CHAPTER 2

VIDEO COMPRESSION TECHNIQUES AND FUNDAMENTALS

2.1 INTRODUCTION

Digital video refers to capturing the stream of pictures at regular intervals of time, manipulation and storing of video in digital format. The raw video contains large amount of data so it requires more memory to store the video sequence. So the need to reduce the amount of data required to store the data is balanced by compressing the digital video using some algorithm. All video compression algorithms are evolved to reduce the correlation that is found on the video signal. This chapter gives introduction about video compression techniques and fundamentals.

2.2 SPATIAL AND TEMPORAL SAMPLING OF VIDEO SEQUENCES

Representing a visual scene from the real word in digital form is nothing but the sampling the real scene spatially and temporally. The spatial and temporal sampling of video sequence is shown in Figure 2.1. Each sample represented in the spatial plane is referred as pixels (picture elements). The brightness or luminance and colour of the scene are determined by the value associated with the pixels. The common format for a sampled image is a rectangle with the sampling points positioned on a rectangular grid. The visual quality of the image is greatly influenced by number of sampling points.
on the grid. If it is a coarse sampling, resolution is low and for fine sampling, resolution of the frame is high. The number of pixels in the row and column of the grid determine the spatial resolution of the frame. The minimum temporal sampling rate of the video signal is 25 frames per second. In order to get smoother motion in the video sequence, the frame rate is increased up to 60 frames/second.

Figure 2.1 Spatial and temporal sampling of frame

2.3 COLOUR SPACES AND CONVERSION

2.3.1 RGB Format

For the monochrome image, the pixel is represented with one value that is luminance component. But for color images, three values are required to represent a pixel. The three values are red, green and blue component (RGB). All the colors represented in the real world scene can be reproduced from these primary colours (red, green and blue). The scene in RGB colour format is nothing but the array of pixels with different levels of primary colours.
In RGB color format, there are 255 different levels for each color component so 8 bits are required to represent each colour component and to represent a pixel 24 bits are needed.

2.3.2 YCbCr Format

Our human visual system is more sensitive to brightness (luminance component) than the chrominance component (colour). So to achieve compression, more importance is given to encode luminance than chrominance component. But in RGB color format, all the components are given equal importance. In YCbCr(Luminance, Chrominance blue, Chrominance red) format, the luminance component (Y) is separated from the chrominance component represented as (Cr, Cb, Cg) and higher preference is given to encode the Y component. Y is calculated as weighted average of RGB component as given in Equation (2.1)

\[ Y = k_r R + k_g G + k_b B \]  

(2.1)

where \( k \) is the weighting factor \( (k_r + k_g + k_b = 1) \). The value of \( K \) as per the ITU recommendation is \( k_b = 0.114 \) and \( k_r = 0.29 \). The color information can be represented as difference between the R, G, B component and Y.

\[ C_b = B - Y \]  

(2.2)

\[ C_r = R - Y \]  

(2.3)

\[ C_g = G - Y \]  

(2.4)

In order to display the color image, it is necessary for us to convert the YCbCr format in to RGB format. The equation for converting the RGB to YCbCr and vice versa is given in Equation (2.5) to (2.10).
$$Y = 0.299R + 0.587G + 0.114B$$ \hspace{1cm} (2.5)

$$C_b = 0.564 (B - Y)$$ \hspace{1cm} (2.6)

$$C_r = 0.713 (R - Y)$$ \hspace{1cm} (2.7)

$$R = Y + 1.402Cr$$ \hspace{1cm} (2.8)

$$G = Y - 0.344C_b - 0.714C_r$$ \hspace{1cm} (2.9)

$$B = Y + 1.772C_b$$ \hspace{1cm} (2.10)

YC\_bC\_r is the popular method to efficiently represent the color image.

### 2.4 YC\_bC\_r SAMPLING FORMAT

There are three variations in the sampling format of YC\_bC\_r supported by standard video coders which are listed below and more information related to the sampling format can be found in Richardson & Iain (2002) & (2003).

1. YC\_bC\_r -4:4:4
2. YC\_bC\_r - 4:2:2
3. YC\_bC\_r -4:2:0

#### 2.4.1 YC\_bC\_r-4:4:4

It means that for the group of four pixel samples, all the component say Y, C\_b and C\_r are of same resolution. For a 176x 144 image with 4:4:4 sampling format, if it is represented with 8 bits per component then the image size is given as 176x144x8x3 = 608256 bits. The Figure 2.2 shows the sampling format of YC\_bC\_r-4:4:4.
In 4:2:2 sampling format for every four luminance pixel in the horizontal direction there will be two $C_b$ and $C_r$ pixel. For a 176x 144 image with 4:2:2 sampling format, if it is represented with 8 bits per component then the image size is given as $176 \times 144 \times 8 + 88 \times 72 \times 8 \times 2 = 304128$. The Figure 2.3 shows the sampling format of YCbCr -4:2:2. For high quality reproduction 4:2:2 sampling format is used.
2.4.3 YCbCr-4:2:0

In 4:2:0 sampling format, for every four luminance pixel in the horizontal direction there will be one C_b and one C_r pixel. For a 176 x 144 image with 4:2:0 sampling format, if it is represented with 8 bits per component then the image size is given as 176x144x8+44x36x8x2 =228096. The Figure 2.4 shows the sampling format of YC_bC_r-4:2:0. This format is widely used in consumer applications like video conferencing, DVD and digital television.

![Figure 2.4 4:2:0 sampling format](image)

2.5 VIDEO FORMAT

In practice, it is common that prior to compression, the captured video sequences are converted in to some common intermediate format (CIF). The resolutions of different CIF format and their usage is listed in the Table 2.1.
Table 2.1 Common video formats

<table>
<thead>
<tr>
<th>S.No</th>
<th>Type of CIF</th>
<th>Resolution</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4CIF</td>
<td>704 × 576</td>
<td>Standard-definition television and DVDs</td>
</tr>
<tr>
<td>2</td>
<td>CIF</td>
<td>352 × 288</td>
<td>Video conferencing</td>
</tr>
<tr>
<td>3</td>
<td>QCIF</td>
<td>176 × 144</td>
<td>Video conferencing</td>
</tr>
<tr>
<td>4</td>
<td>SQCIF</td>
<td>128 × 96</td>
<td>Mobile multimedia applications</td>
</tr>
</tbody>
</table>

2.6 VIDEO QUALITY MEASUREMENT

There are two different ways to measure the video quality

1. Subjective quality measurement
2. Objective quality measurement

2.6.1 Subjective Quality Measurements

Subjective video quality defines the video quality based on the perception of the user. There are several factors that influence the subject or user while judging the quality. Few of them are listed below.

1. Viewing environment
2. Observer state of mind (Active or Passive)
3. Visual attention

Different testing methods for subjective quality measurement can be found in Wu & Rao (eds) (2005). The commonly used testing method as recommended by ITU-R is Double Stimulus Continuous Quality Scale (DSCQS). The block diagram of DSCQS is shown in Figure 2.5.
The user is made to observe a pair of video sequences A and B and asked to rate the quality of the video sequence in the 5 point scale. Here in each pair of video sequence, one is the original sequence and another is the encoded sequence. The order of the test sequence is randomized so that the observer does not know which one is original and which one is encoded sequence. This method is costly and time consuming and the results are not accurate. Mostly subjective quality measurements are not preferred for judging the video quality of compressed video sequences.

2.6.2 Objective Quality Measurement

The widely used objective quality measurement tool in video compression is peak signal to noise ratio (PSNR). Here the signal is considered as original data and the error introduced by the coder is considered as noise. It is computed by finding the mean squared error between the original and the encoded sequence. The PSNR is given by

$$PSNR = 10 \log_{10} \left[ \frac{255^2}{\text{Mean Squared error}} \right] \text{db.}$$  \hspace{1cm} (2.11)

Mean Squared error = \[
\frac{1}{N_r N_c N_d} \sum_{r=0}^{N_r} \sum_{c=0}^{N_c} \sum_{d=0}^{N_d} [f(r,c,d) - \tilde{f}(r,c,d)]^2
\]  \hspace{1cm} (2.12)
where \( f(r, c, d) \) represent the original frame and \( \tilde{f}(r, c, d) \) represent the reconstructed frame and \( N_r \times N_c \) represent the frame size, \( N_d \) represent the number of frames in the sequence. Higher PSNR value indicates better reconstruction of video sequence.

### 2.6.2.1 Structural SIMilarity index (SSIM)

The structural Similarity index is a measure of perceptual image quality between original and reconstructed image. It is stated in Zhou Wang et al (2004) that compared to PSNR, SSIM gives much better indication of image quality. Also it can be used to measure similarity between two video sequences. In a video sequence, the dependency between the pixels carries information regarding the structure of the object. So by finding mean, variance and covariance of original and reconstructed video sequences SSIM can be calculated it is given in Equation (2.13). The value of SSIM ranges from 0 to 1. Higher the similarity, maximum is the value.

\[
SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)}
\]  

(2.13)

where \( \mu_x, \mu_y, \sigma_x^2, \sigma_y^2 \) are the mean and variance of the original and reconstructed sequence it is given in Equation (2.14) to (2.17). The covariance \( (\sigma_{xy}) \) of the sequence \( x \) and \( y \) is given in Equation (2.18)

\[
\mu_x = \bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i
\]  

(2.14)

\[
\mu_y = \bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i
\]  

(2.15)

\[
\sigma_x^2 = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \bar{x})^2
\]  

(2.16)
\[
\sigma_y^2 = \frac{1}{N-1} \sum_{i=1}^{N} (y_i - \bar{y})^2
\]  
\[\text{and } \sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y}) \]  
\[(2.17) \quad (2.18)\]

and \(C_1\) and \(C_2\) are arbitrary constants given by

\[C_1 = (K_1 L)^2, \quad C_2 = (K_2 L)^2\]

where \(L = 255\) represents the dynamic range of the signals, \(N\) represents the window dimension and \(K_1 = 0.01, \ K_2 = 0.03\). For measuring the SSIM usually [8×8] size is chosen. It is only a measure of similarity so any size can be chosen.

### 2.7 VIDEO CODER BASED ON MOTION ESTIMATION AND COMPENSATION TECHNIQUE

The aim of all video coder is to compress the video sequence. This is achieved through exploiting the spatial and temporal redundancy. The frame rate of video sequences ranges between 25 and 60 frames per second. At this rate, there is no much change between the consecutive frames unless any scene change occurs in the video sequence. It means that all video sequences have high temporal correlation. For reducing the temporal and spatial correlation the video coders use block based motion estimation and compensation algorithm.

In block based video coding, the basic processing unit is macroblock, so the video sequences are divided in to integral blocks of dimension \(N_r \times N_c\) in the raster scan order, where \(N_r, N_c\) varies between 4 to 16. The Figure 2.6 shows the block diagram of video coder based on motion estimation and compensation technique.
The encoder consists of two paths, forward path indicated by bold lines and the reverse path indicated by dotted lines. The spatial and temporal correlation is exploited by intra prediction and inter prediction block. Further the blocks are processed through transform, quantization and finally entropy encoded. The significance of each block is described in detail below.

### 2.7.1 Intra Prediction

Block based video encoders use prediction as a tool for reducing redundant information. Intra prediction is used to remove the spatial correlation that is present in a video frame. For that, motion prediction is performed on the current frame by considering the previously encoded block as a reference frame. If it reasonably matches with the block in the current frame then the predicted block is subtracted from the current block, which results in residual block. It is the residual block that is encoded along with the motion vector (MV) and transmitted. The motion vector represents the displacement of the reference block with reference to the current block. It is shown in Figure 2.6. The decoder receives the coded residual block and motion vector to reconstruct the original block.
2.7.2 Inter Prediction

Inter prediction block is used to remove the temporal correlation. Here the motion estimation is performed between the reference frame and the current frame. The previously encoded frame is considered as a reference frame. Motion estimation in macroblock is performed by finding a macroblock in the reference frame that reasonably matches with the current frame. For example, if the current macroblock say ‘A’ forms a part of a moving object in the video scene, a good match can be found if the same part of the moving object is found from the reference image. The Figure 2.7 illustrates the motion estimation process between the reference frame and current frame.

![Motion estimation process between the reference frame and current frame](image)

Figure 2.7 Motion estimation process between the reference frame and current frame

Usually it is very hard to find an exact match. However, a reasonably accurate match can be found, if the search is performed in restricted region of the image. In exhaustive search, it takes more time for the algorithm to converge, so search region is restricted. Many optimal search
algorithms are developed in Sangjoong et al (1997) & Tsung-Han & Yu-Nan (2006) to enhance the motion estimation. Few of them are

1. Three step search algorithm
2. Two dimensional logarithmic search
3. Binary search
4. Four step search
5. Orthogonal search
6. Cross search algorithm
7. Spiral search
8. Hierarchical search block matching algorithm
9. Diamond search
10. Hexagonal search

2.7.3 Transform Coding

The residual block obtained from the motion estimation and compensation is then processed through transform coding block for further data reduction. It is achieved by reducing the spatial correlation in the residual block. In practice, Discrete cosine transform (DCT) is the widely used transform in video compression technique to reduce the spatial correlation.

2.7.4 Quantization and Entropy Coding

Usually, the lossy compression is achieved through the quantization. The quantization table is constructed based on the statistical properties of the DCT coefficients. The entries in the quantization table are usually fixed and few rate control mechanisms are there to vary the step size of the Q-table to get variable data rate. After quantization, majority of DCT coefficients become zero. To reduce the data rate further, reordering of
coefficients is done in zigzag manner so that more run of zeros are appended at last. There is no need to encode each zeros separately; instead all the run of zeros are encoded with minimum bits using variable length coding. According to the run/level pairs, bits are generated by referring the Huffman table.

2.8 VIDEO CODING STANDARDS

Due to the greater variation in constructing the codec, it is required for manufacturers to follow some standards so that interoperability between the various electronic equipments is easy for the consumer. The two international standardization bodies are


The compression standards developed under ITU-T is represented as H.26x series and the standards developed under ISO/IEC are represented as MPEG-X. The feature and applications of some of the standards developed by the two international bodies are discussed below.

2.8.1 MPEG-1

In MPEG-1, the video sequence is first divided into a group of pictures (GOP), which is often called GOP. Example of GOP is shown in Figure 2.8. Each GOP consists of three types of frames:
1. **Intracoded Frame (I-frame)**

   I-frame is entirely coded as one frame by intraframe technique such as DCT. This type of frame has no need for previous information. It belongs to intra-frame coding.

2. **Predictive Frame (P-frame)**

   Each macroblock in P-frame is assigned a best matching macroblock from the previously coded I- or P- frame, it is called forward prediction.

3. **Bidirectional predictive frame (B-frame)**

   In addition to forward prediction, a backward prediction is also performed, in which the best matching macroblock is obtained from a future I- or P-frame in the video sequence. The reference frame can be either I-frame or P-frame.

![Figure 2.8 Representation of GOP in MPEG](image)
MPEG-1 Standard is developed for multimedia application. The primary interest behind development of this standard was storage of audio and video on standard CD-ROM that support a data rate of 1.5 Mb/s. However the compression is lossy, and the frames reconstructed from the compressed bit stream is not identical to the original frame. The features of other MPEG compression standards are listed in the Table 2.2.

Table 2.2 Features of different MPEG standards

<table>
<thead>
<tr>
<th>Features of the standard</th>
<th>Motion Picture Experts Group standards</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MPEG-1</td>
</tr>
<tr>
<td>Macroblock size</td>
<td>16 X 16</td>
</tr>
<tr>
<td>Block size</td>
<td>8x8</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Transform</td>
<td>DCT</td>
</tr>
<tr>
<td>Transform size</td>
<td>8x8</td>
</tr>
<tr>
<td>Quantization step size</td>
<td>Increases with constant increment</td>
</tr>
<tr>
<td>Entropy coding</td>
<td>VLC</td>
</tr>
<tr>
<td>Motion estimation and compensation</td>
<td>Yes</td>
</tr>
<tr>
<td>Play back and random access</td>
<td>Yes</td>
</tr>
<tr>
<td>Transmission rate</td>
<td>1.5 Mbps up to 2Mbps</td>
</tr>
<tr>
<td>Pel accuracy</td>
<td>Integer ½ pel accuracy</td>
</tr>
<tr>
<td>Frame type</td>
<td>I, P, B</td>
</tr>
<tr>
<td>Reference frame</td>
<td>One frame</td>
</tr>
</tbody>
</table>
2.8.2  H.261

This standard is widely used in video conferencing and video phone application. It was approved in the year 1993 and it was designed to transmit the data over Integrated Services Digital Network (ISDN) lines in the multiples of 64Kbps. The coding rate of H.261 varies between 40 Kbps to 2Mbps. It supports CIF and QCIF format.

2.8.3  H.263

This standard was developed in the year 1995 and it is considered as enhancement over H.261 format. It is mainly designed to support low bit rate video conferencing applications. H.263 has also found applications in the internet in displaying the flash video content for certain period of time. It supports SQCIF, QCIF, CIF, 4CIF, 16CIF format.

2.8.4  H.264/ MPEG-4 part 10

The H.264/AVC or MPEG-4 part 10 was jointly developed by the ITU-T VCEG and the ISO/IEC MPEG. The H.264/AVC is capable of achieving improved compression efficiency and flexibility compared with all previous compression standards, Javier Del Ser (eds) (2011) for additional information. Some of the features are listed in Table 2.3.
Table 2.3 Features of H.264/MPEG-4 part 10 standard

<table>
<thead>
<tr>
<th>Features of the standard</th>
<th>H.264/MPEG-4 Part 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macroblock size</td>
<td>16 X 16</td>
</tr>
<tr>
<td>Block size</td>
<td>8x8, 16x16, 4x8, 8x16, 16x8</td>
</tr>
<tr>
<td>Transform</td>
<td>Integer 4x4 transform</td>
</tr>
<tr>
<td>Transform size</td>
<td>4x4</td>
</tr>
<tr>
<td>Quantization step size</td>
<td>Step size increases at the rate of 12.5%</td>
</tr>
<tr>
<td>Entropy coding</td>
<td>VLC, CAVLC, CABAC.</td>
</tr>
<tr>
<td>Motion estimation and compensation</td>
<td>Yes, 16 motion vectors per macroblock.</td>
</tr>
<tr>
<td>Play back and random access</td>
<td>Yes</td>
</tr>
<tr>
<td>Transmission rate</td>
<td>64 Mbps, 150Mbps</td>
</tr>
<tr>
<td>Pel accuracy</td>
<td>Integer ½ pel, ¼ pel accuracy</td>
</tr>
<tr>
<td>Frame type</td>
<td>I, B, P, Switching Intra coded frame (SI),</td>
</tr>
<tr>
<td></td>
<td>Switching Predictive frame (SP).</td>
</tr>
<tr>
<td>Reference frames</td>
<td>Multiple reference frames are allowed (up to a maximum of 5)</td>
</tr>
</tbody>
</table>

2.9 THREE DIMENSIONAL DISCRETE COSINE TRANSFORM (3D-DCT)

All the existing standard video coders rely on reducing the spatial and temporal redundancy by motion compensation and prediction. However, these algorithms have complex motion estimation and prediction algorithms and no symmetry exists between encoding and decoding block. This made implementing this algorithm in hardware more complex. It was explained in Badawy & Bayoumi (2002) & Junhao et al (2008). A video sequence often can be viewed as three dimensional signals, because pixels in full motion video are also correlated in temporal domain. By applying 2D-DCT,
concentration of energy in spatial domain can be achieved and many fast algorithms are available to implement 2D-DCT. Computing one more 1D-DCT along the temporal domain will give the similar concentration of energy in the temporal domain that leads to 3D-DCT. It is proposed in Raymond & Borko (1997) and it is considered as an alternate for traditional video compression techniques. The block diagram of 3D-DCT based video coder is shown in the Figure 2.9.

![Figure 2.9 Block diagram of 3D-DCT video coder](image)

The video frames are arranged as group of frame (8 frames=1 group). Each group of frames is then partitioned into macroblock of size \([N_r \times N_c \times N_d]\) in the raster scan order, where \(N_r\), \(N_c\), \(N_d\) represent the number row, column and depth of the cube. Here the basic element for processing is macroblock cube. Each macro block was processed through all the blocks of the encoder, starting from computing 3D-DCT, quantization, zigzag scanning and variable length coding to get the compressed video. The same process is reversed to get the original sequence.