EFFECTIVE COMPRESSION OF DIGITAL VIDEO

Anil Lohar[1], Komal Khedekar[2], Amruta Kadu[3],
Namrata Raut[4] , Aniruddha Kale[5],Suraj Deshmukh[6]

[1]*Professor, Dept. of Computer Engineering, ABMSP's APCOER Pune, Maharashtra, India*
[2,3,4,5,6]*Student, Dept. of Computer Engineering, ABMSP's APCOER Pune, Maharashtra, India*

Abstract - *In recent few decades digital video compression technologies have become a necessity for the designing, communication and utilization of visual information. Users today have gotten used to taking and posting photos and videos to record daily life, share experiences. According to recent reports, Instagram users have been posting an average of 55 million photos and videos every day. How to store, backup and maintain these amount of photos and videos in an efficient way has become an urgent problem. Video compression techniques plays a vital role in storage and transmission of data through the limited bandwidth capability. This review paper introduces advance technology in video compression technique by providing similar quality and perceiving more content of video. This study proposes a compression method that can obtain a highest compression ratio and high quality for video compression by using Discrete cosine transform (DCT), Motion estimation (ME), Embedded zero wavelet (EZW) algorithms.*

Key Words: **Video compression, Discrete Cosine Transform, Motion estimation, Embedded zerotree wavelet, H.264.**

1. INTRODUCTION

Digital video is a representation of moving visual images in the form of encoded digital data. Digital video comprises a series of digital images displayed in rapid succession.

Video compression is the process of encoding a video file in such a way that it consumes less space than the original file and is easier to transmit over the network/internet. It is a type of compression technique that reduces the size of video file formats by eliminating redundant and non-functional data from the original video file..

There are two basic types of compression techniques:

1) Lossy compression:

Lossy compression is the class of data encoding methods that uses inexact approximations and partial data discarding to represent the content. This technique is used to reduce size for storage, handling and transmitting content.

2) Lossless compression:

Lossless compression reduces a file's size with no less of quality. It is a class of data compression algorithm that allows the original data to be perfectly reconstructed from the compressed data. Lossless compression is used in various applications. For example, it is used in zip file format.

Need of compression:

Now a day every mobile comes with HD cameras. Even if you take 5 min video, your video file size will be around 200 MB or more. When you want to send this file through Facebook or WhatsApp, it will not accept or it will take too much time and bandwidth. It must to reduce video file size.

High bit rates that result from the various types of digital video make their transmission through their intended channels very difficult. Even entertainment video with modest frame rates and dimensions would require bandwidth and storage space far in excess of that available from CD-ROM. Thus delivering consumes quality video on compact disk. Even if high bandwidth technology (example, fiber optic cable) exists, the per byte cost of transmission would have to be very low before it would be feasible to use it for the staggering amount of data require by HDTV. Finally, even if the storage and transportation problems of digital video were overcome, the processing power needed to manage such volumes of data would make the receiver hardware very expensive. Although significant gain in storage, transmission and processor technology have been achieved in recent years, it is primarily the reduction of amount of data that needs to be stored, transmitted and processed that has made wide spread use of digital video a possibility. This reduction of bandwidth is possible by advance in compression technology.

2. VIDEO COMPRESSION USING DCT

Main objective of video coding in video applications is to reduce the amount of video data for storing or transmission purposes without affecting the original quality of the video. Performance of generated output video depends on the applications requirement in terms of quality, bandwidth, and memory space. Encoders based on the Motion Estimation makes use of inter frame correlation to provide efficient compression and its real-time implementation is difficult as well as costly. [9]

The video compression system consist of video encoder at transmitter side, which encodes the video to be transmitted in form of bits and the video decoder at the receiver side which reconstructs the video in its original form from the bit sequence received. Video is combination of number of frames. First frames which are nothing but image sequences. The intra coding method is used for compression of I-frames.

2.1 Spatial redundancy

Elements that are duplicated within a structure such as pixel in a still image and bit pattern in a file. Exploiting spatial redundancy is how compression is performed. Spatial redundancy is redundancy within a single picture or object.

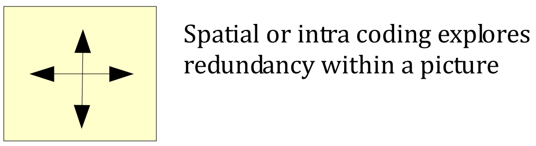


Fig -1: Spatial Redundancy

- Initially create an I-frames.
- Human eye is insensitive to HF color changes, we convert RGB into luminance and two color different signals that is YUV model. Y is the luminance component that is brightness while U and V are the chrominance (color) components. We can remove more U and V components than Y.
- DCT values of every pixel are calculated from all other pixel values, so taking 8*8 blocks reduces the processing time.
- The top left pixel in a block is taken as the dc datum for the block. DCT's to the right of the datum are increasingly higher horizontal spatial frequencies. DCT values which are at lower level have higher vertical spatial frequencies.
- Using inverse DCT we can reconstruct each pixels value in the 8*8 block.

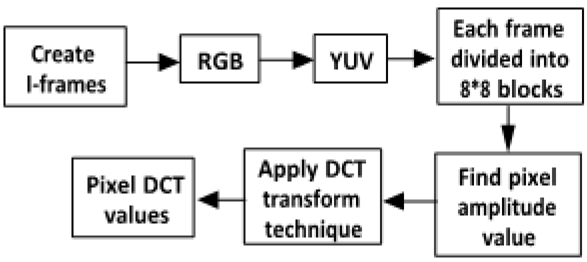


Fig -2: Block Diagram for Spatial Redundancy using DCT

2.2 Temporal redundancy

Pixels in two video frames which have same values in the same location. Exploiting temporal redundancy is one of the spatial redundancy. The temporal redundancy exists between successive pictures. Adjacent frames are related with each other so can predict one from other and getting residue from them which will give temporal compression in video. According to this, there are three types of frames.

1) I-frame: I-frames are the least compressible but don't require other video frames to decode.

2) P-frames: P-frames can use data from previous frames to decompress and are more compressible than I-frames.

3) B-frames: B-frames can use both previous and forward frames for data reference to get the highest amount of data compression.

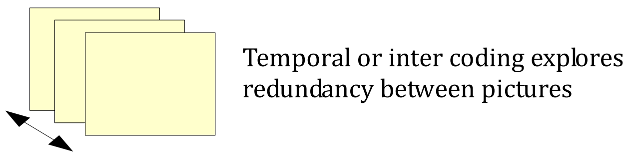


Fig -3: Temporal Redundancy

Inter coded frame is divided into blocks known as macroblocks. After that, instead of directly encoding the raw pixel values for each block, the encoder will try to find a block similar to one it is encoding on a previously encoded frame, referred to as a reference frame. This process is done by a block matching algorithm. If the encoder succeeds on its search, the block could be encoded by a vector known as motion vector, which points to the position of the matching block at the reference frame. The process of determination of motion vector is known as motion estimation.

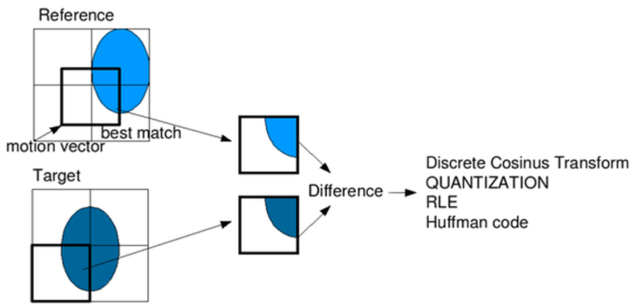


Fig -4: Block matching algorithm

Following are the steps of Inter frame coding:

1) Frame segmentation: The frame is divided into the number of blocks which are equal in size, non overlapping and rectangular.

2) Search threshold: If the difference between target block and candidate block at the same position in past frame then no any motion is take place and it returned as a zero.

3) Block matching: Every blocks in current frame is compare with the every another blocks in past

frame. If block size is m and maximum displacements in horizontal and vertical direction are dx and dy respectively, then (2x + m)(2y + m) will be the search area.

4) Matching criteria: Matching criteria is used to find the similarity between target block and candidate block.

5) Motion vector: It describe the location of matching block from past frame with reference to the position of the target block in the current frame.

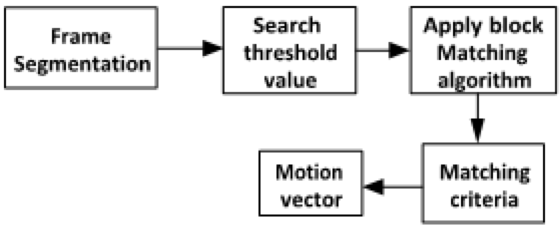


Fig -5: Block diagram of Inter-Frame prediction

2.3 Video Decoder

1. By using run length decoder bit stream is converted into the original coefficients of DCT.

2. These coefficients are quantized using quantization matrix to original DCT coefficients.

3. To get the approximate component of frames inverse DCT is used.

4. Then by using motion estimation algorithm the frames are motion compensated.

5. YUV is back to converted into the RGB .

6. At last the sequence of the frames are converted into the video which is compressed video.[6]

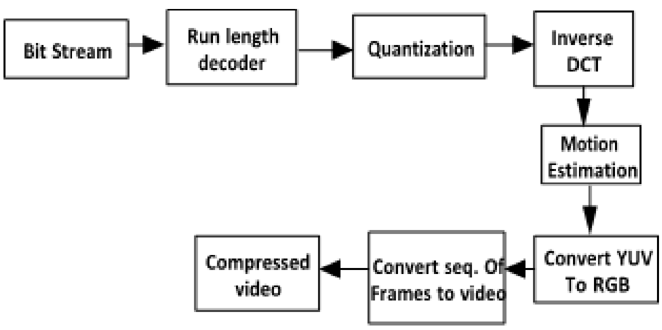


Fig -6: Block diagram of video decoder

Following formulas are used to measure the degradation after compression:

1. Peak signal to noise ratio:

$$PSNR=10*log\ 2552/MSE$$

2. Structural similarity:

$$SIMM(x,y)=\frac{(2\mu_X\mu_y+c1)(2\sigma_{XY}+c2)}{(\mu_X^2+\mu_Y^2+c1)(\sigma_X^2+\sigma_Y^2+c2)}$$

3. Compression ratio:

$$CR=100*\frac{Compressed\_data\_rate}{Uncompressed\_data\_rate}$$

3. MOTION ESTIMATION IN VIDEO COMPRESSION:

Motion estimation based encoders are widely used in video compression techniques, because of its simplicity and coding efficiency. Using model of motion of objects between frames the encoder estimates the motion between two frames that is reference and current frame. This process is called motion estimation. Motion estimation is core and very important component in video processing. Efficiency of video is determined by motion estimation algorithm. Efficiency also depends on coding speed, compression ratio and image quality of decoded video. Motion estimation hard to do as exploration of temporal redundancy between reference and current frame of video sequence allows for significant data compression.

3.1 Motion Detection

Motion detection is nothing but identifying motion between two frames provided there should be motion. Motion detection is sensible to perform motion estimation algorithm only where motion is take place. By using following formulae motion is detected,

In(x)=In-1(x+dn,n-1)+e(x) (1)

In(i,j)=In-1(i+dx.j+dy)+e(i,j) (2)

where,
x=[i,j] and d=[dx,dy]

When d=0 for equation (1) , In(x)=In-1(x)+e(x)
In-1= reference frame;
In= current frame;
d= displacement function;
e= error function;

Pixel difference (PD) can be noisy, as noise is there in the sequence can cause problem which can be overcome by smoothing pixel difference before thresholding. Motion estimation is the estimation of displacement of image structures from one frame to another frame in the time sequence of 2D image. Changes between frames is are to space time objects. In motion estimation, motion compensation technique provides a better prediction of current frame. The encoder uses motion model and information to move the content of reference frame. This is motion compensation and prediction produced for this purpose is called motion compensated prediction. [4]

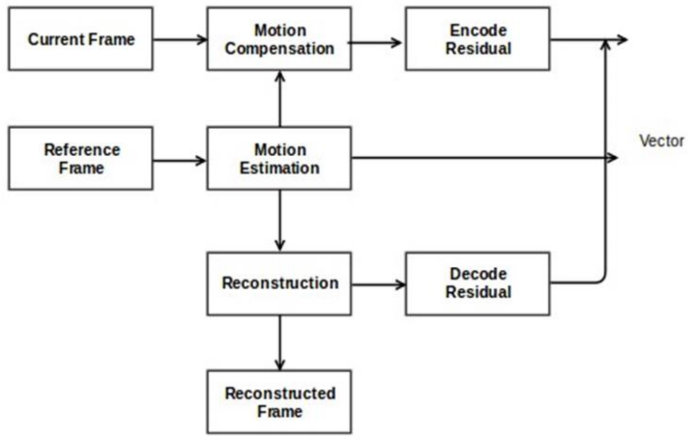


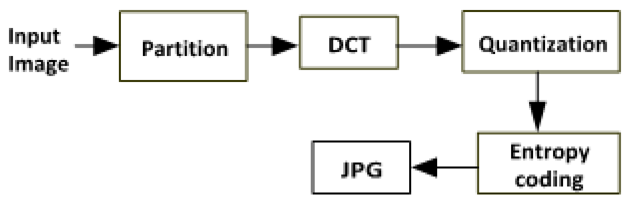
Fig -7: Block diagram of Motion Estimation

4. LOSSLESS COMPRESSION OF JPEG CODED PHOTO COLLECTIONS:

JPEG compression is designed for reducing the size of photos captured in realistic scenarios with smooth variations of tone and color. The JPEG baseline is exclusively used as a common and default format in almost all imaging devices like digital cameras and smart phones. Here the focus is on the lossy coded baseline JPEG image, which is referred to as the JPEG coded image. The JPEG baseline, reduces inter-block redundancy among DC coefficients via a differential coding and exploits the statistical correlation inside each block through a table based Huffman coding. The performance of image compression can be enhanced by introducing both advanced intra prediction methods, such as pixel-wise and block-wise intra predictors, and high efficiency entropy coding methods like arithmetic coders. To further compress JPEG coded images, these methods have to first decode the JPEG coded images to the spatial domain(e.g. YUV or RGB space) and then perform intra or inter predictions.

1. Baseline JPEG compression:

Input image is divided into 8*8 blocks. Each block is then converted into a frequency domain by an 8*8 DCT, followed by the scalar quantization.



2. Lossless Compression of Individual JPEG images:

The most straightforward way to reduce the storage size of a JPEG coded image losslessly is to replace the Huffman coding by an arithmetic coder. In fact, JPEG extension has already supported an adaptive binary arithmetic coder which can reduce the file size by 8-10%.

3. Image set compression:

When dealing with a group of correlated images, several image set compression schemes have been proposed in the literature. They can be roughly divided into two classes.

The first class of approaches generates a representative signal(RS) (e.g. an average image) from all correlated images to extract the common redundancy among them. Then both the RS and the difference signal between each correlated image and the RS are compressed. The centroid method, the max-min differential method, the max-min predictive method and low-frequency template. Accordingly, these approaches work efficiently when images in a set are similar enough, but share the same limitations when dealing with general image sets.

The second class of approaches focuses on pair-wise correlation between images. Hierarchical clustering methods for image set compression. The image set can be clustered by a minimum spanning forest(MSF) and each cluster can be described by a minimum spanning tree(MST) and coded with inter-image prediction.[2]

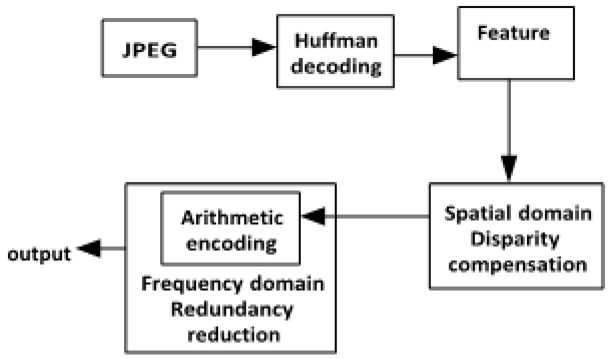


Fig -9: Architecture of encoders

5. RUN LENGTH ENCODER (RLE):

Significant improvement in compression ratio can be done using run length encoding i.e. RLE we can use bit stuffing in RLE. As large sequence of bits affects compression ratio of image so we break it into small sequence by bit stuffing. RLE is very simple form of data compression it does losses compression in this each run is translated by encoding algorithm in pair (u, v), l is length or run y is value of run elements. This encoding scheme also performs expansion of data, this technique is less reliable so it's a poor technique and practically not efficient for big/huge data. In this, if there is large number of sequence that may lead to large run length and if there is sequence or single zeros/one's double zeros/one's, that lead to expansion data, so this are 2 main problem of RLE. But by using Bit stuffing in run length encoding, it limits the number of consecutive bits of same value in data to be transmitted bit of opposite value inserted in the sequence after max. allowed number or consecutive bits. Small sequence result in expansion, so we ignore single zero/ones or double zeros/ones. In this way we leave small sequences out of RLE and another sequence is same as it is, after that intelligent coding is done, and then the processed RLE data is sent to receiver, after that at receiver the steps are performed in reverse order, then receiver does run length decoding (RLD) on that, by reverse order bit stuffing, inside worse case also this algorithm gives fair results. [8]

6. H.264/AVC VIDEO COMPRESSION STANDARDS:

The main aim of H.264/AVC (Advanced Video Coding) standard is to upgrade compression performance and provision of network friendly video representation and addressing video telephony and storage, broadcast and streaming of video applications. H.264 is widely used for Standard Definition (SD) and High Definition (HD) TV signals satellite, cable and the storage of high-quality SD video signals onto DVDs.

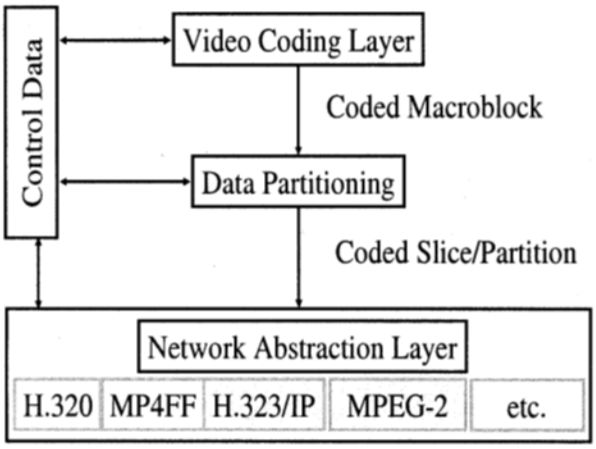


Fig -10: H.264 Video Encoder

To overcome the problem of flexibility and customizability the H.264/AVC covers a VCL (Video coding standard which follows block based hybrid video coding approach in which each coded picture is represented in block shaped units of associated luma and chroma samples all called macroblocks.) which is designed for efficient representation of video content, and a NAL (designed to provide network friendliness.) which formats the VCL representation of the video and provides header information. Following are the features of H.264/AVC design that enable enhanced coding efficiency which includes following enhancements of the ability to predict the values of content of picture to be encoded:

1) Variable block size motion compensation with small block size.
2) Quarter-Sample-Accurate motion compensation.
3) Motion vectors over picture boundaries.
4) Multiple reference picture motion compensation.
5) Decoupling of referencing order from display order.
6) Weighted prediction.
7) Improved skipped and direct motion influence.
8) Directional spatial prediction for intra coding.
9) In-the-Loop de-blocking filtering.
10) Exact match inverse transform.
11) Arithmetic Entropy coding.
12) Hierarchical block transform.
13) Redundant pictures.
14) Data Partitioning.
15) Short word-length transform. [10]

7. HIGH EFFICEINCY VIDEO CODING (HEVC):

HEVC stands for High Efficiency Video Coding. HEVC is standard and developed by ITU-T video coding expert group and ISO/IEC Moving Pictures Group. This provides high compression performance. There is 50% bit rate reduction is seen for same video quality.

The first version came in January 2013. Later other tasks and features are added like extended-range uses, enhanced precision and color format support. ITU also created H.261 and H.263.These two standards are used in various products. Further research and development is still going on to increase compression without compromising quality. H.264 is popular standard used widely. Several improvements were made in H.264/MPEG-2. It is used in HD broadcast, cable, camera, security applications, Internet and Mobile network video, Blu-Ray discs, and real-time conversational applications example video chat, hangout. But in today's world we need beyond HD formats so more mature and effective standards are required. Ultra HD or 4K are very high quality formats. Due to large file size of these file formats more bandwidth and disk space is consumed. HEVC provides us generic syntax so it can be used for anywhere. In HEVC we only structure the bitstream and syntax.

Meaning of syntax element is defined and decoder will produce the same output. But there is no guarantee that it will produce end-to-end reproduction quality.

8. VIDEO COMPRESSION USING EZW:

The EZWT algorithm was introduced for two-dimensional signal. But it can be used for one dimension also by translating key ideas from two dimensions to one dimension. In EZWT algorithm multi-level wavelet decomposition is applied to the original signal. [5]

Embedded Zerotrees of Wavelet transforms (EZW) is a lossy image compression algorithm. It is low-bit rate image coder. EZW produced a fully embedded bit stream. Hence it will provide better compression performance, as the bit stream produces exactly the same images. We know that if more bits are added to the streams during the encoding then in the decoded image will contain more details.

8.1 Features of EZW

1. It is possible to stop the compression algorithm at any time and obtain an approximation of the original image, the greater the number of bits received, the better the image.

2. Due to the way in which the compression algorithm is structured as a series of decisions, the same algorithm can be run at the decoder to reconstruct the coefficients, but the decisions being taken according to the incoming bit streams.

3. In practical implementations, it would be usual to use an entropy code such as arithmetic code to further to improve the performance of the dominant pass. Bits from the subordinate pass are usually random enough that entropy coding provides no further coding gain. The coding performance of EZW has since been exceeded.

8.2 Encoding algorithm:

Step 1:

Structure which contain the sequence of frames is used to open the AVI file. Also used the ASCII long function to convert FCC ASCII to long.

Step 2:

In this step, conversion of RGB color to YUV color space is done by using DCT. Following are used to this conversion,

Y=0.299 R + 0.587 G + 0.114 B
U=0.492 (B-Y)
V=0.877 (R-Y)

Step 3:

Sub-sampling takes the U and V color components and reduced each component to half by using algorithm. These two components are divided in height and width direction, it will reduce the size of file.

Step 4:

By using DCT algorithm each frame is divided into the 8*8 block to transfer the image data from spatial domain to frequency domain. It produces 64 coefficients of each block using DCT. Following formulae are used to find out the coefficient:

Quantization process is used to reducing the number of possible values of block, it reduces the data which required to represent the image.

Step 5:

EZW algorithm consist of four steps i.e. threshold, dominant, subordinate and iterations.

5.1. Initial threshold T0:
In this step the threshold value is evaluate for each block. Threshold value is used to determine the coefficients whether it is significant (greater than threshold value) or insignificant (lesser than threshold value). Threshold value is calculated by,

T0=2|log2(max_coeff)|

5.2. Test function:

Test function is used to check if there is at least one absolute value which is greater than minimum absolute value and it should be lesser than current threshold value.

5.3 Dominant pass:

The dominant pass encodes the significance of the coefficients which have not yet been found significant in earlier iterations (number of iterations by dominant pass), by scanning the trees and emitting one of the four symbols.
Encoding a coefficient of the significance map:

1. Zerotree root: If the magnitude of a coefficient is less than a threshold T, and all its descendants are less than T, then this coefficient is called zerotree root. And if a coefficient has been labeled as zerotree root, it means that all of its descendants are in significance, so there is no need to label its descendants.

2. Isolated Zero: If the magnitude of coefficients that is lesser than threshold T, but it still has some significant descendants then this coefficient isolated zero.

3. Positive Significant Coefficient: If magnitude of coefficient is greater than threshold T and positive ,then it is positive significant Coefficient.

4. Negative Significant Coefficient: If magnitude of coefficientis greater than threshold T and negative ,then it is negative significant Coefficient.

5.4: Subordinate pass:

Subordinate pass algorithm is used to check the value of coefficient is greater or smaller than current threshold. The current threshold is calculated by

Ti=1/2*Tc

Where,
Tc=It refers current threshold which computed to assign Zero or One to the coefficient.
Ti=It represent the threshold come from dominant pass for current phase.

5.5: Filter:

The main task of this procedure is to search for the values that are coded as significance and replace with zeros.

Step 6:

Run Length Encoding(RLE):

This algorithm is used to reduce data by replacing long sequence of same symbol by shorter order.

8.3 Decoding algorithm:

Step 1:

The compressed file is read byte after byte by using inverse Run Length coding algorithm. First byte is assigned to the counter and if it is less than 64 then loop is continued to read the data.

1.1 If next character is '#' then assign current threshold value to T variable, assign DC to F variable and initial value to flag 1,flag 2.

1.2 If Cod(I+1) character is '@' then call the function which make the new block by using DC for first element at the (0,0) and other positions filled with threshold. To reach the '#' character increase the counter by 2. Change the value of flag 1 to 1.

1.3 Else increment counters I,K.

1.4 Go to the Inverse EZW algorithm and send the block result to final array.

1.5 Else read the character.

1.6 Increment counter of character.

1.7 If the coming character is '#' then change the value of flag 1 to 1 , else go back to the step 1.6

Step 2:

Inverse embedded zerotree algorithm:

2.1 Initially CKL variable is to zero.

2.2 Read the string array which contain p,n,a,t and also read array in which counts are stored.

2.3 If the number of cycles is become less than final cycle that is number of cycles taken from the number of rows of coming array. Else control goes to final value of block and reconstructed block is getting stop.

2.4 If coming array contain 'y' character then it is consider as a special case, where there should be at least one absolute value which is greater than zero. But this value is less than threshold value. Then in this case CKL counter is increase and halving the threshold value.

2.5 Dominant pass procedure is used for assigning one of four symbols from p,n,z,t to each position of default block.

2.6 Subordinate pass, it uses default array coming from dominant pass procedure, array which contains counter value and threshold coming from the 2.2 step. This procedure is used to reconstruct the original block.

Step 3:

Inverse of quantization take the 2D coefficients of 8*8 block and multiply by factor R to reach up to the coefficient from source value. Then inverse DCT to produce RGB image matrices.

Step 4:

Inverse of sub-sampling take only U and V color component i.e. red and blue color and multiply them by two in horizontal and vertical direction.

Step 6:

Reconstruct avi file.[1]

9. PERFORMANCY ANALYSIS PARAMETERS:

To compress digital video various combinations are made and the output of the compressed video is compared to see which output is better. This gives the basic idea about its performance and makes us easy to select the best out of various combinations.

9.1 Video Quality Measure:

Video quality is a metric that is used to measure the degradation of video as compared to the original video. Video quality can be evaluated either using mathematical models (objective) or asking users for their rating (subjective). In subjective method, a group of viewers is chosen, video sequences are shown to them for their feedback to rate the quality of video. In objective method mathematical models are created to approximate results of subjective method.

- Full Reference Methods (FR): FR method is the most accurate method but it requires higher computational effort. In this method every pixel received is compared against the corresponding pixel of the original video, and no knowledge about the encoding or transmission process in needed.

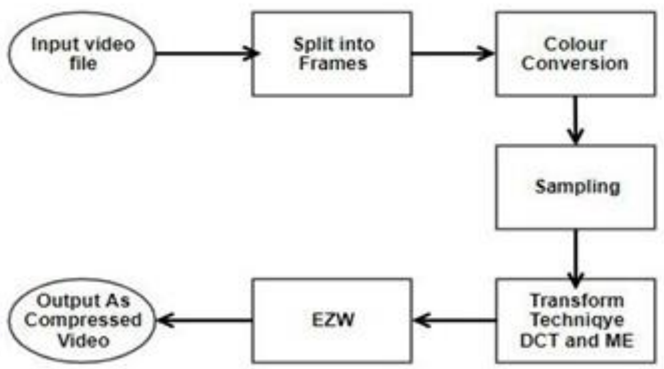
- Reduced Reference Methods (RR): In RR method the entire source video is not needed. Some features of received and source videos are extracted and compared to obtain a quality score. This method can be used when some part of the original video is only available.

- No-Reference Methods (NR): As the name indicates, NR method tries to access the quality of a distorted video without considering the original video for comparison. It is less accurate than FR or RR methods due to the unavailability of an original signal.

9.2 Peak signal-to-noise ratio (PSNR):

Peak signal-to-noise ratio (PSNR) is ratio of the maximum power of a signal to the maximum power of noise. To measure the quality of video reconstructed from Lossy compression codec's PSNR is most commonly used. Lower the value of PSNR, poor will be the quality of the reconstructed video.

It is the most widely used objective image quality metric. The average PSNR over all frames can be considered a video quality metric. With the increasing demand for digital video, other metrics which are more precise were also developed. But these metrics require more computationally efforts and are also potentially more complex than PSNR.[3]



10. PROPOSED SYSTEM:

The proposed method consist of basically two parts one is uploading a video file and another is compression of the video file. When user uploads the video, at the backend video gets stored into the database and then it gets divided into number of frames. In first phase the transform DCT algorithm is used to transform image data to frequency domain from the spatial domain by dividing each block of frame. Quantization process is used to reduce the number of blocks. In the compression phase of the video we will be using Embedded Zero-Tree Wavelet algorithm which is effective data compression algorithm.

Proposed System consists of the following algorithms:

1) DCT (Discrete cosine transform).

2) ME (Motion Estimation)

3) EZW (Embedded Zerotree Wavelet).

11. CONCLUSION:

In this project we proposed a system which compresses a video file by providing similar quality of video. This system highlights DCT (Discrete Cosine Transformation), ME (Motion Estimation) and EZW (Embedded Zero-Tree Wavelet) algorithms. The proposed method first splits a video file into frames. Then transformation is done from RGB to YUV luminance and chrominance. After that Discrete Cosine Transform technique and EZW Embedded Zero-T ree Wavelet algorithms are applied. After processing we get compressed video file. The proposed system provides good quality or similar quality of video after compression.

REFERENCES

[1] Samaher Al-Janabi, and Ibrahim Al-Shourbaji, "A Smart and Effective Method for Digital Video Compression", 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2016.

[2] Hao Wu, Xiaoyan Sun, Senior Member, IEEE, Jingyu Yang, Wenjun Zeng, Fellow, IEEE and Feng Wu, Fellow, "Lossless Compression of JPEG Coded Photo Collections", DOI 10.1109/TIP.2016.2551366, IEEE Transactions on Image Processing, 2016.

[3] Renuka G. Deshpande, and Lata L. Ragha, "Performance Analysis of Various Video Compression Standards", International Conference on Global Trends in Signal Processing, Information Computing and Communication, 2016.

[4] Srinivas Bachu, and Dr. K. Manjunatha Chari, "A Review on Motion Estimation in Video Compression", Department of ECE, K L UNIVERSITY, SPACES-2015.

[5] Jesmin Khan, Sharif Bhuiyan, and Gregory Murphy, and Morgan Arline, "Embedded Zerotree Wavelet Based Data Denoising and Compression for Smart Grid", DOI 10.1109/TIA.2015.2420613, IEEE, 2015.

[6] Ms. S. S. Wadd, and Prof. Mrs. S. B. Patil, "Video Compression using DCT", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 9, ISSN: 2277 128X, September 2014.

[7] Gary J. Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand, "Overview of the High Efficiency Video

Coding (HEVC) Standard", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 22, NO. 12, DECEMBER 2012.

[8] Asjad Amin, Haseeb Ahmad Qureshi, Muhammad Junaid, Muhammad Yasir Habib, Waqas Anjum, "Modified Run Length Encoding Scheme with Introduction of Bit Stuffing for efficient Data Compression", 6th International Conference on Internet Technology and Secured Transactions, 11-14 December 2011

[9] Tarek Ouni, Walid Ayedi, and Mohamed Abid, "New low complexity DCT based video compression method", 978-1-4244-2937-0/09/$25.00 © IEEE, 2009.

[10] Thomas Wiegand, Gary J. Sullivan, Gisle Bjøntegaard, and Ajay Luthra, "Overview of the H.264/AVC Video Coding Standard", IEEE.