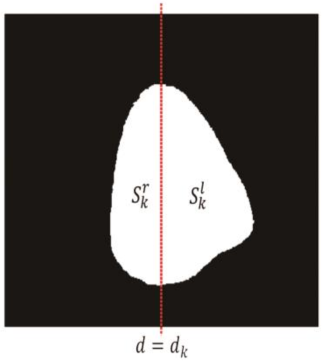
CT image of a patient's abdomen



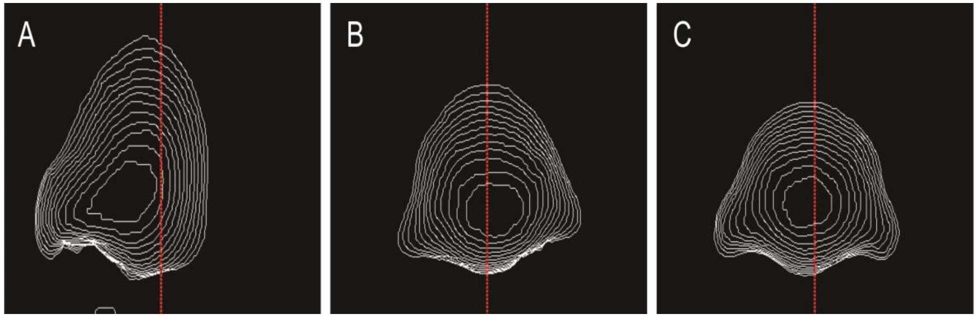
Find the organs to avoid during radiation.

Medical Applications

child with cleft



nose region

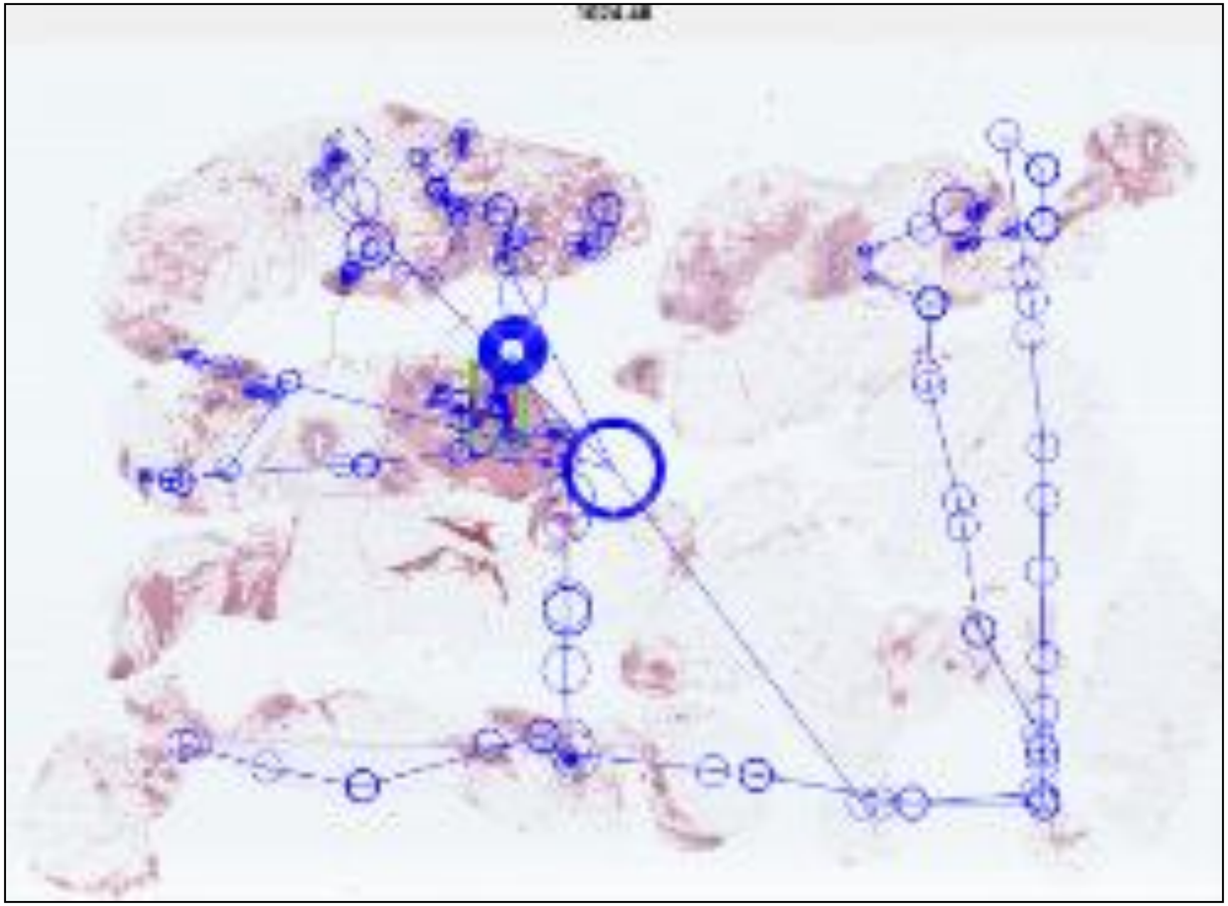


depth area difference

before surgery after surgery control

Medical Applications

Breast Cancer Biopsy Analysis



Robotics

Robot Navigation

Object Recognition

3D Object Reconstruction

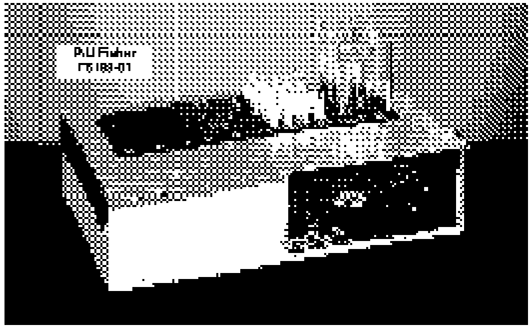
Building Rome in a Day

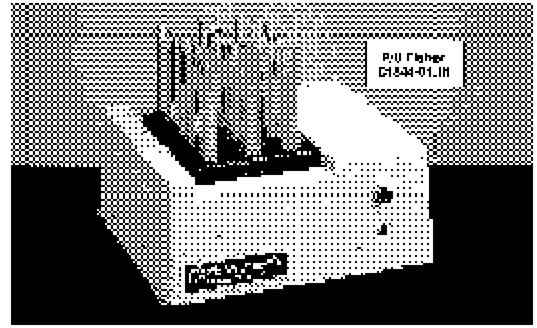
Image Databases:

Images from my Ground-Truth collection.



- Retrieve all images that have trees.
- Retrieve all images that have buildings.
- Retrieve all images that have antelope.





Model 145 Isotemp* Dry Bath Incubator

- *Holds 1 to 4 heating blocks with choice of 11 well sizes*
- *Maintains every sample to within ±0.1° C of temperature*

Unique sample wells are shaped so that aluminum blocks deliver same amount of heat to all parts of the sample tube. No temperature gradient—neither too hot at the bottom nor too cold at the top—that may invalidate tests in baths with drilled cylindrical wells. Sample tubes rest on insulating plugs to prevent localized heating. A low watt-density heater is mounted on a thick aluminum heat distribution plate in the floor of the bath. Plate is ⅜" thick (9.5 mm). Dry bath minimizes cleanup problems because tubes stay dry.

Ambient to 125° C (257° F) with ±0.1° C control. Dial temperature control graduated from 25° to 125° C. Ideal for enzyme reactions, inactivation of sera, Rh studies, blood crossmatching and thrombin determinations. Dimensions: § L x 15 ¾ W x 4" H (23 x 38 x 10 cm). With line cord and plug. **Heating blocks sold separately** (see lower right).

Electrical Requirements	Cat. No.	Each
120V 50/60 Hz, 800W (CSA approved)	11-715-100	[illegible]
240V 50/60 Hz, 800W	11-715-101	395.25

Average deviation from control 37° C
Production Model

Incu-Block* Partial Immersion Thermometers

For all standard heating blocks and water baths. Critical temperatures (25°, 30°, 37°, 56° C) are marked with arrows. Available with shatter-proof, contamination-proof Teflon* coating. Total length: 175 mm. Immersion: 35 mm.

Range, °C	Div., °C	Teflon-Coated	Cat. No.	Each
25-57	0.5	No	14-992	45.24
25-57	0.5	Yes	14-993	45.15

More Thermometers

For more thermometers, including digital types,

see page 952

Model 147 Isotemp* Dry Bath

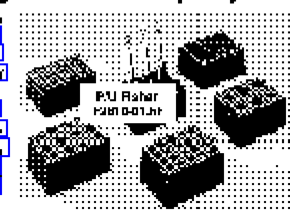
- *Holds single heating block with choice of 11 well sizes*

Similar to Model 145, but with ⅜" thick (9.5 mm) plate. Ideal for labs with smaller volumes of enzyme and coagulant assays, Rh studies, and dry incubations. Screwdriver-adjusted temperature control between ambient and 95° C (203° F). Observe thermometer placed in unused sample tube to set, adjust control through hole in front panel. Maintains set temperature with consistency and uniformity of ±0.5° C.

Supplied with strong nylon case, thermostatically controlled heater, pilot indicator lamp, line cord and plug, and instructions. Dimensions: 6 L x 6¾ W x 3" H (15 x 17 x 8 cm). CSA approved. **Heating blocks sold separately** (see below).

Electrical Requirements	Cat. No.	Each
120V 50/60 Hz, 120W	11-715-102	223.59

Interchangeable Heating Blocks for Isotemp* Dry Baths



For Models 145 and 147 Dry Baths. Composed of black anodized aluminum alloy. Chemical-resistant. Dimensions: 4 L x 3 W x 2" H (10 x 7.5 x 5 cm).

The 11-715-123 block provides a safe, dry bath alternative for warming the 3 pairs of tissue forceps. Avoids hazardous use of burners around flammable histological solvents.

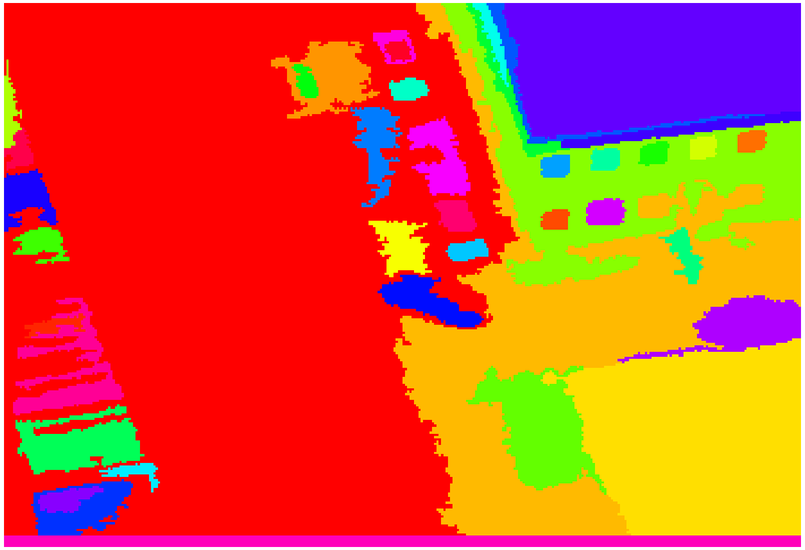
The 11-715-120 block is specifically designed to hold twenty 9.5 mm Roche Diagnostics Placentex* pregnancy test tubes. This special shallow well block is similar to the other blocks with 10 mm holes, but sample wells are only ⅜" deep (1.0 cm) to meet test requirements. Wells in all other blocks are 1¾" deep (4.4 cm).

Tube Size, mm	Wells/Block	Cat. No.	Each
6	35	11-715-105	71.[illegible]
10	20	11-715-107	71.[illegible]
9.5	20 (shallow)	11-715-120	71.[illegible]
12	12	11-715-109	71.[illegible]
12.5†	12	11-715-121	71.[illegible]
13	12	11-715-111	71.[illegible]
15	12	11-715-113	71.[illegible]
16‡	9	11-715-123	71.[illegible]
18	12	11-715-115	71.[illegible]
20	6	11-715-117	71.[illegible]
25	5	11-715-119	71.[illegible]

†Square [illegible]
‡For warming tissue forceps [illegible]

Surveillance: Object and Event Recognition in Aerial Videos

Original Video Frame



Color Regions

Structure Regions

The Three Stages of Computer Vision

- low-level (image processing)

  image $\longrightarrow$ image

- mid-level (feature extraction)

  image $\longrightarrow$ features

- high-level (the intelligent part)

  features $\longrightarrow$ analysis

High-Level Computer Vision

- Detection of classes of objects (faces, motorbikes, trees, cheetahs) in images
- Recognition of specific objects such as George Bush or machine part #45732
- Classification of images or parts of images for medical or scientific applications
- Recognition of events in surveillance videos
- Measurement of distances for robotics

High-level vision uses techniques from AI

- Graph-Matching: A*, Constraint Satisfaction, Branch and Bound Search, Simulated Annealing
- Learning Methodologies: Decision Trees, Neural Nets, SVMs, EM Classifier
- Probabilistic Reasoning, Belief Propagation, Graphical Models

Graph Matching for Object Recognition

- For each specific object, we have a geometric model.
- The geometric model leads to a symbolic model in terms of image features and their spatial relationships.
- An image is represented by all of its features and their spatial relationships.
- This leads to a graph matching problem.

Model-based Recognition as Graph Matching (Constraint Satisfaction)

- Let U = the set of model features.
- Let R be a relation expressing their spatial relationships.
- Let L = the set of image features.
- Let S be a relation expressing their spatial relationships.
- The ideal solution would be a subgraph isomorphism f: U-> L satisfying
- if $(u_1, u_2, \ldots, u_n) \varepsilon R$, then $(f(u_1), f(u_2), \ldots, f(u_n)) \varepsilon S$

House Example

2D model

2D image

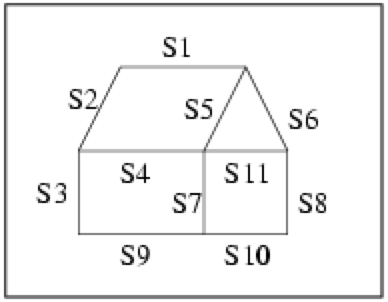


Image 1

P

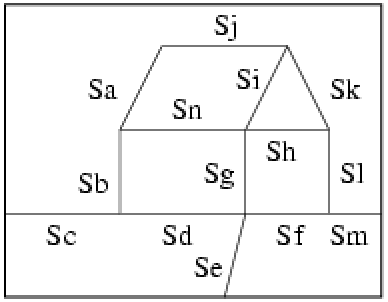


Image 2

L

RP and RL are connection relations.

$P = \{S1,S2,S3,S4,S5,S6,S7,S8,S9,S10,S11\}$.

$L = \{Sa,Sb,Sc,\cancel{Sd,Se},Sf,Sg,Sh,Si,Sj,Sk,Sl,Sm\}$.

$R_P = \{$ (S1,S2), (S1,S5), (S1,S6), (S2,S3), (S2,S4), (S3,S4), (S3,S9), (S4,S5), (S4,S7), (S4,S11), (S5,S6), (S5,S7), (S5,S11), (S6,S8), (S6,S11), (S7,S9), (S7,S10), (S7,S11), (S8,S10), (S8,S11), (S9,S10) $\}$.

$R_L = \{$ (Sa,Sb), (Sa,Sj), (Sa,Sn), (Sb,Sc), (Sb,Sd), (Sb,Sn), (Sc,Sd), (Sd,Se), (Sd,Sf), (Sd,Sg), (Se,Sf), (Se,Sg), (Sf,Sg), (Sf,Sl), (Sf,Sm), (Sg,Sh), (Sg,Si), (Sg,Sn), (Sh,Si), (Sh,Sk), (Sh,Sl), (Sh,Sn), (Si,Sj), (Si,Sk), (Si,Sn), (Sj,Sk), (Sk,Sl), (Sl,Sm) $\}$.

f(S1)=Sj
f(S2)=Sa
f(S3)=Sb

f(S4)=Sn
f(S5)=Si
f(S6)=Sk

f(S7)=Sg
f(S8) = Sl
f(S9)=Sd

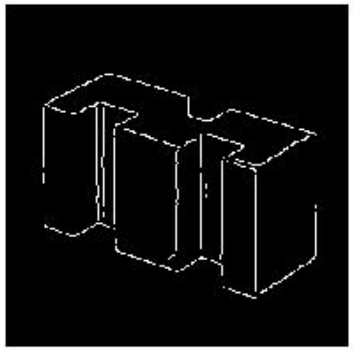
f(S10)=Sf
f(S11)=Sh

But this is too simplistic

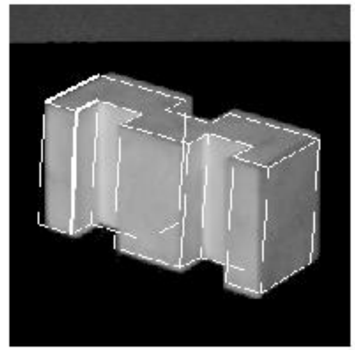
- The model specifies all the features of the object that may appear in the image.
- Some of them don’t appear at all, due to occlusion or failures at low or mid level.
- Some of them are broken and not recognized.
- Some of them are distorted.
- Relationships don’t all hold.
- We need some kind of inexact matching.

1st Try: TRIBORS: view class matching of polyhedral objects

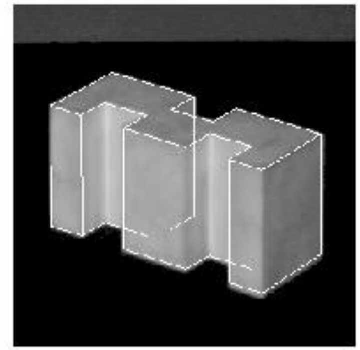
edges from image



model overlayed



improved location



- A view-class is a typical 2D view of a 3D object.

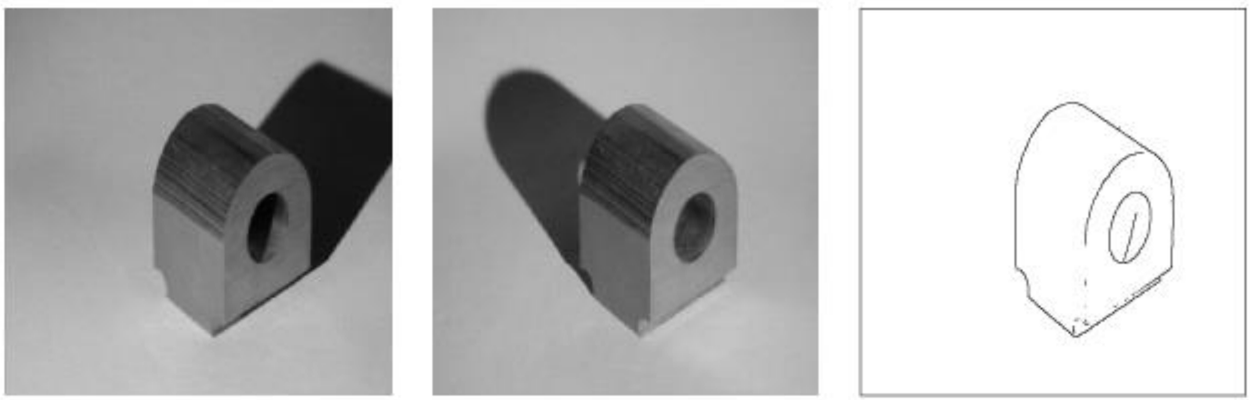
- Each object had 4-5 view classes (hand selected).

- The representation of a view class for matching included:
  - triplets of line segments visible in that class
  - the probability of detectability of each triplet

The first version of this program used iterative-deepening A* search.
STILL TOO MUCH OF A TOY PROBLEM.

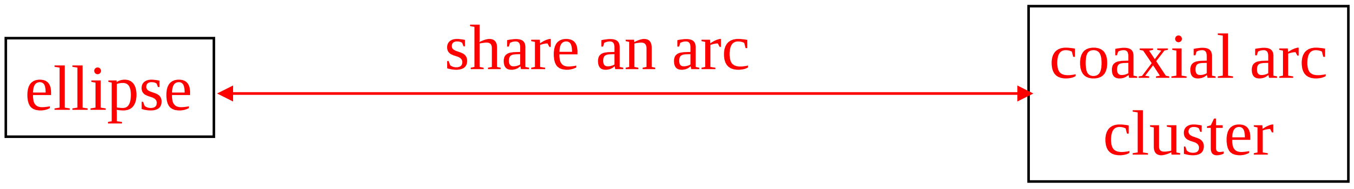
RIO: Relational Indexing for Object Recognition

- RIO worked with more complex parts that could have
  - planar surfaces
  - cylindrical surfaces
  - threads

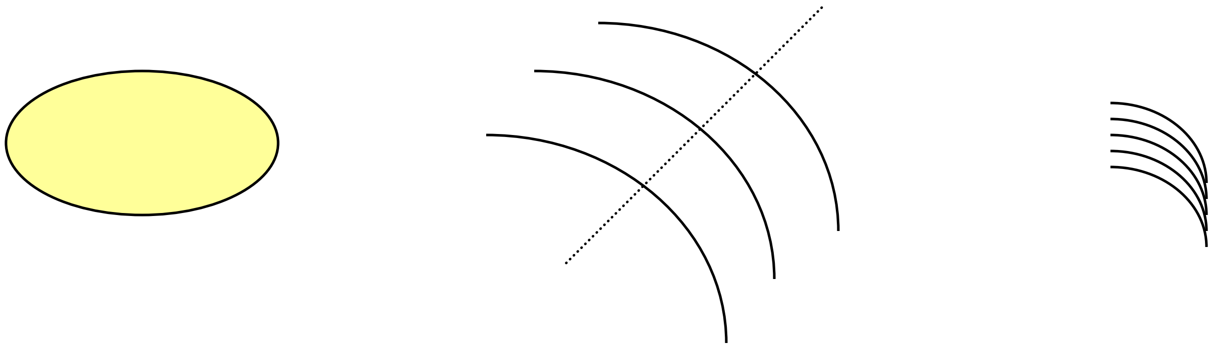


Object Representation in RIO

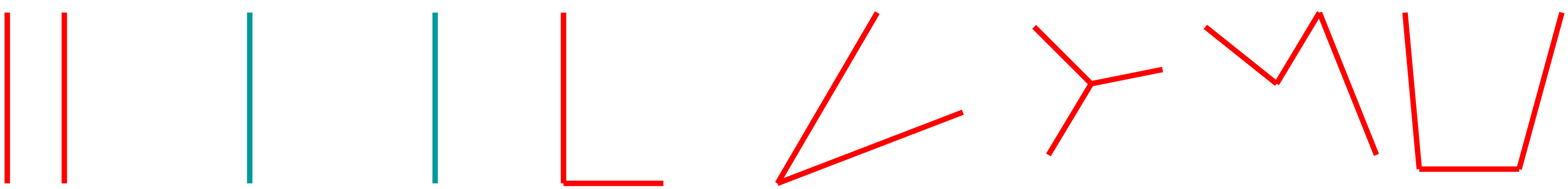
- 3D objects are represented by a 3D mesh and set of 2D view classes.
- Each view class is represented by an attributed graph whose nodes are features and whose attributed edges are relationships.
- For purposes of indexing, attributed graphs are stored as sets of 2-graphs, graphs with 2 nodes and 2 relationships.



RIO Features



ellipses coaxials coaxials-multi



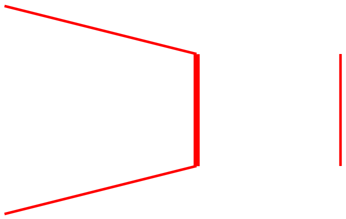
parallel lines close and far

junctions L V

Y Z

triples U

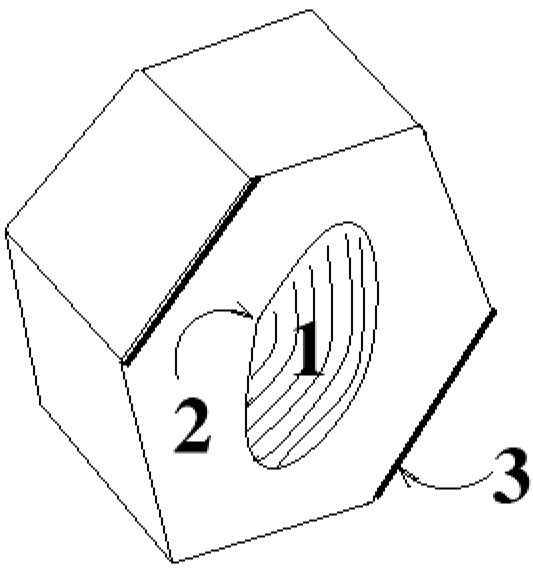
RIO Relationships



- share one arc
- share one line
- share two lines
- coaxial
- close at extremal points
- bounding box encloses / enclosed by

Hexnut Object

MODEL-VIEW



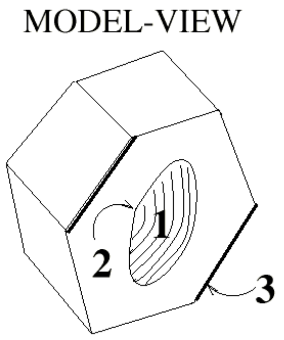
RELATIONS:
a: encloses
b: coaxial

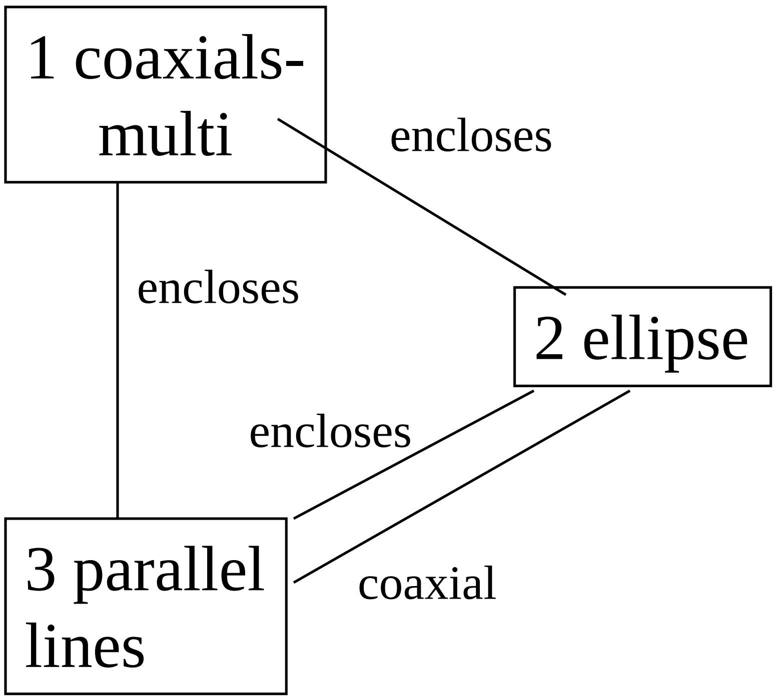
FEATURES:
1: coaxials-multi
2: ellipse
3: parallel lines

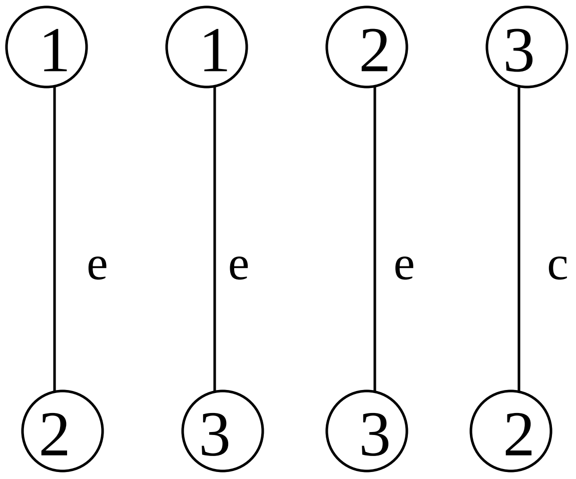
How are 1, 2, and 3 related?

What other features and relationships can you find?

Graph and 2-Graph Representations







RDF!

Relational Indexing for Recognition

Preprocessing (off-line) Phase

for each model view Mi in the database

- encode each 2-graph of Mi to produce an index
- store Mi and associated information in the indexed bin of a hash table H

Matching (on-line) phase

1. Construct a relational (2-graph) description D for the scene

2. For each 2-graph G of D

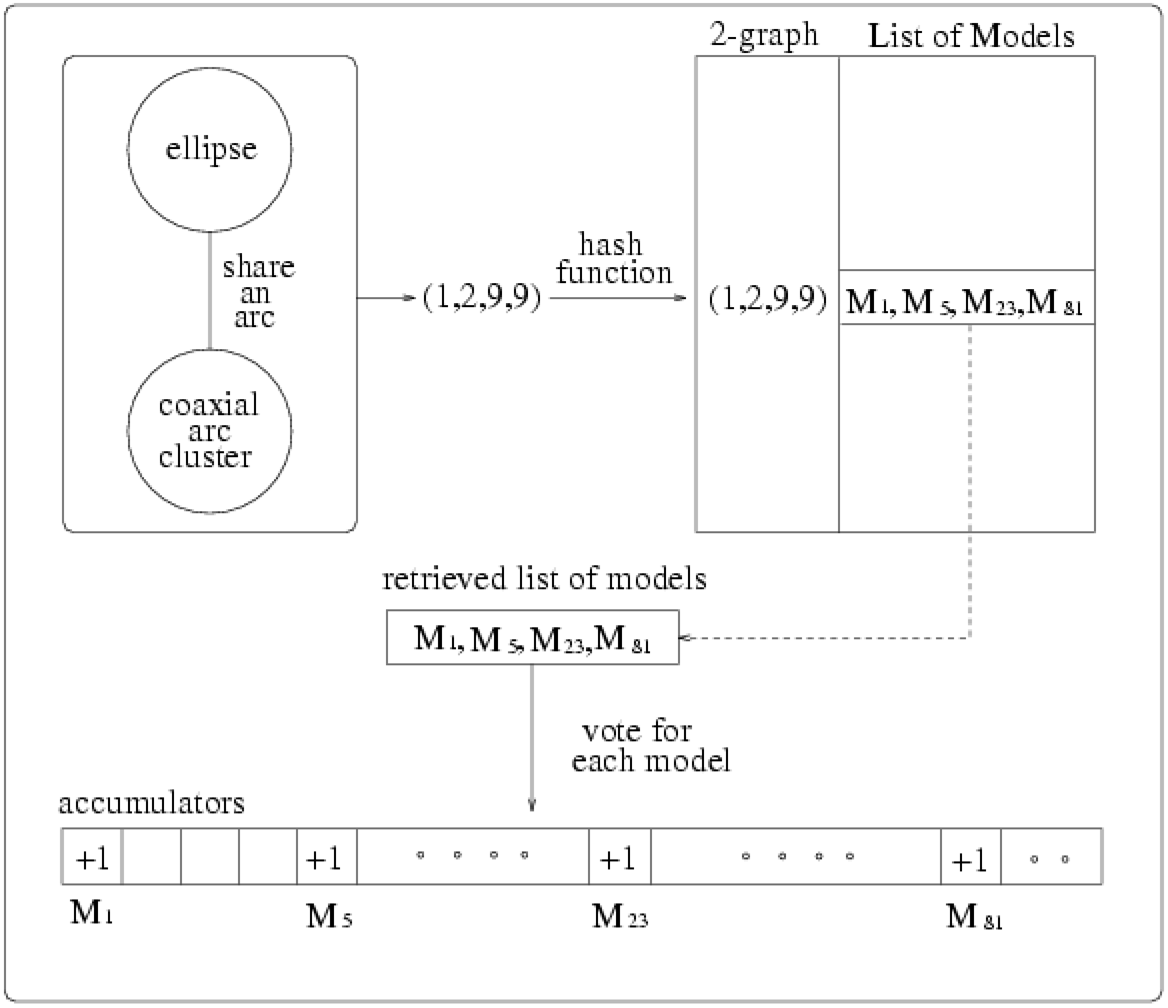
   - encode it, producing an index to access the hash table H

   - cast a vote for each Mi in the associated bin

3. Select the Mi's with high votes as possible hypotheses

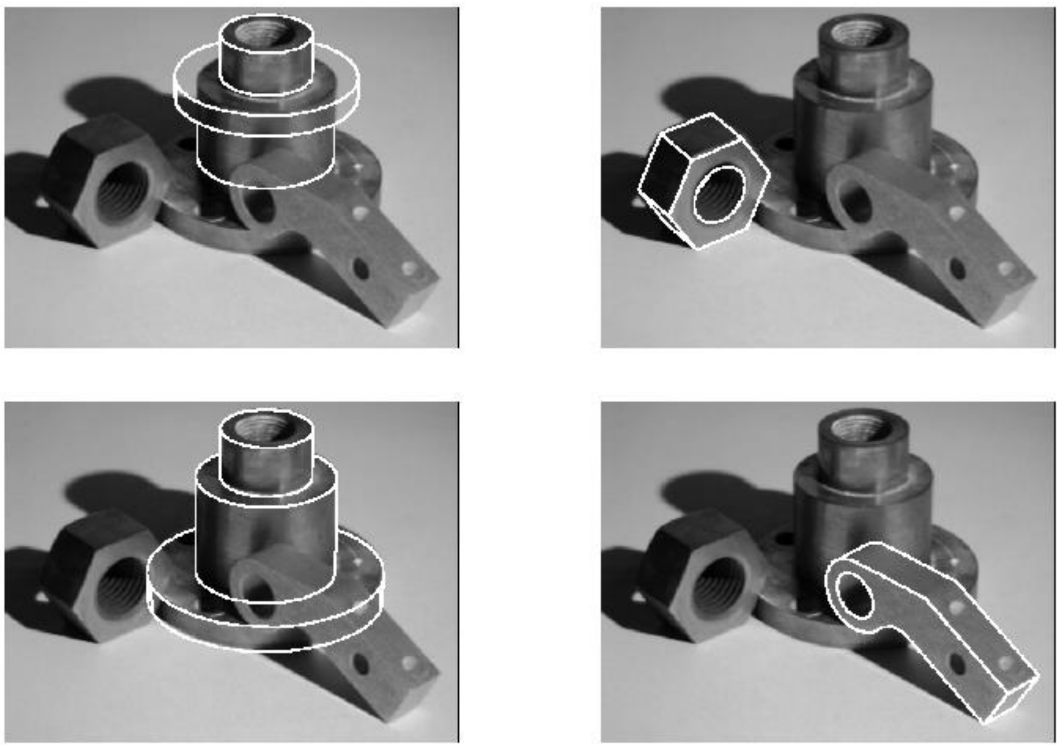
4. Verify or disprove via alignment, using the 3D meshes

The Voting Process



RIO Verifications

incorrect
hypothesis



1. The matched features of the hypothesized object are used to determine its **pose**.

2. The **3D mesh** of the object is used to project all its features onto the image.

3. A **verification procedure** checks how well the object features line up with edges on the image.

(Edge operator finds edges. Hausdorf distance compares image edges with object edges)

Use of classifiers is big in computer vision today.

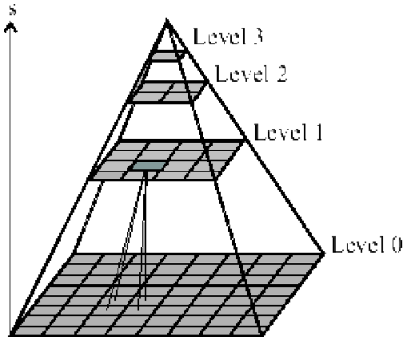
- 2 Examples:
  - Rowley’s Face Detection using neural nets
  - Yi’s image classification using EM

Object Detection: Rowley's Face Finder

1. convert to gray scale
2. normalize for lighting
3. histogram equalization
4. apply neural net(s) trained on 16K images

What data is fed to the classifier?

32 x 32 windows in a pyramid structure



Object Class Recognition using Images of Abstract Regions

Yi Li, Jeff A. Bilmes, and Linda G. Shapiro
Department of Computer Science and Engineering
Department of Electrical Engineering
University of Washington

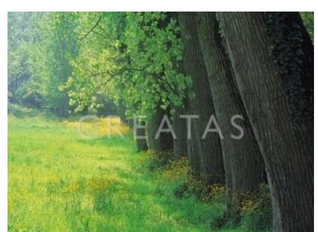
Problem Statement

Given: Some images and their corresponding descriptions

•••

{trees, grass, cherry trees} {cheetah, trunk} {mountains, sky} {beach, sky, trees, water}

To solve: What object classes are present in new images

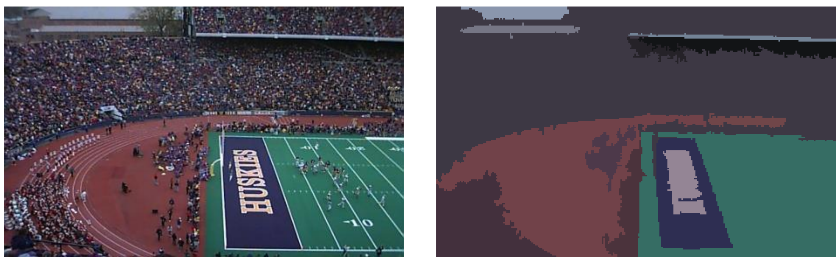


•••

? ? ? ?

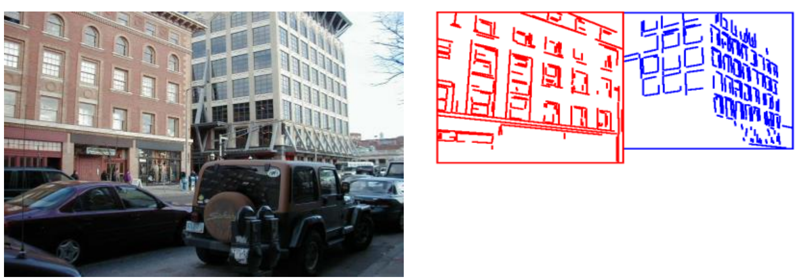
Image Features for Object Recognition

- Color

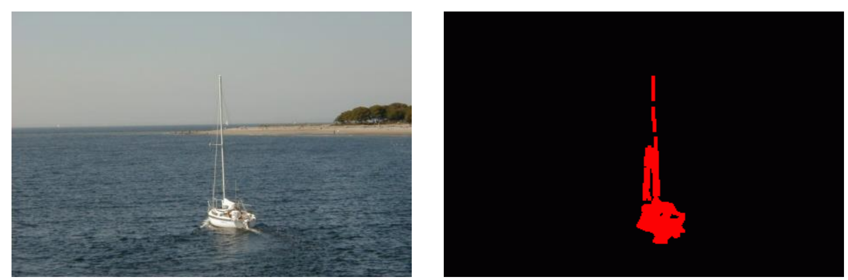


- Texture

- Structure

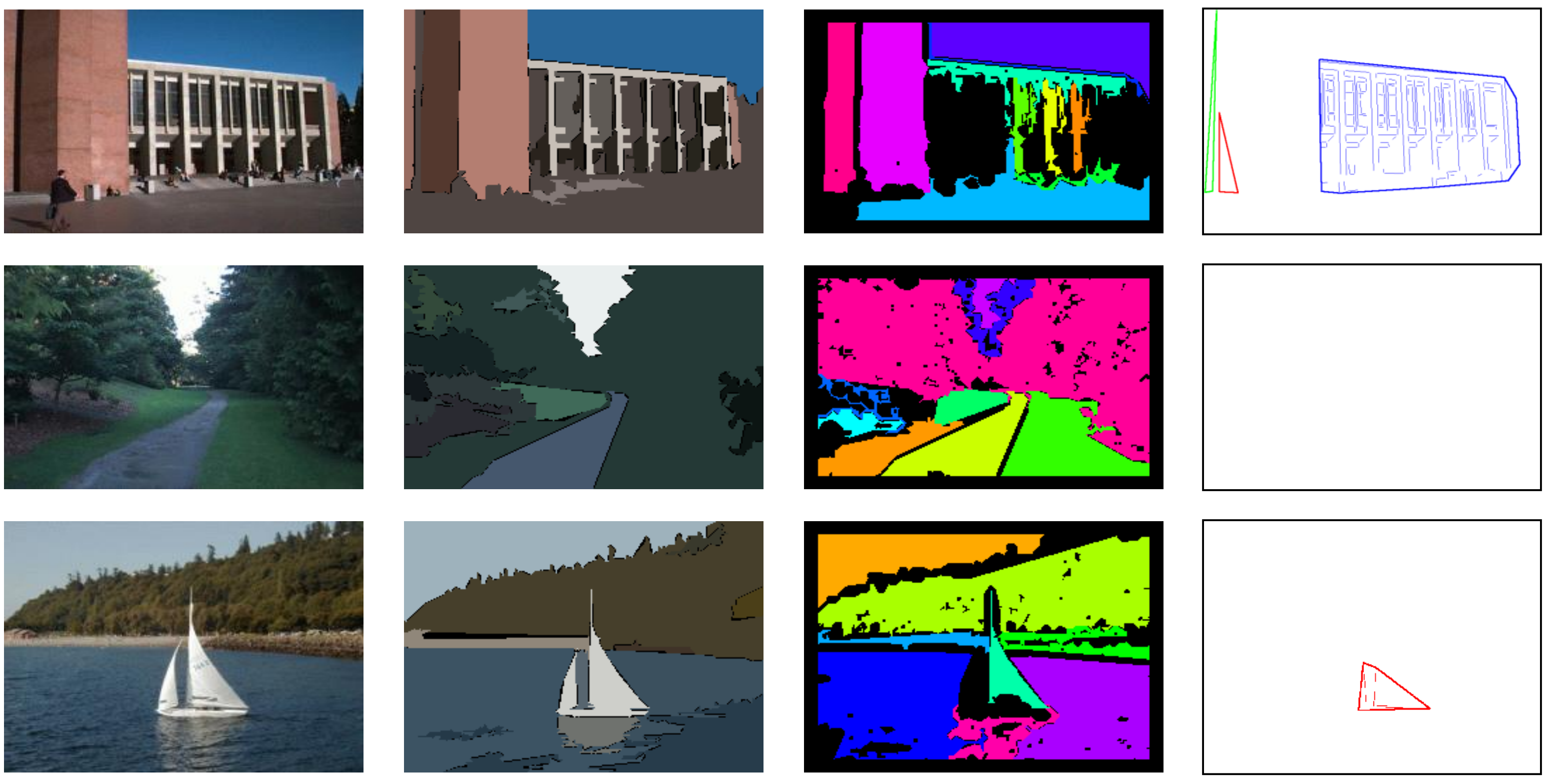


- Context



Abstract Regions

Original Images | Color Regions | Texture Regions | Line Clusters



Abstract Regions

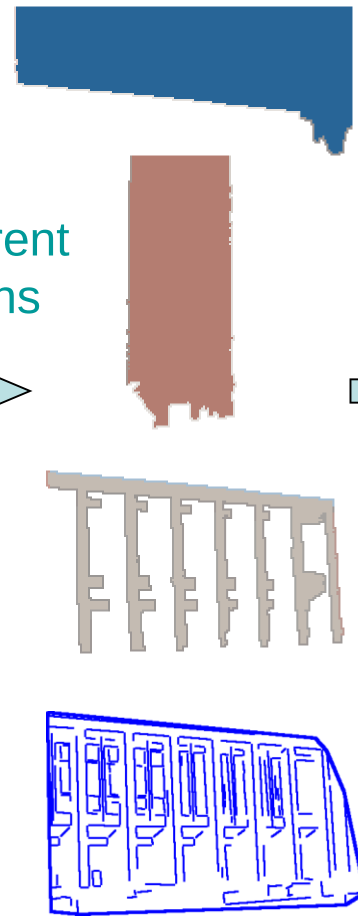
Multiple segmentations whose regions are not labeled; a list of labels is provided for each training image.

image

labels

{sky, building}

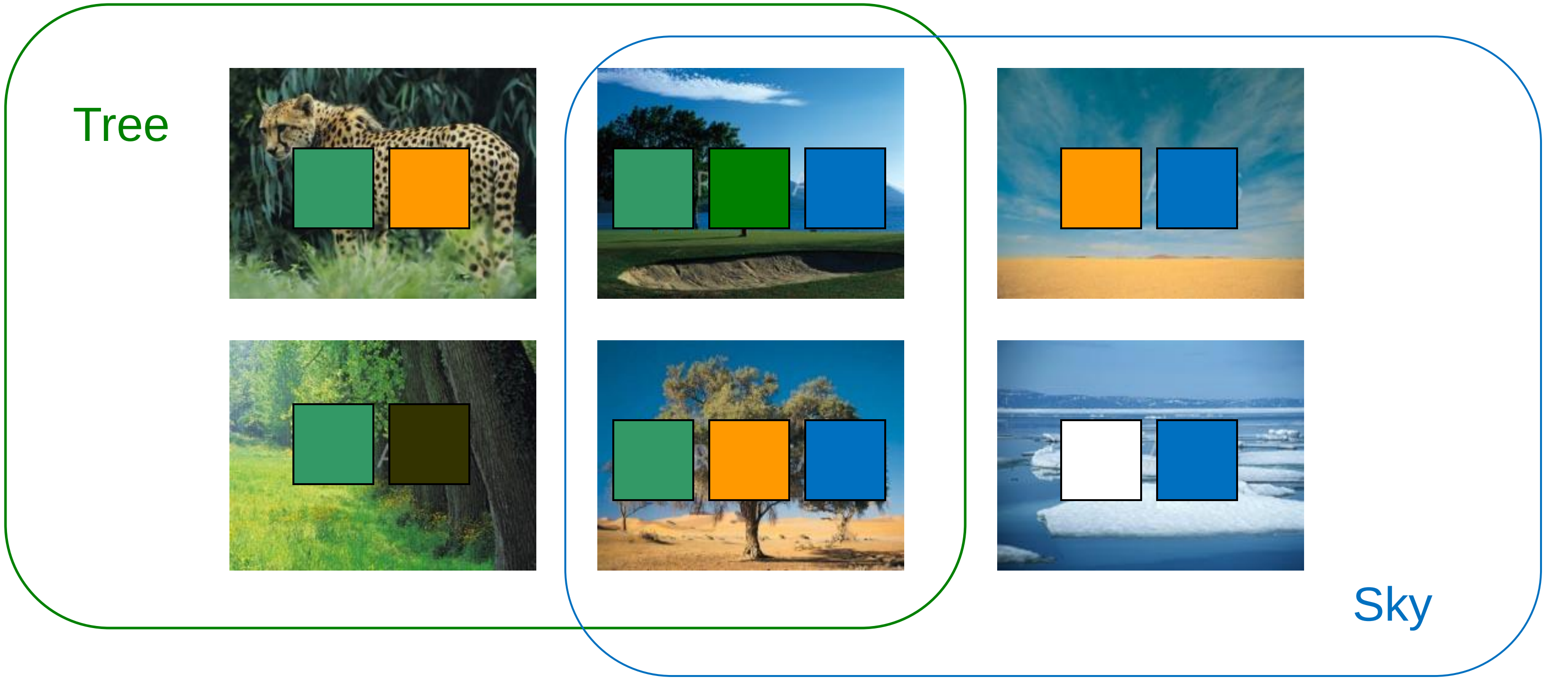
various different segmentations



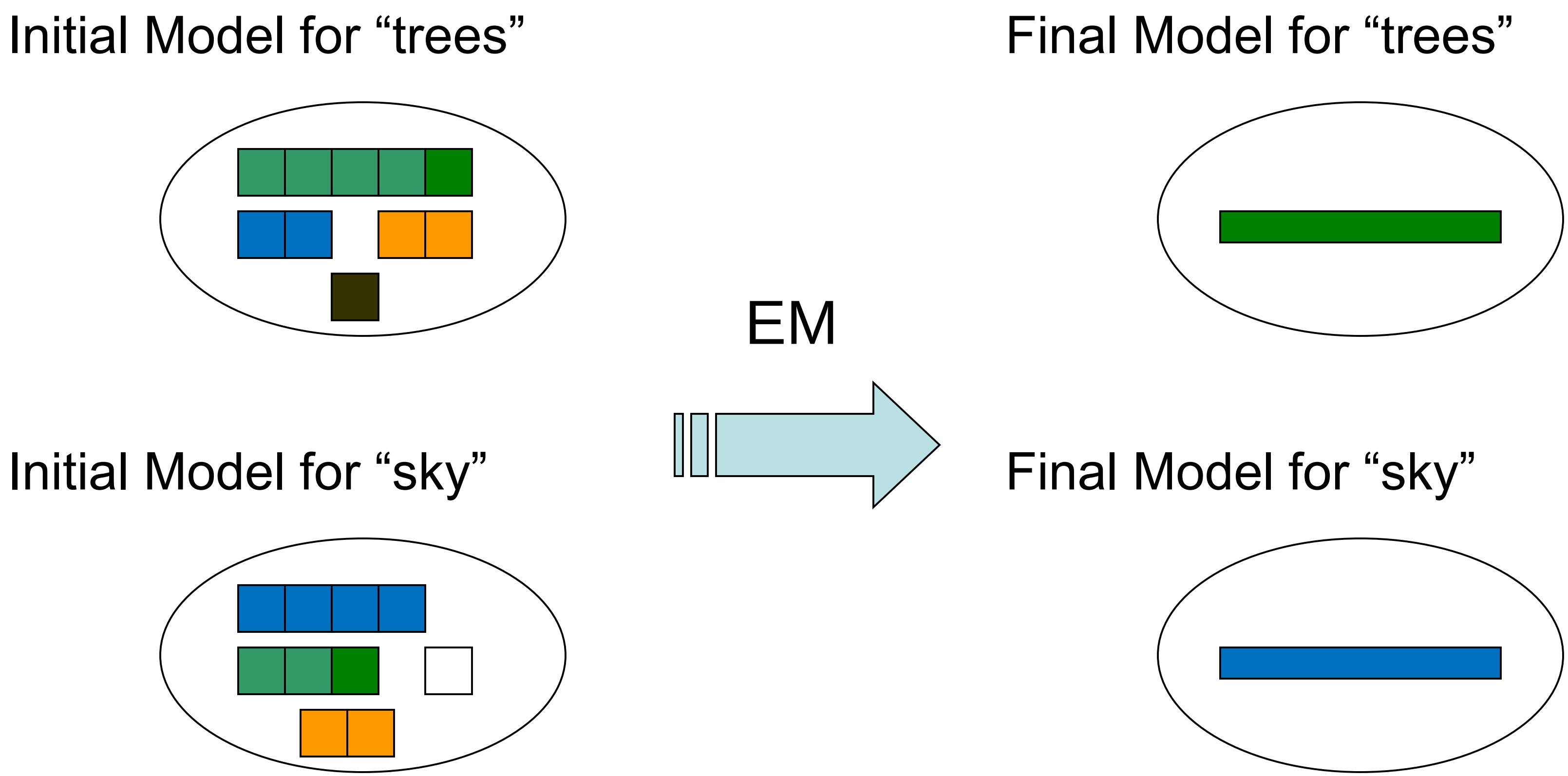
region attributes from several different types of regions

Model Initial Estimation

- Estimate the initial model of an object using all the region features from all images that contain the object



EM Classifier: the Idea

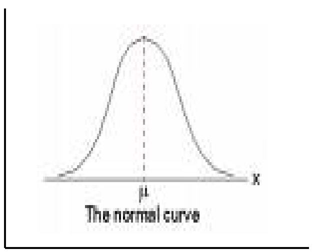


EM Algorithm

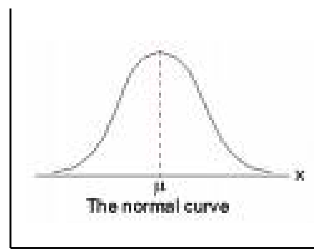
- Start with K clusters, each represented by a probability distribution
- Assuming a Gaussian or Normal distribution, each cluster is represented by its mean and variance (or covariance matrix) and has a weight.
- Go through the training data and soft-assign it to each cluster. Do this by computing the probability that each training vector belongs to each cluster.
- Using the results of the soft assignment, recompute the parameters of each cluster.
- Perform the last 2 steps iteratively.

1-D EM with Gaussian Distributions

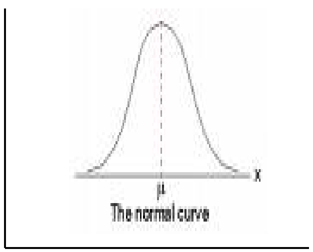
- Each cluster C_j is represented by a Gaussian distribution $N(\mu_j , \sigma_j)$.
- Initialization: For each cluster C_j initialize its mean μ_j , variance σ_j, and weight α_j.



$N(\mu_1 , \sigma_1)$
$\alpha_1 = P(C_1)$



$N(\mu_2 , \sigma_2)$
$\alpha_2 = P(C_2)$



$N(\mu_3 , \sigma_3)$
$\alpha_3 = P(C_3)$

- With no other knowledge, use random means and variances and equal weights.

Standard EM to EM Classifier

- That's the standard EM algorithm.
- For n-dimensional data, the variance becomes a co-variance matrix, which changes the formulas slightly.
- But we used an EM variant to produce a classifier.
- The next slide indicates the differences between what we used and the standard.

EM Classifier

1. Fixed Gaussian components (one Gaussian per object class) and fixed weights corresponding to the frequencies of the corresponding objects in the training data.

2. Customized initialization uses only the training images that contain a particular object class to initialize its Gaussian.

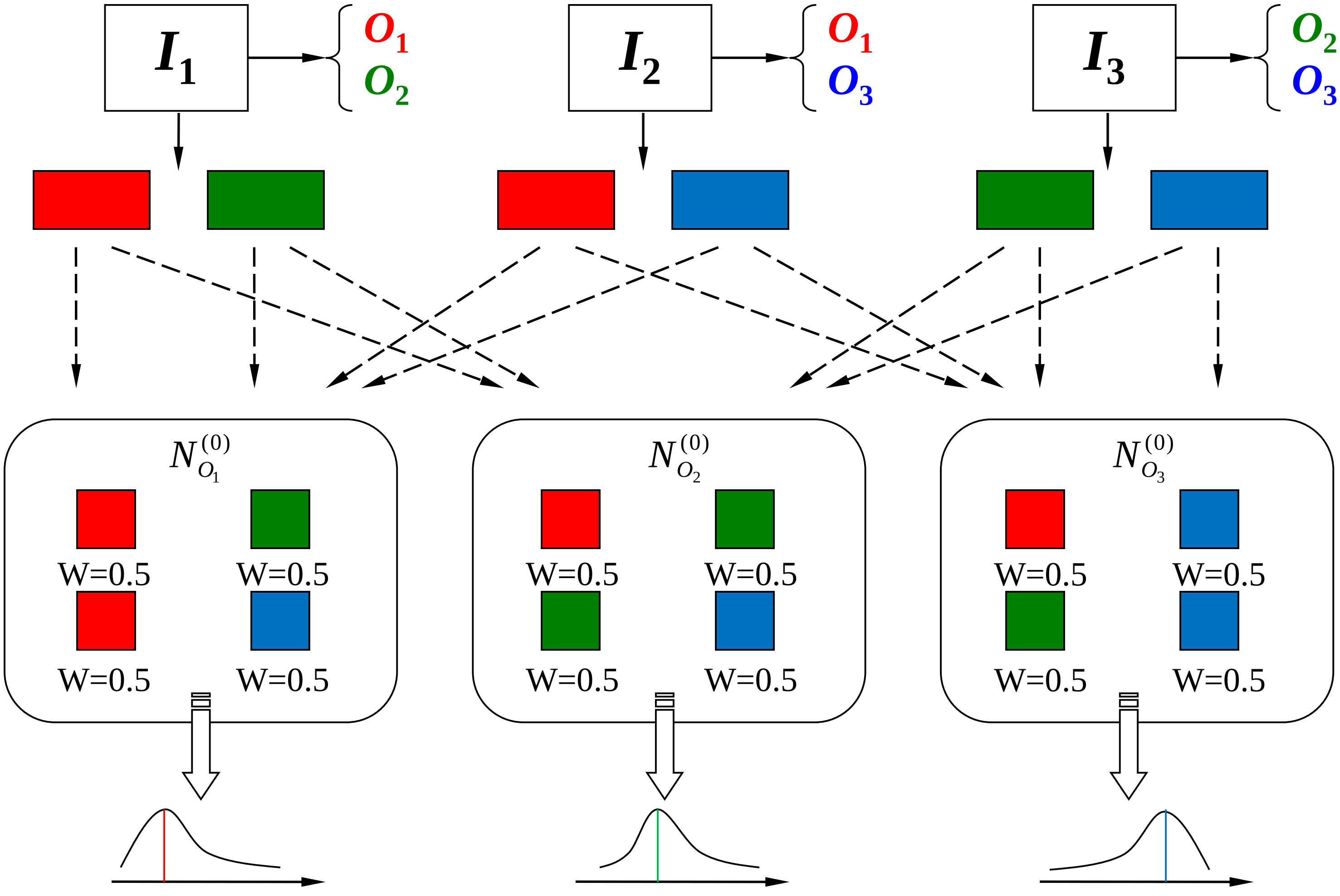
3. Controlled expectation step ensures that a feature vector only contributes to the Gaussian components representing objects present in its training image.

4. Extra background component absorbs noise.

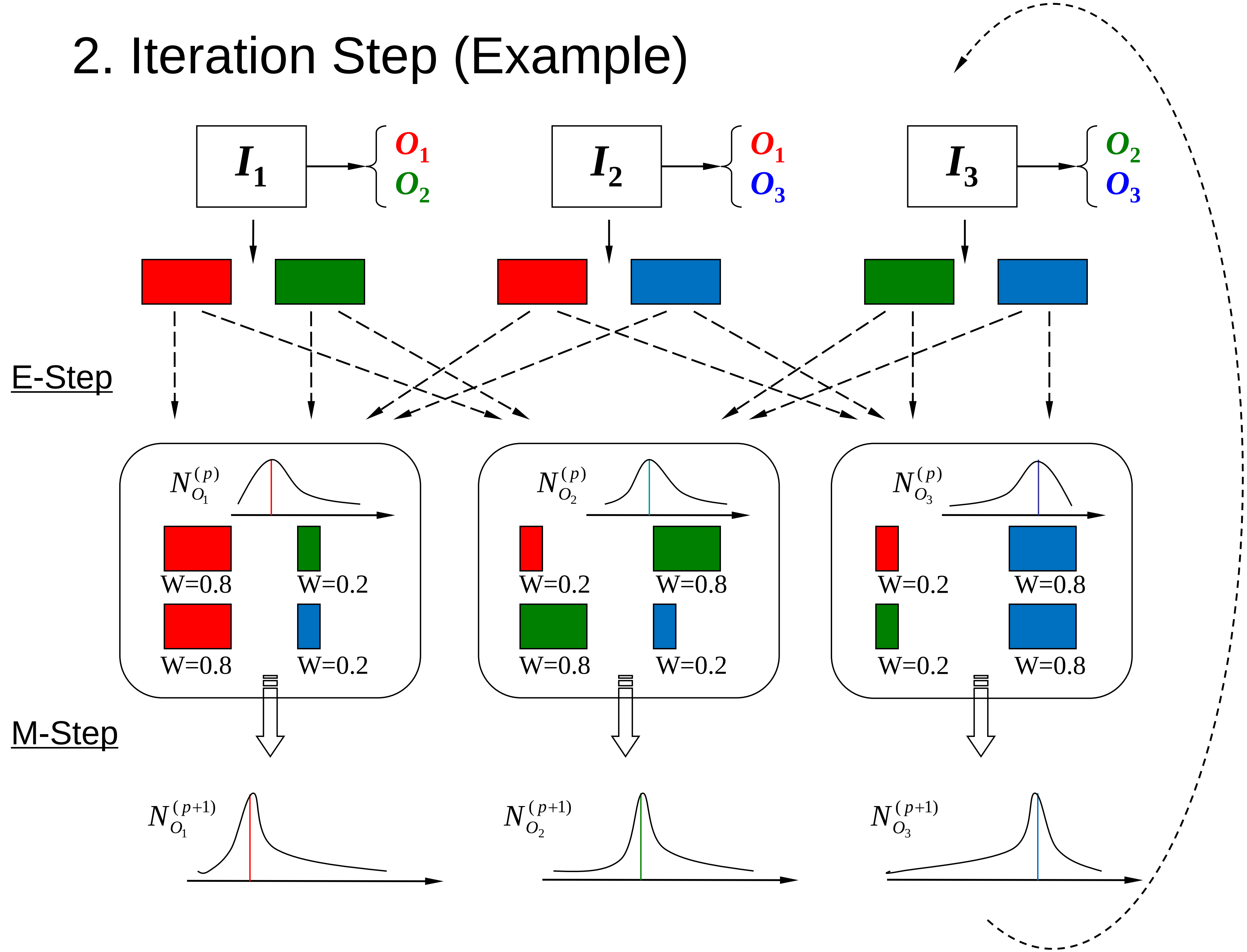
Gaussian for trees	Gaussian for buildings	Gaussian for sky	Gaussian for background

1. Initialization Step (Example)

Image & description



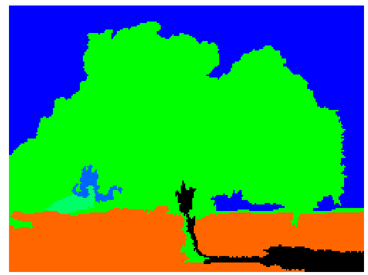
2. Iteration Step (Example)



Recognition

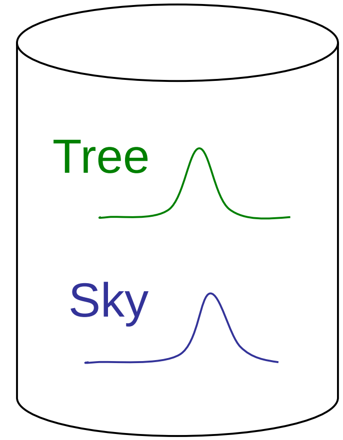
Test Image

Color Regions



compare

Object Model Database



How do you decide if a particular object is in an image?

To calculate $p(tree \mid image)$

$p(tree \mid image) = f$ (

$p(\ tree \mid$  $)$

$p(\ tree \mid$  $)$

$p(\ tree \mid$  $)$

$p(\ tree \mid$  $)$

)

$$p(o \mid F_I^a) = \underset{r^a \in F_I^a}{f} \left(p(o \mid r^a) \right)$$

f is a function that combines probabilities from all the color regions in the image.

e.g. max or mean

Combining different types of abstract regions: First Try

- Treat the different types of regions independently and combine at the time of classification.

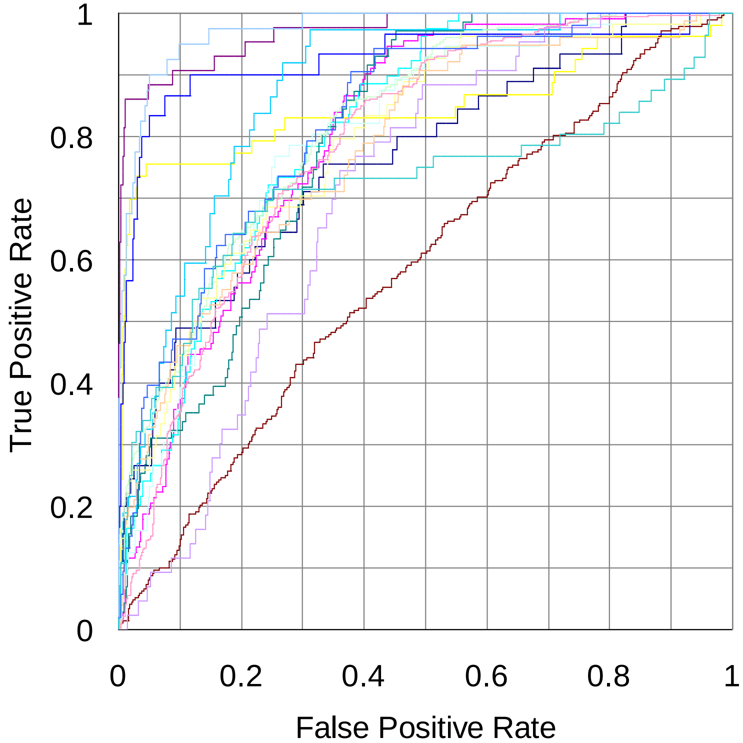
$$p(o \mid \{F_I^a\}) = \prod_a p(o \mid F_I^a)$$

- Form intersections of the different types of regions, creating smaller regions that have both color and texture properties for classification.

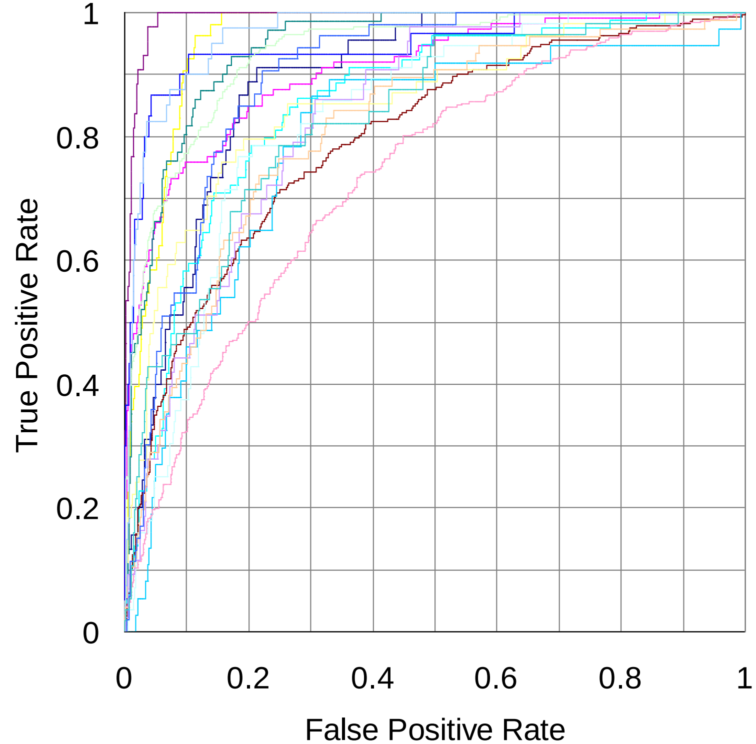
Experiments (on 860 images)

- 18 keywords: mountains (30), orangutan (37), track (40), tree trunk (43), football field (43), beach (45), prairie grass (53), cherry tree (53), snow (54), zebra (56), polar bear (56), lion (71), water (76), chimpanzee (79), cheetah (112), sky (259), grass (272), tree (361).
- A set of cross-validation experiments (80% as training set and the other 20% as test set)
- The poorest results are on object classes "tree," "grass," and "water," each of which has a high variance; a single Gaussian model is insufficient.

ROC Charts:
True Positive vs. False Positive



Independent Treatment of
Color and Texture



Using Intersections of
Color and Texture Regions