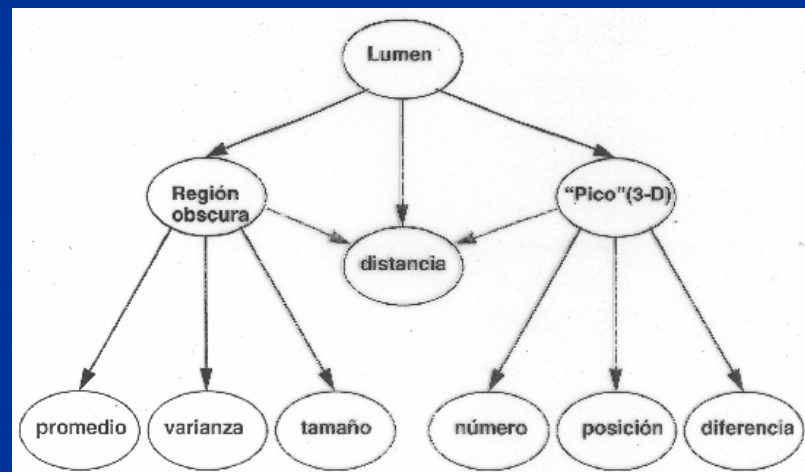


# Visión de Alto Nivel

Dr. Luis Enrique Sucar

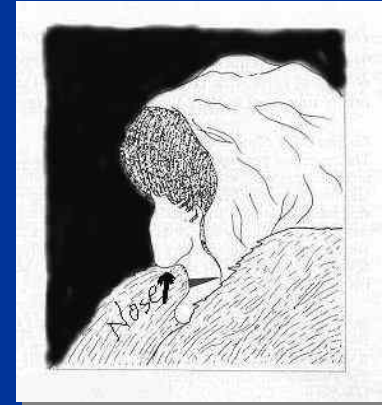
INAOE

[esucar@inaoep.mx](mailto:esucar@inaoep.mx)  
[ccc.inaoep.mx/~esucar](http://ccc.inaoep.mx/~esucar)



Sesión 5  
Visión Bayesiana

# What do you see?



What we see depends on our previous knowledge (model) of the world and the information (data) from the images → Bayesian perception

# Bayesian visual perception

- The perception problem is characterized by two main aspects:
  - The properties of the world that is observed (prior knowledge)
  - The image data used by the observer (data)
- The Bayesian approach combines these two aspects which are characterized as probability distributions

# Representation

- Scene properties –  $S$
- Model of the world – prior probability distribution –  $P(S)$
- Model of the image – probability distribution of the image given de scene (likelihood) –  $P(I/S)$

# Recognition

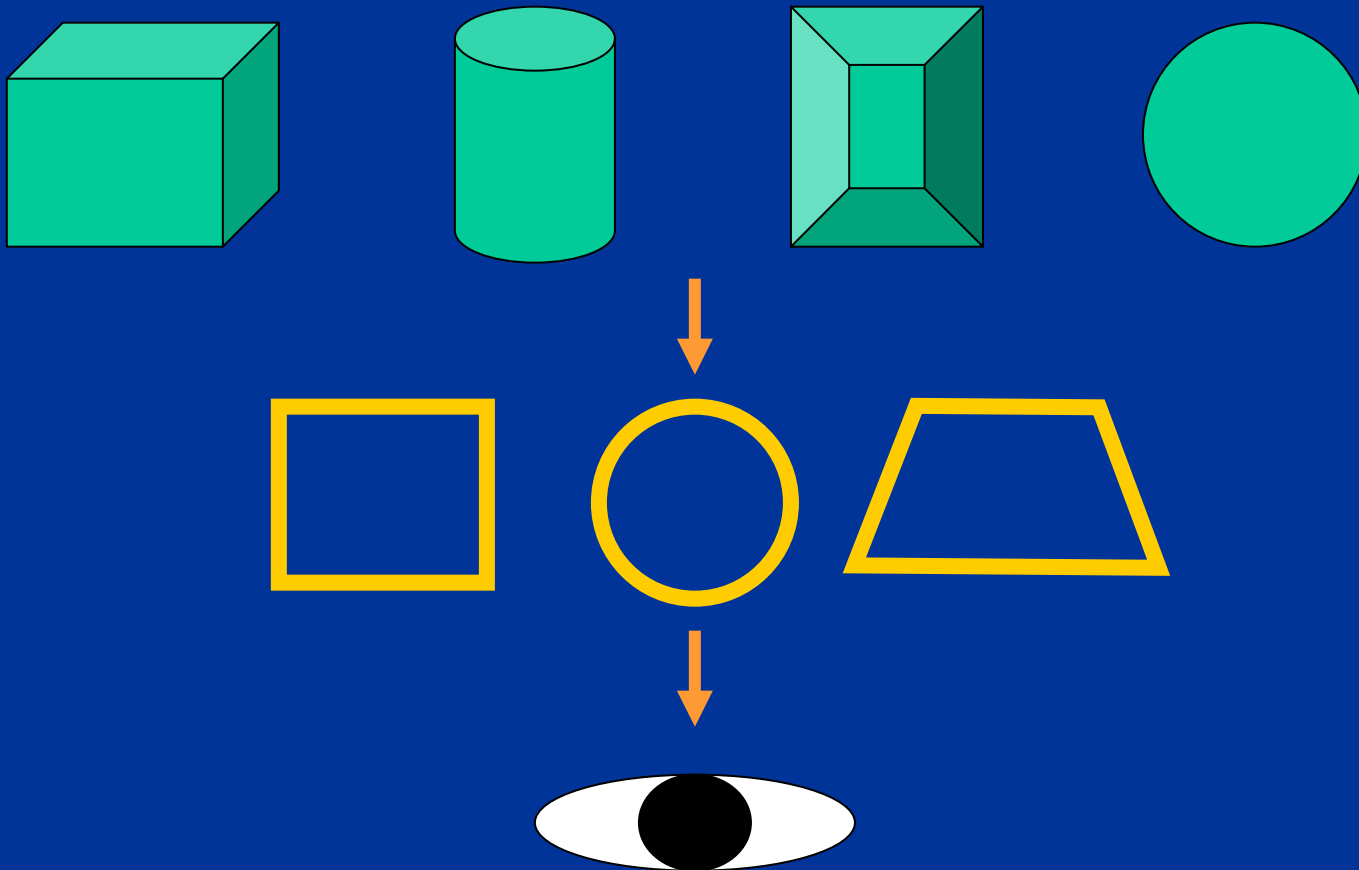
- The scene (object) is characterized by the posterior probability distribution –  $P(S/I)$
- By Bayes theorem:

$$P(S/I) = P(S) P(I/S) / P(I)$$

- The denominator can be consider as a normalizing constant:

$$P(S/I) = k P(S) P(I/S)$$

# Example



# Example

- Prior distribution of objects –  $P(O)$ 
  - Cube            0.2
  - Cylinder        0.3
  - Sphere          0.1
  - Prism            0.4

# Example

- Likelihood function  $P(\text{Silhouette}|\text{Object}) - P(S|O)$

	Prism	Cube	Cylinder	Sphere
Square	1.0	0.6	0.0	0.4
Circle	0.0	0.4	1.0	0.0
Trapezoid	0.0	0.0	0.0	0.6



# Example

- Posterior distribution  $P(\text{Object}|\text{Silhouette}) - P(O|S)$

- Bayes rule:

$$P(O|S) = k P(O) P(S|O)$$

- For example, given  $S=\text{square}$

$$P(\text{Cube} | \text{square}) = k 0.2 * 1 = k 0.2 = \mathbf{0.37}$$

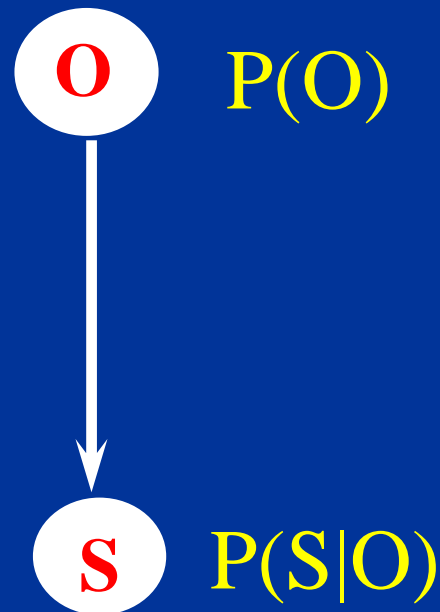
$$P(\text{Cylinder} | \text{square}) = k 0.3 * 0.6 = k 0.18 = \mathbf{0.33}$$

$$P(\text{Sphere} | \text{square}) = k 0.1 * 0 = \mathbf{0}$$

$$P(\text{Prism} | \text{square}) = k 0.4 * 0.4 = k 0.16 = \mathbf{0.30}$$

# Graphical Model

- We can represent the dependence relation in this simple example graphically, with 2 variables and an arc



# Graphical Models

- This graphical representation of probabilistic models can be extended to more complex ones.
- There are several types of probabilistic graphical models (PGMs) that can be applied to different problems in perception

# Probabilistic Graphical Models

- A graphical model is specified by two aspects:
  - A Graph,  $G(V,E)$ , that defines the structure of the model
  - A set of local functions,  $f(Y_i)$ , that defines the parameters (probabilities), where  $Y_i$  is a subset of  $X$
- The joint probability is defined by the product of the local functions:

$$P(X_1, X_2, \dots, X_N) = \prod_{i=1}^n f(Y_i)$$

# Probabilistic Graphical Models

- This representation in terms of a graph and a set of local functions (called potentials) is the basis for *inference* and *learning* in PGMs
  - **Inference**: obtain the marginal or conditional probabilities of any subset of variables  $Z$  given any other subset  $Y$
  - **Learning**: given a set of data values for  $X$  (that can be incomplete) estimate the structure (graph) and parameters (local function) of the model

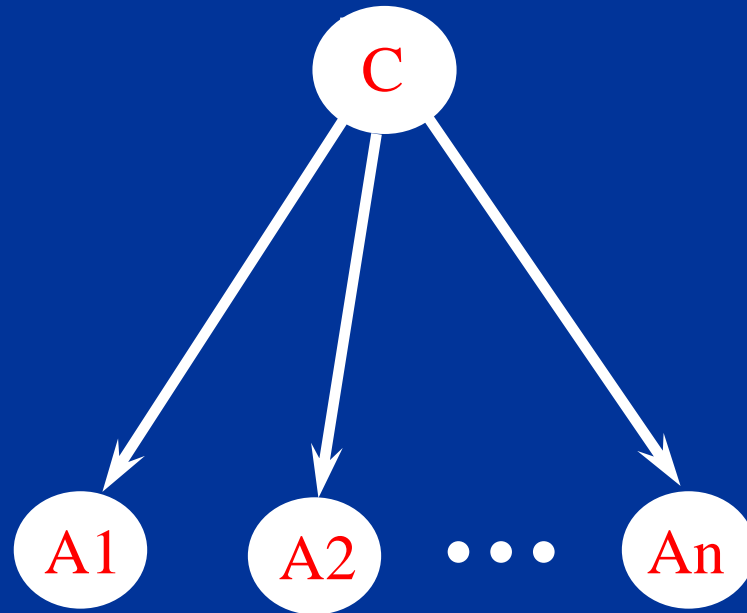
# Types of PGMs

- We will consider the following models and their applications in vision:
  - Bayesian classifiers
  - Bayesian networks
  - Hidden Markov models
  - Dynamic Bayesian networks
  - Markov Random Fields

# Bayesian Classifier

- A Bayesian classifier is used to obtain the probability of certain variable (the class or hypothesis,  $H$ ) given a set of variables known as the attributes or evidence ( $E = E_1, \dots, E_N$ )
- It is usually assumed that the attributes are independent given the class – **Naive Bayesian Classifier** – so its PGM is represented as a “star” with the class as the root and the attributes as the leafs

# Naive Bayesian Classifier





# Bayesian Classifier

- The posterior probability of each hypothesis (**H**) based on the Evidence (**E**) is:

$$P(H | E) = P(H) P(E | H) / P(E)$$

# Naive Bayesian classifier Inference

- Consider each attribute independent given the hypothesis:

$$P(E_1, E_2, \dots, E_N | H) = P(E_1 | H) P(E_2 | H) \dots P(E_N | H)$$

- So the posterior probability is given by:

$$\begin{aligned} P(H | E_1, E_2, \dots, E_N) &= \\ &= [P(H) P(E_1 | H) P(E_2 | H) \dots P(E_N | H)] / P(E) \\ &= k P(H) P(E_1 | H) P(E_2 | H) \dots P(E_N | H) \end{aligned}$$

# Naive Bayesian classifier Learning

- Structure:
  - the structure is given by the naive Bayes assumption
- Parameters:
  - we need to estimate the prior probability of each class

$$P(C_i)$$

- and the individual conditional probabilities of each attribute given the class

$$P(A_k / C_i)$$

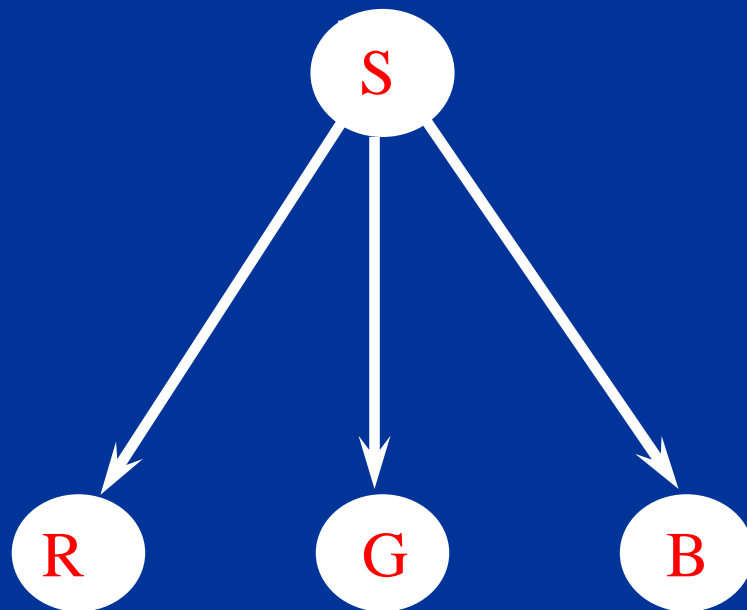
## Example

- Skin classification based on color
  - Hypothesis: skin, no-skin
  - Attributes: red, green, blue (256 values each)

- Probability function:

$$P(S|R,G,B) = k P(S) P(R|S) P(G|S) P(B|S)$$

# *Naive Bayes*



# Skin detection

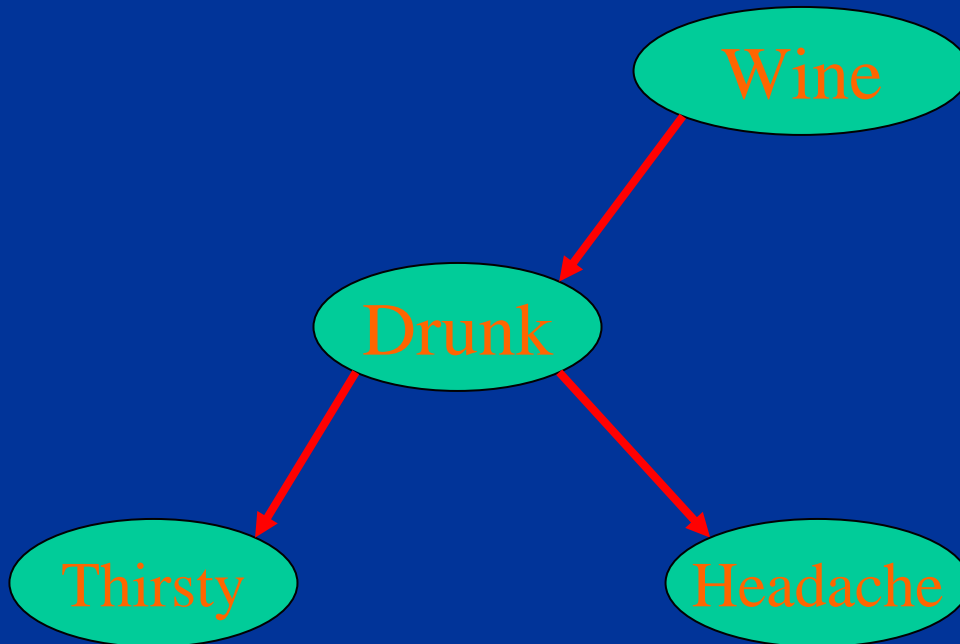
Detection of skin pixels based on color information and a Bayesian classifier



# Bayesian Networks

- Bayesian networks (BN) are a graphical representation of dependencies between a set of random variables. A Bayesian net is a Directed Acyclic Graph (DAG) in which:
  - **Node: Propositional variable.**
  - **Arcs: Probabilistic dependencies.**
- An arc between two variables represents a direct dependency, usually interpreted as a *causal* relation.

# An example of a BN

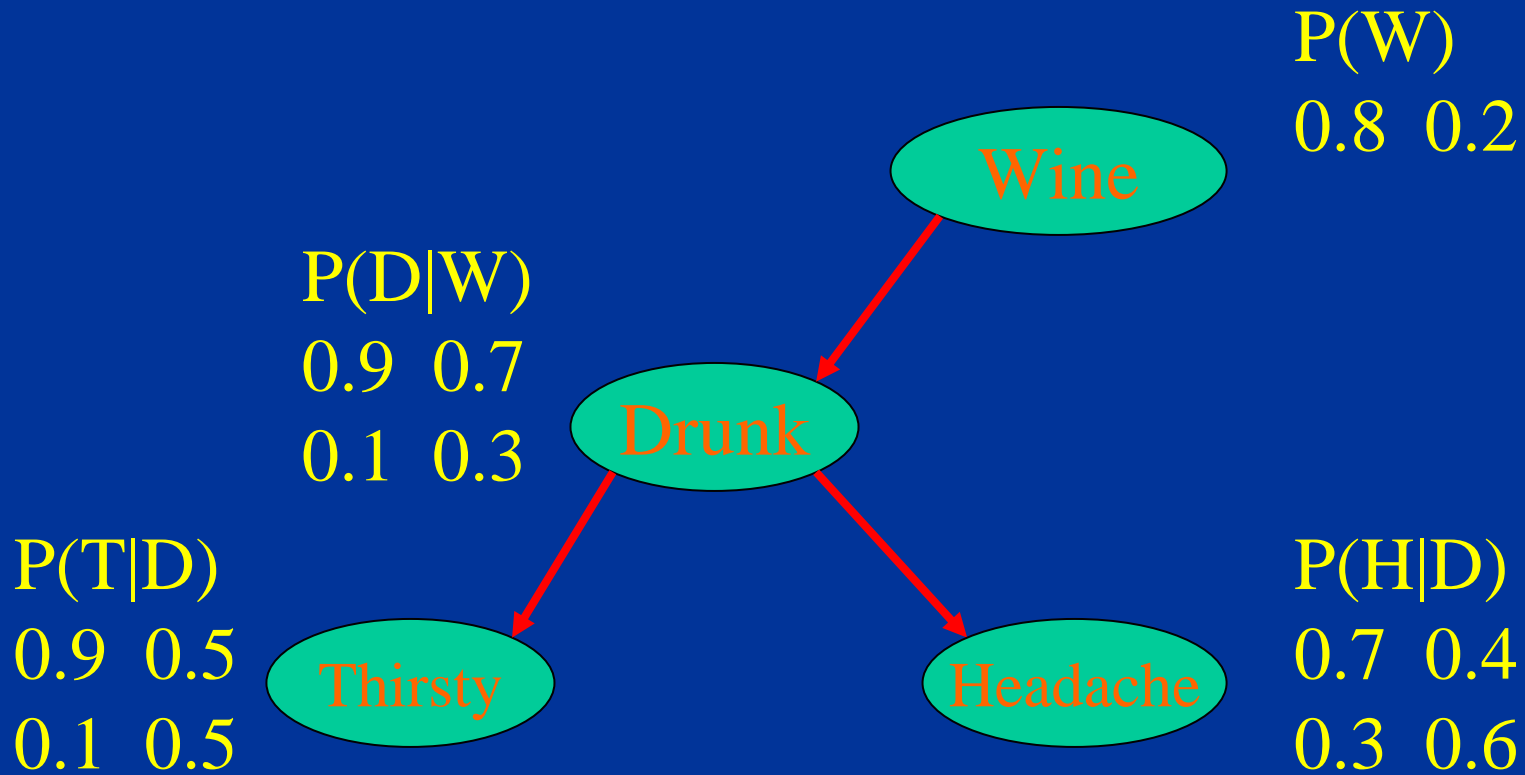




# Parameters

Conditional probabilities of each node given its parents.

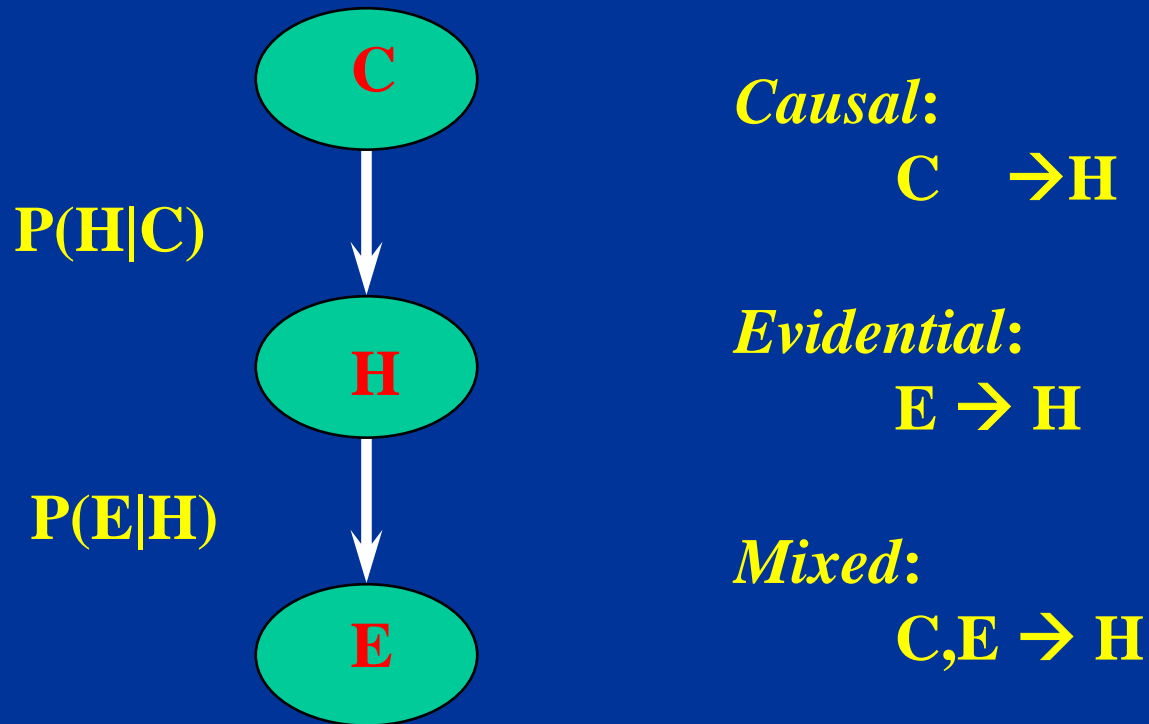
- **Root nodes:** vector of prior probabilities
- **Other nodes:** matrix of conditional probabilities



# Inference

- Inference in a Bayesian Network consists on estimating the posterior probability of some variables (unknowns) given the values of some other variables (evidence)
- There are several algorithms for probability propagation in BN
- All the methods are based on Bayes theorem

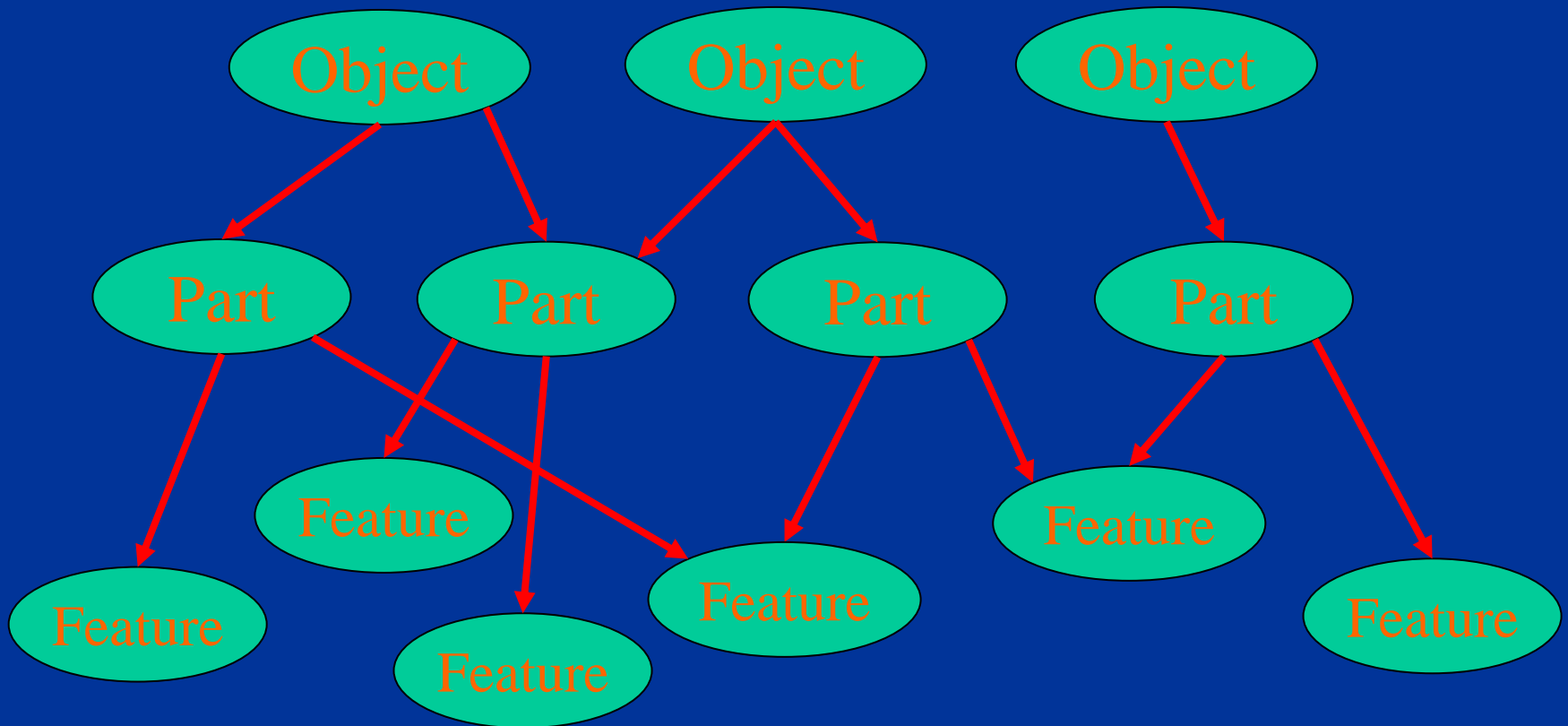
# Inference



# BN in Vision

General Model  
Example

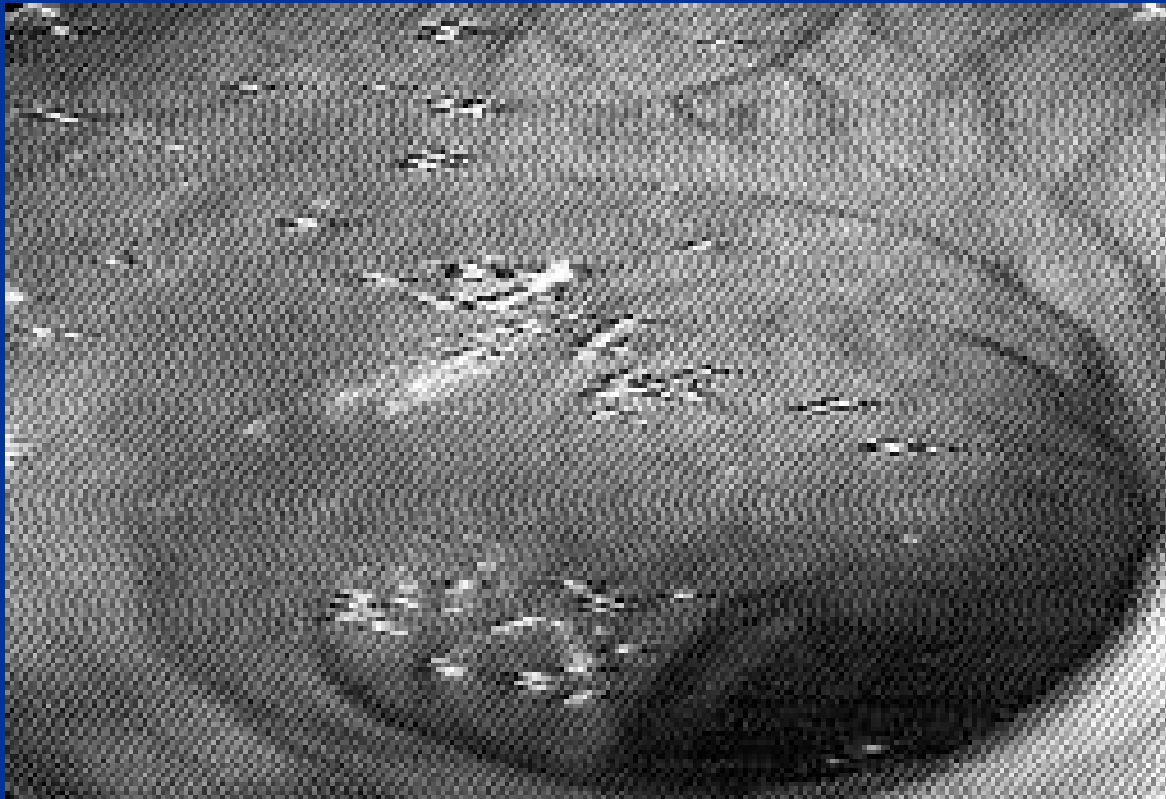
# A "general" BN model for Vision



## Example - endoscopy

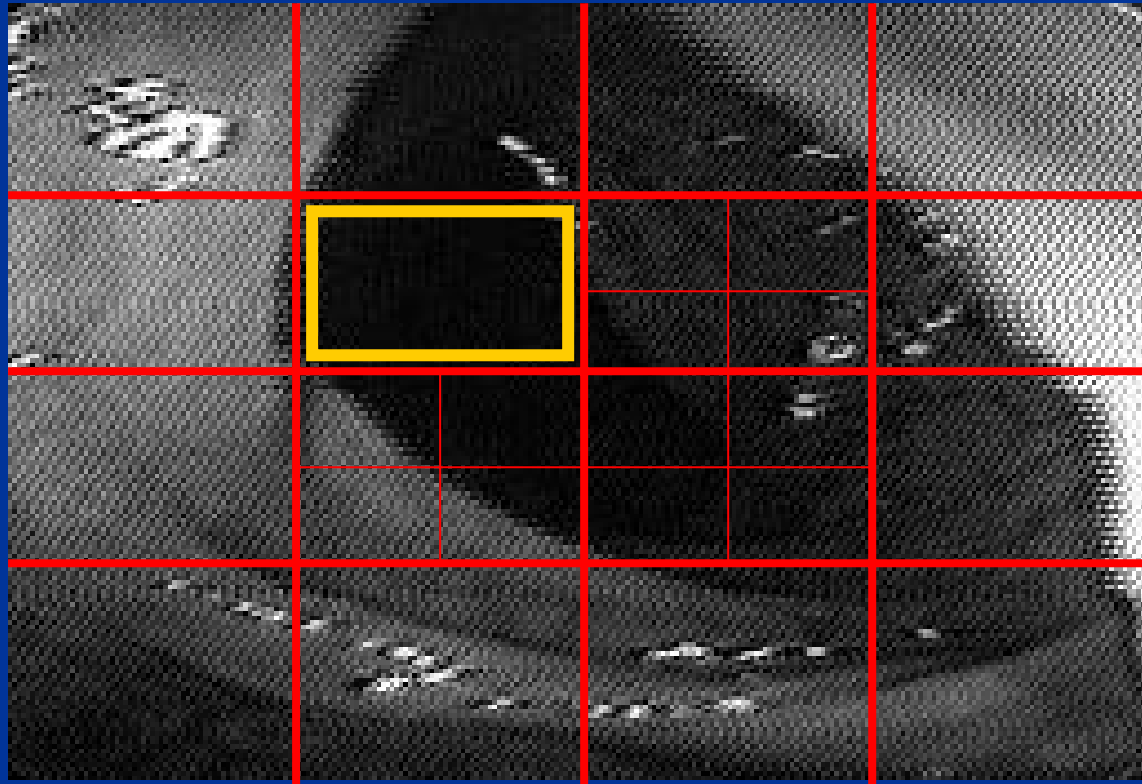
- Endoscopy is a tool for direct observation of the human digestive system
- Recognize “objects” in endoscopy images of the colon for semi-automatic navigation
- Main feature – dark regions
- Main objects – “*lumen*” & “*diverticula*”

# Colon Image

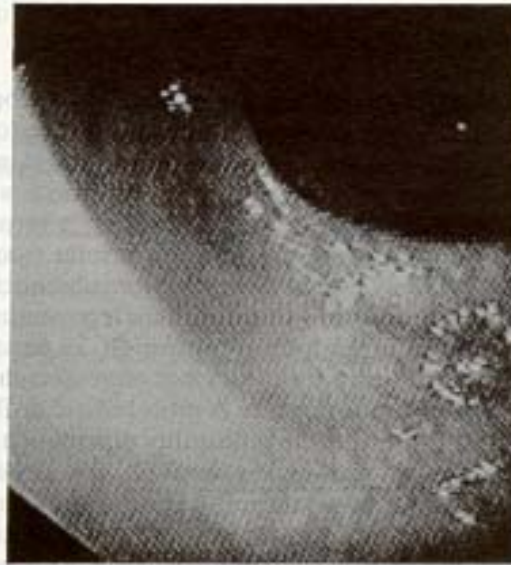




# Segmentation – dark region



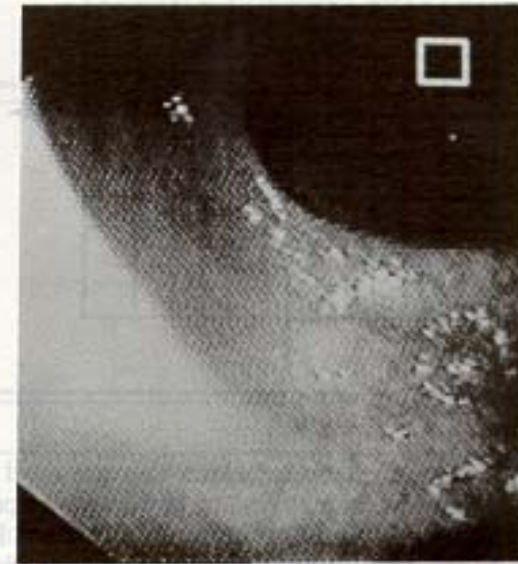
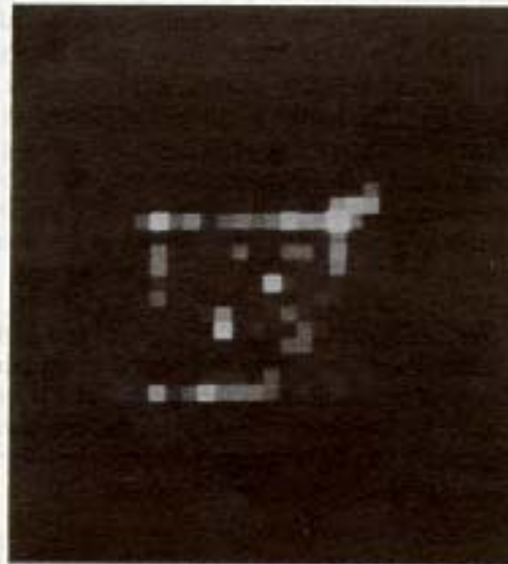
# Features – pq histogram



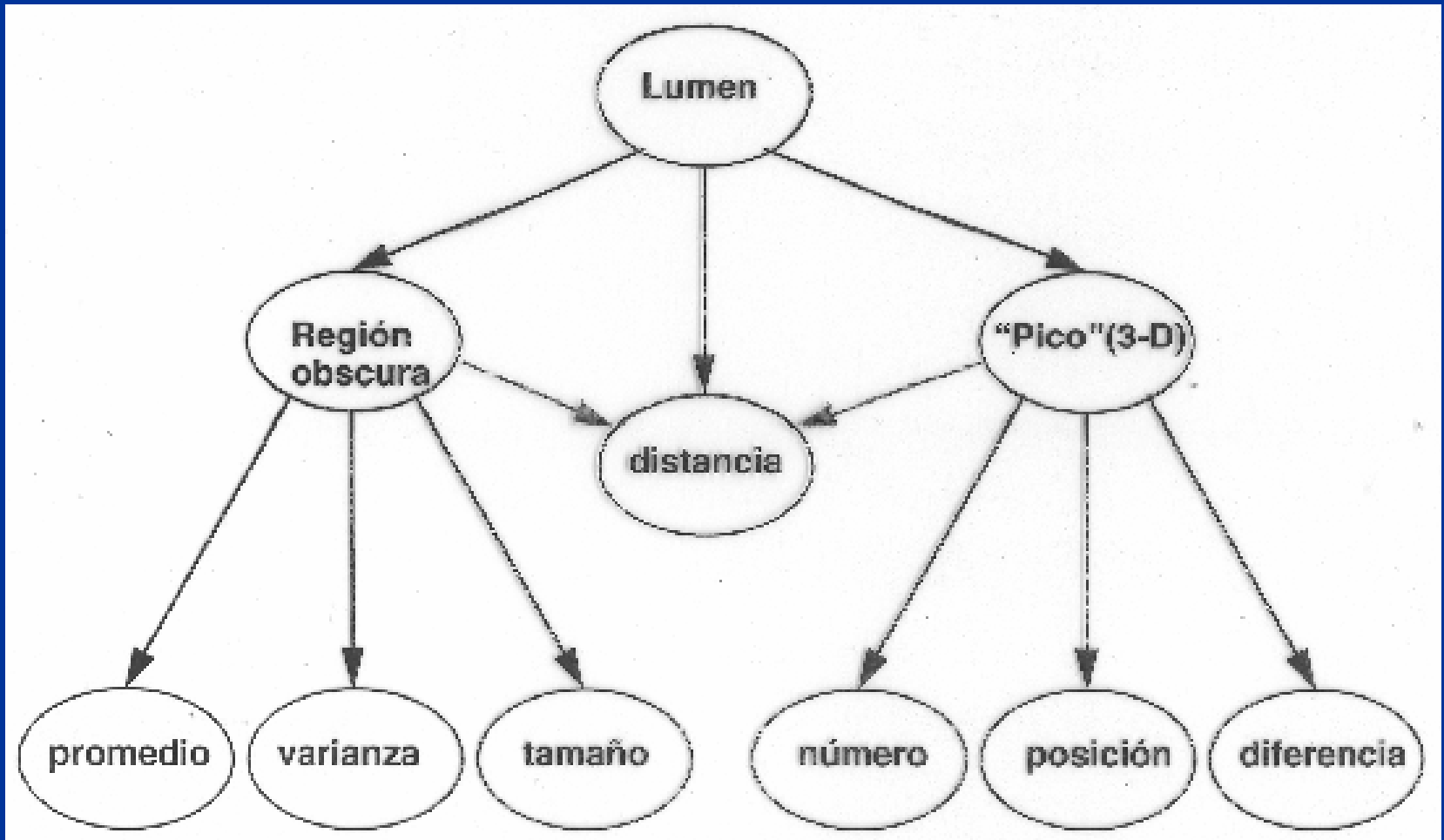
(a) Colon Image



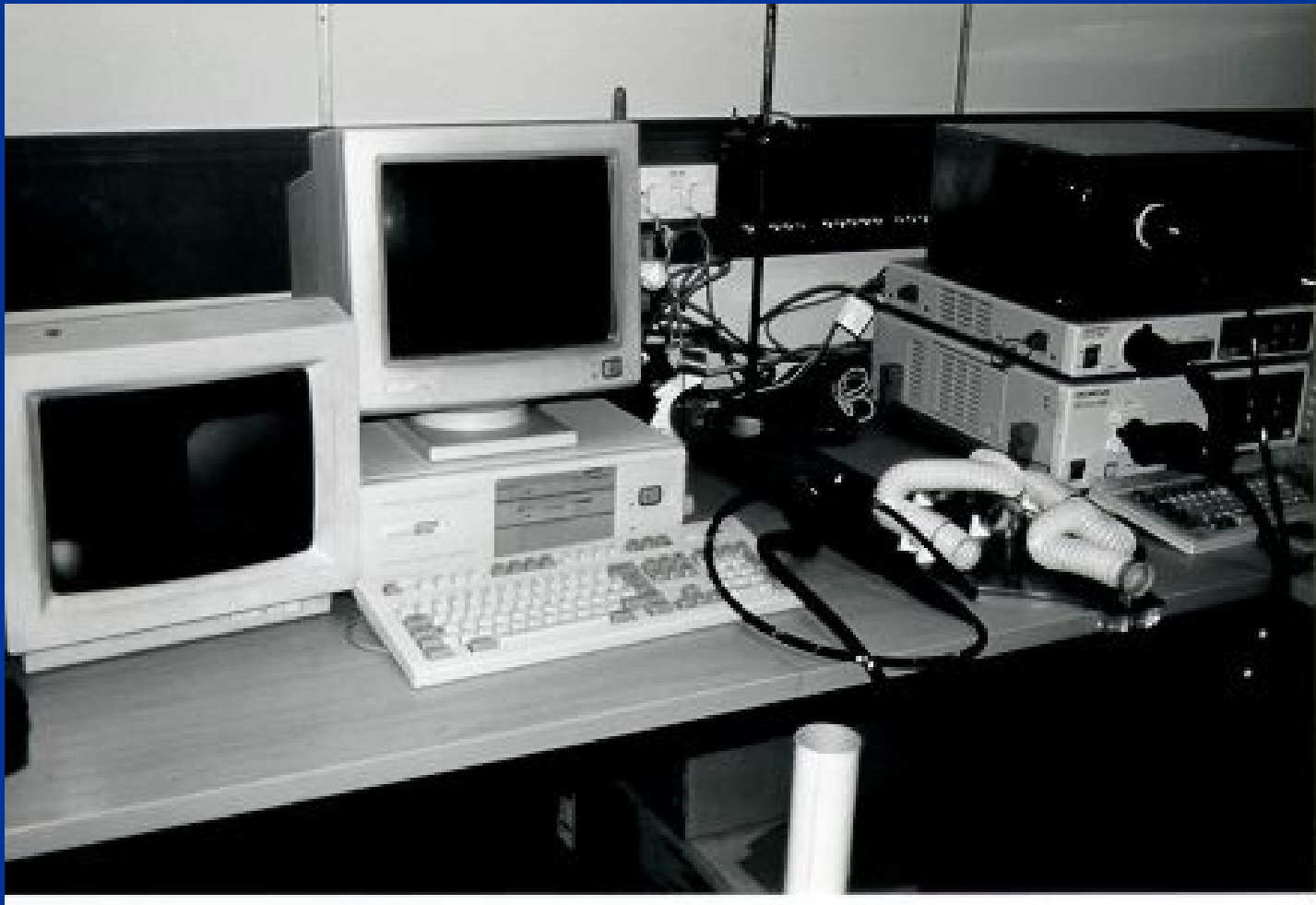
(b) Depth map (needle diagram)



# BN for endoscopy (partial)



# Semi-automatic Endoscope



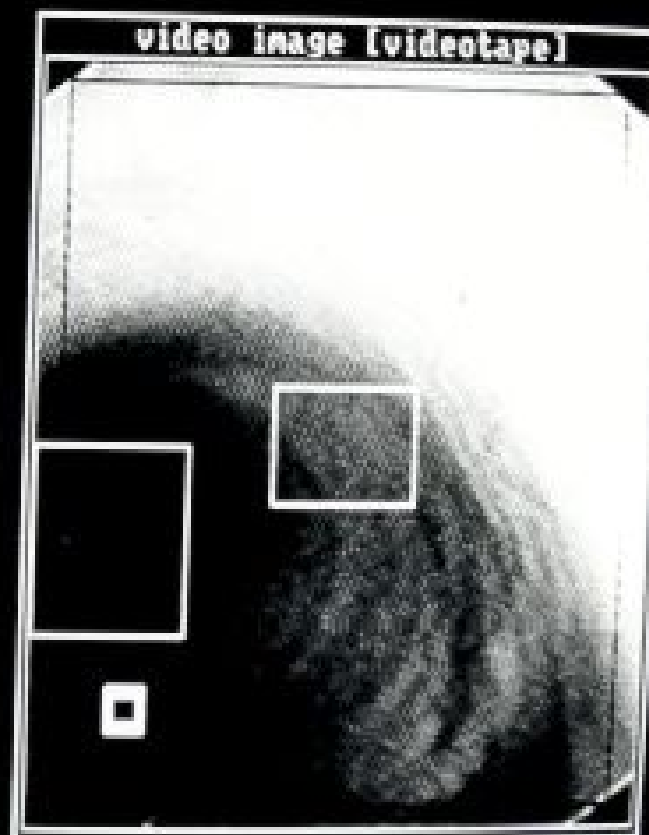
**status**  
auto OFF search OFF

**interpretation**  
LUMEN

**advice**  
Push Endoscope

**pg-hist**  


**commands**  
Auto ON/OFF  
sEarch ON/OFF  
Centre tip  
Tape/scope  
Learning  
Quit

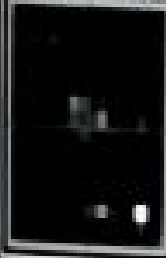


status  
auto OFF search OFF

interpretation  
No Lumen - lost view

advice  
Pull-back and then  
push endoscope again

pq-hist



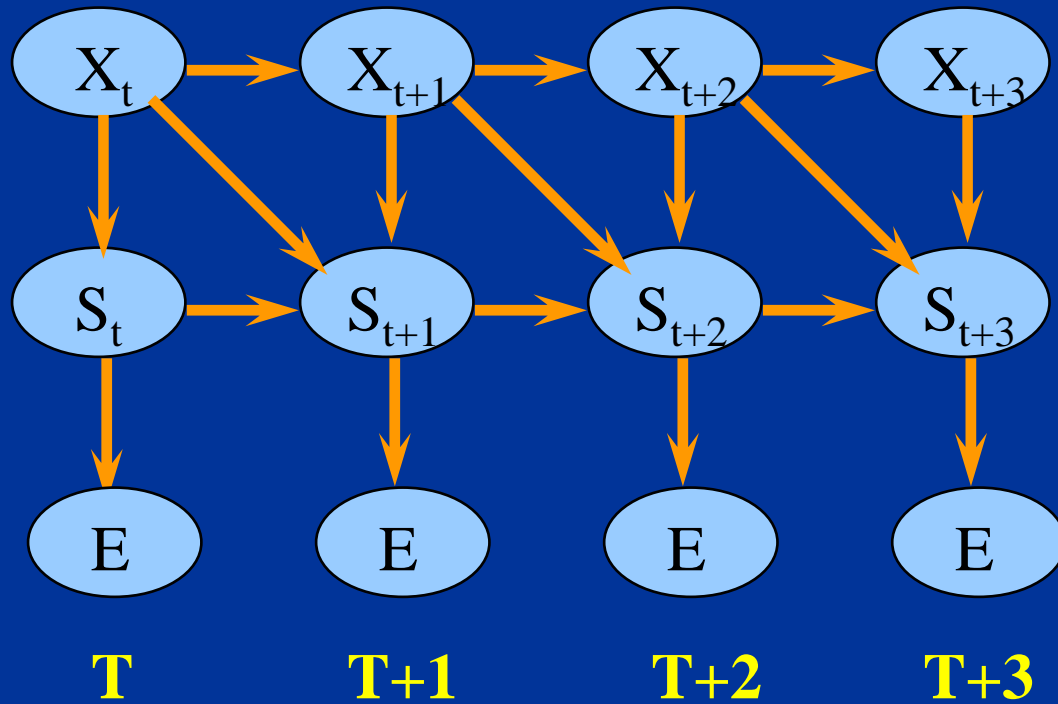
commands  
Auto ON/OFF  
sEarch ON/OFF  
Centre tip  
Tape/scope  
Learning  
Quit



# Dynamic Bayesian networks (DBN)

- BN for modeling temporal processes
- A “static BN” is repeated at each time (discrete time)
- Dependencies (arcs) between temporal slices (Markov assumption)
- Dependencies and parameters between time slices are repeated (Stationary assumption)
- Hidden Markov models (HMMs) are a special case of DBN

# Example of a DBN

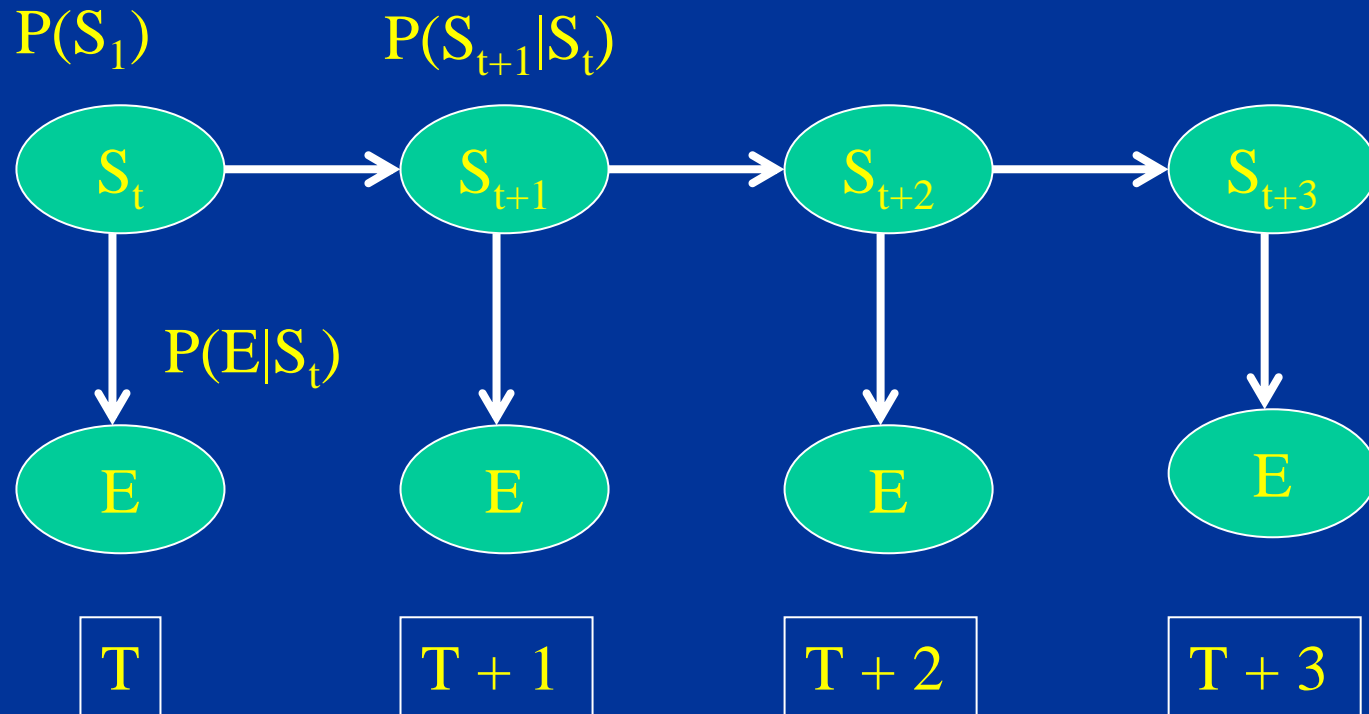




# Hidden Markov Models (HMM)

- A stochastic process in which the states are not directly observable
- It consists of:
  - A finite set of states
  - A probability distribution for transitions between states
  - A probability distribution for observations given the state

# HMM – Graphical Model



# HMMS

- **Representation**
  - Usually one model (HMM) is used to represent each *dynamic object*
- **Learning**
  - The structure is set by the designer and the parameters are learned from data
- **Recognition**
  - The data is fed to all the models and the one with the highest probability is selected

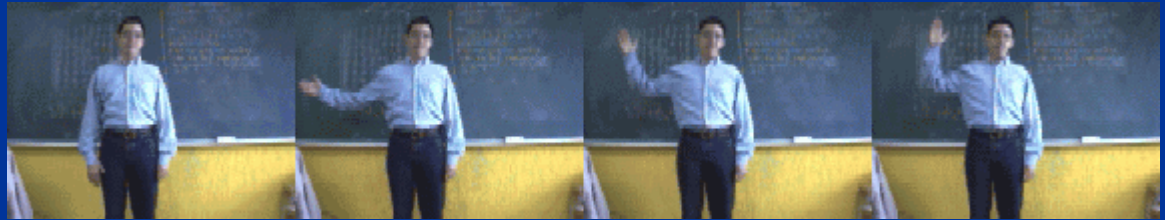
# Example HMMs: Gesture Recognition

- Motion features only
- HMMs
  
- Recognize 5 dynamic gestures with the right hand
- The gestures are for commanding a mobile robot
- Recognition based on HMM

Come



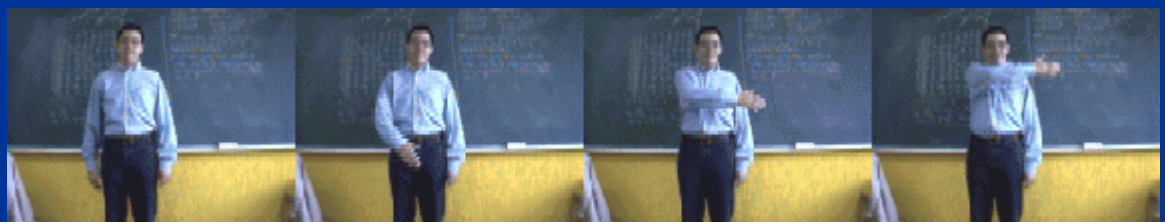
attention



go-right



go-left



stop

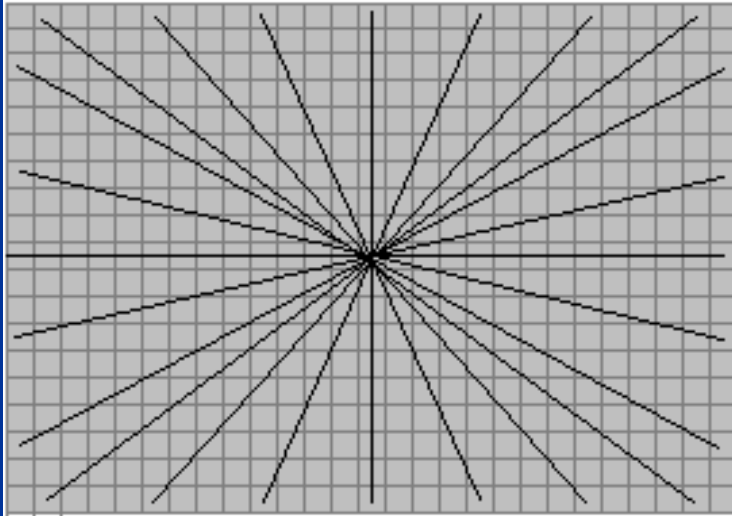


# Feature Extraction

- Skin detection
- Face and hand segmentation
- Hand tracking
- Motion features

# Segmentation

Radial scan for skin pixel detection

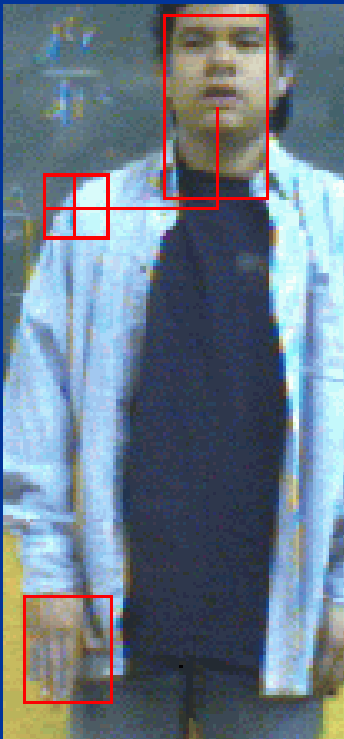


Segmentation by grouping skin pixels in the scan lines



# Tracking

Locate face and hand based on antropometric measures

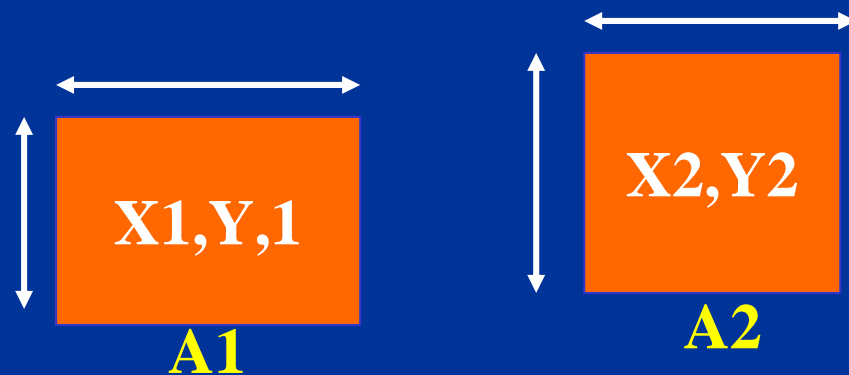


Track the hand by using the radial scan segmentation in region of interest



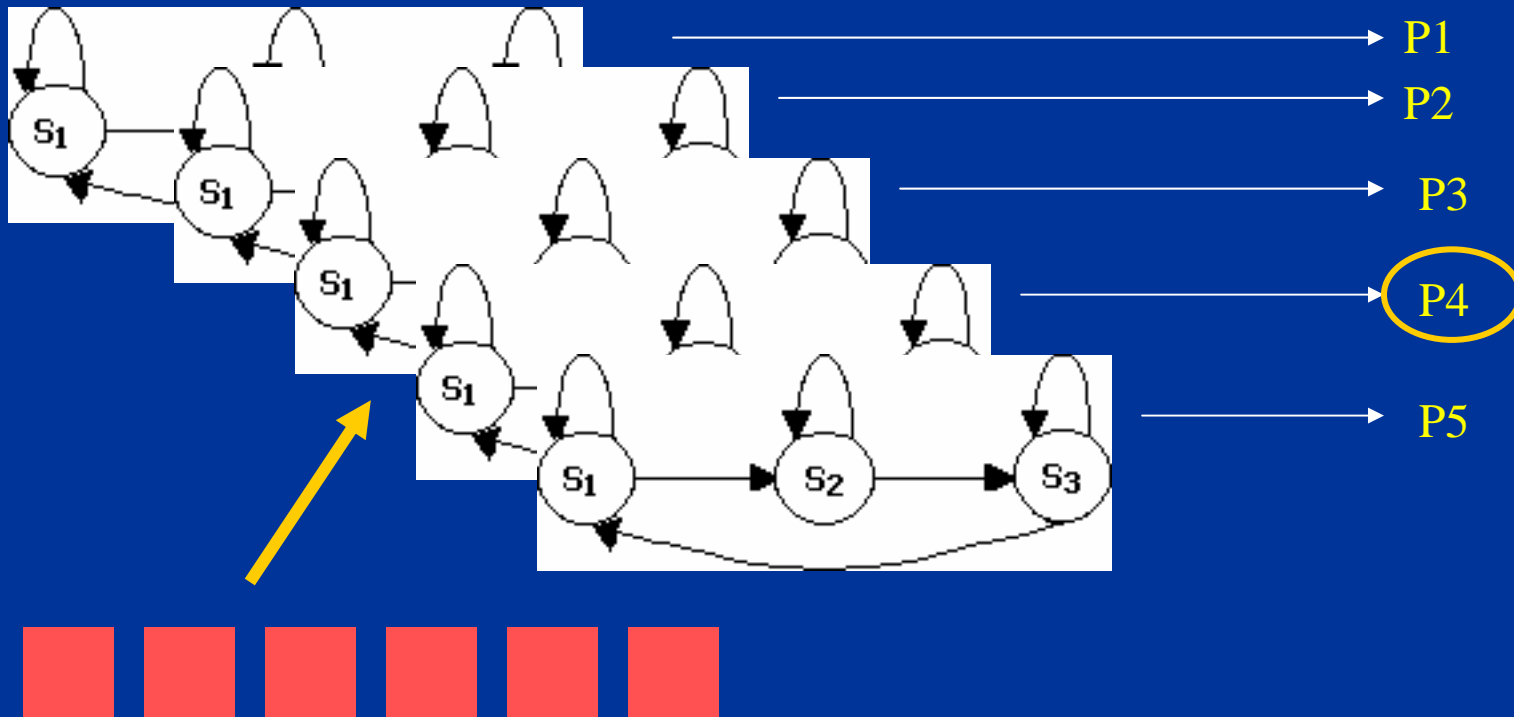
# Features

- From each image we obtain the features:
  - change in  $X$  ( $\Delta X$ )
  - change in  $Y$  ( $\Delta Y$ )
  - change in area ( $\Delta A$ )
  - change in size ratio ( $\Delta R$ )
- Each one is codified in 3 values: (+, 0, -)



# Recognition

- Recognition based on HMM
- One HMM for each gesture
- The probability of each model given the observations is estimated (forward algorithm) and the one with highest probability is selected



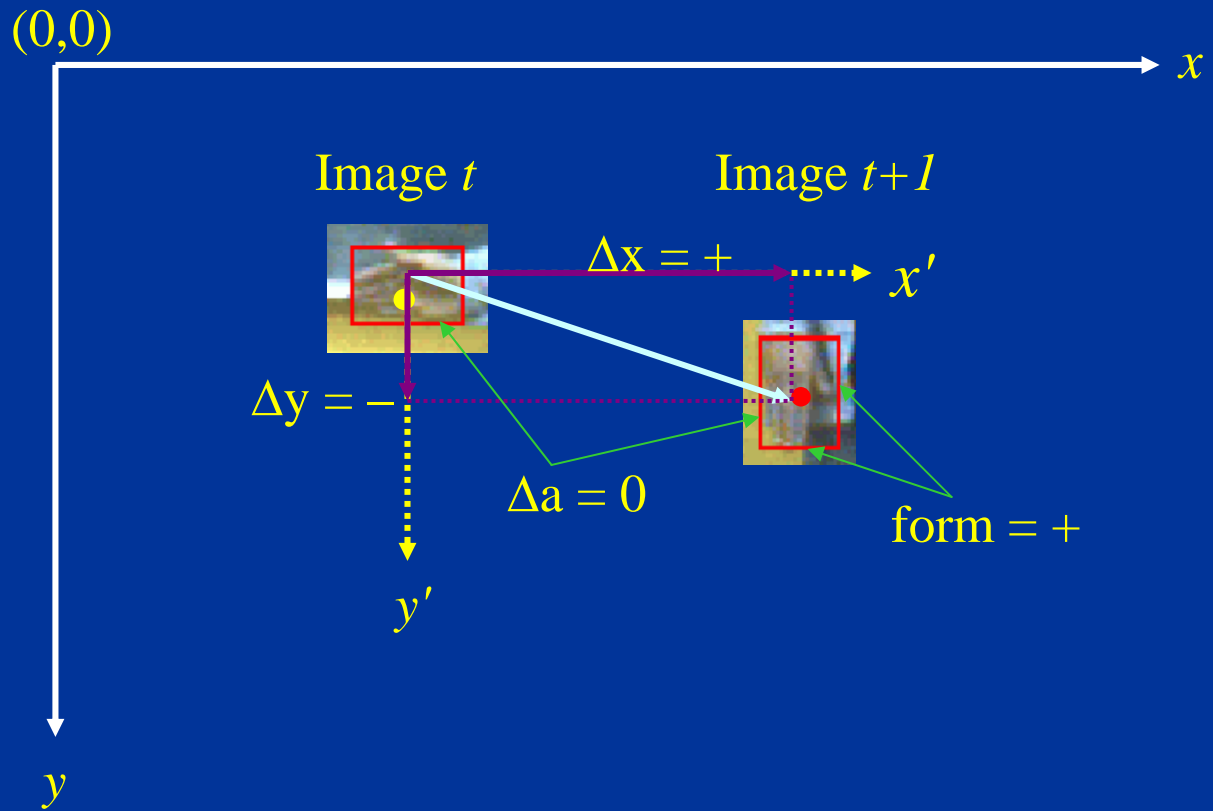
# Results

- Training with 128 sequences per gesture
- Testing with 80 sequences for each gesture
- Correct recognition:
  - come 75.3 %
  - attention 70.7%
  - stop 65.8 %
  - go-right 100 %
  - go-left 100 %
  - Average: 82%

## Example DBNs: Gesture recognition

- Recognize 5 dynamic gestures with the right hand (same as for HMM)
- Recognition based on **dynamic Bayesian networks**
- Include more features:
  - **Motion**
  - **Posture**

# Motion Features

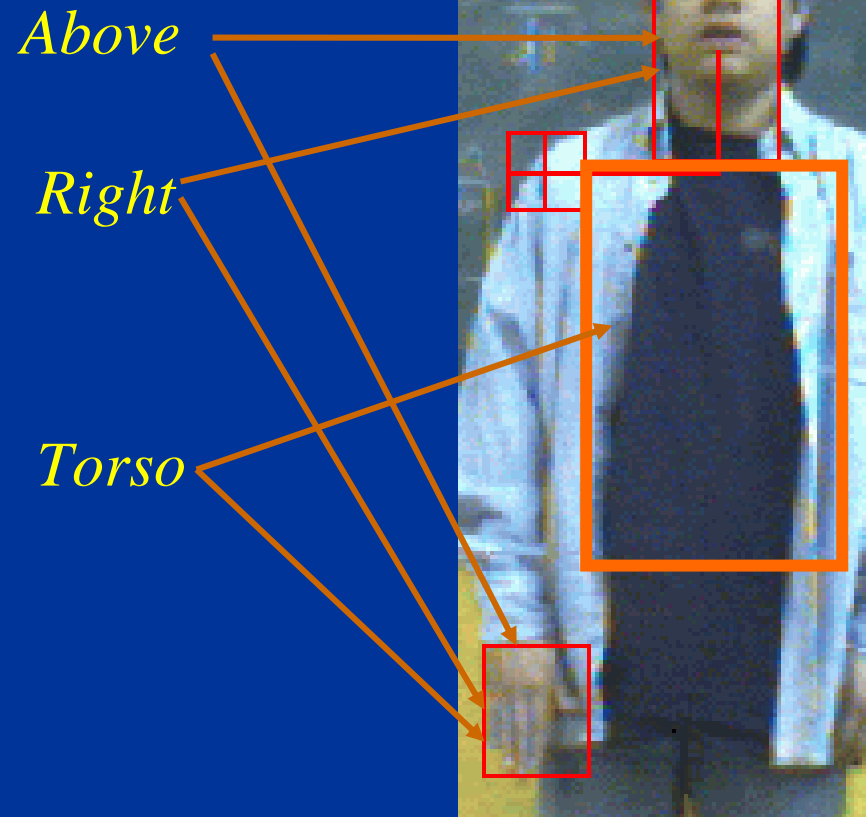


# Posture features

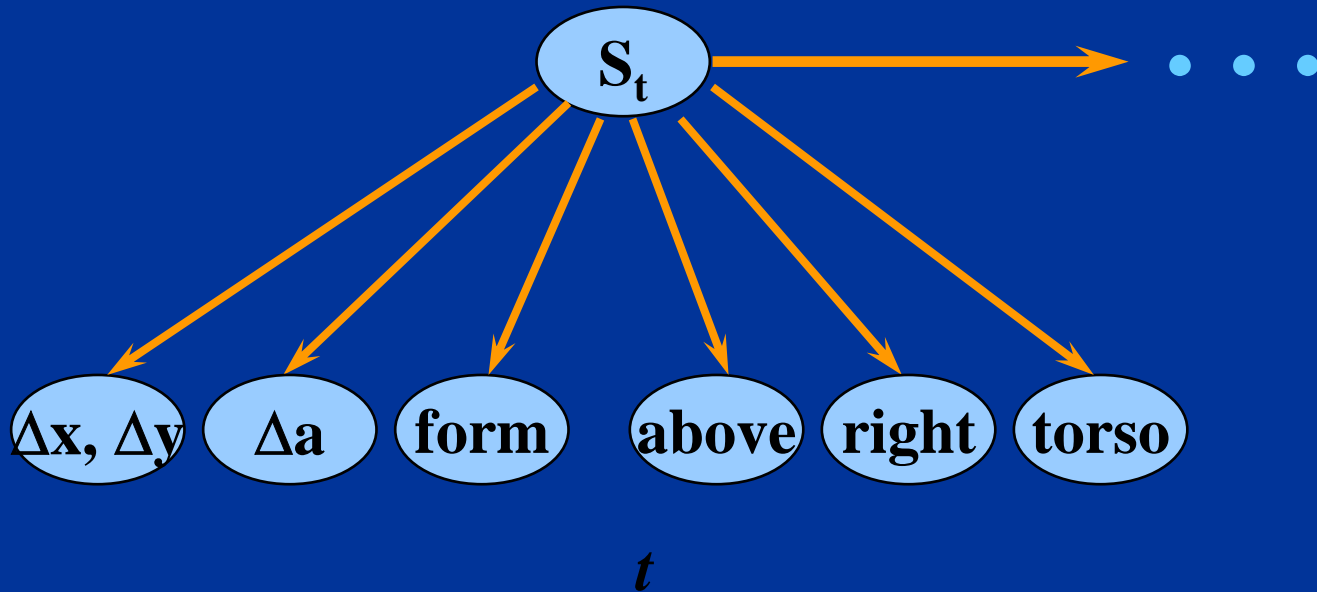
Posture features are simple spatial relations between the user's right hand and other body parts:

- *Right*
- *Above*
- *Torso*

Each one can take one of 2 values: (yes, no)



# Dynamic naive Bayesian Classifier with posture information



# Experiments

- 150 samples of each gesture taken from one user
- Laboratory environment with different lighting conditions
- Distance from the user to the camera varied between 3.0 m and 5.0 m
- The number of training samples varied between 5% to 100% of the training set



## Confusion matrix: DNBCs without posture information

	come	attention	go-right	go-left	stop
come	98 %			2 %	
attention	3 %	87 %	10 %		
go-right			100 %		
go-left				100 %	
stop	4 %	39 %		1 %	56 %

The average recognition rate is 87.75 %

## Confusion matrix: BCs with posture information

	come	attention	go-right	go-left	stop
come	100 %				
attention		100 %			
go-right			100 %		
go-left				100 %	
stop	11 %	6 %			83 %

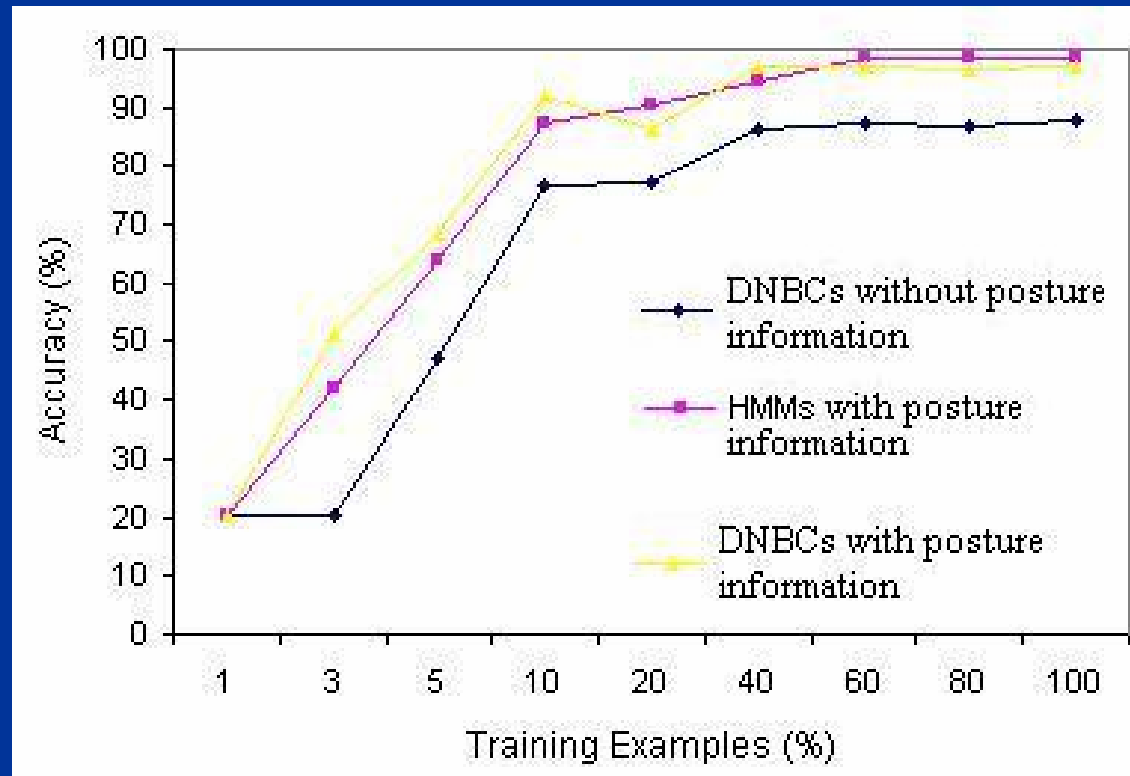
The average recognition rate is 96.75 %

## Confusion matrix: HMMs with posture information

	come	attention	go-right	go-left	stop
come	100 %				
attention		100 %			
go-right			100 %		
go-left				100 %	
stop	8 %				92 %

The average recognition rate is 98.47 %

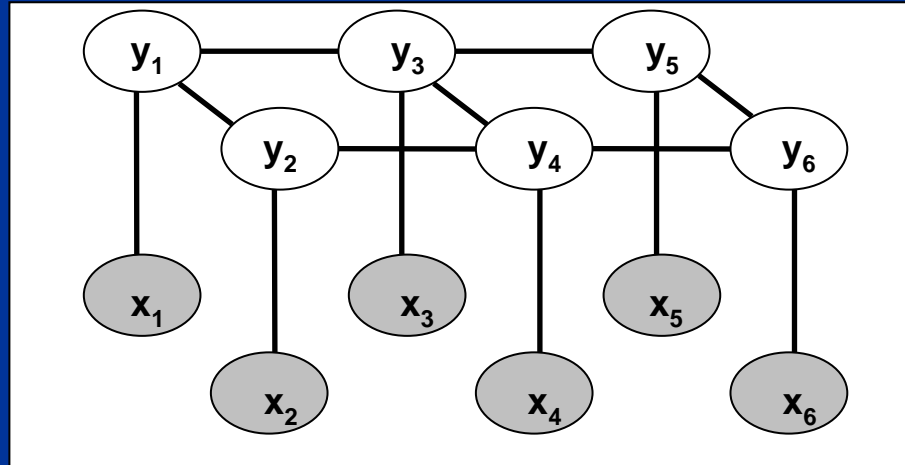
# Accuracy vs Training Size



Average recognition results of five repetitions of the experiment

# Markov Random Fields

- A set of random variables indexed by nodes in a graph.



- Observations: labels generated by base classifier
- Prior knowledge: associations between labels

$$P(f) > 0, \forall f \in \mathbb{F}$$

$$P(f_i | f_{s-\{i\}}) = P(f_i | N(f_i))$$

# Markov Random Field

$$P(f) = Z^{-1} \times e^{-\frac{1}{T}U(f)}$$

$$U(f) = \sum_{c \in C} V_c(f) + \lambda \sum_o V_o(f)$$

$$V_c(f) = \sum_c (P(w_c | w_i))^n$$

**Label's association**

$$V_o(f) = \left( \frac{1}{p_o^R(w_i)} \right)^n$$

**Confidence of the AIA method**

# Markov Random Field

- Representation
  - The fields through a graph a set of objects (nodes) and their relations (arcs)
- Learning
  - Usually the structure (vicinity) and parameters (potentials) are specified by the designer, although they could be learned from data
- Recognition
  - Based on the observed data, recognition consists on obtaining the most probable values for all the *objects* (configuration) via an optimization process

# Example

- Use of a MRF to improve automatic image annotation by incorporating:
  - Semantic associations
  - Spatial relations



# Semantic association between labels

# Use of external corpus

- Association:

$$V_c(f_i) = \sum_c (P(E_c|E_{f_i}))^n$$

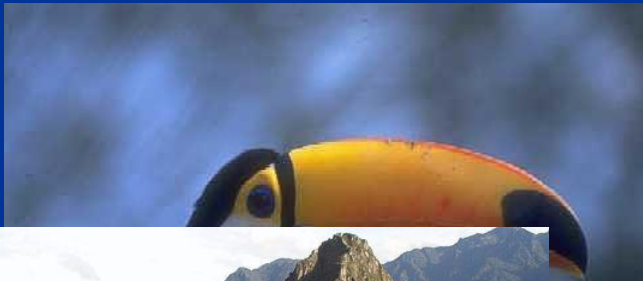
- *Association between two labels is given by the number of documents in which both labels occur within an external corpus of documents*

## Occurrence frequency

$$P(w_i|w_j) = \frac{P(w_i, w_j)}{P(w_j)} \approx \frac{c(w_i, w_j)}{c(w_j)}$$

# Label's association

- External corpus: IAPR-TC12-2006
  - 20,000 manually annotated images
  - Title, description,
  - location, date, notes



<TITLE>National Palace in Asunción</TITLE>  
<DESCRIPTION>a white building with lots of columns and arches, a neat lawn and neatly cut trees and bushes in the foreground; the flag of Paraguay is waving at the top of the building; there is a flower bed on the left;</DESCRIPTION>  
<NOTES>The National Palace was built in Versailles style from 1960 to 1992;</NOTES>  
<LOCATION>Asunción, Paraguay</LOCATION>  
<DATE>March 2002</DATE>

# Label's association

- External corpus: IAPR-TC12-2006

<b>SEA</b>	<b>MOUNTAIN</b>	<b>LAKE</b>	<b>CITY</b>	<b>PEOPLE</b>	<b>BLACK</b>	<b>BEACH</b>
beach	sky	shore	sky	light	man	light
light	range	light	buildings	black	light	man
coast	landscape	snow	clouds	street	woman	people
man	snow	man	mountain	man	wall	rocks
rocks	clouds	people	skyline	buildings	girl	waves
people	light	slope	light	woman	road	palm
black	lake	landscape	river	slope	house	woman
waves	slope	woman	street	wall	street	coast
woman	man	range	sea	plaza	shirt	black
island	people	black	landscape	meadow	landscape	boats

# Label's association

- External corpus: IAPR-TC12-2006

SEA	MOUNTAIN	LAKE	CITY	PEOPLE	BLACK	BEACH
beach	sky	shore	sky	light	man	light
light	range	light	buildings	black	light	man
coast	landscape	snow	clouds	street	woman	people
man	snow	man	mountain	man	wall	rocks
rocks	clouds	people	skyline	buildings	girl	waves
people	light	slope	light	woman	road	palm
black	lake	landscape	river	slope	house	woman
waves	slope	woman	street	wall	street	coast
woman	man	range	sea	plaza	shirt	black
island	people	black	landscape	meadow	landscape	boats

# Label's association

- External corpus: IAPR-TC12-2006

SEA	MOUNTAIN	LAKE	CITY	PEOPLE	BLACK	BEACH
beach	sky	shore	sky	light	man	light
light	range	light	buildings	black	light	man
coast	landscape	snow	clouds	street	woman	people
man	snow	man	mountain	man	wall	rocks
rocks	clouds	people	skyline	buildings	girl	waves
people	light	slope	light	woman	road	palm
black	lake	landscape	river	slope	house	woman
waves	slope	woman	street	wall	street	coast
woman	man	range	sea	plaza	shirt	black
island	people	black	landscape	meadow	landscape	boats

# Label's association

## Occurrence frequency

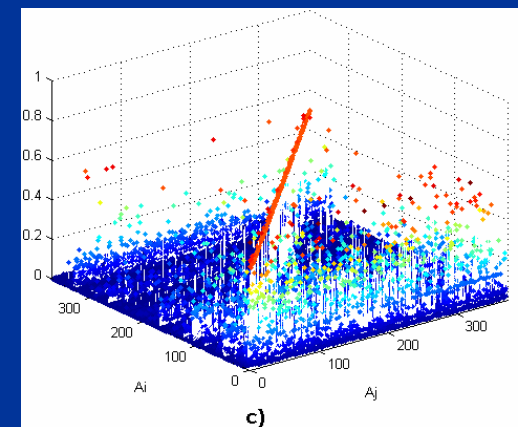
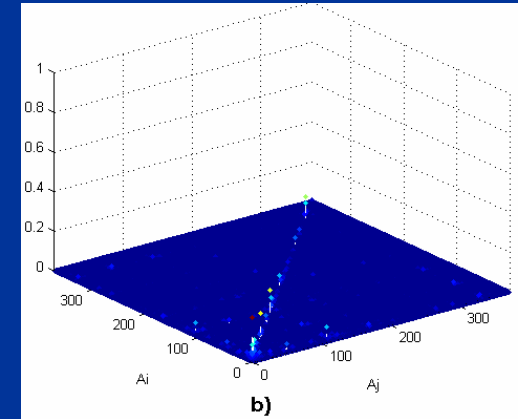
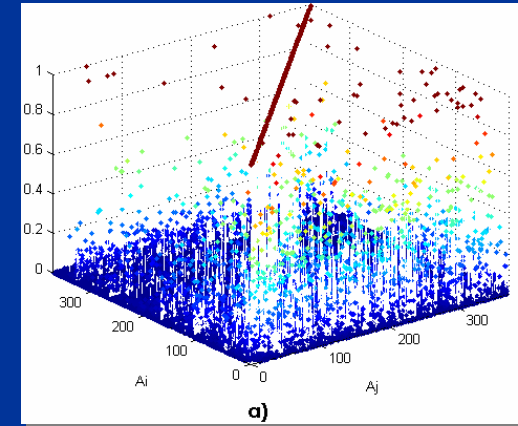
$$P(w_i|w_j) = \frac{P(w_i, w_j)}{P(w_j)} \approx \frac{c(w_i, w_j)}{c(w_j)}$$

## Laplacian Smoothing

$$P(w_i|w_j) \approx \frac{c(w_i, w_j) + 1}{c(w_j) + |V|}$$

## Interpolation smoothing

$$P(w_i|w_j) \approx \lambda * \frac{c(w_i, w_j)}{c(w_j)} + (1 - \lambda) * c(w_j)$$





# Experiments: Corel data set

Data set	# Images	Words	Training blobs	Testing blobs
<i>A-NCUTS</i>	205	22	1280	728
<i>A-P32</i>	205	22	3288	1632





# Experiments

- Annotation method: *knn*

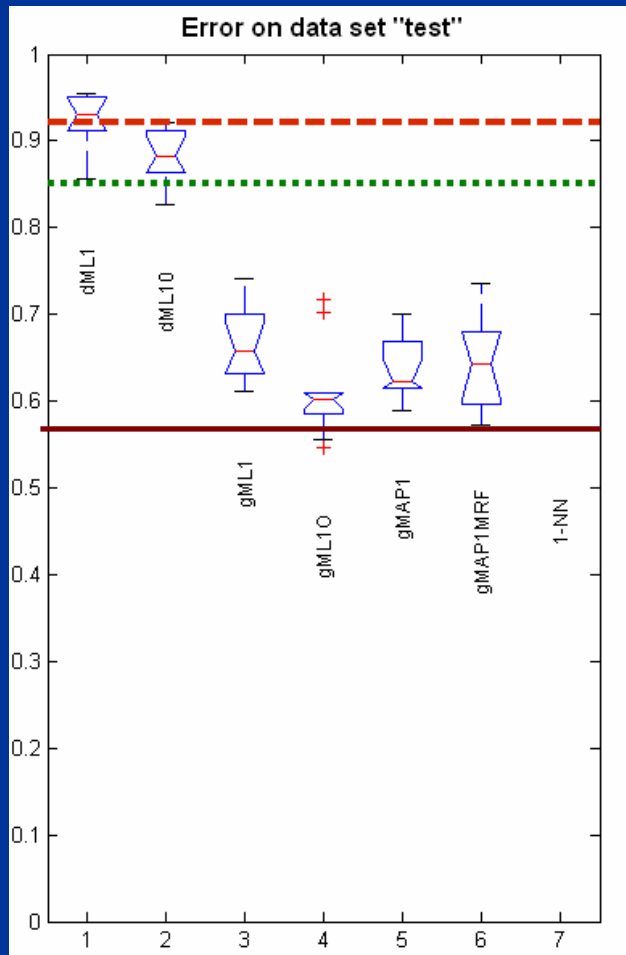
$$P^R(y_j^t) = \frac{d_j(x^t)}{\sum_i^k d_i(x^t)}$$

- Comparison of KNN against semi-supervised methods  
*dML1, dML10, gML1, gML10, gMAP1, gMAP1MRF*
- Comparing error (*e*) at the top *x*-labels

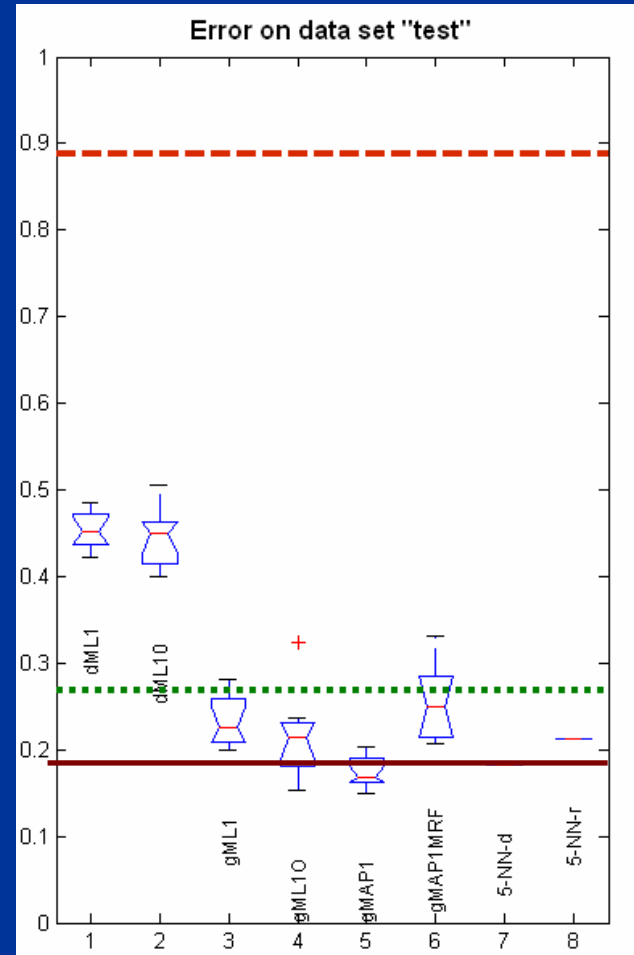
$$e = \frac{1}{N} \sum_{n=1}^N \frac{1}{M_n} (1 - \delta(\bar{a}_{nu} = a_{nu}^{max}))$$

# Experimental results

## k-NN versus others

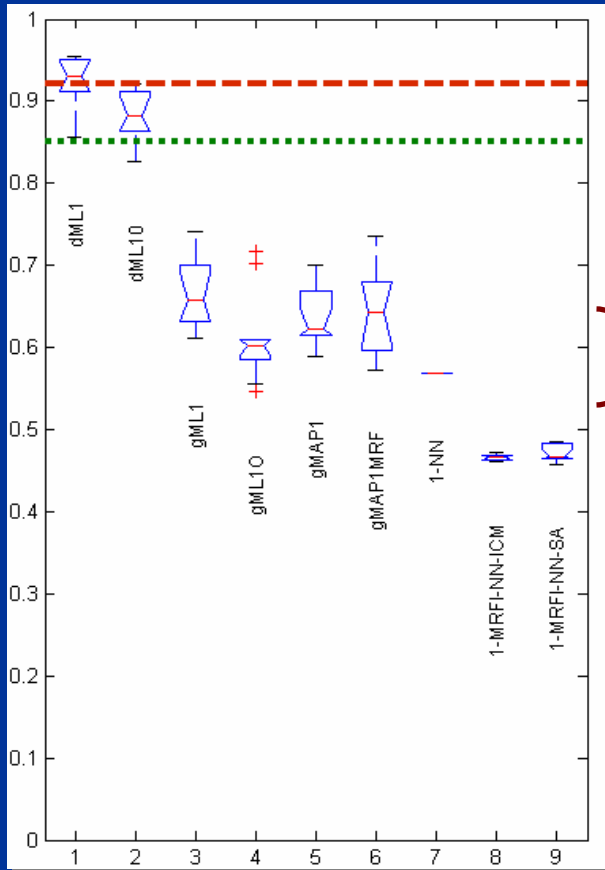


Error at the first label

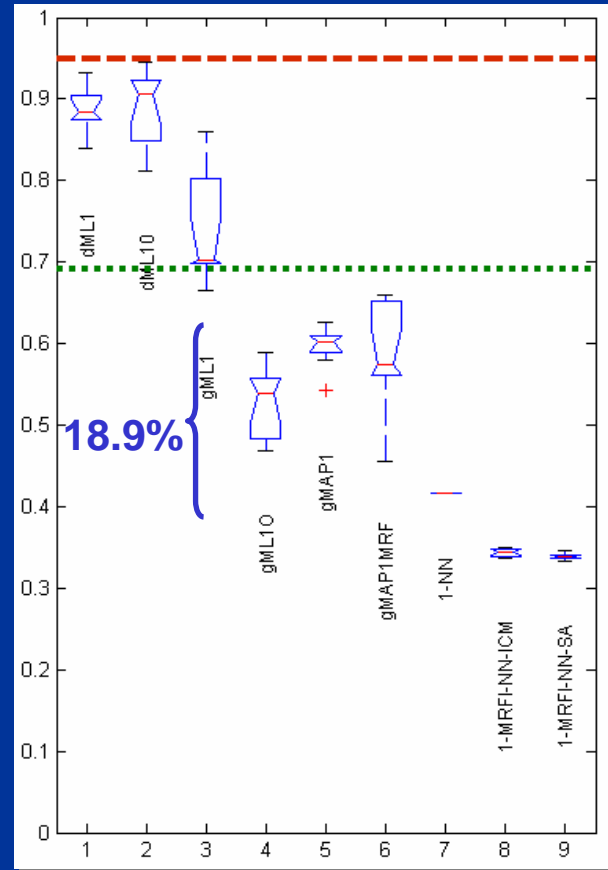


Error at the 5-labels

# Experimental results



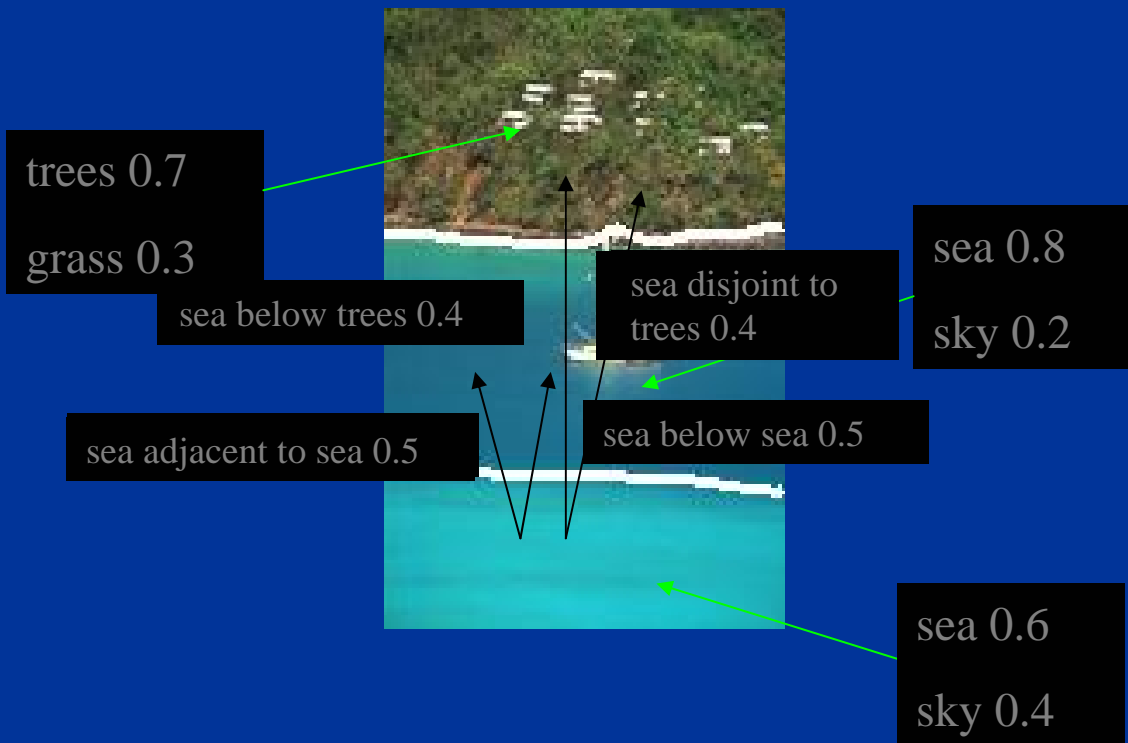
COREL-A ncuts segmentation



COREL-A grid segmentation

# Incorporating Spatial Relations

# Considering spatial relations



Sky below sea?

Sky below trees?

# Spatial Relations

- Types of relations:
  - Topological
  - Order
  - Metric

		Directed	Undirected
Topological relations		1	Adjacent
		2	Disjoint
Order relations	Horizontal relations	3	Beside (either left or right)
		4	Horizontally aligned
	Vertical relations	5	Above
		6	Below
		7	Vertically aligned

# Potentials

$$U_p(f) = \alpha_1 V_T(f) + \alpha_2 V_H(f) + \alpha_3 V_V(f) + \lambda \sum V_o(f)$$

- Energy potentials are inversely proportional to the probability of each relation:

$$V_T(f) = \frac{1}{\sum_c P_{1c}(f) \oplus P_{2c}(f)}$$
$$V_H(f) = \frac{1}{\sum_c P_{3c}(f) \oplus P_{4c}(f)}$$
$$V_V(f) = \frac{1}{\sum_c P_{5c}(f) \oplus P_{6c}(f) \oplus P_{7c}(f)}$$

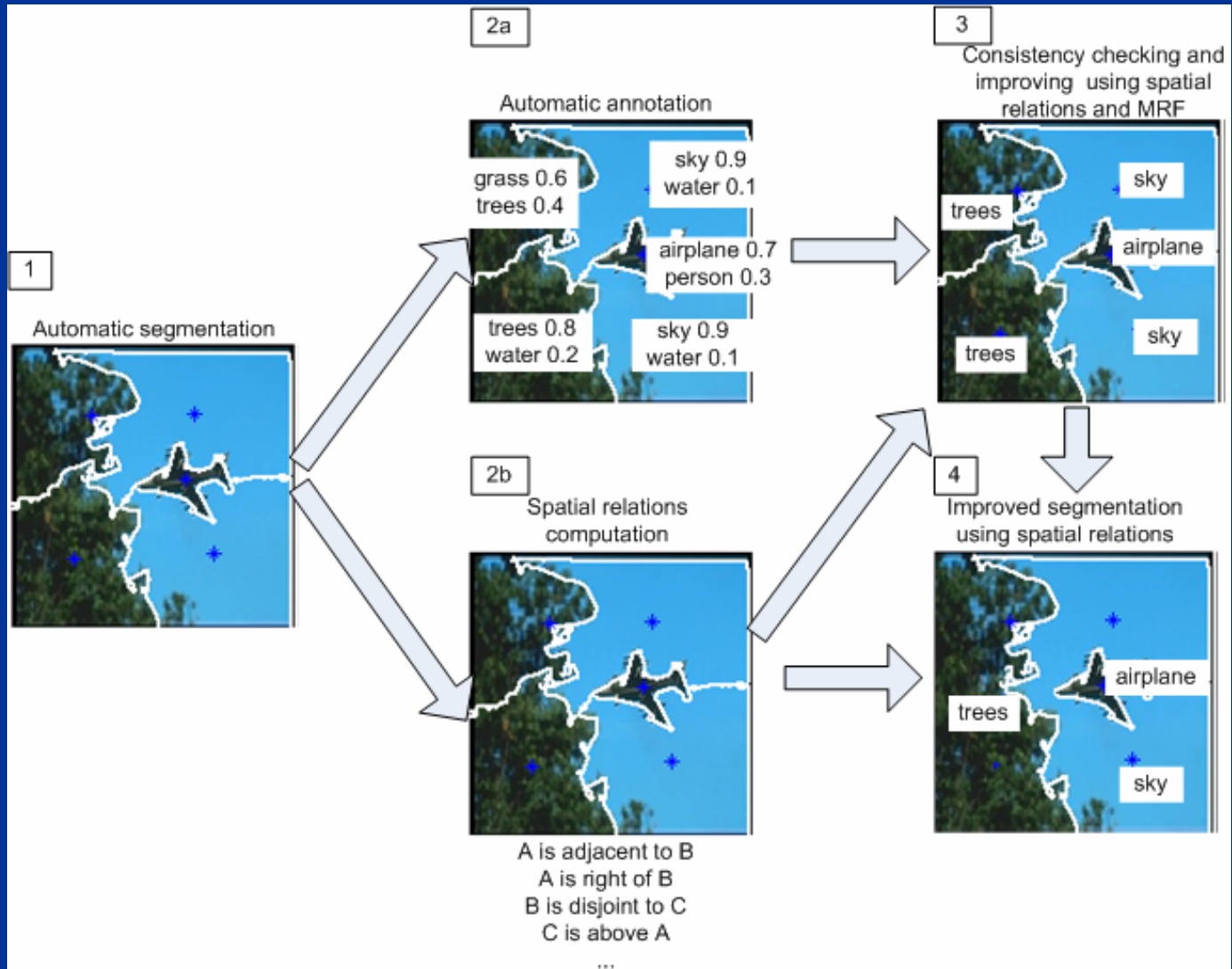
## Estimating potentials

- The probability of each relation is obtained by combining a subjective estimate ( $E_k$ ) with data from labeled images ( $Rel_k$ ):

$$P_{kc}(f) = \frac{Rel_k(i,j) + \delta E_k(i,j)}{NR(i,j) + \delta 100}$$



# Methodology



# Experiments

- Data set:
  - Corel A database
  - 205 landscape images, segmented with normalized cuts
  - 137 training images and 68 test images
  - 22 possible labels

airplane	grass	mountains	sky
bird	ground	pilot	snow
boat	horse	road	trees
church	house	rock	water
cow	lion	sand	
elephant	log	sheep	

# Experiments

	Expert knowledge	Laplacian smoothing	No smoothing
Topological relations			
Vertical relations			
Horizontal relations			
All relations			

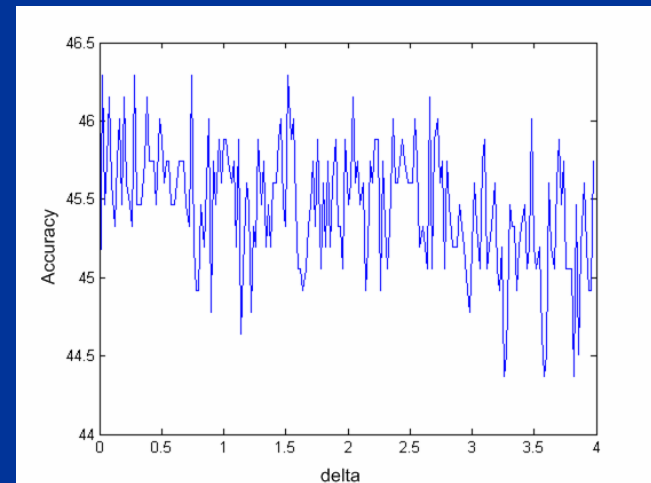
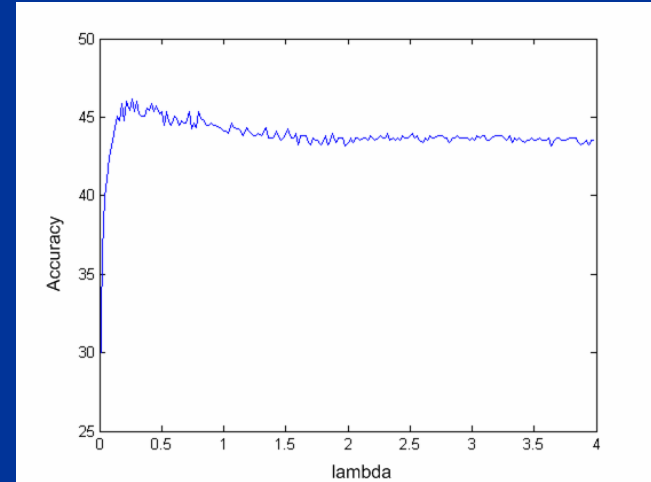
- We used kNN as base classifier for region annotation

# Parameters

Parameter	Value
$\lambda$	0.25
$\delta$	0.25
$\alpha_1$	1
$\alpha_2$	1
$\alpha_3$	1
$T$	116
$n$	400

The best configuration is based on MAP using simulated annealing:

$$T = \frac{T}{\log(100 + j)} \log(100)$$



# Results

Algorithm	Relation group	Smoothing	Accuracy
kNN	None	None	36.81%
MRFs	Topological	None	42.72%
		Laplacian	43.51%
		Expert info.	43.25%
	Horizontal	None	41.72%
		Laplacian	43.08%
		Expert info.	43.58%
	Vertical	None	43.73%
		Laplacian	44.93%
		Expert info.	44.88%
	All	None	43.29%
		Laplacian	45.41%
		<b>Expert info.</b>	<b>45.64%</b>

# Improving segmentation

**Original  
segmentation**



**Segmentation  
improvements after  
kNN**



**Segmentation  
improvements after  
using spatial  
relations and MRF**



**Correct  
segmentation  
(manual)**





# Referencias

- Ullman: Cap. 22, 23
- C. Hernández--Gracidás, L.E. Sucar, "Markov Random Fields and Spatial Information to Improve Automatic Image Annotation", *Advances in Image and Video Technology, Lecture Notes in Computer Science 4872*, Springer-Verlag, pp. 879--892, 2007.
- H. J. Escalante, M. Montes, and L. E. Sucar, "Word cooccurrence and Markov Random Fields for Improving Automatic Image Annotation", *British Machine Vision Conference, Vol. II*, pp. 600--609, 2007.