Computer Vision – Lecture 7
Pattern recognition concepts

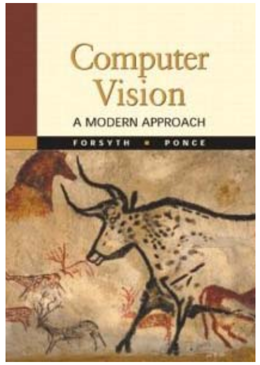
UCT2 – Information Technologies
MAP-I Doctoral Programme

Pedro Quelhas

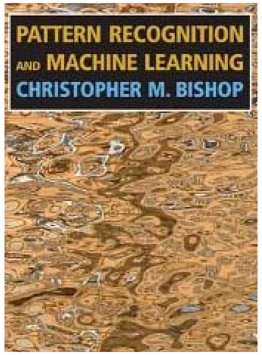
23 November 2010

References

- **Computer vision**
  - David A. Forsyth and Jean Ponce, “Computer Vision: A Modern Approach“ (chapters 22,23,24)



- **Machine learning/ Patter recognition**
  - Christopher M. Bishop , “Pattern Recognition and Machine Learning”



- **Lectures from previous years:**
  - http://www.dcc.fc.up.pt/~mcoimbra/lectures/mapi_0809.html

Outline

- **Pattern recognition in computer vision**
  - Understanding the visual world
  - Semantic gap
  - Dimensionality problem and need for features
  - An Example

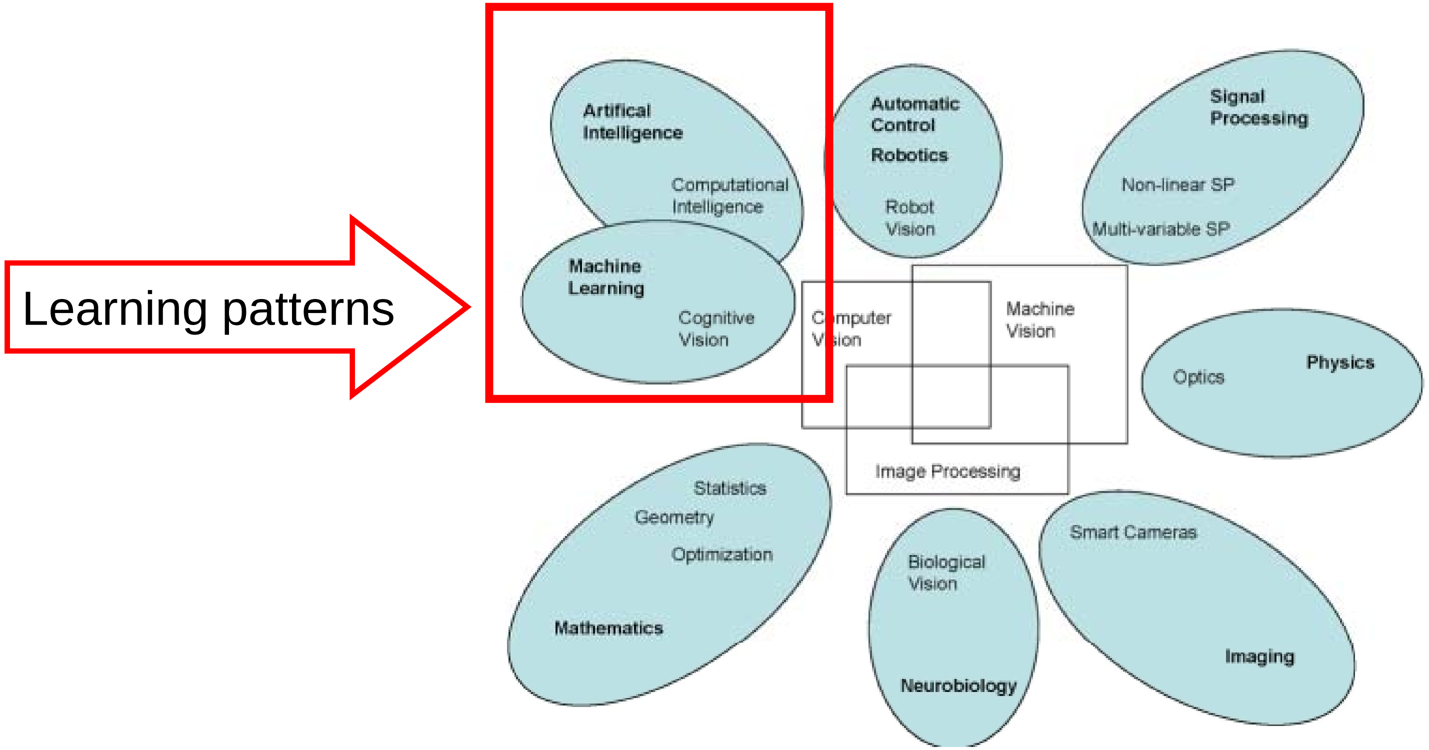
- **Feature extraction**
  - Range of features
    - Low/middle/high level
    - Global/Local
  - Mpeg-7 features

  - Image subdivision
    - The need for partial image analysis
    - Detection by segmentation
    - Grid and exhaustive search

- Automatic feature extraction
  - PCA

Pattern recognition in computer vision

- Multi-disciplinary computer vision:



Wikipedia

Pattern recognition in computer vision

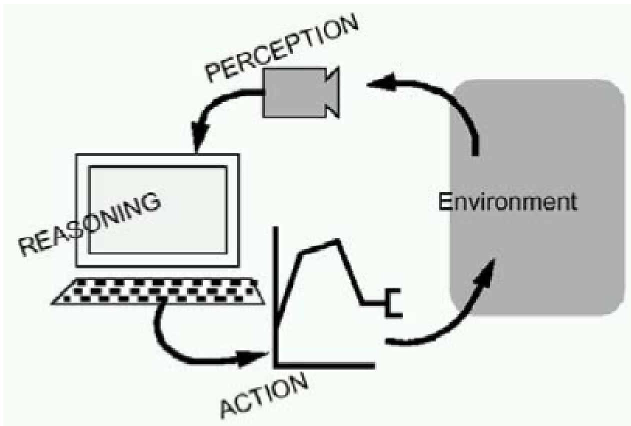
Goal of computer vision

Provide computers with human-like perception capabilities so that they can sense the environment, understand the sensed data, take appropriate actions (make decisions), learn from this experience in order to enhance future performance

- Understand the visual information with no accompanying structural, administrative or descriptive text information

Sources of difficulties:

- Sensory gap
- Semantic gap



Pattern recognition in computer vision

What is Pattern Recognition?

- The vision field has evolved from the application of classical image processing techniques to image understanding, model-based vision, knowledge-based vision, and systems that exhibit learning capability
  - Ability to reason
  - Ability to learn
- *Pattern recognition* is the study of how machines can
  - observe the environment
  - learn to distinguish patterns of interest
  - make sound and reasonable decisions about the categories of the patterns
- *Pattern recognition* on Wikipedia: "the act of taking in raw data and taking an action based on the category of the data"

Pattern recognition in computer vision

- What is an image?

135	229	212	232	151	173	103	206	197	191	180	27
203	12	1	42	179	173	143	204	124	150	213	165
111	42	110	212	104	97	184	63	211	150	202	61
239	28	25	204	220	48	152	113	253	92	44	23
212	7	66	37	114	178	240	66	106	3	24	252
219	130	29	142	157	119	83	168	132	11	25	190
234	194	43	190	146	14	39	250	108	41	70	139
159	131	198	87	95	242	54	68	120	110	59	108
118	59	141	186	74	153	31	233	141	90	9	200
207	149	3	85	215	68	155	21	236	252	195	207
29	62	152	103	31	208	203	33	213	35	11	160
212	125	204	101	83	190	91	136	221	88	116	81
72	159	53	241	156	210	127	192	122	6	82	77
240	62	143	103	195	103	184	247	[illegible]	195	253	13
254	145	247	[illegible]	108	[illegible]	[illegible]	[illegible]	223	23	249	154
154	194	63	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	18	10	57
131	71	117	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	147	182	219
154	39	178	47	[illegible]	150	42	83	202	37	16	192
101	40	239	6	252	170	33	4	174	233	195	67
53	145	23	231	234	234	185	180	197	175	245	171
209	75	99	164	204	242	192	242	108	18	45	220
207	131	226	144	114	182	23	230	18	250	169	214
99	110	47	71	125	108	194	72	248	69	197	5
175	160	249	252	34	189	81	20	117	170	175	205
240	13	168	194	78	125	12	60	147	251	97	136
180	131	27	81	153	104	40	92	95	22	104	79
125	83	79	70	24	151	189	212	133	77	117	32
234	2	48	32	6	198	58	38	248	46	212	20

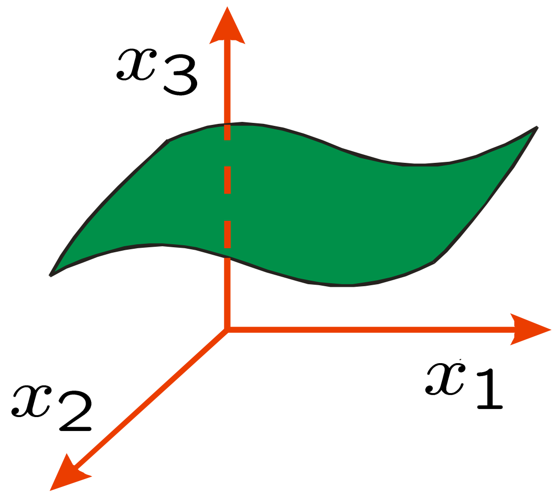
Apple?

Pattern recognition in computer vision

- The dimensionality problem:

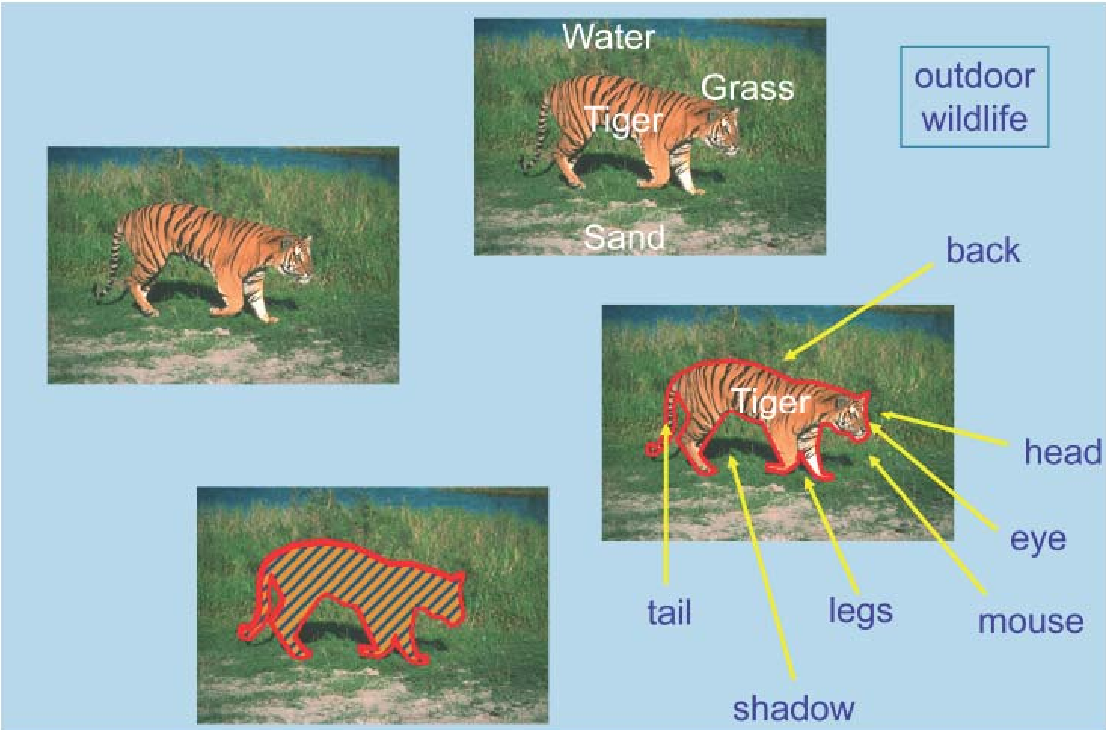
"An image is a point in a high dimensional space of pixel intensities"

- Object continuous transformations (rotations, translations etc) sweep out continuous manifolds
- Visual concept don't -> Highly non-linear
- Machine Learning Nightmare
- Feature extraction is needed
- Meaningful
- Invariant
- Compact



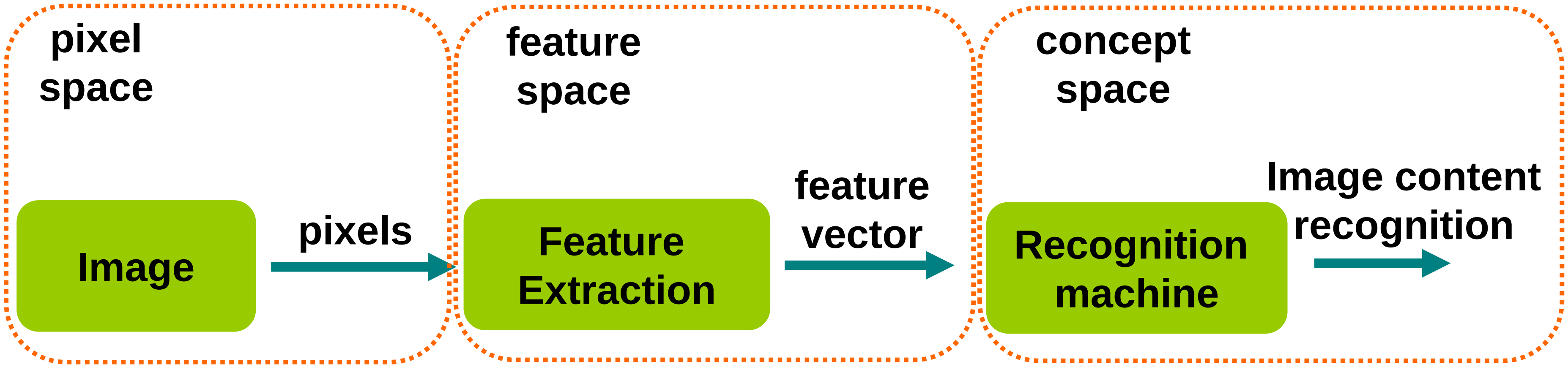
Pattern recognition in computer vision

- We need to analyze images in a more meaningful space than pixels:



Pattern recognition in computer vision

- Image recognition system:
  - **<u>Feature extraction:</u> captures meaningful information from the image (for the specific task at hand), reducing dimensionality.**
  - **<u>Pattern recognition:</u> Does the actual job of classifying or describing observations, relying on the extracted features.**
- System diagram:



PROBLEM: finding meaningfull features

Pattern recognition in computer vision

- An example:

  - Problem: Sorting incoming fish on a conveyor belt according to species
  - Assume that we have only two kinds of fish:
    - salmon
    - sea bass

Figure: Picture taken from a camera.

Pattern recognition in computer vision

- An example: The problem

Figure: What *we* see.

0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

Figure: What a *computer* sees.

Pattern recognition in computer vision

- An example: Decision problem
  - What kind of information can distinguish one species from the other?
    - length, width, weight, number and shape of fins, tail shape, etc.
  - What can cause problems during sensing?
    - lighting conditions, position of fish on the conveyor belt, camera noise, etc.
  - What are the steps in the process?
    - capture image $\longrightarrow$ isolate fish $\longrightarrow$ take measurements $\longrightarrow$ make decision

Pattern recognition in computer vision

- An example: Selecting features

  - Assume a fisherman told us that a sea bass is generally longer than a salmon.
  - We can use length as a *feature* and decide between sea bass and salmon according to a threshold on length.
  - How can we choose this threshold?

- In this case we have expert knowledge on how to solve the problem. This is almost never the case!

Pattern recognition in computer vision

- An example: Selecting features

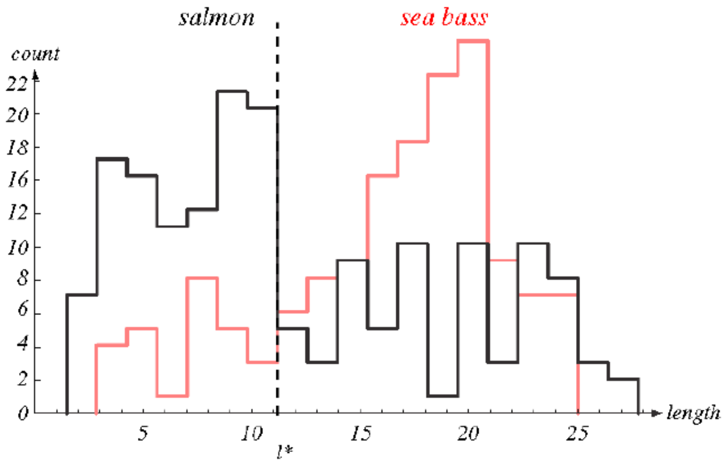


Figure: *Histogram* of the length feature for two types of fish in *training samples*. How can we choose the threshold l^* to make a reliable decision?

- Even though sea bass is longer than salmon on the average, there are many examples of fish where this observation does not hold.
- Try another feature: average lightness of the fish scales.

Pattern recognition in computer vision

- An example: Selecting features

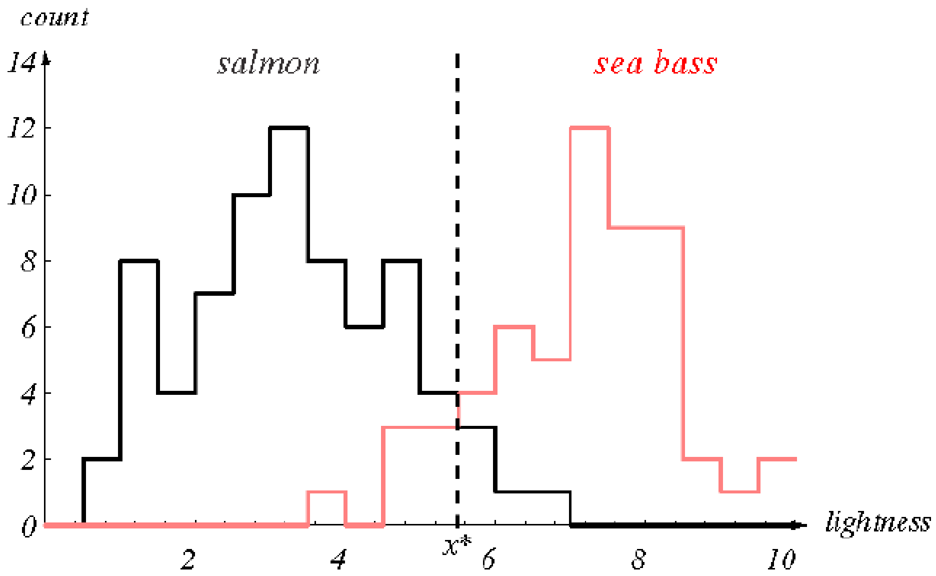


Figure: Histograms of the lightness feature for two types of fish in training samples. It looks easier to choose the threshold x^* but we still cannot make a perfect decision.

Pattern recognition in computer vision

- An example: Feature vector
  - We can use two features in our decision:
    - lightness: x_1
    - length: x_2
  - Each fish image is now represented as a point (feature vector)

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

in a two-dimensional *feature space*.

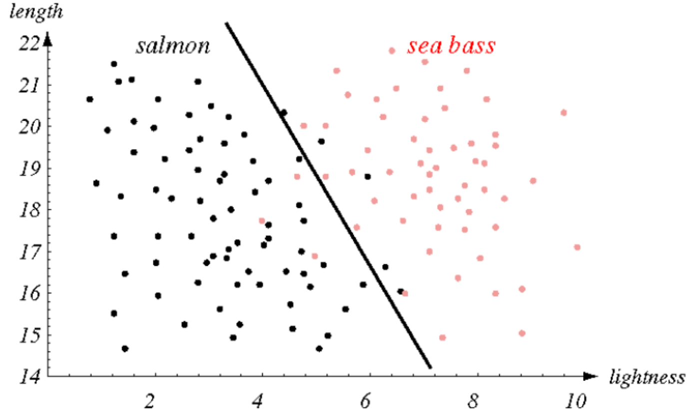


Figure: Scatter plot of lightness and length features for training samples. We can draw a *decision boundary* to divide the feature space into two regions.

Feature extraction

- Observers capture the meaning of an image discarding all unnecessary information.
- Different image content is described by different features:
  - shape
  - colour
  - texture
  - …
- Features can have local or global meaning, depending on the image content.
- By selecting specific features we are introducing prior knowledge on the problem.

Feature extraction

- Broad classification of features
  - Low-level
    - Color, texture, shape, motion, ...
  - Middle-level
    - Pedestrian in the image
    - Visible sky
    - Existence of trees
  - High-level
    - Car moving fast.
    - Person smiling

- Features can also be classified based on extent
  - Global/Local

Feature extraction

- Low level features
  - These features are very objective features
    - Color, texture, shape, motion, ...
- Middle level features
  - Features resulting from a decision process (related to the existence of some subjective details).
    - Segmentation of certain shapes
    - Occurrence of determined optical flow
    - Identification of certain objects, types of content
- High level features
  - Features with some semantic content information, highly contextual and based on prior knowledge.
    - Person A is talking to person B

Feature extraction

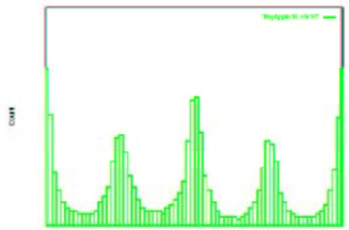
- From low-level to high-level:
  - While decisions must be made at each level we must always start from the low-level, as that is the information readily available to us.
  - The fundamental problem is how to reach high-level knowledge from initial low-level features.

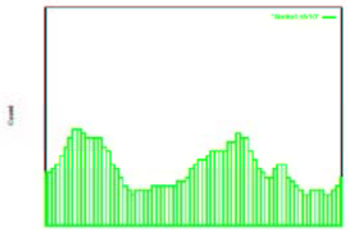
Feature extraction

- Global features:
  - These features highly summarize the image content enabling good description of global content or context but missing fine detail.
- Histograms

Colour

Edge



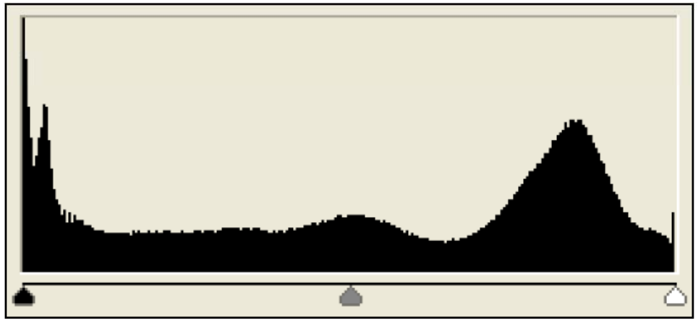


These can also be used at a semi-global level by subdividing the image into regions.

Feature extraction

- Grey-level histogram:
  - Intensity distribution (HSI).
  - We can define the number of histogram bins.
  - Histogram bins = Feature coefficients.

$$F = [f_0, \ldots, f_{255}]$$

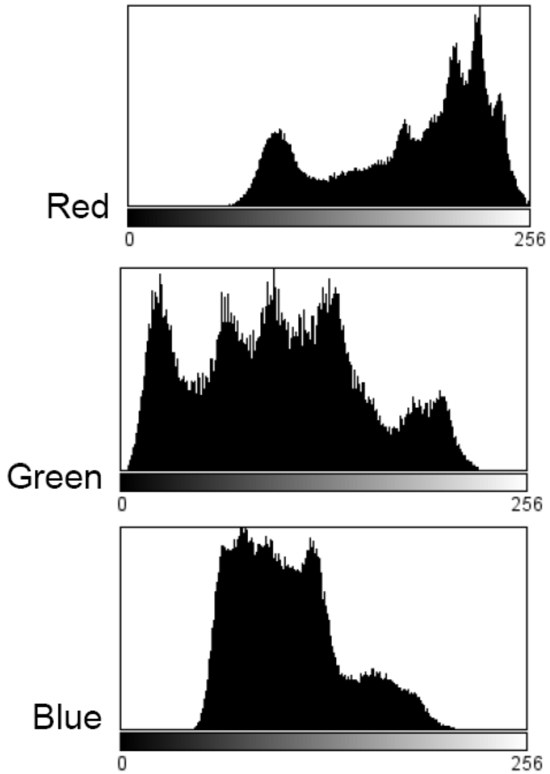


Feature extraction

- Colour histogram:
  - We typically have three histograms

    Ex: RGB Colour space

    - Red Histogram
    - Green Histogram
    - Blue Histogram
  - How do we build a feature vector?
    – Concatenate vectors.
    – Multi-dimensional quantization of colour space.

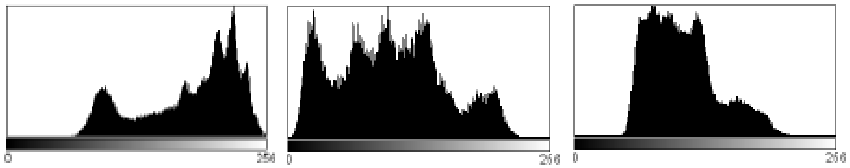


Feature extraction

- Colour histogram: simple concatenation? (simpler but ambiguous)

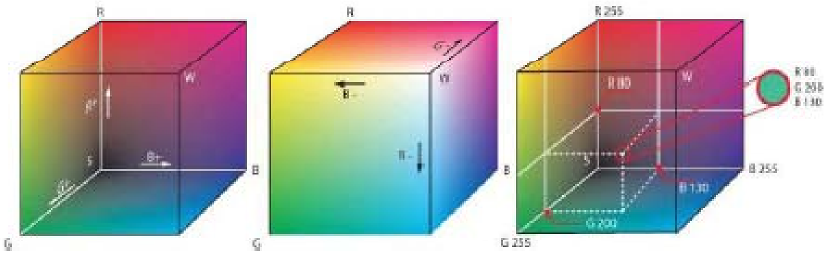
$$F_R = [f_{R0}, \dots, f_{R255}]$$
$$F_G = [f_{G0}, \dots, f_{G255}]$$
$$F_B = [f_{B0}, \dots, f_{B255}]$$

$$F_{RGB} = [F_R \mid F_G \mid F_B]$$



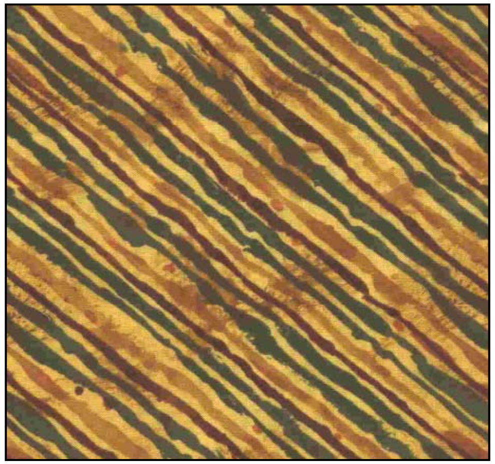
- Or retaining co-relation? RGB joint quantization (better but needs quatization to be meaningful)
  - Each coefficient is a small 'cube' inside the RGB cube.

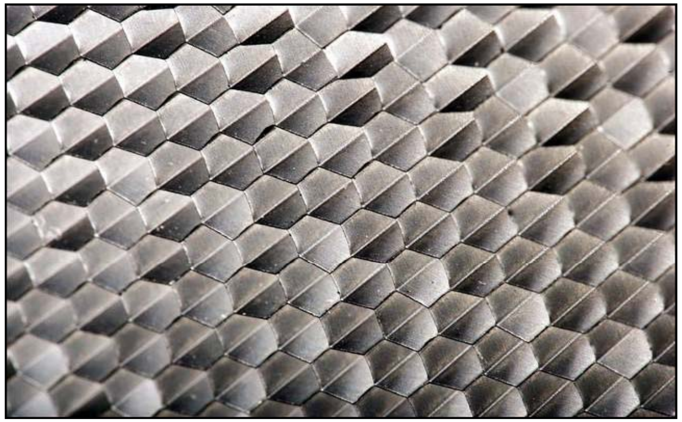
$$F = [f_0, \dots, f_N]$$



Feature extraction

- Classic features for image understanding: **Texture analysis**
  - Describing pixels' spatial relations in the image
    - Pixel's co-occurence matrices
    - Fractal descriptors
    - Markov random fields

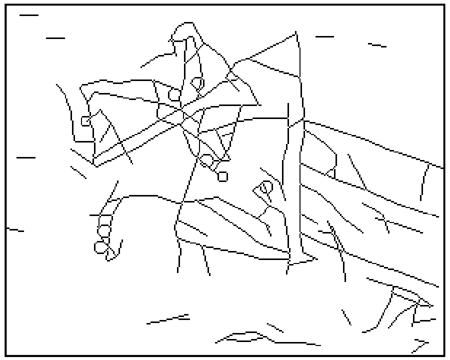


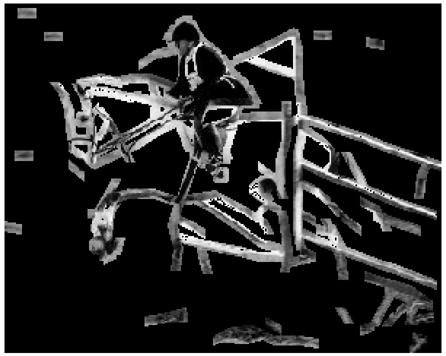


Feature extraction

- Classic features for image understanding: **Shape:**
  - **Edges**
  - **Ribons**

- **Example:** Primal sketch (David Marr)





  - **Very high computational cost**
  - **Features are very complex**
  - **Over simplification (colour and texture are important)**

Feature extraction

- MPEG-7 standard:
  - Developed by the Moving Pictures Expert Group: “is a standard for describing the multimedia content data that supports some degree of interpretation of the information meaning, which can be passed onto, or accessed by, a device or a computer code”
  - Provides a rich set of standardized tools to describe multimedia content.
    - Computer annotation.
    - Human annotation.
  - Audiovisual Description Tools
    - Descriptors
    - Descriptor Schemes
  - Target functionality:
    - Efficient search, filtering and browsing of multimedia content.
  - MPEG website:
    - http://www.chiariglione.org/mpeg

Feature extraction

- MPEG-7 dominant colour:
  - Clusters colors into a small number of representative colors (salient colors)
  - **F = { {c_i, p_i, v_i}, s}**
    - c_i : Representative colors
    - p_i : Their percentages in the region
    - v_i : Color variances
    - s : Spatial coherency
- MPEG-7 scalable colour:
  - HSI Histogram
  - Typical quantization: 256 bins.
    - 16 levels in H
    - 4 levels in S
    - 4 levels in I

$$F_{SC} = [f_0, \ldots, f_{255}]$$

  - Very popular for CBIR (Content-Based Image Retrieval).

Feature extraction

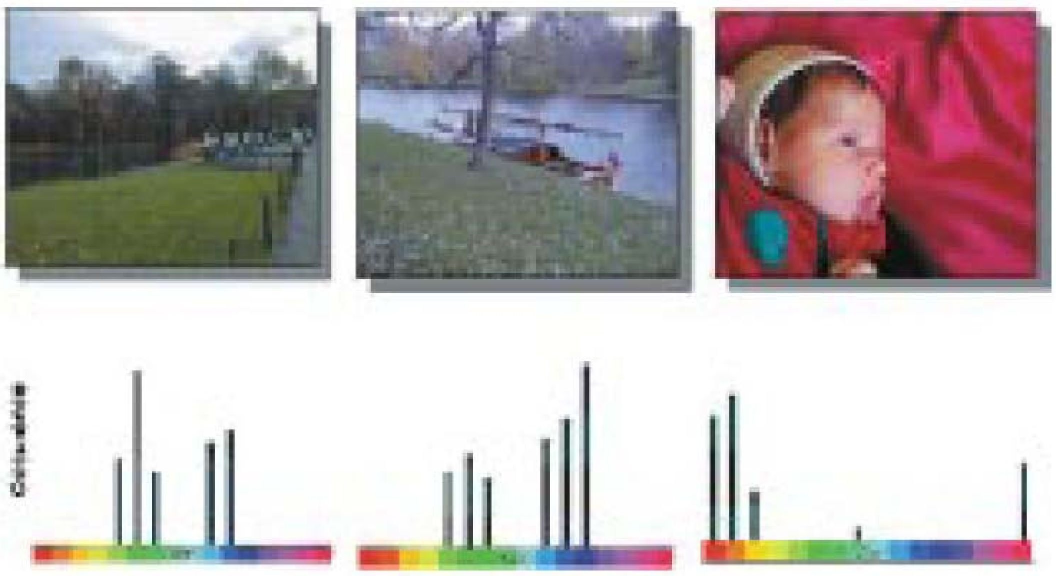


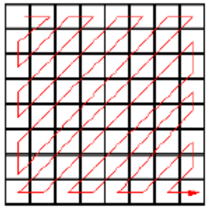
Fig. 2. Three color images and their MPEG-7 histogram color distribution, depicted using a simplified color histogram. Based on the color distribution, the two left images would be recognized as more similar compared to the one on the right.

[Sikora 2001]

Feature extraction

- MPEG-7 colour layout:

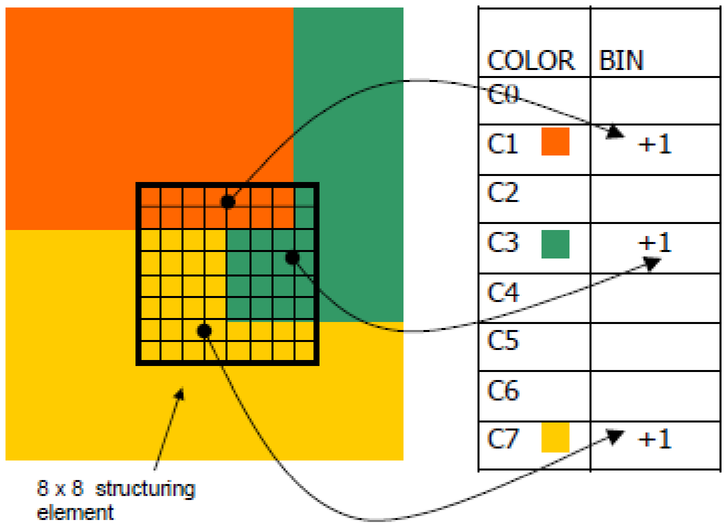
  - Clusters the image into 64 (8x8) blocks
  - Derives the average color of each block (or using DCD)
  - Applies (8x8)DCT and encoding



  - Efficient for
    – Sketch-based image retrieval
    – Content Filtering using image indexing

Feature extraction

- MPEG-7 colour structure:
  - Scanns the image by an 8x8 struct. element
  - Counts the number of blocks containing each color
  - Generates a color histogram (HMMD/4CSQ operating points)



Feature extraction

- MPEG-7 homogeneous structure:
  - Filters the image with a set of orientation and scale sensitive filters.
  - Computes mean and standard deviation of response.
  - 30 channels
    - 6 in angular direction, 5 in radial direction.

$$F_{HT} = [f_{DC}, f_{SC}, e_1, e_2, \ldots, e_{30}, d_1, d_2, \ldots, d_{30}]$$

fDC, *fSC* are the mean intensity and the standard deviation of image texture), where *ex* and *dx* are the logarithmically scaled texture energy and texture energy deviation coefficients.

Feature extraction

- MPEG-7 homogeneous structure:

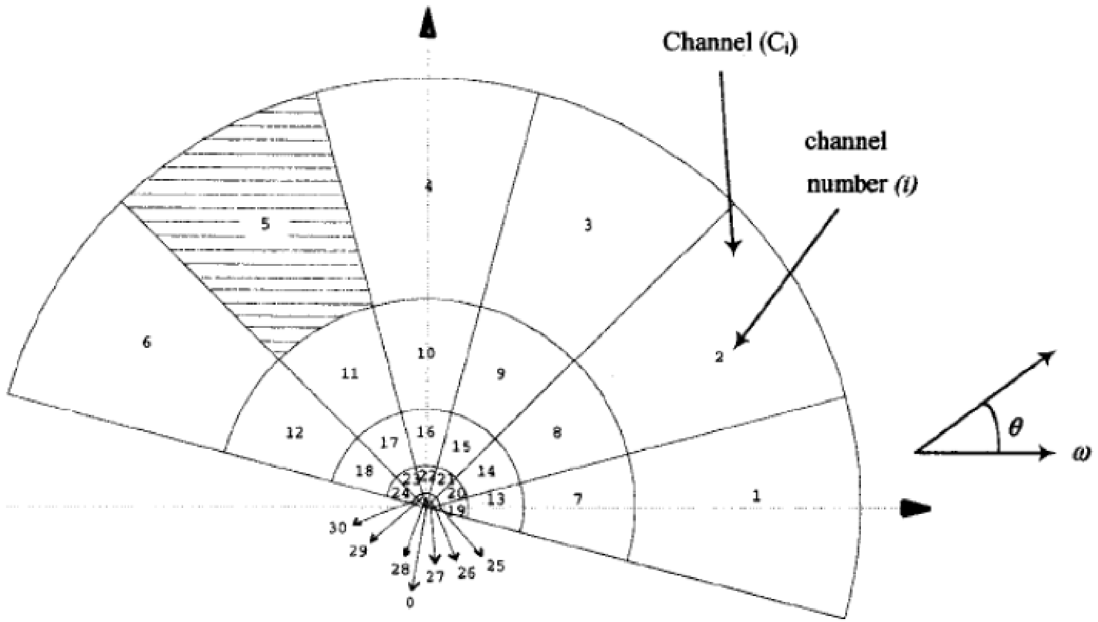
- Filters the image with a set of orientation and scale sensitive filters.
- Computes mean and standard deviation of response.
- 30 channels
  – 6 in angular direction, 5 in radial direction.

$$F_{HT} = [f_{DC}, f_{SC}, e_1, e_2, \ldots, e_{30}, d_1, d_2, \ldots, d_{30}]$$

fDC, *fSC* are the mean intensity and the standard deviation of image texture), where
ex and *dx* are the logarithmically scaled texture energy and texture energy deviation coefficients.

Feature extraction

- MPEG-7 homogeneous structure channels:



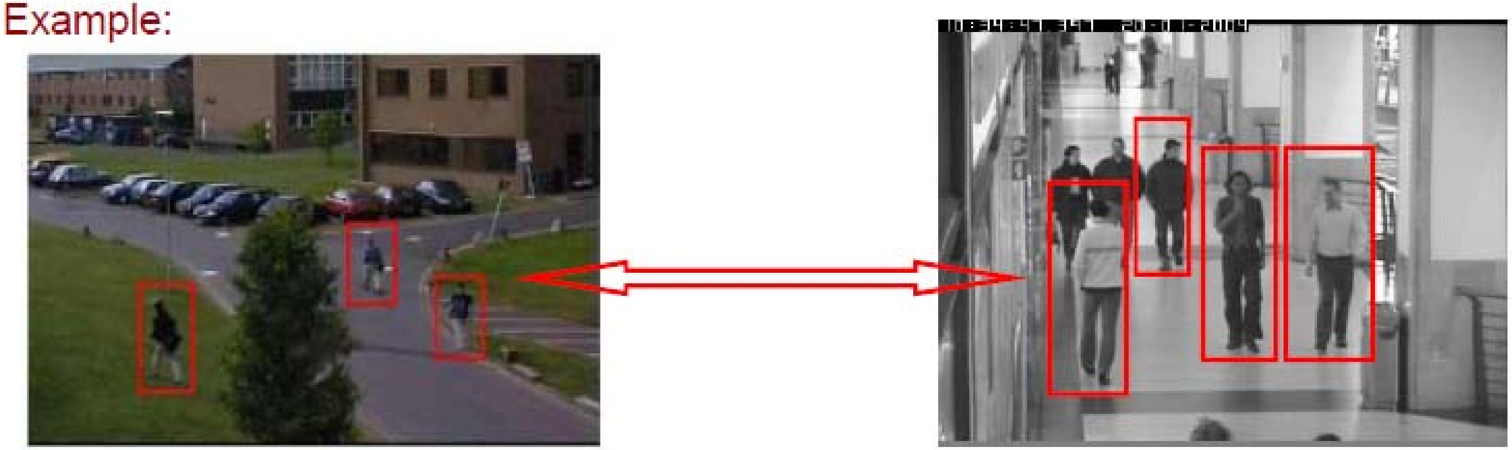
Feature extraction

- MPEG-7 Local edge histogram:
  - Image divided into 4x4 sub-regions.
  - Edge histogram computer for each sub-region.
  - Five bins:
    - Vertical, horizontal, 45 diagonal, 135 diagonal, and isotropic.
  - 80 total bins.

$$F_{LEH} = [f_0, \ldots, f_{79}]$$

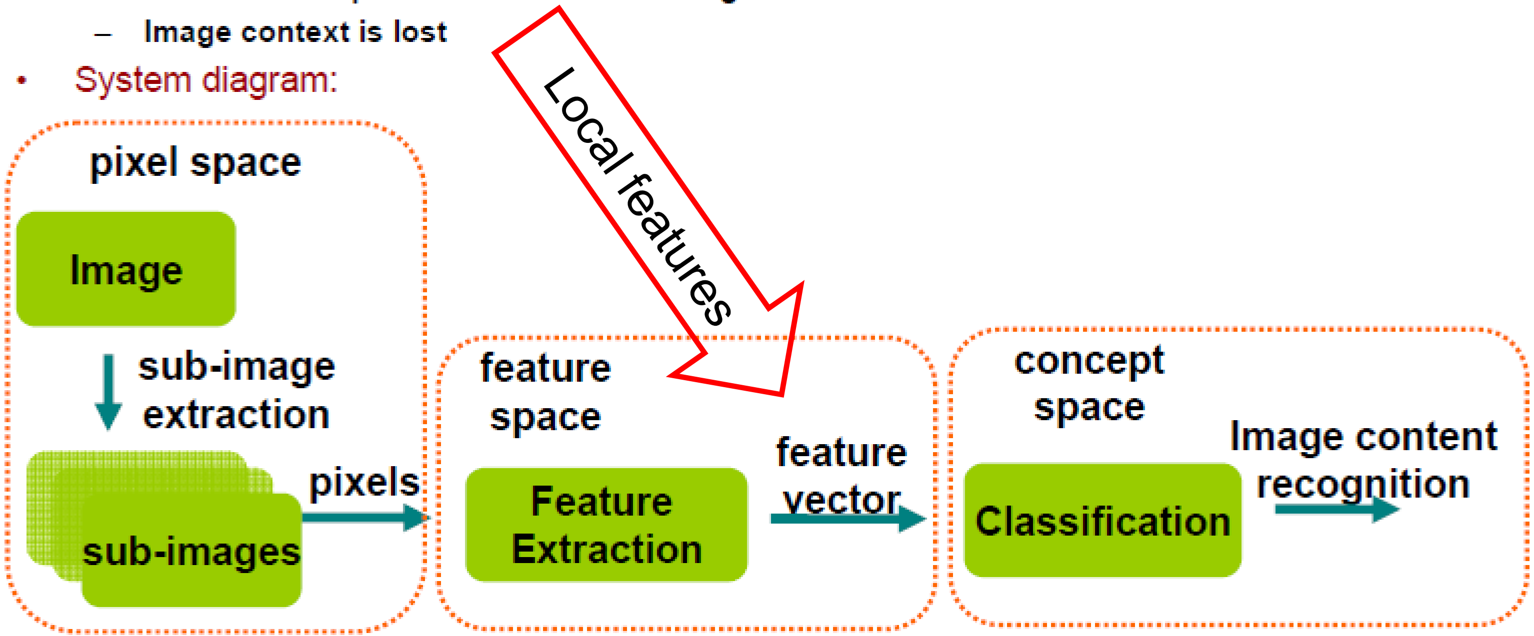
Feature extraction

- Image subdivision:
  - An image often requires a part based analysis:
    - Context is global, but object are defined locally.
    - Most image content is described at a local level.
    - By dividing an image into parts we simplify recognition.
    - Separating objects from context makes recognition more robust
  - Example:



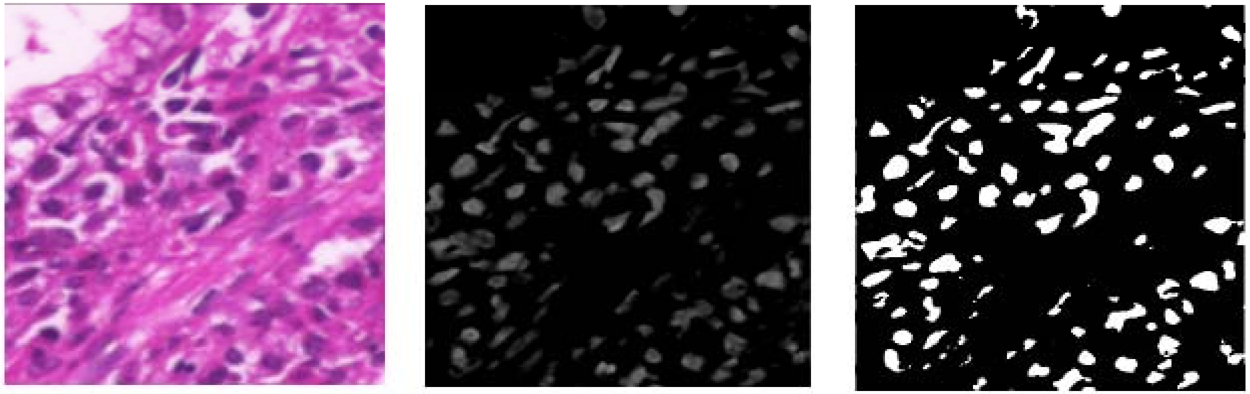
Feature extraction

- Image subdivision:
  - Sub-image integration:
    - Feature extraction is now performed at the sub-image level.
    - Classification performed at the sub-image level.
    - Image context is lost
  - System diagram:



Feature extraction

- Image subdivision: using segmentation
  - By removing unwanted background we can obtain each individual object in the image for recognition (object segmentation).
    - **Only possible when it is easy to separate between object and background**
    - **Feasible when dealing with video data (motion segmentation, tracking)**
  - Cell image segmentation Classification

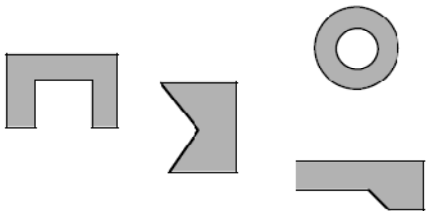


Feature extraction

- Morphological features:

For industrial recognition tasks it is often required to distinguish
- a small number of different shapes
- viewed from a small number of different view points
- with a small computational effort.

In such cases simple 2D shape features may be useful, such as:
- area
- boxing rectangle
- boundary length
- compactness
- second-order momentums
- polar signature
- templates



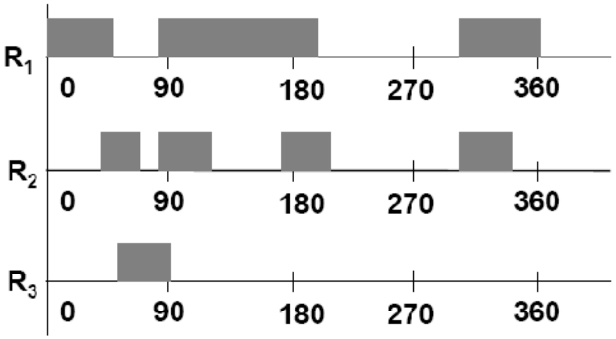
Features may or may not have invariance properties:
- 2D translation invariance
- 2D rotation invariance
- scale invariance

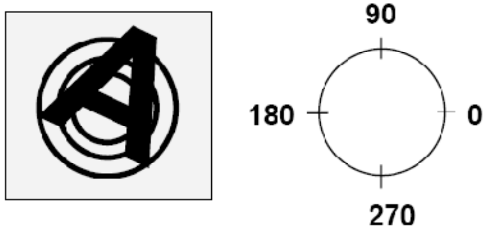
Feature extraction

- Shape features for object recognition:

Polar signature

The polar signature records the angular segments where circles around the center of gravity lie within a shape.





- scalable from coarse to fine by appropriate number of circles
- radii of circles must be chosen judiciously
- translation-invariant
- rotation-invariance can be achieved by cyclic shifting

Image composition

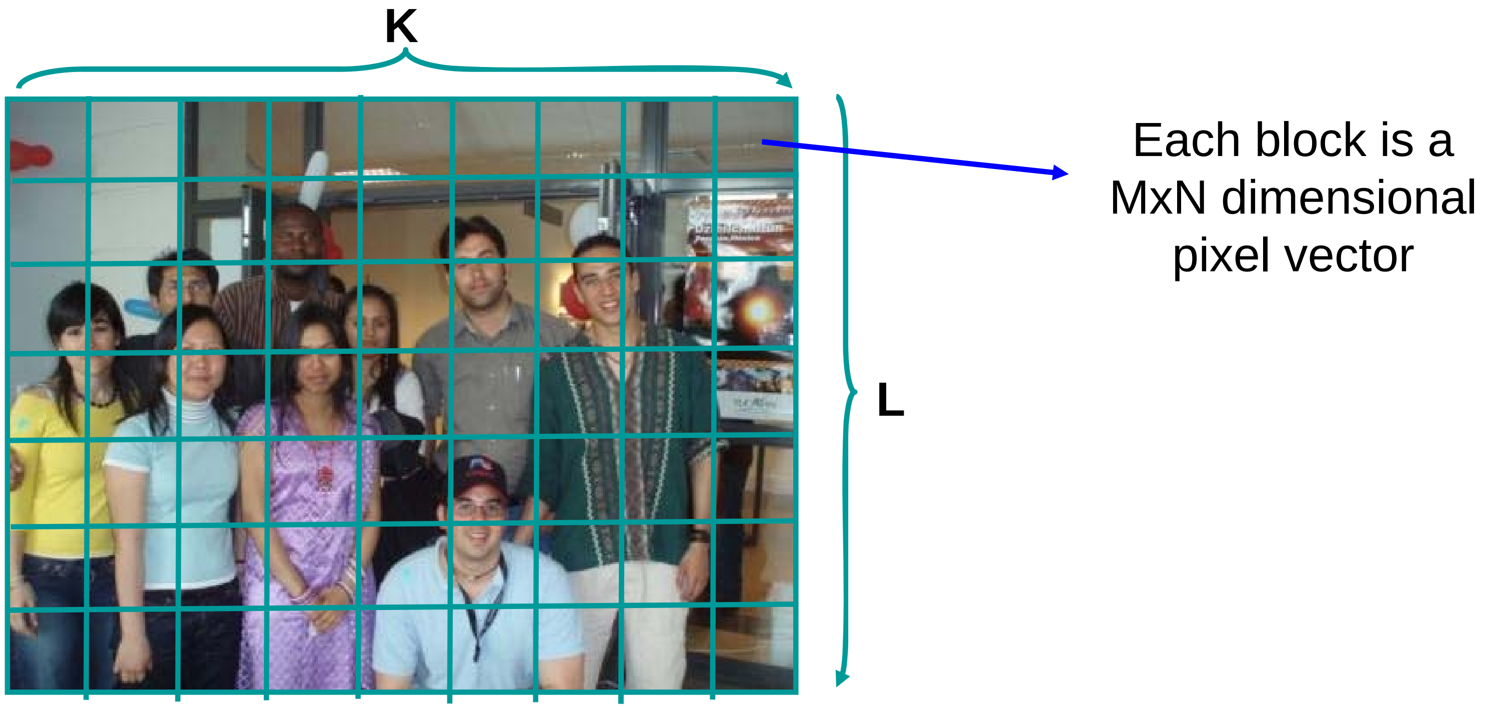
- Objects may not be segmentable (equally difficult problem to recognition)
  - Most image content is not easily segmentable (optical flow sometimes helps).

- Recognition becomes related to detection, as need to perform both to fully understand the image's content.

- Different approaches for image sub-division
  - **Exhaustive search**
  - **Grid sub-division**
  - **Over sampled grid subdivision**
  - **Local interest point descriptors (lecture 9)**

Image subdivision

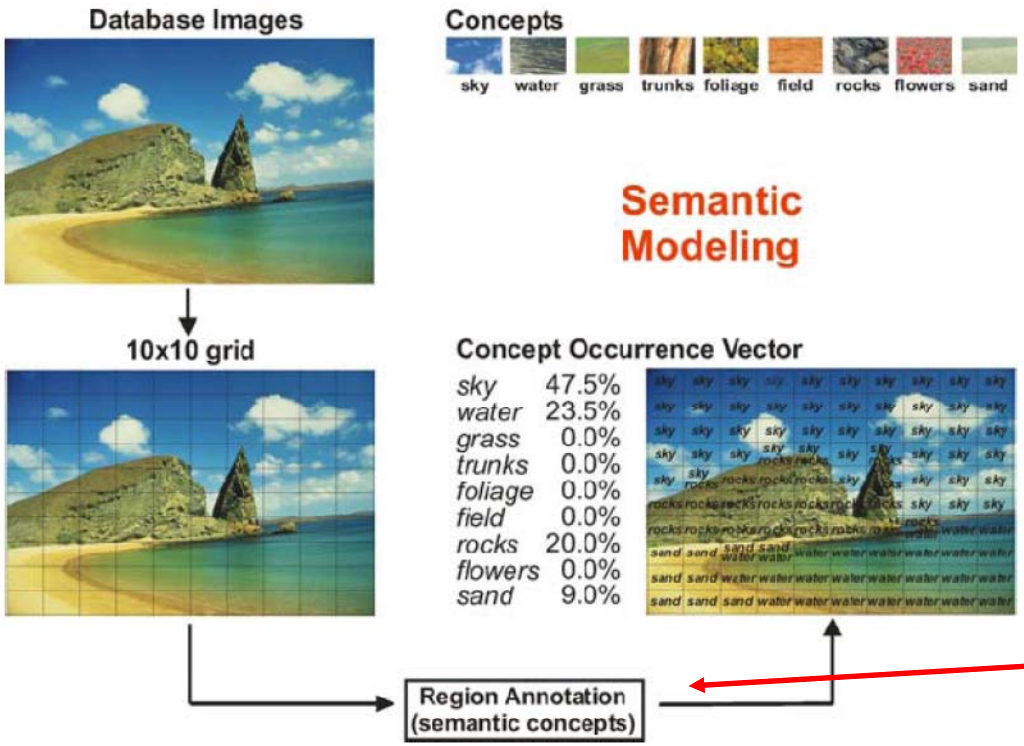
- Image subdivision
  – **exhaustive grid division: the whole image is divided into blocks with no overlap.**



  – **Sampling problem:** may not recognize objects in the image which are split over several image blocks

Image subdivision

- If we assume repeatability, regular sampling is enough:



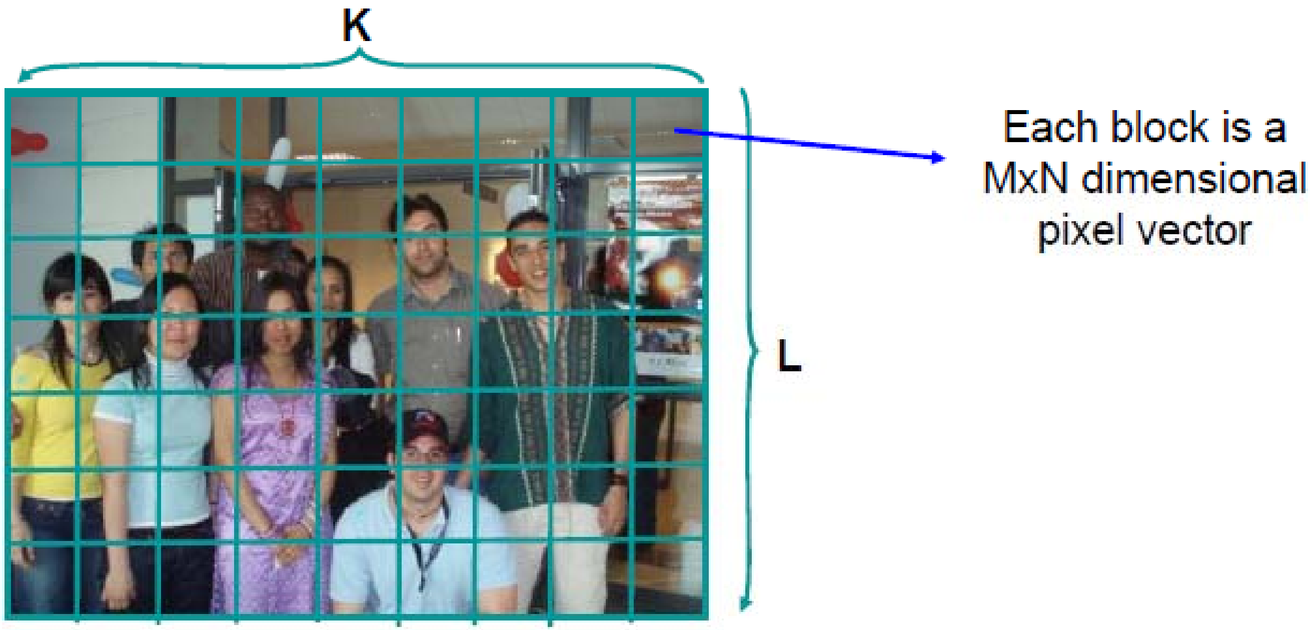
Concepts modeled by color and texture features as in the global case. However, we can extract more information per pixel as the area under analysis is smaller

Classification for each image subdivision obtained by SVM classifier

  – Semantic Scene Modeling and Retrieval for Content-Based Image **Retrieval. Julia** Vogel and Bernt Schiele. *International Journal of Computer Vision*. Vol. 72, No. 2, pp. 133-157, April 2007.

Feature extraction

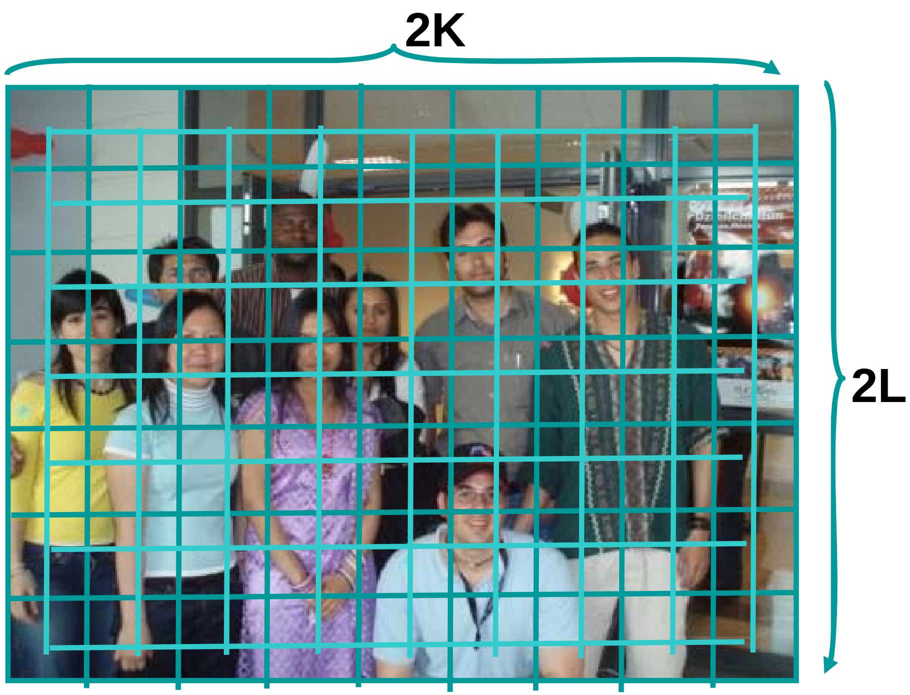
- Image subdivision: search
  - **exhaustive grid division: the whole image is divided into blocks with no overlap.**



  - Sampling problem: may not recognize objects in the image which are split over several image blocks

Feature extraction

- Image subdivision
  - over-sampling grid division: the whole image is divided into blocks with overlap.



  - Redundant, but less prone to miss objects.

A.Bosch, A.Zisserman, X.Muñoz. *Scene Classification via PLSA.* European Conference on Computer Vision, vol. IV, pp. 517-530. Graz, Austria. May 2006.

Image subdivision

– **Scanning image division: the image is scanned with a fine regular sampling into block (very redundant).**

– **Similar to a grid division. However, it is more exhaustive.**
– **To detect object at several scales several passes have to be made with variable window size (same applies to rotation)**

Template matching

- Having defined a sub-image area, the most straightforward way to recognize image content is to cross-correlate the pixels in the area with a model template:

$$S_i = \frac{\sum_{x,y}[f(x,y)-\bar{f}_{u,v}][t(x-u,y-v)-\bar{t}]}{\left\{\sum_{x,y}[f(x,y)-\bar{f}_{u,v}]^2 \sum_{x,y}[t(x-u,y-v)-\bar{t}]^2\right\}^{0.5}}$$

- Best match wins

$$\arg\max_i S_i = \mathrm{I}_i^{\mathrm{T}}\mathrm{I}$$

- Sensitive to noise
- Computationally expensive, i.e. requires presented image to be correlated with every image in the database ! (no generalization power)

Automatic Feature Extraction

- Principle component analysis
  - Principal component analysis computes the most meaningful basis to re-express a noisy, garbled data set.
  - The hope is that this new basis will filter out the noise and reveal hidden dynamics.
  - By using PCA on the pixels of several image patches that represents the same content (faces for example) we assume that we can create a basis (meaningful feature dimensions) for the representation of such data.
  - Each of the new basis axis has an associated value which indicates that axis importance for data representation.
  - By retaining the most important axis (90% of energy or more) we isolate the most important features to represent our data (less than 10% dimensionality for adequately chosen training patches).
  - Keeping only a few axis leads to a great dimensionality reduction and in that new low dimensional, meaningful, feature space it is possible to obtain a feature vector from input image pixels.

Automatic Feature Extraction

- Principle component analysis
  - Starting point: Image patch
    - *t*-dimensional feature space of all pixels in the image patch.
  - Objective: data driven template construction
    - *f*-dimensional feature subspace
    - *W* linear transformation matrix that maps the original t-dimensional space onto the smaller f-dimensional space
    - Obtained subspace is a meaningful basis to represent the data
    - Reduces the dimension of the raw signal, while maintaining the “important” information (signal energy is mostly maintained)
  - Problems:
    - Projection may suppress important detail.
    - Method does not take discriminative task into account.

Automatic Feature Extraction

- **PCA algorithm:**
  - Obtain input data vectors and compute mean input vector
  - Subtract mean input vector to all data vectors
  - Compute input data covariance matrix
  - Compute eigenvectors and eigenvalues from the covariance matrix
  - Choosing the components and forming the feature vector
    - removing components which correspond to the small eigen vectors

- **Computation:**

SVD for PCA

- SVD can be used to efficiently compute the image basis

$$PP^t = (U\Sigma V^t)(U\Sigma V^t)^t = U\Sigma V^t V \Sigma^t U^t = U\Sigma^t \Sigma U^t = U\Sigma^2 U^{-1}$$

$$(PP^t)U = U\Sigma^2$$

$$(PP^t)v = v\lambda$$

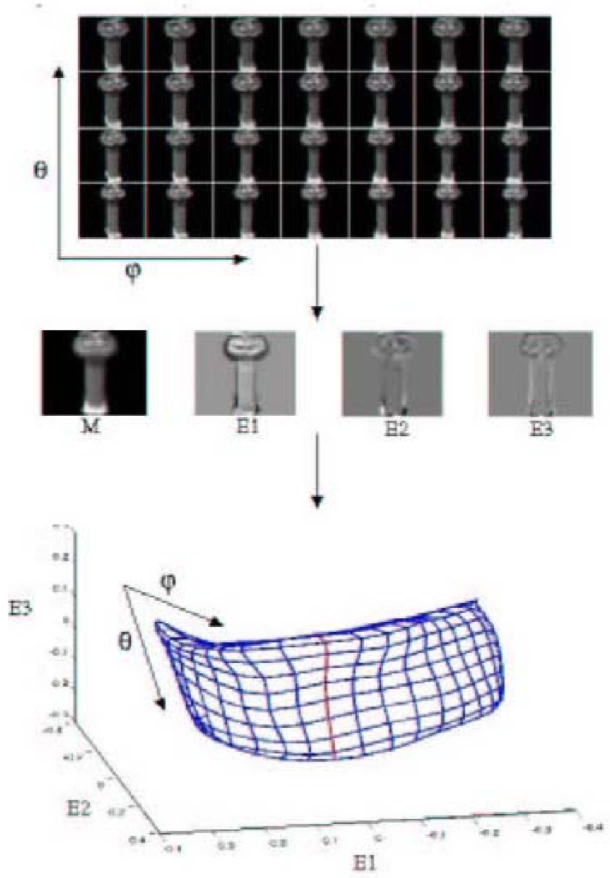
- U are the eigen vectors (image basis)

- Most important thing to notice: Distance in the eigen-space is an approximation to the correlation in the original space

$$\|x_i - x_j\| \approx \|c_i - c_j\|$$

Automatic Feature Extraction

- Meaningful image representation basis:
  - Starting from selected patches where the same content exists but under specific transformations we obtain a basis where each basis axis is strongly correlated to each transformation.
  - It is important to observe that the initial patches in pixel space had no meaningful basis for recognition or comparison.



Automatic Feature Extraction

- Meaningful image representation basis:
  - PCA is widely used for face recognition where the resulting basis allow for the characterization of specific changes in appearance between faces.
  - The main weakness of such methodology is that it requires almost perfect positioning of the face both for training patches (somewhat easy) and for test patches (very difficult due to detection errors).

average face