Chapter 2

Preliminaries and Background

This chapter presents some basic definitions and background related to video coding. The basic understanding of those specificities will be helpful to the readers to interpret the rest of the thesis. Some important and relevant features as well as parameters corresponding to video coding are discussed at the beginning. Then basic overview of video coding mechanism is presented. Finally, the HEVC video encoding process is discussed in brief.

2.1 Video Characteristics

Although every digital video is a sequence of images, their characteristics may differ. Different characteristics like resolution, frame-rate, format and color depth, etc. are used to categorize the video sequences.

Table 2.1: Different video formats and their resolutions

Video format	**Resolution**
Standard Definition (SD)	720 × 526
720p High Definition (HD)	1280 × 720
1080p HD	1920 × 1080
Ultra-High Definition (UHD)	3840 × 2160
2160p 4K UHD	4096 × 2160
8K UHD	7680 × 4320

2.1.1 Video Resolution

As stated earlier video is a sequence of digital images. Any digital image is an array of color pixels. The number of pixels per image is known as resolution usually quoted as **width × height**. For example, the resolution of "720 × 526" represents an array of pixels having 720 rows (width) and 526 columns (height). Table 2.1 shows different video formats and their resolutions.

2.1.2 Frame-rate

The optical illusion persists in Human Visual System (HVS) for a short period of time after removal of visual object and the phenomenon is known as persistence of vision. The persistence of vision limits the number of images per second HVS can recognize as an individual image. It is perceived as motion when a sequence of images is displayed beyond this limit. The number of images or frames displayed per second in a sequence is known as frame-rate usually measured as Frames per Second (FPS).

Frame-rate is an important aspect of the video encoding process because high frame-rate increases the amount of raw data and decreases the difference between consecutive frames in the sequence. Video quality, bit-rate required and the throughput of the encoding process proportionally depends on the frame-rate. Frame-rate depends on the application and its typical value is 24, 25, 30, 50, 60 or 120 fps.

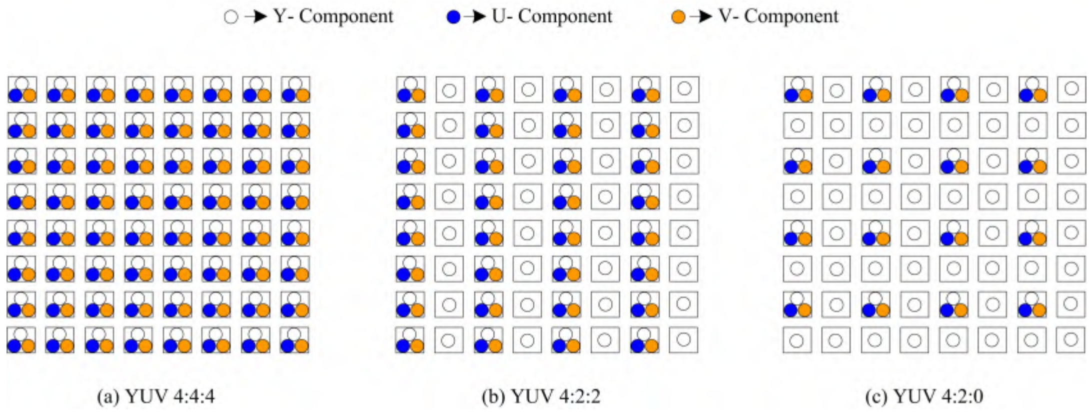
2.1.3 Color Space

Colour image is usually an array of color pixels and each color pixel comprises of three different color components for a specific color space used. There are several color space formats which are chosen depending on the applications. Among them, the RGB format is likely the most popular in which a pixel is represented by the combination of red, green and blue color components. The most common color space for video compression is YUV, where 'Y' represents luminescence and 'U' as well as 'V' represent chrominance components. YUV is an invertible transform of the RGB color space [86] and is computed from RGB information as follows:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.1}$$

HVS is more sensitive to the 'Y' and less sensitive to 'U' and 'V' components. Using this limitation of HVS, chroma components are encoded with reduced resolution to save bit-rate and storage area requirements.

There are different YUV color space formats with different sub-sampling mechanism of chroma components. Fig. 2.1 shows few common YUV sampling patterns like 4:4:4, 4:2:2, and 4:2:0. In the 4:4:4 sampling pattern, for every square consisting



(a) YUV 4:4:4

(b) YUV 4:2:2

(c) YUV 4:2:0

FIGURE 2.1: Chroma sub-sampling

of four ‘Y’ samples (two-by-two), there are also four ‘U’ and four ‘V’ samples. It means the number of ‘Y,’ ‘U,’ and ‘V’ samples are exactly the same in both vertical and horizontal directions. Hence, this format fully retains the fidelity of the chroma components and it has the same effect as the RGB [86]. In YUY2 or 4:2:2 formats, the number of ‘Y,’ ‘U,’ and ‘V’ samples are same in the vertical direction, but only half ‘U’ and ‘V’ samples are present in the horizontal direction. This pattern is used for high-quality color reproduction. The most popular chroma sub-sampling pattern is 4:2:0, where the number of samples for each of the ‘U’ and ‘V’ components is half in the vertical as well as horizontal direction as compared to that of the ‘Y’ component. In other words, for every four ‘Y’ samples, there exist only one ‘U’ and ‘V’ samples.

2.1.4 Bit-depth

The intensity of a sample in luma and chroma planes varies from complete dark to maximum bright, and the amount of brightness is represented by digits. The number of bits used to represent the intensity of a sample is known as bit-depth or color depth. More number of tones can be represented with higher bit-depth. However, it increases the bit-rate and storage size.

2.2 Video Quality Assessment

Assessment of any video coding methodology is done by the quality of the video it produces. Therefore, perceptual quality assessment is an important requirement in video processing technology. The performance comparison of different codecs under similar conditions and measuring the influence of different aspects in a codec are the main motives behind video quality assessment.

Both subjective and objective methods are available to assess the quality of video [87]. The most accurate way of assessing video quality is to ask a viewer. In the subjective method, a group of viewers under different circumstances are used to assess the video quality. However, visual perception is influenced by several factors and highly difficult to model. Subjective assessment tests are costly, time taking, and difficult to use in real time [87]. Hence, objective assessment method is preferred over the subjective assessment.

Objective assessment is performed by computing the difference of the video under test with a reference one. The most widely used objective assessment metrics are Mean Squared Error (MSE) and Peak Signal to Noise Ratio (PSNR), which are defined as follows:

$$\begin{aligned} MSE &= \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2 \\ PSNR &= 10\log_{10}\frac{(2^b - 1)^2}{MSE}, \end{aligned} \tag{2.2}$$

where N is the total number of samples, b is the bit-depth, x_i and $\hat{x_i}$ are i-th sample in the original and the processed signals, respectively.

There are few other evaluation metrics for video quality assessment. However, MSE and PSNR are widely used because they are simple to calculate, have clear physical meanings, and are mathematically easy to deal with for optimization purposes.

2.2.1 Bjøntegaard Algorithm for Video Quality Comparison

There is a trade-off between the quality of a video and the bit-rate required to encode it. A high-quality video always requires high bit-rate. A video coding algorithm which produces better video quality (i.e., PSNR) with low bit-rate is definitely better than others. But, it is often essential to compare the performance of the video coding algorithms which produce different bit-rate with different PSNR values for the same raw input video. In this case, only PSNR comparison is not enough and bit-rate also must be considered for effective comparison. In this situation, the Bjøntegaard Delta metric is applied. Gisle Bjøntegaard proposed a curve fitting algorithm [88] to compare the coding efficiency of two different algorithms. It computes the average bit-rate and PSNR differences between two rate-distortion curves obtained by coding the same video sequence but, using different algorithms. The report of the model is represented in two different ways as follows:

1. Bjøntegaard delta PSNR (BD-PSNR): It computes the average PSNR difference between two algorithms for the same bit-rate.

2. Bjøntegaard delta rate (BD-Rate): It computes the average bit-rate difference between two algorithms for the same PSNR.

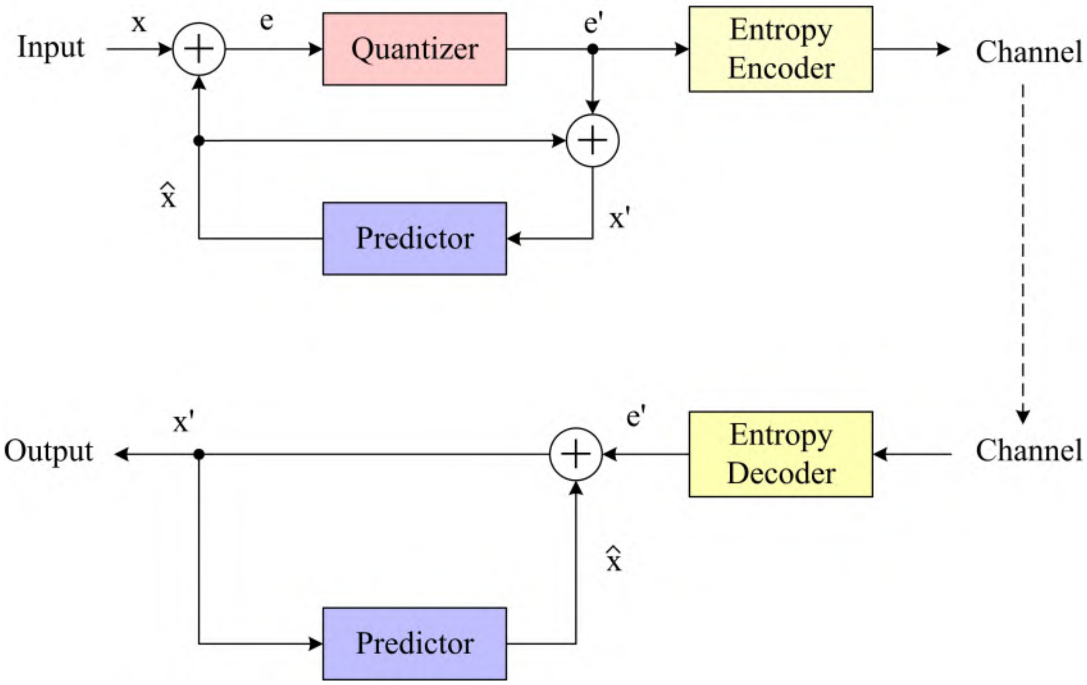


FIGURE 2.2: Block diagram of Differential Pulse Code Modulation (DPCM) technique

In depth discussion about BD-PSNR and BD-rate calculations is available in [88], [89].

2.3 Video Coding Architecture

In recent time, most of the image and video compression algorithms employ some sort of predictive coding, wherein a particular pixel is predicted using correlation among the neighboring pixels. The predictive coding approach, to compress images or video, works on the general principle of DPCM [90]. In DPCM, the present value of the signal is predicted from the past value. Thereafter, the prediction error resulting from the difference between the original signal and the predicted signal is encoded. Fig. 2.2 shows the block diagram of lossy compression using DPCM approach. In this figure, x, $\hat{x}$ and e represents original, predicted and error signals, respectively.

Encoding the prediction error (or the residual signal) instead of the original signal requires lesser number of bits. Hence, instead of the original signal x, the error signal e is quantized (e'), entropy coded and transmitted through the channel. In lossy compression approach, the residual signal is quantized to convert a predefined range of values into a single value. Entropy coding is performed wherein quantized residual signal is converted to the fewer number of bits by exploiting statistical redundancy.

The decoder performs the entropy decoding to recover the quantized residual signal (e'). A prediction method similar to encoder side is applied to estimate the present value from previously decoded signal value. The reconstructed signal is obtained by summing the decoded quantized residue signal e' with the prediction signal $\hat{x}$.

Most of the video coding algorithms apply the prediction mechanisms based on the basic principle of DPCM. In general, two types of prediction method are used: intra-prediction and inter-prediction. When a pixel of a frame is predicted from the adjacent pixels of the same frame, it is termed as intra-prediction. This type of prediction method removes the spatial redundancy without reference to any other frame. Currently, all the image coding algorithms use intra-prediction for data compression.

The inter-prediction technique uses inter-frame similarities and removes the temporal redundancy. In this prediction method, the pixels of the target frame are estimated from the co-located pixels of the previously coded frames known as reference frames. There are three different methods of inter-prediction, which are based on the location of reference frames [91] as mentioned below.

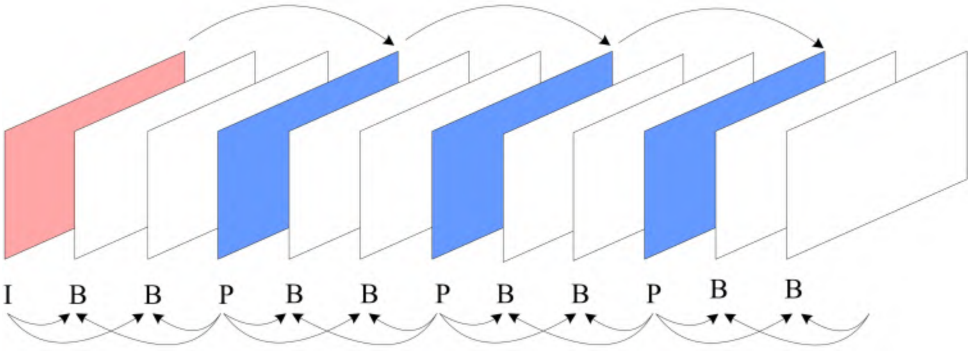


FIGURE 2.3: Group of Pictures (GOP)

- **Forward prediction:** The reference frame arrives temporally before the current frame.

- **Backward prediction:** The reference frame arrives temporally after the current frame.

- **Bidirectional prediction:** Out of two reference frames, one arrives temporally before and the other arrives temporally after the current frame. Sometime more than two frames are also used in bidirectional prediction.

A video sequence is a series of still images and hence, can be coded using only the intra-prediction method. But, image is a 2D signal and due to temporal information video is a 3D signal. A significant amount of compression efficiency can be achieved by exploiting the temporal redundancy that exists in each video sequence. Hence, to remove spatial as well as temporal redundancy, intra- and inter- prediction is applied to a Group of Pictures (GOP). A GOP is a collection of sequential frames in an encoded video sequence, as shown in Fig. 2.3. Either intra- or inter- prediction is applied to each frame in GOP according to their order of arrival.

Three different types of frame are used in the GOP and those are as follows:

- **I-frame:** The first frame of the GOP is known as I-frame. Only intra-prediction is applied to this frame because previously coded reference frames are not available for inter-prediction.

- **P-frame:** These are the inter predicted frames from another I-frame or P-frame.

- **B-frame:** These frames are bi-directional inter predicted frames from more than one I- and/or P-frames.

2.4 Operation of the Hybrid Video Codec

The hybrid video coding technique is the most popular method to efficiently compress a video sequence and is recommended by ITU-T and ISO. Since H.261, almost all video coding algorithms use the hybrid video coding technique [92]. Although the basic structure has remained the same, the algorithms have been refined with more additional features and increased flexibility over the years. The coding technique is called hybrid as it uses both the spatial and temporal prediction between consecutive frames as well as it uses transform coding to compress the prediction errors [93]. Transformed coefficients are quantized in lossy video coding to remove the irrelevant information. Thus, a hybrid video codec is able to eliminate the spatial, temporal and perception redundancy from a video sequence.

2.4.1 Operation of the Video Encoder

The basic block diagram of a hybrid video encoder is shown in Fig. 2.4. The encoder works on block-based coding method and each frame of the input video

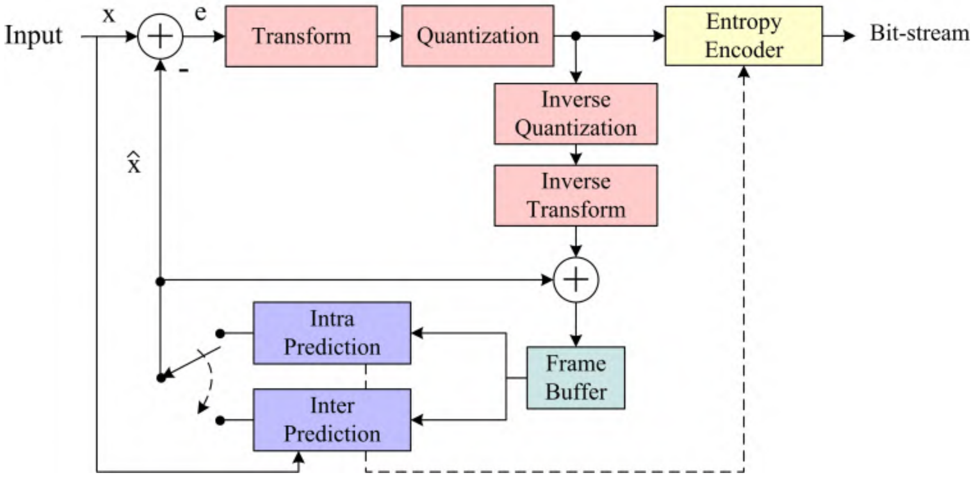


Figure 2.4: Basic block diagram of hybrid video encoder [92]

signal is segmented into several non-overlapping blocks. The encoder processes all the blocks one by one.

Either intra- or inter- prediction is applied to each of the blocks, as shown in Fig. 2.4. Next, the prediction signal ($\hat{x}$) is subtracted from the input signal (x) and the resultant prediction error known as residual signal (e) is transformed, quantized, and entropy coded into the bit-stream. A control engine is also necessary for the encoder implementation, which generates different control signals to take certain decisions like selection of optimum prediction mode, different prediction and filtering parameters, applicable quantization parameters. The building blocks to generate these control signals is not shown in Fig. 2.4. These control information are also necessary to the decoder side. Hence, control signals and parameters are also encoded into bit-stream before sending it over the channel.

To the encoder side, plenty of alternatives are available for encoding a frame depending on the features supported by a codec. It is possible to select the best mode

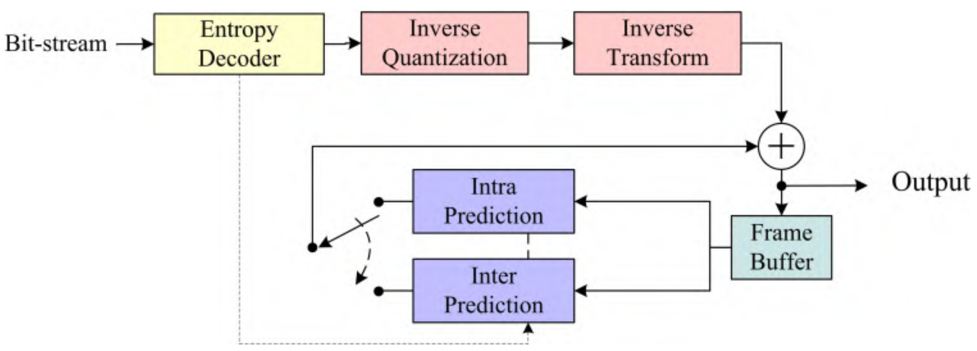


FIGURE 2.5: Basic block diagram of hybrid video decoder [92]

among all the alternatives in each step of video coding. RDO algorithm [94] is adopted to take all these decisions.

In general, RDO algorithm exploits the trade-off between bit rate and distortion. The video sequence may be coded with a low bit rate but at the expense of possible high error propagation. On the other hand, the higher bit-rate produces low error vulnerability. The trade-off between bit rate and distortion can be formulated by Lagrangian cost function [94]:

$$J = D + \lambda.R, \tag{2.3}$$

where $\lambda \geq 0$ denotes the so-called Lagrange multiplier, R denotes the bit rate and D is the distortion measure.

2.4.2 Operation of Video Decoder

The decoder performs the inverse operation to that of the encoder and recovers the video signal. The block diagram of a hybrid video decoder is shown in Fig. 2.5.

First, the entropy decoder decodes the bit stream. Next, inverse quantization and inverse transform are applied sequentially and the residual signal is obtained. This residual signal is added with the available prediction signal to reconstruct the video sequence. Sometimes different loop filtering (e.g., a deblocking filter) techniques are applied [92] to eliminate the distortion. The reconstructed picture is stored in the decoded picture buffer and fetched from the buffer during prediction operation, if required.

2.5 Overview of HEVC

HEVC (H.265), is the successor of the most popular video codec H.264/AVC and is developed by the JCT-VC [91]. One of the main objectives of HEVC is to support the emerging UHD video with reduced bit-rate. In comparison to H.264/AVC, the coding structure of HEVC is more flexible and hence, requires 50% less bit-rate for the same perceptual quality [11]. HEVC adopted hybrid video coding structure with a large number of coding features such as flexible block-based coding structure, large number of prediction modes, novel filtering methods, new entropy coding scheme and different parallelization tools.

A large number of new features are adopted in HEVC to cater a wide range of video applications. All of those features may not be needed for a particular application. Therefore, HEVC introduces different profiles with a different set of coding tools. According to the first version of the HEVC draft [95], there are three profiles as mentioned below:

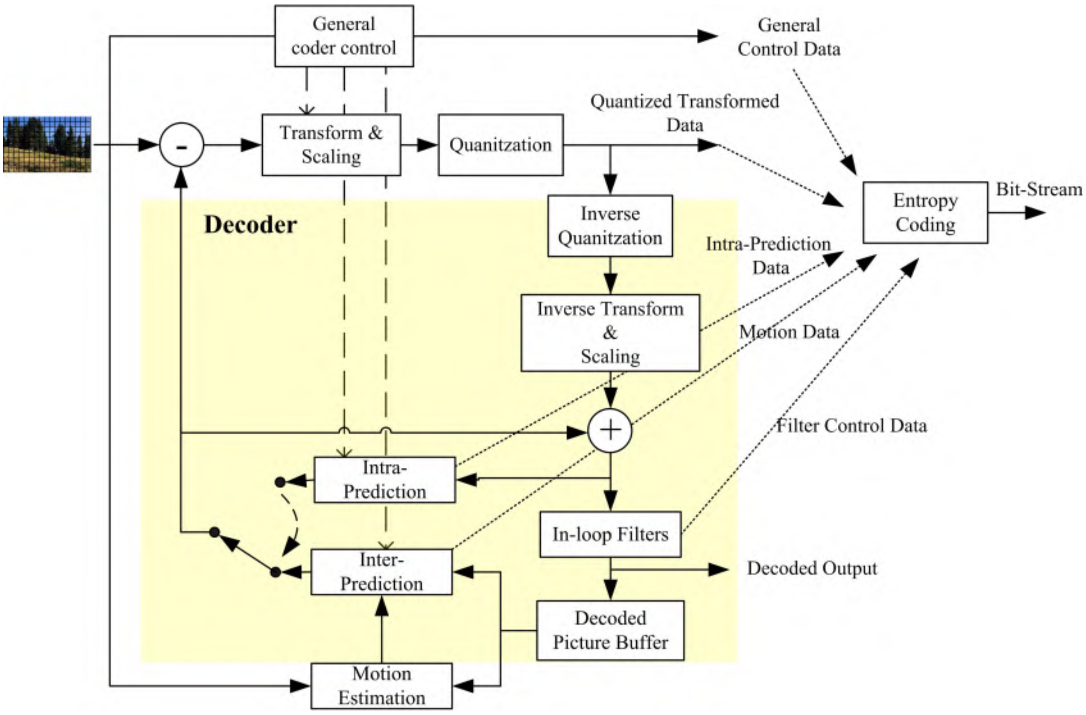


FIGURE 2.6: Encoding mechanism of HEVC [11]

- **Main:** The use of the main profile is limited to the sequences with bit-depth of 8 and 4:2:0 color space. It is the most common type of sequences used in video applications.
- **Main Still Picture:** This profile is a subset of the main profile. It is used to encode a single still picture. So, inter picture prediction is not used.
- **Main 10:** This profile is a superset of main profile and supports more color sheds, better brightness dynamic range and hence, improved video quality as compared to Main profile. It supports sequences with bit-depth up to 10 in 4:2:0 color space.

An abstract level block diagram of HEVC coding mechanism is shown in Fig. 2.6. It is clear from Fig. 2.6 that HEVC follows the same hybrid video coding approach as

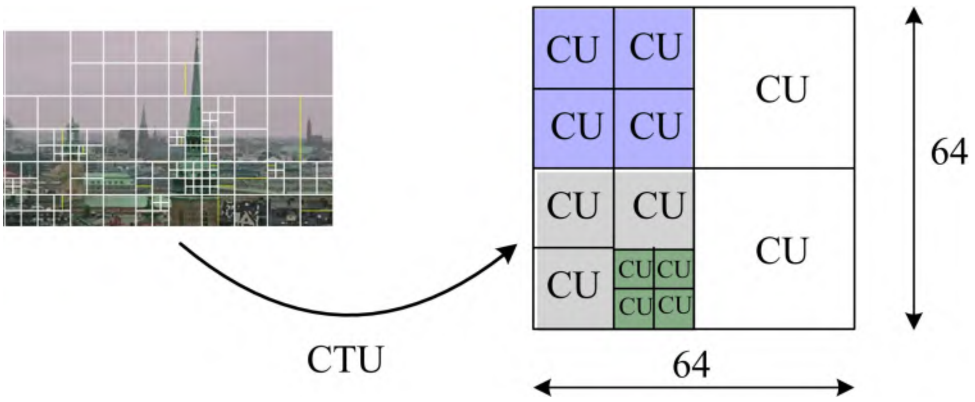


Figure 2.7: A Coding Tree Unit (CTU) structure

used in the other standards, i.e., inter-prediction, intra-prediction and 2D transform.

The block-based coding method is the baseline for the HEVC. Each frame of a sequence is partitioned into several block based regions which are processed independently [91]. HEVC has adopted a highly efficient and flexible block partitioning method by introducing various type of blocks such as Coding Tree Unit (CTU), Coding Unit (CU), Prediction Unit (PU) and Transform Unit (TU). A frame at the input is partitioned into blocks of size 64×64, called as CTU. Each CTU consists of a luma and two chroma Coding Tree Blocks (CTB). As per the HEVC main profile, each CTU is further split into multiple CUs of different sizes which may vary from 8×8 to 64×64. Fig. 2.7 shows the CTU and its quadtree partitioning method into several CUs. Each and every CU is processed individually and is further partitioned into PU during prediction and into TU during transform depending on the characteristics of the contents.

HEVC supports multiple PU sizes to capture the content characteristics as well as to increase prediction efficiency. Various size of PUs used in HEVC, for both intra- and inter- prediction, are shown in Fig. 2.8.

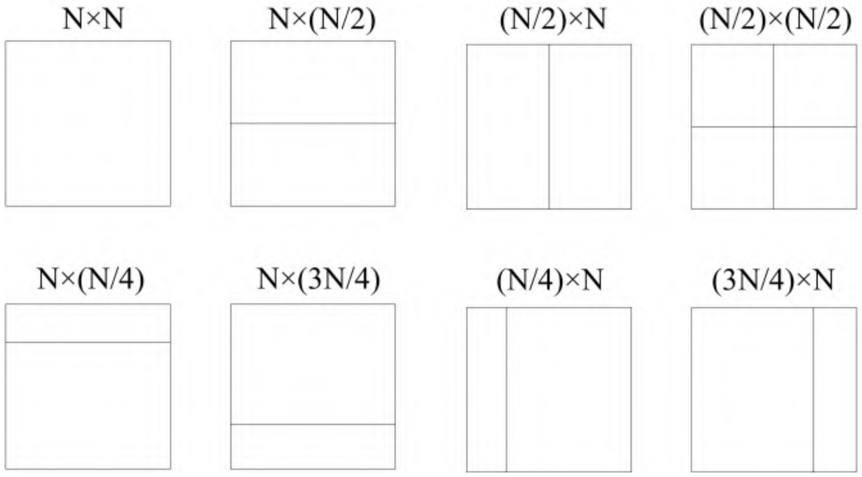
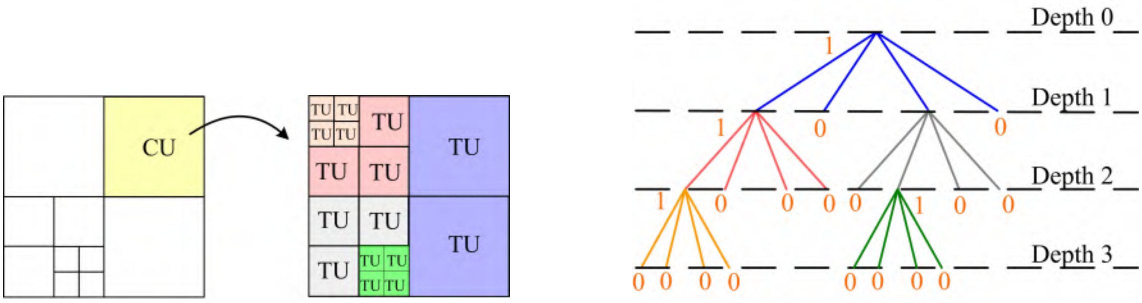


Figure 2.8: Splitting of Coding Unit (CU) into Prediction Unit (PU) [91]

After prediction, the resultant residue (i.e., error signal) is transformed into the frequency domain using 2D integer transform which is an approximation of DCT. To adequately capture the characteristics of the residual signal, another quadtree based partitioning method is adopted in the HEVC draft. This Residual Quadtree (RQT) based partitioning method produces TUs of different size which varies from 4×4 to 32×32. Transform and quantization process is applied to each TU. Fig. 2.9(a) illustrates the TU partitioning within a CU for a HEVC encoded video frame, whereas Fig. 2.9(b) represents the RQT of the same partitioning method [91].

In summary, CU, PU and TU within a CTU form the flexible coding structure in HEVC. This highly flexible coding structure is one of the primary reasons behind the superior coding performance of the HEVC. But, it is also one of the main causes for high complexity of HEVC codec.



(a) (b)

Figure 2.9: (a) Transform Unit (TU) partitioning and (b) Residual Quad Tree (RQT) inside a Coding Unit (CU) [91]

2.5.1 HEVC Prediction Modes

The prediction mode for a particular PU is decided by the RDO method. To obtain the best RD performance, the HEVC encoder performs an exhaustive search with all the possible combinations and apply the best prediction mode for a particular PU [91]. A PU may either be selected for intra- or inter- prediction to minimize the RD cost.

2.5.1.1 Intra-prediction

In all the intra-prediction modes, a block is predicted from the neighboring samples from the already reconstructed blocks. Thirty-five different intra-prediction modes are supported by HEVC [96] and those are shown in Fig. 2.10. Among them, 33 are angular predictions, whereas the remaining two are DC and planer predictions. Table. 2.2 shows the name and the index value corresponding to different intra-prediction modes. This is the standard convention used in the HEVC draft [95]. Details of HEVC intra-prediction can be found in [96], [91].

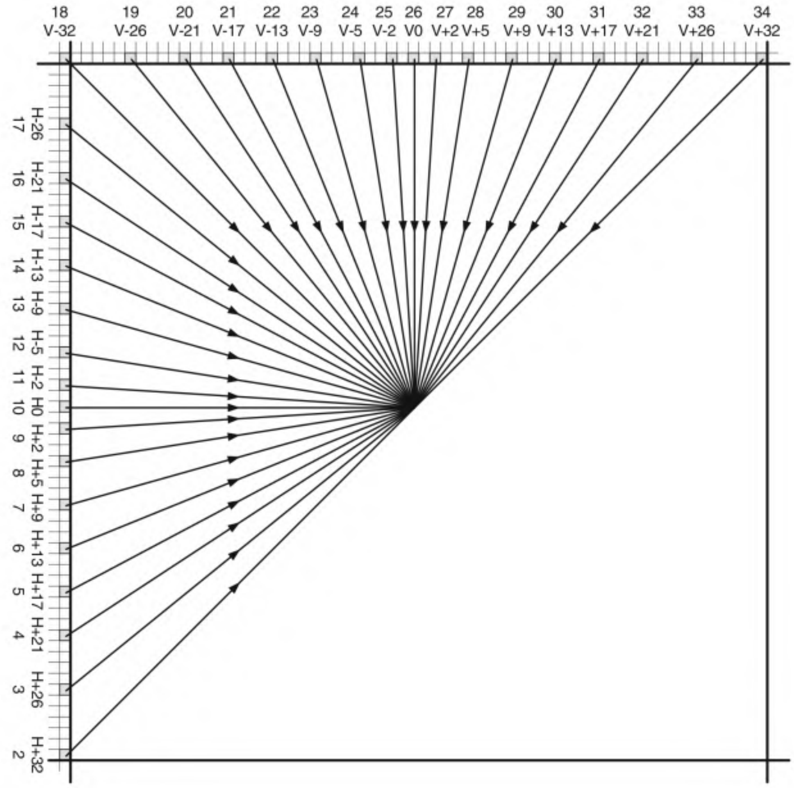


FIGURE 2.10: HEVC intra-prediction modes [96]

TABLE 2.2: Different intra-prediction modes with associated numbers [96]

Mode number	Mode name
0	INTRA_PLANAR
1	INTRA_DC
2,...,34	INTRA_ANGULAR[i], i=2,...,34

2.5.1.2 Inter-prediction

Inter prediction in HEVC exploits the correlation between adjacent frames and derive a Motion Compensated Prediction (MCP) for a prediction block. A block in the previously decoded frame can be found in the subsequent frames and hence, motion estimation is performed to estimate the position coordinate known as a motion vector. To obtain such vectors, the block of the current frame is compared with the reference frames. Advanced Motion Vector Prediction (AMVP) is introduced in HEVC which uses a Motion Vector Competition (MVC) scheme [91]. In this scheme, a predicted motion vector from a set of spatial and temporal motion vectors is picked for a particular PU. Skip and merge mode are the other two modes which are applied to any PU. A detailed discussion about the inter-prediction method in HEVC can be found in [91].

2.5.1.3 Transform and quantization

After the intra- and inter- prediction process, the residuals are coded using 2D transform [91]. The residual frame is split into TU and a scaled approximation of DCT is applied at each TU. Size of the DCT may vary from 4×4 to 32×32. Additionally, a scaled approximation of the Discrete Sine Transform (DST) is also used for 4×4 intra predicted luma block as an alternative transform. Further details of HEVC transform have been discussed in the subsequent chapters of this theses.

HEVC quantization process is similar to that of H.264/AVC, where a Quantization Parameter (QP) is used to select the quantization step size. For HEVC, QP varies in the range of 0–51 for 8-bit video sequences and quantization step size gets doubled whenever the QP value increases by six [66]. The QP value can be transmitted (in the form of delta QP) for a quantization group as small as 8×8 samples for rate control and perceptual quantization purposes. HEVC also supports frequency-dependent quantization by using quantization matrices for all transform block sizes. The details regarding HEVC transform and quantization can be found in [66].

2.5.1.4 In-loop filtering

After the inverse transform, filtering operations are performed on the reconstructed samples to remove various types of noise introduced during the coding process. HEVC encoder and decoder use two types of in-loop filters for this purpose. These are known as de-blocking filter [97] and Sample Adaptive Offset (SAO) filter [98]. The de-blocking filter is used to remove the blocking artifacts arisen due to block-based coding. SAO is newly introduced in HEVC [91]. Just after the de-blocking filter, SAO is applied to reduce the mean sample distortion of a region.

2.5.1.5 Entropy coding

Entropy encoding is a lossless compression method which uses the statistical properties to remove the redundancy. It is the last stage of video encoding and the first stage of the video decoding process. HEVC supports Context Adaptive Variable Length Coding (CAVLC) [99, 100] as well as Context Adaptive Binary Arithmetic Coding (CABAC) [101] for low and high complexity entropy coding, respectively. HEVC uses a variation of CABAC used in H.264/AVC with the parallelization technique to increase throughput.

2.6 Summary

Determining the characteristics of the video sequence is the key to assess a video coding algorithm. Different video coding algorithms are compared based on the quality of video sequence they produced. In this chapter, a few fundamental concepts related to video coding are introduced. Some parameters and specifications related to video quality assessment are also defined. Video quality assessment process is also briefly discussed. The basic building blocks of most of the video coding algorithms are the same. Hence, the block diagram of a hybrid video coding process is discussed in this chapter. Finally, block diagram of HEVC and its different features are presented as an example of the hybrid video coding process.