

Real Time Video Compression, Transmission and Detection through Digital Communication System

Sowmyashree C
M.tech Student, DCN branch
Dr.AIT
Bangalore

B. Sivakumar
Prof & HOD, Dept. of TCE
Dr.AIT
Bangalore

Abstract— This paper presents a simulation based Video Compression, transmission and detection of video for Communication applications. Image and Video Compression addresses the problem of reducing the amount of data required to represent the information that is to be transmitted, which in turn saves Bandwidth required for transmission of data and as well as the memory required to store it. Hence Video is compressed (based on human eye perception property) using Compression Algorithms which reduces the volume of the Video data with some small trade off of the quality of Video. For the efficient transmission of a compressed Video across a channel, source coding at the transmitter side is done using Huffman coding & then Channel coding is done using Hamming Codes. This data is BPSK modulated and sent through channel. The received data is demodulated, channel decoded and source decoded etc., using inverse of the techniques used in the transmitter to get back the transmitted Video (Compressed Video). And the above flow is also carried on Real Video input taken by Web Camera and the Compressed Video is transmitted and detected through Digital Communication System(DCS) Simulated in MATLAB.

Index Terms— MPEG; Hamming; BPSK; DCS.

I. INTRODUCTION

Study has shown that the 90% of total volume of data in internet access consists of image and video related data. Image and video in their raw (uncompressed) form requires huge storage space. Such raw data needs large transmission bandwidth for the transmission over the network. Hence, lots of researches have been conducted in the field of data compression system. However, in this modern internet age, the demand for data transmission and the data storage are increasing. In this concern, data compression and reconstruction is the only option to relieve the network congestion. The compression technique reduces the size of data, which in turn

requires less bandwidth and less transmission time and related cost.

Video compression algorithms manipulate video signals to dramatically reduce the storage and bandwidth required

while retaining perceived video quality with small trade off in quality.

II. VIDEO COMPRESSION

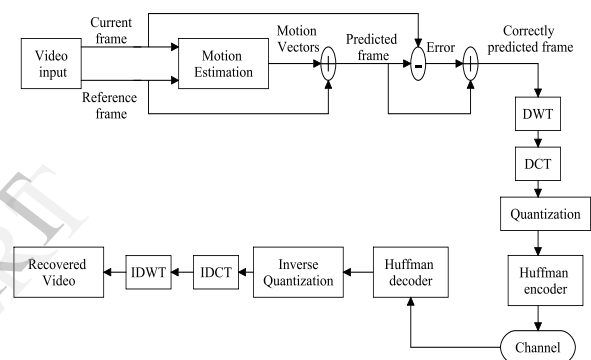


Fig. 1. Implemented Video Compression Block Diagram

Video Compression consists of the following four main stages:

- Dividing each video frame into blocks of pixels so that processing of the video frame can be conducted at the block level.
- Exploiting the spatial redundancies that exist within the video frame by coding some of the original blocks through spatial prediction, transform, quantization and entropy coding (or variable-length coding).
- Exploiting the temporal dependencies that exist between blocks in successive frames. This is accomplished by using motion estimation and compensation. For any given block, a search is performed in the previously coded one or more frames to determine the motion vectors that are then used by the encoder and the decoder to predict the subject block.
- Exploiting any remaining spatial redundancies that exist within the video, again through transform, quantization and entropy coding.

A. Full Search Algorithm

Steps to get the motion estimation using full search algorithm is to divide each frame in a video into macro blocks. Each MB in reference frame compared with all blocks in P or B frame. If match founds take difference along horizontal & vertical. Calculate SAD (Sum of

Absolute Difference). Calculate MAD (minimum absolute difference) based on SAD generates motion vector [10, 11].

III. PROPOSED DESIGN MODEL

The algorithms which are used in the design of Digital Communication System are as follows:

A. Discrete Wavelet Transform

Compression using Discrete Wavelet Transform

Compression is one of the most important applications of wavelets. Like de-noising, the compression procedure contains three steps:

1. Decomposition: Choose a wavelet, choose a level N. Compute the wavelet decomposition of the signal at level N.
2. Threshold detail coefficients: For each level from 1 to N, a threshold is selected and hard thresholding is applied to the detail coefficients.
3. Reconstruct: Compute wavelet reconstruction using the original approximation coefficients of level N and the modified detail coefficients of levels from 1 to N.

Global Thresholding: The compression features of a given wavelet basis are primarily linked to the relative scarceness of the wavelet domain representation for the signal. The notion behind compression is based on the concept that the regular signal component can be accurately approximated using the following elements: a small number of approximation coefficients (at a suitably chosen level) and some of the detail coefficients.

B. Discrete Cosine Transform and Quantization

A DCT converts Pixel value of the image to its corresponding Frequency values. In practice, the human eye is less sensitive to higher frequency component associated with the image than the lower frequency components. Moreover if amplitude of higher frequency component falls below certain threshold amplitude they will not be detected by the eye. By means of DCT and Quantization these high frequency components, which eye cannot detect can be readily identified and eliminated thereby reducing volume of data to be transmitted without degrading perceived quality of original image. DCT represents the input data points in the form of a sum of cosine functions that are oscillating at different frequencies and magnitudes. Since an image is represented as a two dimensional matrix, for this work, 2-D DCT is considered. The 2-D DCT for an N*N input sequence can be defined as follows:

$$D_{DCT}(i,j) = \frac{1}{\sqrt{2N}} B(i)B(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} M(x,y) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right) \rightarrow (1)$$

Where,

$$B(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } u = 0 \\ 1 & \text{if } u > 0 \end{cases}$$

$M(x, y)$ is the input data of size $x*y$. In the DCT compression, almost all of the information is concentrated in a small number of the low frequency coefficients. These low frequency coefficients are also known as DC components and the rest of the components are AC components. The lower right values represent higher

frequencies, & are often small (usually small enough to be neglected with little visible distortion) min the image reduction process, the DCT is applied to 8X8 pixel blocks of image. Hence if the image is 256X256 pixels in size, we break it into 32X32 square blocks of 8X8 pixels & treat each one independently. The 64 pixel values in each block are transformed by the DCT into a new set of 64 values. These new 64 values, known also as the DCT coefficients represent the spatial frequency of the image sub-block. The DCT coefficients are quantized and quantized value is rounded. At the receiver image is dequantised, and then IDCT is performed to recover the transmitted image. Dequantization is done as inverse of quantization.

IDCT is done using formula [1][2]:

$$D_{IDCT}(i,j) = \frac{1}{\sqrt{2N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} B(i)B(j) D_{dequant}(i,j) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right) \rightarrow (2)$$

C. Huffman Coding and Decoding Algorithm

Huffman coding is an efficient source coding algorithm for source symbols that are not equally probable [6, 7]. A variable length encoding algorithm was suggested by Huffman in 1952, based on the source symbol probabilities $P(x_i)$, $i=1, 2, \dots, L$. The algorithm is optimal in the sense that the average number of bits required to represent the source symbols is a minimum provided the prefix condition is met [7].

As in any given Huffman code, no codeword is a prefix of any other codeword, decoding is very simpler provided that receiver has a copy of Huffman code table generated at transmitter. Thus any string of Huffman encoded symbols can be decoded by examining the individual symbols of the string in left to right manner [7, 8]. Because we are using an instantaneous uniquely decodable block code, there is no need to insert delimiters between the encoded pixels.

D. Channel Encoding and Decoding

All real life channels are affected by noise. Noise causes discrepancies (errors) between the input & the output data sequences of a digital communication system. In order to achieve reliability we have to resort to the use of channel coding. The basic objective of channel coding is to increase the resistance of the digital communication system to channel noise. This is done by adding redundancies in the transmitted data stream in a controlled manner. In channel coding we map the incoming data sequence to a channel input sequence. This encoding procedure is done by the channel encoder. The encoded sequence is then transmitted over the noisy channel. The channel output sequence at the receiver is inversely mapped to an output data sequence. This is called the decoding procedure, and is carried out by the channel decoder [6, 8].

Channel coding is also referred to as Error Control Coding. It is interesting to note here that the source code reduces redundancy to improve efficiency, whereas the channel coder adds redundancy, in a controlled manner, to improve reliability. The algorithm which is used for channel coding is Hamming Code [7].

Consider a message source that can generate M equally likely messages. Then initially we represent each message by k binary digits with $2^k=M$. These k bits are the information bearing bits. We next add to each k bit message, r redundant bits (parity check bits). Thus each message has been expanded into a codeword of length n bits with $n=k+r$. The total number of possible n bit codeword is 2^n while the total number of possible messages is 2^k . There are therefore $2^n - 2^k$ possible n bit words which do not represent possible messages. Codes formed by taking a block of k information bits & adding r (= n-k) redundant bits to form a codeword are called block codes & designated as (n, k) codes.

The generation of a block code starts with a selection of the number r of parity bits to be added & thereafter with the specification of H matrix known as Parity Check Matrix.

To generate a codeword T [t1 t2 t3.....tn-k] from the uncoded word A [a1 a2 a3.....ak], we form a generator matrix G such that: $GH^T=0$, and the codeword T will be given by $T=AG$. Consider the block coding technique called Hamming code. The number n of bits in codeword, the number k of bits in the uncoded word & the number r of parity bits are related by

$$\begin{matrix} n=2^r-1 \\ k=2^{r-1} \end{matrix} \rightarrow (3)$$

Let R be the received message which may or may not be the transmitted codeword. We find the value of HR^T (R^T is transpose of R). If $HR^T = 0$, then R is the transmitted codeword, but if $HR^T \neq 0$, we know that R is not the possible message & one or more bits are in error. Considering again the block coding technique called Hamming code in which single error can be corrected by using the syndrome vector $S=HR^T$ and comparing S with H^T . The received corrected code word block R is decoded in order to get back received code word is correctly decoded by extracting the 1st k information bit from R.

E. Binary Phase Shift Keying (BPSK) Modulation and Demodulation

In binary phase shift keying (BPSK) the transmitted signal is a sinusoid of fixed amplitude. It has one fixed phase when the data is at one level & when the data is at the other level the phase is different by 180 degree [6, 8]. If the sinusoid is of amplitude A, the transmitted signal is either $A \cdot \cos(wt)$ or $-A \cdot \cos(wt)$.

In BPSK, the data b(t) is a stream of binary digits with voltage levels which as a matter of convenience, we take to be at +1V or -1V. When $b(t)=1V$ we say it is at logic level 1 & when $b(t) = -1V$ we say it is at logic level 0. Hence $V(bpsk)$ can be written, with no loss of generality, as

$$V(bpsk)=b(t) \cdot A \cos(wt+\phi) \rightarrow (4)$$

The received signal has the form $V(bpsk)=b(t) \cdot A \cos(wt+\phi)$. Here ϕ is the phase shift corresponding to the time delay which depends on the length of the path from transmitter to receiver & the phase shift produced by the amplifiers in the

front end of the receiver preceding the demodulator. The original data b(t) is recovered in the demodulator.

The demodulation technique usually employed is called synchronous demodulation & requires that there be available at the demodulator the wave form $\cos(wt + \phi)$. A scheme for generating the carrier at the demodulator & for recovering the baseband signal is shown in Fig.2

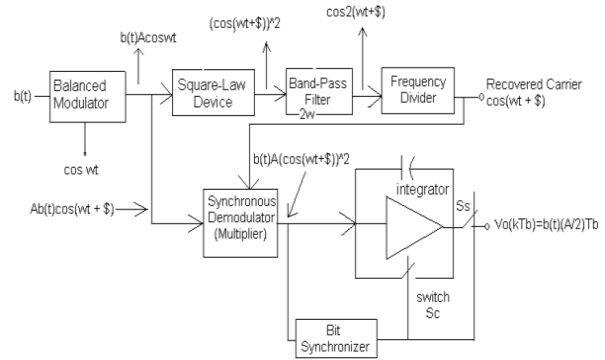


Fig. 2. Scheme to recover the baseband signal in BPSK

F. Channel Equalization

Most popular inverse model, which is also called an Equalizer, is used to reduce the channel distortion. We concentrate on the application of Inverse Modeling in Channel Equalization [2].

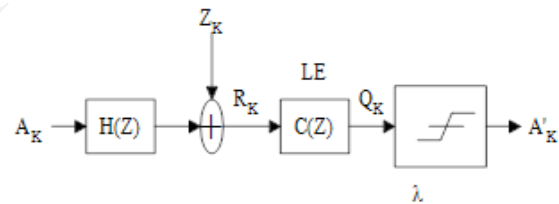


Fig. 3. Calculating the Equalizer Coefficients Iteratively –LMS Approach

The role of the Equalizer, as a filter, is to resolve the distortion caused by the channel, through rejection or minimization of ISI, while minimizing the effect of additive noise at its output. For a channel $H(Z)$, an Equalizer with transfer function $C(Z)=1/H(Z)$ could perform the job, as it produces an overall Channel-Equalizer transfer function $H(Z) C(Z)=1$, which implies that the transmitted data sequence, A_k , will appear at the Equalizer's output without any distortion, assuming no additive noise.

That is the basic idea of linear equalization is to remove the effect of the ISI using the equalization filter $C(z)$. Roughly this means designing an inverse filter to remove the effect of $H(z)$. More specifically, the optimization of the equalizer $C(z)$ depends on (i) the optimization criterion, e.g., Zero-Forcing (ZF) or Mean-Squared-Error (MSE) and (ii) the available computational resources. The idea of ZF is to minimize ISI (without thinking the additive noise at all) whereas the idea of MSE is to minimize the square error between the output of the equalizer Q_k and transmitted symbol A_k . Obviously, the MSE criterion takes the ISI and noise into account, yet the both methods (ZF and MSE) being equal in the high SNR case. In other words, when

SNR increases and if computational resources (equalizer length) are sufficient, $C(z)$ approaches $1/H(z)$. The equalizer output Q_k , as shown in Fig.3, is given by $Q_k = C'R_k$. If this equalizer output is above certain threshold, then decision is made as symbol '1', and if it is below threshold it is decided as symbol '0'.

As we are considering simulation wise, we assume channel to be a linear phase Finite Impulse Response (FIR) filter with defined filter co-efficient, which gives the desired response $H(Z)$ [2]. The noise is to be added for this signal passing through the channel, and it is considered to be a random noise (random numbers generated) Then Channel Equalization is done using LMS technique.

Equalizer coefficients can also be calculated iteratively using the LMS (least-mean-square) approach as

$$c(k+1) = c(k) + \beta * E(k) * R_k \quad \rightarrow (5)$$

Where $E(k)$ is the error signal given by

$$E(k) = A(k) - Q(k) = A(k) - C'R(k) \quad \& \quad \beta \text{ is the step size}$$

IV. DCS DESIGN METHODOLOGY

Fig.4 shows the Block diagram of Digital Communication System. Video is compressed using Compression algorithms such as Fast Full Search and SEA (Successive Elimination Algorithm) for Motion Estimation and to find Motion Vectors (which is based on human eye perception property), and the Compressed output is Source Encoded and transmitted. All real life channels are affected by noise. Noise causes discrepancies (errors) between the input & the output data sequences of a digital communication system. In order to achieve reliability we have to resort to the use of channel coding. The basic objective of channel coding is to increase the resistance of the digital communication system to channel noise. This is done by adding redundancies in the transmitted data stream in a controlled manner. In channel coding we map the incoming data sequence to a channel input sequence. This encoding procedure is done by the channel encoder using Hamming Codes.

The encoded sequence is then modulated by BPSK modulator and transmitted over the noisy channel and Channel Equalization is done.

The received signal is demodulated by demodulation technique. The channel output sequence at the receiver is inversely mapped to an output data sequence. This is called the decoding procedure, and is carried out by the channel decoder, source decoded i.e., inverse process of various processes carried out at transmitter is carried out which results in sufficient data for recovery to be done for efficient detection and recovery of transmitted data (Video).

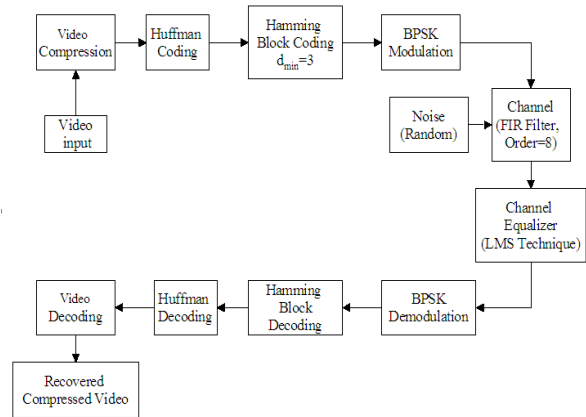


Fig.4: Block diagram of Digital Communication system.

V. RESULTS

Results Observed (in i7 Processor) for 26 frames of video of BIRDS.mp4 and ELEPHANT.mp4 (shown below) are shown below.

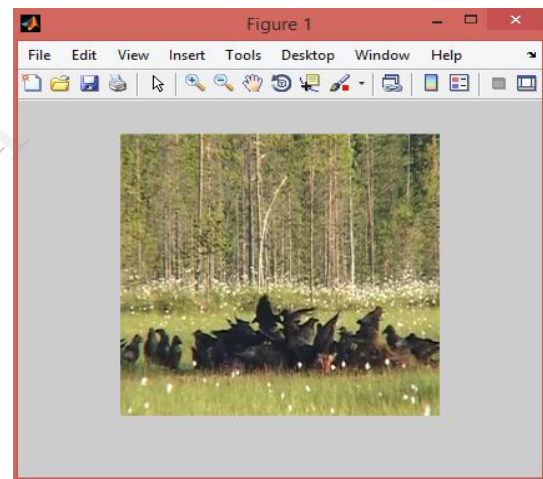


Fig. 5. Input Video

Elapsed time: 85.063524 seconds.

Frame size of Input video: 640X360

Original size of Video: 4995KB

Compressed Video Size: 525KB (256X256)

Mean Square Error (MSE): 12.5151

Peak Signal to Noise Ratio (PSNR): 74.3129

Compression ratio=9.5142

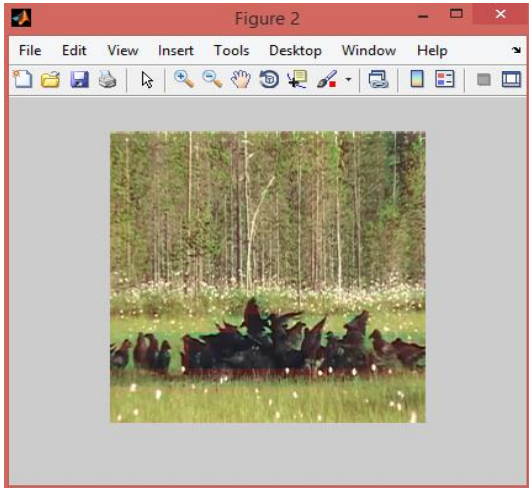


Fig. 6. Compressed Video

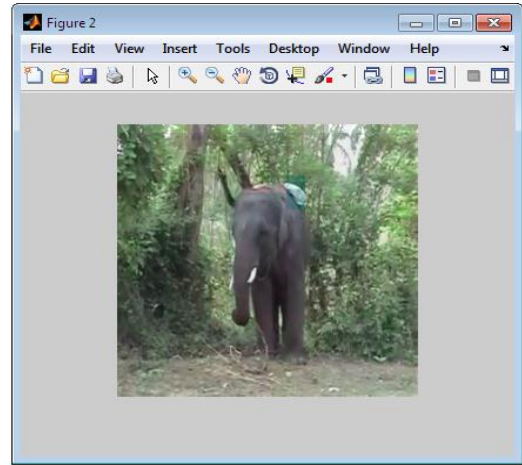


Fig. 8. Compressed Video

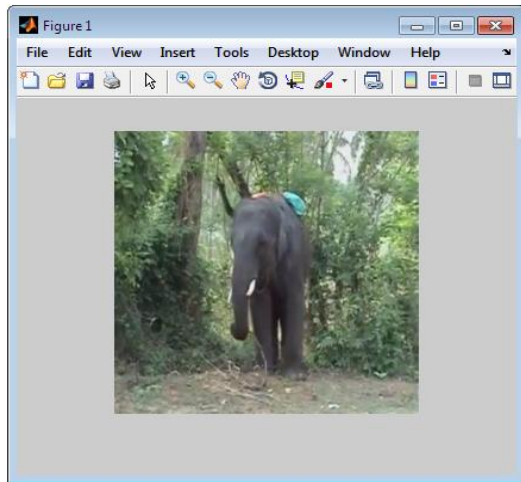


Fig. 7. Input Video

Results Observed (in i3 Processor) for 10 frames of Real time video is shown below.



Fig. 9. Input Video



Fig. 10. Compressed Video

Elapsed time: 90.538522 seconds.
 Frame size of Input video: 384x288
 Original size of Video: 4995KB
 Compressed Video Size: 523KB (256X256)
 Mean Square Error (MSE): 6.6139
 Peak Signal to Noise Ratio (PSNR): 79.8525
 Compression ratio=9.5506

Elapsed time: 118.261858 seconds..
 Frame size of Input video: 320x240
 Original size of Video: 1923KB
 Compressed Video Size: 184KB (256X256)
 Mean Square Error (MSE): 14.6836
 Peak Signal to Noise Ratio (PSNR): 72.9250
 Compression ratio= 10.4510.

VI. CONCLUSION AND DISCUSSION

In this paper, a video is taken and compressed, and the compressed video is transmitted and detected through designed Digital Communication System. The system is simulated in Matlab. MSE, PSNR, Compressed video size and Compression ratio are computed. The System is also run for a real time video captured in Web Camera.

REFERENCES

1. Prabhakar.Telagarapu,V.JagaNaveen, A.Lakshmi..Prasanthi, G.Vijaya Santhi, Image Compression Using DCT and Wavelet Transformations, International Journal of Signal Processing, Image Processing and Pattern Recognition Vol. 4, No. 3, September, 2011.
2. Dr. S. Meher, Dantim Maida, Shrikant Vaishnav, Image Compression & Transmission through Digital Communication System, 2009, National Institute of Technology Rourkela.
3. VirenderPoswal and Dr.Priyanka, Analysis of Image Compression Techniques using DCT, International Journal of Electronics and Computer Science Engineering,Volume1, Number-3,pp.1730-1731.
4. D. Slepian and J.K. Wolf, Noiseless coding of correlated information sources, IEEE Trans. Inform. Theory IT-19 (March 1973), pp. 471–480.
5. Simon Haykin, Digital Communications, John Wiley and Sons, Wiley publications,4th edition,1988.
6. John G. Proakis, Digital Communications, McGraw Hill publications, 4th edition, 2001.
7. Giridhar , Information Theory and Coding, Pooja publications, 1st edition 2001.
8. Fred Halsall, Multimedia Communications, Pearson Publications, 1st edition, 2012.
9. H.264/MPEG-4 Advanced Video Coding, Alexander Hermans, Matriculation Number: 284141RWTH, September 11, 2012.
10. Yih-Chuan Lin and Shen-Chuan Tai, Fast Full-Search Block-Matching Algorithm for Motion-Compensated Video Compression, IEEE Transactions on Communications, Vol. 45, No. 5, May 1997.
11. Yoshihiro Noguchit, Jun Furukawatt and Hitoshi Kiyatt, A Fast Full Search Block Matching Algorithm for MPEG-4 Video.