المحور السادس

Video Basics (26th)

INTRODUCTION:

Video can be defined as the photographic images that are played back at speeds of 15, 25, 30 frames per second and provide the appearance of full motion (see **Figure 1**). **A video consists of a time-ordered sequence of frames (images).**

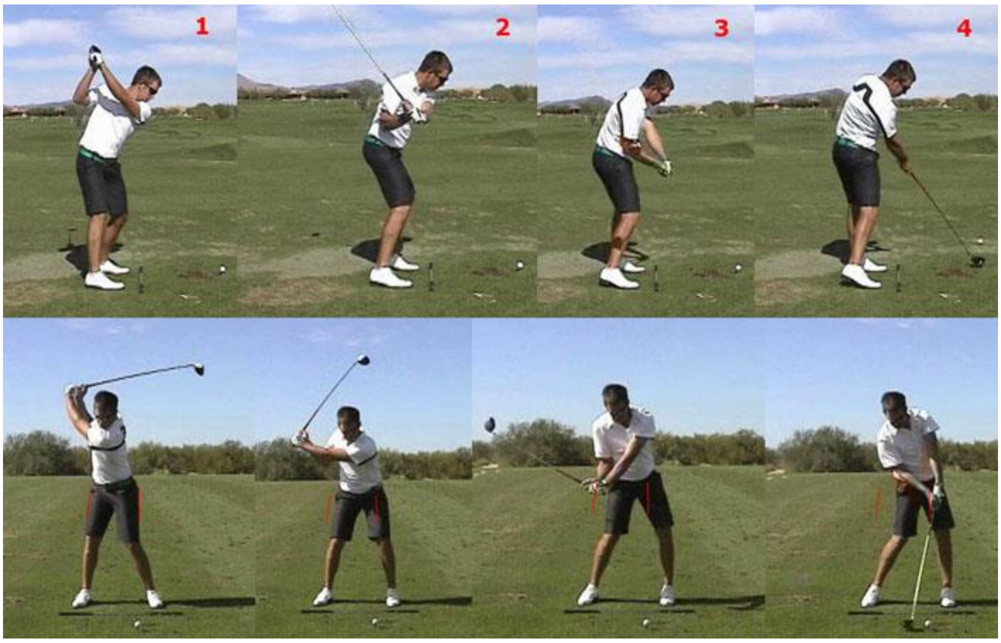


Figure 1: Video Frame

DIGITAL VIDEO:

Digital video refers to the capturing, manipulation, and storing of moving images that can be displayed on computer screens. This requires that the moving images be digitally handled by the computer. The word digital refers to a system based on discontinuous events (sampling), as opposed to analogue, a continuous event. Visual **pixels** are the basic unit in digital video, where each color component sorted digitally in each pixel.

5 Things To Consider When Playing Video:

- **Frame rate**: is the frequency (normally per second) at which frames in a television picture, film, or video sequence are displayed.
- **Video resolution**: is the number of distinct pixels in each dimension that can be displayed.
- **Video colouring model (colour space):** A colour model is an abstract mathematical model describing the way colours can be represented as tuples of numbers.
- **Raw video:** It is the unprocessed data from a camera's image sensor. Most

photographers prefer shooting raw film due to the high quality of images that the camera sensor could possibly produce.

- **Aspect Ratio:** The aspect ratio of an image describes the proportional relationship between its **width** and its **height**. It is commonly expressed as two numbers separated by a colon, as in 16:9.

Frame Dimensions, Number of Lines, and Resolution
A video frame is composed of lines. In digital video, each line is sampled to create a number of pixels (samples) per line. The more lines per frame, the higher the image resolution. The more pixels per line, the higher the resolution of each line.
Number of Lines

In analog video, NTSC uses 525 lines, whereas PAL uses 625, many lines are not actually used for picture information, so the total numbers relevant for the picture are somewhat smaller: 486 lines for NTSC and 576 lines for PAL. HD formats defined by the ATSC have either 1080 or 720 active picture lines per frame.
Pixels per Line

In digital video formats, each line is sampled a number of times. In an attempt to create a single digital VTR that could digitize and record both NTSC and PAL signals. Therefore, a digital NTSC video frame is 720 pixels x 486 lines, and a PAL video frame is 720 pixels x 576 lines.
HD video with 1080 lines uses 1920 pixels per line (1920 x 1080). HD video with 720 lines uses 1280 pixels per line (1280 x 720). Both of these formats have an aspect ratio of 16:9.

Common video frame sizes are shown in the table.

Width	Height	Pixel aspect ratio	Screen aspect ratio	Description
320	240	1:1	4:3	Used for web distribution or offline video editing.
640	480	1:1	4:3	An early standard for analog-to-digital video editing, and an ATSC video specification.
720[1]	480	Height greater than width	4:3	NTSC DV and DVD image dimensions. Also part of the ATSC video specification.
720[1]	486	Height greater than width	4:3	NTSC SD video dimensions used for professional digital formats such as Digital Betacam, D-1, and D-5.
720[1]	576	Width	4:3	PAL SD video dimensions used for digital

		greater than height		formats such as Digital Betacam, D-1, and D5, as well as DVD and DV.
1280	720	1:1	16:9	An HD video format, capable of higher frame rates in exchange for smaller image dimensions.
1920	1080	1:1	16:9	An HD video format with very high resolution.
960	720	4:3	16:9	Some 720p formats (such as DVCPRO HD and HDV) subsample 1280 pixels to 960 to minimize the data rate.
1440 1280	1080	4:3 3:2	16:9	Some 1080-line formats (such as HDV and DVCPRO HD) subsample 1920 pixels to 1440 or even 1280 to minimize the data rate.

In order to know calculate the size of a given video clip (without compression), we use the following equation:

video size (bits) = width x height x bit per pixel x frame rate x duration (time)

Where:

Bit per pixel = 24 for RGB, 8 for gray-level and 1 for binary image

Frame rate or frame per second = 15, 25, 30 fps or even more

Duration = time in seconds

Example:
Find the size in Kbyte for a 10 minutes Full HD (1920x1080) video sequence (RGB type) with a frame rate = 30 fps.

Solutions:
RGB type = 24 bits=3Bytes

Time in seconds = 10 x 60 = 600

Video size = 1980 x 1080 x 3 x 30 x 600

Video size = = 115473600000 Bytes

To convert to KBytes

Video size = 115473600000/1024

= 112767187.5 Kbytes

Video Rate (bits) = width x height x bit per pixel x frame rate

Example:
What is the bit rate for high-definition TV (HDTV)?

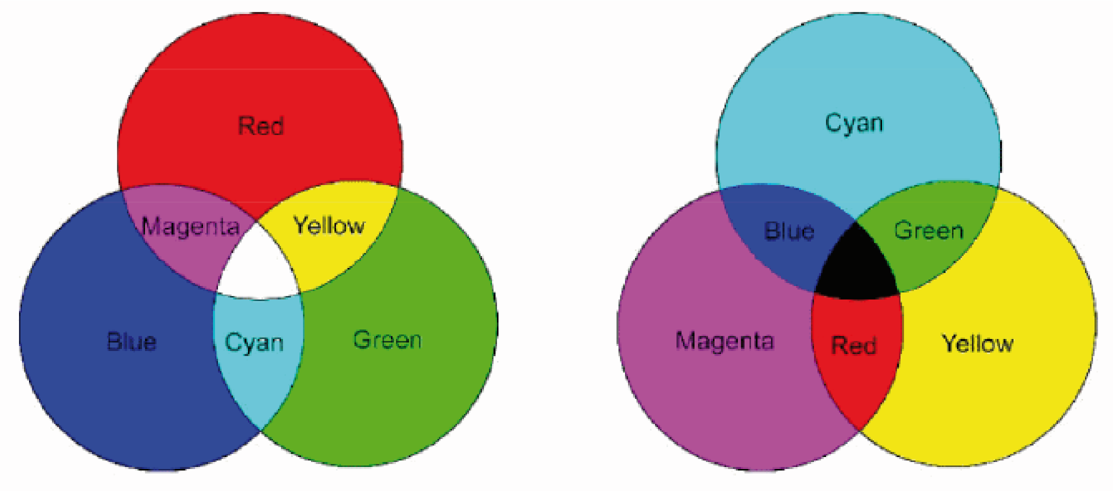
Solution: 1920 x 1080 X 30 x 24 = 1492992000 = 1.39 Gbps
HDTV uses digital signals to broadcast high quality video signals. The HDTV screen is normally a ratio of 16: 9.
There are 1920 by 1080 pixels per screen, and the screen is renewed 30 times per second. Twenty-four bits represents one color pixel.

27th Video Color Models

Color Model

1. **RGB Model**

- Generate all others colours based on three basic colours which are Red , Green, and Blue.
- (Newer) Color LCD panels (typically thin-film-transistor liquid-crystal displays (TFT LCD)): Transistor switch for each (R, G or B) pixel
- Used in display (monitor) LED, data-show etc.



2. **CMY and CMYK model**

- Combinations of additive and subtractive colors
- Cyan, Magenta, and Yellow (CMY) are complementary colors of RGB - Subtractive Primaries.
- CMY model is mostly used in printing devices where the color pigments on the paper absorb certain colors.
- E.g., convert White from (1, 1, 1) in RGB to (0, 0, 0) in CMY.
- Used in color printer

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}$$

(R G B)=➔((1-R)=C,(1-G)=M,(1-B)=y)

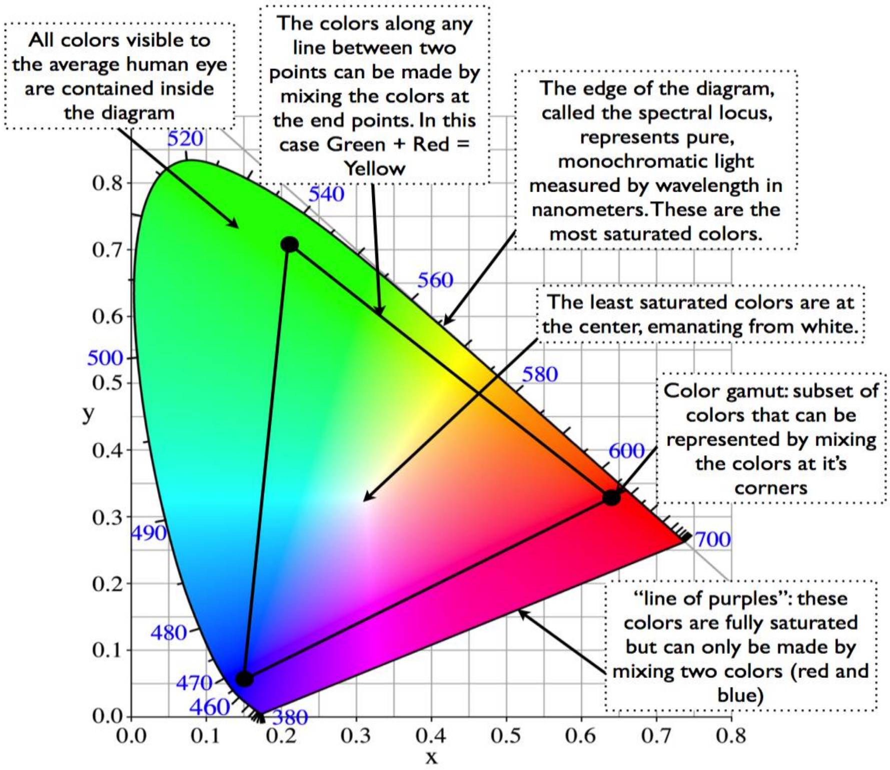
CMYK model *: An improved Printing Color Model Sometimes, an alternative CMYK model (K stands for Black) is used in color printing*

Original Color Image

C, M, Y, K image Intensities

3. CIE Color model

- In 1931, the CIE defined three standard primaries (X, Y, Z)
- The Y primary was intentionally chosen to be identical to the luminous-efficiency function of human eyes (Perceptual Model).
- All visible colors are in a horseshoe shaped cone in the X-Y-Z space. Consider the plane X+Y+Z=1 and project it onto the X-Y plane, we get the CIE chromaticity diagram.
- The edges represent the pure colors



Anatomy of a CIE Chromaticity Diagram

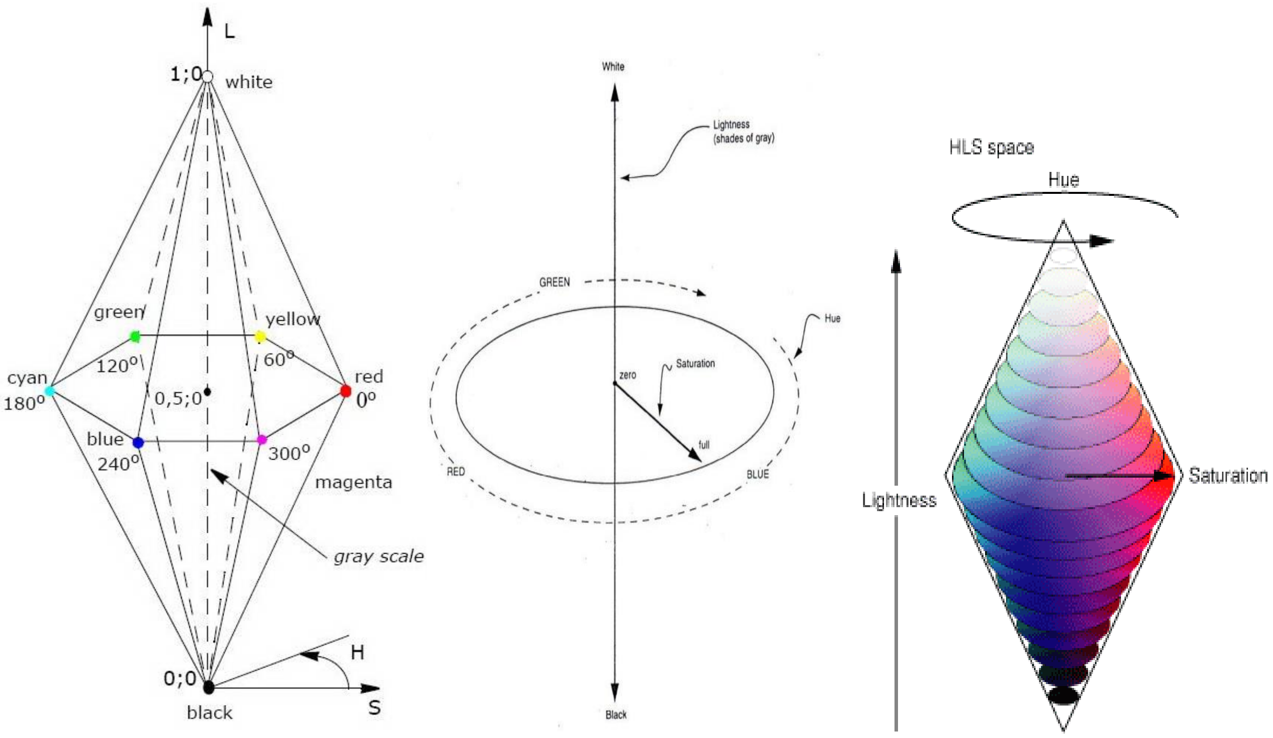
4. YUV Color Model

- Digital video standard established in 1982
- Video is represented by a sequence of fields (odd and even lines). Two fields make a frame.
- Works in PAL (50 fields/sec) or NTSC (60 fields/sec)
- Uses the Y, U, V color space.

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.299 & -0.587 & 0.886 \\ 0.701 & -0.587 & -0.114 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

5. HSL color model

For example, the **H**ue/**S**aturation/**L**ightness (HSL color transform allows us to describe colors in terms that became more readily understand (see Figure (2-b). The ***lightness*** is the brightness of the color, and the ***hue*** is what we normally think of as "color" (for example green, blue, or orange). The ***saturation*** is a measure of how much white is in the color (for example, pink is red with more white, so it is less saturated than a pure red). Most people can relate to this method of describing color. For example, "a deep, bright orange" would a have a large intensity ("bright"), a hue of "orange" and a high value of saturation ("deep"). We can picture this color in our minds, but if we defined this color in terms of its RGB component R=245, G=110, and B = 20, most people would have no idea how this color appears. Because the HSL color space was developed based on heuristics relating to human perception, various methods are available to transform RGB pixel values into the HSL color space. Most of these are algorithmic in nature and are geometric approximations to mapping the RGB color cube into some HSL-type color space.



Figure(2) a HSL image

Video Compression

Why need video compression?

Imagine that One movie video without compression 720 x 480 pixels per frame Full color with 30 frames per second for 90 minutes needs 167.96 G Bytes !! (storage[1]). In addition, the bit rate for high-definition TV (HDTV) needs 1.39 Gbps. Which is huge bit rate required. (transmssion[2]).

$$Compression\ Ratio = \frac{Uncompressed\ File\ Size}{Compressed\ File\ Size} = \frac{size_U}{size_C}$$

The following techniques are commonly employed to achieve desirable reductions in image data:

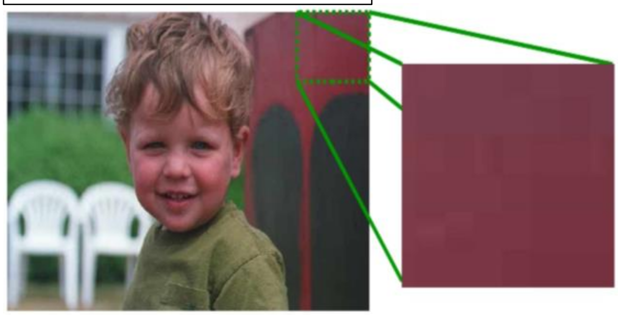
1) Reduce color nuances within the image
2) Reduce the color resolution with respect to the prevailing light intensity
3) Remove small, invisible parts, of the picture
4) Compare adjacent images and remove details that are unchnaged between two images

The first three are image based compression techniques, where only one frame is evaluated and compressed at a time. The main one that does the real image compression is the **Discrete Cosine Transform (DCT)** followed by a quantization that removes the redundant information (the “invisible” parts).

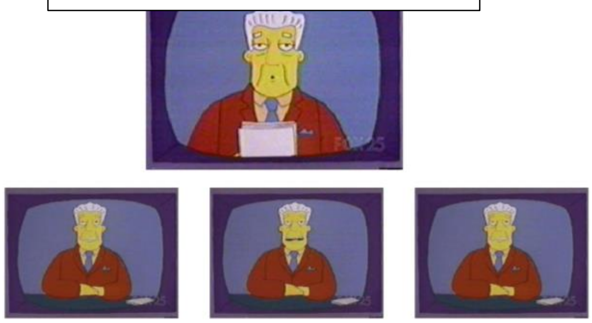
In video there is redundancy data.

a) **Spatial redundancy**: take advantage of similarity among most neighboring pixls
b) **Temporal redundancy:** take advantage of similarity between frames

a)Spatial redundancy



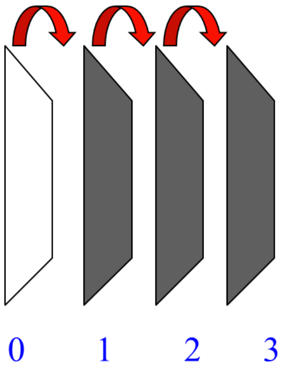
b)Temporal redundancy



To compress video should be either within a frame or multi-frame

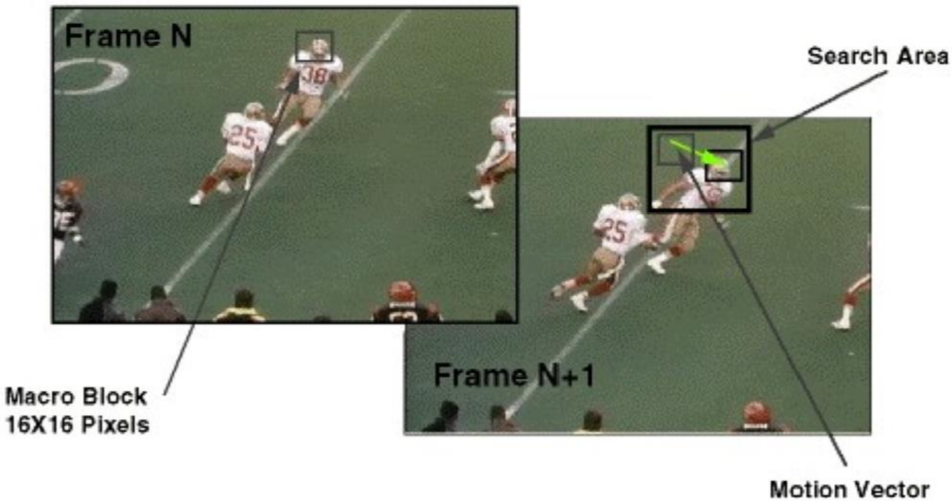
1- Compresses each frame in isolation, treating it as a bitmapped image. Based on quantization of DCT coefficients

2- Compresses sequences of frames by only storing differences between them. Based on Motion Compensation (MC).



$Frame_0$ (still image)
Difference $frame_1$ = ($Frame_1$ – $Frame_0$)
Difference $frame_2$ = ($Frame_2$ – $Frame_1$)
If no movement in the scene, all difference frames are 0.
Can be greatly compressed!

A motion vector (MV) describes the offset between the location of the block being coded (in the current frame) and the location of the best-match block in the reference frame



An obvious solution for video compression would be predictive coding based on previous frames. For example, suppose we simply created a predictor such that the prediction equals the previous frame. Then compression proceeds by subtracting images: instead of subtracting the image from itself (i.e., use a derivative), we subtract in time order and code the residual error. And this works. Suppose most of the video is unchanging in time.

The idea of looking for the football player in the next frame is called motion estimation, and the concept of shifting pieces of the frame around so as to best subtract away the player is called motion compensation.

The basic principle for video compression is the image-to-image prediction. The first image is called an I-frame and is self-contained, having no dependency outside of that image. The following frames may use part of the first image as a reference. An image that is predicted from one reference image is called a P-frame and an image that is bidirectionally predicted from two reference images is called a B-frame.

- **I**-frames: I (**I**ntracoded) frames, self-contained
- **P**-frames: (**P**redicted) from last I or P reference frame

- B-frames: (Bidirectional); predicted from two references one in the past and one in the future, and thus out of order decoding is needed

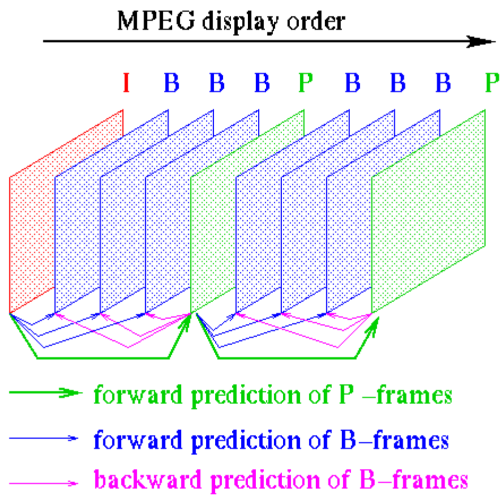
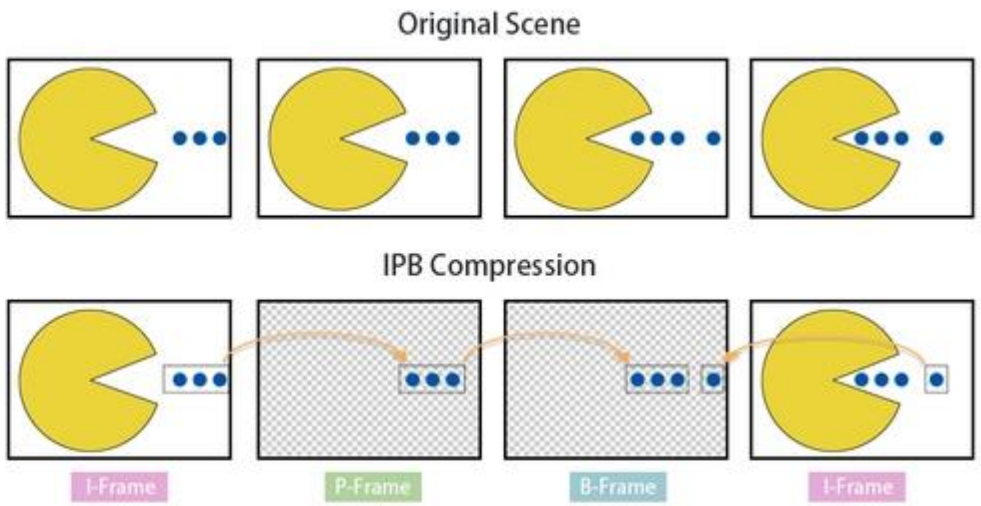


Figure: The illustration above shows how a typical sequence with I-, B-, and P-frames may look. Note that a P-frame may only reference a preceding I- or P-frame, while a B-frame may reference both preceding and succeeding I- and P-frames.



The video decoder restores the video by decoding the bit stream frame by frame. Decoding must always start with an I-frame, which can be decoded independently, while P- and B-frames must be decoded together with current reference image(s).

Steps to compress video

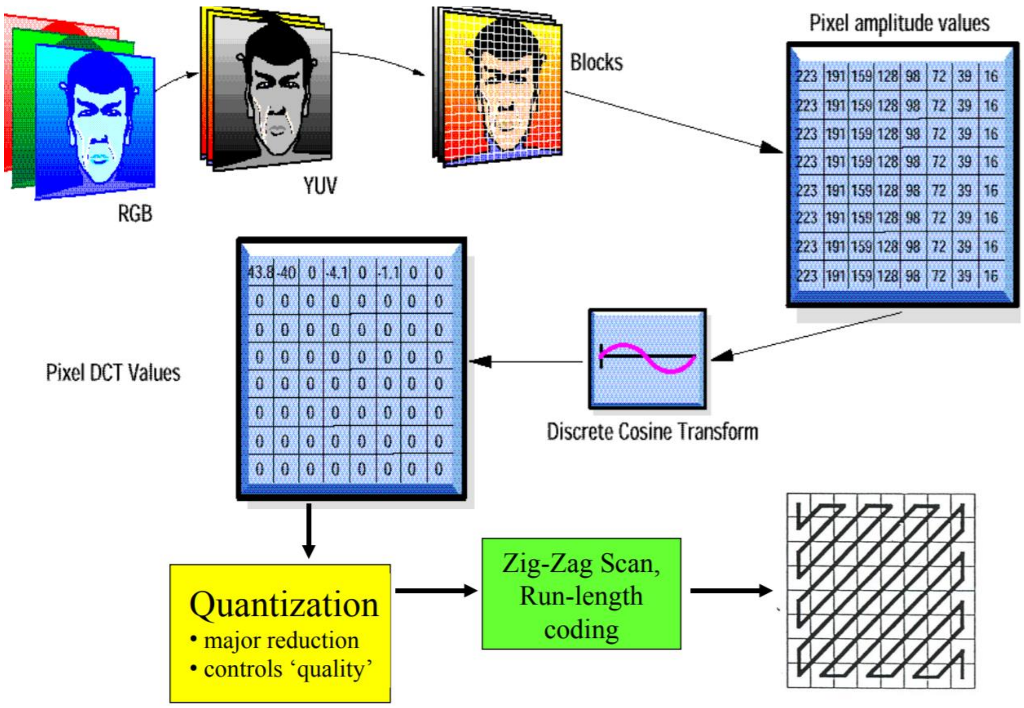
1- Divide Image to blocks (16x16 or 8x8)

2- Use DCT based techniques for spatial redundancy removal (Intra-frame compression).

3- Use MC (Motion Compensation) techniques for temporal redundancy removal (Inter-frame compression).

4- Final stage is two dimensional run-length coding.

Encoding of I-frames


RGB
YUV
Blocks
Pixel amplitude values
223 191 159 128 98 72 39 16
Discrete Cosine Transform
Pixel DCT Values
43.8 -40 0 -4.1 0 -1.1 0 0
Quantization
• major reduction
• controls ‘quality’
Zig-Zag Scan,
Run-length
coding