Video Basic Training

Fundamentals of Analog Video:

Analog video refers to the method of transmitting and displaying video signals in a continuous, varying format. It was widely used before the digital era and is still found in certain applications today. Here are some fundamentals of analog video.

1.Signal Representation: Analog video represents images and motion as continuously varying voltage signals. These signals are created by capturing light through a camera's lens and converting it into electrical signals. The voltage levels correspond to the intensity of light captured by the camera's sensor.
2.Signal Components:

- **Luminance (Y)**: Represents the brightness or intensity of the image. It carries the grayscale information.
- **Chrominance (C)**: Carries the color information. It consists of two components:
- **Chrominance Blue (Cb)**: Represents the blue color information.
- **Chrominance Red (Cr)**: Represents the red color information.

3.Scan Lines: Analog video frames are composed of scan lines, which are horizontal lines of pixels. The more scan lines per frame, the higher the resolution of the video.
4.Frame Rate: Analog video is displayed as a series of still frames in rapid succession. The frame rate is the number of frames displayed per second (fps). Common frame rates include 24 fps (film), 30 fps (standard video), and 60 fps (higher refresh rates).
5.Video Standards:

- **NTSC (National Television System Committee)**: Used in North America, Japan, and other regions. It has a frame rate of 30 fps and 525 scan lines.
- **PAL (Phase Alternating Line)**: Used in Europe, Asia, and other regions. It has a frame rate of 25 fps and 625 scan lines.
- **SECAM (Séquentiel Couleur à Mémoire)**: Used in France and some other regions. Similar to PAL but with some technical differences.

6.Composite Video: In composite video, all video components (luminance and chrominance) are combined into a single signal. This is achieved through modulation and demodulation processes. Composite video cables typically use RCA connectors and are often color-coded yellow.
7.Component Video: Component video separates the video signal into its luminance and chrominance components. This results in better color accuracy and reduced interference compared to composite video. It uses three separate cables for Y, Pb, and Pr signals.
8.Analog Video Transmission: Analog video signals can degrade over distance due to interference and attenuation. Coaxial cables were commonly used to transmit analog video signals, and their quality impacted signal integrity.
9.Analog to Digital Conversion: With the advent of digital technology, analog video signals need to be converted to digital format for modern displays and storage. This conversion involves sampling the analog signal at regular intervals and assigning digital values to represent the signal's amplitude.

What Is Picture Basics In Fundamentals Of Analog Video?

In the fundamentals of analog video, understanding the basics of picture refers to the foundational aspects related to the visual representation and characteristics of analog video signals. Here are the key components and concepts related to the picture basics in analog video:

Luminance (Y):

Luminance represents the brightness or intensity component of a video signal.

It carries information about the light intensity of each pixel or scan line.

Luminance is typically represented as a grayscale signal, ranging from black to white.

Chrominance (C):

Chrominance carries color information in an analog video signal.

It represents the color difference signals, often referred to as color components, which determine the hue and saturation of the image.

Chrominance signals are usually encoded as color-difference signals, such as R-Y (red minus luminance) and B-Y (blue minus luminance) in the YUV color space.

Aspect Ratio:

The aspect ratio determines the proportional relationship between the width and height of the video frame.

Resolution:

Analog video resolution refers to the number of lines and pixels used to create the video image.

Horizontal and Vertical Sync:

Horizontal sync is a synchronization signal used to mark the beginning of each line in the video signal.

Vertical sync is a synchronization signal that marks the beginning of each frame.

These sync signals ensure proper scanning and timing synchronization between the video source and the display device.

Signal-to-Noise Ratio (SNR):

The signal-to-noise ratio measures the quality of the analog video signal by comparing the strength of the desired video signal to the level of unwanted noise.

A higher SNR indicates a cleaner and better-quality video signal with less visible noise.

Refresh Rate:

The refresh rate determines the number of times the video image is redrawn on the screen per second.

Standard analog video refresh rates are typically 50 or 60 Hz, depending on the video standard.

Interlaced and progressive

Interlaced and progressive are two different methods of scanning and displaying video images. Here's an explanation of the differences between interlaced and progressive:

Interlaced Scanning:

- Interlaced scanning is an older video scanning method commonly used in analog television systems.
- It divides each frame into two interlaced fields: the odd lines and the even lines.
- The first field, consisting of the odd-numbered lines, is displayed, followed by the second field with the even-numbered lines.
- The interlaced fields are displayed alternately, resulting in a perceived full frame.
- Interlaced scanning is denoted by "i" in video resolutions (e.g., 480i, 1080i).

Progressive Scanning:

- Progressive scanning is a newer video scanning method commonly used in digital video systems and modern display technologies.
- It scans and displays the entire frame in a single pass, sequentially from top to bottom.
- Each frame is displayed progressively, line by line, without the interlacing effect.
- Progressive scanning is denoted by "p" in video resolutions (e.g., 720p, 1080p).

Differences between Interlaced and Progressive:

- **Image Quality:**
- Progressive scanning generally provides better image quality and clarity, especially for still images and slow-motion content.
- Interlaced scanning can sometimes introduce visible artifacts, such as flickering or combing effects, particularly with fast-moving objects or motion-intensive content.
- **Motion Handling:**
- Progressive scanning is more effective at displaying fast motion smoothly, as each frame represents a complete image without interlacing.
- Interlaced scanning may exhibit motion artifacts due to the alternating display of interlaced fields, resulting in potential motion blur or jagged edges.
- **Display Compatibility:**
- Most modern display devices, such as LCD and OLED screens, are designed for progressive scanning and support higher refresh rates.
- Interlaced content may not be displayed optimally on modern screens, often requiring deinterlacing or conversion to progressive format.
- **Video Standards:**
- Interlaced scanning was historically used in analog television standards like NTSC and PAL, as well as earlier digital formats.
- Progressive scanning is now the standard for digital video formats and high-definition content.

When it comes to choosing between interlaced and progressive, progressive scanning is generally preferred for its improved image quality, motion handling, and compatibility with modern display technologies. However, interlaced scanning may still be relevant in certain contexts, particularly when dealing with legacy systems or specific broadcasting requirements.

Video resolutions and formats in analog video:

In analog video, the resolutions and formats differ from those in digital video. Analog video resolutions and formats primarily revolve around standard definition (SD) systems. Here are the commonly used analog video resolutions and formats:

NTSC (National Television System Committee):

- NTSC is an analog video standard used in North America, parts of South America, and some other regions.
- NTSC has a video resolution of 480i, which refers to 480 interlaced scan lines.
- The aspect ratio is typically 4:3, representing the width-to-height ratio of the video frame.

PAL (Phase Alternating Line):

- PAL is an analog video standard used in Europe, Asia, Africa, and parts of South America.
- PAL has a video resolution of 576i, which represents 576 interlaced scan lines.
- Similar to NTSC, the aspect ratio is generally 4:3.

SECAM (Sequential Color with Memory):

- SECAM is an analog video standard used in France, some parts of Eastern Europe, and other regions.
- SECAM has a video resolution of 576i, like PAL, with an aspect ratio of 4:3.

Computer standards – VGA, XGA

VGA (Video Graphics Array) and XGA (Extended Graphics Array) are computer display standards that define the resolution and other specifications for video output. Here's an overview of each standard:

VGA (Video Graphics Array):

- VGA is an analog computer display standard that was introduced by IBM in 1987.
- It has a maximum resolution of 640x480 pixels with a 4:3 aspect ratio.
- VGA supports a 60 Hz refresh rate, which means the display is refreshed 60 times per second.
- It uses an analog RGB (Red, Green, Blue) signal to transmit video information.

XGA (Extended Graphics Array):

- XGA is an extension of the VGA standard, introduced by IBM in 1990.
- It offers higher resolution and improved graphics capabilities compared to VGA.
- XGA has a maximum resolution of 1024x768 pixels with a 4:3 aspect ratio.
- XGA supports a 60 Hz refresh rate, similar to VGA.
- It also uses an analog RGB signal for video transmission.

Both VGA and XGA standards utilize analog signaling and were widely used in older computer systems and displays. However, with the advancements in digital display technology, these standards have become less prevalent.

It's important to note that modern computer display standards, such as DVI (Digital Visual Interface), HDMI (High-Definition Multimedia Interface), and DisplayPort, have largely replaced VGA and XGA. These digital standards offer higher resolutions, faster refresh rates, and support for advanced features like audio transmission and multiple displays.

While VGA and XGA may still be encountered in certain legacy systems or specialized applications, the focus has shifted to digital display technologies for modern computing devices.

Video interfaces in analog video

In analog video, various video interfaces were developed to facilitate the connection and transmission of analog video signals between devices. Here are some commonly used video interfaces in analog video:

Composite Video (RCA):

- Composite video is a widely used analog video interface.
- It uses RCA connectors (typically yellow) to transmit the composite video signal, which combines luminance (brightness) and chrominance (color) information into a single signal.
- Composite video is commonly found in consumer electronics, such as VCRs, DVD players, and older televisions.

S-Video (Y/C):

- S-Video, also known as Y/C, separates the luminance (Y) and chrominance (C) components of the video signal.
- It uses a round mini-DIN connector with multiple pins.
- S-Video provides better image quality compared to composite video by keeping the luminance and chrominance signals separate, reducing color bleeding and enhancing overall sharpness.

Component Video:

- Component video splits the video signal into three separate components: red (Pr), green (Y), and blue (Pb).
- It offers higher image quality and color accuracy compared to composite and S-Video connections.
- Component video uses RCA or BNC connectors and is commonly used in professional video production and high-quality consumer video devices.

VGA (Video Graphics Array):

- VGA is an analog video interface primarily used for computer displays.
- It uses a 15-pin D-sub connector and can carry both video and, in some cases, audio signals.
- VGA supports a wide range of resolutions and refresh rates, making it versatile for various display devices, such as computer monitors and projectors.

SCART (EURO AV):

- SCART is a European analog video interface that combines audio and video signals.
- It uses a 21-pin connector and supports composite video, RGB video, and stereo audio.
- SCART connectors were commonly found on European televisions, VCRs, and other AV equipment.

These are just a few examples of analog video interfaces commonly used in the past. It's important to note that with the transition to digital video, many of these analog video interfaces have been replaced by digital interfaces like HDMI, DVI, and DisplayPort, which offer better quality and compatibility with modern display devices.

Fundamentals of Digital Video:

Digital video is a modern method of representing and transmitting video signals using discrete, digital values. Unlike analog video, which uses continuously varying voltages, digital video breaks down the signal into binary data. Here are the fundamentals of digital video:

- **Pixel Representation**: Digital video represents images as a grid of individual picture elements (pixels). Each pixel is assigned a specific digital value that corresponds to its color and intensity.
- **Color Models**: Common color models used in digital video include RGB (Red, Green, Blue) and YUV. RGB represents color using combinations of red, green, and blue primary colors, while YUV separates luminance (Y) and chrominance (UV) components, facilitating more efficient compression and transmission.

- **Resolution**: Resolution refers to the number of pixels in each dimension of the video frame. Common resolutions include HD (High Definition), Full HD, 4K, and 8K. For example, Full HD (1920x1080) consists of 1920 pixels in width and 1080 pixels in height.
- **Frame Rate**: Frame rate in digital video refers to the number of frames displayed per second (fps). Common frame rates include 24, 30, and 60 fps. Higher frame rates can provide smoother motion but also require more storage and processing power.
- **Bit Depth**: Bit depth, also known as color depth, determines the number of bits used to represent the color of each pixel. Higher bit depth allows for more colors and shades to be accurately represented. Common bit depths include 8-bit, 10-bit, and 12-bit.
- **Compression**: Digital video often employs compression algorithms to reduce file sizes for storage and transmission. Compression can be lossless (no quality loss) or lossy (some quality loss). Common video compression standards include MPEG-2, MPEG-4 (H.264 and H.265), and AV1.
- **Container Formats**: Digital video files are typically stored in container formats like MP4, AVI, MKV, and MOV. These formats can contain video and audio streams, subtitles, metadata, and more.
- **Digital Interfaces**: Digital video signals are transmitted using various interfaces, such as HDMI (High-Definition Multimedia Interface) and DisplayPort. These interfaces ensure high-quality transmission of both video and audio.
- **Digital to Analog Conversion**: To display digital video on analog devices (e.g., older televisions), digital signals need to be converted to analog format using devices like digital-to-analog converters (DACs).
- **Editing and Post-Production**: Digital video's inherent discrete nature makes it highly suitable for nonlinear video editing. Software tools allow for precise editing, effects, color correction, and more.
- **Distribution and Streaming**: Digital video is easily distributable over the internet and other digital platforms. Video streaming services deliver content in real time over networks, and video-sharing platforms allow users to upload and share videos globally.
- **Resolution and Scaling**: Digital video can be scaled to different resolutions for various display devices, but upscaling can lead to loss of quality.
- **Aspect Ratio**: The ratio of the width to the height of the video frame. Common aspect ratios include 16:9 (widescreen) and 4:3 (standard).
- **Metadata**: Digital video files can contain metadata such as title, author, creation date, and more. This information aids in organizing and searching for video content.

Analog vs. Digital Video:

- Analog video is represented by continuous electronic signals that vary in amplitude or frequency. It is the traditional form of video before the advent of digital technology.
- Digital video, on the other hand, uses binary code to represent video information as discrete values (0s and 1s).
- Analog video signals are susceptible to degradation and noise during transmission, resulting in quality loss over long distances. Digital video signals, being discrete, can be transmitted and reproduced without degradation.
- Analog video is more susceptible to interference and signal degradation compared to digital video.

- Digital video offers advantages such as improved image quality, flexibility in editing and processing, and compatibility with digital devices.
-

Colorspace Concepts:

- Colorspace refers to the way colors are represented and encoded in a video signal or digital file.
- **RGB** (Red, Green, Blue) colorspace is commonly used in digital video, where colors are represented as combinations of red, green, and blue components. This is the primary colorspace used in displays and digital image sensors.
- **YUV** (Luma, Chroma) colorspace is often used for video compression and transmission. It separates the luminance (brightness) component (Y) from the chrominance (color) components (U and V).
- **YCbCr** is a specific digital representation of the YUV colorspace, commonly used in digital video formats like MPEG and JPEG.
- Colorspaces also define the color gamut, which is the range of colors that can be represented. Common color gamuts include sRGB, Adobe RGB, and DCI-P3.

The YUV colorspace, also known as YCbCr, is a color representation that separates the image into three components: Y (luma or brightness), U (chrominance or blue projection), and V (chrominance or red projection). It is widely used in video compression and transmission for several reasons:

- **Human visual perception:** YUV is designed to take advantage of the characteristics of human vision. Our eyes are more sensitive to changes in brightness (luma) than changes in color (chroma). By separating the luma and chroma components, it is possible to allocate more bits to the luma information, preserving image quality where our eyes are most sensitive to it.
- **Compression efficiency:** Video compression algorithms, such as those used in codecs like H.264 or HEVC, can exploit the redundancy in the YUV colorspace more effectively. Temporal and spatial redundancies can be better identified and removed, allowing for higher compression ratios without a significant loss in visual quality.
- **Color subsampling:** As mentioned earlier, chroma information can be subsampled at a lower resolution than the luma without noticeable degradation in quality. Common chroma subsampling formats are 4:4:4, 4:2:2, and 4:2:0, which indicate the number of chroma samples compared to luma samples.
- In 4:4:4, there is no chroma subsampling, meaning that both luma and chroma components have the same resolution.
- In 4:2:2, the chroma resolution is halved horizontally, meaning every two pixels share the same chroma information.
- In 4:2:0, the chroma resolution is halved both horizontally and vertically, resulting in a quarter of the chroma information compared to the luma.

Video Coding:

Video coding, also known as video compression or video coding standards, involves the process of reducing the size of digital video data while attempting to maintain acceptable visual quality. Video compression is crucial for efficient storage, transmission, and streaming of video content. Various video coding standards and techniques have been developed to achieve this goal. Here are some key aspects of video coding:

Lossless vs. Lossy Compression:

Lossless Compression: In lossless compression, the original video data can be perfectly reconstructed from the compressed data. This is achieved by removing redundant information without compromising quality.
Lossy Compression: In lossy compression, some data is intentionally discarded to achieve higher compression ratios. This can lead to a loss of visual quality, but modern techniques aim to minimize perceptual impact.

Video Coding Standards:

- **MPEG (Moving Picture Experts Group)**: MPEG has developed several widely used video coding standards, including MPEG-2 (used for DVDs), MPEG-4 Part 2 (used for video streaming and storage), H.264 (also known as AVC, widely used for online streaming), and H.265 (also known as HEVC, offers improved compression efficiency compared to H.264).

Intra-frame and Inter-frame Compression:

- **Intra-frame Compression**: Also known as I-frame compression. Each frame is compressed independently, without reference to other frames. I-frames are useful for seeking within a video but generally result in larger file sizes.
- **Inter-frame Compression**: Also known as P-frame (Predictive) and B-frame (Bi-directional Predictive) compression. Inter-frame compression exploits temporal redundancies by referencing previous and future frames to predict pixel values.

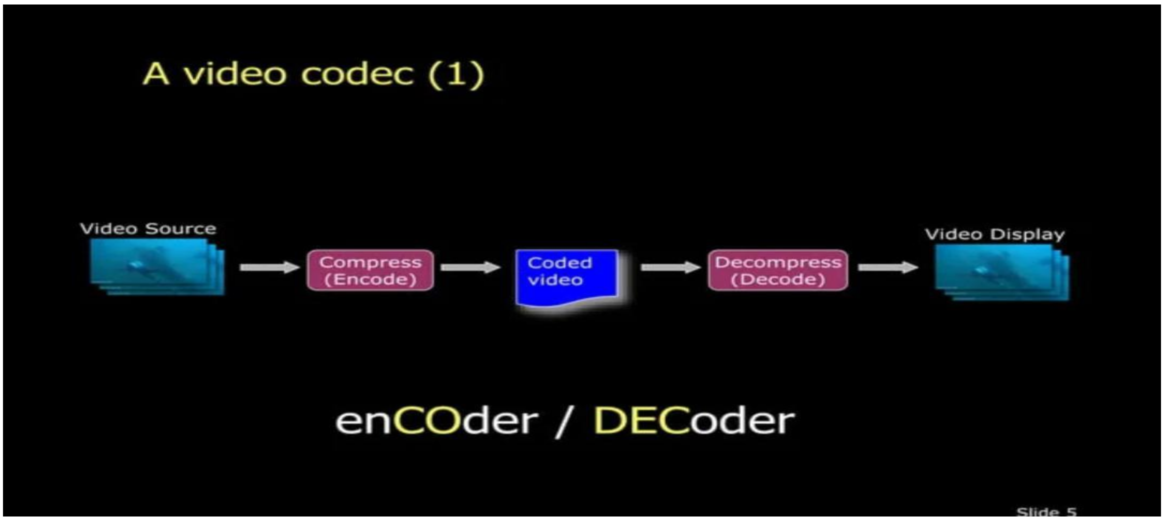
Quantization:

- Quantization involves reducing the precision of pixel values. In lossy compression, higher quantization levels lead to more data reduction but also increased loss of quality.

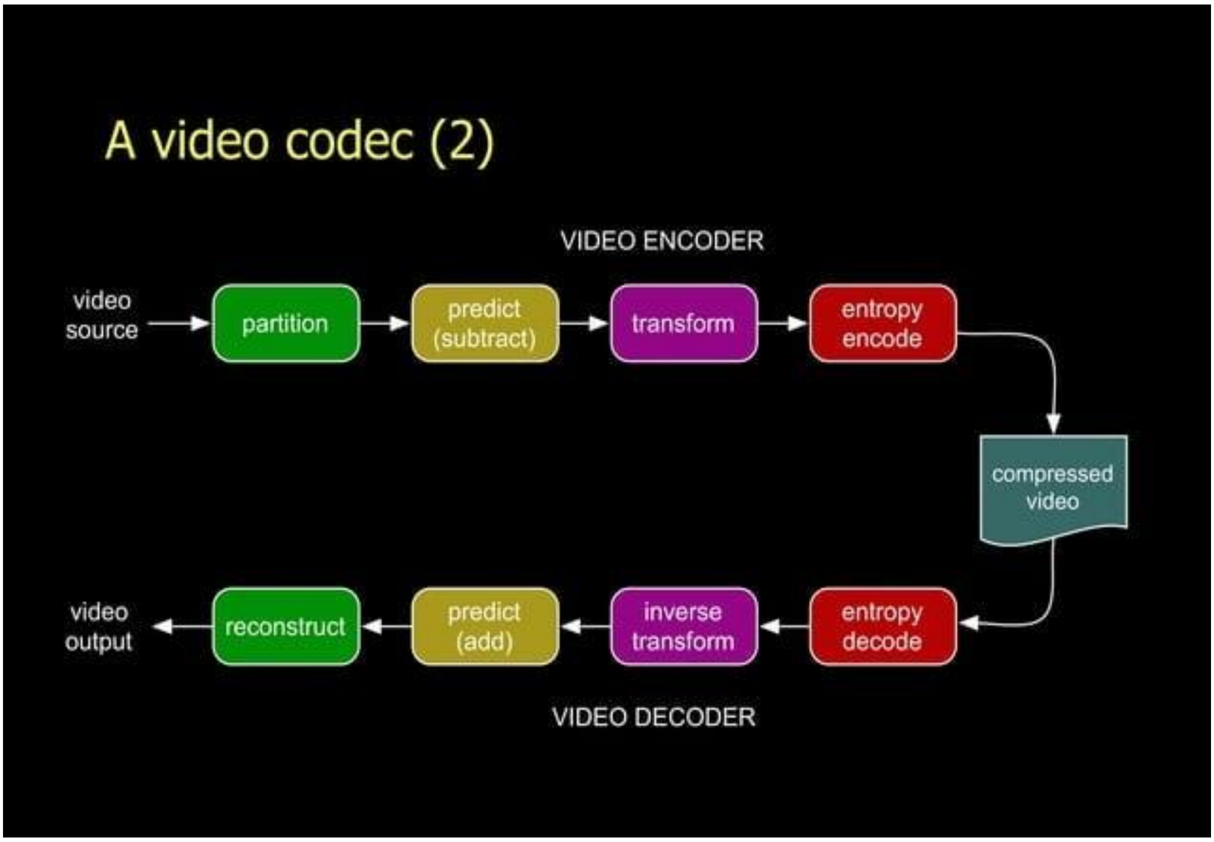
Entropy Coding:

- Entropy coding techniques, such as Huffman coding and Arithmetic coding, are used to represent frequently occurring patterns with shorter codes and less frequent patterns with longer codes.

Flow of video coding:

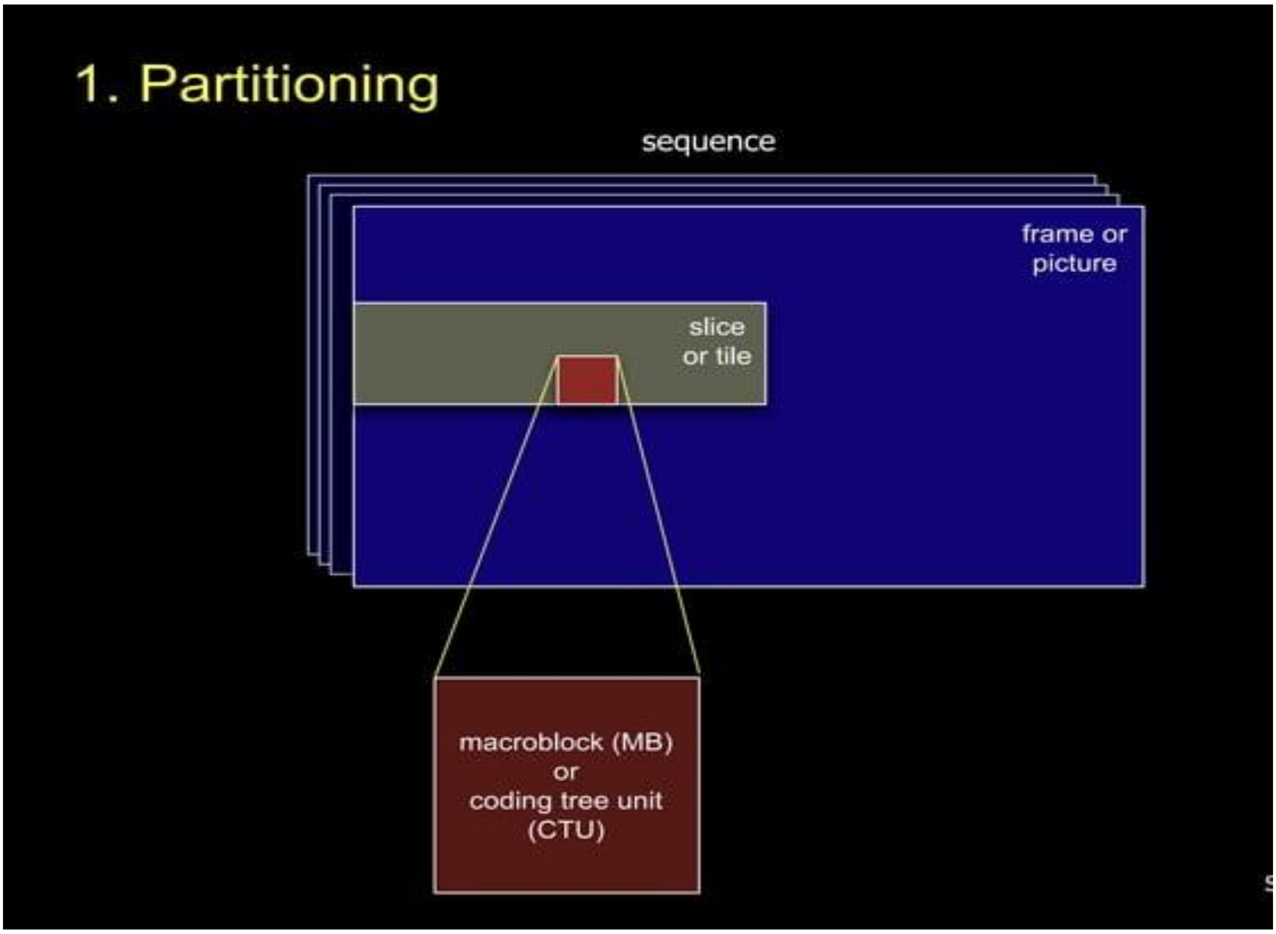


Steps of Video Codec:



Partitioning refers to the process of dividing a video frame into smaller blocks or regions to facilitate efficient compression and encoding. The goal of partitioning is to identify and encode regions of the video frame that have similar visual characteristics, which allows for better compression and reduces the overall data size without significant loss of visual quality.

Partitioning is a crucial step in various video coding standards, such as H.264 (AVC), H.265 (HEVC), and VP9. These standards use various techniques to partition frames, and the most common approach is to divide the frame into a hierarchical structure of blocks. Here are some key concepts related to partitioning in video coding.



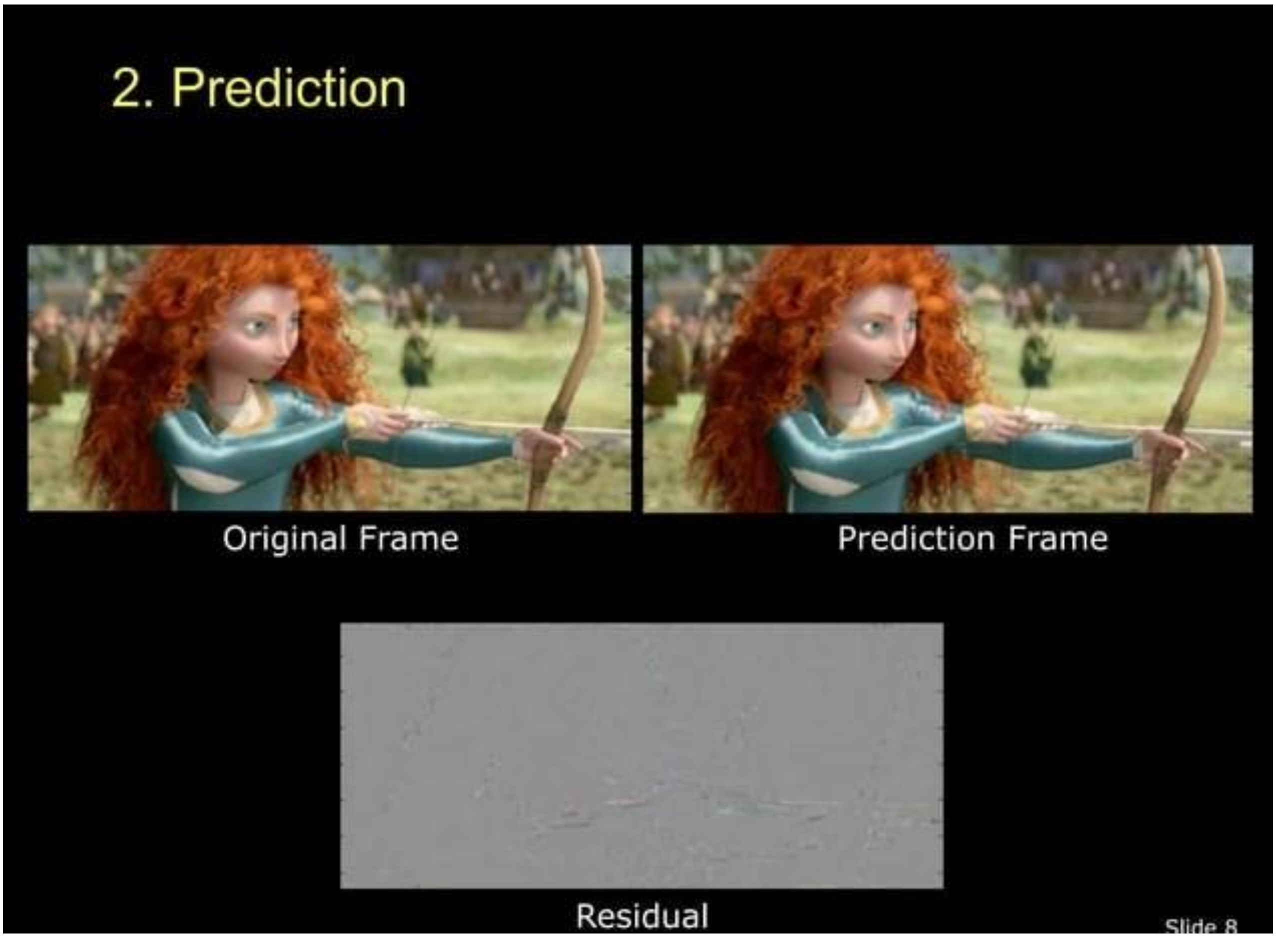
- **Macroblocks (MBs):** In older video coding standards like MPEG-2, a video frame is divided into fixed-size blocks called macroblocks. Each macroblock typically consists of a square region of 16x16 pixels.
- **Coding Tree Units (CTUs)**: Modern video coding standards, like H.265 (HEVC), use a more flexible partitioning approach based on Coding Tree Units (CTUs). The CTU is a larger block that can be subdivided into smaller blocks of varying sizes. The CTU size can be adjusted to optimize the coding efficiency based on the content complexity and motion characteristics of the video frame.
- **Quad-tree partitioning**: One common method used in modern video coding is quad-tree partitioning. In this approach, the frame is recursively subdivided into four equal-sized sub-blocks, and each sub-block is further subdivided if needed. This process continues until the desired block size or a predefined depth level is reached.

Prediction:

In video coding, prediction is a fundamental technique used to efficiently represent video frames by exploiting temporal and spatial redundancies. The main idea behind prediction is to estimate the current video frame's content based on previously encoded frames or neighboring pixels, rather than coding the raw pixel values directly. By predicting the content of the current frame, the video codec can encode the difference between the prediction and the actual frame, which is typically much smaller and requires fewer bits for representation.

There are two main types of prediction used in video coding:

Intra Prediction (Spatial Prediction): Intra prediction is used to exploit spatial redundancies within a single frame. It involves predicting the content of a block of pixels within the current frame using neighboring pixels. Typically, the prediction is based on the surrounding pixels, which are already coded in the same frame. Popular intra prediction modes include horizontal, vertical, and diagonal predictions, as well as various directional predictions.
For example, in a 4x4 block of pixels, the codec can predict the content of each pixel based on its neighboring pixels' intensity values. The differences between the actual pixel values and the predicted values are then encoded and transmitted as residuals.
Inter Prediction (Temporal Prediction): Inter prediction exploits temporal redundancies between consecutive frames in a video sequence. It involves predicting the content of a block of pixels in the current frame using corresponding blocks from previously encoded frames (reference frames). The motion compensation technique is used for inter prediction, which estimates the motion vectors of objects or regions between the reference frame and the current frame.
By shifting and aligning the corresponding block from the reference frame to match the position of the current block, the codec can generate a prediction of the current block. The differences between the actual pixel values and the motion-compensated prediction are then encoded as residuals and transmitted.



Transform And Quantization:

Transform coding is a key technique used in video codecs to achieve efficient compression of video data. It involves transforming the pixel values in a video frame from the spatial domain (original pixel values) to the frequency domain, where redundancies and compression opportunities are more apparent. This transformation is typically achieved using mathematical operations, such as the Discrete Cosine Transform (DCT) or the Discrete Wavelet Transform (DWT).

Here's how transform coding works within video codecs:

Transform Domain Representation:

- The spatial domain represents pixel values in a video frame as they appear on the screen.
- The transform domain represents the pixel values as a combination of different frequency components. High-frequency components represent rapid changes in pixel values, while low-frequency components represent gradual changes and overall brightness.

Discrete Cosine Transform (DCT):

- The DCT is a widely used transformation technique in video codecs, especially in standards like MPEG and H.264.
- It converts a block of pixel values into a set of coefficients that represent the amount of each frequency component present in the block.
- DCT coefficients tend to concentrate energy in a few low-frequency coefficients, allowing for efficient quantization and compression.

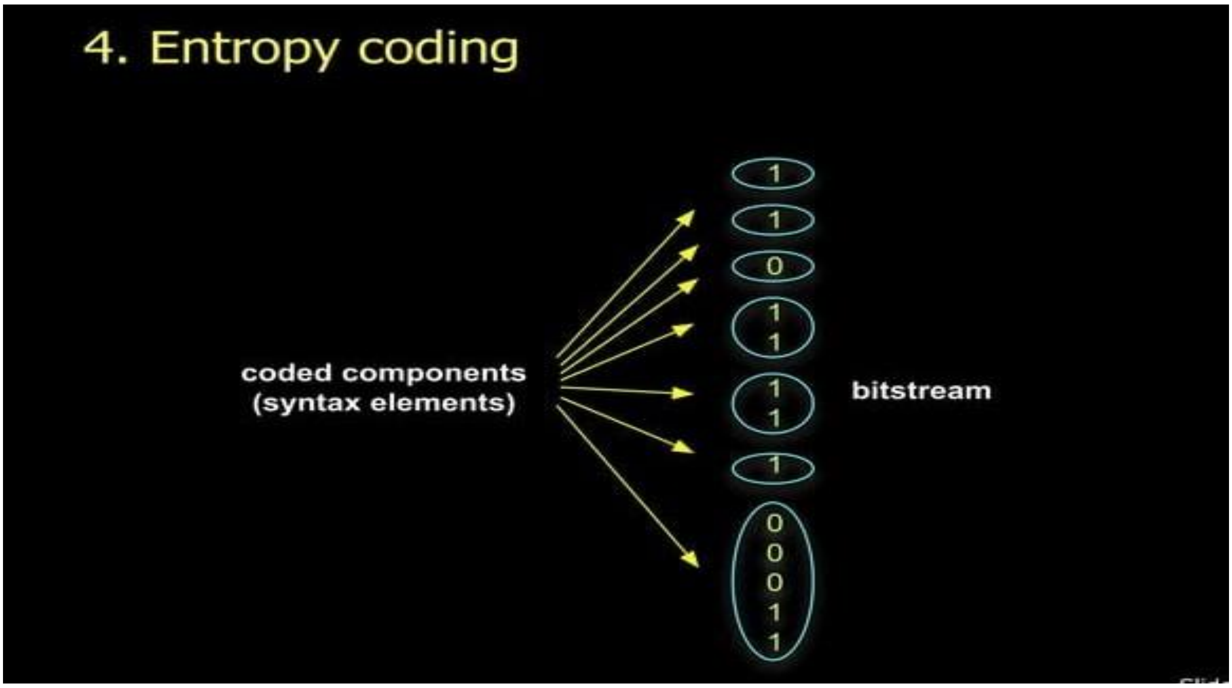
Discrete Wavelet Transform (DWT):

- The DWT is another transform technique used in video codecs, prominently in newer standards like H.265 and AV1.
- It decomposes an image into multiple sub-bands at different scales, capturing both horizontal and vertical frequency information.
- Wavelet coefficients can be better localized in both space and frequency, potentially improving compression efficiency.

Quantization:

Quantization is the process of reducing the precision of the transformed coefficients by mapping them to a smaller set of discrete values. In other words, it introduces loss of information by approximating the coefficients with a quantization step size or quantization step table.
Quantization is a crucial step in video coding because it reduces the number of bits required to represent the transformed coefficients significantly. High-frequency coefficients, which often contribute less to the overall visual quality, can be quantized more aggressively, leading to fewer bits allocated for their representation.

Entropy encoding :



Entropy encoding is the final stage in the video coding process, following prediction, transform, and quantization. It is a crucial step that further compresses the data by assigning shorter codes to more frequent symbols and longer codes to less frequent symbols. Entropy encoding exploits the statistical properties of the data, specifically its redundancy, to achieve compression without losing any information.

In video coding, the transformed and quantized coefficients, as well as other data such as motion vectors and mode information, are represented as symbols that need to be transmitted or stored. These symbols can have different probabilities of occurrence, with some being more likely to appear than others. Entropy encoding helps in reducing the average number of bits needed to represent each symbol, resulting in more efficient data representation.

There are two common types of entropy encoding used in video coding:

Huffman Coding: Huffman coding is a widely used entropy encoding technique. It constructs variable-length codes, with shorter codes assigned to more frequent symbols and longer codes assigned to less frequent symbols. The probability distribution of the symbols is used to build a Huffman tree, where frequently occurring symbols are placed closer to the root of the tree, and less frequent symbols are placed farther away.
The video codec uses the Huffman tree to assign unique binary codes to each symbol. As a result, more common symbols are represented using fewer bits, leading to higher compression efficiency. During video decoding, the receiver uses the same Huffman tree to decode the encoded symbols back into their original values.

Symbols (like intensity levels)	Probabilities (sorted)	Source Reduction (do till two values are left) (Maintain in sorted order here as well)			
		1	2	3	4
a2	0.4 1	0.4	0.4	0.4	0.6 0
a6	0.3 00	0.3	0.3	0.3 00	0.4 1
a1	0.1 011	0.1	0.2 010	0.3 01	
a4	0.1 0100	0.1 0100	0.1 011		
a3	0.06 01010	0.1 0101			
a5	0.04 01011				

Encoded String: 010100111100

Decoding : a3 a1a2 a2 a6

Parameters:

1. Average length of code

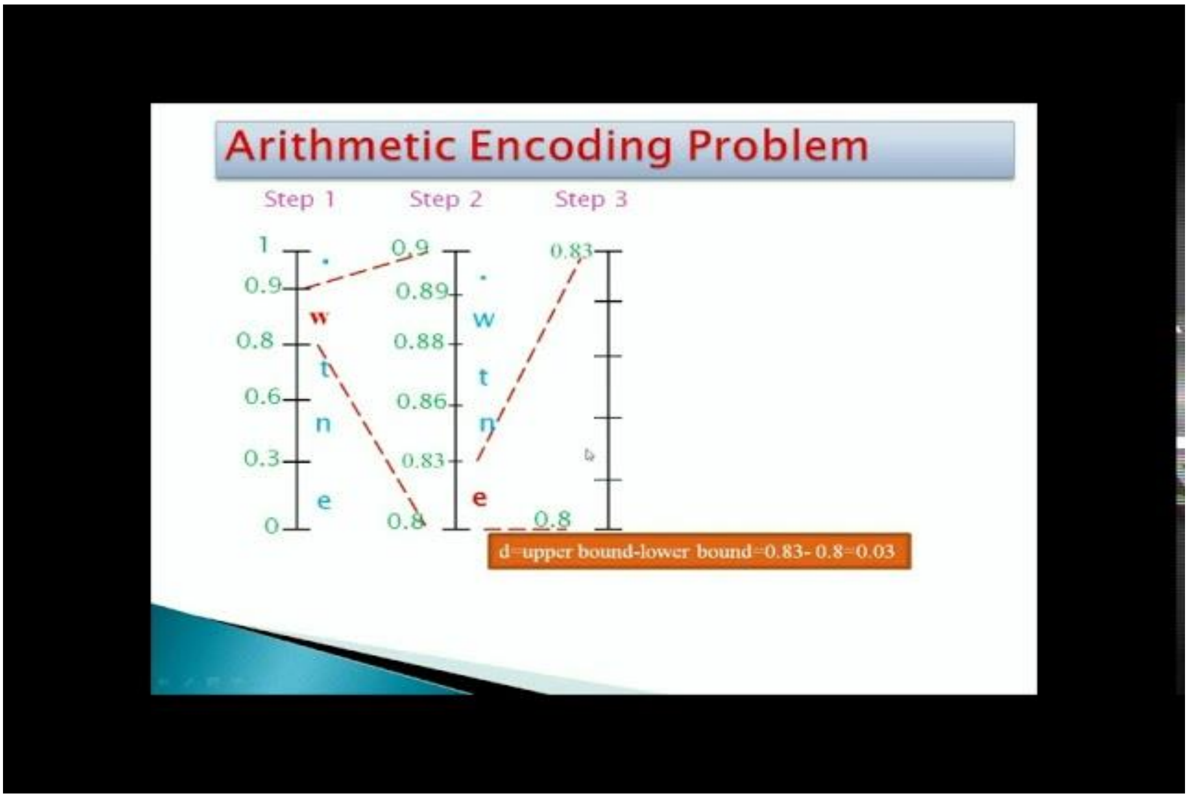
   Lavg = 0.4 * 1 + 0.3 * 2 + 0.1 * 3 + 0.1 * 4 + 0.06 * 5 + 0.04 * 5 = 2.2 bits/symbol
2. Total no. of bits to be transmitted

   10 * 10 * 2.2 = 220 bits
3. Entropy = 2.1396
4. How much you saved = $\frac{10 * 10 * 5 - 10 * 10 * 2.2}{10 * 10 * 5}$ = 0.56 = 56%

Arithmetic Coding: Arithmetic coding is another advanced entropy coding technique that can achieve even higher compression ratios than Huffman coding. Unlike Huffman coding, where each symbol corresponds to a fixed-length code, arithmetic coding represents the entire data stream as a single fractional number within a specified range.
The probability distribution of the symbols is used to partition the range into subintervals, with each subinterval representing a symbol. The encoder then narrows down the range to the subinterval corresponding to the next symbol, and this process continues for each symbol. The resulting fractional number is then converted into a binary representation for transmission or storage.

Arithmetic coding can be more efficient than Huffman coding because it can take advantage of more precise probabilities and use fractional bits, allowing for more accurate representation of the data. However, it is computationally more complex than Huffman coding.

Both Huffman coding and arithmetic coding are lossless compression techniques, meaning that no information is lost during the encoding and decoding process. They are widely used in video coding standards like H.264 (AVC), H.265 (HEVC), and VP9 to achieve high compression efficiency while maintaining the original video quality.



Most video coding methods exploit both temporal and spatial redundancy:

Absolutely correct! Most video coding methods, such as those based on modern video compression standards like H.264, H.265 (HEVC), and VP9, are designed to exploit both temporal and spatial redundancies in video data to achieve high compression efficiency. Let's explore what these redundancies mean:

- **Temporal Redundancy:**
- Temporal redundancy refers to the similarity between consecutive frames in a video sequence.
- In many videos, successive frames are often very similar, especially in scenes with slow or no motion.
- Rather than coding each frame independently, video codecs can exploit temporal redundancy by only encoding the differences (motion vectors) between consecutive frames.
- This technique is known as inter-frame or temporal prediction, and it significantly reduces the amount of data needed to represent the video.
- **Spatial Redundancy:**
- Spatial redundancy refers to the similarity between pixels within a single frame.
- In most images and videos, neighboring pixels often have similar color or intensity values, leading to spatial redundancy.

- Video codecs use spatial redundancy reduction techniques, such as transform coding (e.g., Discrete Cosine Transform - DCT) and quantization, to concentrate the image energy in fewer coefficients and then code those coefficients efficiently.
- Additionally, spatial prediction is used within a frame, predicting certain blocks or regions based on neighboring blocks, and only encoding the differences.

What is motion vector in video codec?

In video codecs, a motion vector is a critical element used in motion compensation, which is a technique employed to reduce temporal redundancy in video data. Motion vectors are essential for predicting the motion of objects or regions between consecutive frames in a video sequence.

Here's how motion vectors work in video codecs:

Motion Estimation: The process begins with motion estimation, where the codec analyzes the current frame (P-frame or B-frame) and searches for a matching region in the reference frame (I-frame or previously encoded frame). This search is performed in a predefined search area surrounding the current block or macroblock (a block of pixels) in the P-frame or B-frame.
Motion Vector Calculation: The motion estimation process finds the best matching region in the reference frame, and the motion vector is then calculated as the displacement between the current block's position in the P-frame or B-frame and the corresponding matching block's position in the reference frame.
Motion Compensation: Once the motion vector is determined, the codec can now predict the motion of the current block or macroblock in the P-frame or B-frame based on its position in the reference frame. The codec then encodes the motion vector and the residual (difference) between the predicted and actual pixel values of the current block.

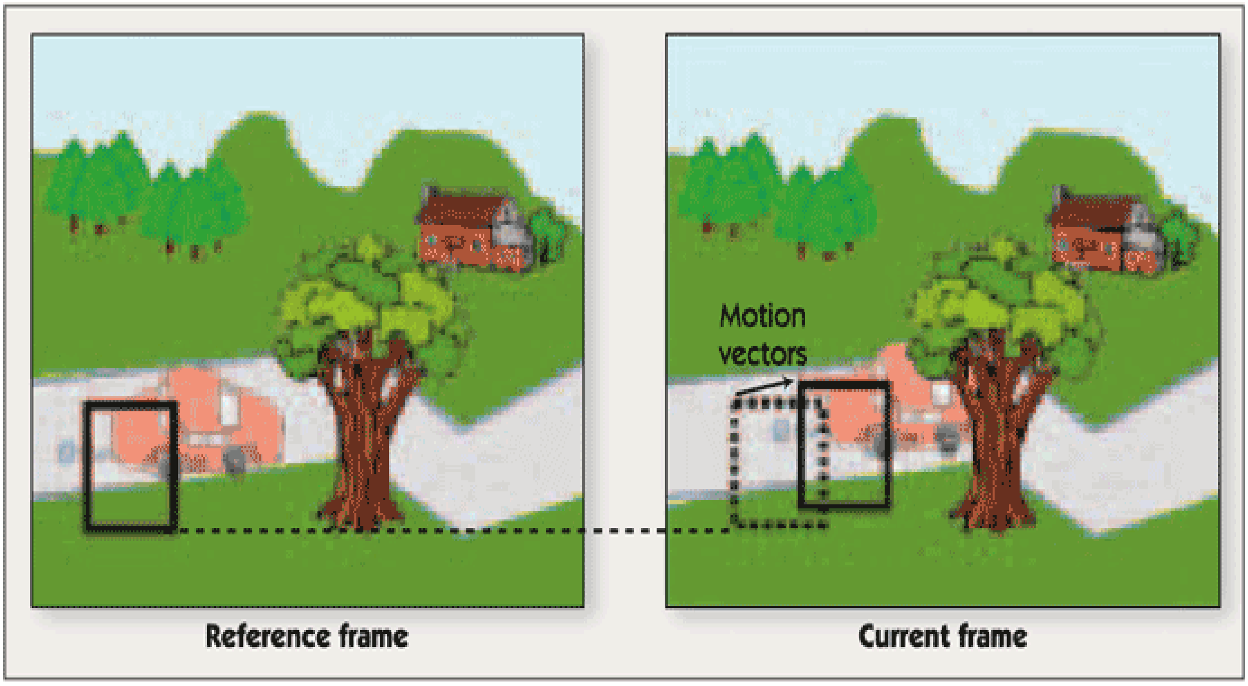
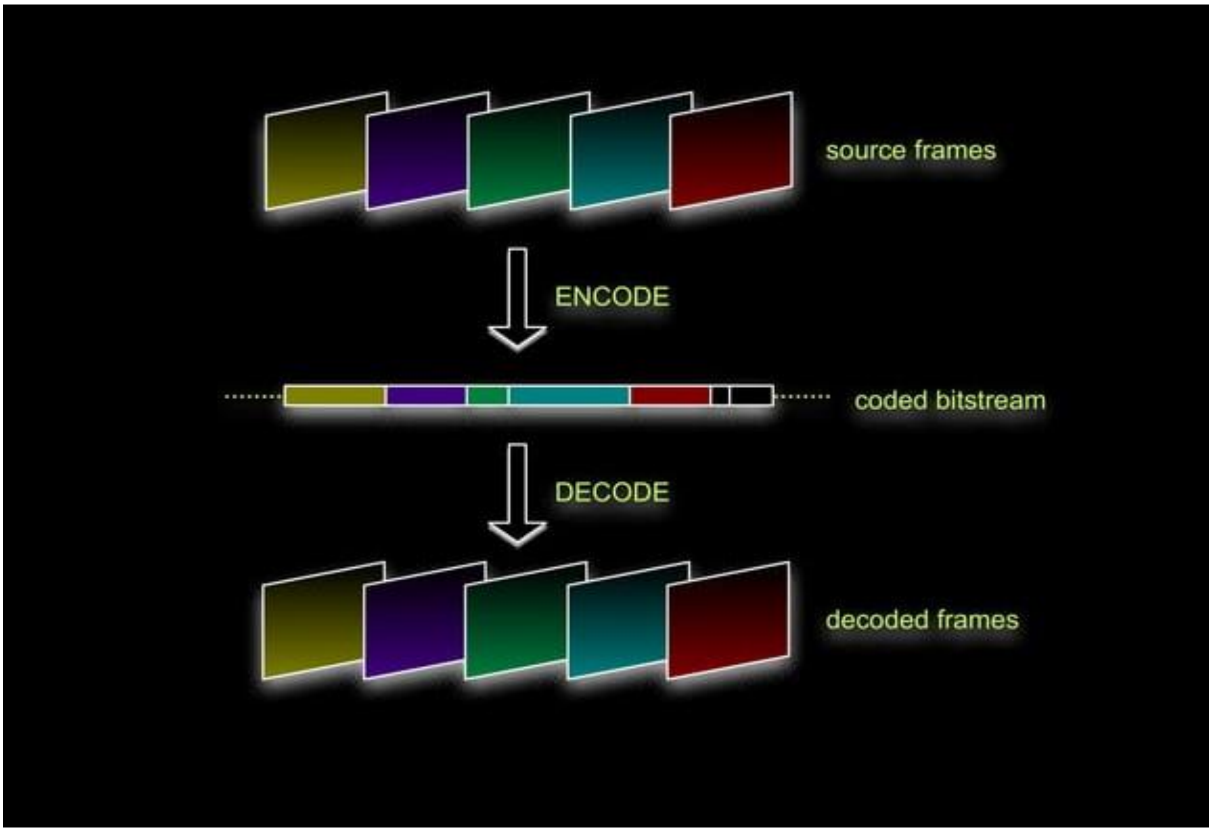
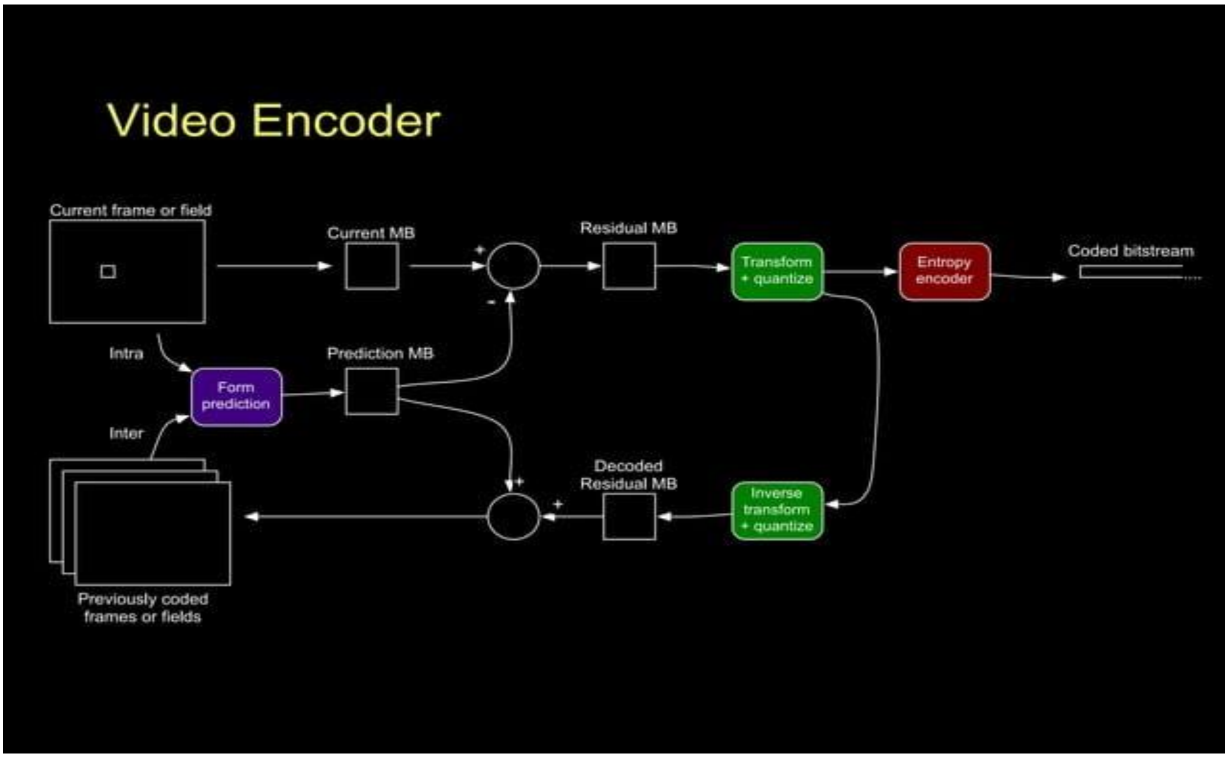


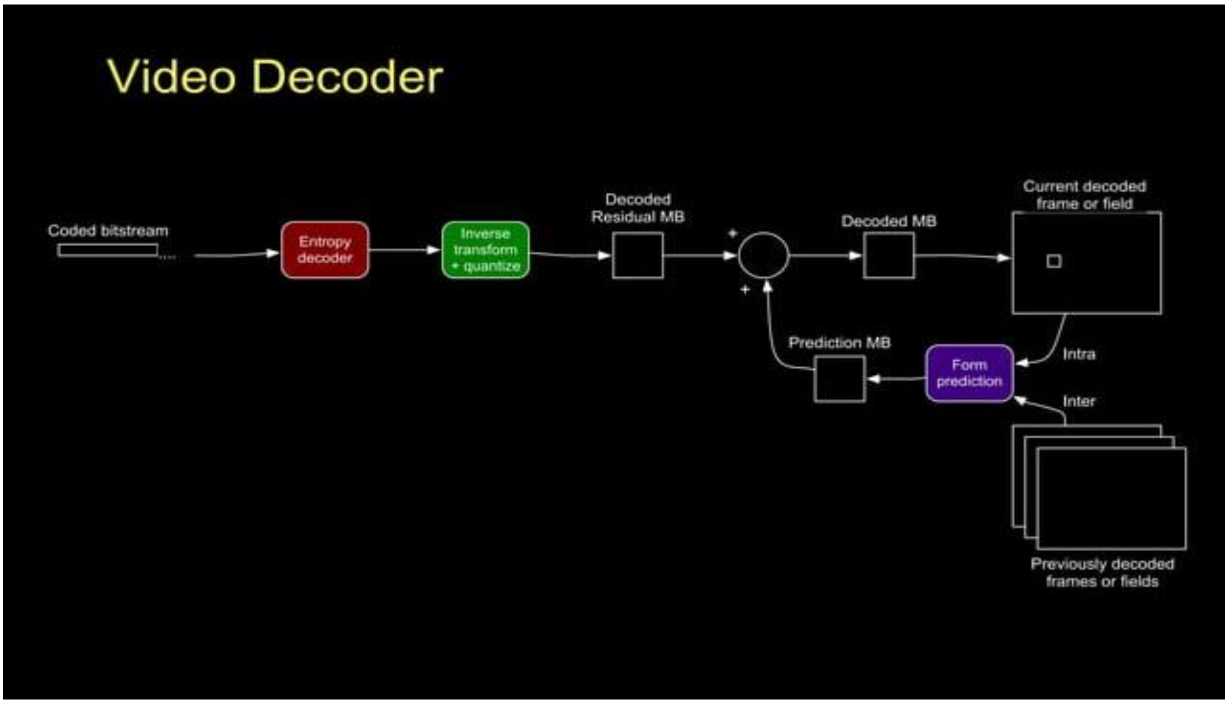
Fig. 4 Motion estimation predicts the contents of each macroblock based on motion relative to a reference frame. The reference frame is searched to find the 16x16 block that matches the macroblock, motion vectors are encoded, and the difference between predicted and actual macroblock pixels is encoded in the current frame.



Video Encoder :



Video Decoder :



Motion Compensation Block Size

In video coding, motion compensation block size refers to the size of the individual blocks or regions used for motion estimation and motion compensation. These blocks are typically square or rectangular and represent the basic units over which motion is estimated and compensated.

The motion compensation block size is a critical parameter in video coding, and its choice can impact the overall compression efficiency and video quality. Common block sizes used in video coding standards include:

- **16x16 (16 by 16 pixels):**
- This block size is one of the most widely used in video coding, especially in older standards like MPEG-2 and H.264 (AVC).
- It provides a balance between capturing fine details and reducing the computational complexity of motion estimation.
- However, using a larger block size can lead to less accurate motion estimation for regions with complex motion.
- **8x8 (8 by 8 pixels):**
- This block size is commonly used in newer video coding standards, such as H.265 (HEVC) and VP9.
- Using smaller block sizes allows for better capturing of fine motion details, which is crucial for achieving higher compression efficiency and better video quality.
- However, using smaller block sizes can increase the computational complexity of motion estimation.
- **4x4 (4 by 4 pixels):**
- Very small block sizes, such as 4x4, are used in some advanced video coding techniques, like High Efficiency Video Coding (HEVC) with Transform Unit (TU) partitioning.
- These small block sizes help capture intricate motion details and texture variations, leading to even higher compression efficiency.
- However, using extremely small block sizes can significantly increase the computational complexity, and it is often used in conjunction with more substantial

Sub-pixel Motion Compensation:

Sub-pixel Motion Compensation (SPMC) is a technique used in video compression and motion estimation to improve the accuracy of motion compensation by allowing motion vectors to represent fractional pixel displacements. In traditional motion compensation, motion vectors are typically integers, indicating the pixel displacement between consecutive frames. However, in real-world video sequences, objects may move between pixels, leading to inaccuracies in motion estimation and potentially introducing artifacts in the compressed video.

SPMC addresses this limitation by refining the motion vectors to sub-pixel precision, enabling more accurate motion compensation and better video quality. Instead of restricting the motion vectors to integers, SPMC allows for fractional values (e.g., 0.5, 1.25) to represent sub-pixel displacements.

There are various algorithms to achieve sub-pixel motion compensation. One common approach is interpolation. After obtaining the integer motion vectors through traditional motion estimation algorithms, such as block matching, the SPMC algorithm refines the motion vectors by interpolating the pixel values at sub-pixel positions. Common interpolation methods include bilinear interpolation and bicubic interpolation, which use neighboring pixels to estimate the pixel value at the sub-pixel position.

By applying sub-pixel motion compensation, the encoder can more accurately estimate and compensate for the motion in the video frames, leading to reduced artifacts and improved video quality.

Bilinear interpolation and Bicubic interpolation:

Bilinear interpolation and bicubic interpolation are two common methods used in image processing and computer graphics for estimating pixel values at non-integer coordinates based on the values of neighboring pixels. Both techniques are often used in sub-pixel motion compensation, as mentioned earlier, and in various other applications where accurate pixel value estimation is required.

- **Bilinear Interpolation:** Bilinear interpolation is a simple and efficient method for estimating pixel values at sub-pixel positions. Given four neighboring pixels, the method computes a weighted average of their values based on the relative distances between the target point and each of the four pixels.
- **Bicubic Interpolation:** Bicubic interpolation is a more advanced and sophisticated method for estimating pixel values at non-integer coordinates. It uses a cubic polynomial to perform the interpolation, considering more neighboring pixels compared to bilinear interpolation.

Scalar Quantisation and vector Quantisation:

Scalar quantization and vector quantization are two different techniques used in data compression and signal processing to reduce the amount of data required to represent information.

Scalar Quantization:

- Scalar quantization is the simplest form of quantization and is applied to individual data points (e.g., samples from an analog signal).
- In scalar quantization, each data point is approximated by a discrete value from a finite set of quantization levels.
- The goal of scalar quantization is to reduce the number of bits required to represent each data point while minimizing the overall distortion (error) introduced by the quantization process.
- Scalar quantization is commonly used in image and audio compression, speech coding, and various other applications.

Vector Quantization:

- Vector quantization is a more advanced technique that groups multiple data points into clusters or code vectors and represents each cluster by its centroid (average) or a representative code vector.
- Unlike scalar quantization, vector quantization deals with sets or blocks of data points rather than individual data points.
- Vector quantization can achieve higher compression efficiency compared to scalar quantization since it takes advantage of correlation and redundancy in the data.

- The process of vector quantization involves codebook design, where the code vectors are created based on the training data to represent different clusters effectively.
- Vector quantization is commonly used in image and video compression, data compression, speech recognition, and other applications where blocks of data are processed.

Run-Level Encoding

As of my last update in September 2021, "Run-Level Encoding" is not a well-known term in the context of computer science or information technology. It is possible that this term has emerged or become more relevant after my last update. However, based on the information available up to that point, I can provide some general information about run-length encoding, which might be what you are referring to:

Run-Length Encoding (RLE) is a simple form of data compression that is often used to reduce the size of repetitive sequences of data. It is particularly effective when dealing with data that contains long sequences of repeated values.

In RLE, rather than storing each data value separately, the algorithm represents the data as a sequence of run-lengths and corresponding values. For example, instead of representing the sequence "AAAAABBBBBCCCCDDDDDD," RLE would encode it as "5A5B4C6D."