COMPRESSION TECHNIQUE
FOR
MULTIMEDIA

By

Prajapati Rashmin B.
[05MCE012]

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY
NIRMA UNIVERSITY OF SCIENCE AND TECHNOLOGY
Ahmedabad 382481

April 2007

COMPRESSION TECHNIQUE
FOR
MULTIMEDIA

Major Project

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology in Computer Science and Engineering

By

Prajapati Rashmin B.
(05MCE012)

Guide
Dr. (Prof.) S.N. Pradhan

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY
NIRMA UNIVERSITY OF SCIENCE AND TECHNOLOGY
Ahmedabad 382481
April 2007

This is to certify that Dissertation entitled

Compression Technique for Multimedia

Presented by

Prajapati Rashmin B.

has been accepted toward fulfillment of the requirement

for the degree of

Master of technology in Computer Science & Engineering

Prof. S. N. Pradhan
Professor In Charge

Prof. D. J. Patel
Head of The Department

Prof. A. B. Patel
Director, Institute of Technology

CERTIFICATE

This is to certify that the Major Project entitled "Compression Technique for Multimedia" submitted by Mr. Prajapati Rashmin B. (05MCE012), towards the partial fulfillment of the requirements for the degree of Master of Technology in Compute Science & Engineering of Nirma University of Science and Technology, Ahmedabad is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Project Guide

Dr. (Prof.) S. N. Pradhan
P. G. Coordinator,
Department of Computer Engineering,
Institute of Technology,
Nirma University,
Ahmedabad

Date :-

ACKNOWLEDGEMENT

It gives me great pleasure in expressing my sincere appreciation and profound gratitude to **Dr. (Prof.) S. N. Pradhan**, P.G.Coordinator, Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad for his valuable guidance and continues encouragement in the completion of this project and development of this report. I am glad to grab the opportunity to work under his guidance.

I would like to express my gratitude to **Prof. D. J. Patel**, Head, Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad and to **Prof. A. B. Patel**, Director, Institute of Technology for providing me with the opportunity to work at Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad and providing all the resources during the course of study.

Further I am thankful to all faculty members of Department of Compter Engineering, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

The blessings of God and my family members makes the way for completion of major project. I am very much grateful to them.

The friends, who always bear and motivate me throughout this course, I am thankful to them.

Prajapati Rashmin B.

[05MCE012]

ABSTRACT

Multimedia system contains different types of data like text, image, audio, video etc. It requires the large storage space and available bandwidth of network does not allowed to transmit the real time multimedia data. For this reason compression of multimedia data is necessary.

The thesis work describes the implementation of 3D-DCT (Three Dimensional Discrete Cosine Transform) video compression technique. It describes the implementation of an Adaptive 3D-DCT compression algorithm. The work required pre-requisite study of available compression standards for multimedia. These standards are based on the 2D-DCT. The algorithm has main three steps, 3D-DCT transform, Quantization and last Entropy encoding. In decompression all the three steps are perform in reverse order. In 3D-DCT compression introduced the quality factor, which will decide the quantization volume vale and quality of the decompressed video. In An adaptive 3D-DCT compression, size of the cube for 3D-DCT transformation and the quantization volume is determined dynamically on the basis of level of motion. The encoder and decoder are written in Matlab tool.

An implementation adaptive 3D-DCT compression technique is works well for low motion video and maintaining the quality of the decompressed video but still not giving the high compression ratio. An implementation of 3D-DCT video compression results in the value of the quality factor should be within the range 1-25, beyond that the quality of the decompressed video is not be acceptable.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS AND ABBREVIATION

1D-DCT	-	One dimensional Discrete Cosine Transform
2D-DCT	–	Two dimensional Discrete Cosine Transform
3D-DCT	–	Three dimensional Discrete Cosine Transform
CR	–	Compression Ratio
MSE	-	Mean Square Error
NRMSE	–	Normalized Root Mean Square Error
PSNR	–	Peak Signal to Noise Ratio
DCT	–	Discrete Cosine Transform
DFT	–	Discrete Fourier Transform
IDCT	–	Inverse Discrete Cosine Transform
MDCT	–	Modified Discrete Cosine Transform
JPEG	–	Joint Photographic Experts Group
MPEG	–	Moving Photographic Experts Group
AVC	-	Advanced Video Coding
VCEG	-	Video Coding Expetrs Group
JVT	–	Joint Video Team
ISO	–	International Organization for Standardization
IEC	–	International Electrotechnical Commission
ITU	–	International Telecommunication Union

1. INTRODUCTION

1.1 MULTIMEDIA SYSTEM

Multimedia system is the combination of the various data. This data can be text, image, animation, audio, videos. Multimedia is the revolution in four areas: communications, computing, entertainment, and consumer electronics. People are demanding such "TV features" as sound, image, and video in their computers, and thus multimedia become popular and growing area in the world.

Multimedia is the technology of 1990s. This system became very popular during this time. And its growth was also very rapid. In 1998 nearly half of business multimedia applications centers on multimedia communications and collaborative applications. So by this growth in 2001 this multimedia system and application was everywhere.

Multimedia content means large amount of data so its stresses all the components of the computer system like memory and storage area because it requires high capacity and also for multimedia, access time also play role, for multimedia access time should be high and also have high transfer rate. In terms of network multimedia require high bandwidth, low latency and jitter to meet the real time application needs. Operating system also stresses by multimedia, because for multimedia operating system should have support of new data types and have real time scheduling and fast interrupting processing. For multimedia system architecture also have high bus bandwidth and efficient I/O. So, multimedia stress all this components of computer.

Multimedia operating system also provide for sharing of multimedia resources by concurrently running applications. Synchronization and streaming functions are also provided by the multimedia operating system. It also supports for record and playback of audio, images, and full-motion video. It also provides Media control interface support: like opening and closing devices, play, record, stop, pause, and seek functions. Multimedia operating system also supports Multimedia I/O manager which allows application independence from data-object format.

Now multimedia workstation has all this operating system support. The components of the multimedia workstation are audio sub systems, video sub systems; codecs include both audio and video codecs, network interfaces, capture devices, etc. Fig. 1.1 [1] shows the basic component of the multimedia workstation.

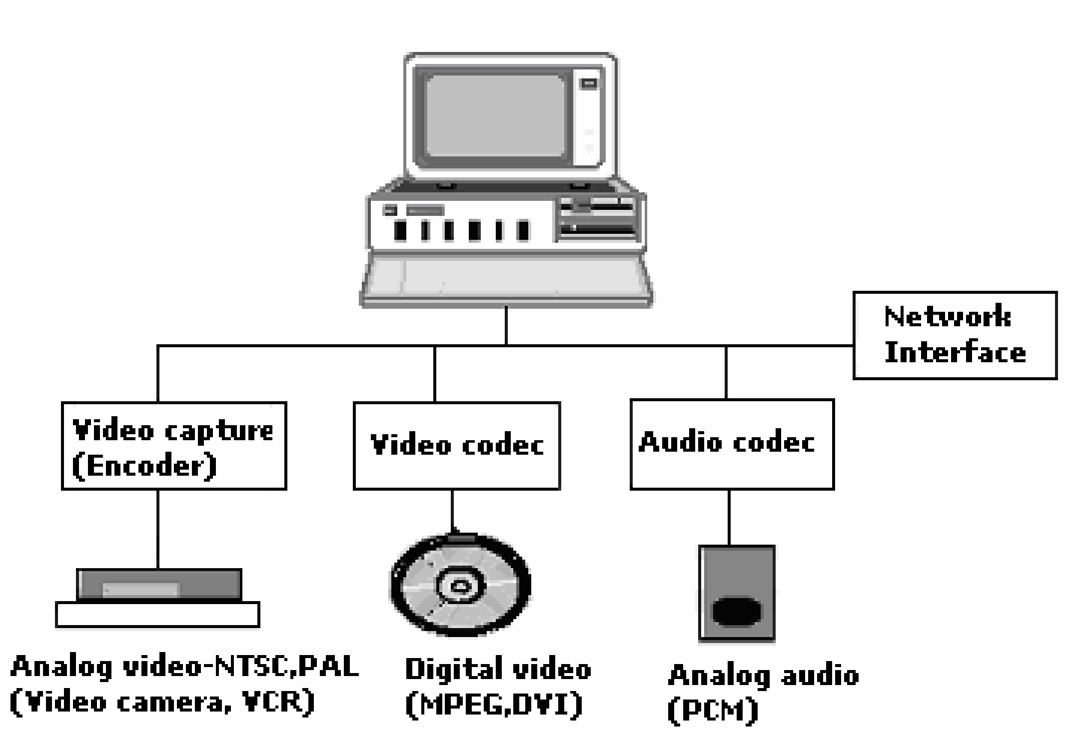

Fig. 1.1 Components of multimedia workstation

1.2 COMPRESSION TECHNIQUES

As we know multimedia contains the large amount of data. So with if transmit this much data on the network channel, we required the very high bandwidth, but with network the bandwidth available is limited. So to work with this limited bandwidth, we need to compress the data and transmit it on the network channel. By compressing the raw data we can reduce the size of the raw data and store only the important data required to reconstruct the video after decompression.

There are mainly two types of compression techniques:

Lossless compression

In lossless compression technique the original representation can be exactly recovered. No loss of single bit of data is there in this technique. This technique is used for the compression of text file or compression of database or for bit level image.

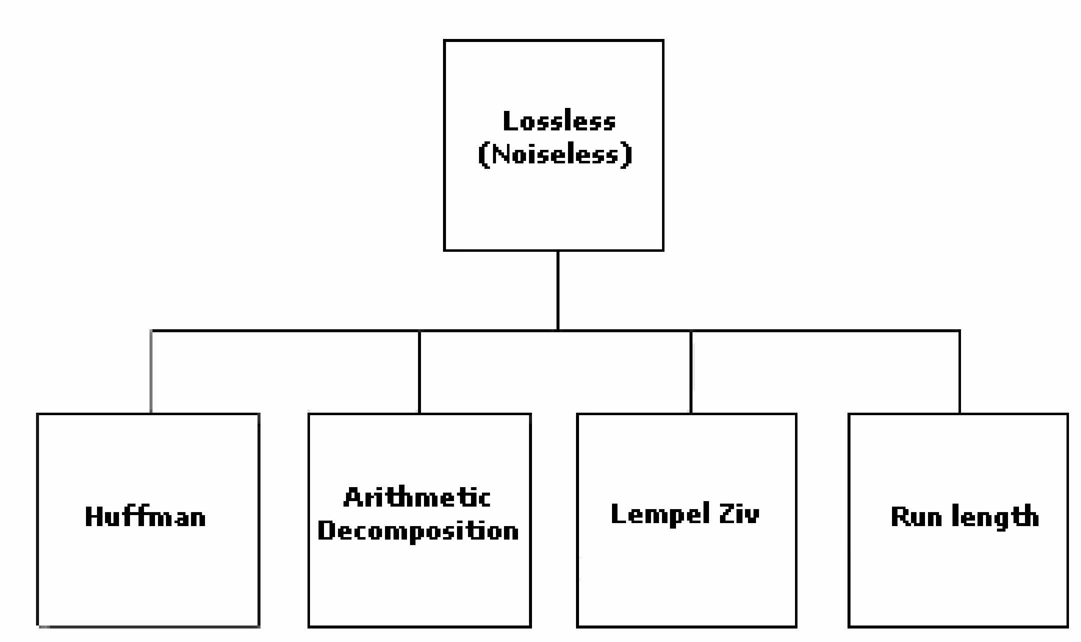

Fig. 1.2 Lossless compression techniques

Lossy compression

In lossy compression technique has some loss of data. So data can be encoded into a form that takes up a relatively small amount of space. In this technique the original representation can not be recovered perfectly. For images and video one can use lossy compression because physovisual redundancy of data.

Lossy compression technique again divided into different categories. Most usable techniques are as follows:

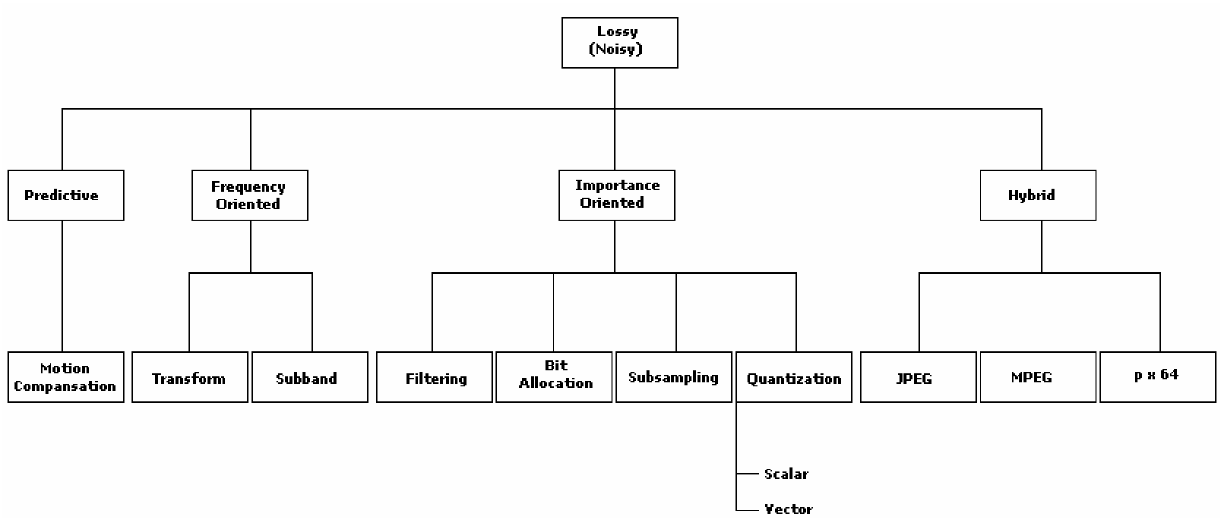

Fig. 1.3 Lossy compression techniques

Prediction compression technique

In Predictive compression technique, predicting subsequent values by observing previous ones, and transmitting only the differences between actual and predicted data. But this technique is computationally expensive. The example of this technique is motion compensation, successive frames in video sequence are often quite similar or have blocks of pixels shifted from one frame to the next. This is computationally expensive.

Frequency-oriented compression technique

In this technique data are transformed from time domain to the frequency domain or spatial domain. Two types of coding are there. In subband coding, it separates the different frequency combinations using a series of filters, and then to code with greater fidelity the frequencies that human pay particular attention to. Because of humans have different sensitivities to various spatial and temporal frequencies combinations. In transform coding, lower spatial frequencies must be carefully coded and higher frequencies need less detailed coding. Example of this is DCT transform.

Importance-oriented compression technique

Other characteristics of image besides frequency are used as the basis for compression. For example, in filter images, to get rid of details that cannot be perceived. Another is to allocate more bits to encode important parts of image like edges.

1.3 PROBLEM STATEMENT

Multimedia consists of large amount of data so its stresses all the components of the computer system as explain before. Most of today established compression standards like MPEG-2/4 and H.263(+) rely on the so called motion-estimation/compensation approach to exploit interframe correlation. This is a highly complex process, which requires a large number of operations per pixel and is therefore less appropriate for the implementation as real-time compression in a portable recording or communication device. Moreover, even though very efficient for scenes with a translatoric character, the algorithm loses much of its efficiency in scenes of morphological character or with extremely fast moving objects. And we can achieve compression ratio of 1:50 and 1:100. The similarity of the encoding and decoding process is one of the interesting properties of the 3D-DCT compression technique since it greatly facilitates the implementations of a joint en-/decoder. In available standards for video compression for motion compensation the number of operation per pixels if large. With 3D-DCT compression technique, no need to do large number of operations per pixels, reduce the computational complexity and with the 3D-DCT compression technique we can achieve the better compression ratio than available compression standard.

The paper presented by Borko Furth et al [3] introduced an adaptive compression technique for the video. In that they adapt the size of the block of video for the three types of motion level in the block of video sequence. This project work introduced an adaptive 3D-DCT compression, in which the level of motion is extended in four levels and the also adapting the quantization table according the adapted size of the block and the level of the motion content.

1.4 OUTLINE OF THESIS

This chapter gives the introduction about the project area. Next Chapter 2 is about the literature study. It is pre-requisite study for this thesis. It covers the basic compression standard for the multimedia system. Also the DCT transform technique. After that Chapter 3 covers the 3D-DCT video compression technique. It describes what are the steps required for algorithm and how 3D-DCT is implemented. Chapter 4 will describe an adaptive 3D-DCT video compression and how dynamically the cube size and quantization volume is chosen for the 3D-DCT transform by finding the motion content of the block is described. The last Chapter 5 gives the summary and conclusion of the thesis along with future work proposed.

2. LITERATURE SURVEY

2.1 INTRODUCTION

For completion of thesis, there is a need for the literature survey which is the pre-requisite study of the thesis area. This chapter included the different available standards for the multimedia system, like image and video compression standard are introduced. Basics DCT transform on which compression standards are rely is given in chapter and introduce three dimensional discrete cosines transform.

2.2 MULTIMEDIA COMPRESSION STANDARDS

Compression is designed to reduce the size of files. Compression algorithms are implemented in computer software as codec. Multimedia systems need codecs and the container formats for transmitting less number of bits over network. There are many technique used for the compression in the multimedia as multimedia contain different types of data. There are also compression standard for the multimedia, for image and video. So there are different audio and video codecs are available.

A *codec* is a compression algorithm, used to reduce the size of a data. Codec is generally used for COmpression-DECompression technique. There are audio codecs and video codecs. MPEG-1, MPEG-2, MPEG-4, Vorbis, DivX, are codecs. As the name implies, codecs are used to encode and decode (or Compress and Decompress) various types of data, particularly those that would otherwise use the large amounts of disk space, such as sound and video files. Different codecs are available for audio and video files.

Container format contains one or several streams already encoded by codecs. Very often, there are an audio streams and a video streams. AVI, Ogg, MOV, ASF, MP4, etc. are container formats. The streams contained in the container format can be encoded using different types of codecs, means you could put any codec in any container format. Unfortunately, there are some incompatibilities.

There are main three compression standards available for multimedia as given below.

1. JPEG
2. MPEG
3. H.261/263/264

Table 2.1 Compression standards for multimedia

Short Name	Official Name	Standards Group	Compression Ratios
JPEG	Digital compression and coding of continuous-tone still images	Joint Photographic Experts Group	15:1 (full color still-frame application)
MPEG	Coding of moving pictures and associated audio	Moving Pictures Experts Group	50:1 Motion-intensive application
H.261 px64	Video encoder/decoder for audio-visual services at px64 Kbps	Specialist Group on Coding for Visual Telephony	100:1 (video-based tele-communications)

Table 2.1 [1] shows the compression standards for image and video files. It shows the compression ratio achieved by these standards and also standard group by whom they are developed.

2.2.1 JPEG standard

JPEG is a commonly used standard method of compression for photographic images. The name JPEG stands for Joint Photographic Experts Group, the name of the committee who created the standard. The group was organized in 1986, issuing a standard in 1992 which was approved in 1994 as ISO 10918-1.

Image files that uses JPEG compression are commonly called "JPEG files". The most common file extension for such a file is **.**jpg, .jpeg, .jpe, .jfif and .jif. It is also possible for JPEG data to be embedded in other file types, such as TIFF format images.

JPEG is standard for a digital compression and coding of still images. We can achieve a compression ratio of 15:1 by using JPEG standard. Multimedia data which includes images uses JPEG compression standards to reduce the size of the images.

JPEG introduced the different types of operations for image compression. These operations are applied to encode the image. Following are different types of operation introduced by JPEG.

1. **Sequential encoding**

   Each mage component is encoded in single left to right and top to bottom scan. Scanning order is sequential.
2. **Progressive encoding**

   In this, the image is encoded in multiple scan for application in transmission time is long.
3. **Lossless encoding**

   The image is encoded to guarantee exact recovery of every source image sample value. In this encoding compression ratio is low due to maintaining the exact quality of reconstructed image.
4. **Hierarchical encoding**

   The image is encoded at multiple resolutions. Lower resolution version may be accessed without first having to decompress the image at its full version.

Fig. 2.1 [1] shows the JPEG encoder for special case, single component image compression.

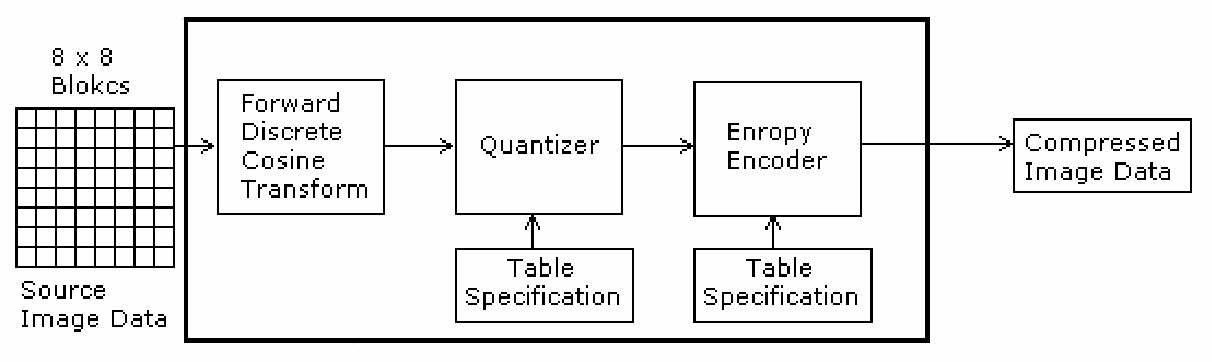

Fig. 2.1 JPEG encoder for Single component (gray-scale) image compression

As shown in Fig. 2.1, JPEG encoder will first take the block from the source image. Then apply the Forward Discrete Cosine Transform (FDCT) to input block. FDCT do the transformation from spatial domain to frequency domain. Output of the FDCT is DCT coefficient.

After this step, encoder will do quantization on the DCT coefficient of that block to make some high frequency coefficient zero. Quantization mechanism discards information which is not visually significant. It is the principal source of loss of information about images in DCT-based encoding. Here some loss of information about higher frequency components of image will occur, this is lossy compression method.

After quantization, encoder will apply the entropy encoding to reduce the number of bits. JPEG specifies two entropy coding methods, Huffman coding and Arithmetic coding. The baseline sequential codec uses Huffman coding. In JPEG entropy coding is a two step process. First step is the zig-zag sequence of quantized coefficients is converted into an intermediate sequence of symbols. And second step is symbols are converted into a data stream.

On the other side JPEG decoder all the steps explained above will apply in the reverse. First entropy decoding will do and then inverse quantization will apply and after inverse quantization last Inverse DCT (IDCT) will apply to reconstruct the image.

2.2.2 H.261/H.263 (px64)/H.264 standard

px64 video coding standard is for the video communication. In this video transmission is at px64 Kbps, (p=1,.....,30). This covers the entire ISDN channel capacity. CCITT H.261/H.263 Video codec for audiovisual services was approved in December 1990 [1].

Applications of this standard are Videophone and Videoconferencing. For videophone (desktop face-to-face visual communication) application value of p is 1 or 2 is appropriate. And for Videoconferencing, p>=6 is appropriate.

This video coding algorithm combines interframe and intraframe coding. This provides fast processing for on-the-fly video compression and decompression. The algorithm begins by coding an intraframe block using DCT, Quantization and entropy encoding like JPEG. The coded block is converted to bit stream output, and also decompressed through a reverse process and stored in an internal buffer. Each subsequent frame is coded in terms of its predecessor using predictive interframe coding. Here at least every 192th frame is coded as intra frame. This standard is used when we want to just send the video over the network. Unlike MPEG it uses only two types of frames. I-frame (intraframe coding) and P-frame (interframe coding). It is not for the video play from the DVD drive or any other device.

H.264, MPEG-4 Part 10, or AVC (Advanced Video Coding), is a digital video codec standard available that is noted for achieving very high data compression. It was introduced by the ITU-T Video Coding Experts Group (VCEG) together with the ISO/IEC Moving Picture Experts Group (MPEG) as the product of a collective partnership effort known as the Joint Video Team (JVT). The ITU-T H.264 standard and the ISO/IEC MPEG-4 Part 10 standard (formally, ISO/IEC 14496-10) are jointly maintained so that they have identical technical content.

Interframe coding

Motion compensation or motion analysis method is used to compensate interframe differences. For Real-time communication, only the closest previous frame is used for prediction to reduce the encoding delay.

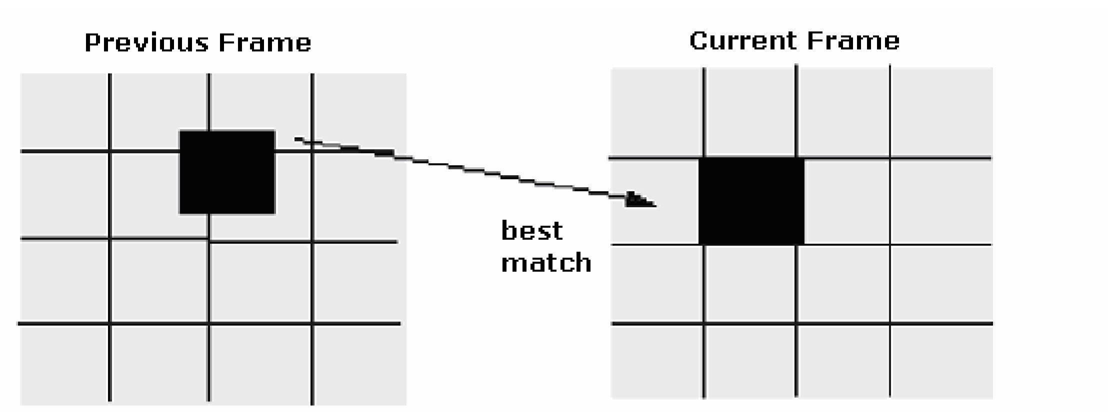

Fig. 2.2 Interframe coding

2.2.3 MPEG standard

MPEG is the standard for the moving object, video compression standard. Video compression refers to reducing the quality of data used to represent the video images. Video data contains spatial and temporal redundancy. Similarities can thus be encoded by finding the differences within a frame (spatial domain) and between frames (temporal domain). Spatial encoding is performed by taking advantage of the fact that the human eye is unable to distinguish small differences in colour as easily as it can changes in brightness and so very similar areas of colour can be encoded in a way similar to jpeg images. With temporal compression only the changes from one frame to the next are encoded as often a large number of the pixels will be the same on a series of frames. This is called as a motion estimation process.

In video compression one need to remove the temporal redundancy. This is done by Temporal Redundancy technique. MPEG do redundancy reduction by using three types of frames.

1. I-frame : Intra pictures
2. P-frame : Predicted pictures
3. B-frame : Bidirectional prediction

MPEG algorithm consider the balance between the interframe and intraframe coding. MPEG algorithm works mainly on two basic techniques. First technique is block based motion compansation or motion estimation for reduction of the temporal redundancy (inter-frame) and second technique is transform domain, DCT-based compression for reduction of spatial redundancy (intra-frame).

Fig. 2.3 [1] shows the basic MPEG compression algorithm flow for the multimedia data. Basic operations are applied on I-frame.

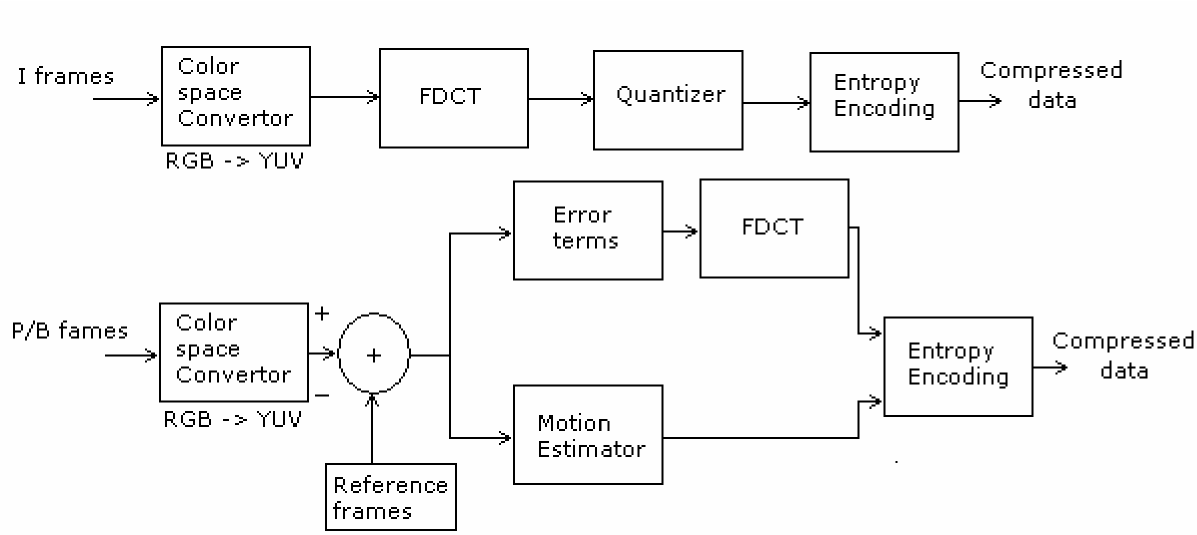

Fig. 2.3 MPEG compression algorithm flow graph

Now a day MPEG is most popular codec for video compression. Compression ratio achieved by MPEG is about 50:1. There are different versions of MPEG: J-MPEG, the first one introduced, then MPEG-1, MPEG-2, MPEG-4.

MPEG-1

It defines a group of audio and video coding and compression standards defined by MPEG (Moving Picture Experts Group). MPEG-1 video is used by the Video CD format and less commonly by the DVD-video format. MPEG-1, Audio Layer 3 is the popular audio compression standard known as MP3. As available decoding hardware were cheaper and more powerful, more advanced formats such as MPEG-2 and MPEG-4 were developed.

MPEG-1 consists of several parts or steps, as follows:

1. Synchronization and multiplexing of video and audio (MPEG-1 Program Stream).
2. Compression codec for non-interlaced video signals.
3. Compression codec for perceptual coding of audio signals. The standard defines three layers or levels of complexity, of MPEG audio coding.
    1. MP1 or MPEG-1 Part 3 Layer 1 (MPEG-1 Audio Layer I)
    2. MP2 or MPEG-1 Part 3 Layer 2 (MPEG-1 Audio Layer II)
    3. MP3 or MPEG-1 Part 3 Layer 3 (MPEG-1 Audio Layer III)
4. Procedures for testing conformance.
5. Reference software.

MPEG-1 video was originally designed to achieve acceptable video quality at resolution of 352x240 (29.97 frame per second) / 352x288 (25 frame per second) and the data rate of 1.5M bit/second. While MPEG-1 applications are often low resolution and low bitrate, the standard allows any resolution less than 4095x4095. Most of the implementations were designed with the Constrained Parameter Bitstream specification.

One major disadvantage of MPEG-1 video is that works only for progressive pictures. This disadvantage helped prompt development of the more advanced MPEG-2 version.

MPEG-2

MPEG-2 is a standard for the coding of moving pictures and associated audio information. It defines a combination of both lossy video compression and lossy audio compression methods which permit storage and transmission of movies using currently available storage media and transmission bandwidth.

MPEG-2 was the second standard developed by the Motion Pictures Expert Group (MPEG) and is an international standard (ISO/IEC 13818). Parts 1 and 2 of MPEG-2 were developed in a joint collaborative team with ITU-T, and they have a respective catalog number in the ITU-T Recommendation Series.

MPEG-2 includes a Systems part which is a part 1 that defines two distinct container formats. One is Transport Stream container format, which is designed to carry digital video and audio over somet unreliable media. MPEG-2 transport stream is commonly used in broadcast applications. MPEG-2 systems also defines Program Stream container format, which is designed for reasonably reliable media such as disks. MPEG-2 program stream is used in the DVD and SVCD standards. MPEG-2 is formally known as ISO/IEC 13818-1 and as ITU-T Rec. H.222.0.

The Video part which is part 2 of MPEG-2 is similar to MPEG-1, but also provides support for interlaced video which is used by analog broadcast TV systems. MPEG-2 video is not optimized for low bit-rates (less than 1 Mbit/s), but works well than MPEG-1 at 3 Mbit/s and above. All standards conforming MPEG-2 video decoder are fully capable of playing back MPEG-1 video streams. MPEG-2/Video is formally known as ISO/IEC 13818-2 and as ITU-T Rec. H.262

MPEG-4

MPEG-4 based on the segmentation of the different object. It divides the video into the different objects. A specific video object and scene background can be individeully defined and allows a separate coding of each object comprising a scene. It is a standard used primarily to compress audio and video digital data. Introduced in late 1998, it is the designation for a group of audio and video coding standards and related technology agreed upon by the ISO/IEC Moving Picture Experts Group (MPEG) under the formal standard ISO/IEC 14496. MPEG-4 applications are interactive multimedia, vieo games, multimedia messaging, mobile multimedia, video broadcast etc.

2.3 DISCRETE COSINE TRANSFORM

A discrete cosine transform (DCT) is part of Fourier-related transform similar to the discrete Fourier transform (DFT), which using only real numbers. DCTs are equivalent to DFTs which is of twice the length, operating on real data with even symmetry, where in some variants the input and/or output data are shifted by half a sample. Fourier transform of a real and even function is real and even. There are eight standard DCT variants available. Out of that four are common. The most common variant of discrete cosine transform is the type-II DCT, which is often called simply "the DCT". Its inverse, the type-III DCT, is correspondingly often called simply "the inverse DCT" or "the IDCT".

The DCT and in the DCT-II, is often used in signal and image processing, especially for lossy data compression, because it has a strong energy compaction property. Most of the signal information is concentrated in a few low-frequency components of the DCT. DCT is used in JPEG image compression, MJPEG, MPEG, and video compression. In these compression the two-dimensional DCT-II of $N \times N$ blocks are computed and the results are quantized and entropy coded. Mormally N will be set to 8 and the DCT-II formula is applied to each row and column of the block. For N equals to 8, after applying the DCT transform the result is an 8×8 transform coefficient (DCT coefficient) array in which the (0,0) element is the DC component whc is zero-frequency component and values with increasing vertical and horizontal index values represent higher vertical and horizontal spatial frequencies. A related transform, the modified discrete cosine transform, or MDCT is used in AAC and MP3 audio compression.

Like any Fourier-related transform, discrete cosine transforms (DCTs) express a function in terms of a sum of sinusoids with different frequencies and amplitudes. Like the discrete fourier transform (DFT), a DCT operates on a function at a finite number of discrete data points. The obvious distinction between a DCT and a DFT is that the DCT uses only cosine functions, while the DFT uses both cosines and sines. So DCT is the part of the discrete fourier transform (DFT).

Discrete cosine transform (DCT) is used to reduce the spatial redundancy in the image and frame in the compression. Discrete cosine transform can transform the value of pixel or image component from spatial domain to the frequency domain. DCT transform is used in some lossy compression algorithms, including JPEG. One of the advantages of DCT is the fact that it is a real transform, whereas DFT is complex. It is the cut-off of the DFT (Discrete Fourier Transform), in which there is real as well as imaginary value transform. This implies lower computational complexity, which is sometimes important for real-time applications.

The 1D-DCT is defined as [2]

$$D_x\{f\}(\omega) = a(\omega)\sum_{x=0}^{N-1} f(x)\cos\left(\frac{(2x+1)\omega\Pi}{2N}\right); \qquad a(\omega) = \begin{cases} \sqrt{\frac{1}{N}} & \omega = 0 \\ \sqrt{\frac{2}{N}} & else \end{cases} \qquad \text{... (2.1)}$$

which is similar to the DFT:

$$F(\omega)=\frac{1}{N}\sum_{x=0}^{N-1} f(x)e^{\frac{i2\Pi x\omega}{N}} \qquad \text{...(2.2)}$$

2D-DCT (two dimensional discrete cosines transform) is defined using the separability property as one dimensional DCT transform on the rows and on the columns, applied separately:

$$F(u,v) = D_y\{D_x\{f(x,y)\}\} \qquad \text{...(2.3)}$$

In the transform image, DC is the matrix element at location (1,1), corresponding to transform value X(0,0). After transformation high spatial X and Y frequencies correspond to high column and row indexes, respectively. Fig. 2.4 [2] shows the distribution of the DCT coefficient.

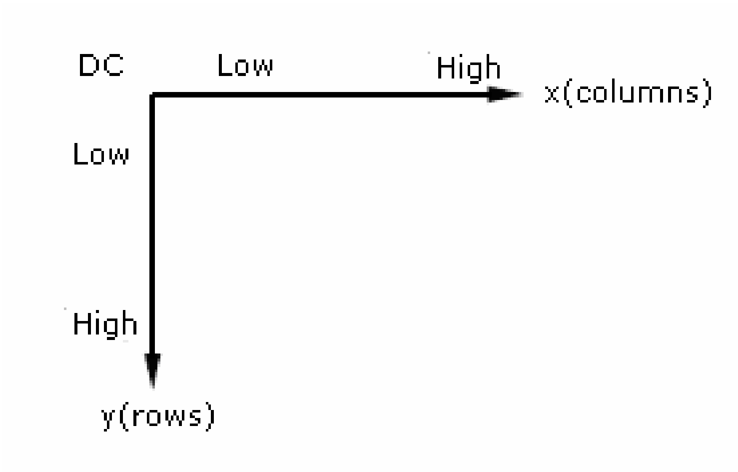

Fig. 2.4 DCT coefficients representation

According to the DCT properties, a DC component is transformed to discrete delta-function at zero frequency. So after transformation the transform image contains only the DC component where the energy of the image is concentrated. The DC value is a sum over the whole block of the image. The majority of the DCT energy is concentrated on low frequencies. The reason is the fact that natural images possess mostly low-frequency features and high-frequency features (edges) are rarely encountered.

For reason of ignoring the higher frequency components, the DCT is used for lossy compression. Energy is concentrated on low frequency. So, by removing the high frequency component, we can reduce the number of DCT coefficient and by that way we can reduce the size of data for images.

To remove the high frequency component, we need to do quantization of the DCT coefficient and then compare that quantized value with some threshold value and if it is greater than the threshold value keep it as it is and if less than quantized value then make the coefficient zero. By making the coefficient zero we can reduce the number of coefficients those are valuable. So now from this much remaining low frequency DCT coefficients we can reconstruct our original data. By applying inverse quantization followed by inverse DCT (IDCT), we can reconstruct the original image with some loss of data.

2.4 3D-DCT COMPRESSION TECHNIQUE

Compared to motion-estimation/compensation based algorithms, the 3D-DCT approach has three major advantages [3].

1. No motion estimation is required, greatly reducing the number of encoding/decoding operations per pixel.
2. Encoder and Decoder are symmetric with almost identical structure and complexity, which facilitates their joint implementation.
3. The complexity of the implementation is independent of the compression ratio.

These factors are key issues for the realization of mobile video compression systems.

Tomas Fryza et al [4] proposed the algorithm based on the 3D-DCT transform. The algorithm was similar used in specification of MPEG. Better result in compression ratio is achieved with the algorithm of 3D-DCT. This paper gives the relation between the video character and the 3D-DCT compression. Here the DCT transform is applied in three dimension, row, column, and frame number in the video sequence.

Video compression with the 3D-DCT (Three-Dimensional Discrete Cosine Transform) is based on the correlation between the image points inside one frame like used by the format JPEG, and the correlation between the same location image points in the adjacent frames. Using this algorithm one can obtain higher compression ratio with conservation of the picture quality.

In order to compare the compression quality, three most common criteria were used: Compression Ratio (CR) and Normalized Root Mean Square Error (NRMSE) and Peak Signal to Noise Ratio (PSNR). Tomas Fryza et al [4] were used the following definitions for comparing the compression quality.

$$CR = \frac{nbAll}{nbC} \qquad \text{...(2.4)}$$

where $nbAll$ is the number of all pixels in the video sequence and nbC is the number of non-zero 3D-DCT coefficient.

$$NRMSE = \sqrt{\frac{\sum_{i=1}^{nbAll} (Si - Si')^2}{\sum_{i=1}^{nbAll} (Si)^2}} \qquad \text{...(2.5)}$$

where Si is the intensity of an original picture element and Si' is the intensity of a element after decompression.

Borko Furth et al [5] proposed an adaptive 3D-DCT compression technique, which dynamically determines an optimal size of video cube based on the motion analysis.

The work presented here focuses on experimenting with the motion analyzer by extending the motion levels and including smaller size video cubes for high motion regions (for example, 4x4x8 or 4x4x4). Further experiments with various adaptive quantization tables that will further improve the quality of the video.

2.5 QUANTIZATION VOLUME

The quantization step plays an important role in the compression scheme since it reduces the magnitude of the DCT coefficients, particularly those which contribute little or nothing to the quality of the image, and it produces more zero-value coefficients. So we need not worry about these zero-value coefficients. The resulting coefficients can thus be represented with a fewer number of coefficient.

In JPEG and MPEG, the quantization values are often given by some standard quantization tables. These tables are based on the 2D-DCT characteristics and the distribution of the AC and DC coefficients for different the distribution of the AC and DC coefficients for different Nevertheless, when using 3D-DCT, a *volume* of coefficients rather than a matrix of coefficients is involved.

M.C Lee et al [6] proposed a technique for generating the quantization volume for the 3D-DCT coefficients. This paper focuses on the quantization for the 3D-DCT coefficient and the mechanism for define the scan order for the video after the quantization for entropy encoding and suggested the general formula for the quantization volume.

The quantization volume for a coefficient cube should have an entry for each coefficient. The values in the quantization volume should depend on if the corresponding coefficients are significant and also on the underlying quality factor being adopted. The magnitudes of such values are important as they affect the extent of the compression and the quality to be achieved.

Considering the dynamic range of the coefficients and the coefficient distributions, M.C. Lee et al [6] designed the quantization values based on the following guide lines.

1. The range of quantization values should be estimated based on the dynamic range of the coefficients and the given bit allocation policy.
2. The quantization values should facilitate the quantizer in discarding the high frequency components outside the selected region.
3. Since the 3D-DCT coefficients with large values are spread along the major axes, to ensure a high quality for the reconstructed video frames the quantization values should have lower values along the major axes and higher values in the remaining parts of the cube.
4. The quantization values should increase as we go down the scan order of the quantization coefficients.

Definition for the quantization volume proposed by M.C Lee et al [6] is defined by following equation.

$$q = \begin{cases} A_i\left(1 - \dfrac{e^{-\beta_i(u+1)(v+1)(w+1)}}{e^{-\beta_i}}\right) + 1 & f(u,v,w) \le C \\ A_0\left(1 - e^{-\beta_i(u+1)(v+1)(w+1)}\right) & f(u,v,w) > C \end{cases} \quad \text{...(2.6)}$$

where the u,v,w represent the location of the coefficient in the volume and f(u,v,w,) is u*v*w. Parameter value for Equation (2.6) is Ai=Ao=255, Bi=0.04, Bo=0.01, and C=10.

Fig. 2.5 [6] shows a region for capturing the dominant 3D-DCT coefficients. The 3D-DCT dominant coefficients are concentrated in along the major axes, which are captured by a shifted complement hyperboloid part. Only the coefficients in shifted complement hyperboloid part are required to be store. This part only required for reconstructing the data on decoding side.

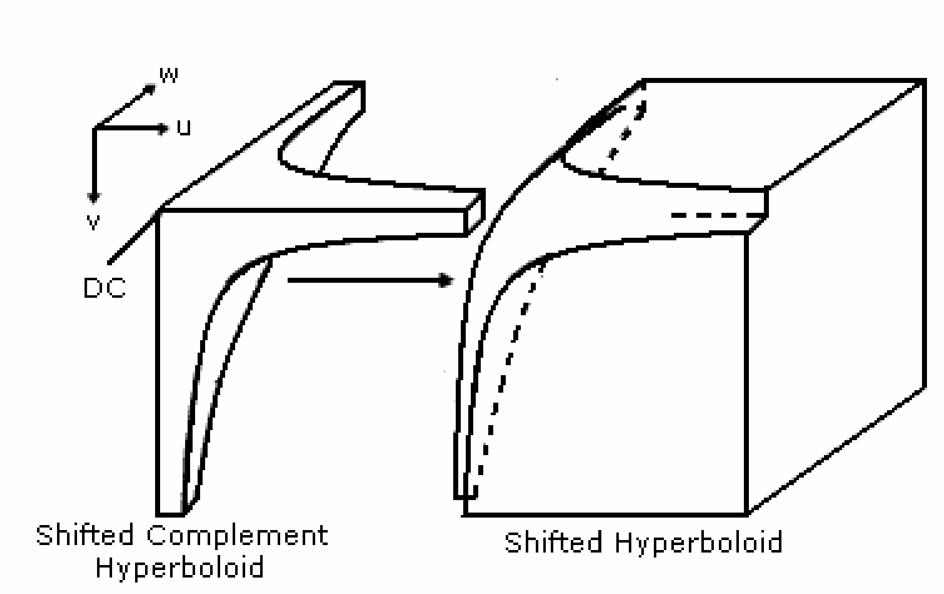

Fig. 2.5 Dominant 3D-DCT coefficient representation

So from the whole volume of the block only the shifted complement hyperboloid part is valuable. In figure left side hyperboloid is the dominant part of the volume which contains the more energy of the 3D-DCT coefficients. The right side shifted hyperboloid contains the zero-valued higher frequency components after the quantization, which is not valuable for the compression.

3. 3D-DCT VIDEO COMPRESSION

3.1 INTRODUCTION

Video compression using the 3D-DCT (Three-Dimensional Discrete Cosine Transform) technique is based on the relation between the image components inside one frame and the correlation between the image components in the adjacent frames. DCT transform is applied in the spatial as well as temporal domain and transform to frequency domain.

Three-dimensional images are images that generate a good sensation of depth, reality and existence. It has three dimension, height, width, and depth. But in the video the third dimension is the number of frame in the video sequence. There are many potential applications, one of which is three dimensional televisions. These images typically require large amounts of memory for storage and thus would occupy much bandwidth during transmission. In the development of a 3DTV system, the reduction of the information stored or transmitted, for the representation of the three-dimensional images, is therefore very important.

JPEG and MPEG standards, which are widely used in image and video compression, make use of the two-dimensional Discrete Cosine Transform (2D-DCT) for compression of an individual frame. Hence, architectures for the 2D-DCT are very popular. 2D-DCT is applied on the individual frames in the video sequence and then motion analysis mechanism applied for interframe coding.

So in multimedia contain three-dimensional images and also video for transmission. And in order to transmit them on network, we need to reduce the information of some data. Three-dimensional discrete cosine transformation (3D-DCT) is useful for such application. We can achieve better compression ratio by using this technique.

3.2 3D-DCT COMPRESSION TECHNIQUE

The increasing demand for portable digital video applications as cellular video phones or fully digital video cameras is pushing the need for highly integrated real-time compression and decompression solutions. For these types of applications, efficient lowcost/complexity implementation and low power consumption are the most critical issues. Most of today's established compression standards like MPEG-2, MPEG-4, H.263, H.264 rely on the so called motion-estimation/compensation approach to exploit interframe correlation.

Motion estimation or, motion compensation is a highly complex process, which requires a large number of operations per pixel and is therefore less appropriate for the implementation as real-time compression in a portable recording or communication device. Moreover, even though very efficient for scenes with a translatoric character, the algorithm loses much of its efficiency in scenes of morphological character or with extremely fast moving objects.

Fig. 3.1 shows basic flow graph or block diagram of the 3D-DCT video compression/decompression process.

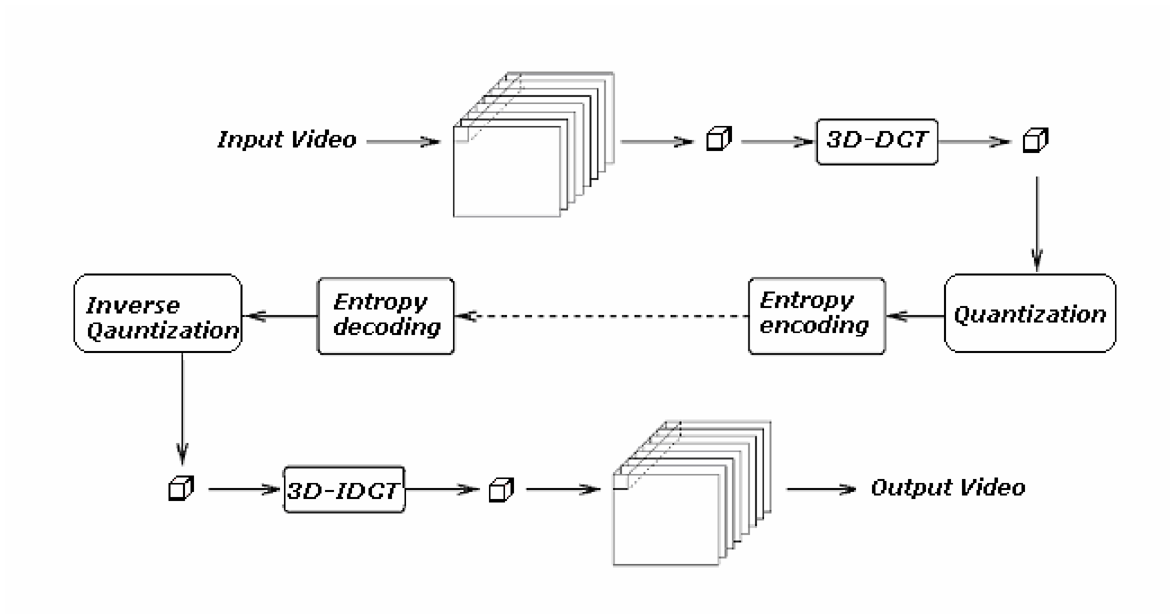

Fig. 3.1 3D-DCT compression/decompression process

3.2.1 3D-DCT Transform

The input of the 3D-DCT encoder is so-called video cube, which is a cube (dimension. *n* x *n* x *n)* made up of small elements of particular frames with the dimensions of *n* x *n*. The 3D-DCT encoder and decoder are symmetrical and each has only three parts, as shown in Figure 3.1. They are firstly the forward or inverse 3D-DCT calculations, then the quantization or inverse quantization and, finally, the variable length coding or decoding.

The equation for an N x N x N point forward 3D DCT is shown as below where the transformed outputs are represented as Z(l, *m, n), where 1, m, n = 0,* I,. . .,*N,* and the three-dimensional input sequence (representing the pixel values) are represented by *X(i, j , k), where i, j , k = 0, 1* ,..., *N.*

Let N=8, then by [7] the forward 3D-DCT is defined in the following way

$$Z_{l,m,n} = \sqrt{\frac{8}{N^3}} . \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} X_{i,j,k} . C_{i,l} . C_{j,m} . C_{k,n}$$

where .

$$C_{p,0} = \frac{1}{\sqrt{2}} \qquad and \qquad C_{p,q} = \cos\left(\frac{(2p+1)q\pi}{2N}\right) \qquad \text{...(3.1)}$$

In implementation of 3D-DCT compression algorithm, the equation used for the three dimensional DCT is equation (3.1). In the implementation of 3D-DCT, the input is the frames of the video. Video is divided into the 8x8x8 blocks and then on that blocks applying the forward three dimensional DCT.

In implementation first of all 2D-DCT is applied on each frame of the block. 2D-DCT is applied in the spatial domain. This will give the DCT coefficient of each frame in the frequency domain. After getting the 2D-DCT coefficients in frequency domain, 1D-DCT is applied on that coefficient but in the temporal domain. That is the same location coefficient in consecutive frames of the block. So by this way 3D-DCT transform is applied on the pixels of the frames. Third dimension is the consecutives frame number. This is in temporal domain.

After applying the 3D-DCT, the output of the forward 3D-DCT is the DCT coefficient. Now the DCT coefficients are represented in the frequency domain. Here the low frequency components are nearer to the DC component which is *Z(0,0,0)*. As shown in the Figure 2.4, the low frequency component are at the lower values of the x and y axis. The majority of the DCT energy is concentrated at the low frequencies, the shifted complement hyperboloid as shown in the Figure 3.5.

3.2.2 Quantization

The quantization step plays an important role in the compression scheme since it reduces the magnitude of the coefficients, particularly those which contributes little or nothing to the quality of the image, and it produces more zero-value coefficients. The resulting coefficients can thus be represented with a fewer number of bits.

The second step in the encoder is the quantization. In this process the DCT coefficient of the video bock is divided by the quantization volume as below formula.

$$F_{Qu,v,w} = round\left(\frac{F_{u,v,w}}{QTab_{u,v,w}}\right) \quad ...(3.2)$$

The quantization volume coefficients are such that it makes the high frequency DCT coefficients zero or nearer to zero. Round function is used to make the resultant coefficient as an integer value. The following equation is used to form the quantization volume.

$$QTab_{u,v,w} = 1 + \left[\left(1 + u + v + v \right) * QTY \right] \qquad ...(3.3)$$

The value of the "QTY" represents the quality factor which defines the quality of the reconstructed videos. As the value of QTY increase, the quality of the output will degrade and the compression ratio will increase. As the value of QTY increase, the value in the quantization volume will also increase and thus dividing the DCT coefficient with the higher value quantization value, more number of higher frequency components become zero and thus loss of information is increases. So the quality of the reconstructed output is degraded and there is more number of zero coefficients so compression ratio will increase.

3.2.3 Entropy Coding

After quantization as explain in previous section, third step in the video compression is entropy encoding. There are mainly to entropy coding are used, Huffman coding and Arithmetic coding. In implementation of this project, Huffman coding is used. By doing Huffman coding of the quantized values of the DCT coefficients, we can reduce the number of bits required to store the data. In Huffman coding, we need to maintain the Huffman table, which is to be used at the decoding side for decoding the Huffman code and reconstruct the original data.

3.3 COMPRESSION QUALITY CRITERIA

In order to compare the compression quality, three most common criteria are used: Compression Ratio (CR), Normalized Root Mean Square Error (NRMSE) and Peak Signal to Noise Ratio (PSNR). Tomas Fryza et al [4] were used the following definitions of compression ratio and normalized root mean square error for comparing the compression quality. Normalized root mean square error is used to compare the quality of the reconstructed frames.

3.3.1 Compression Ratio

Compression ratio is given by the following Equation (3.4). It is the ratio between the size of the original data and the decompressed data.

$$CR = \frac{nbAll}{nbC} \qquad \text{...(3.4)}$$

where $nbAll$ the number of all pixels in the video sequence and nbC is the number of non-zero 3D-DCT coefficient.

3.3.2 Normalized Root Mean Square Error

Mean squared error or MSE of an estimator is the expected value of the square of the "error”. NRMSE is used for the quality comparisons. It finds the error in the decompressed and the original values of pixels. It finds the difference in the intensity of the original pixels and decompressed pixel values in terms of error. NRMSE is given in the following Equation (3.5).

$$NRMSE = \sqrt{\frac{\sum_{i=1}^{nbAll} \left(Si - Si'\right)^2}{\sum_{i=1}^{nbAll} \left(Si\right)^2}} \qquad \text{...(3.5)}$$

where Si is the intensity of an original picture element and Si' is the intensity of a element after decompression.

3.3.3 Peak Signal to Noise Ratio

The phrase peak signal-to-noise ratio, often abbreviated PSNR, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale. The PSNR is most commonly used as a measure of quality of reconstruction in image compression etc. It is most easily defined via the mean squared error (MSE).

The PSNR is define by the following Equation (3.6)

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \quad ...(3.6)$$

$$MSE = \frac{1}{MNP}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\sum_{p=0}^{P-1}\left(X_{mnp} - X'_{mnp}\right)^2 \quad ...(3.7)$$

Where M x N is the size of the frame and the P is the number of the frames.

3.4 IMPLEMENTATION DETAILS

In the implementation of the 3D-DCT video compression, there are mainly four steps as following.

1. Color space conversion
2. 3D-DCT transform
3. Quantization
4. Entropy encoding

First step in the implementation is the color space conversion because as the human eye is less sensitive to color than to contrast. So color images have the mainly two component Chrominance component and Luminance component. In the implementation video is read and extract the frames of the video sequences. The original RGB frame of the video sequence is initially converted into the YCbCr color space, separating the luminance component (Y) of the video from the two chrominance components (Cb and Cr). For color space conversion Matlab provide the function which convert the RGB image to the Luminance and chrominance component.

```
rgb2ycbcr();
```

this function return the matrix containing the luminance components (Y) and two chrominance component (Cb and Cr).

After color space conversion, 3D-DCT transform is applied on the Luminance components. 3D-DCT is applied on the block of the video sequence of size 8 x 8 x 8. For applying the 3D-DCT transform, we need to divide the video sequence into the block. In 3DDCT video compression algorithm, third dimension is the frame number. So 3D-DCT transform is applied block-wise. For applying the 3D-DCT transform block-wise following function is used.

```
blkproc3D(Y,[8 8 8],@ThreeDCT_blk);
```

this function apply the *ThreeDCT_blk* function on each block of size 8 x 8 x 8 on the matix of the Luminance components Y. And for the 3D-DCT transform *ThreeDCT_blk* in Matlab is define below.

```
ThreeDCT_blk(X)
[L M N] = size(X);
for i=1:N
D(:,:,i) = dct2(X(:,:,i));
end

for i=1:L
for j=1:M
 B(i,j,:) = dct(D(i,j,:));
end
end
```

here first the 2D-DCT is applied on the each frame and then 1D-DCT is applied on the same location pixels in the different frames.

After applying the 3D-DCT transform the next step is the quantization. This is also applied block-wise on the 3D-DCT coefficients after transformation. For the quantization the quantization volume is defined by the Equation (3.2) and Equation (3.3). So the matrix of the 3D-DCT coefficient divided by the quantization volume defined by the Equation (3.3) Quantization volume is of size 8 x 8 x 8, the equations are applied block-wise. In the quantization, the QTY is a quality factor and is used to maintain the quality of the video after decompression. As the value of QTY increases the number of higher frequency coefficient becomes zero and the quality of the reconstructed frames degraded. In this decide the quality of the reconstructed video.

After the quantization steps, the entropy encoding is applied which is lossless compression technique. In the implementation Huffman coding is used to reduce the number of bits. The input to Huffman coding is the symbol of the quantized value and their frequency of occurrences. After Huffman encoding the output is the Huffman code for the every symbols and the dictionary which is used on the decoding side for the decoding the Huffman code.

On the decoding side all the steps are applied in the reverse order. First of all Huffman decoding is applied using the dictionary which is the essential information for decode the Huffman code of the symbols. After that the inverse quantization is applied by multiplying the quantization volume with the decoded coefficients. Once inverse quantization is finished, inverse 3D-DCT is applied and finally the video frames is reconstructed.

3.5 EXPERIMENTAL RESULTS

Video sequence named “CLEAN.avi” is used for experiment which has the following parameters.

Filename	:	'C:\CLEAN.avi'
FileSize	:	958330
NumFrames	:	243

FramesPerSecond	:	15
Width	:	260
Height	:	200
ImageType	:	'truecolor'
VideoCompression	:	'Cinepak'
Quality	:	100

This file is low motion video file and 3D-DCT works well for the low motion video file. This video sequence has the dimension of 260 x 200 and video sequence contain the 243 frames, but only 8 frames are taken for analyze.

Following Table 3.1 shown the different values of the compression quality parameter. Here the QTY represent the quality factor define in the Equation (3.3).

Table 3.1 Values of Compression quality parameter for different values of QTY

QTY	CR	NRMSE	PSNR
1	12.1	0.037694	44.819672 dB
5	35.8	0.093910	36.890920 dB
10	60.6	0.128708	34.153057 dB
15	83.0	0.151866	32.715913 dB
20	102.2	0.169948	31.738841 dB
25	120.1	0.185026	31.000467 dB

As the value of the quality factor QTY is increases the more number of higher frequency DCT coefficient after quantization become zero as a result compression ratio of the decompressed video also increases. So for maintaining the quality of the decompressed video we need to choose the appropriate QTY value.

Also as the QTY increases the error in the intensity of the original and decompressed pixel also increases, thus the NRMSE also increases with higher value of QTY. And Peak Noise-to-signal ratio decreases as the quality factor increases.

PSNR in image compression are between 30 and 40 dB. And for the video compression it is 30 – 60 dB. From the result, it is clear that QTY value for appropriate quality in 3D-DCt compression technique is within range of 1 to 25. QTY with value 1 gives better quality and with QTY equal to 25 we can get the appropriate quality in the video after the decompression, which is within the recommended range. If we increaes the value of QTY more than 25 then the PSNR value is less than 30dB which is not recommended range. Finally with the 3D-DCt compression technique we can get the compression ratio more than 100:1 with the good quality of video.

Fig. 3.2 Original frame video sequence “CLEAN”

Fig. 3.2 shown the original frames of the video sequence “CLEAN”, which is low motion video, background in the video is still image. And then Fig. 3.3 (a)-(f) shown below are the video frames after decompression for the different values of quality factor defines in Equation (3.3). Both the figure shows the first frame of the video sequence.

Fig 3.3 (a) Reconstructed frame, QTY=1, CR=12.1, NRMSE=0.037694, PSNR=44.819672 dB

Fig 3.3 (b) Reconstructed frame, QTY=5, CR=35.8, NRMSE=0.093910, PSNR=36.890920 dB

Fig 3.3 (c) Reconstructed frame, QTY=10, CR=60.6, NRMSE=0.128708, PSNR=34.153057 dB

Fig 3.3 (d) Reconstructed frame, QTY=15, CR=83.0, NRMSE=0.151866, PSNR=32.715913 dB

Fig 3.3 (e) Reconstructed frame, QTY=20, CR=102.2, NRMSE=0.169948, PSNR=31.738841 dB

Fig. 3.3 (f) Reconstructed frame, QTY=25, CR=120.1, NRMSE=0.185026, PSNR=31.000467 dB

4. ADAPTIVE 3D-DCT VIDEO COMPRESSION

4.1 INTRODUCTION

In recent years, several research efforts were made to apply 3D-DCT compression in order to achieve high compression ratios required for the transmission of video sequences over low bandwidth channels. For such applications, MPEG compression techniques that are based on 2D-DCT and motion estimation produce artifacts and blockness effects. Therefore, they are not well suited for transmission of videos at low bit rates. MPEG-2 and MPEG-4 are suited for the high bit rate communication. In the 3D-DCT compression, we assume that pixels in a full motion video are correlated in the spatial domain as well as in the temporal domain. Therefore, it is reasonable to expect that a similar concentration of energy will take place in both spatial dimensions and the time dimension. Then, the DCT can also be an appropriate transformation for representation of motion data. However, the performance of the traditional 3D-DCT will be degraded for video sequences with high motion.

In this chapter, proposed an adaptive 3D-DCT technique that dynamically performs motion analysis and according to motion level algorithm adapts the size of the video cube to be compressed and also the quantization volume of the same size of the adapted cube size. In this way, for video cubes with high motion the algorithm achieves good quality by compromising compression, while for video cubes with lower motion, it gives higher compression while still maintaining good quality.

4.2 ADAPTIVE 3D-DCT VIDEO COMPRESSION

In three dimensional DCT video compression the video sequence is divided into the blocks or we can say video cubes. In 3D-DCT compression the size of the video cube is fixed that is 8 x 8 x 8. And also the three dimensional DCT compression not works well for the high motion video.

In an adaptive three dimensional DCT compression, we have variable size of video cube instead of fixed size video cube. The size of the video cube is adapted according to the motion content of the cube.

The basic steps in adaptive 3D-DCT are as follows.

- Divide the video into the 16x16x8 blocks.
- Find the Normalized Pixel Difference (NPD) for finding the MOTION CONTENT for each block.
- According to the value of NPD, decide the sizer of the video cunbe for applying the forward 3D-DCT.
- After forward 3D-DCT, DCT coefficient is divided by the Quantization volume to make higher frequency components zero.
- Then apply the Huffman coding on the Quantized values.

Fig. 4.1 show the basic process flow of an adaptive three dimensional process

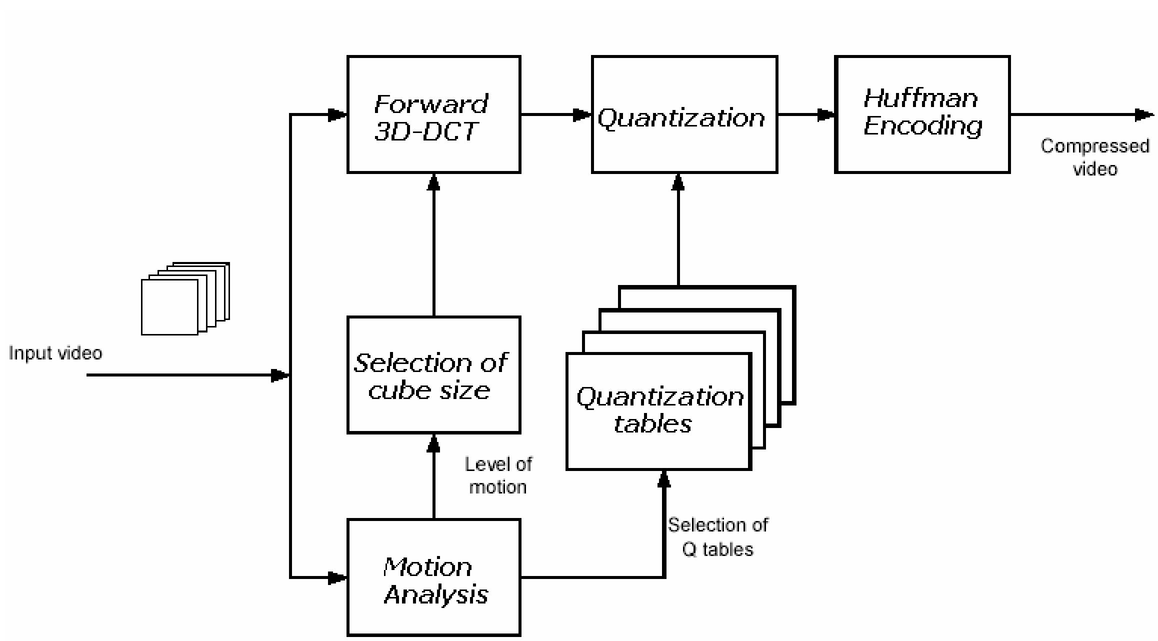

Fig. 4.1 An adaptive 3D-DCT encoding process

4.2.1 Motion Analysis

In adaptive three dimensional DCT, first of all we need to divide the video sequence of frames into the video cubes. After dividing into the cube each cube is analyzed to find the Normalized Pixel Difference. Depending on the value of the normalized pixel difference, level of motion or motion content will decided and then depending on that size of the cube for forward 3D-DCT and the quantization volume is chosen.

In the implementation, originally video sequence is divided into the video cube of size 16 x 16 x 8. And then this 16 x 16 x 8 block is analyzed for motion content, which is depends on values of normalized pixel difference.

To find the normalized pixel difference of the video cube following equation is used.[3]

$$NPD = \left(\frac{1}{MN} \right) . diff \qquad \text{...(4.1)}$$

$$diff = \sum_{i=1}^{M} \sum_{j=1}^{N} \left| \left(X_{i,j,1} - X_{i,j,8} \right) \right| \qquad \text{...(4.2)}$$

In equation (4.1), M x N is the size of the frame in video cube. Here value of M,N=16. And in Equation (4.2), the sum of difference of each pixel in first frame and eighth frame is taken.

After finding the NPD of the cube, motion level is decided depending on NPD. Following are the motion level for NPD value.

- If the NPD <= 5, then there is NO MOTION.
- If the 5 > NPD <= 25, then there is LOW MOTION.

- If the 25 > NPD <=50, then there is MIDIUM MOTION.
- If the NPD > 50, then there is HIGH MOTION.

4.2.2 Determining the cube size

After determining the level of motion, the size of the cube is determined for applying forward 3D-DCT transform. The cube size according to motion level is chosen is given in Table 4.1. Here the level of motion and respective cube size is given.

Table 4.1 the cube size for different motion levels

Level of Motion	Cube size
No Motion	16 x 16 x 1
Low Motion	16 x 16 x 8
Medium Motion	8 x 8 x 8
High Motion	4 x 4 x 4

If there is no motion then the only first frame of the video cube is considered and apply the 2D-DCT transform on that frame only, forgot about remaining seven frames. Applying 3D-DCT on a single frame is same as to apply the 2D-DCT on that frame. On decoding side the inverse 2D-DCT is applied and pixels in the reconstructed frame of the block will be duplicated in remaining seven frames. Thus we need to store the data for only one frame. Here we get the high compression ratio. For medium and high motion the cube is further sub divided into the smaller cube size as given in Table 4.1.

4.2.3 Quantization

After the pixels are transformed into the frequency domain coefficients, algorithm proceeds with the Quantization and Huffman coding. Most of the energy is now concentrated in a few low frequency coefficient, while the majority of high frequency coefficient have zero or near-zero values, and need not to be encoded.

Quantization process reduces the amplitude of the coefficients that contribute little or nothing to the quality of video, with the purpose of increasing the number of zero coefficients.

In the algorithm, the quantization table is chosen according to the level of motion. In the case of high motion cubes, cube is sub divided and the smaller size of cube is adapted. And also the quantization volume of the same size as adapted cube size is also chosen in such a way that it contains larger coefficients to maintain better quality after decoding. While for low motion or no motion cubes, the size of the cube is same as original and the quantization volume of the same size is chosen in such a way that it contain low coefficients for the better quality.

For the Quantization volume the same Equations (3.2) and (3.3) is used in quantization process. But the difference is in values of QTY. According to the level of motion of the cube, the value of quality factor “QTY” is chosen and applied for the quantization.

Following Table 4.2 shown the values of quality factor determined by the level of motion.

Table 4.2 Value of quality factor for different motion levels

Level of Motion	QTY
No motion	5
Low motion	10
Medium motion	15
High motion	20

For no motion and low motion the value QTY is chosen to be low, so the quantization volume contain smaller value coefficient. So it gives the better quality, and it increases the compression ration as per the chosen cube size. For medium and high motion, value of QTY chosen to be high, as a result quantization coefficients are larger. So the zero coefficients are high and thus maintaining the better compression ratio and quality both.

4.2.4 Huffman Coding

The entropy encoder further compresses the data using a lossless variable length encoding Huffman algorithm. By doing Huffman coding of the quantized values of the DCT coefficient, one can reduce the number of bits required to store the data. In Huffman coding, we need to maintain the Huffman table, which is to be used at the decoding side for the decoded the Huffman code and reconstruct the original data.

By using this Adaptive 3D-DCT compression technique, we can get both better compression ratio as well as quality of the reconstructed output. With the High motion block we can further divide that block into smaller sub-block, so as a result we use the quantization volume also small for that sub-block, and choose the appropriate quantization volume to get the better quality result. And with No motion block, we apply DCT transform on the first frame of the block, so need to worry about remaining seven frames. So we need to send only transform values of only first frame only, on the other side, we need to apply the inverse transform on the one frame and duplicate it to the other seven remaining frames to reconstruct the block in this way we can also increase the compression ratio. So, in adaptive 3D-DCT compression technique, we can choose the appropriate block size for 3D-DCT as well as we can choose quantization volume for required quality. We need to maintain the better quality for the medium, high motion block and for no, low motion we need to maintain high compression ratios.

In the 3D-DCT decoder, all steps from the encoding process are inverted and implemented in the reverse order.

4.3 EXPERIMENTAL RESULTS

Video sequence named “VEHICLE” is used for experiment which has the following parameters.

Filename	:	'C:\VEHICLE.avi'

FileSize	:	3261764
NumFrames	:	575
FramesPerSecond	:	25
Width	:	320
Height	:	240
ImageType	:	'truecolor'
VideoCompression	:	'DIV3'
Quality	:	4.2950e+007

Video sequence has the dimension of 320 x 240 and video sequence contain the 575 frames.

In the experiment different parts of the video sequence is taken for analysis. The size of these parts is 320 x 240 x 8, means bunches of eight frames from the video sequence are taken for experiments. These parts of video sequence contain the consecutives frames of different level of motions. Motion level is decided by the finding the motion content of the blocks in the sequence of the frames as define in section 4.2.1. The blocks of size 16 x 16 x 8 from part of the video sequence is analyzed and define the motion content of the block as define by the Equation (4.1) and (4.2) in section 4.2.1.

Part 1 contain the 251 blocks with no motion, 48 blocks with low motion, 1 block with Medium motion, and no block with high motion. Thus it contains the frames with no motion.

Part 2 contain the 157 blocks with no motion, 128 blocks with low motion, 8 blocks with Medium motion, and 7 blocks with high motion. Thus it contains the frames with low motion.

Part 3 contain the 97 blocks with no motion, 91 blocks with low motion, 64 blocks with Medium motion, and 48 blocks with high motion. Thus it contains the frames with medium motion.

Part 4 contain the 58 blocks with no motion, 84 blocks with low motion, 89 blocks with Medium motion, and 69 blocks with high motion. Thus it contains the frames with high motion motion.

For compression quality comparison, the compression quality criteria used are same as define in previous chapter in section 3.2. Same criteria are used here for comparing the compression quality. Following Table 4.3 show the different compression quality parameters for the different parts of the video sequence "VEHICLE".

Table 4.3 Compression quality parameters for different parts of video sequence

Part of video sequence (320 x 240 x 8)	CR	NRMSE	PSNR
Part 1	93.7	0.065099	30.844370 dB
Part 2	67.3	0.067886	31.102917 dB
Part 3	26.6	0.087255	28.304058 dB
Part 4	24	0.077832	29.243121 dB

As the result show above the algorithm gives the better compression ratio for the low motion video also maintaining the quality of the reconstructed video. For high motion video block the result show that the quality is maintain but with the less compression ratio.

So the proposed algorithm is works well for the video contain the no or low motion frames but it gives worst result for the video sequence contain the high motion video.

Following Fig. 4.2 (a)-(d) shown the original first frame and the first frame after decompressing from the sequence of the frames of the Parts as per Table 4.3. In the figure the upper one frame is original and the lower one frame is reconstructed frame.

Fig. 4.2 (a) Part 1, CR = 93.7, NRMSE = 0.065099, PSNR = 30.844370 dB

Fig. 4.2 (a) Part 2, CR = 67.3, NRMSE = 0.067886, PSNR = 31.102917 dB

Fig. 4.2 (a) Part 3, CR = 26.6, NRMSE = 0.087255, PSNR = 28.304058 dB

Fig. 4.2 (a) Part 4, CR = 24, NRMSE 0.077832, PSNR = 29.243121 dB

Fig. 4.2 (a)-(d) Original (upper) and reconstructed (lower) frame of video “VEHICLE”

5. SUMMARY AND CONCLUSION

5.1 SUMMARY

Multimedia system is the combination of the various data. It contains the large size of data and we need to compress the data for two main reasons, multimedia requires large storage space and limited available bandwidth of network does not allow real time video data transmission. There are different compression standards are available for the multimedia like JPEG, MPEG and H.261 px64. JPEG is the image compression standard whereas MPEG and H.261 px61 are video compression. In video compression technique three main steps are there. First, DCT transform of the picture element from spatial domain to time domain. Second, Quantization which makes the high frequency components zero. Third, Entropy encoding which include the Huffman coding or arithmetic coding and which is lossless compression technique. Compression standards for multimedia are works with the 2D-DCT transform. DCT transform picture elements from spatial domain to frequency domain. The major energy of the image is concentrated at low frequency component. So the low frequency coefficients are most dominant component for the compression. Standards give the compression ratio up to 100. In MPEG and H.26(+), compression algorithm is complex and require more operations per pixels.

In project video compression is based on three dimensional DCT (3D-DCT). 3D-DCT transform works with spatial as well as temporal domain and transform it to frequency domain. 3D-DCT compression technique is relatively less complex unlike MPEG/H.26(+) and also requires less number of operations per pixels. With the 3D-DCT compression technique, we get the higher compression ratio. But 3D-DCT compression technique works well for the low motion video rather then high motion video. In this project also an Adaptive 3D-DCT compression technique is also proposed. In this algorithm the block size for applying 3D-DCT transform and the quantization volume are dynamically chosen based on the level of motion of that block of the video sequence. Four level of motion is define, no motion, low motion, medium motion, and high motion.

5.2 CONCLUSION

The 3D-DCT (Three Dimensional Discrete Cosine Transform) for video compression was examined. First, an encoder and decoder based on the 3D-DCT transform were introduced. Second, the equation of quantization volume with quality factor “QTY” was introduced for the quality of the decompressed video. As the quality factor in the quantization volume increases the quality of the decompressed video is degrade and the compression ratio is increase. The range of the QTY is 1 to 25. For these values of QTY, PSNR and NRMSE are within the recommended range.

Also an Adaptive 3D-DCT video compression was introduced. In an adaptive 3D-DCT compression technique, both the side of the block for 3D-DCT transform of video sequence and the quantization volume is determine dynamically based on the level of motion in the block. For the low motion frame sequences, compression ratios obtained are up to 1:100 and maintaining the good quality of decompressed sequence of frame. But algorithm is not well suited for the high motion frame sequence. so proposed algorithm is well suited for the application like videoconferencing, videophone, wireless video and video over Internet.

5.3 FUTURE WORK

As proposed algorithm for an adaptive 3D-DCT video compression is yet not well suited for the high motion video and still the compression ratio for low motion video is not high as 100:1, future work will focus on experimenting with high compression ratio. And intend to experiment with quantization volume for further improvement in the compression ratio and quality for the high motion video. And also intend to experiment with the video streaming with this compression algorithm.

REFERENCES

[1] Borko Furth, *"Multimedia System and Compression Standards",* in Lecture notes, Workshop on Multimedia Systems, 2006, Nirma University.

[2] *"DCT-Based Image compression"*,Vision Research and Image Science Laboratory. http://visl.technion.ac.il

[3] Andreas Burg, Roni Keller, Juergen Wassner, Norbert Felber, Wolfgang Fichtner, *"A 3D-DCT Real-Time Video Compression System for Low Complexity Single-Chip VLSI Implementation".*
Integrated Systems Laboratory, Swiss Federal Institute of Technology (ETH) Zurich.

[4] Tomas Fryza, Stanislav Hanus, *"Relation Between Characters of Real Video Sequences and 3D-DCT Compression".*
Video/Image processing and Multimedia Communication,2003. 4th EURASIP conference, Volume 1, Issue, 2-5 July 2003, Page(s):107-112.

[5] Borko Furth, Ken Gustafson, Hesong Haung, and Oge Marquos, *"An Adaptive Three-Dimensional DCT Compression Based on Motion Analysis".*
Florida Atlantic University, Department of Computer Science and Engineering, Boca Raton, Florida, U.S.A.

[6] M.C Lee, Raymond K.W. Chan, Donald A. Adjeroh*, " Quantization of 3D-DCT Coefficients and Scan Order for Video Compression".*
Journal of Visual Communication and Image Representation, Volume 8, Issue 4, December 1997, Page(s):405-422.

[7] A. Aggoun, I. Jalloh, *"A Parallel 3D DCT Architecture for the Compression of Integral 3D Image".*
Electronics, Circiuts and system, 2001, ICECS 2001, The 8th IEEE International conference, Volume 1, Page(s):229-232.