

## EBU706U - Multimedia Systems

### Introduction

Frame- vs Object- based multimedia  
“Semantic”

Dr. Yi-Zhe Song

EBU706U



### Objectives

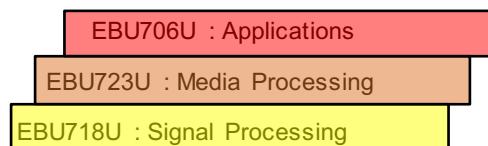
- To provide an understanding of the **types of data** that make up multimedia systems - focusing on audio, 2D - 3D images and video
- To introduce state-of-the-art **coding** and compression techniques used to store and transmit media over networks
- To discuss important topics related to the **creation** of digital media archives and portals
- To discuss important aspects of **advanced** 3D multimedia systems, augmented and mixed reality
- To introduce techniques for **copyright protection** and authentication of digital media content

EBU706U



### What is EBU706U about?

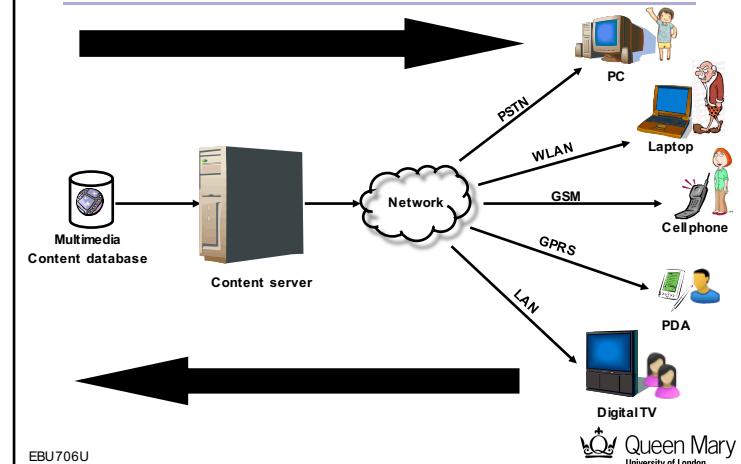
Complementary with EBU723U (Image and Video Processing) and EBU718U (Advanced Transforms)



EBU706U



### Scenario



EBU706U

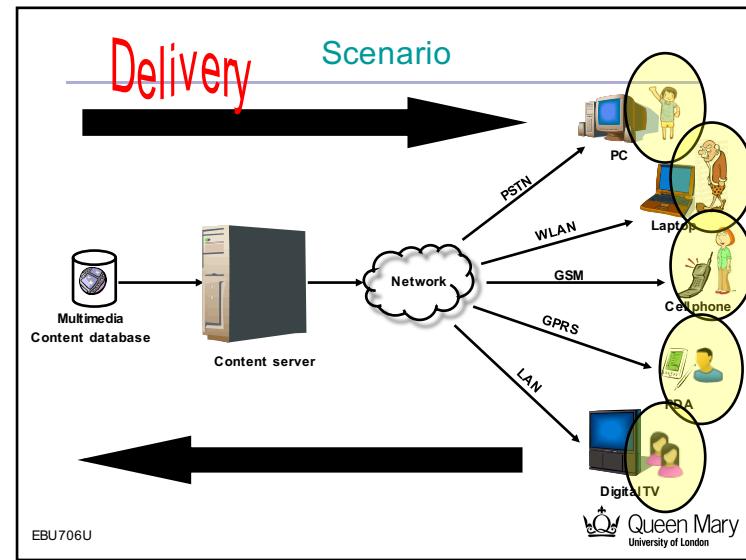
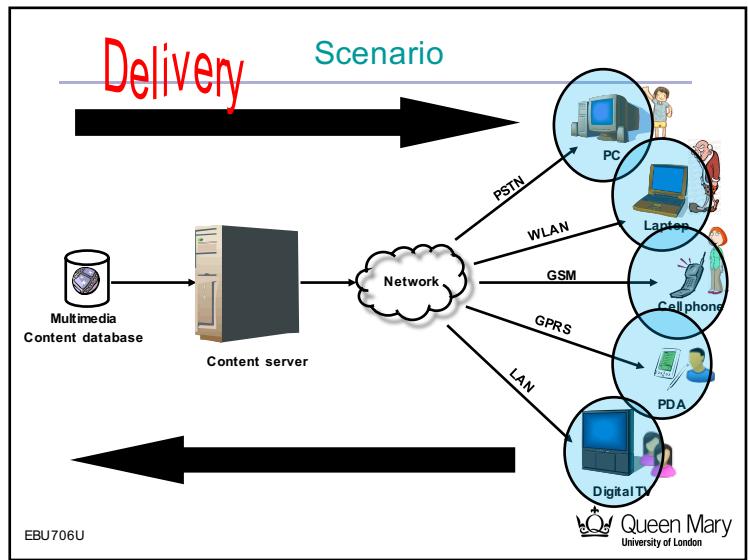
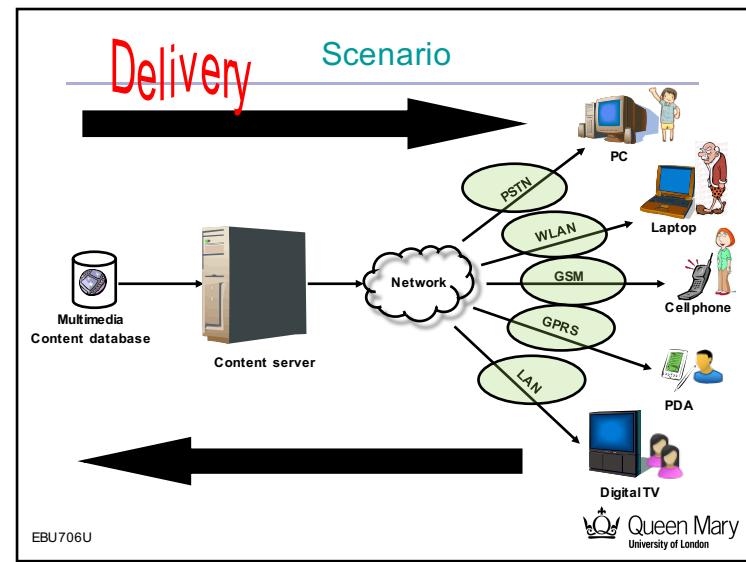
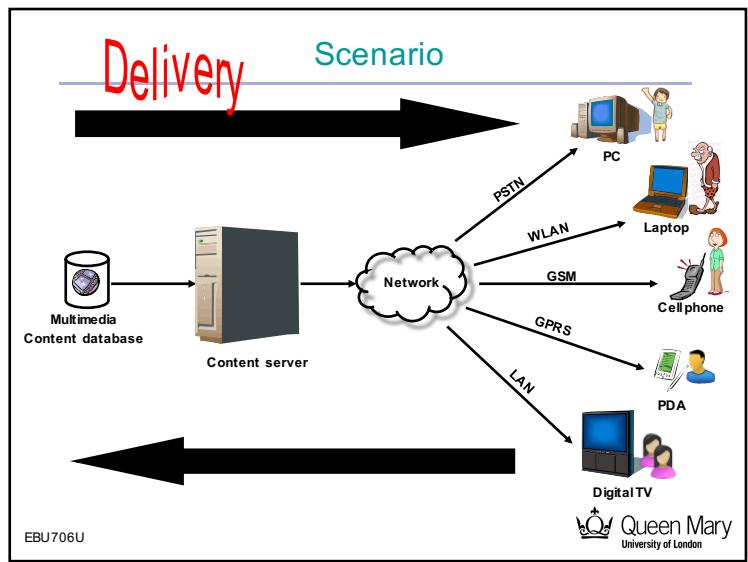


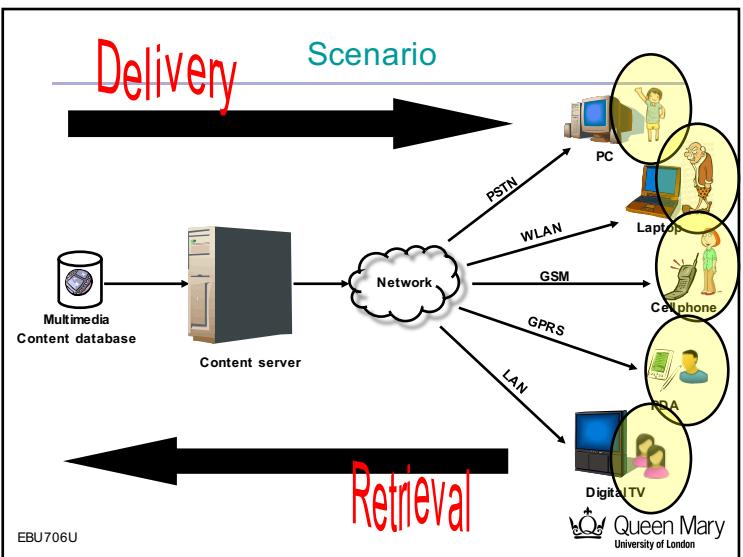
self driving car:  
pedestrian detection; car detection; traffic sign detection

criterion of compression:  
“cheating”, human perception

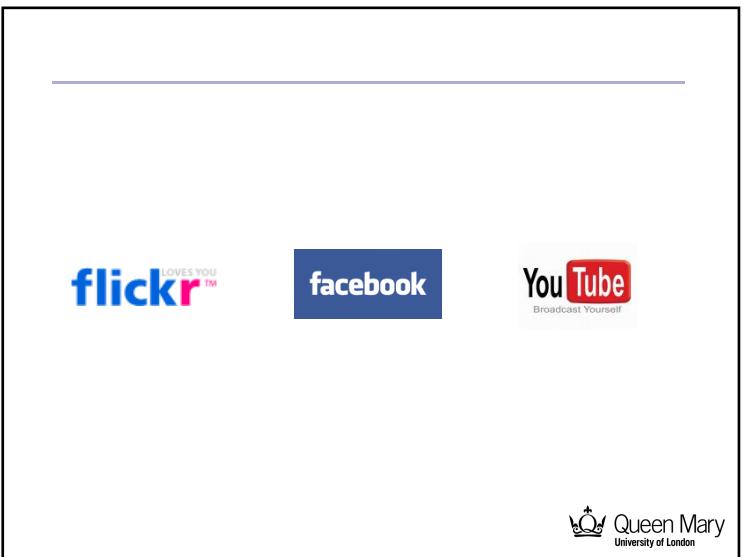
data vs. information: SEMANTIC

jpeg not suitable for segmentation, because of artefacts





- Current scenario**
- **Convergence** of telecommunications, IT and media
    - opportunities for businesses and consumers
    - creates new ways
      - of doing business
      - of entertaining and informing
      - for individual users to manage and control vast amounts of information from many diverse sources
    - blurring of boundaries
      - between business and consumer markets
      - between work and home
  - Very large and new **personalised** market that cannot be satisfied using existing concepts of service provision
- EBU706U
- Queen Mary University of London



- Design of multimedia applications**
- Need to understand
    - tasks
    - users
    - context of use
  - Design requires the following inputs
    - technical
    - creative
    - human factors
    - user experience
  - Currently → what can we do with the technology?
- EBU706U
- Queen Mary University of London

## Challenges

- **What?** New and innovative methods for
  - modelling
  - processing
  - mining
  - organising
  - indexing
- **Why?** For effective and efficient
  - searching and retrieval
  - delivery
  - management
  - sharing of multimedia content

EBU706U



## Motivation for EBU706U

- **Multimedia Systems** have grown over the last years to a massive industry involving
  - image, video and audio processing
  - information retrieval
  - integration, generation and manipulation of different media
  - media synthesis
- **Multimedia Systems (EBU706U)** will provide
  - an understanding of these media, their creation, integration and processing
  - an overview on how state-of-the art multimedia systems are built and work

EBU706U



## Video production



EBU706U



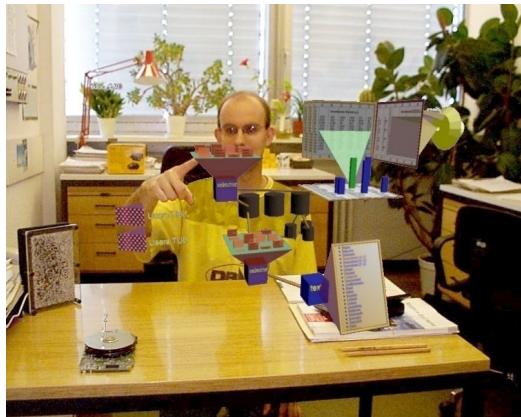
## Virtual presence



EBU706U



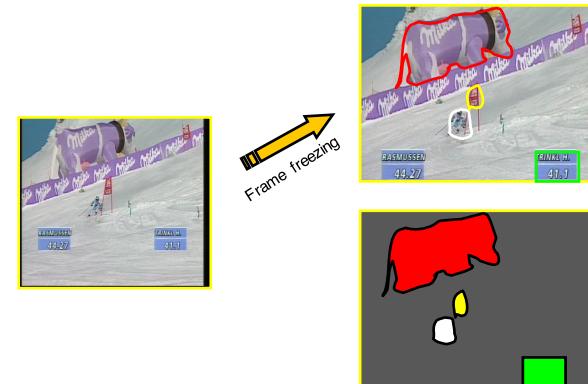
## Augmented reality



EBU706U



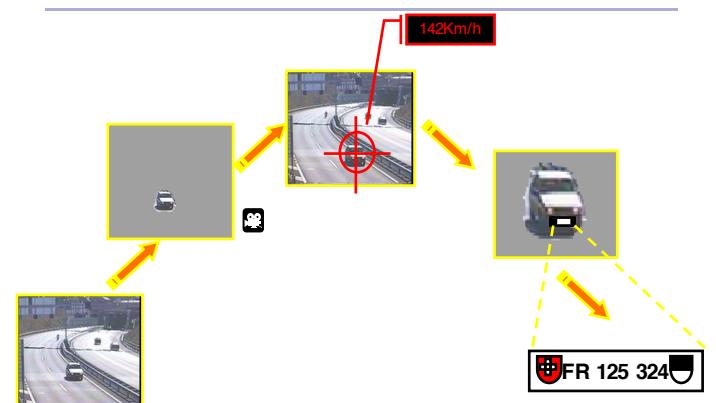
## Hyper video



EBU706U



## Traffic monitoring



EBU706U



## Portals, archiving, indexing, retrieval

## Lecture plan (1)

- Week 1:
  - Introduction to the course
  - Frame-based multimedia (reminder) : JPEG / MPEG1&2
  - Object-based multimedia : MPEG4
  - Segmentation
- Week 2:
  - Multimedia delivery
  - Multimedia Content Annotation (MPEG7)

EBU706U



## Lecture plan (1)

- Week 1:
  - Introduction to the course
  - Frame-based multimedia (reminder) : JPEG / MPEG1&2
  - Object-based multimedia : MPEG4
  - Segmentation
- Week 2:
  - Multimedia delivery
  - Multimedia Content Annotation (MPEG7)

EBU706U



## JPEG: 1% Compression



EBU706U



## 80% Compression



EBU706U



## 99% Compression



EBU706U



## Object-based Multimedia



EBU706U



## Lecture plan (1)

- Week 1:
  - Introduction to the course
  - Frame-based multimedia (reminder) : JPEG / MPEG1&2
  - Object-based multimedia : [MPEG4](#)
  - Segmentation
- Week 2:
  - Multimedia delivery
  - Multimedia Content Annotation (MPEG7)

EBU706U



## Lecture plan (1)

- Week 1:
  - Introduction to the course
  - Frame-based multimedia (reminder) : JPEG / MPEG1&2
  - Object-based multimedia : [MPEG4](#)
  - Segmentation
- Week 2:
  - Multimedia delivery
  - Multimedia Content Annotation (MPEG7)

EBU706U



## Region or object segmentation

Two basic concepts

Regions



Homogeneous according to given criteria  
(colour, motion, texture...)

Objects



Semantically meaningful:  
selection depends on the application

EBU706U



## Lecture plan (1)

- Week 1:

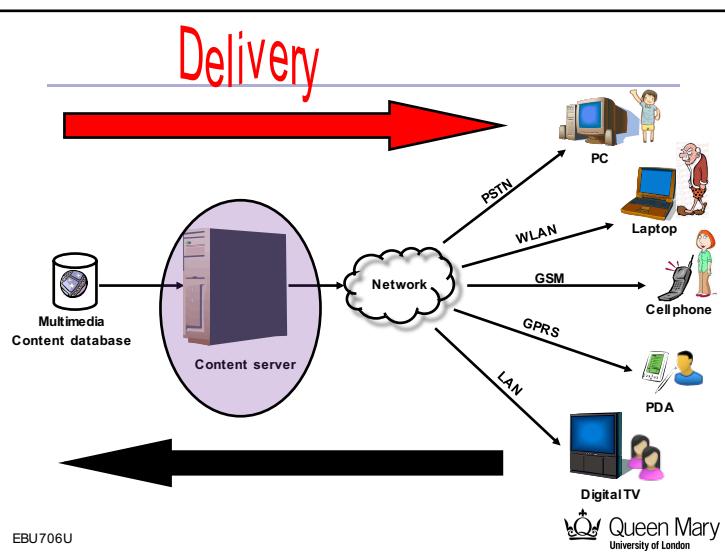
- Introduction to the course
- Frame-based multimedia (reminder) : JPEG / MPEG1&2
- Object-based multimedia : MPEG4
- Segmentation

- Week 2:

- [Multimedia delivery](#)
- Multimedia Content Annotation (MPEG7)



# Delivery

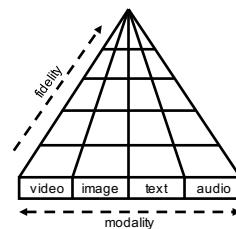


EBU706U



## Info Pyramid

- Content is processed many times
  - Generation of **variations**
  - Server selects most appropriate variation
- Variations: different versions of media objects with
  - Different **modalities**
    - Video
    - Image
    - Text
    - Audio
  - Different **fidelities**
    - Summarised
    - Compressed
    - Scaled variations



EBU706U



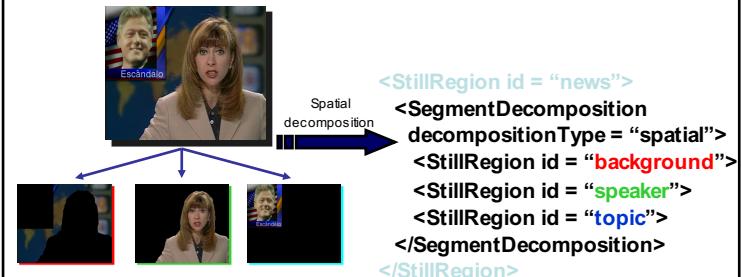
## Lecture plan (1)

- Week 1:
  - Introduction to the course
  - Frame-based multimedia (reminder) : JPEG / MPEG1&2
  - Object-based multimedia : MPEG4
  - Segmentation
- Week 2:
  - Multimedia delivery
  - Multimedia Content Annotation (MPEG7)

EBU706U



## XML image description



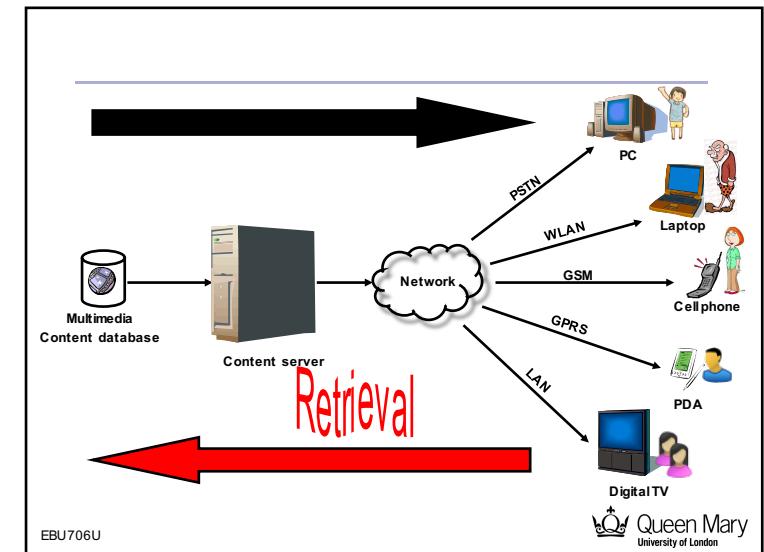
EBU706U



## Lecture plan (2)

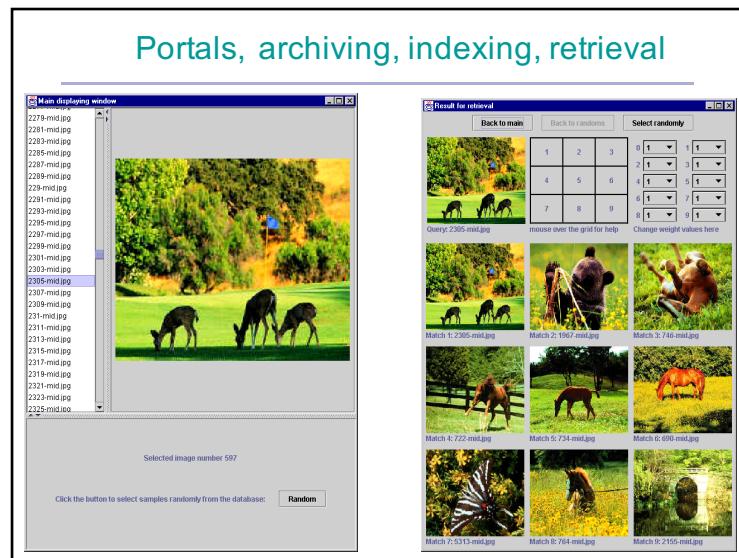
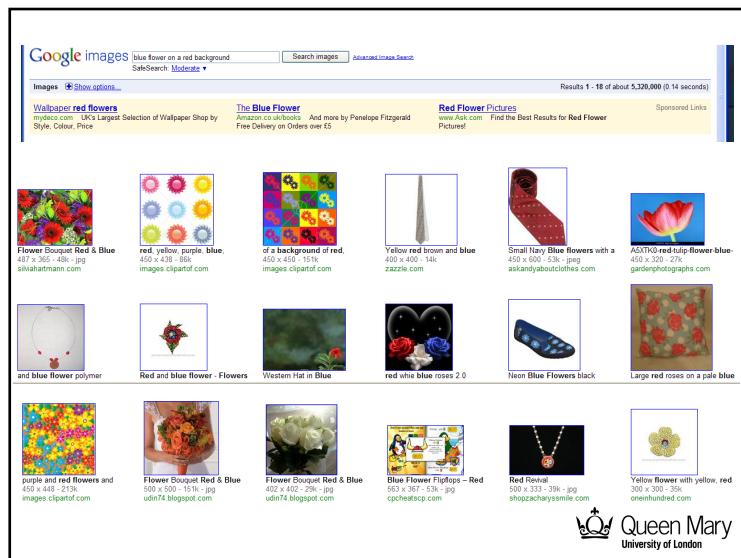
- Week 3:
  - Indexing and retrieval of still images
  - Indexing and retrieval of video
  - Indexing and retrieval of audio
- Week 4:
  - 3D multimedia
  - Copyright protection and authentication
  - Wrap up and final revisions

EBU706U



EBU706U



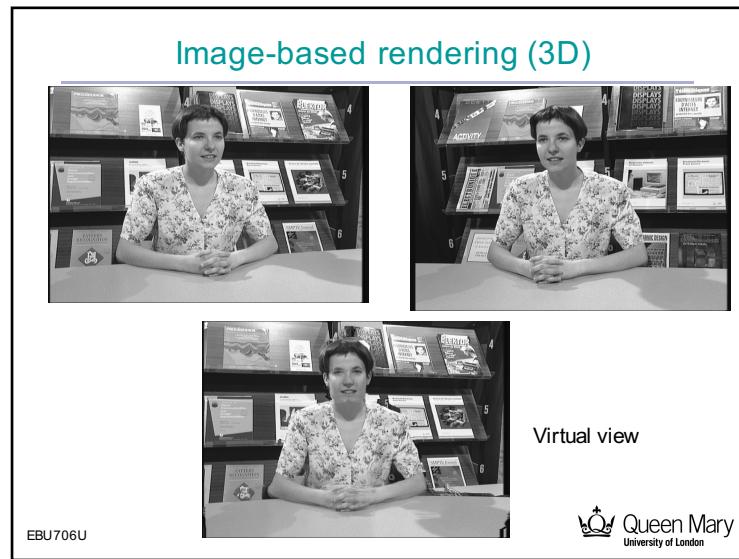


## Lecture plan (2)

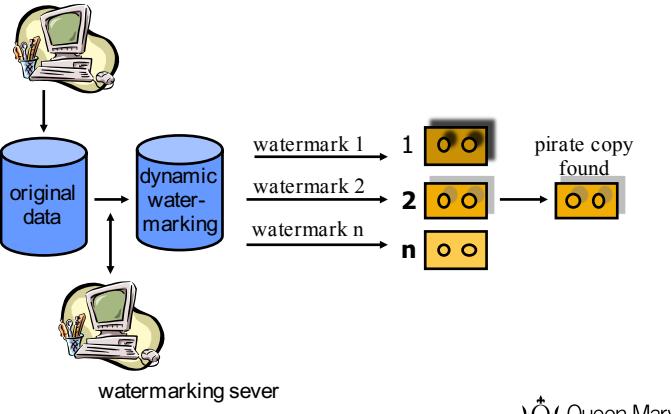
- Week 3:
  - Indexing and retrieval of still images
  - Indexing and retrieval of video
  - Indexing and retrieval of audio
- Week 4:
  - 3D multimedia
  - Copyright protection and authentication
  - Wrap up and final revisions

EBU706U

Queen Mary University of London



## Copyright protection



## Course material : QMplus

### GENERAL INFORMATION

### MODULE INFORMATION

### LECTURES AND TUTORIALS

All lecture slides and tutorials material can be found here.

Week 1

- 1\_Introduction
- 2\_FrameBased
- 3\_ObjectBased
- 4\_Segmentation
- Exercises
- MPEG4Paper

MPEG4PaperQuestions

EBU706U



## Assessment

### • Group Coursework: 20%

In this coursework you will study in detail an existing advanced multimedia application through reading academic articles.

Learning objectives:

- learning how to read a scientific article
- practicing searching for and understanding scientific articles.
- being able to compare various approaches to solve a scientific problem.

### • Final exam : 80%

EBU706U



## Resources

### Books

- Multimedia Communication Systems – Techniques, Standards, and Networks  
K.R. Rao, Z.S. Bojkovic, D.A. Milovanovic (Prentice Hall)
- Digital Multimedia
  - N Chapman & J Chapman, Wiley
  - Video Coding – An Introduction to Standard Codecs
    - M Ghanbari, IEE Publishing
  - Handbook of Image and Video Processing
    - A Bovik, Academic Press, ISBN 0-12-119790-5
  - Internetworking Multimedia
    - J Crowcroft, M Handley, I Wakeman, Taylor & Francis
  - Computer Vision
    - L G Shapiro and G C Stockman, Prentice Hall, ISBN 0-13-030796
- QMplus
- Web

EBU706U



## EBU706U - Multimedia Systems

Revisions : frame-based multimedia & compression

Dr. Yi-Zhe Song

EBU706U



JPEG: 1% Compression



EBU706U



80% Compression



EBU706U



99% Compression



EBU706U



## Today's agenda

- Redundancy
- Discrete cosine transform (DCT)
- Image coding: JPEG
  - Quantization
  - Zig-zag scan
  - Differential Pulse Code Modulation
  - Run Length Coding
  - Entropy coding
- Video coding
  - MPEG

EBU706U



## Data and information

- Data is not the same thing as information!
  - Data is the means with which information is expressed
  - The amount of data can be much larger than the amount of information
  - Data that provide no relevant information = redundant data or redundancy
- Goal of image coding or compression
  - to reduce the amount of data by reducing the amount of redundancy

EBU706U



## Redundancy

- Types of redundancy
  - Coding redundancy      Huffman coding
    - some grey levels / colours are more common than others
  - Spatial redundancy (e.g. inter-pixel redundancy)
    - the same grey level covers large areas
    - neighboring samples on a scanning line are normally similar
  - Temporal redundancy
    - neighboring images in a video sequence may be similar
  - Psycho-visual redundancy
    - the eye can only resolve 32 grey levels locally

EBU706U



## variable-length coding Coding redundancy

- Redundancy when mapping from the pixels (symbols) to the final compressed binary code
- Example:

Symbol	Occurrence Probability	Code 1	Code 2
a <sub>1</sub>	0.1	000	0000
a <sub>2</sub>	0.2	001	01
a <sub>3</sub>	0.5	010	1
a <sub>4</sub>	0.05	011	0001
a <sub>5</sub>	0.15	100	001

- $L_{avg,1} = 3$  bits/symbol
- $L_{avg,2} = 4 \times 0.1 + 2 \times 0.2 + 0.5 + 4 \times 0.05 + 3 \times 0.15 = 1.95$  bits/symbol.
- Code 2 is also unique and shorter

EBU706U



## Coding redundancy

- Redundancy when mapping from the pixels (symbols) to the final compressed binary code
- Example:

Symbol		Code 1	Code 2
$a_1$	0.1	000	0000
$a_2$	0.2	001	01
$a_3$	0.5	010	1
$a_4$	0.05	011	0001
$a_5$	0.15	100	001

Variable Length Coding (VLC)

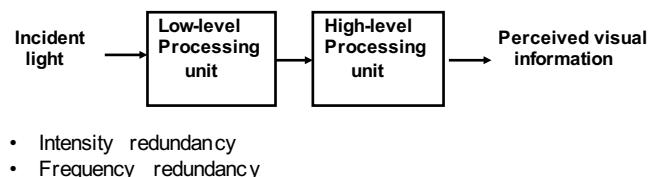
- $L_{avg,1} = 3$  bits/symbol
- $L_{avg,2} = 4 \times 0.1 + 2 \times 0.2 + 0.5 + 4 \times 0.05 + 3 \times 0.15 = 1.95$  bits/symbol.
- Code 2 is also unique and shorter

EBU706U



## Psycho-visual redundancy

- The "end-user" is a human => only represent the info. which can be perceived by the Human Visual System (HVS)
- From a data's point of view => lossy
- From the HVS's point of view => lossless perception is lossless



EBU706U



## Spatial redundancy

- Pixel values are not spatially independent
- High correlation among neighbor pixels

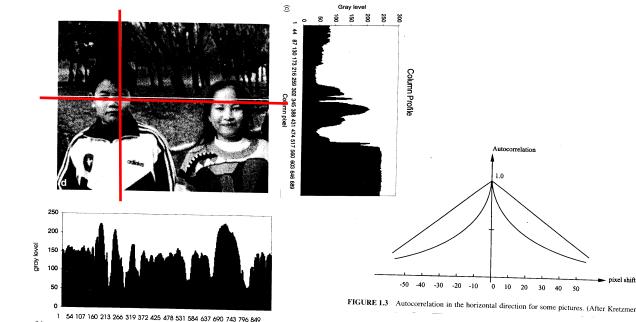


FIGURE 1.3 Autocorrelation in the horizontal direction for some pictures. (After Kritzer, 1952.)

EBU706U



## Intensity redundancy

$$\frac{\Delta I}{I} = K$$

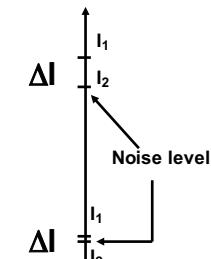
The Weber fraction



greyscale

- Weber's Law states that the ratio of the **increment threshold** (i.e. perceptible difference) to the **background intensity** is a constant.
- The high (bright) values need a less accurate representation compared to the low (dark) values
- Weber's law holds for all human senses!

EBU706U



## Frequency redundancy

- The human eye functions as a lowpass filter =>
  - High frequencies in an image can be "ignored" without the Human Visual System noticing
  - Key issue in lossy image compression

EBU706U



## Image compression

- **Reversible** image compression
  - *lossless* → no loss of information
  - new image is identical to original image (after decoding).
  - necessary in most image analysis
  - compression ratio → typically 2-10x.
- **Non reversible** image compression
  - *lossy* → loss of some information
  - often used in image communication, video, web
  - important: the image should be visually "nice"
  - compression ratio → typically 10-30x.

EBU706U



## Question ..

- Which types of redundancy are exploited in the following compression methods ?
  - Run Length Encoding (RLE) *lossless*
  - Huffman encoding *lossless*
  - Prediction encoding (DPCM) 差分编码 *lossless*
  - JPEG (i.e. quantisation) *lossy*

EBU706U



## Today's agenda

- Redundancy
- **Discrete cosine transform (DCT)**
- Image coding: JPEG
  - Quantization
  - Zig-zag scan
  - Differential Pulse Code Modulation
  - Run Length Coding
  - Entropy coding
- Video coding
  - MPEG

EBU706U



## Discrete Cosine Transform

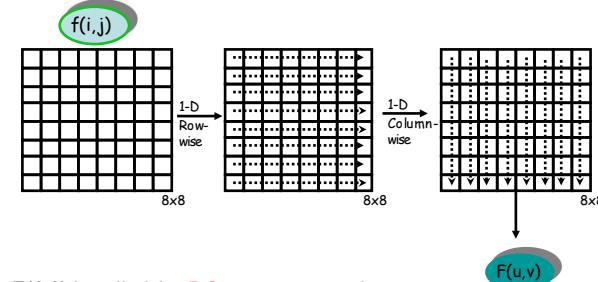
- DCT
  - changes spatial **intensity** values to spatial **frequency** values.
  - roughly **arranges** values from lowest frequency to highest frequency:
    - lowest frequencies → represent coarse details
    - highest frequencies → represent fine details
  - some high frequency parts can be dropped
- exploits features of the human eye
  - the eye is unable to perceive brightness levels above or below certain thresholds
  - less sensitive to the higher spatial frequency components than the lower frequencies → transform coding

EBU706U



## Computing a 2-D DCT for images

- Two series of 1-D transforms result in a 2-D transform



$F(0,0)$  is called the **DC component** and the rest of  $F(i,j)$  are called **AC components**

EBU706U



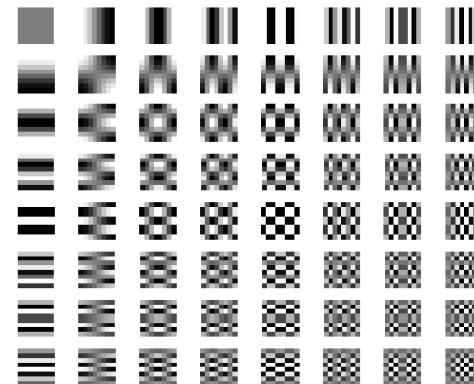
## DCT: properties

- The DCT is **separable**
  - to estimate the DCT coefficients of a two-dimensional signal for an image  $I(x, y)$ , the 1-D transform can be performed twice
    - once for the rows (y-axis)
    - once for the columns (x-axis)
- The DCT is **invertible**
  - it allows us to move back and forth between **spatial** and **frequency domains**

EBU706U



## 2-D DCT basis



EBU706U



80% Compression



EBU706U

Queen Mary  
University of London

99% Compression



EBU706U

Queen Mary  
University of London

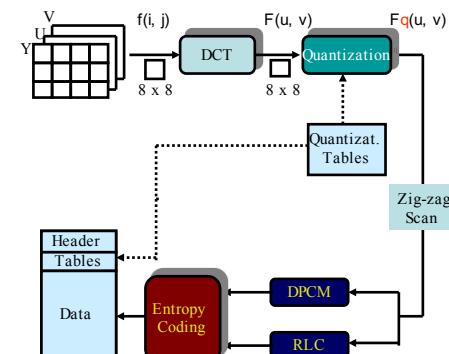
## Today's agenda

- Redundancy
- Discrete cosine transform (DCT)
- **Image coding: JPEG**
  - Quantization
  - Zig-zag scan
  - Differential Pulse Code Modulation
  - Run Length Coding
  - Entropy coding
- Video coding
  - MPEG

EBU706U

Queen Mary  
University of London

## JPEG overview (block diagram)



EBU706U

Queen Mary  
University of London

## JPEG overview

- **JPEG**

- DCT of each 8x8 pixel array  $f(x,y) \rightarrow F(u,v)$
- Quantization using a table or using a constant
- Zig-zag scan to exploit redundancy
- Differential Pulse Code Modulation (DPCM) on DC components
- Run Length Coding (RLC) of the AC components
- Entropy coding (Huffman) of the final output

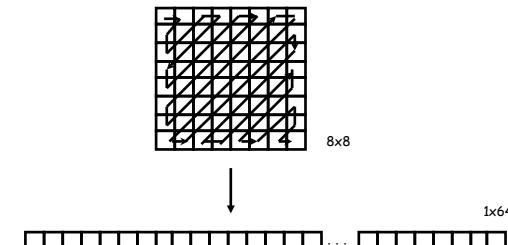
EBU706U



## Zig-zag scan

- **Zig-zag scan**

- To group low frequency coefficients in top of vector
- Maps  $8 \times 8$  to a  $1 \times 64$  vector



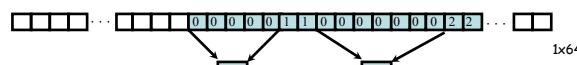
EBU706U



## RLC (RLE) on AC Components

- Run Length Coding (RLC)

- the  $1 \times 64$  vectors have a lot of **zeros**, esp. towards the end of the vector
  - higher up entries in the vector capture higher frequency (DCT) components which tend to capture less of the relevant content
  - could have been as a result of using a quantization table
- encode a series of 0s as a (**skip,value**) pair, where **skip** is the number of zeros and **value** is the next non-zero component
  - **run**: repeated occurrence of the same character
  - **length of the run**: number of repetitions



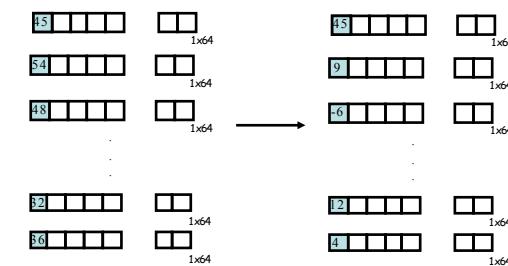
EBU706U



## DPCM on DC Components

- Differential Pulse Code Modulation (DPCM)

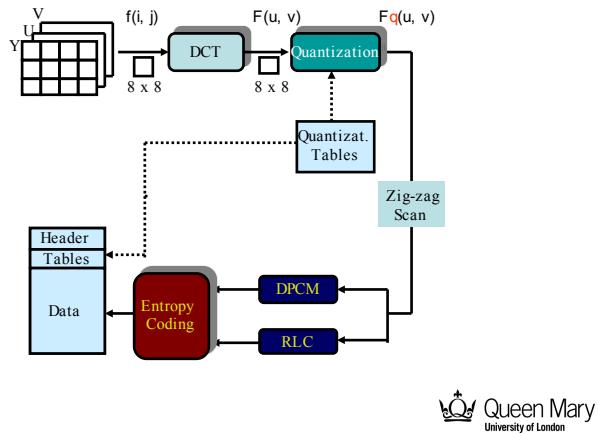
- Encode the **difference** between the current and previous  $8 \times 8$  block
- The DC component value in each  $8 \times 8$  block is large and varies across blocks, but is often close to that in the previous block.



EBU706U



## JPEG overview (block diagram)



## Entropy

- Entropy
  - amount of information  $I$  in a symbol of occurring probability  $p$

$$I = \log_2(1/p)$$

- symbols that occur rarely convey a large amount of information
- average information per symbol  $i \rightarrow$  entropy  $H$

$$H = p_i \times \log_2(1/p_i)$$



## Entropy

- Entropy of an information source  $S$ :

$$H(S) = \sum_i (p_i \times \log_2(1/p_i))$$

$\log_2(1/p_i)$ : amount of information contained in  $S_i$ ,  
(the number of bits needed to code  $S_i$ )

### Example

Image with uniform distribution of gray-level intensity:  $p_i = 1/256$   
number of bits needed to code each gray level: 8 bits  
 $\rightarrow$  entropy of this image: 8

EBU706U



## Entropy coding

- DC components are differentially coded as  $(SIZE, Value)$ 
  - The code for a  $Value$  is derived from the following table

SIZE	Value	Code
0	0	---
1	-1, 1	0, 1
2	-3, -2, 2, 3	00, 01, 10, 11
3	-7, ..., -4, 4, ..., 7	000, ..., 011, 100, ..., 111
4	-15, ..., -8, 8, ..., 15	0000, ..., 0111, 1000, ..., 1111
.		.
.		.
11	-2047, ..., -1024, 1024, ..., 2047	...

EBU706U



## Entropy coding

SIZE	Code Length	Code
0	2	00
1	3	010
2	3	011
3	3	100
4	3	101
5	3	110
6	4	1110
7	5	11110
8	6	111110
9	7	1111110
10	8	11111110
11	9	111111110

EBU706U

- DC components are differentially coded as (SIZE, Value). The code for a SIZE is derived from the table on the left

**Example:** If a DC component is 40 and the previous DC component is 48 → the difference is -8 → it is coded as

**1010111**

- 0111:** The value for representing -8  
(table in previous slide)  
**101:** The size from the same table reads 4. The corresponding code from the table on the left is 101



## Huffman encoding

- Let an alphabet have  $N$  symbols  $S_1 \dots S_N$
- Let  $p_i$  be the probability of occurrence of  $S_i$
- Order the symbols by their probabilities  

$$p_1 \geq p_2 \geq p_3 \geq \dots \geq p_N$$
- Replace symbols  $S_{N-1}$  and  $S_N$  by a new symbol  $T_{N-1}$  such that it has the probability  $p_{N-1} + p_N$
- Repeat until there is only one symbol
- This generates a **binary tree**

EBU706U



## Example

If a DC component is 83 and the previous DC component is 80 → How is this component coded?

The difference is 3.

11: The value for representing 3.

The size for 3 (from the same table) reads 2.

The corresponding code from the next table is 011.

Therefore it is coded as:

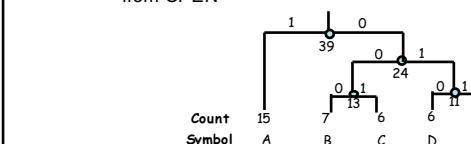
01111

EBU706U



## Huffman encoding

- Initialization**
  - put all nodes in an OPEN list  $L$
  - keep it sorted at all times (e.g., ABCDE)
- Repeat** the following steps until the list  $L$  has only one node left
  - From  $L$  pick two nodes having the lowest frequencies
  - Create a parent node of them
  - Assign the **sum** of the children's frequencies to the parent node and insert it into OPEN
  - Assign **code 0, 1** to the two branches of the tree, and delete the children from OPEN



EBU706U



## Huffman encoding

- Fixed-length inputs become variable-length outputs
- Average codeword length  $\rightarrow \sum l_i p_i$

Symbol	Count	Info. $-\log_2(p_i)$	Code	Subtotal# of Bits
A	15	1.38	1	15
B	7	2.48	000	21
C	6	2.70	001	18
D	6	2.70	010	18
E	5	2.96	011	15

EBU706U



## Huffman encoding

- Fixed-length inputs become variable-length outputs
- Average codeword length  $\rightarrow \sum l_i p_i$

Symbol	Count	Info. $-\log_2(p_i)$	Code	Subtotal# of Bits
A	15	1.38	1	15
B	7	2.48	000	21
C	6	2.70	001	18
D	6	2.70	010	18
E	5	2.96	011	15

Question: Why is Huffman coding sub-optimal?

EBU706U



## Example: JPEG 1% compression



EBU706U



## Example: JPEG 80% compression



EBU706U



## Example: JPEG 99% compression



EBU706U



## Today's agenda

- Redundancy
- Discrete cosine transform (DCT)
- Image coding: JPEG
  - Quantization
  - Zig-zag scan
  - Differential Pulse Code Modulation
  - Run Length Coding
  - Entropy coding
- Video coding
  - MPEG

EBU706U



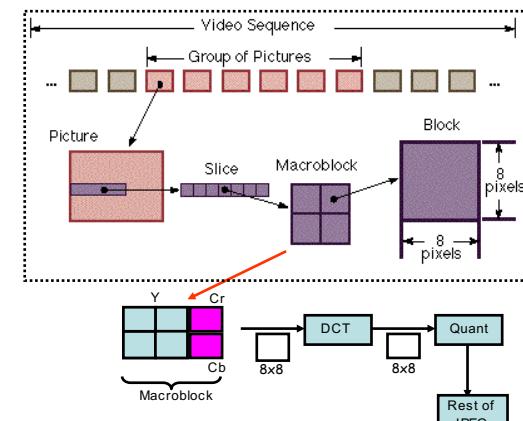
## Video compression

- Video as a sequence of pictures (or frames)
- JPEG algorithm applied to each frame
  - moving JPEG (**MJPEG**)
  - exploits only intra-frame redundancy
- High correlation between successive frames
  - only small portion of each frame is involved with any motion
  - use a combination of actual frame contents and predicted frame contents
  - Motion estimation and motion compensation
  - **Inter-frame** and **intra-frame** coding
    - high compression ratios can be achieved by using both
    - random access requirement of image retrieval is satisfied by **pure intraframe coding**

EBU706U



## From video to blocks



EBU706U



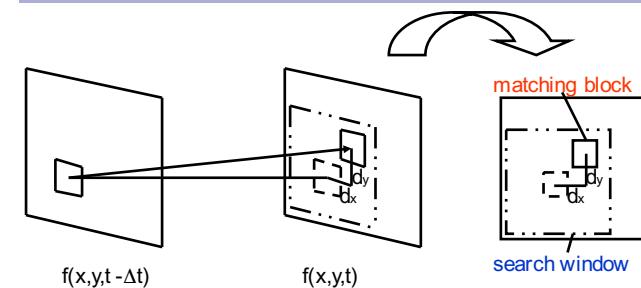
## MPEG video encoding

- MPEG video encoding
  - input frames are **preprocessed**
    - color space conversion
    - spatial resolution adjustment
  - frame types** are decided for each frame/picture
  - each picture is divided into **macroblocks** of 16 X 16 pixels
  - macroblocks
    - are **intracoded** for I frames
    - are **predictive coded or intracoded** for P and B frames
    - are divided into **six blocks** of 8 X 8 pixels
      - 4 luminance and 2 chrominance
    - DCT is applied to each block → **transform coefficients**
      - quantized
      - zig-zag scanned
      - variable-length coded

EBU706U



## Motion estimation

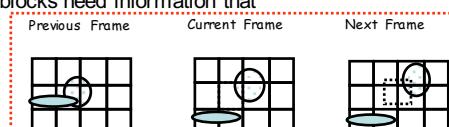


$$\min \sum_B \|f(x, y, t) - f(x - d_x, y - d_y, t - \Delta t)\|$$

EBU706U



## MPEG

- Prediction
  - Limit:** some macroblocks need information that is not present in the *previous reference frame*

  - Such information might be available in a *subsequent frame* ...
- MPEG
  - Uses a Bi-directional frame type (**B-frame**)
    - to form a B-frame: search for matching MBs in both past and future frames
    - typical pattern is **IBBPBBPBB IBBPBBPBB IBBPBBPBB**
    - actual pattern is up to encoder, and need not be regular

EBU706U



previous frame



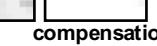
current frame



next frame

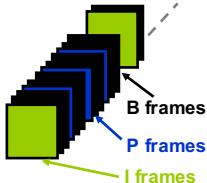


EBU706U



### Predictive coding + interpolation with motion compensation

- Images coded **INTRA**:
  - Random access
  - Error resilience
- Images coded **INTER** (P):
  - Prediction from previous decoded image (I, P)
- Images coded **BI-INTER** (B):
  - Prediction from previous and or future decoded image (I, P)
  - allow effective prediction of **uncovered background** (areas of the current picture that were not visible in the past and visible in the future)

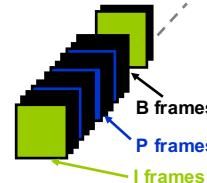


EBU706U



### Predictive coding + interpolation with motion compensation

- Question:** State two advantages to the fact that B-pictures are **not** used for prediction.



EBU706U

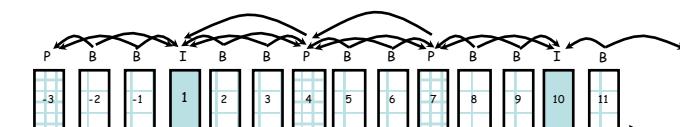


### Bitstream order vs. display order

- Bitstream order

1(I), 4(P), 2(B), 3(B), 7(P), 5(B), 6(B), 10(I), 8(B), 9(B)

- Display order



EBU706U



### Image and video compression: summary

- Two dimensional array of pixel values
- Spatial redundancy and temporal redundancy
- Statistical encoding
  - exploits the fact that not all **symbols** in the source information occur with equal **probability**
  - variable length **codewords** are used with the shortest ones
- Human eye is less sensitive
  - to **chrominance** signal than to luminance signal → U and V can be coarsely coded
  - to the higher spatial **frequency** components
  - to quantizing **distortion** at high luminance levels

EBU706U



## Frame types: summary

- I-frames
  - coded **without reference** to other frames
  - serve as reference pictures for predictive-coded frames
- P-frames
  - coded using motion compensated **prediction** from a past I-frame or P-frame.
- B-frames
  - **bi-directionally** predictive-coded
  - highest degree of compression, but require both past and future reference pictures for motion compensation.
- D-frames
  - are **DC-coded**
  - only the DC coefficients of the DCT coefficients are present
  - used in interactive applications
    - E.g., VoD for rewind and fast-forward operations

EBU706U



## What did we learn today?

- Redundancy
- Discrete cosine transform (DCT)
- Image coding: JPEG
  - Quantization
  - Zig-zag scan
  - Differential Pulse Code Modulation
  - Run Length Coding
  - Entropy coding
- Video coding
  - MPEG

EBU706U



## Multimedia Systems

### Object-based multimedia

Dr. Yi-Zhe Song

EBU706U



### Object-based Multimedia



EBU706U



## Agenda

- Frame-based vs Object-based multimedia
- Introduction to segmentation
  - Chroma-keying
  - Video analysis
  - Motion segmentation and tracking
- Applications

EBU706U



### Object-based Multimedia



EBU706U

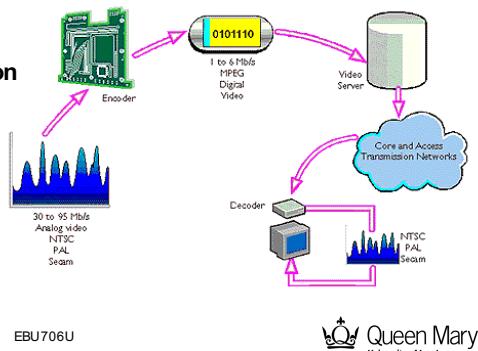


## The frame-based model

### A video codec

#### + An audio codec

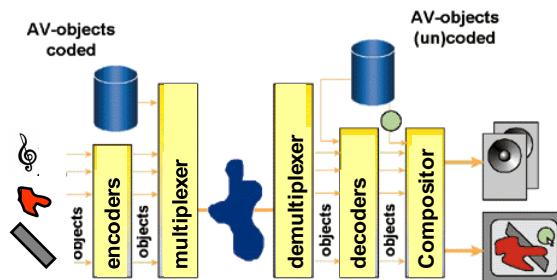
#### + Synchronisation / Multiplexing



EBU706U



## The object-based model



EBU706U



## Context: object-based video processing

- First generation image/video coding
  - Pixel (frame)-based schemes: JPEG, MPEG-1/2, H.261/3/4
- Object-based image/video representation
  - coding : MPEG-4 low-level semantic
  - content description: MPEG-7 high-level semantic
- Object-oriented functionalities in image/video processing
  - editing, browsing, composition
  - scalability (spatial, temporal, content)
  - interactivity   
MPEG-7 is not a compression technique, but a structured metadata

EBU706U



## Object-based multimedia

- Audiovisual scenes represented as a **composition of objects**
- Integration** of objects of different nature: audio & video, natural & synthetic, text & graphics, speech & music, animated faces, 3D models ...
- Object-based hyperlinking, processing, coding and description**
- Interaction** with objects
- Object-based content may be **reused in different contexts**



EBU706U



## Object-based video

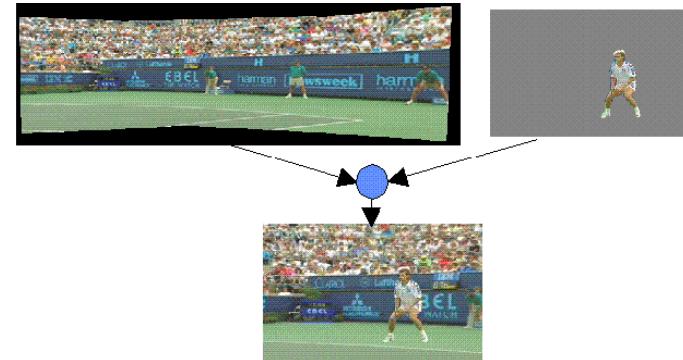
- Video technology and video coding
  - From frame-based to object-based techniques
  - From limited to high capabilities to access, manipulate and create video content



EBU706U



## Video composition / coding



EBU706U



## Agenda

- Frame-based vs Object-based multimedia
- Introduction to segmentation
  - Chroma-keying
  - Video analysis
  - Motion segmentation and tracking
- Applications

EBU706U



## Segmentation

- Segmentation
  - Objective: to divide an image or video into objects
- Why do we want to segment?
  - To achieve higher compression ratios
    - object-based coding
  - To enable model-based coding
  - To offer new functionalities
    - MPEG-4 sprites for integration and composition
  - Accurate estimation of motion parameters
    - for single objects using the segmentation information efficiently

EBU706U



Q: What kind of technique is behind portrait mode in phone photography?

A: Segmentation

## Segmentation

hard to segment – no edge

EBU706U

Queen Mary University of London

(if the image is 3d: depth)      Segmentation      Background: texture

ball: colour      table: shape      Tricky thing: self-shadow

EBU706U

Queen Mary University of London

## Simple segmentation scheme

- Segmentation scheme
  - Recognition of uniform image areas using variance-based interest operators
  - Thresholding techniques
- Assumptions
  - The foreground can be easily distinguished from the background
  - The background is almost uniform

HOMOGENEOUS PROPERTY

EBU706U

Queen Mary University of London

colour-based segmentation + thresholding  
reason: background colour is uniform

## Chroma-keying

EBU706U

Queen Mary University of London

Q: What if the scene contains all the colors? How to segment?

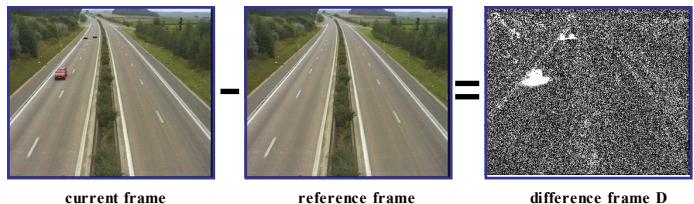
A: Texture-based segmentation.

Q: Segment two teams in a ball game.

A: 1. team shirts;  
2. directions in which people are moving.

## Moving objects segmentation

- Background subtraction



- Problem

$$D = \{d_k\}, d_k \neq 0 \text{ even if there is no structural change in } k$$

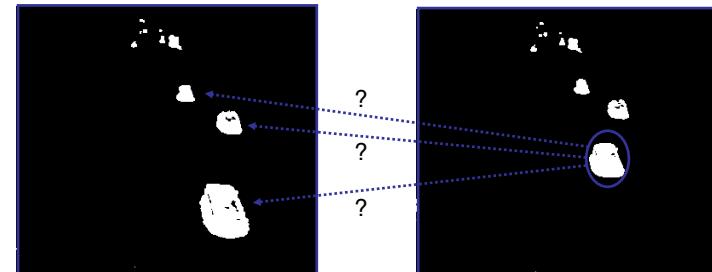
EBU706U



## Moving object tracking

- Motion segmentation (change detection)

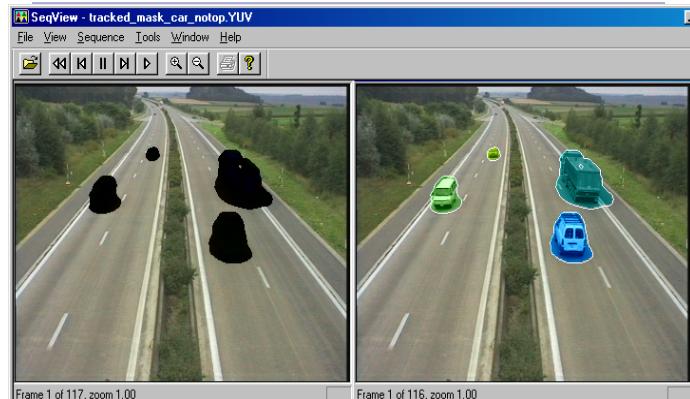
- Result:* localization of the objects in each frame
- Problem:* link instances of objects between frames



EBU706U



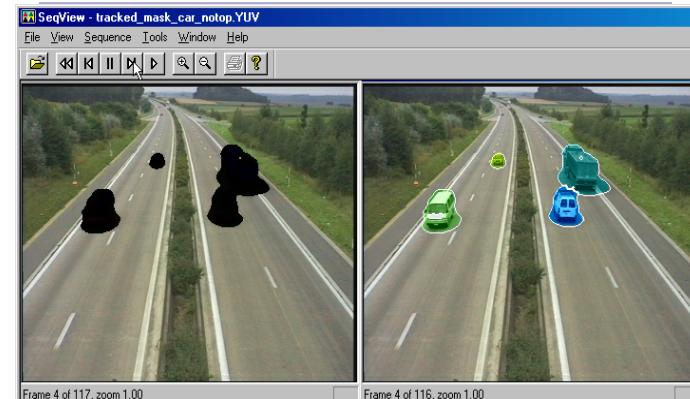
## Example: tracked objects



EBU706U



## Example: tracked objects



EBU706U



bigest problem: Occlusion

## Chroma-keying



- The principal subject is filmed or photographed against a background consisting of a single colour.
- The portions of the video which match the preselected colour are replaced by the alternate background video.

EBU706U



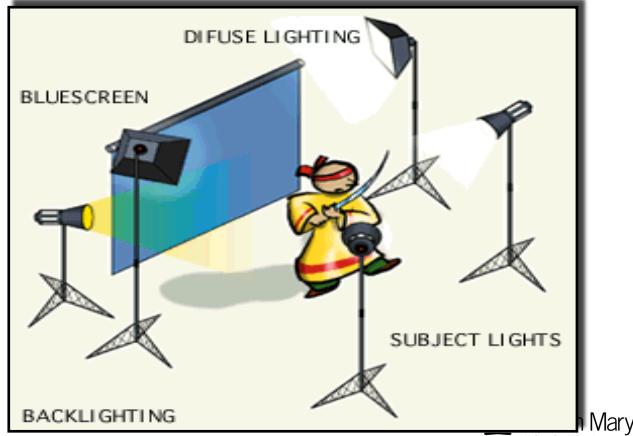
## Chroma-keying



EBU706U



## Chroma-keying: studio



EBU706U

## Chroma-keying: One-sentence Summary

EBU706U



## Chroma-keying: One-sentence Summary

Chroma keying is a technique for mixing two images together, in which a colour from one image is removed, revealing another image behind it.

考试别写原句

EBU706U



## Video analysis for object-based multimedia

- Objective
  - Automatic video object segmentation
    - To extract the **main content message**
  - Classification of the pixels in the video sequence into two classes:
    - *foreground pixels*
    - *background pixels*
  - Decompose each frame of the reference sequence into sets of mutually exclusive and jointly exhaustive segments
  - Use a priori information → **application dependent**

EBU706U



## Question time ...

Q1. Within the universe of Harry Potter, an invisibility cloak is used to make the wearer invisible. How can chroma-key help create the invisibility cloak effect in Harry Potter's movies?

wear a cloak that is the same colour with the background.

Q2. When filming a subject in front of a blue screen, why should the subject:

- not stand too close to the blue screen? shadows
- not wear shiny jewellery? light reflecting on the bg screen

Q3. When creating greyscale images, what would you use instead of a blue screen?

black or white

EBU706U



## Video analysis

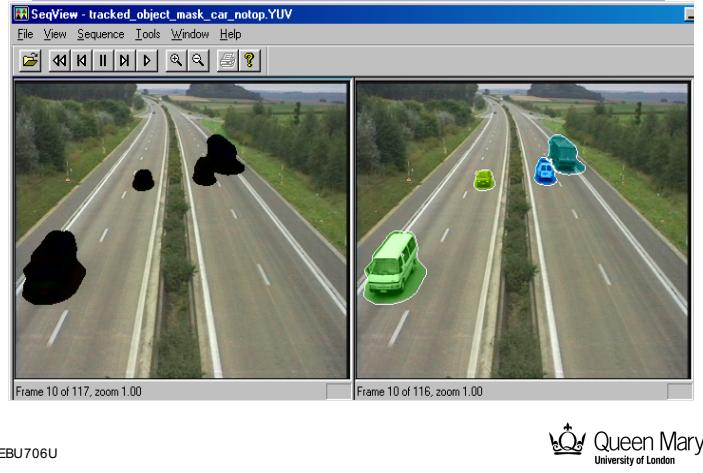
a sliding window

– calculate matching score

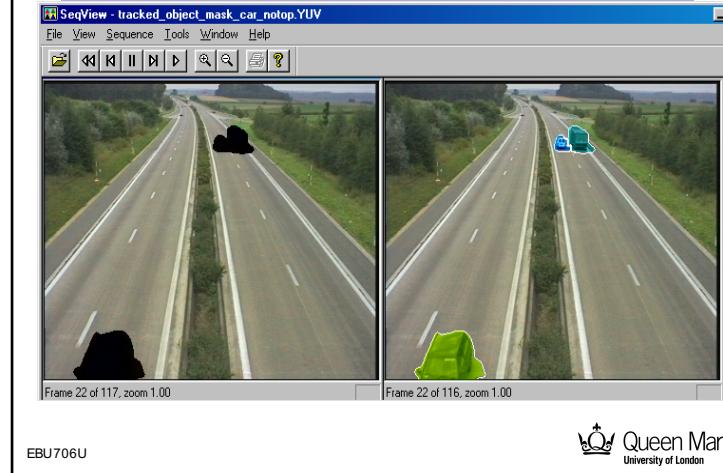
- A priori information: examples
  - Template matching (shape)
  - Extraction of captions and text (geometry)
  - Face detection (colour)
  - **Moving object segmentation** (change)
    - Sport broadcasting motion vector
    - Video surveillance



### Example: tracked objects

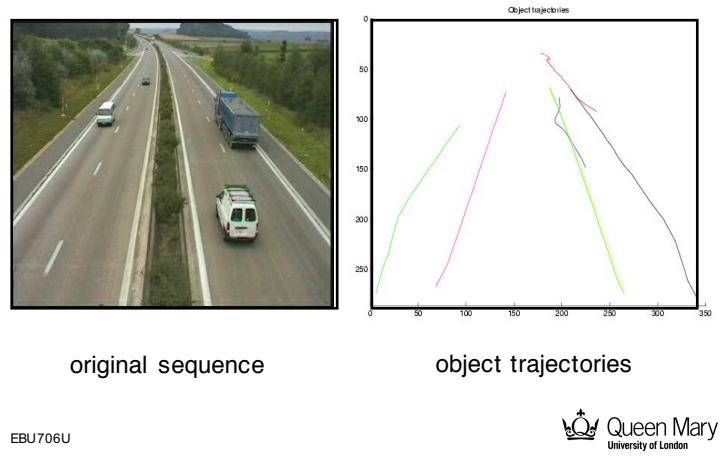


### Example: tracked objects

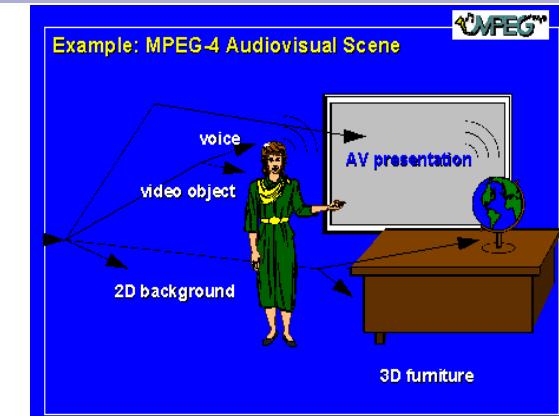


application: traffic violation detection

### Example: object trajectories



### MPEG-4



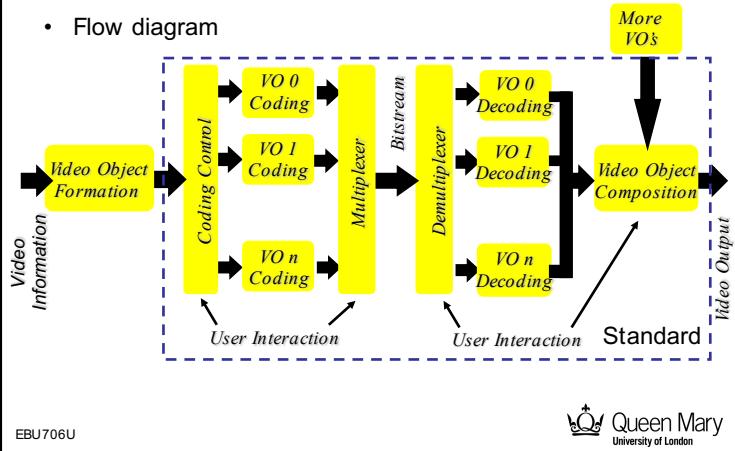
EBU706U

Queen Mary University of London

嗯。。不考

## MPEG - 4

- Flow diagram



EBU706U



## Agenda

- Frame-based vs Object-based multimedia
- Introduction to segmentation
  - Chroma-keying
  - Video analysis
  - Motion segmentation and tracking
- Applications



## Object-Based Multimedia Applications

- Video coding
- Video indexing retrieval – performed by human – because it's more semantics
- Video editing / production
- Hyper video
- Augmented reality
- Virtual presence
- Advanced surveillance
- Etc.

EBU706U



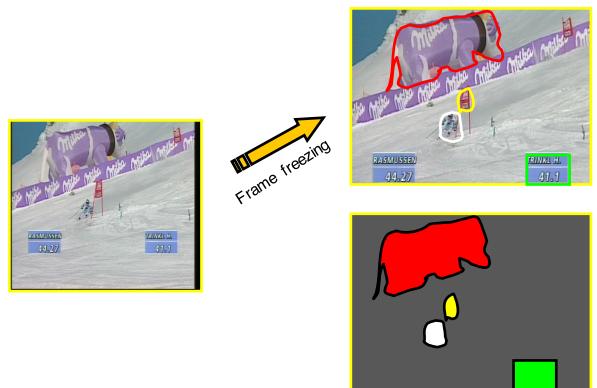
## Video production



EBU706U



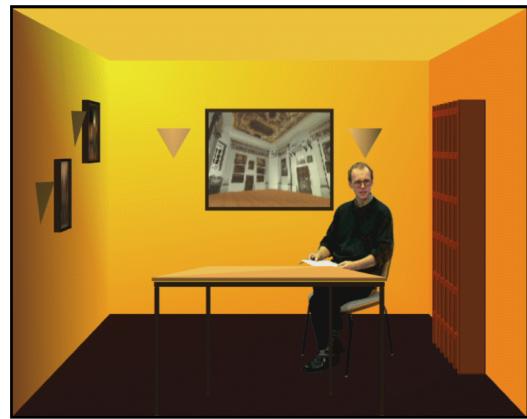
## Hyper video



EBU706U



## Virtual presence



EBU706U



## Augmented reality

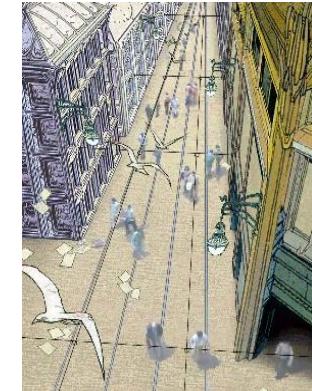


EBU706U



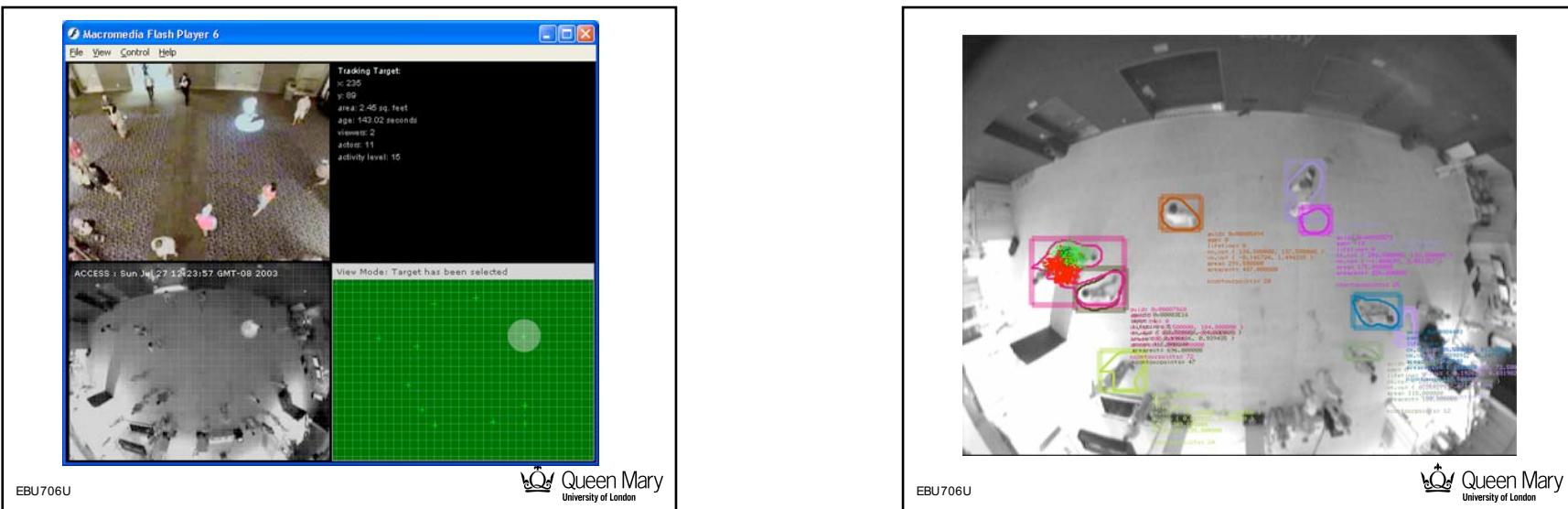
## Virtual presence

- Web cam
- Virtual background
- Object segmentation
- Video objects and virtual space integration



EBU706U





### Summary: object-based multimedia

- Storage and transmission
  - Low bit rate coding
  - Object-based video database
- Analysis and manipulation
  - Virtual presence
  - Augmented reality
  - Advanced surveillance
  - Hyper-video
  - Post-production
  - Video games



EBU706U

Queen Mary  
University of London

## Multimedia Systems

### Segmentation

Dr. Yi-Zhe Song

EBU706U



## Agenda

- Image segmentation
- Video segmentation
- Segmentation quality

EBU706U



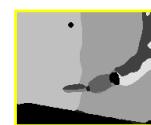
### Segmentation

- Segmentation subdivides an image  $R$  in  $N$  disjoint regions

$$R = \bigcup_{i=0}^{N-1} R_i \quad R_i \cap R_j = \emptyset \quad i \neq j$$

No overlapping

- Each region is assigned a label represented by a grey level or by a colour



denoted as a  
'segmentation map'

EBU706U



### Region or object segmentation

Two basic concepts



Regions

Homogeneous according to given criteria  
(colour, motion, texture...)

Objects

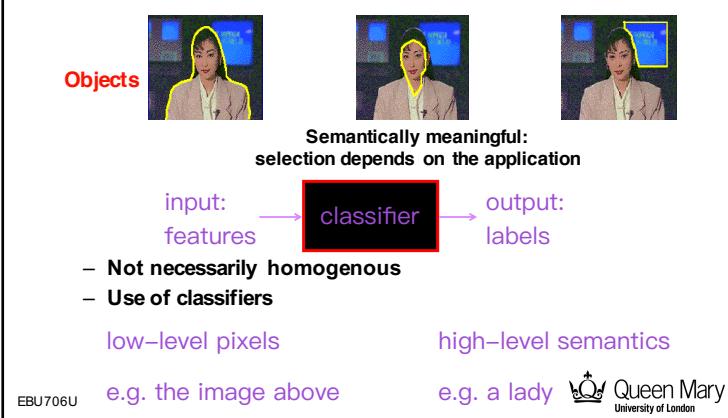


Semantically meaningful:  
selection depends on the application

EBU706U



## Region or object segmentation



## Types of segmentation

- Thresholding
    - based on pixel intensities (or colours)
      - e.g. chroma keying
      - e.g. use the shape of the histogram for automation
  - Region-based
    - group similar pixels
      - e.g., region growing, merge & split.
  - Edge-based
    - search for discontinuities in the image, and
    - try to connect objects or borders (often by a region based technique)
- only dependent on the size of window thus causing mass noises

Note: Segmentation is often the most difficult problem to solve in image analysis: there is no universal solution!



EBU706U

## Thresholding

- Thresholding
  - threshold  $T$  for pixel intensity classifies every pixels as belonging to objects (foreground) or background.
  - **Fixed** thresholds
    - the same value is used in all images
  - **Optimal** thresholding
    - based on the shape of the current image histogram.
    - Search for valleys, Gaussian distributions, etc.
  - **Local** (or **dynamic**) thresholding
    - The image is divided into overlapping sections which are thresholded one by one

global  
thresholding

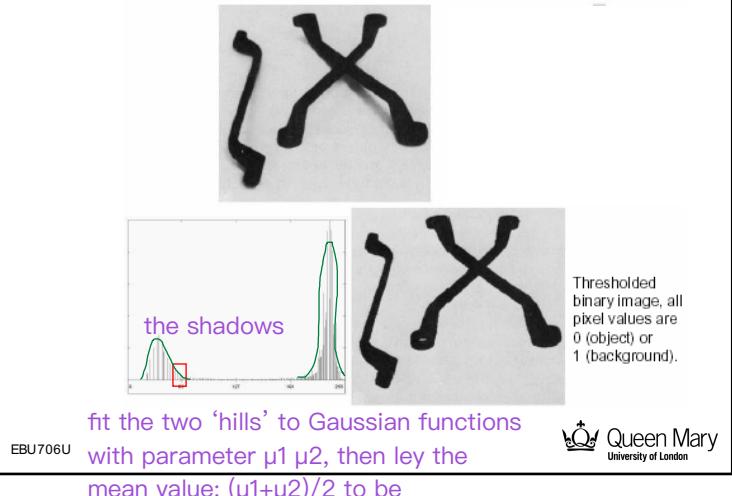
Note: Lighting conditions are extremely important, and it will only work under very controlled circumstances (remember chroma keying for example)

EBU706U



scenario: same blue screen under different lighting

## Example



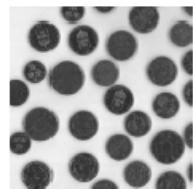
problem:  
reflected  
lights on  
coins

– global  
thresholding  
might fail

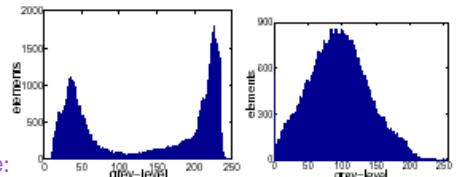
alternative:  
shape thresholding – circularity

EBU706U

## Examples



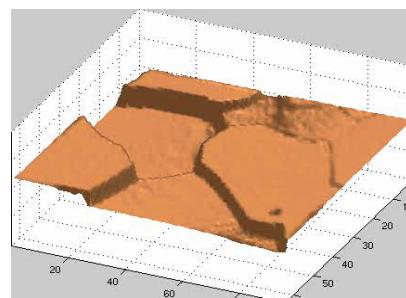
colour thresholding  
completely fails



alternative:  
texture thresholding



## Segmentation



EBU706U



## Simple segmentation scheme

- Segmentation scheme: (e.g. chroma-keying)
  - Recognition of uniform image areas
  - Thresholding techniques
- Assumptions
  - The foreground can be easily distinguished from the background
  - The background is almost uniform

EBU706U



## Edge-based segmentation

- Edge detection by gradient operators
  - Linking, e.g. local processing
  - natural for encoding curvilinear grouping
  - hard decisions often made prematurely



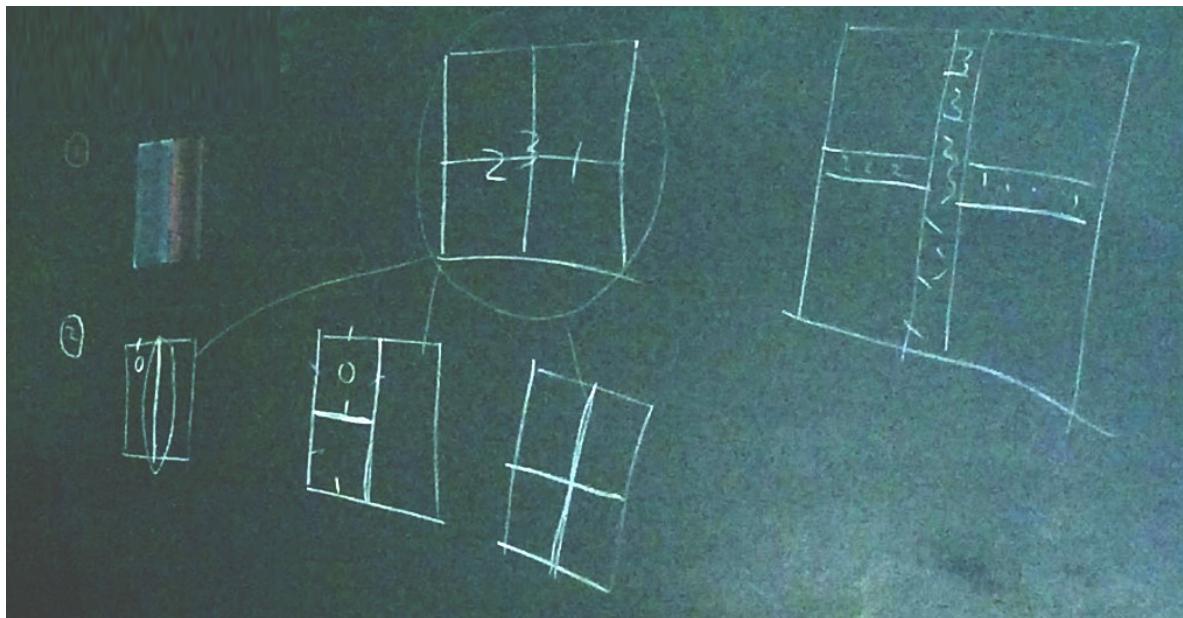
EBU706U



ways to present the segmentation:

1. segmentation map

2. edge detection



how human brains work:

1. edge detection

2. perceptual grouping

## Edge extraction in scale-space

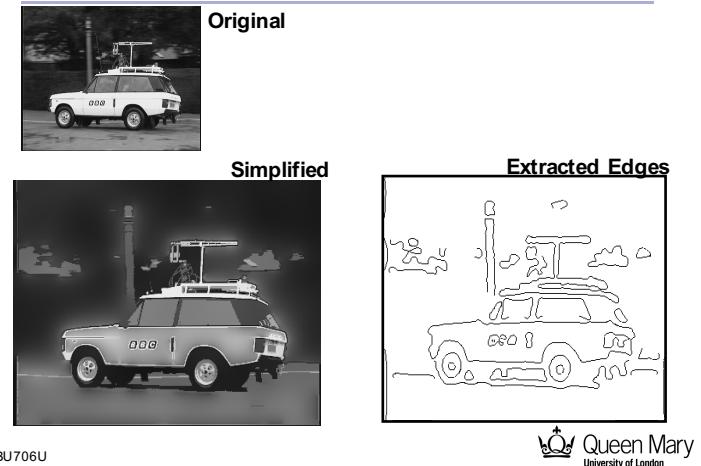
- Main image contours
  - selected by a **first order differential** edge detector
- Relevant edges
  - identified at **large scales** and
  - completed using edges at **small scales**
- Remaining small gaps
  - closed by **straight lines**

Note: Edges whose lengths do not exceed a given value are removed

EBU706U



## Example



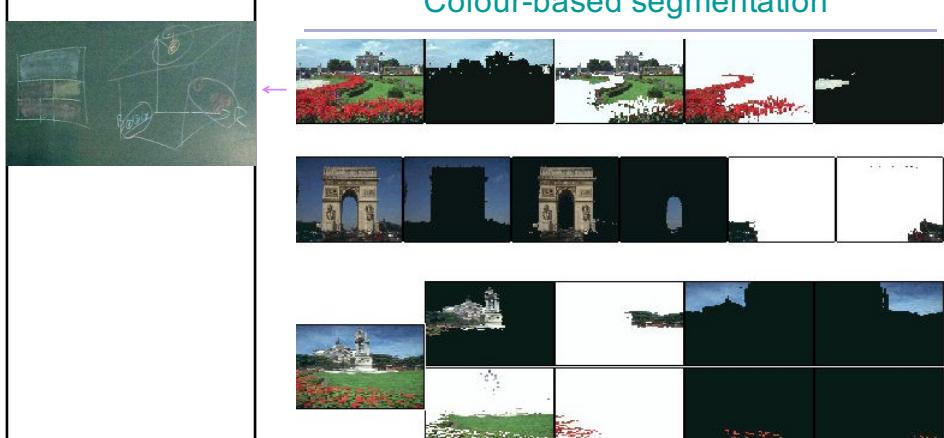
## Other features for segmentation

- Boundaries of **image regions** defined by a number of attributes
  - colour
  - texture
  - motion
  - stereoscopic depth
  - ...
- Challenges
  - interaction of multiple cues
  - local measurements to global percepts
  - interplay of image-driven and semantics-driven processing

EBU706U



## Colour-based segmentation

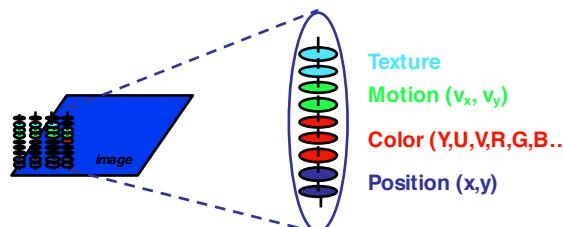


EBU706U



## Multi-feature approach

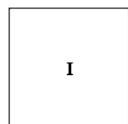
- Use of multiple features:
  - a **vector of features** for each pixel (“**feature vector**”)
  - exploit **coherence and redundancies** among features at the pixel level



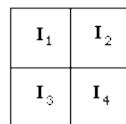
EBU706U



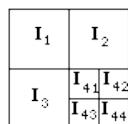
## Top-down segmentation



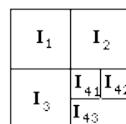
(a) Whole Image



(b) First Split



(c) Second Split



(d) Merge

EBU706U



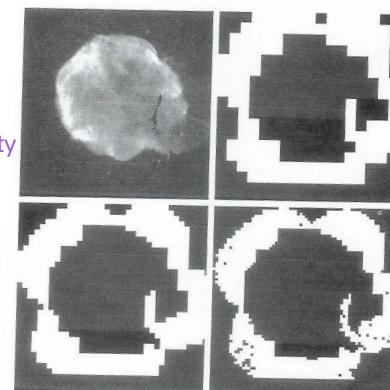
## Top-down segmentation

- Region splitting and merging
  1. Define some **criteria** for what is a **uniform** area
    - e.g., mean, variance, bimodality of histogram, texture, etc.
  2. Start with the full image and split it into 4 sub-images (**quadtree** method)
  3. Check each sub-image.
    - If not uniform → divide into 4 new sub-images.
  4. Compare regions with neighbouring regions
    - If uniform → merge
  5. Repeat 2-4 until nothing more happens

EBU706U



## Top-down segmentation



virtue:  
the uniformity  
is localised

EBU706U



## Bottom-up segmentation

- Region growing
  - 1. Find **starting points**
  - 2. Include neighbouring pixels with **similar** features
    - e.g., grey-level, texture, colour
  - 3. Continue until all pixels have been included with one of the starting points
- Problems
  - Not trivial to find **good starting points**, difficult to automate
  - Need good criteria for similarity.

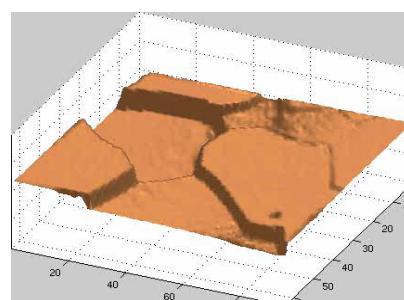
**very sensitive to local minima**

**virtue:** accurate

EBU706U



## Segmentation



EBU706U



## Watershed

- Think of the grey-level image as a **landscape**.
  - Let water rise from the bottom of each **valley** (the water from each valley is given its own label).
  - As soon as the water from two valleys meet, build a dam (to prevent the merging), or watershed lines.
  - These watershed lines will then define the borders between different regions
- Three types of points
  - a) Points belonging to a regional minimum
  - b) Points at which a drop of water would fall with certainty to a single minimum
  - c) Points at which water would be equally likely to fall to more than one such minimum

**virtue:** gives small uniform regions

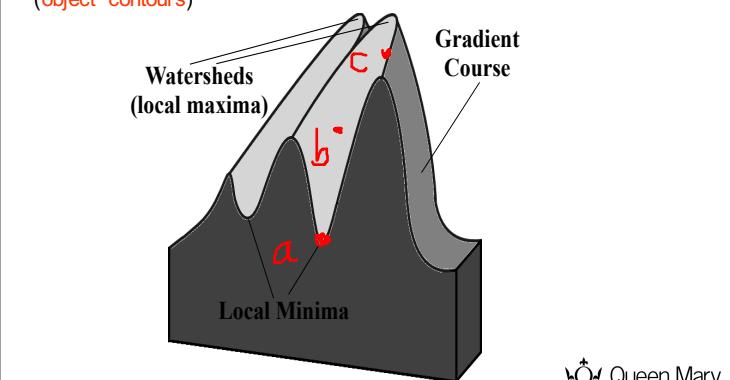
**'super pixel':** a group of pixels that have the same **property** (can be represented by one pixel)

EBU706U



## The watershed

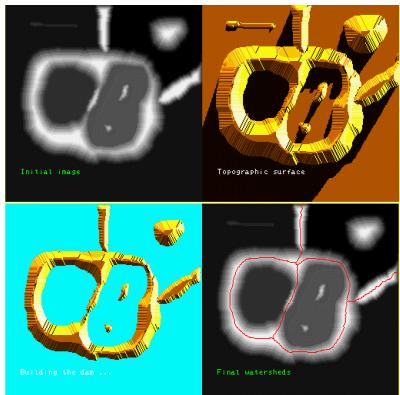
local maxima of the gradient → interpreted as watersheds  
(object contours)



EBU706U



## The watershed



EBU706U



## Agenda

- Image segmentation
- **Video segmentation**
- Segmentation quality

EBU706U



(a) (b)

(c) (d)

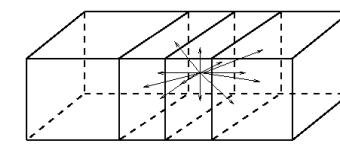
EBU706U Queen Mary University of London

Figure 5.51: Watershed segmentation: (a) original; (b) gradient image,  $3 \times 3$  Sobel edge detection, histogram equalized; (c) raw watershed segmentation; (d) watershed segmentation using region markers to control oversegmentation. Courtesy W. Higgins, Penn State University.

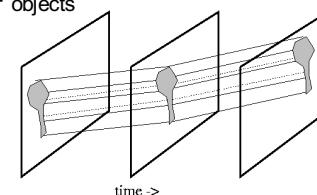
## Video segmentation

- Video segmentation
  - **Temporal** segmentation
    - Video  $\rightarrow$  shots
  - **Object/region** segmentation
    - Video  $\rightarrow$  regions and/or objects

breaking a video into shorter ones



time ->



time ->

EBU706U



## Temporal segmentation

- Temporal segmentation
    - Objective: to segment the video into basic units (**shots**)
    - Common shot transitions types
      - **Cuts** → abrupt shot change between two consecutive frames
      - **Dissolves** → first shot images get dimmer, while second ones get brighter, with frames superimposed
      - **Wipes** → Image of the second shot replaces the first one in a regular pattern, such as vertical line
      - **Fade-in /Fade-out** hard to segment – use activity identification
    - Algorithms
      - Colour-based
      - Edge-based
      - Motion-based
- optimal thresholding

EBU706U



## Object/region segmentation

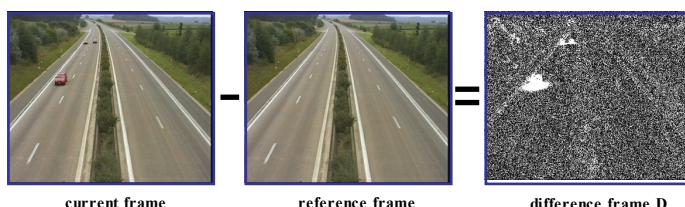
- Object/region segmentation
  - Objective: to segment video into spatio-temporal regions/objects
  - Ideally: automatic segmentation
- Example: **change detection**
  - classification of the pixels in the video sequence into two classes:
    - **foreground** pixels
    - **background** pixels
  - assumption: moving objects generate significant temporal changes
  - approach: evaluation of temporal differences between frames
  - limitations: camera noise, local and global illumination changes

EBU706U



## "Chroma-keying" without blue screen

- Background subtraction



- Problem

$D = \{d_k\}$ ,  $d_k \neq 0$  even if there is no structural change in k

EBU706U



## Motion segmentation

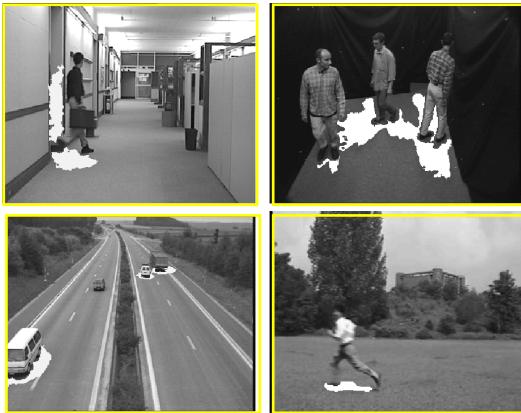
- Objective
  - Extraction of rough object masks from **motion fields**
- A diagram showing a grid of small arrows representing motion vectors. Some vectors point in various directions, while others are grouped together, forming small polygons. This represents the extraction of rough object masks from motion fields.
- Small isolated vector groups are removed by applying a median filter
- Neighboring positions corresponding to vectors of similar size are surrounded by a closed polygon

EBU706U



cue1: shadows move with object  
cue2: darker

### Shadow segmentation



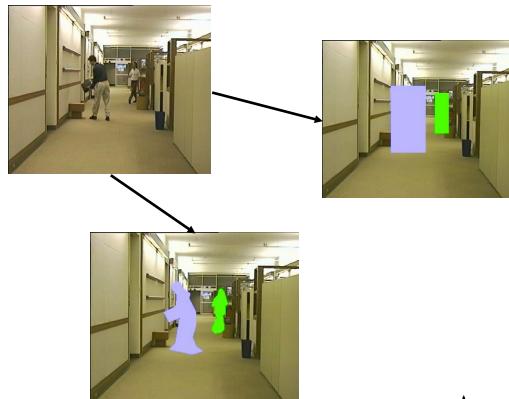
ELEM023

(people don't use it anymore...)

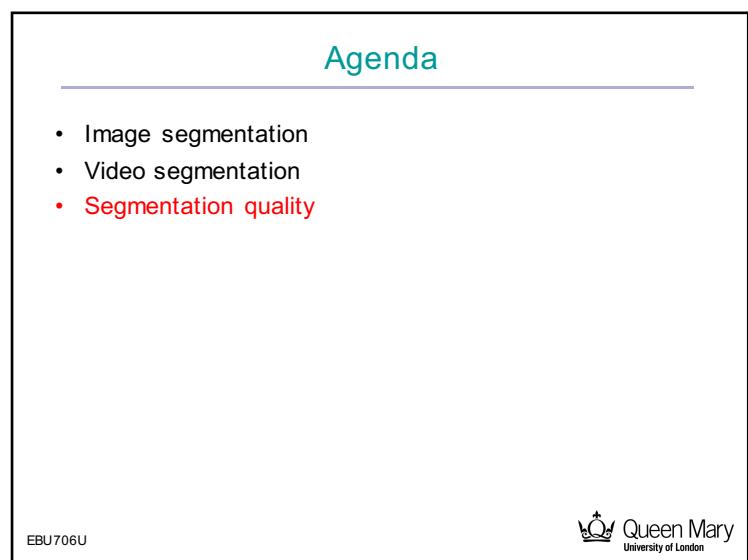
CORNERS are more likely to be seen  
on object rather than shadows



### Advanced applications



ELEM023



## Segmentation quality

- Assessment of the results of a segmentation algorithm
  - to compare different segmentation techniques
  - to improve one technique
- Requirements
  - Spatial accuracy
  - Temporal coherence
- Assessment methodologies
  - **Subjective** deals with semantics segmentation
    - time consuming
    - large number of subjects
  - **Objective**
    - automatic can only deal with low-level segmentation
    - with/without reference

EBU706U



## Evaluation metrics

- Analytical methods
  - principles
  - requirements
  - complexity of algorithms
- Empirical goodness methods
  - **desirable properties** of object segmentation results
- Empirical discrepancy methods
  - **deviation** from ground-truth segmentation done by humans
- Multi-metric methods
  - for interactive extraction tools

EBU706U



## Agenda

- Image segmentation
- Video segmentation
- Segmentation quality

EBU706U



## Question ...

Which of the following statements describes best the difference between frame-based multimedia (FBM) and object-based multimedia (OBM)?

- A. FBM encodes each video frame separately (intra-frame coding); OBM encodes each object separately.
- B. FBM encodes entire video frames (using both intra and inter-frame coding); OBM encodes video objects which have been extracted from other videos.
- C. FBM encodes each video frame separately (intra-frame coding); OBM encodes video objects which have been extracted from other videos.
- D. FBM encodes entire video frames (using both intra and inter-frame coding); OBM encodes objects of various natures and origins.

EBU706U



D?

### Question ...

Which of the following attributes does not apply to MPEG4?

- A. Multiplex
- B. Coding
- C. Segmentation
- D. Composition

C

EBU706U



### Question ...

Can a variant of the chroma-keying technique be used on a grayscale image?

- A. No, we need colours.
- B. No, there is not enough image data.
- C. Yes, we can use a luminance key.
- D. Yes, but we have to convert the image to RGB first.

C



### Question ...

Which of the following statements describes best edge-based image segmentation?

- A. It is looking for homogenous groups of pixels in an image.
- B. It is looking for semantically meaningful groups of pixels.
- C. It is looking for discontinuities in image data.
- D. It does all of the above.

D

EBU706U



### Question ...

One of the main difficulties in the bottom-up image segmentation approach is:

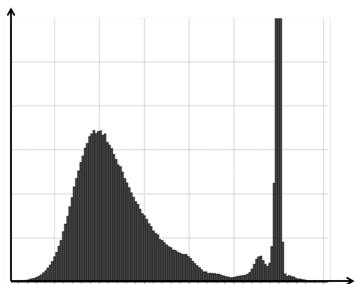
- A. Knowing when to stop segmenting.
- B. Selecting appropriate pixels (the seeds) to start segmenting.
- C. Converting the image data into a 3D landscape view.
- D. Selecting a colour key.

B

EBU706U



### Question ...

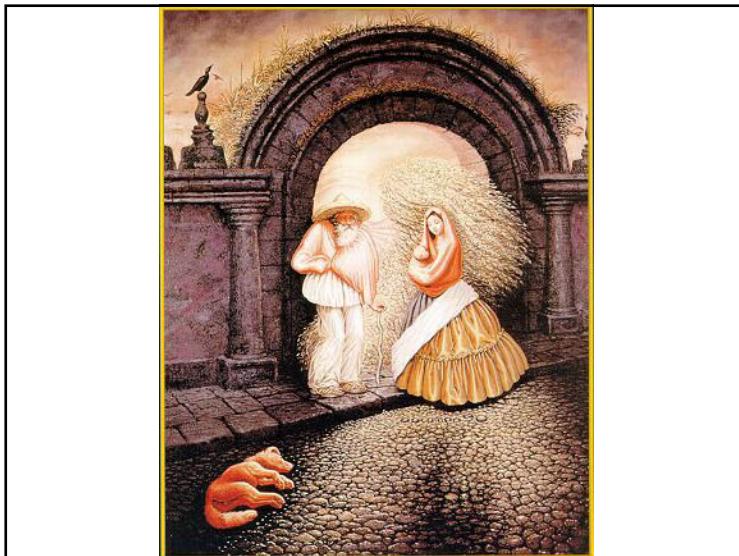


Explain how you would use this histogram to separate the pixels into background and foreground. In particular, discuss the use of different thresholds.

EBU706U



用两个高斯函数拟合，取他们平均值



## Gestalt psychology or gestaltism

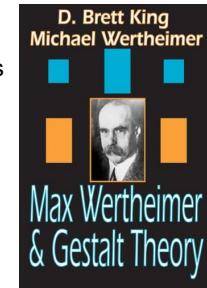
German: *Gestalt* - "form" or "whole"

Berlin School, early 20th century

Kurt Koffka, Max Wertheimer, and Wolfgang Köhler

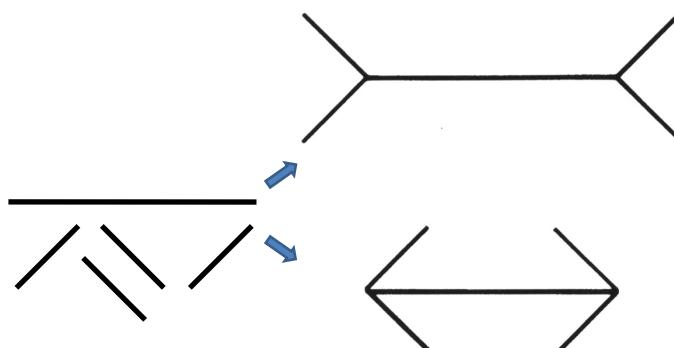
View of brain:

- whole is more than the sum of its parts
- holistic
- parallel
- analog
- self-organizing tendencies



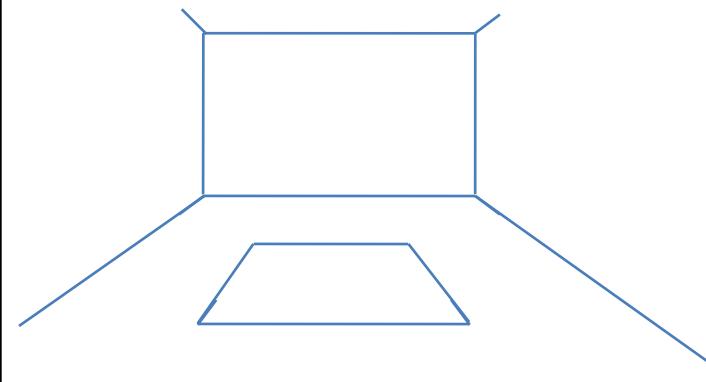
Slide from S. Saverese

### Gestaltism

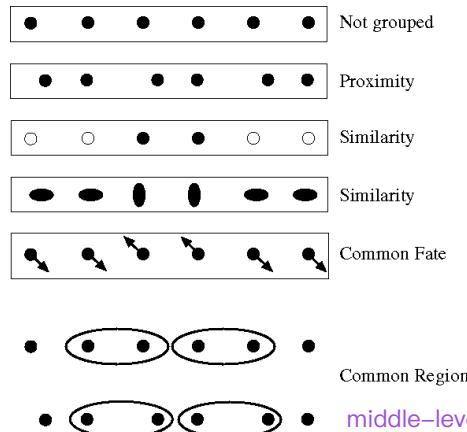


The Müller-Lyer illusion

### Explanation

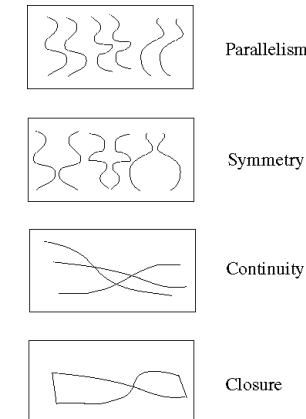


## Principles of perceptual organization

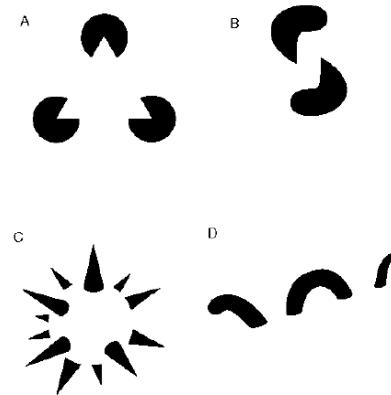


From Steve Lehar: The Constructive Aspect of Visual Perception

## Principles of perceptual organization

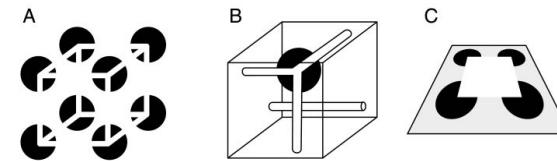


## Grouping by invisible completion



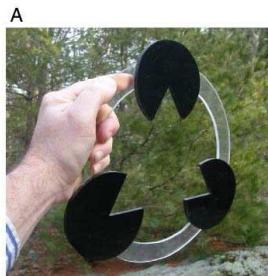
From Steve Lehar: The Constructive Aspect of Visual Perception

## Grouping involves global interpretation



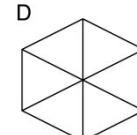
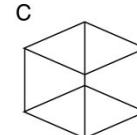
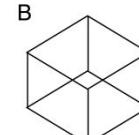
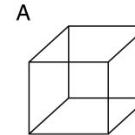
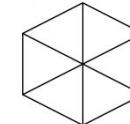
From Steve Lehar: The Constructive Aspect of Visual Perception

Grouping involves global interpretation



From Steve Lehar: The Constructive Aspect of Visual Perception

Gestaltists do not believe in coincidence



Emergence



Gestalt cues

- Good intuition and basic principles for grouping
- Difficult to implement in practice
- Sometimes used for occlusion reasoning

colour based segmentation  
Similarity; Proximity

### Moving on to image segmentation ...

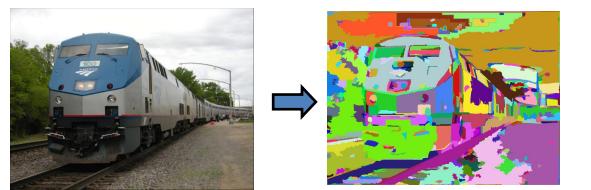
Goal: Break up the image into meaningful or perceptually similar regions



### Segmentation for feature support



### Segmentation for efficiency



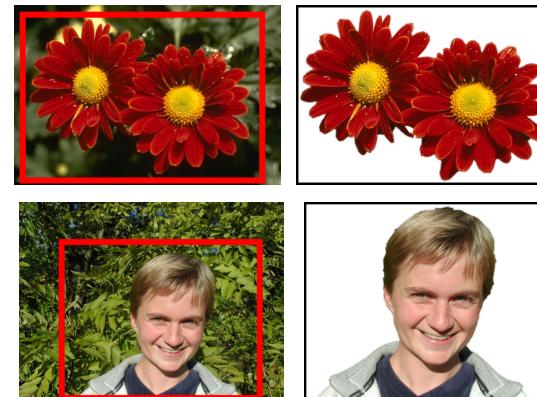
[Felzenszwalb and Huttenlocher 2004]



[Shi and Malik 2001]

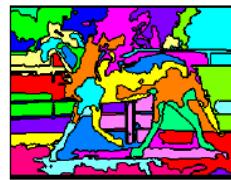
[Hoiem et al. 2005, Mori 2005]

### Segmentation as a result

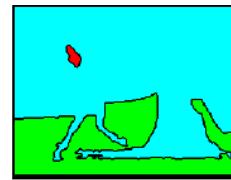


Rother et al. 2004

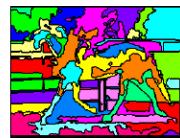
### Types of segmentations



Oversegmentation



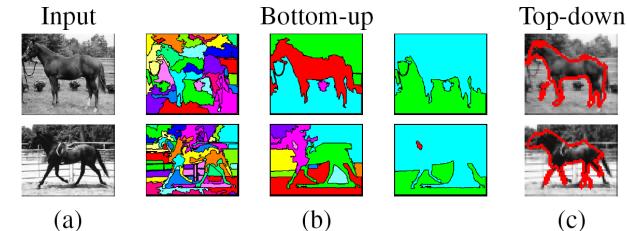
Undersegmentation



Multiple Segmentations

### Major processes for segmentation

- Bottom-up: group tokens with similar features
- Top-down: group tokens that likely belong to the same object

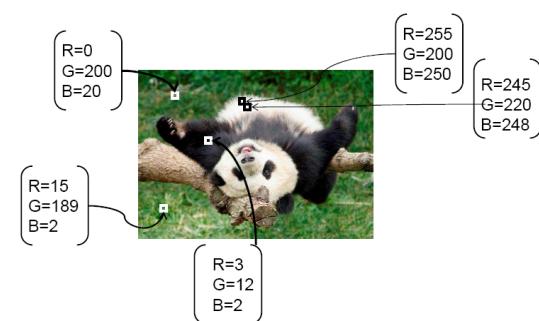


[Levin and Weiss 2006]

### Segmentation using clustering

- Kmeans
- Mean-shift

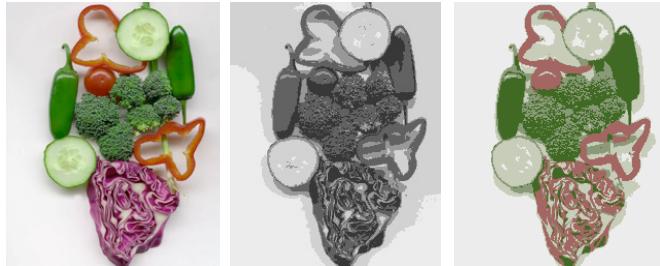
### Feature Space



Source: K. Grauman

### K-means clustering using intensity alone and color alone

Image      Clusters on intensity      Clusters on color

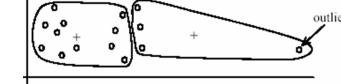
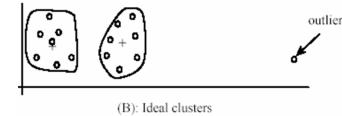


### K-Means pros and cons

- Pros
  - Simple and fast
  - Easy to implement

- Cons
  - Need to choose K
  - **Sensitive to outliers**

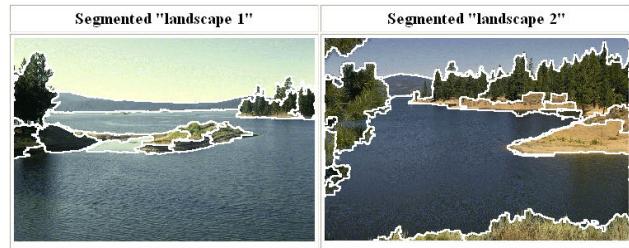
- Usage
  - Rarely used for pixel segmentation



### Mean shift segmentation

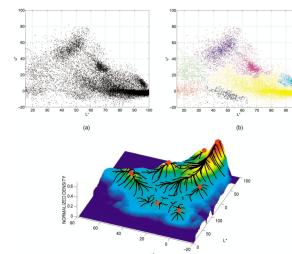
D. Comaniciu and P. Meer, Mean Shift: A Robust Approach toward Feature Space Analysis, PAMI 2002.

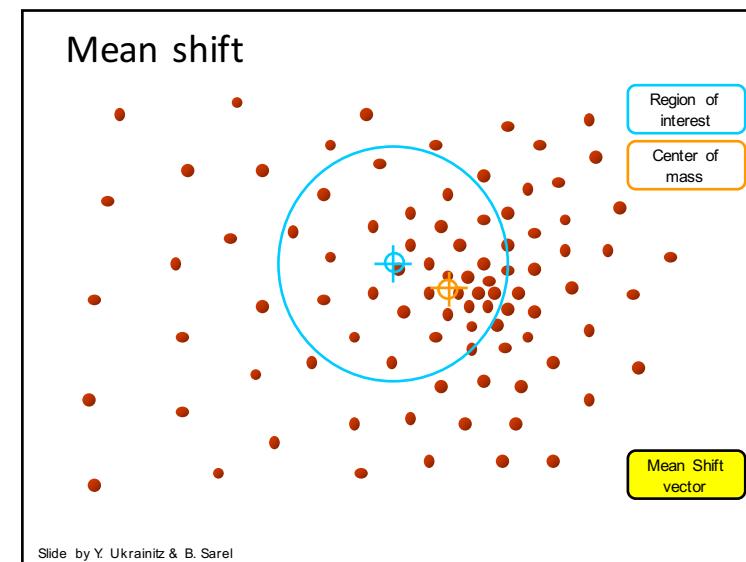
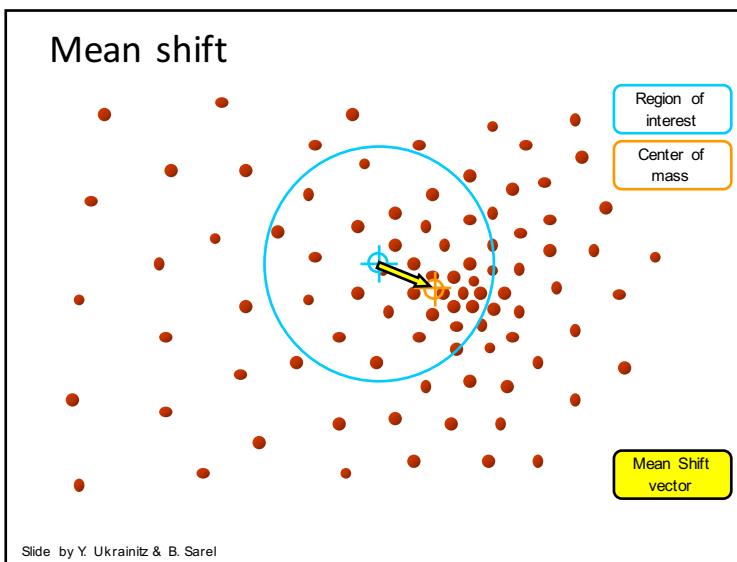
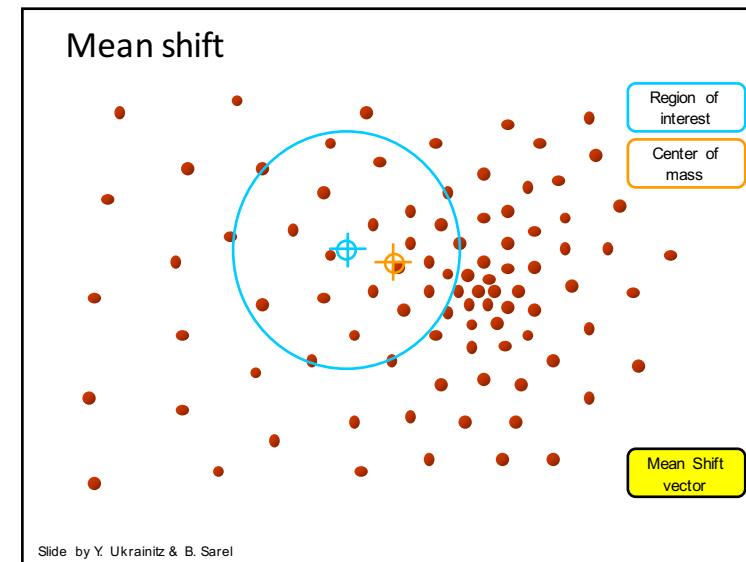
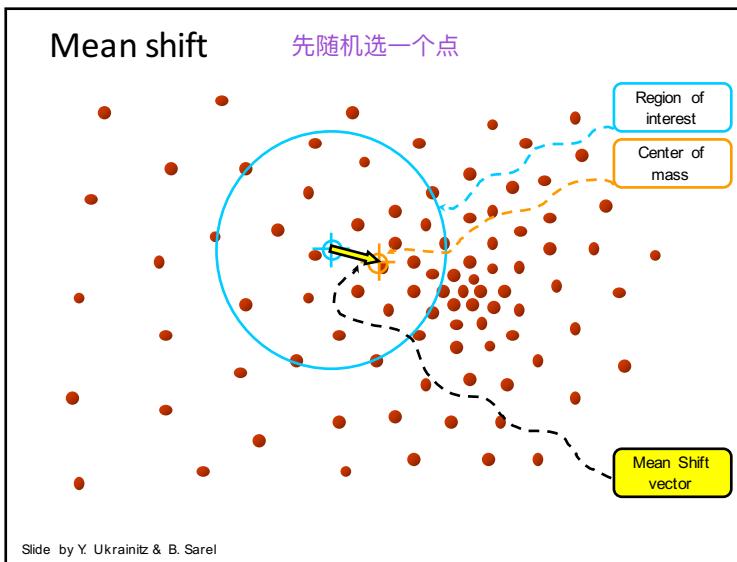
- Versatile technique for clustering-based segmentation



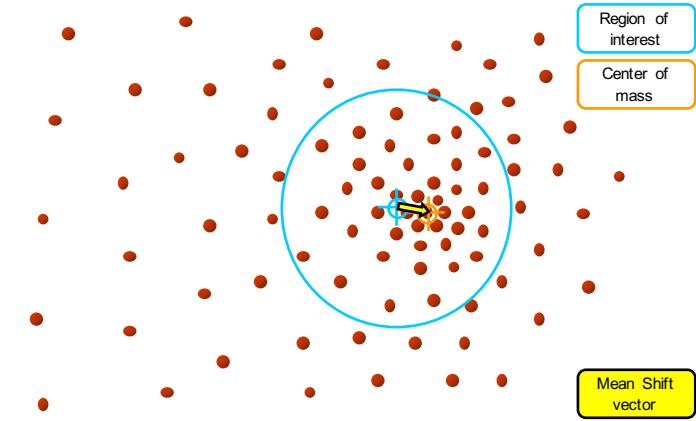
### Mean shift algorithm

- Try to find *modes* of this non-parametric density



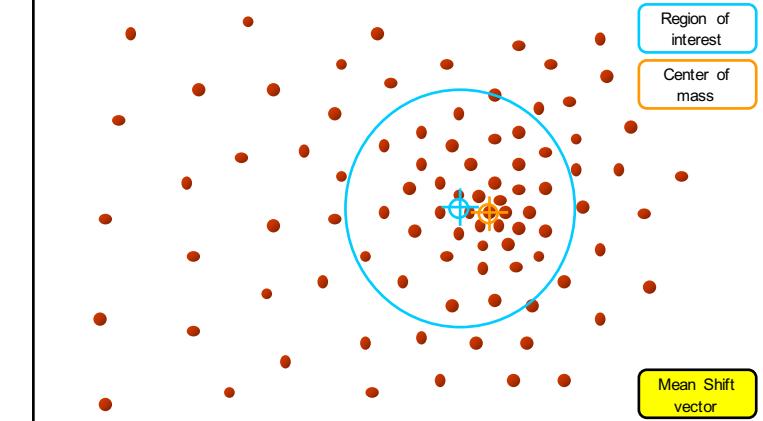


### Mean shift



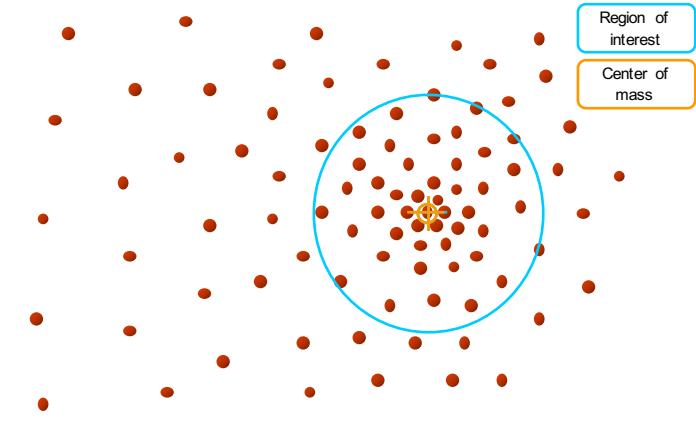
Slide by Y. Ukrainitz &amp; B. Sarel

### Mean shift



Slide by Y. Ukrainitz &amp; B. Sarel

### Mean shift



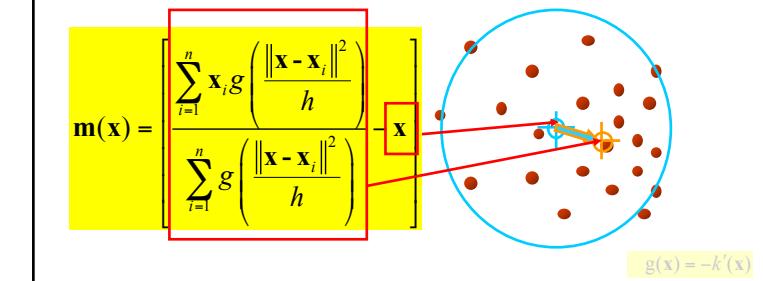
Slide by Y. Ukrainitz &amp; B. Sarel

### Computing the Mean Shift

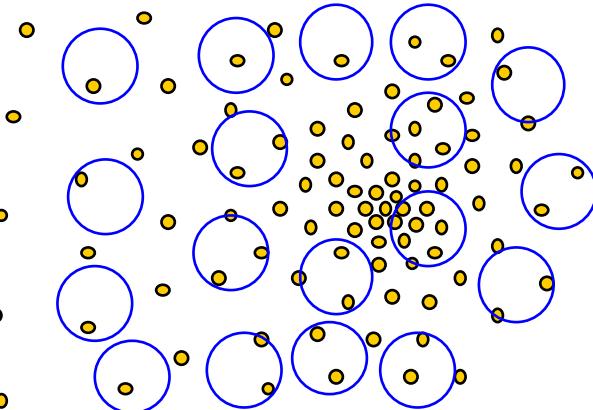
#### Simple Mean Shift procedure:

- Compute mean shift vector
- Translate the Kernel window by  $\mathbf{m}(\mathbf{x})$

$$\mathbf{m}(\mathbf{x}) = \left[ \frac{\sum_{i=1}^n \mathbf{x}_i g\left(\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{h}\right)}{\sum_{i=1}^n g\left(\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{h}\right)} \right]$$

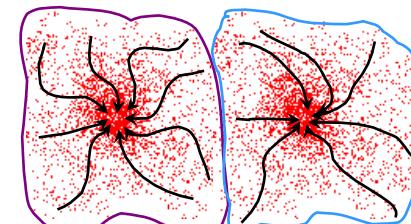


### Real Modality Analysis



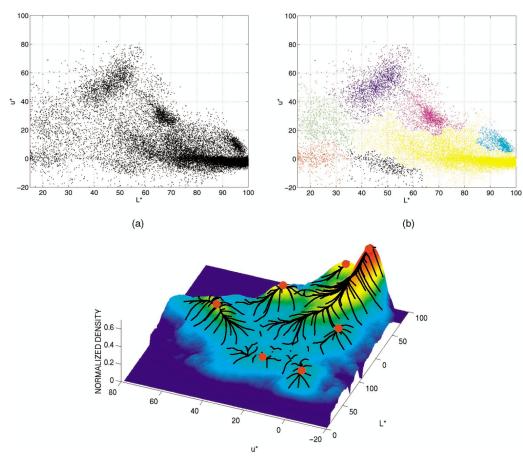
### Attraction basin

- **Attraction basin:** the region for which all trajectories lead to the same mode
- **Cluster:** all data points in the attraction basin of a mode



Slide by Y. Ukrainitz & B. Sarel

### Attraction basin

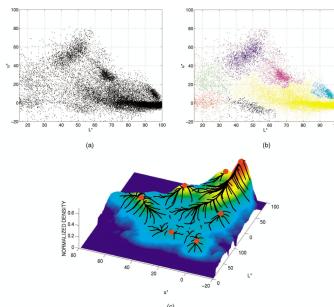


### Mean shift clustering

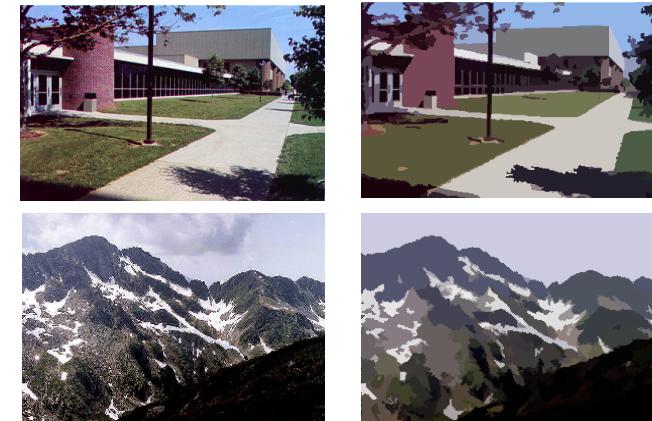
- The mean shift algorithm seeks *modes* of the given set of points
  1. Choose kernel and bandwidth
  2. For each point:
    - a) Center a window on that point
    - b) Compute the mean of the data in the search window
    - c) Center the search window at the new mean location
    - d) Repeat (b,c) until convergence
  3. Assign points that lead to nearby modes to the same cluster

## Segmentation by Mean Shift

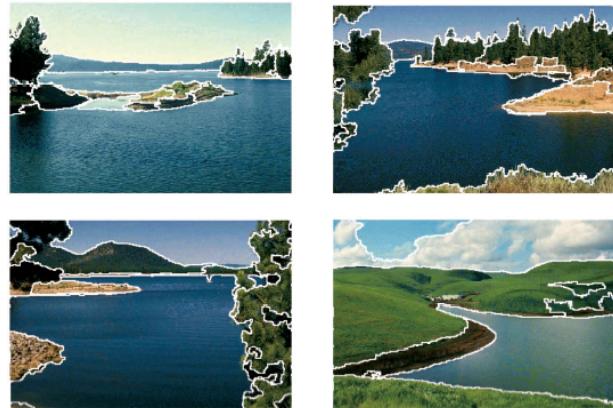
- Find features (color, gradients, texture, etc)
- Set kernel size for features  $K_f$  and position  $K_s$
- Initialize windows at individual pixel locations
- Perform mean shift for each window until convergence
- Merge windows that are within width of  $K_f$  and  $K_s$



## Mean shift segmentation results



<http://www.caip.rutgers.edu/~comanici/MSPAMI/msPamiResults.html>



<http://www.caip.rutgers.edu/~comanici/MSPAMI/msPamiResults.html>

## Mean-shift: other issues

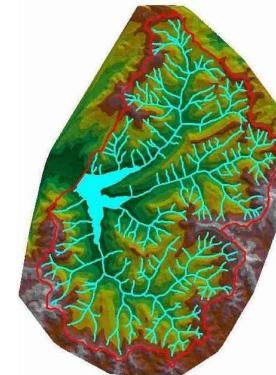
- Speedups
  - Uniform kernel (much faster but not as good)
  - Binning or hierarchical methods
  - Approximate nearest neighbor search
- Methods to adapt kernel size depending on data density
- Lots of theoretical support

D. Comaniciu and P. Meer, Mean Shift: A Robust Approach toward Feature Space Analysis, PAMI 2002.

## Mean shift pros and cons

- Pros
  - Good general-practice segmentation
  - Finds variable number of regions
  - Robust to outliers
- Cons
  - Have to choose kernel size in advance
  - Original algorithm doesn't deal well with high dimensions
- When to use it
  - Oversegmentation
  - Multiple segmentations
  - Other tracking and clustering applications

## Watershed algorithm



## Watershed segmentation



## Meyer's watershed segmentation

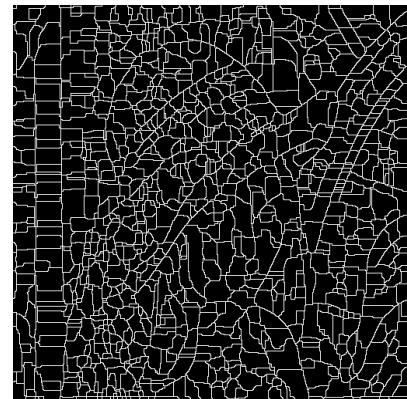
1. Choose local minima as region seeds
2. Add neighbors to priority queue, sorted by value
3. Take top priority pixel from queue
  1. If all labeled neighbors have same label, assign to pixel
  2. Add all non-marked neighbors
4. Repeat step 3 until finished

Matlab: `seg = watershed(bnd_im)`

Meyer 1991

## Simple trick

- Use median filter to reduce number of regions



## Watershed usage

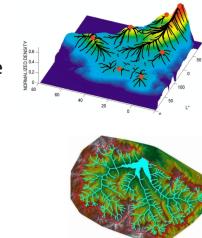
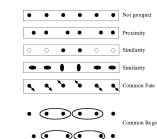
- Use as a starting point for hierarchical segmentation
  - Ultrametric contour map (Arbelaez 2006)
- Works with any soft boundaries
  - Pb
  - Canny
  - Etc.

## Watershed pros and cons

- Pros
  - Fast (< 1 sec for 512x512 image)
  - Among best methods for hierarchical segmentation
- Cons
  - Only as good as the soft boundaries
  - Not easy to get variety of regions for multiple segmentations
  - No top-down information
- Usage
  - Preferred algorithm for hierarchical segmentation

## Things to remember

- Gestalt cues and principles of organization
- Uses of segmentation
  - Efficiency
  - Better features
  - Want the segmented object
- Mean-shift segmentation
  - Good general-purpose segmentation method
  - Generally useful clustering, tracking technique
- Watershed segmentation
  - Good for hierarchical segmentation
  - Use in combination with boundary prediction



## Further reading

- Mean-shift paper by Comaniciu and Meer  
<http://www.caip.rutgers.edu/~comanici/Papers/MsRobustApproach.pdf>
- Adaptive mean shift in higher dimensions  
<http://mis.hevra.haifa.ac.il/~ishimshoni/papers/chap9.pdf>
- Contours to regions (watershed): Arbelaez et al. 2009  
[http://www.eecs.berkeley.edu/~arbelaez/publications/Arbelaez\\_Mairal\\_Fowlkes\\_Malik\\_CVPR2009.pdf](http://www.eecs.berkeley.edu/~arbelaez/publications/Arbelaez_Mairal_Fowlkes_Malik_CVPR2009.pdf)

## If time...

- Graph-based segmentation
  - Normalized cuts
  - Graph cuts

