Object Recognition using Vision and Machine Learning

Henrik Andreasson
Learning Lab, AASS 2005

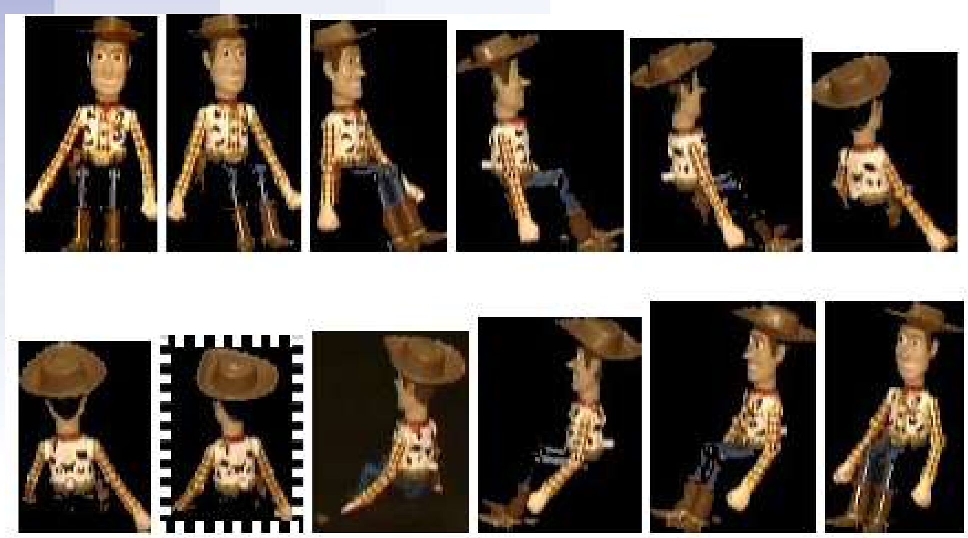
Introduction

- **Def: Object Recognition**
  - **Detect known (*learned*) objects in an image. Most often, (*”always”*) the objects to be learned are pre-segmented.**

Samples from CODID database.

More definitions

- **Object Detection :**
  - Detect the location of the objects in the image.
  - Some recognition algorithm need to have pre segmented images in the unknown image as well.

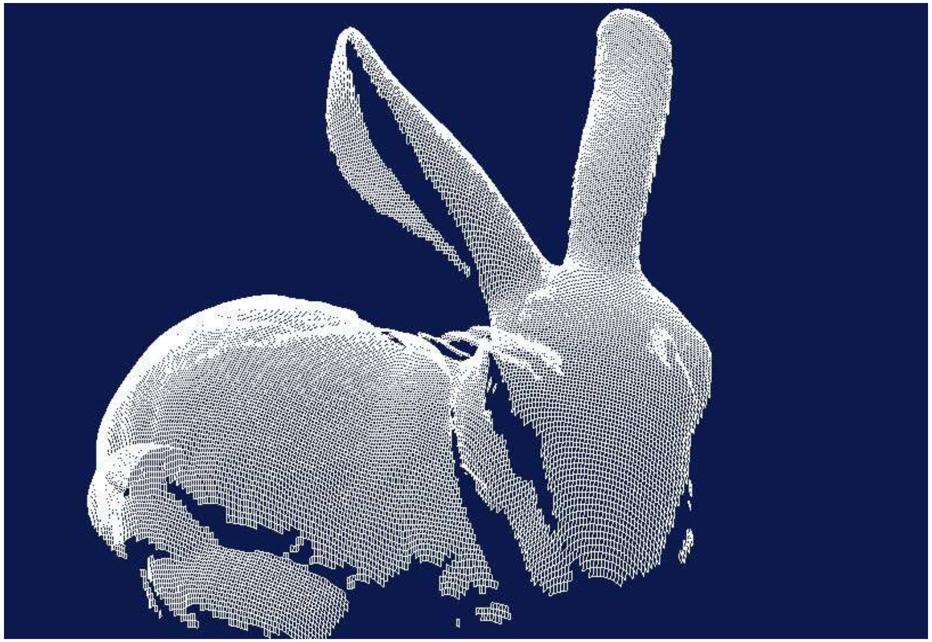


Colour co-occurrence histogram

Sensors

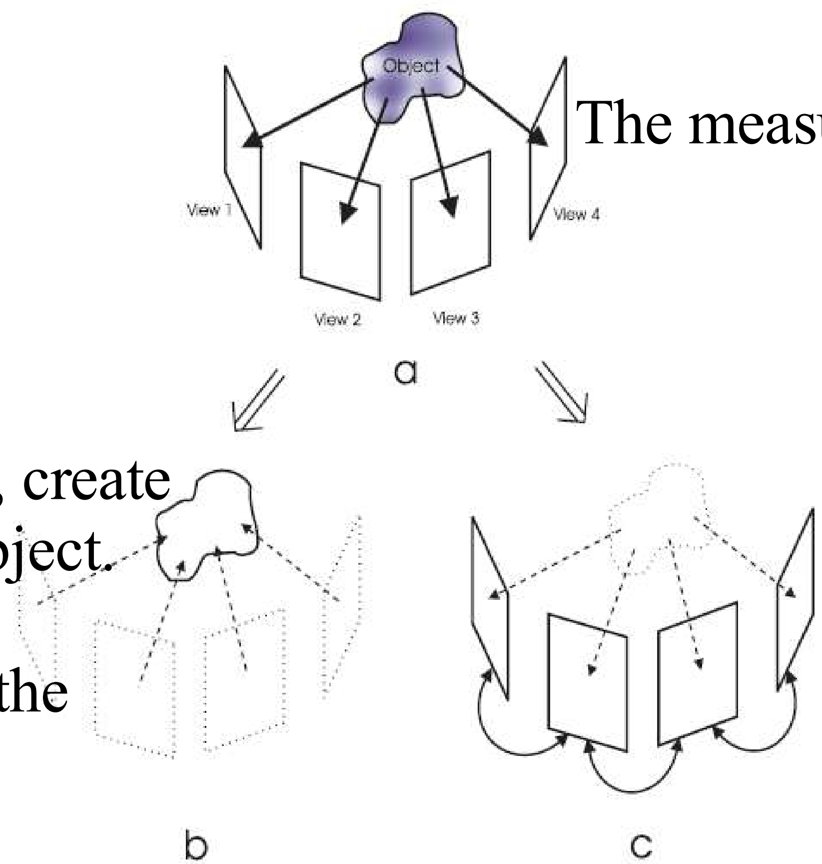
- **Sensor used in this presentation :**
  - **Single camera**
- **Other sensors**
  - **Range data - laser, stereo camera**
  - **Gas sensors**
  - **...**

Bunny, IVPR database



How to represent an object?

- **Geometry based**
- **Appearance based**



The measure

Geometry Based:
By using the images, create a 3D model of the object.

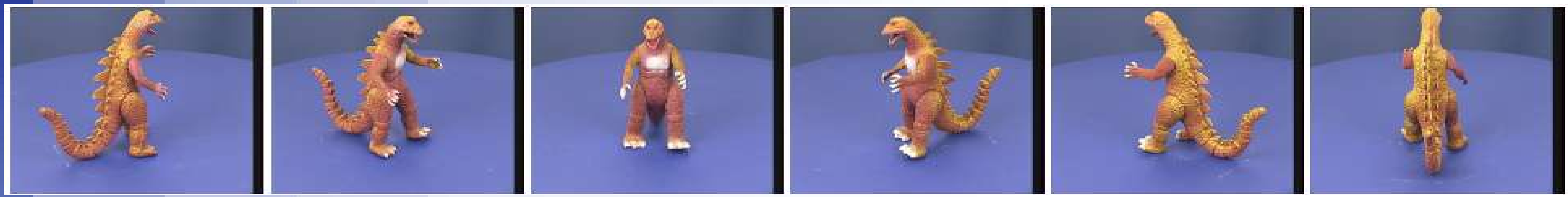
A 3D model will be the representation.

Appearance based
The object is represented by the recorded images.

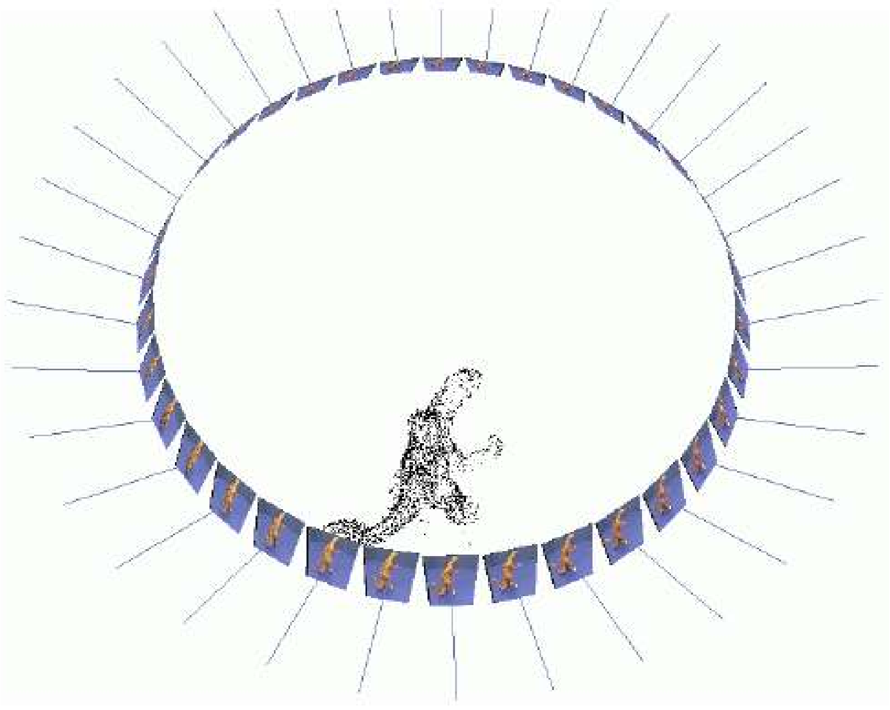
Features are extracted from each image.

How to represent an object?

- **Appearance based :**



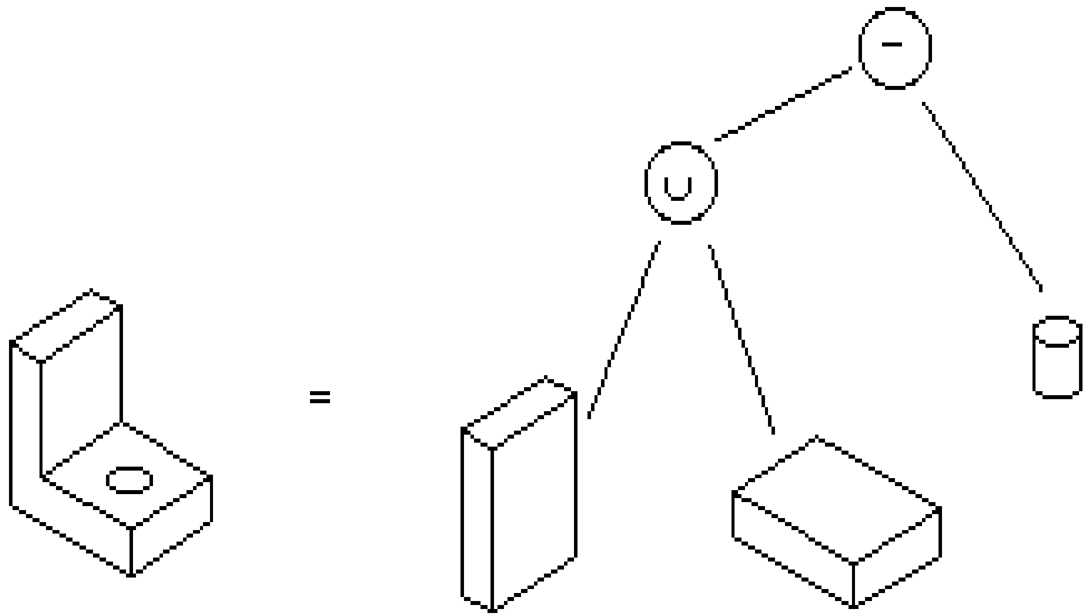
Geometry based :



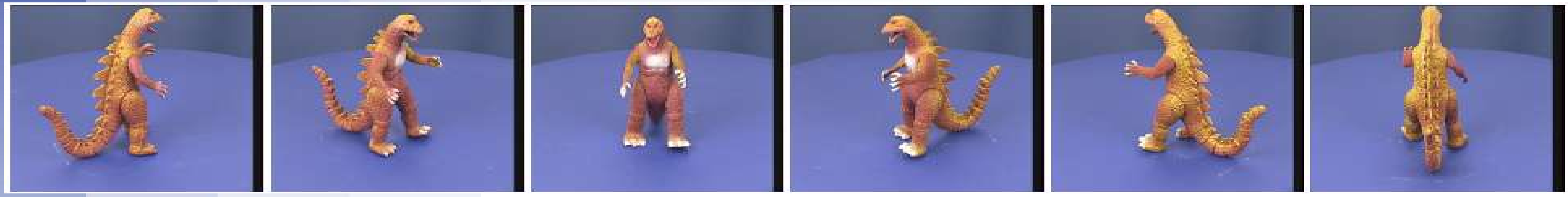
Oxford group
Zisserman et. al

How to represent an object? (one theoretical example)

- Computational Solid Geometry (CSG)
  - Each object consists of a set of simple building blocks. (*This approach is not really used with real data*).



Method used in this presentation



Overview of the method

1) Extract features from the segmented object to be learned (*for each view*).

2) ”Store” the features *(for all objects that should be learned)*

3) For an unknown image with unknown content, extract features and estimates which objects that fits best to the learned objects.

Connection to Machine Learning

1) **Extract features from the segmented object to be learned (*for each view*).**

(NOT HERE!) *(AdaBoost and face detection.)*

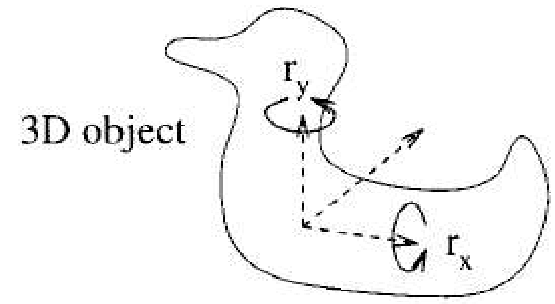
2) **"Store" the features *(for all objects that should be learned)***

(HERE!)

3) **For an unknown image with unknown content, extract features and estimates which objects that fits best to the learned objects.**

(HERE!)

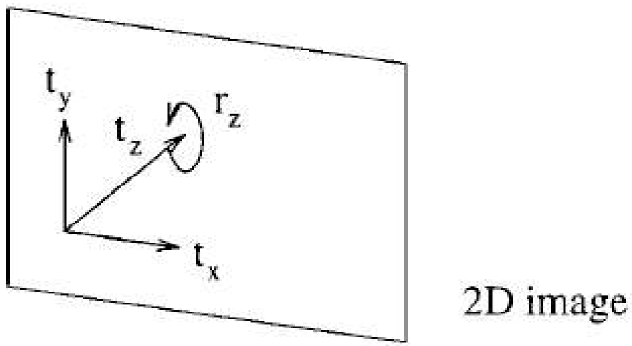
Why use features...



... and not simply

pixel subtraction?

BECAUSE:



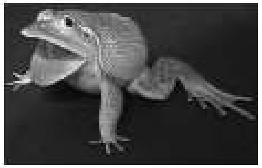
- *Similarity transforms :*
  - Don't know where in the image the object is, translation in x, y, rotation around z-axis. Change of scale : translation in z.
- *3D transformation :*

  Rotation around x and y-axis.

Need some invariance (!)

- *Scene changes :*
  - Occlusion of the object (partial NOT the whole object). Changes in the background.
- *Light condition :*
  - Different lightning changes the pixel values.
- *Changes in the sensor :*
  - Different level of noise, blur *(lens)*, saturation.

Images
from D. Lowe



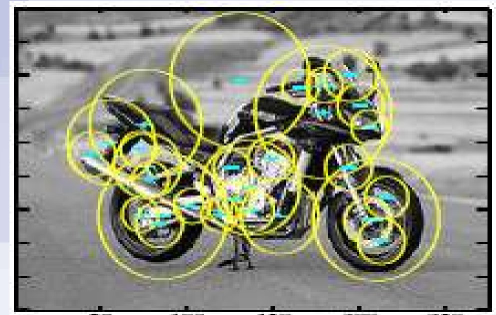
Invariant features

- **Usually NOT strong connection with ML**
  - **(can be used to extract which (how they should be extracted) features to use, used for generic recognition such as faces).**
- **Very core part in Object Recognition**

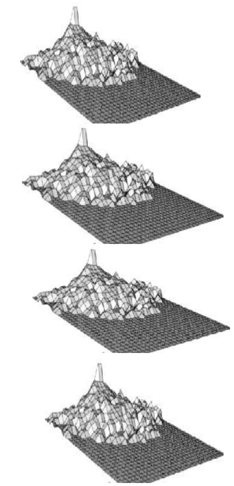
- ***Global***
  - **the whole training image is used at once.**
- ***Local***
  - **small regions in the training image is used.**

Local vs. Global

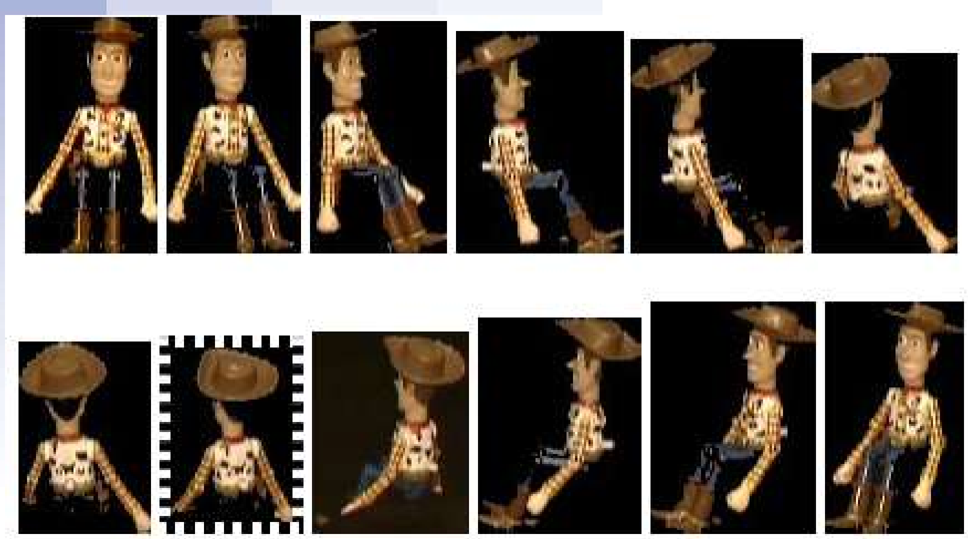
- **Local**



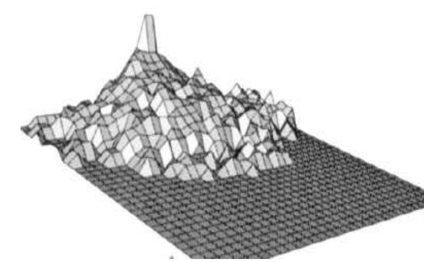
One set of feature per point



Global



One set of features per image

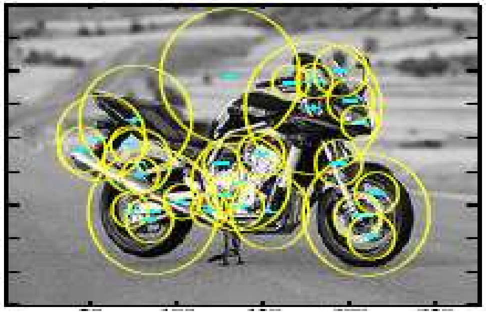


How is the matching done?

- 2 sets of feature
    - One set extracted from the image
    - The other is the database.
- Database contains of knowledge of learned objects.
- How the matching is done is depending on the database (i.e. think of ANN's).
    - Simple method is to calculate the Euclidean distance and find the closest match.
        - Hash tables and approaches similar to KD-trees is commonly used due to their speed.

Which chapters in Mitchell's ML book?

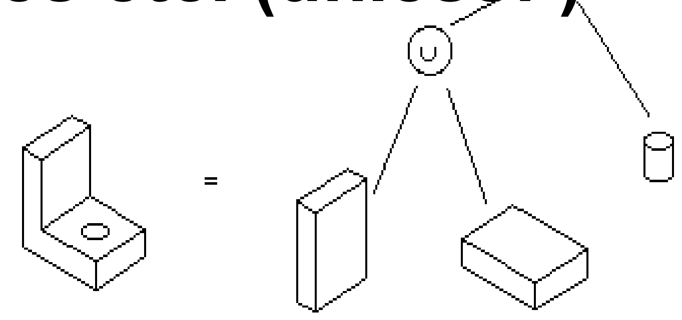
- The properties what we want of a feature.
  - Stable
  - Invariant
  - Discriminative
  - Small changes should give small changes in the descriptor



- Note that there is some contradiction... how to have a feature that is both very invariant and yet very discriminative?!?
- Don't have ”<play tennis> yes/no”.
- IF / THEN / ELSE won't work here.

Which chapters in Mitchell's ML book?

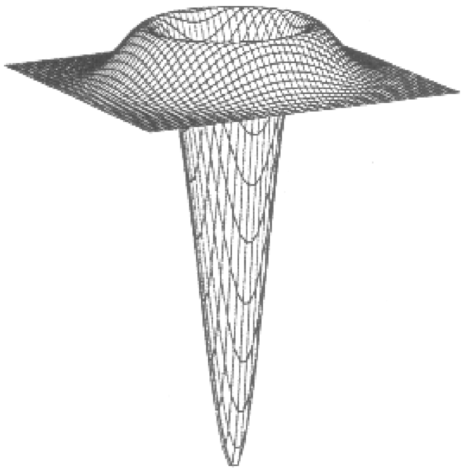
- What we have:
    - Many inputs (long feature vectors), where each number gives only very limited knowledge.
  - Artificial Neural Network
    - Face recognition example in the book.
  - Bayesian Learning.
  - Instance-Based Learning (k-Nearest Neighbour).
  - ”NOT” : Learning Sets of Rules etc. (unless:-)



An approach using Bayesian Learning

- From B. Schele et al, *Recognition without Correspondence using Multidimensional Receptive Field Histograms, from 2000 (a bit old, nice work)*

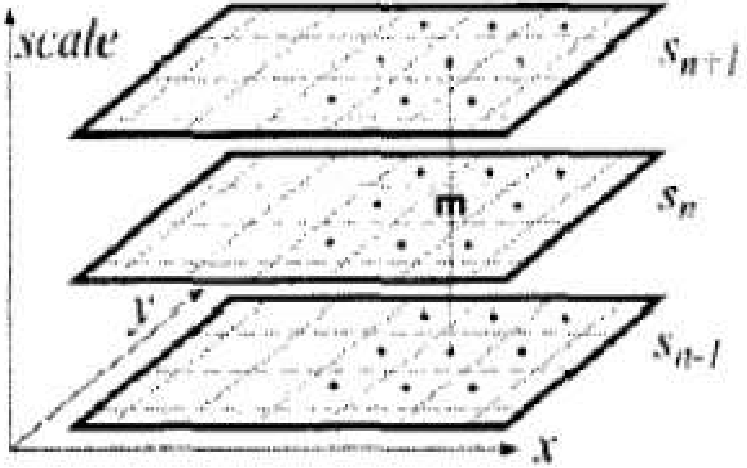
- **Uses a set of features based on derivatives of different scales.**
- **Derivatives : stable towards illumination changes.**
- **Magnitude is rotational invariant.**

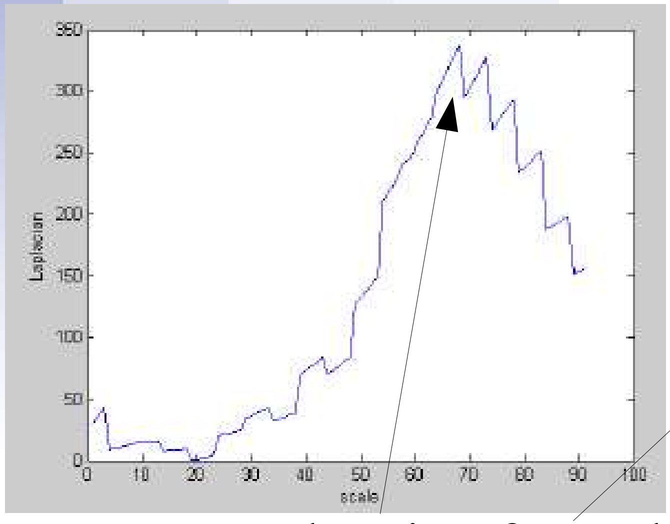


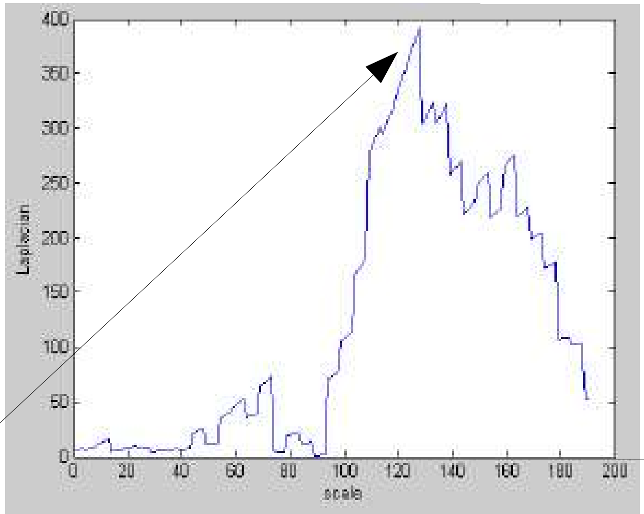
$$Mag(x,y) = \sqrt{\left(G_x^{\sigma}(x,y)\right)^2 + \left(G_y^{\sigma}(x,y)\right)^2}$$

What is scale?

$$\left|\sigma^2\left(L_{xx}(\vec{x},\sigma)+L_{yy}(\vec{x},\sigma)\right)\right|$$







x-axis :
scale - level of blur
(sigma in a Gaussian
convolution kernel)

Get the size from the peaks

Probability density function

$$P\left(M \mid object, other\right)$$

Feature vector

The object

Orientation
Transformation
Lightning
Scale :-)

The "core" idea of the algorithm

- **Learn the PDF (Probability Density Function) for an image of the object**

$$P(M \mid object)$$

Feature vector The object

Use Bayes rule to get :

  - **Means, the probability that the image contain object with feature vector M.**

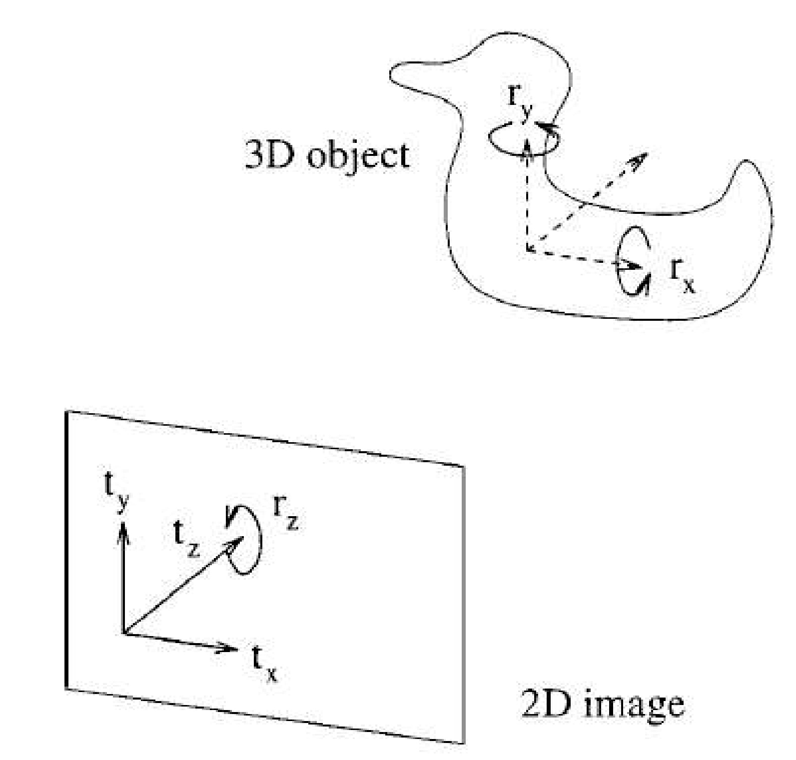
$$P(object \mid M)$$

Other?

$$P(M \mid object, other)$$

- Other consists of:
  - Translation, (Tx, Ty, Tz) *note Tz = scale*
  - Rotation (Rx, Ry, Rz)
  - Scene changes – occlusion rates
  - Light conditions – intensity, colour etc.
  - Imaging condition – blur, noise, saturation.

$$p(M \mid o_n, R, T, S, L, I)$$



$$p(M \mid o_n, R, T, S, L, I)$$

$$\downarrow$$

$$p(M \mid o_n, r_x, r_y, r_z, t_z)$$

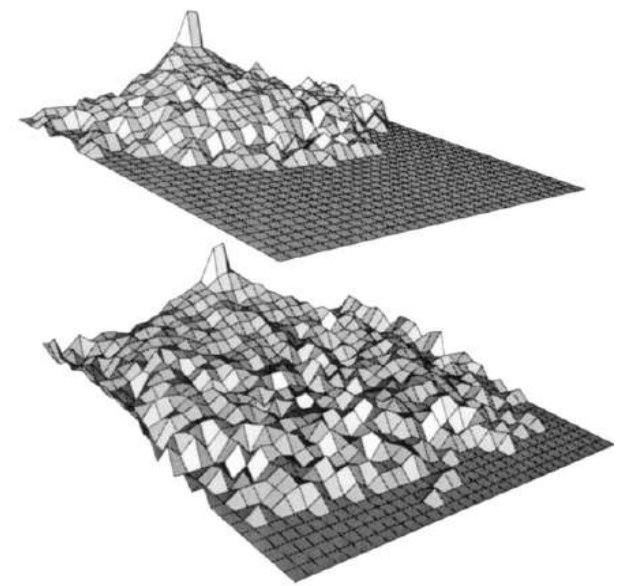
- ”Nothing to do about”:
  - *Scene changes – occlusion rates*
- Feature invariance tries to take care of :
  - *Tx, Ty*

    Light conditions – intensity, colour etc.

    Imaging condition – blur, noise, saturation.

The image database consists of...

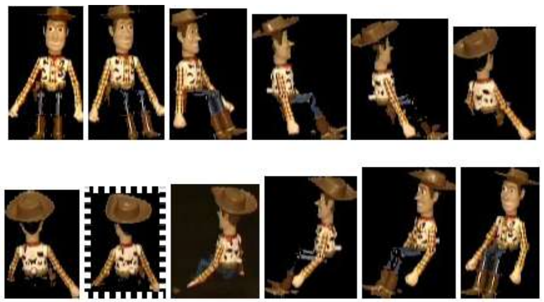
- **103 different objects**
- **2130 images -> with different scales and rotations (Rx, Ry, Rz)**



Different scales...

The created database consist of…

- The whole image is used (global)
- 6D histogram, 24 bins each
- 3 different scale
- First order derivatives
- Total of 10.000 histograms



How is the segmentation done? - Object detection

- **It's not done**
  - **Assumes the object is the whole image**

Bayes rule

$$p(o_n \mid m_k) = \frac{p(m_k \mid o_n)\, p(o_n)}{p(m_k)} = \frac{p(m_k \mid o_n)\, p(o_n)}{\sum_i p(m_k \mid o_i)\, p(o_i)}$$

- **What if we have two M-vectors (features)?**
  - **Use them both!**

$$p(o_n \mid m_k \wedge m_j) = \frac{p(m_k \wedge m_j \mid o_n)\, p(o_n)}{\sum_i p(m_k \wedge m_j \mid o_i)\, p(o_i)}$$

  - **Assume they're independent:**

$$p(o_n \mid m_k \wedge m_j) = \frac{p(m_k \mid o_n)\, p(m_j \mid o_n)\, p(o_n)}{\sum_i p(m_k \mid o_i)\, p(m_j \mid o_i)\, p(o_i)}$$

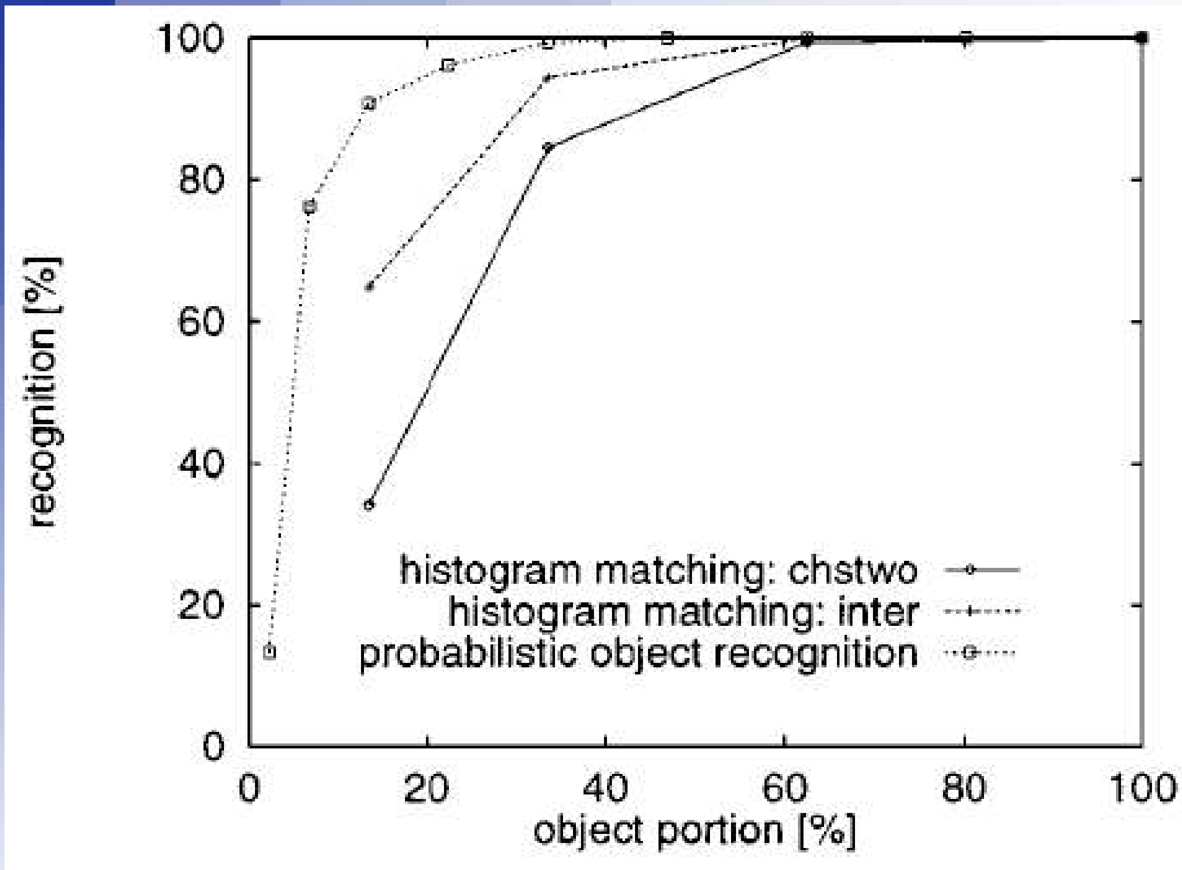
Bayes rule – more features

$$p(o_n \mid m_k \wedge m_j) = \frac{p(m_k \mid o_n)\, p(m_j \mid o_n)\, p(o_n)}{\sum_i p(m_k \mid o_i)\, p(m_j \mid o_i)\, p(o_i)}$$

- **What if we have more than two M-vectors (features)?**
  - **Still assuming independence :**

$$p\left(o_n \,\middle|\, \bigwedge_k m_k\right) = \frac{p\left(\bigwedge_k m_k \mid o_n\right) p(o_n)}{\sum_i p\left(\bigwedge_k m_k \mid o_i\right) p(o_i)} = \frac{\prod_k p(m_k \mid o_n)\, p(o_n)}{\sum_i \prod_k p(m_k \mid o_i)\, p(o_i)}$$

Results compared to "ordinary" histogram matching...

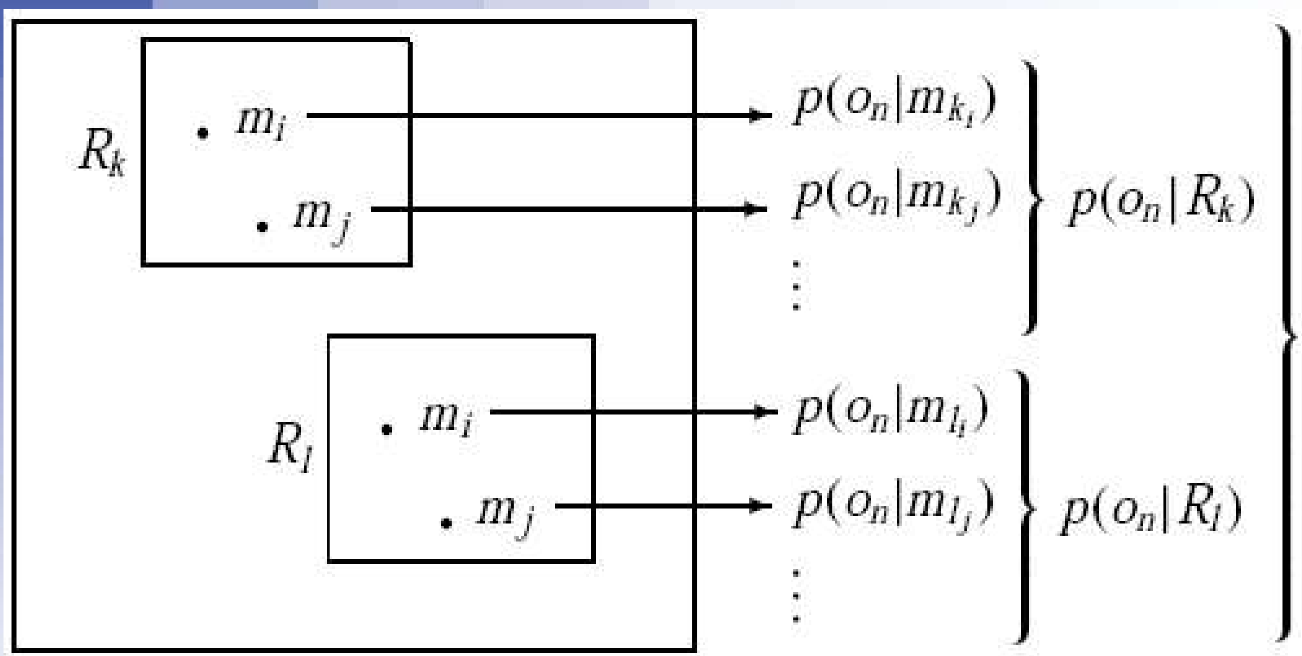


$$\chi_v^2(Q, V) = \sum_{\mathbf{i}} \frac{(q_{\mathbf{i}} - v_{\mathbf{i}})^2}{v_{\mathbf{i}}}$$

$$\cap(Q, V) = \sum_{\mathbf{i}} \min(q_{\mathbf{i}}, v_{\mathbf{i}})$$

For multiple object

- **Divide the input image and run the recognition locally in each region.**



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

Some results for multiple objects

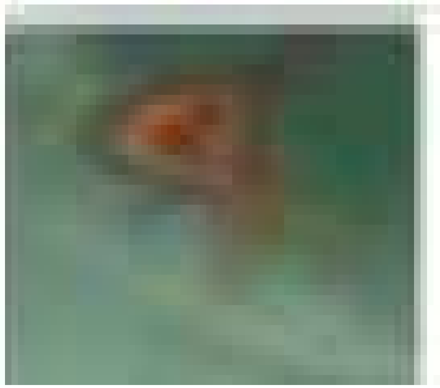
Other Subjects within Object Recognition

- Assumption we have (up to now)
  - No generalization, only recognize the object that is presented.
- Non rigid bodies, e.g. no animals have the same shape all the time -> they can move!
- Classes, e.g. the class of chairs, tables, cars etc.

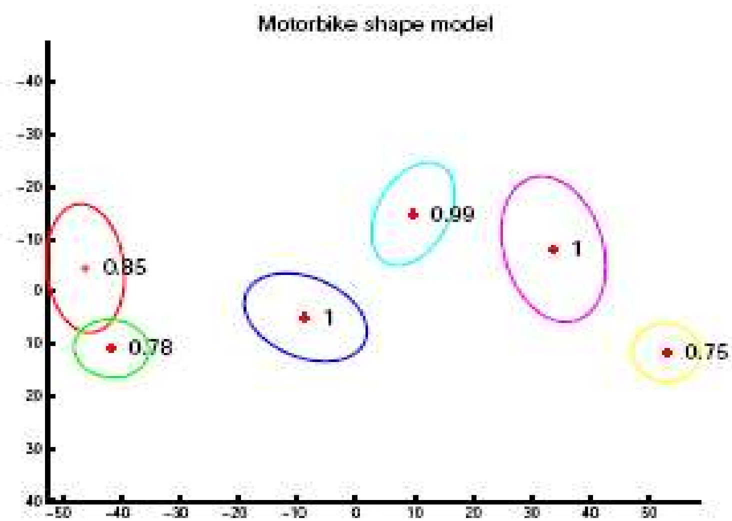
How does the features look like now? - Non rigid objects

- **Local feature can still be the same (*if no geometrical constrains were added*)**
- **Global feature that not contains geometrical information.**
- **Keyword : Encoded / not encoded geometrical information.**

Note the background helps a lot here...



How does the feature look like now?
- Classes of objects



Some publications used

- **D. Lowe,** *Distinctive Image Features from Scale-Invariant Key points*
- **S. Ekvall et al,** *Receptive Field Cooccurrence Histograms for Object Detection*.
- **P.Chang et al,** *Object Recognition with Color Coocurrence Histograms*.
- **G. Granlund,** *Unrestriced Recognition of 3-D Objects for Robotics Using Multi-Level Triplets Invariants*.
- **Fergus et al,** *Object Class Recognition by Unsupervised Scale-Invariant Learning*.
- **B. Schiele et al,** *Recognition without Correspondence using Multidimensional Receptive Field Histograms*.