Graphics & Multimedia

School of Information Technologies
The University of Sydney

Week04 • Semester 2 • 2018

Video Data Representation & Processing

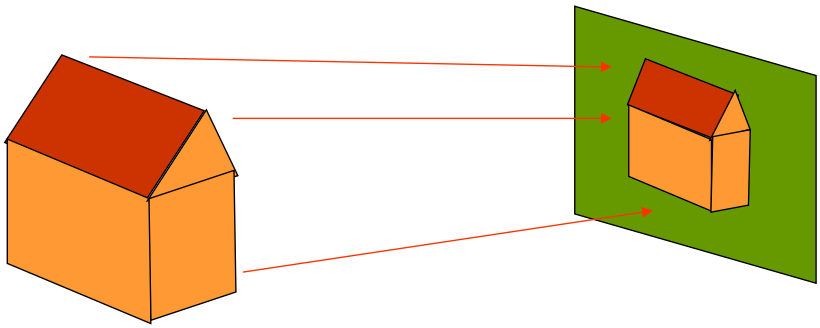
- Video color systems
- Video digitization
- Spatial / temporal sampling
- Digital video standards
- Video hierarchical structure
- Video segmentation
- Motion estimation
- Video object extraction
- Appendix:
  - Optical flow computation
  - Reading: *Estimating Motion in Image Sequences*

Video

- There are two different ways of generating moving pictures in a digital form for inclusion in a multimedia production.
- **Video** – we can use a video camera to capture a sequence of frames recording actual motion as it is occurring in the real world;
- Animation – we can create each frame individually, either within the computer or by capturing single images one at a time.

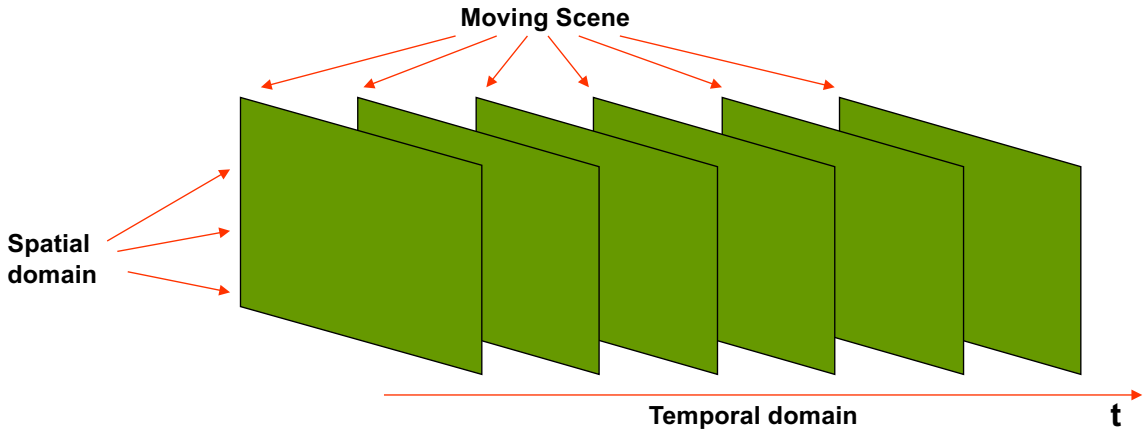
Video Image

- A **video image** is a projection of a 3D scene onto a 2D plane.
  - A 3D scene consisting of a number of objects each with depth, texture and illumination is projected onto a plane to form a 2D representation of the scene.
  - The 2D representation contains varying texture and illumination but no depth information.
- A still image is a "snapshot" of the 2D representation at a particular instant in time whereas a video sequence represents the scene over a period of time.



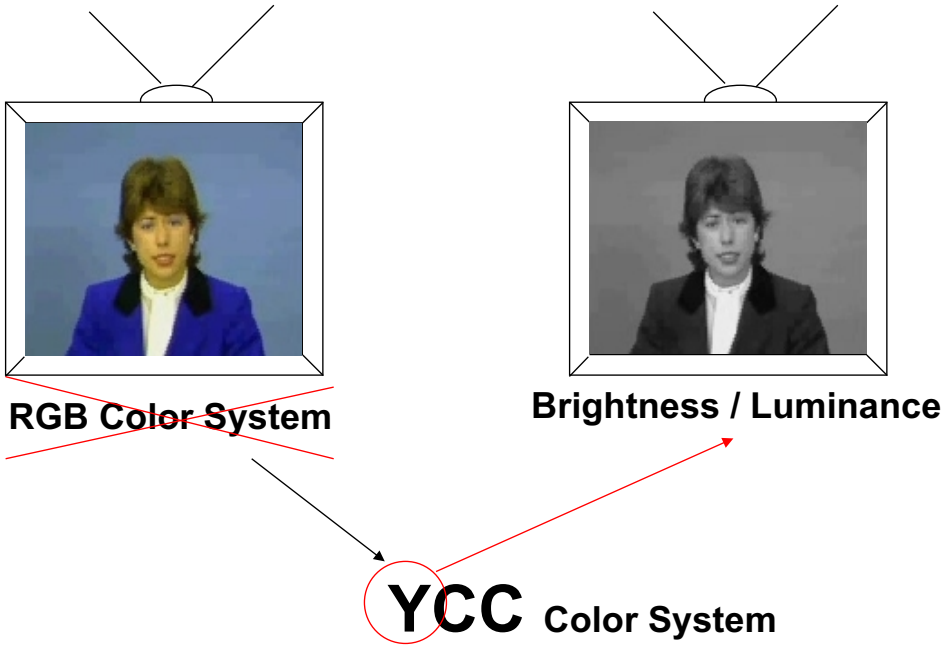
Video Sequence

- A "real" visual scene is continuous both spatially and temporally.



Video Color Systems

- Largely derive from older analog methods of coding color for TV
- Luminance is separated from color information

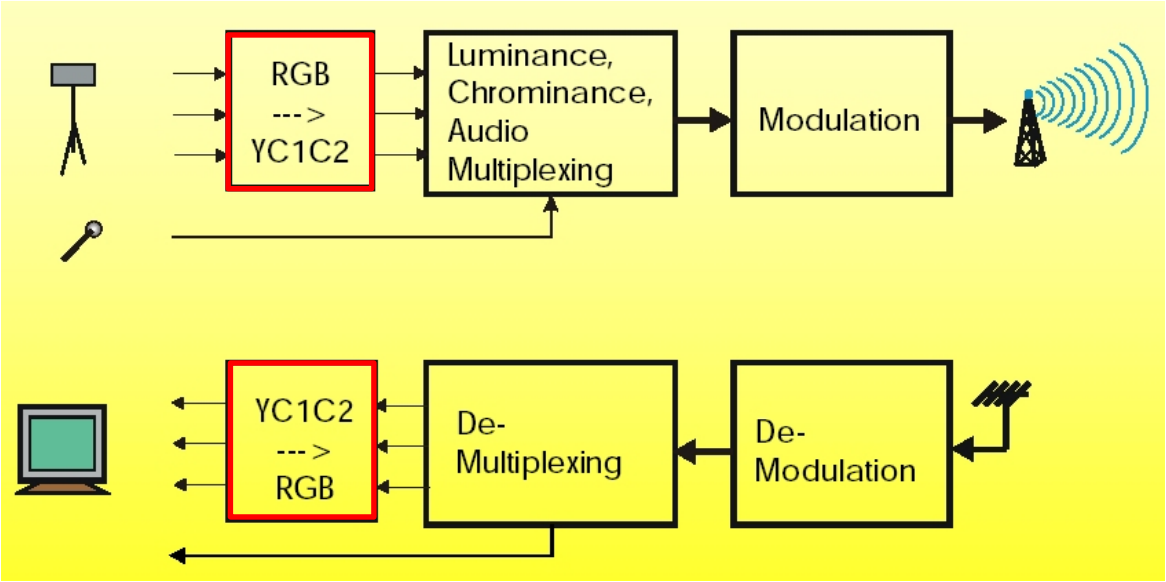


Video Color Systems

- Televisions display video using the RGB system based upon three types of phosphor that emit light in the red, green, and blue regions of the spectrum. However, unlike computers, television / video signals are not transmitted or stored in RGB. Why not?
  - When television was first invented, it worked only in black and white. The term “black and white” is actually something of a misnomer, because what you really see are the shades of grey between black and white. That means that the only piece of information being sent is the brightness (known as the luminance) for each dot on the TV screen.
  - When **color television** was being developed, it was imperative that color broadcasts could be viewed on black and white televisions, so that millions of people didn’t have to throw out the sets they already owned. Rather, there could be a gradual transition to the new technology. Therefore, instead of transmitting the new color broadcasts in RGB, they were (and still are) transmitted in something called **Y.C.C.** scheme.

Video Color Systems

- Largely derive from older analog methods of coding color for TV
- Luminance is separated from color information



Video Color Systems

- In **Y.C.C.** scheme, the "**Y**" was the same old luminance (brightness) signal that was used by black and white televisions, while the "**C**'s" stood for the Color components.
- The two color components would determine the hue of a pixel, while the luminance signal would determine its brightness. Thus, color transmission was facilitated while black and white compatibility was maintained.

- There are numerous video color systems :
  - **YIQ** color system (NTSC)
  - **YUV** color system (PAL)
  - **YDrDb** color system (SECAM)

YIQ Color System

- In the United States, the National Television Systems Committee (NTSC)* defined a video color system called **YIQ** for transmission.
- The Y component captures the luminance (brightness) information that can be compatibly decoded by a black-and-white television. Supplementing the Y component are two chrominance (color) components, I and Q. The I component captures flesh tones and near flesh tones (orange-to-cyan), and the Q component captures other colors (green-to-purple, on which human eye is less sensitive).
- To simplify the color decoding of YIQ to RGB, the YIQ components are defined to be linearly related to the RGB components:
  - $Y = 0.299R + 0.587G + 0.114B$
  - $I = 0.596R - 0.274G - 0.322B$
  - $Q = 0.211R - 0.523G + 0.896B$

* The NTSC is also used in North America, Japan, Korea, Mexico, Taiwan and parts of the Caribbean.

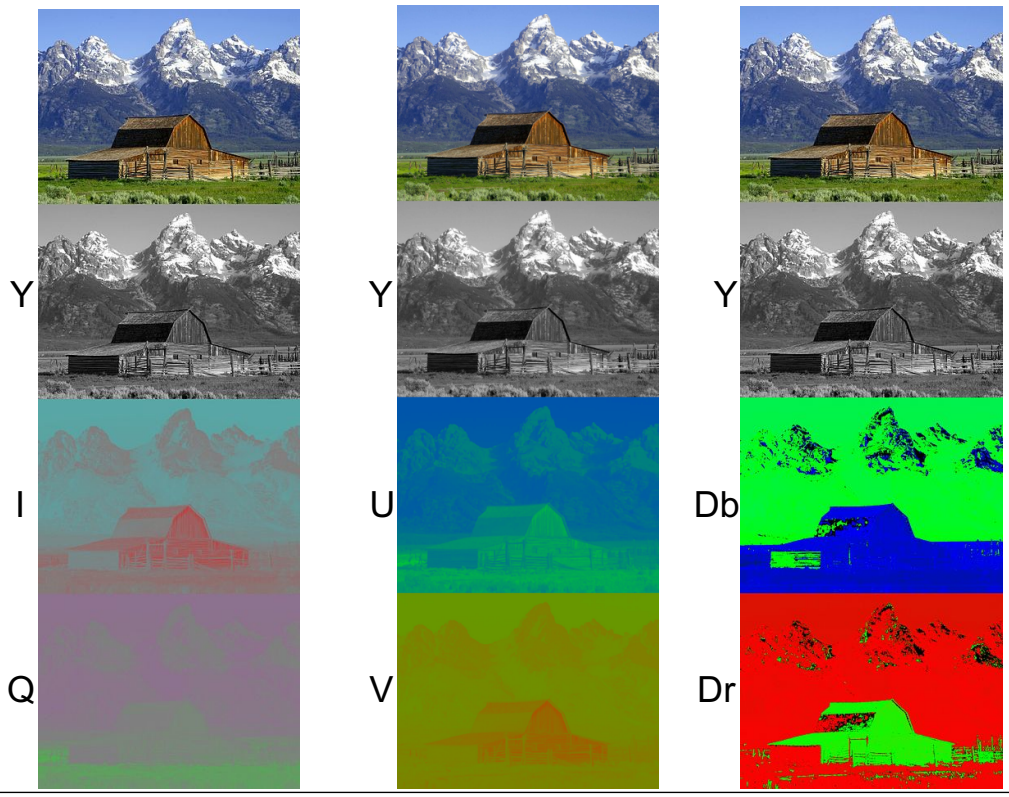
YUV Color System

- In most of Western Europe, Australia, New Zealand, China and South America, the PAL (Phase Alternating Line) television system is widely used.
- The PAL system uses a video color system called **YUV**, in which the Y component is identical to that of YIQ. As with YIQ, the YUV components are linearly related to the RGB components.
  - $Y = 0.299R + 0.587G + 0.114B$
  - $U = -0.147R - 0.289G + 0.436B = 0.492\,(B - Y)$
  - $V = 0.615R - 0.515G - 0.100B = 0.877\,(R - Y)$

YDrDb Color System

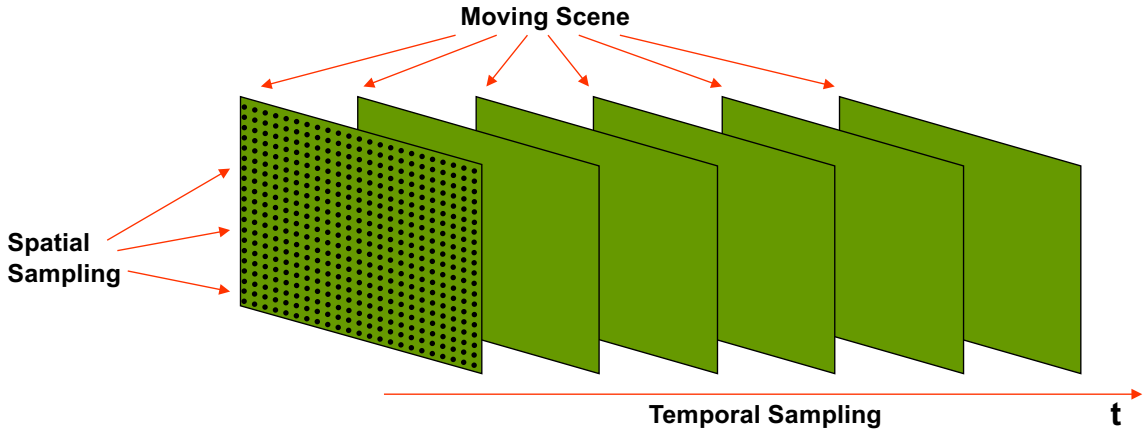
- In France, Eastern Europe, Middle East and much of Africa, the SECAM (SEquential Couleur Avec Memoire) television system is widely used.
- The SECAM system uses a video color system called YDrDb, in which the Y component is identical to that of YIQ. As with YIQ, the YDrDb components are linearly related to the RGB components.
  - $Y = 0.299R + 0.587G + 0.114B$
  - $Dr = -0.450R - 0.883G + 1.333B = 3.059U$
  - $Db = -1.333R + 1.116G - 0.217B = -2.169V$

Video Color Systems (YIQ / YUV / YDrDb)



Digital Video Sequence

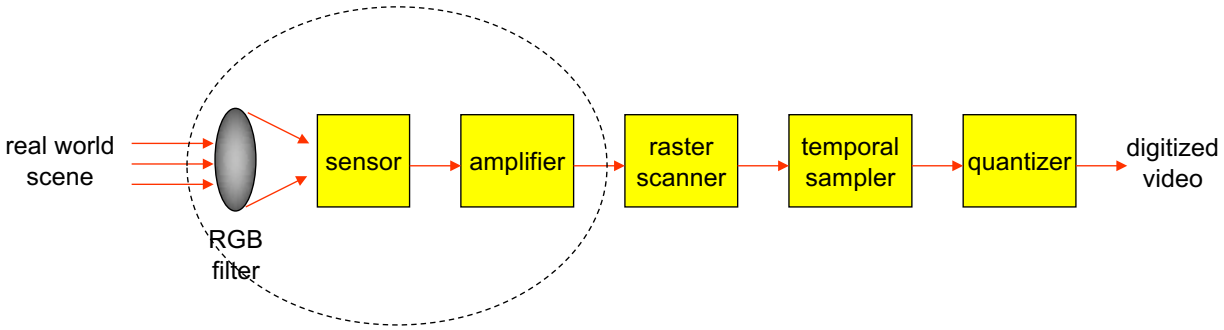
- A “real” visual scene is continuous both spatially and temporally.
  - In order to represent and process a visual scene digitally, it is necessary to sample the real scene spatially (typically on a rectangular grid in the video image plane) and temporally (typically as a series of “still” images or frames sampled at regular intervals in time).
- **Digital video** is the representation of a spatio-temporally sampled video scene in digital form.



Digitization

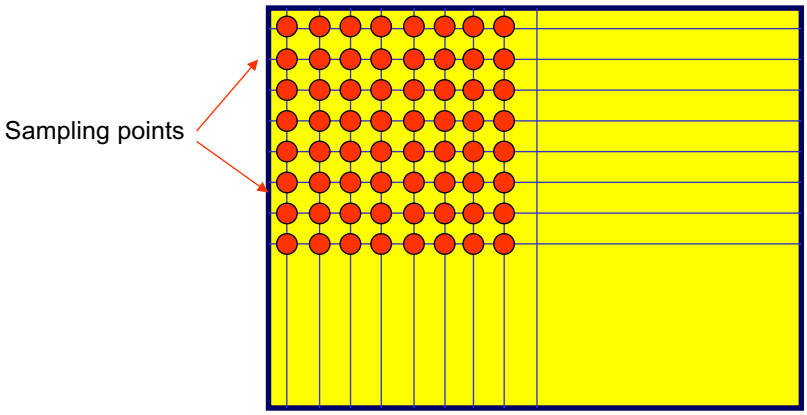
- The typical processing steps involved in the digitization of video.

  After signal acquisition and amplification, the key processing steps are spatial sampling, temporal sampling, and quantization.

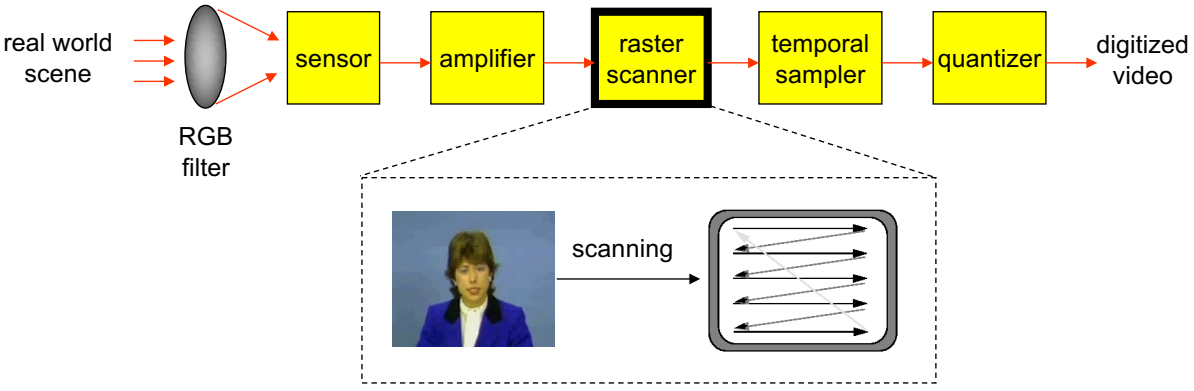


Spatial Sampling

- Spatial sampling consists of taking measurements of the underlying analog signal at a finite set of sampling points in a finite viewing area (or frame).
- To simplify the process, the sampling points are restricted to lie on a lattice, usually a rectangular grid.



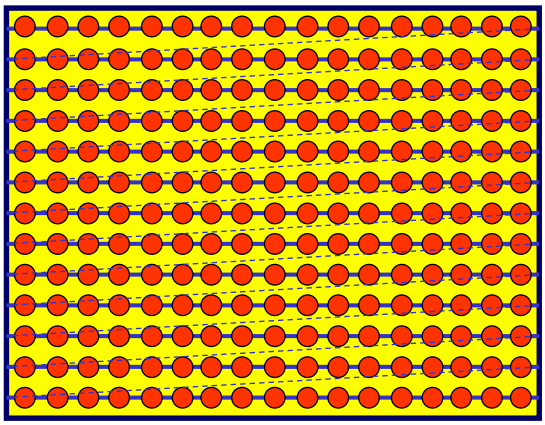
Raster Scanning



- The two-dimensional set of sampling points are transformed into a one-dimensional set through a process called raster scanning.
- The two main ways to perform raster scanning are progressive scanning and interlaced scanning.

Progressive Scanning

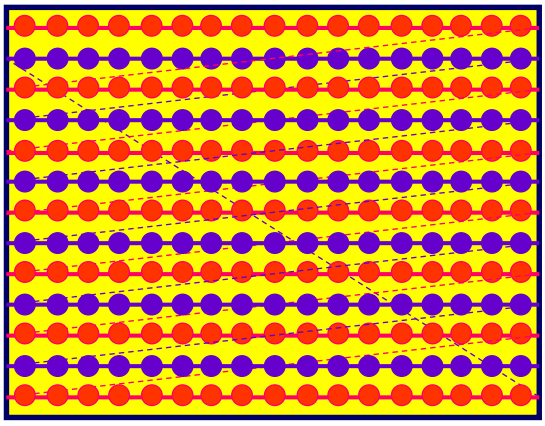
- In a progressive (or non-interlaced) scan, the sampling points are scanned from left to right and top to bottom.
- Progressive scanning is typically used for film and computer displays.



(dots x lines)

Interlaced Scanning

- In an interlaced scan, the points are divided into odd and even scan lines. The odd lines are scanned first from left to right and top to bottom. Then the even lines are scanned.
- The odd (respectively, even) scan lines make up a field. In an interlaced scan, two fields make up a frame.
- Interlaced scanning is commonly used for television signals.



(dots x lines)

- Upper (odd) field
- Lower (even) field

De-interlace Processing

- Interlaced video sometimes may produce unpleasant visual artifacts when displaying certain textures or types of motion.
- If you are capturing images from a video signal, you can filter them through a de-interlacing filter provided by image-editing applications.

Original
video image with
interlaced scanning

Improved
video image with
software de-interlacing

Improved
video image with
hardware de-interlacing

De-interlace Filter

- Example: GIMP software
  > Filters > Enhance > Deinterlace > Mode: Keep Odd / Even Fields

Original
video image with
interlaced scanning

Improved
video image with
de-interlace filter
(keep odd fields)

Improved
video image with
de-interlace filter
(keep even fields)

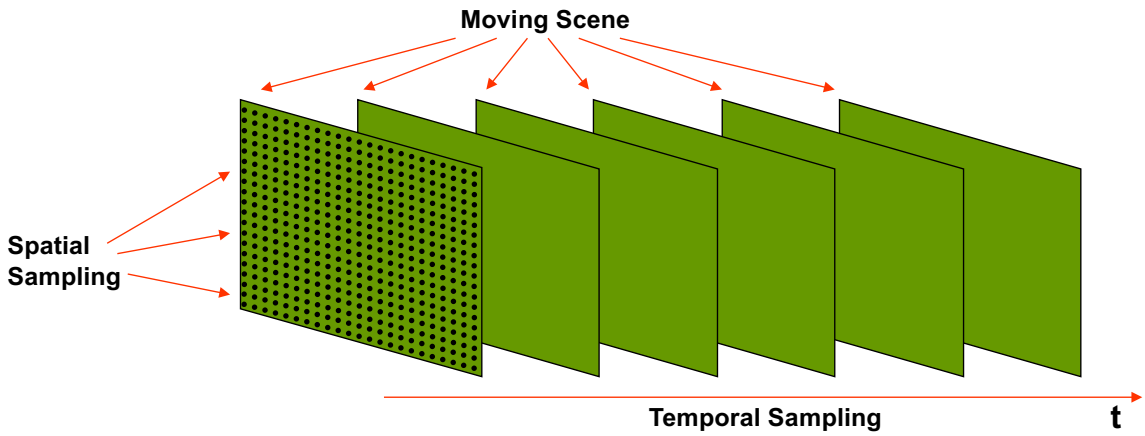
Video Image Resolutions

- The visual quality of the video image is influenced by the number of sampling points. More sampling points (a higher sampling resolution) give a “finer” representation of the image: however, more sampling points require higher storage capacity.

Typical video image resolutions

Video resolution (dots x lines)	Number of pixels (sampling points)	Standard
352 x 288	101,376	VHS video (Video CD)
480 x 576	276,480	Super Video CD (SVCD)
704 x 576	405,504	Standard-definition television (SDTV)
1280 x 720 progressive scan	921,600	High-definition television (HDTV): HD
1920 x 1080 split into two interlaced fields of 540 lines	2,073,600	High-definition television (HDTV): Full HDi
1920 x 1080 progressive scan	2,073,600	High-definition television (HDTV): Full HD
3840 x 2160 progressive scan	8,294,400	Ultra-high-definition television (UHDTV): 4K UHD
7680 x 4320 progressive scan	33,177,600	Ultra-high-definition television (UHDTV): 8K UHD

Temporal Sampling



Temporal Sampling

- A moving video image is formed by sampling the video signal temporally, taking a rectangular "snapshot" of the signal at periodic time intervals.
- The human visual system is relatively slow in responding to temporal changes. By showing at least 16 frames of video per second, an illusion of motion is created. This observation is the basis for motion picture technology, which typically performs temporal sampling at a rate of 24 frames / sec.

Video frame rate	Appearance
Below 10 frames/sec	"Jerky", unnatural appearance to movement
10 ~ 20 frames/sec	Slow movements appear OK; rapid movement is clearly "jerky"
20 ~ 30 frames/sec	Movement is reasonably smooth
50 ~ 60 frames/sec	Movement is very smooth

Why Digital Video ?

- Storing video on digital devices or in memory, ready to be processed (noise removal, cut and paste, and so on) and integrated into various multimedia applications
- Direct access, which makes nonlinear video editing simple
- Repeated recording without degradation of image quality
- Ease of encryption and better tolerance to channel noise

Digital Video Standards

- To promote the interchange of digital video data, several formats for representing video data have been standardized.
- Popular standards for representing digital video:
  - CCIR–601 standard
  - Source Input Format (SIF)
  - Common Intermediate Format (CIF)
  - Quarter–CIF (QCIF)

CCIR–601 Standard

- Since the YIQ, YUV, and YDrDb color systems are designed for analog television, these systems are inherently analog.
- The CCIR* (International Consultative Committee for Radio) Recommendation 601 (CCIR-601) digital video standard defines a standard digital representation of video in terms of digital **YCrCb** color components.

* **FYI**: The CCIR has changed its name to the International Telecommunication Union Radio-communication Assembly (ITU-R), and the latest revision of the CCIR-601 standard is formally known as Recommendation ITU-R BT.601-5.

CCIR–601 Standard

- CCIR-601 defines both 8-bit and 10-bit digital encodings. In the 8-bit encoding, assuming that the RGB components have been digitized to the range [0, 255], the YCrCb components are defined as follows:
  - $Y = 0.257R + 0.504G + 0.098B + 16$
  - $Cr = 0.439R - 0.368G - 0.071B + 128$
  - $Cb = -0.148R - 0.291G + 0.439B + 128$

CCIR–601 Standard

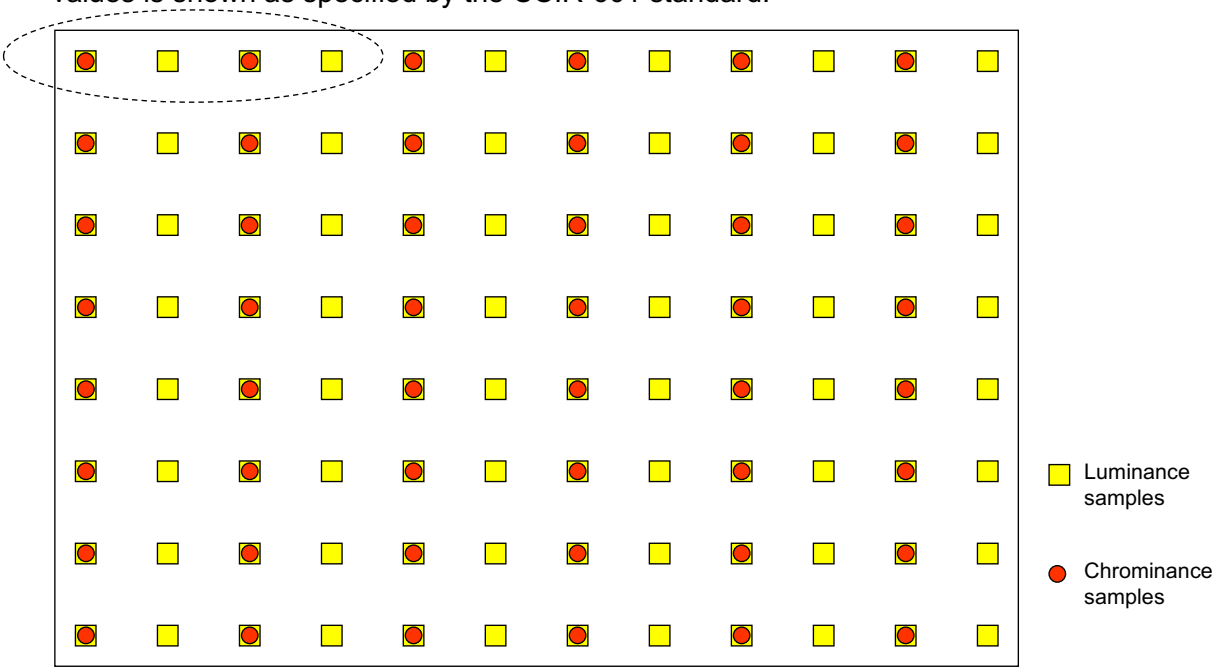
- The CCIR-601 standard defines a family of digital video formats. The most commonly used member of the family is the 4:2:2, 13.5 MHz format.
- In this format, the luminance component is sampled at a rate of 13.5MHz with 720 active samples per line. The chrominance components, Cr and Cb, each are sampled at 6.75 MHz with 360 active samples per line.
- For NTSC, this sampling yields 486 active lines per frame at 60 fields/sec.
- For PAL, the sampling yields 576 active lines per frame at 50 field/sec.

CCIR–601 Standard

- In term of pixels per frame, the 4:2:2 CCIR-601 format specifies spatial sampling of 720 x 486 for NTSC and 720 x 576 for PAL.
- Temporal sampling is interlaced 60 fields/sec for NTSC and interlaced 50 fields/sec for PAL.
- The chrominance components are subsampled horizontally with respect to the luminance component to take advantage of the human visual system's reduced spatial sensitivity to color. This subsampling process is referred to as the 4:2:2 format.

4:2:2 Color Subsampling

- With 4:2:2 chroma subsampling, the two chroma components are subsampled by a factor of two horizontally. The positioning of the chrominance values relative to the luminance values is shown as specified by the CCIR-601 standard.



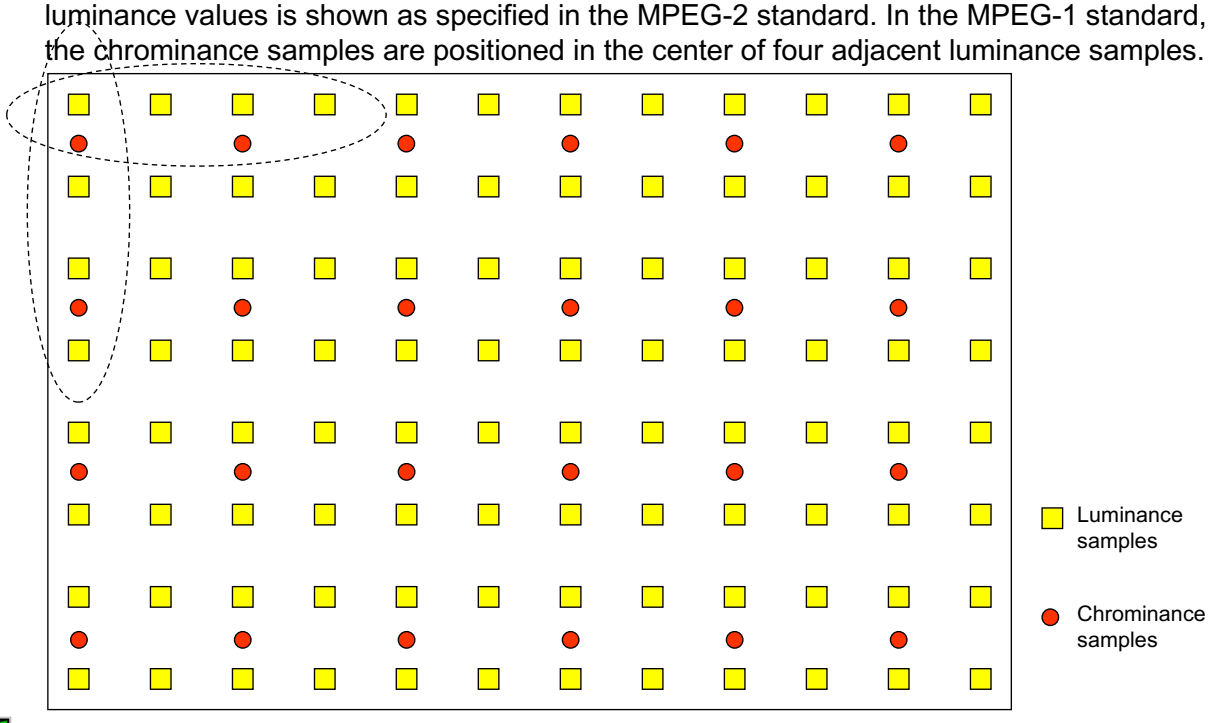
Source Input Format (SIF)

- The **Source Input Format (SIF)** specifies spatial sampling of 360 x 240 and progressive temporal sampling at 30 frames/sec for NTSC-originated video, and 360 x 288 spatial sampling at a progressive frame rate of 25 frames/sec for PAL-original video*.
- As with CCIR-601, color is represented using three components: Y, Cr, and Cb. Each component is quantized linearly using eight bits.
- The chrominance components, Cr and Cb, are subsampled by a factor of two both horizontally and vertically, yielding a chrominance sampling of 180 x 120 at 30 frames/sec and 180 x 144 at 25 frames/sec. This subsampling format is referred to as the 4:2:0 format.

* For some applications, such as MPEG-1 and MPEG-2 video, it is convenient for the spatial dimensions to be a multiple of 16, For this reason, a horizontal dimension of 352 is often used.

4:2:0 Color Subsampling

- With 4:2:0 subsampling, the chroma components are subsampled by a factor of two both horizontally and vertically. The positioning of the chrominance values relative to the luminance values is shown as specified in the MPEG-2 standard. In the MPEG-1 standard, the chrominance samples are positioned in the center of four adjacent luminance samples.



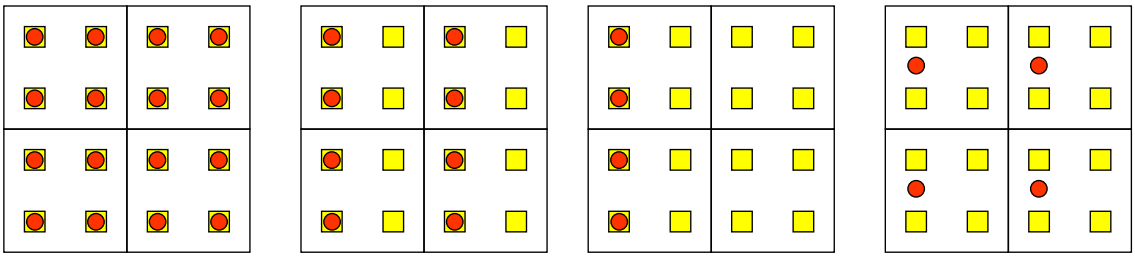
Common Intermediate Format (CIF)

- One drawback with the CCIR-601 and SIF formats is that they specify different spatial and temporal sampling parameters for NTSC and PAL systems.
- As its name suggests, the Common Intermediate Format (CIF) was proposed as a bridge between NTSC and PAL.
- As with CCIR-601, color is represented using YCrCb, quantized linearly using eight bits.
- The CIF format uses 4:2:0 color subsampling with an image size of 352 x 288.
- Temporal sampling is set at 30 frames/sec. For use with PAL systems, the CIF format requires conversion of the frame rate to 25 frames/sec. For NTSC systems, a spatial resampling may be necessary.

Quarter-CIF (QCIF)

- For video conferencing and other low-bit-rate, low-resolution applications, a scaled-down version of CIF called Quarter-CIF (QCIF) is commonly used.
- QCIF specifies an image with half the resolution of CIF in each spatial dimension: 176 x 144. For many low-bit-rate applications, the frame rate is reduced from 30 frames/sec to as low as 5 frames/sec.
- There is a non-standard "sub-QCIF" format of 128 x 96 for very low bandwidth applications. Whichever size is used, the video signal is chroma sub-sampled to 4:2:0.

Chrominance Subsampling Formats



4:4:4

For every 2 x 2 Y samples
4 Cr and 4 Cb sample
(No subsampling)

4:2:2

For every 2 x 2 Y samples
2 Cr and 2 Cb sample
(Subsampling by 2:1
horizontally only)

4:1:1

For every 4 x 1 Y samples
1 Cr and 1 Cb sample
(Subsampling by 4:1
horizontally only)

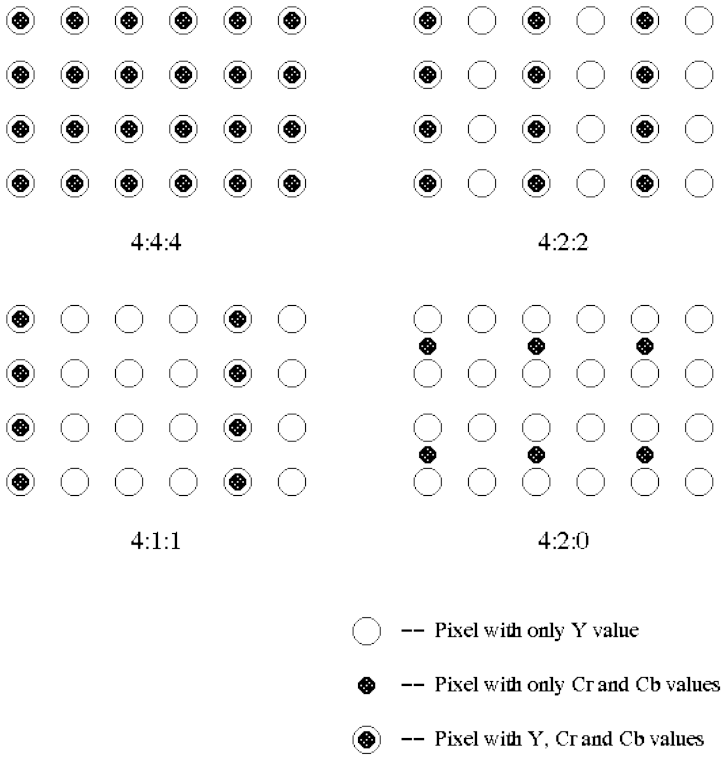
4:2:0

For every 2 x 2 Y samples
1 Cr and 1 Cb sample
(Subsampling by 2:1 both
horizontally and vertically)

Y: Luminance samples

Cr & Cb: Chrominance samples

Chrominance Subsampling Formats



The Human Visual System (HVS)

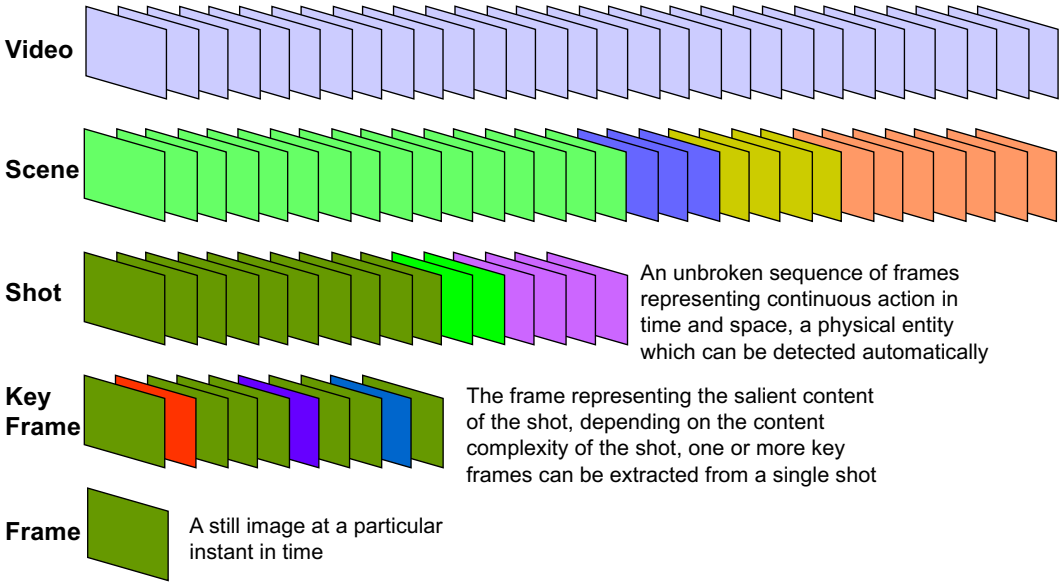
- A critical design goal for a digital video system is that the visual images produced by the system should be “pleasing” to the viewer.
- In order to achieve this goal it is necessary to take into account the response of the human visual system (HVS). The HVS is the “system” by which a human observer views, interprets and responds to visual stimuli.

The HVS and Digital Video Systems

- The operation of the HVS is a large and complex area of study. Some of the important features of the HVS have implications for digital video system design.

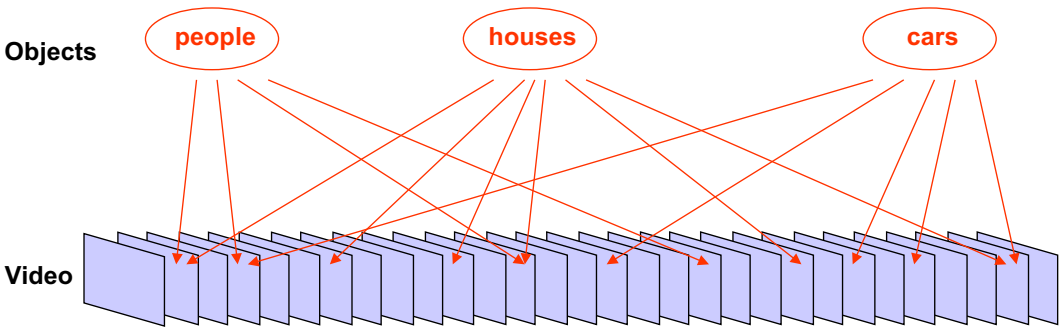
Feature of the HVS	Implication for digital video systems
The HVS is more sensitive to luminance detail than to color detail	Color resolution may be reduced without significantly affecting image quality
The HVS is more sensitive to high contrast (i.e. large differences in luminance) than low contrast	Large changes in luminance (e.g. edges in an image) are very important to the appearance of the image
The HVS is more sensitive to low spatial frequencies (i.e. changes in luminance that occur over a large area) than high spatial frequencies (rapid changes that occur in a small area)	It may be possible to compress video images by discarding some of the less important higher frequencies (however, edge information should be preserved)
The HVS is more sensitive to image features that persist for a long duration	It is important to minimize temporally persistent disturbances or artifacts in an video image
The illusion of “smooth” motion can be achieved by presenting frames at a rate of 20 ~ 30 Hz or more	Digital video systems should aim for frame repetition rates of 20 Hz or more for “natural” moving video
HVS responses vary from individual to individual	Multiple observers should be used to assess the quality of a digital video system

Hierarchical Structure of Video



Semantic Organization of Video

- **Objects**: e.g. people, cars, houses etc.
- **Activities or events**: e.g. talk, walk, dance, fight, crash etc.
- **Sites**: e.g. inside, kitchen, concert hall, outside, field, city, street etc.



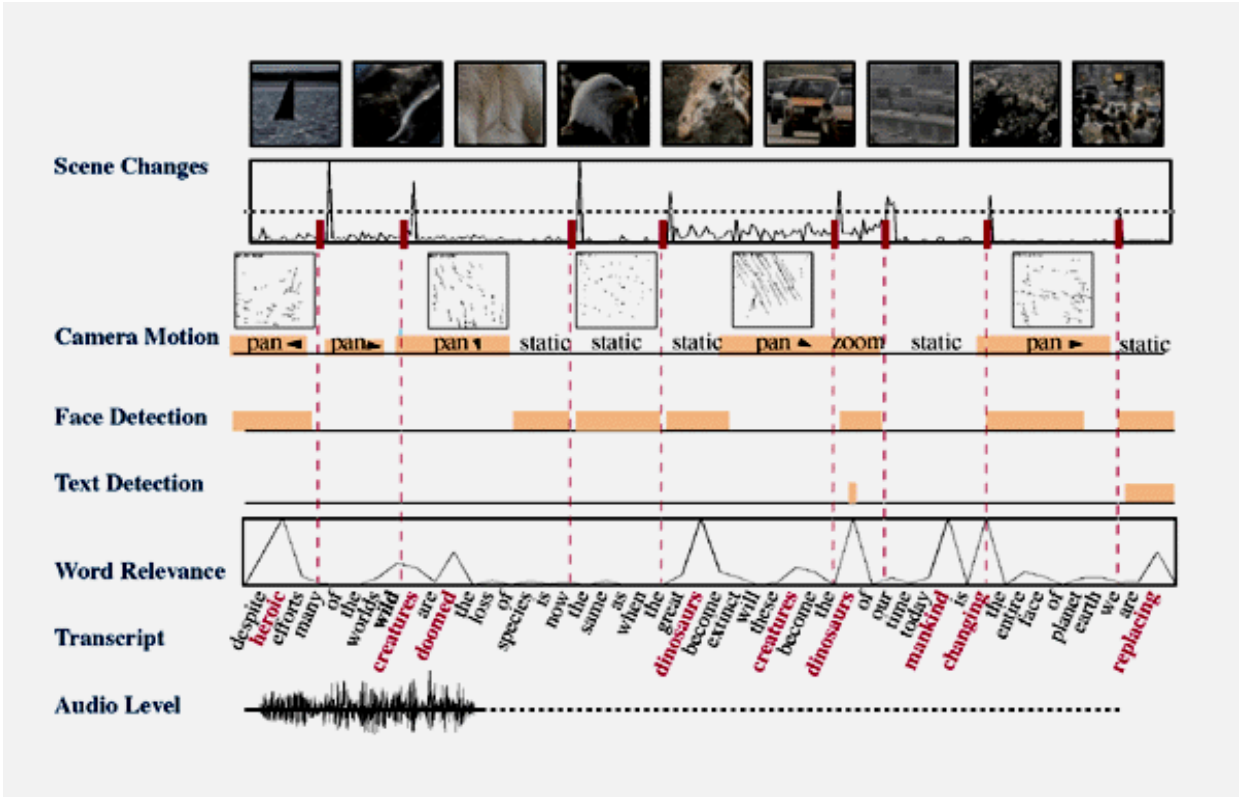
Video Segmentation

- Partition a video stream into a set of meaningful and manageable segments, which then serve as the basic unit for indexing
- Construct video TOC to convert an unstructured raw video into the structured video hierarchy to assist user's access

Video Segmentation

- Manual Segmentation
  - Labor intensive and time consuming
  - Prone to error
- Semi-automatic or Automatic Segmentation
  - Camera shot transitions: Abrupt and gradual
    - Abrupt transitions occur when two individual shots are simply pasted together;
    - Gradual transitions connect two shots smoothly by applying special editing techniques, such as fade, dissolve.
    - Detecting method – difference measures (e.g., pixel intensity difference, histogram difference).
  - Segmentation Cues: Video frame data, audio data, closed caption.

Example_



Video Segmentation

Shot Cut Detection

Intensity and Color Template Matching

$$S_1(f_m, f_n) = \sum_{i=1}^{X} \sum_{j=1}^{Y} D(f_m, f_n, i, j)$$

$$= \begin{cases} \sum_{l=1}^{3} | P(f_m, c_l, i, j) - P(f_n, c_l, i, j) | & for \quad color \quad image \\ | P(f_m, I, i, j) - P(f_n, I, i, j) | & for \quad gray \quad image \end{cases}$$

$P(f_m, c_l, i, j) \quad l = 1,2,3$
: color components of the pixel at (i,j)

$P(f_m, I, i, j)$
: the intensity of the pixel at (i,j)

A scene change is declared whenever $S_1(f_m, f_n)$ exceeds a pre-specified threshold

Shot Cut Detection

Histogram-based Matching

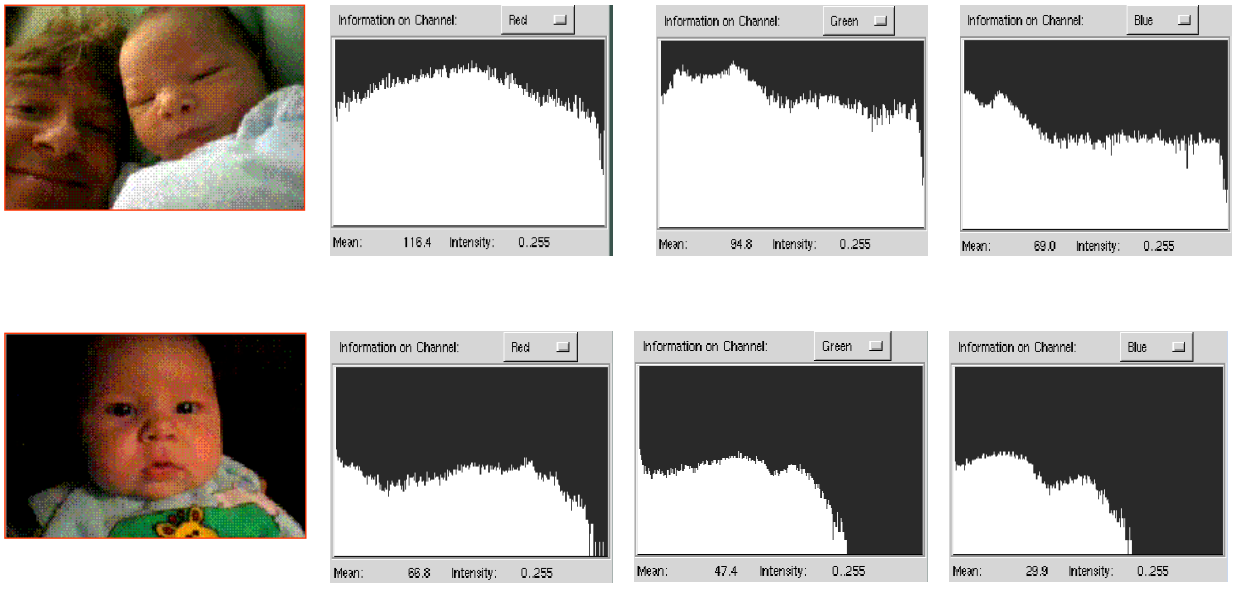
Two frames with minor changes in their intensity/color distributions are likely of a similar scene

$$S_2(f_m, f_n) = \sum_{i=1}^{N} \frac{[H(f_m,i) - H(f_n,i)]^2}{H(f_m,i)}$$

A scene change is declared whenever $S_2(f_m, f_n)$ exceeds a pre-specified threshold

Example_

Shot Cut Detection

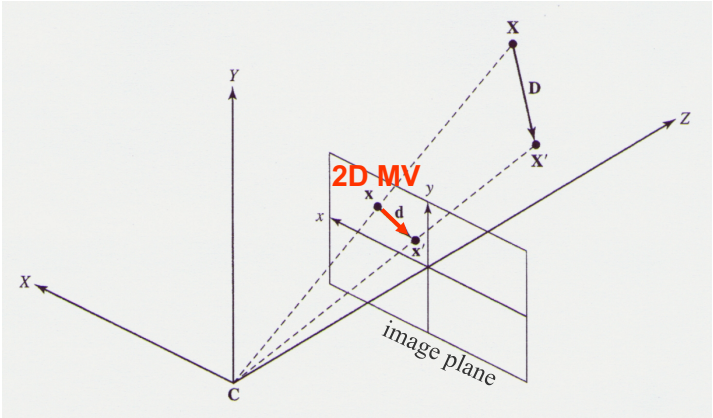


Motion Estimation

- Movement of objects and changes of camera settings during the capturing of image sequences result in temporal content variations.
  - Objects – translation, rotation, shape variations
  - Camera – zooming, tilting, panning, traveling
- Motion estimation is the estimation of the parameters of a video model that describes the temporal variations, usually from consecutive frames.
- Applications:
  - Motion compensated prediction (video compression)
  - Motion compensated interpolation (video stabilization)
  - Video image sequence analysis (computer vision)

2D Motion Model

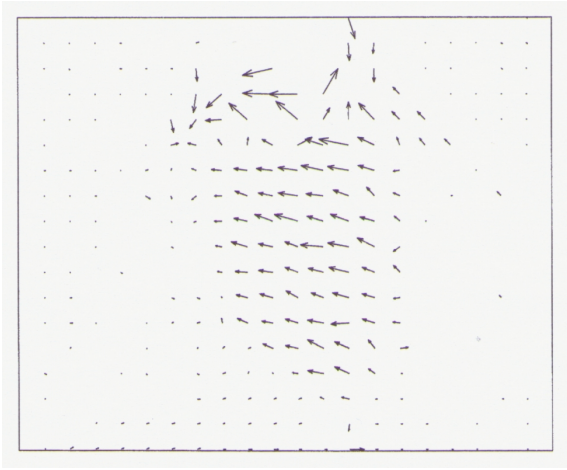
- *When the camera or the object in the scene is moving, the image of the same 3D object point will also change.*



Projection of a moving object onto an image plane

- When an object point is moved from $\mathbf{X}=[X,Y,Z]^T$ at time t_1 to $\mathbf{X'}=[X',Y',Z']^T=[X+D_X,Y+D_Y,Z+D_Z]^T$ at time $t_2=t_1+d_t$, its projected image is changed from $\mathbf{x}=[x,y]^T$ to $\mathbf{x'}=[x',y']^T=[x+d_x,y+d_y]^T$.
- We call the 3D displacement, $\mathbf{D}(\mathbf{X};t_1,t_2)=\mathbf{X'}-\mathbf{X}=[D_X,D_Y,D_Z]^T$, the 3D motion vector (MV) at $\mathbf{X}$, and the 2D displacement, $\mathbf{d}(\mathbf{x};t_1,t_2)=\mathbf{x'}-\mathbf{x}=[d_x,d_y]^T$, the 2D motion vector at $\mathbf{x}$.
- In general, the motion vectors (MVs) are position-dependent.
- As a function of all image positions $\mathbf{x}$ at time t_1, $\mathbf{d}(\mathbf{x};t_1,t_2)$ represents a **2D motion field** from t_1 to t_2.

2D Motion Field

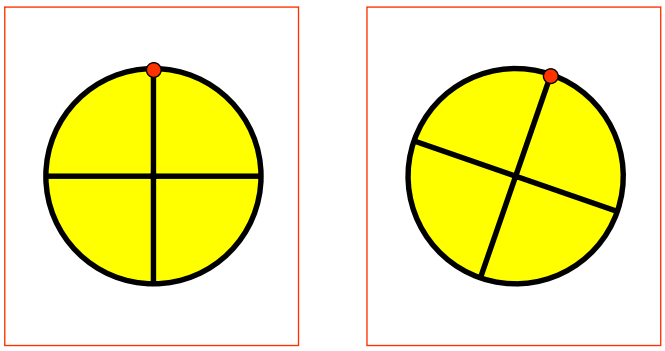


- Here we deal only with digital video signals that have finite and discrete image domains, described by a truncated lattice, Λ. The notation $\mathbf{x}=[x,y]^T \in \Lambda$ represents a pixel index.
- We further assume that the time interval $d_t=t_2-t_1$ is equal to either the temporal sampling interval (i.e., the frame interval) or an integer multiple of this interval. The motion field for a given time interval is a finite set of 2D vectors arranged in a 2D array, in the same order as the pixels.

- Such a **discrete motion field** is often depicted by a vector graph, where the direction and magnitude of each arrow represents the direction and magnitude of the MV at the pixel where the arrow originates.

2D Motion vs Optical Flow

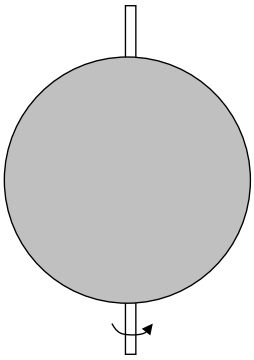
- The human eye perceives motion by identifying corresponding points at different times.
- *The correspondence is usually determined by assuming that the color or brightness of a point does not change after the motion.*

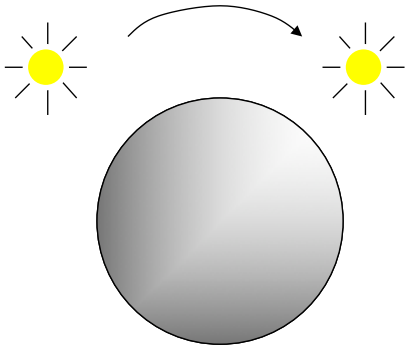


Frame at time t_1 Frame at time t_2

2D Motion vs Optical Flow

- It is interesting to note that **observed 2D motion can be different from the actual projected 2D motion under certain circumstances**.
- *Observed or apparent 2D motion is referred to as **optical flow** in computer vision literature.*

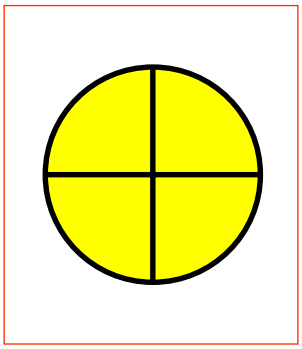




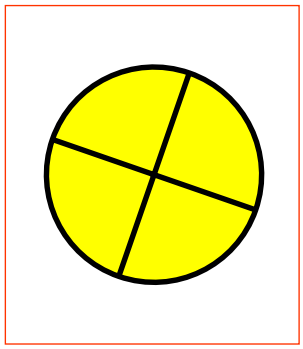
- **Case1**: a sphere with a uniform flat surface is rotating under a constant ambient light, but the observed image does not change.
- **Case2**: a point light source is rotating around a stationary sphere, causing the highlight point on the sphere to rotate.

Optical Flow

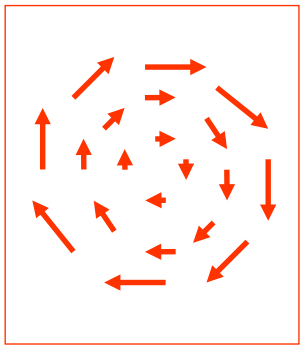
- **Optical flow** reflects the image / frame changes due to motion during a time interval dt, and the optical flow field is the velocity field that represents the 3D motion of object points across a 2D image / frame.
- A simulated example of two consecutive frames and a corresponding optical flow image



Frame at time t_1



Frame at time t_2



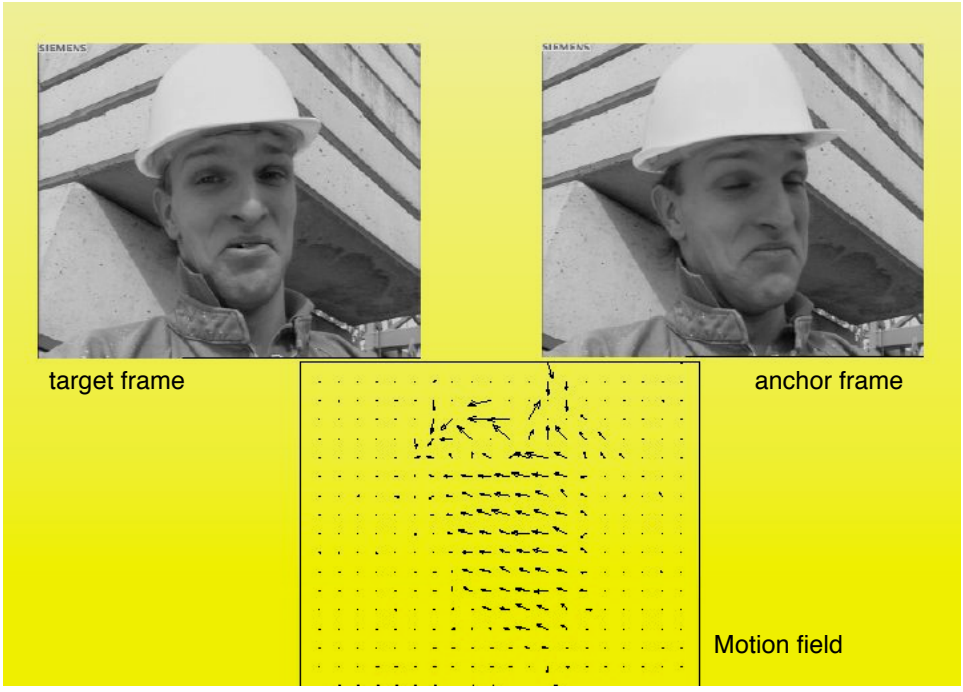
Optical flow

- Optical flow computation (see Appendix)

Sample Motion Field

"*Foreman*"

Sample Motion Field



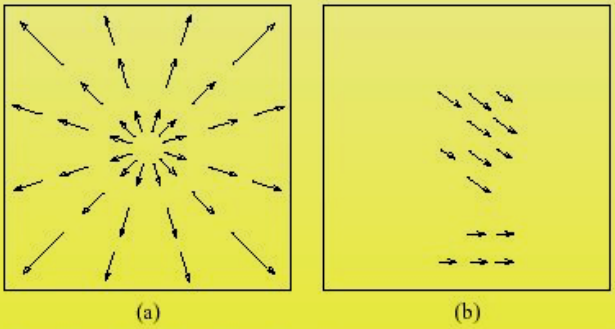
Motion Estimation

Motion Representation

"*Claire*"

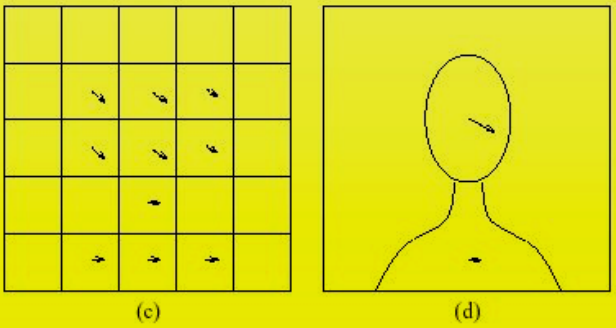
A head-and-shoulder scene

Global:
Entire motion field is represented by a few global parameters



Pixel-based:
One MV at each pixel, with some smoothness constraint between adjacent MVs.

Block-based:
Entire frame is divided into blocks, and motion in each block is characterized by a few parameters.

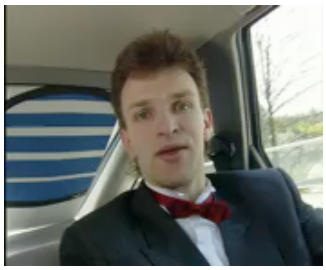


Region-based:
Entire frame is divided into regions, each region corresponding to an object or sub-object with consistent motion, represented by a few parameters.

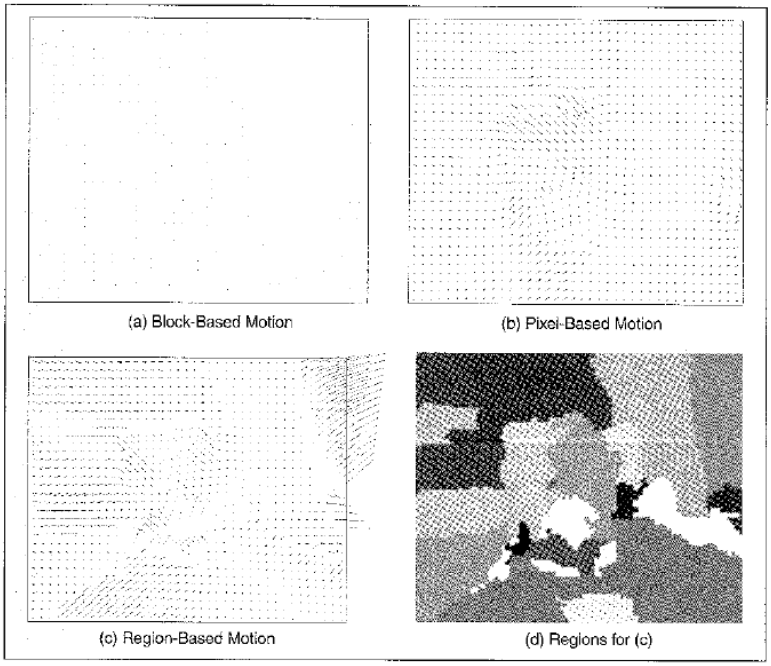
Ⓟ

Motion Estimation

Motion Representation

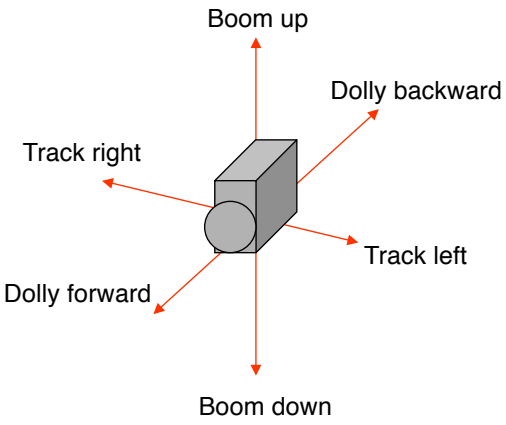


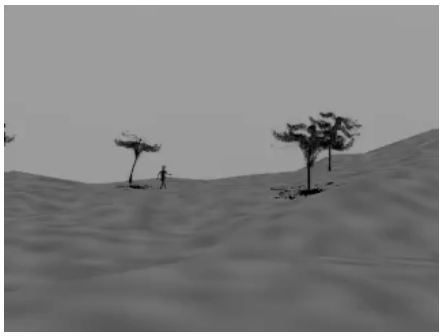
Example: "*Carphone*" – frame #171



(See Appendix: Reading – *Estimating Motion in Image Sequences)*

Typical Camera Motions

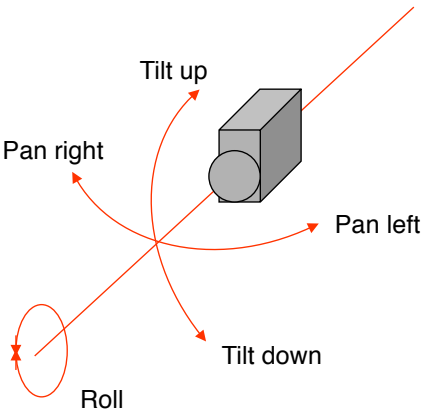


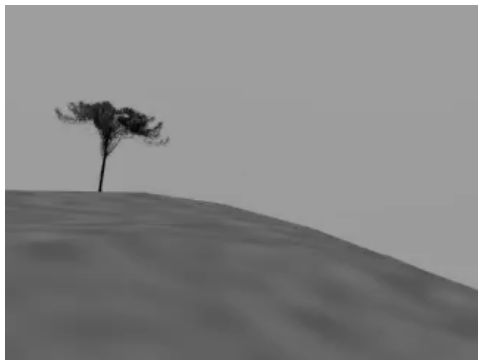


Dolly forward

Track right

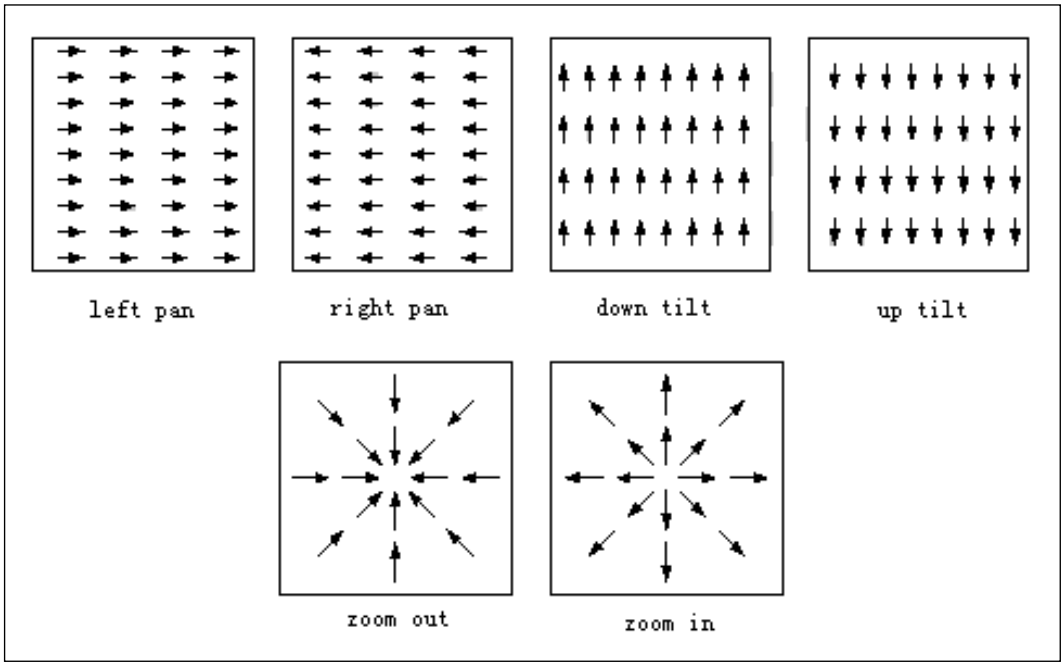
Typical Camera Motions



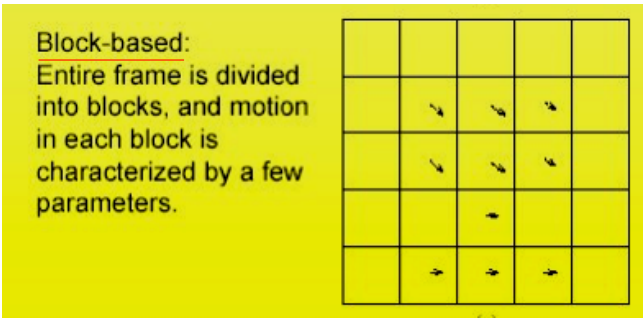


Pan left

2D Motion Corresponding to Camera Motion

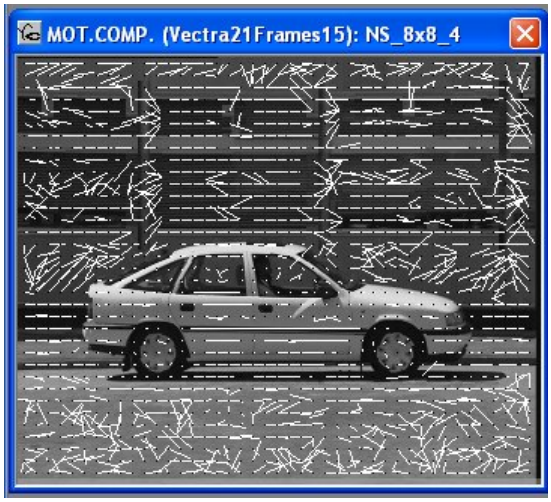


Block-based Motion Estimation

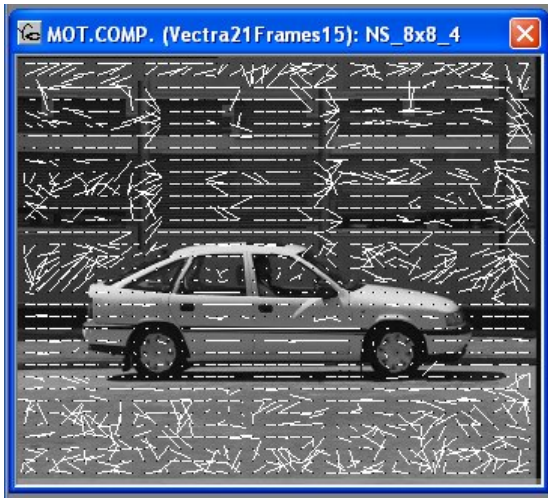


- In block-based motion estimation, we assume all pixels in a block undergo a coherent motion, and search for the motion parameters for each block independently.

Block-based Motion Estimation



Block-based Motion Estimation



Video Object Extraction

Example_ Automated Video Surveillance

- Automatic detection and recognition of objects is of prime importance for security systems and video surveillance applications.
- Automated video surveillance addresses real time observation of people and vehicles within a busy environment.

- Boundary-box-based segmentation

Object-Based Representation

Example_ Content-based Video Retrieval

- Key-frame-based representation, shot-based representation, scene-based representation, … → video frame-based representation techniques.
- If a query to a video database involves objects in a video clip, the above frame-based representation methods do not provide sufficient resolution to support such queries.
- Therefore, incorporating object-based primitives into video content representation, → object-based representation, is advantageous when dealing with queries about objects and their motions.

Object-Based Representation

- In the object-based representation, the main attributes attached to key or dominant objects are motion, shape and life cycle, as well as other image features such as color and texture.

Original sequence
from the movie "*Legally Blonde*"

Segmented sequence
using color and motion

The Life Cycle of Video Object

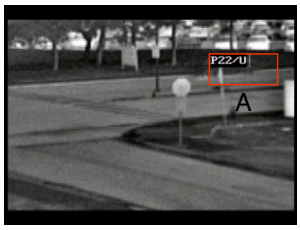
- The life cycle of a video object is the duration from the time (or relative frame number) when the object appears into a shot (i.e., birth of object) to the time when the object disappears (i.e., death of object).
- The object motion can be decomposed into a global component and a local / object-based component to reflect motion relative to the background and other objects in the scene.
- Without this distinction, the object motion would instead represent motion relative to the image frame.

The Life Cycle of Video Object

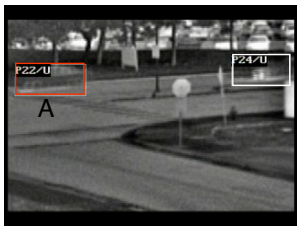
Example_ Traffic surveillance system

Infrared (IR) video

before
the birth of
object A



the birth of
object A



the death of
object A

after
the death of
object A

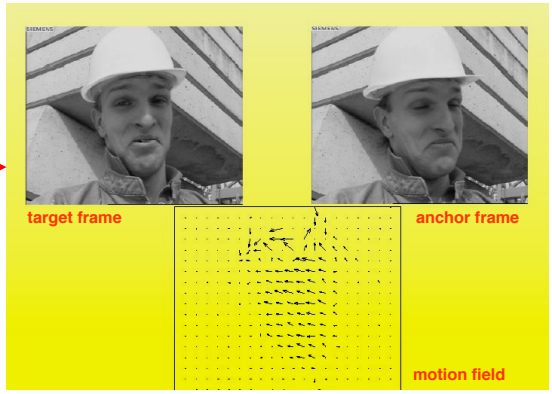
Application: Visual Fire Object Detection in Video

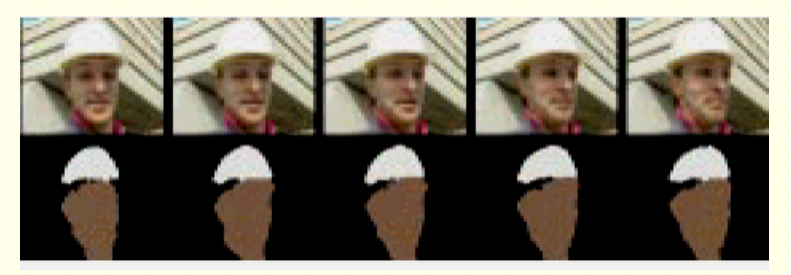
- Visual fire detection has the potential to be useful in conditions in which conventional methods cannot be used – especially in the recognition of fire in movies / video. For example, categorizing movies according to the level of violence.

Video Object Extraction

Original video: "*Foreman*"

motion estimation



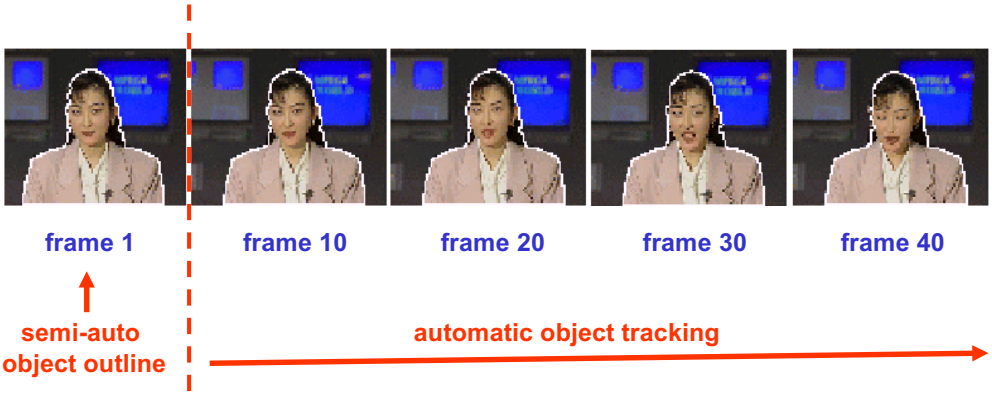


- Motion-based segmentation for video scene with stationary background

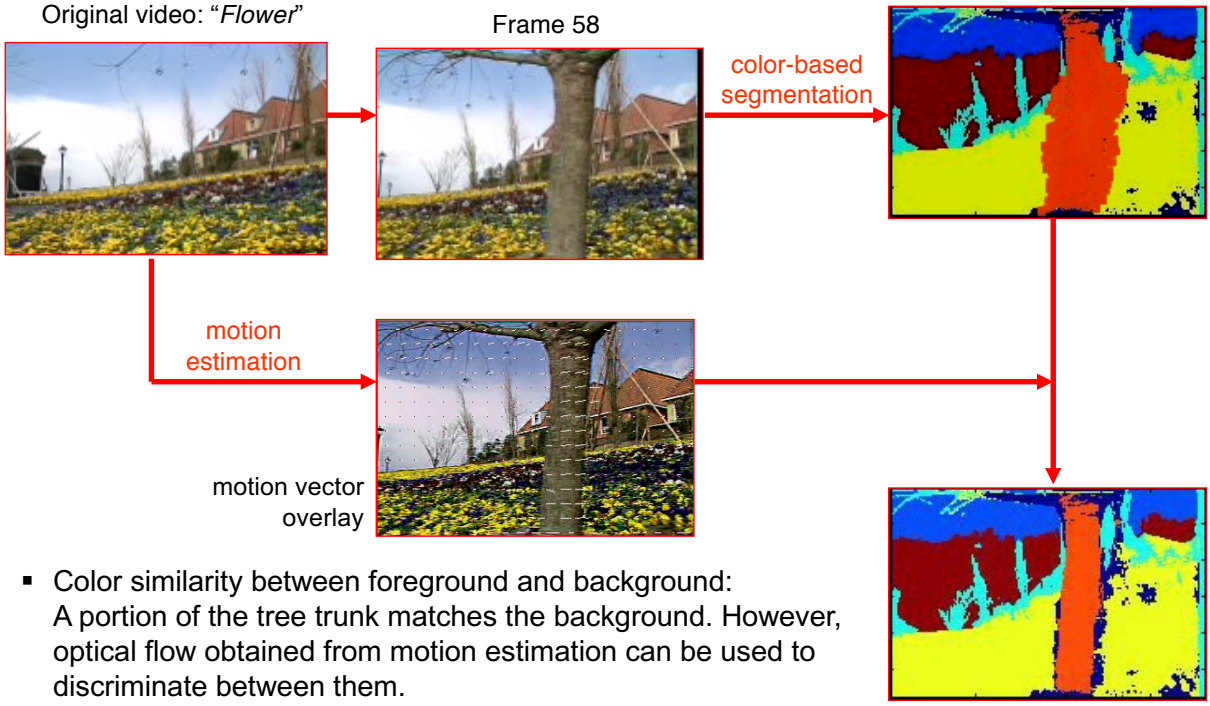
Video Object Extraction

- User assistance is available in defining the video object, e.g., initial approximate outline + automatic tracking

Original video: "Akiyo"

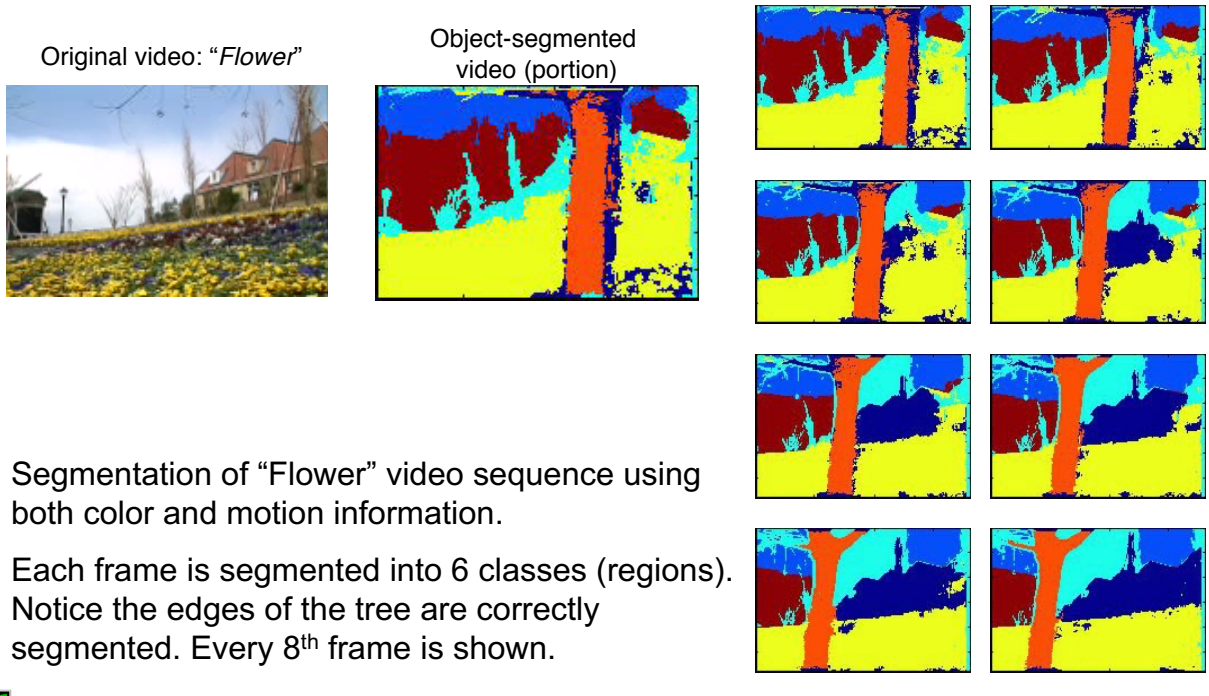


Video Object Extraction



- Color similarity between foreground and background:
  A portion of the tree trunk matches the background. However, optical flow obtained from motion estimation can be used to discriminate between them.

Video Object Extraction



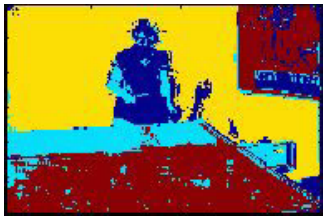
- Segmentation of "Flower" video sequence using both color and motion information.

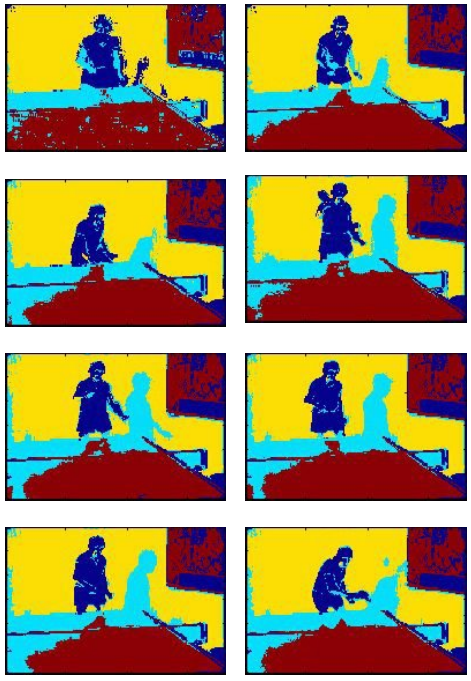
  Each frame is segmented into 6 classes (regions). Notice the edges of the tree are correctly segmented. Every 8th frame is shown.

Video Object Extraction

Original video: “Tennis”

Object-segmented video (portion)





- Segmentation of “Tennis” video sequence using both color and motion information.

  Each frame is segmented into 6 classes (regions). Notice that the player has inconsistent motion, and is often stationary, matching with the background. However, color based segmentation takes over in this situation. Every 10th frame is shown.

Week04 • Semester 2 • 2018

Video Data Representation & Processing

Appendix

- ❑ **Optical flow computation**
- ❑ **Reading:** ***Estimating Motion in Image Sequences***