DECODING OF MPEG-2 VIDEO BITSTREAM

THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF

MASTER OF ENGINEERING

OF THE BHARATHIAR UNIVERSITY

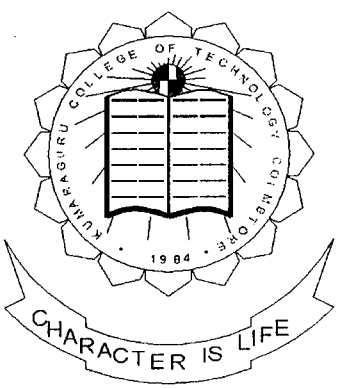
By

C.UMAPATHY

(Reg. No. 9937K0015)

Under the Guidance of

Dr. S. Thangasamy, Ph.D,



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE – 641 006

DECEMBER 2000

CERTIFICATE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE – 641 006

Certified that this is a bonafide report of thesis work of entitled DECODING MPEG-2 VIDEO BITSTREAM done by

Mr. C. UMAPATHY

(Reg. No. 9937K0015)

at

KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE – 641 006

During the year 1999 – 2000

S Thangasamy 11/1/01 ..... S. Thangasamy 11/1/01 .....

Guide **Head of the department**

Dr.S. Thangasamy **Dr.S.Thangasamy.**

Department of Computer Science and Engineering,
Kumaraguru College of Technology.

Place : Coimbatore

Date :

Submitted for viva – voce examination held at

Kumaraguru College of Technology on 20.1.2001

S Thangasamy 20/1. .....

INTERNAL EXAMINER

M Vale 20/1/2001. .....

EXTERNAL EXAMINER

DECLARATION

I hereby declare that this thesis work entitled " **DECODING OF MPEG-2 VIDEO BITSTREAM** " being submitted by me for the award of the degree of Master of Engineering (COMPUTER SCIENCE AND ENGINEERING) of the Bharathiar University, Coimbatore is a bonafide work carried out by me and the results embodied in the thesis have not been submitted to any other university or institute for the award of any degree or any diploma.

C.UMAPATHY
REG.NO : 9937K0015

SIGNATURE

Place : Coimbatore
Date : 11-1-2001

ACKNOWLEDGEMENT

First and foremost I would like to thank the almighty for showering his blessings on me to complete this project.

I express my profound gratitude to **Dr. S. Thangasamy, The Head of the Computer Science and Engineering Department** in helping me in my endeavour to complete my project.

I am also committed to thank **Mr. R. Kannan, Asst Professor** who has provided me the requisite and valuable guidance in completing my project.

I also thank **our principal Dr. K. K. Padmanaban,** for providing me an opportunity to do this project.

I have no words to express my sense of thankfulness to all those who have helped me in completing my project.

SYNOPSIS

This project will examine the Digital video compression and the different standards of MPEG. The focus is on the handling of the video at the receiving end with a software only decoder to decompress the video clips encoded based on the MPEG-2 Standard. One important thing about MPEG is that it is a standard for the decoders. The standard defines the bitstream so that a MPEG compliant decoder can display the video. Quality issues arise and are determined by the choice of the decoder.

Codecs can either be symmetric or asymmetric. A symmetric algorithm uses an equal amount of time for compression and decompression. It is common in real time video captures in applications like videoconferencing. An asymmetric codec takes more time in the compression stage. This process works well for the CD-ROMs. Once the codec saves the non-redundant data, it statistically processes the arrangement of pixels in the image. The statistical data arrangements are then encoded and the codec then uses a variety of compression methods.

In the case of MPEG, discrete cosine transform, DCT, is used. A frame is divided into blocks, usually 8 X 8 pixels each and transform mechanisms work on each of the individual blocks. DCT converts pixel intensities into a frequency- based equivalent. The result is a series of numbers that represent every fine detail in a pixel block. Compression then comes by eliminating the representative numbers after a certain point in the series. The consequence is compressed video with a loss of fine detail but the level of detail is not detectable.

CONTENTS

CHAPTER 1: INTRODUCTION

With the development of various Multimedia compression techniques and significant increase in desktop computer performance and storage, the widespread exchange of multimedia information is becoming a reality. The key obstacle for many applications of digital imaging is the vast amount of data required to represent a digital image directly.

Modern image compression technology offers a possible solution to minimize the cost involved in storage and transmission. But this alone is not sufficient as a standard image compression method is the need of the hour to enable the interoperability of equipment of different manufacturers. The Moving Pictures Experts Group or MPEG was formed under the auspices of ISO (International Standards Organization) and IEC (International Electro technical Committee with its goal to generate standards for digital imaging.

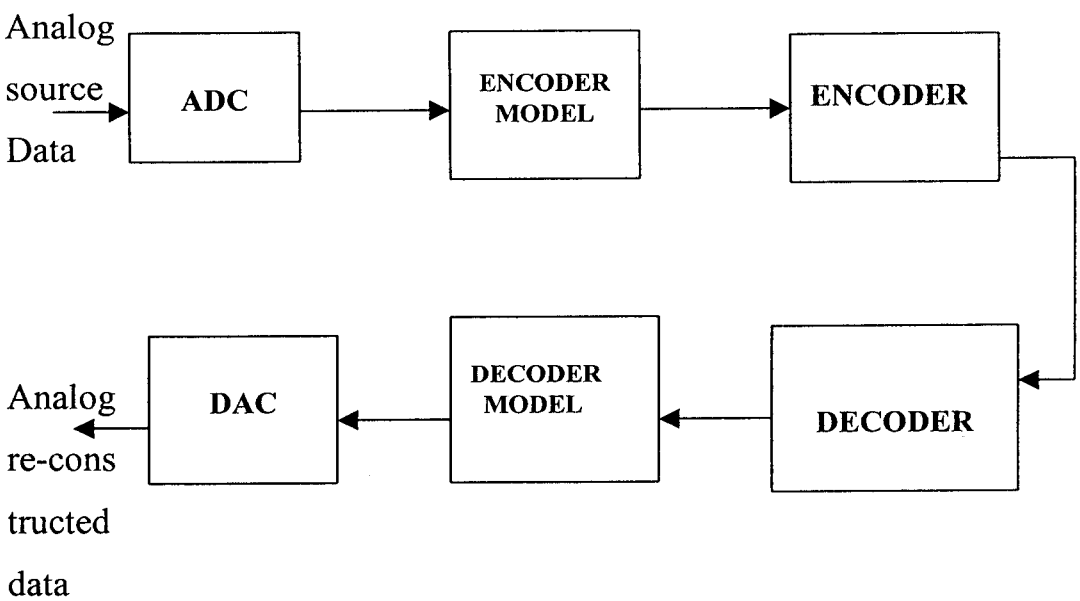
MPEG is regarded by many as the only world recognized standard for digital video compression. Compression is a reversible conversion of data to a format that requires fewer bits, usually performed so that the data can be stored or transmitted more efficiently. The size of the data in the compressed form (C) relative to the original size (O) is known as the compression ratio R = C/O.

If the inverse of the process ie decompression, produces an exact replica of the original data then the compression is a loss less one. On the other hand lossy compression has a higher compression ratio.

Lossy compression allows only an approximation of the original to be generated. For image compression, the approximation usually becomes poor as the compression ratio increases. The success of data compression depends largely on the data itself and some data types are inherently more compressible than others. Generally some elements within the data are more common than others and more compression algorithms exploit this property known as redundancy. The greater the redundancy within the data, the more successful the compression will be. As the compression of video is nothing but the compression of still pictures dispersed in time, there redundancy phenomenon comes in handy and on this account digital video is ideal for compression.

A device that compresses data is known as an encoder whereas a device that decompresses the compressed data is known as a decoder. This device can either be a software or may be a hardware. A device that can act as both an encoder and a decoder is known as a codec. Development in recent years, of lossy techniques specifically for image data has contributed a great deal to the realization of digital video applications. In video compression certain redundant details are detected and then stripped off.

Fig : Overview of a compression system



(1.1) THE NEED FOR IMAGE COMPRESSION

A sequence of pictures with accompanying sound track can occupy a vast amount of storage space when represented in digital form. For example suppose the pictures in a sequence are digitized as discrete grids or arrays with 360 pixels per raster line and 288 lines/picture and assuming that the picture sequence is in colour a three colour separation can be used for each picture.

If each colour component in the separation is sampled at a 360 X 288 resolution with 8bit precision, each picture occupies approximately 311 KB. If the moving pictures are sent uncompressed at 24 pictures/s, the raw data rate for the sequence is about 60Mbps and an oneminute video clip occupies 448 MB.

The high bit rate that result from the various types of digital video make their transmission through their intended channels very difficult. Even entertainment video with the modest frame rates and dimensions would require bandwidth far in excess of that available from a single speed CDROM. Thus the delivery of entertainment video on CD would be impossible in the normal form.

Typical parameters for various applications of digital video:

Application	Frame rate	Dimension	Pixel depth
Multimedia	15	320 X 240	16
Entertain Ment TV	25	640 X 480	16
Video Telephony	10	320 X 240	12
HDTV	25	1920 X 1080	24

(a) Pixels: The component sample values at a particular point form a pixel (picture element). If all the three components use the same sampling grid, each pixel has three samples, one from each component.Thus, a pixel is defined to be the colour representation at the highest sampling resolution, but not all samples that make up the pixel are at that resolution.

(b) Frame rate: The number of frames per second. The illusion of motion can be experienced at frame rates as low as 12 frames/sec, but modern cinema uses 24 frames/sec and television uses 725 frames /sec.

(c) Frame dimensions: The width and height of the image expressed in the number of the pixels. Digital video Comparable to television requires dimensions of around 640 X 480 pixels.

(d) Pixel depth: It is the number of bits per pixel. In some cases it might be possible to separate the bits dedicated to luminance from those for chrominance. In others it might be used to reference one of a range of colours from a known palette.

Data transfer rate required by a video telephony system is far greater than the bandwidth available over the Plain Old Telephone System (POTS). 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 gains in storage transmission and processor technology have been achieved in recent years, it is primarily the reduction of the amount of data that needs to be handled makes the digital video usage a widespread possibility. This reduction of bandwidth has been made possible by the advances in the compression technology.

(1.2) THE NEED FOR STANDARDIZATION

When the mpeg standardization effort was initiated in 1988, several different industries were converging on digital video technology. The computer industry was looking to expand beyond its traditional text and graphics capabilities into interactive multimedia with audio, video and still images. The consumer industry saw digital video as a means for improving the capabilities and interactivity of video games and entertainment media.

Compact discs were already being used for digital storage and the storage capacity was sufficient for compressed digital video. The telecommunications industry was taking advantage of the maturing compression technology to standardize teleconferencing. The debate about broadcasting a digital versus a higher bandwidth analog signal for high definition television (HDTV) was about to start and direct broadcast of digital video from satellites was also being considered. Cable television companies were exploring the use of digital video for satellite uplinks and downlinks and they were also considering delivery of digital video to home. Together with the microprocessors that would be used for control of home converters. Direct digital links into the home raised the possibility of new function through the cable networks.With the sharing of technologies among these diverse industrial interests, standardization was a natural development. If standardization of digital video could be achieved, the potentially high cost involved in making the interoperability of equipments of different manufacturers possible can be brought down to a great extent.

Thus the cost reduction implicit in having a common digital video technology for all has been a key driving force for standardization. The convergence of these interests and the resulting standardization is lowering the barriers that are usually present to the deployment of new technology.

(1.3) APPLICATIONS OF DIGITAL VIDEO

Digital video has a number of unique properties that makes possible applications that could not be realized using analog video. Firstly, digital video can be manipulated more easily than analog video. In addition to this digital video can be stored on random access medium unlike its analog counterpart which is usually stored in a sequential access medium like a magnetic tape. This random access allows for interactivity as individual video frames are addressable and can be accessed quickly which forms the basis for many applications which rely on such properties which are unique only to digital video. Digital video is capable of being duplicated without loss of quality, which is vital for editing applications.

Desktop video editing is possible on most high-end desktop computers and may come with a special hardware to digitize video. Using applications such as Adobe premiere users can edit digital video on their desktop to produce digital video of modest dimensions and integrate it into other applications. However desktop computers would not be able to cope with digital video, unless it was easy to store and transmit.

The ability to easily store and transmit digital video is the most important property. It allows the users to add video attachments to e-mail,

sometimes referred to as v-mail and makes possible video telephony. The attraction of video telephone is primarily due to the high cost of travel. Video conferencing is expected to become so widespread that it will have a significant impact on business air travel. Digital video in compressed form can be transmitted using less bandwidth than the analog one thereby making it possible to provide many channels where before there were only a few or none. The low bandwidth requirements of digital video also makes it easy to store and digital video can be stored on a Compact Disc. Integrating digital video into interactive applications along with sound, animation, photographs and text known as multimedia is one of the most important applications of digital video.

Multimedia applications exploit all of the properties of digital video. The ease with which video can be manipulated allows low cost production of video sequences by non-television professionals. The low bandwidth requirements of digital video allows it to be stored on compact discs, hard discs and to be displayed on computer screens.

In addition the ease with which various segments can be accessed allows them to be integrated into highly interactive applications. Video on demand is currently available on a trial basis. It differs from pay per view (where viewers call up and gain access to a predetermined movie at a set time) in that viewers can choose the movie and the time that they want to watch. As the individual video segments can be addressed individually video can appear in response to the viewers requests.

CHAPTER 2: OVERVIEW OF MPEG

The Moving Pictures experts group or MPEG is a committee that was formed under the auspices of the International Standards Organization in the year 1988. It is a working group whose mandate is to generate standards for digital video and audio compression. Two standards have been ratified and they are called MPEG-1 and MPEG –2. They are the encoding standards that convert analog video and audio input signals into compressed digital files.

Today there are three phases defined:

MPEG –1: Coding of moving pictures and the associated audio for digital storage media at upto 1.5 Mbps. (speed of data retrieval from the storage)
MPEG –2: Compression at a high resolution and a higher bitrate of about 6.0 Mbps.
MPEG –3: Compression at a very low rate ie in the order of Kbps.

The MPEG –1 in turn consists of four main parts. They are,
1. System: describes the synchronization and multiplexing of video and audio.
2. Video: describes compression of video signals.
3. Audio: describes compression of audio signals.

4. Compliance testing: describes the procedures for determining the characteristics of coded bitstreams and the decoding process and for testing compliance with the requirements stated in the other parts.

(2.1) SYSTEM LAYER

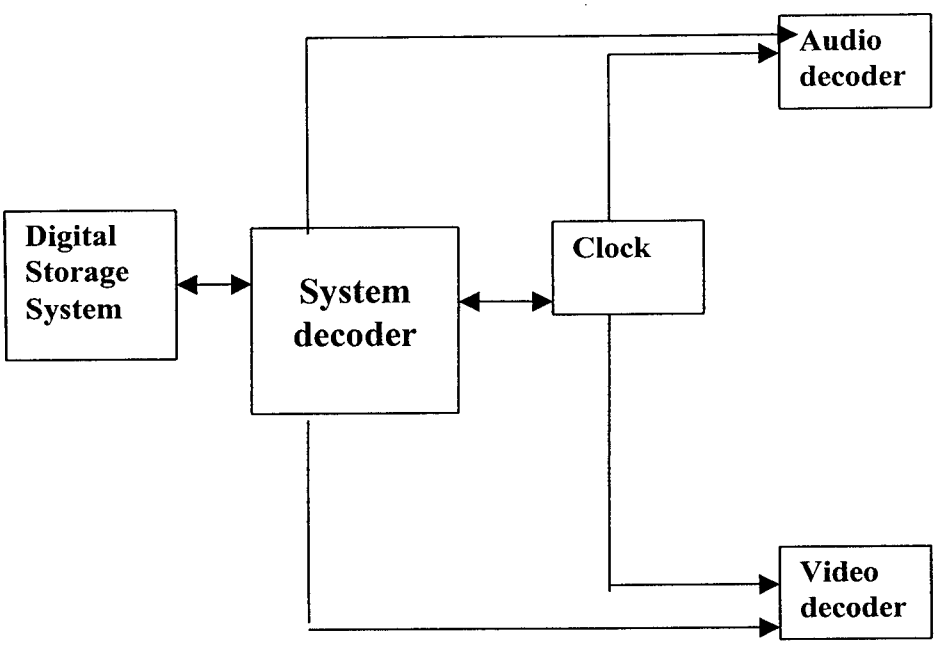
The mpeg system layer has the basic task of combining one or more audio and video compressed bitstreams into a single bitstream. It defines the data stream syntax that provides for timing control and the interleaving and synchronization of audio and video bitstreams.

From the systems perspective, an mpeg bit stream is made up of a system layer and compression layers. The compression layers contain the data fed to the audio and video decoders, whereas the system layer provides the controls for de-multiplexing the interleaved compression layers.

The bitstream consists of a sequence of packs that are in turn divided into packets. Each pack consists of a unique 32 bit pack start code and header followed by one or more packets of data. Each packet consists of a packet start code and header, followed by the packet data (compressed audio and video data).

The system decoder parses this bitstream and feeds the separated audio and video data to the appropriate decoder along with the timing information.

Fig: System structure - mpeg decoding.



System target decoder:

The mpeg system uses an idealized decoder called the system target decoder (STD). This idealized decoder interprets the pack and packet headers, delivering the elementary bitstreams to the appropriate audio or video decoder. A major role of this decoder is to prevent buffer overflow and underflow.

The buffering requirements are described in terms of the decoder rather than the encoder; for this reason, buffer overflow occurs when the decoder does not remove data quickly enough, ie when the compression is too good. Conversely the buffer underflow occurs when the decoder

removes data too quickly; ie the encoding process produces too much data per picture.

In the STD the bits for an access unit are removed from the buffer instantaneously at a time dictated by a decoding time stamp (DTS) in the bistream. The bitstream also contains another type of time stamp, the Presentation Timestamp (PTS). Buffer overflow and underflow is controlled by the DTS; synchronization between audio and video decoding is controlled by the PTS. At some points in the video sequence both the DTS and PTS are identical and only the PTS is used.

Data are fed to the respective buffers from the system decoder at irregular intervals as needed to demultiplex the interleaved bitstreams. Conceptually, each of the interleaved bitstreams is allocated a separate buffer in the STD, with a size determined by the encoder and communicated to the decoder in the system layer.

On the basis of the header information in the bitstream itself, the system control knows when to switch the bitstream to the appropriate buffer. Bits not intended to those buffers are sent to the control buffer. The bits are removed in blocks called access units.

An mpeg packet of data has a length defined in the packet header. Packet lengths are typically structured to match to the requirements of the digital storage or transmission medium, and therefore do not necessarily conform to the audio and video access units created by the encoders. However a packet can contain only one type of compressed data.

(2.2) MPEG AUDIO

The mpeg audio coding standard defines three layers of increasing complexity and subjective quality. It supports sampling rates of 32,44.1 and 48 KHz. At 16 bits/sample the uncompressed audio would require about 0.5Mbps. After compression, the bit rates for monophonic channels are between 32 and 192 Kbps and that for the stereophonic channels is between 128 and 384 Kbps.

The mpeg audio coding techniques take advantage of the psycho acoustic properties of human hearing. Similar to the threshold for observing visible patterns, there is also a frequency dependant threshold for the perception of audio stimuli. There is also an effect known as simultaneous masking in which the presence of audio signal masks the other signal's perception. A temporal masking effect can also occur in which audio signals immediately before and after a masking signal are less perceptible.

The mpeg audio system first segments the audio into windows 384 samples wide. Layers I and II use a filter bank to decompose each window into 32 sub bands, each with a width of about 750 Hz (for a sampling of 48 KHz).

As is typical for subband coding, each sub band is decimated such that the sampling rate per sub band is 1.5KHz and there are 12 samples per channel. A fast Fourier transform of the audio input is used to compute a

global masking threshold for each sub band and from this a uniform quantizer is chosen that provides the least audible distortion at the required bitrate.

One of the problems introduced by quantization is pre–echoes. Pre-echoes can occur when a sharp percussive sound is preceded by silence. When the signal is re-constructed, the errors due to quantization tend to be distributed over the block of samples, thereby causing an audible distortion before the actual signal.

Pre–echo control is an important part of layer III of audio coding. It adds a modified discrete Cosine Transform (DCT) decomposition of the sub bands to get much finer frequency distribution. It also adds non-uniform quantization (large signals mask large quantization errors), entropy coding and dynamic window switching which provides better time resolution and allows better control of Pre–echoes.

(2.3) MPEG VIDEO

A video sequence is made up of individual pictures occuring (usually) at fixed time increments. There are three basic components for each of these pictures. Colour is expressed in terms of a luminance component and two chrominance components.

The luminance provides a monochrome picture whereas the two chrominance components express the equivalent of colour hue and saturation in the picture. Any colour can be synthesized by an appropriate mixture of

three properly chosen primary colours and Red, Green and Blue (RGB) are usually chosen. Another system known as YUV, allows colour to be approximated using only two variables where Y represents the luminance and U and V the chrominance.

When an analog signal is digitized it is quantized. Quantisation is the process by which a continuous range of values from an input signal is divided into non-overlapping discrete ranges and each range is assumed a unique symbol.

A digitized monochrome photograph, for example, might contain only 256 different kinds of illumination or gray level. Such an image is said to have a pixel depth of 8 bits. A higher quality colour image might be quantized allowing 24 bits per pixel.

Digital video can be characterized by a few variables:

(a) Sequences, Pictures and samples:

An mpeg video sequence is made up of individual pictures occuring at fixed time increments. Since the pictures are in colour, each picture must have three components. Colour is expressed in terms of a luminance component and two chrominance components.

The luminance provides a monochrome picture whereas, the two chrominance components express the colour hue and saturation in the

picture. These components are a mathematical equivalent of the primary colours and can be more efficiently compressed.

Each component of a picture is made up of a 2-D grid or array of samples. Each horizontal line of samples in the 2-D grid is called a raster line, and each sample in a raster line is a digital representation of the intensity of the component at the point of the raster line. However the luminance and the chrominance components need not have the same sampling grid. Since the eye does not resolve rapid spatial changes in chrominance as readily as changes in luminance, the chrominance components are typically sampled at a lower spatial resolution.

(b) Frames and fields

In broadcast analog video standards such as NTSC (National Television System Committee) or PAL (Phase Alternating Line) video sequences are temporally subdivided into frames and raster lines. However the signal within each raster line is analog rather than digital.

Each frame is further divided into interlaced fields. Each field has half the raster lines of the full frame and the fields are interleaved such that alternate raster lines in the frame belong to alternate fields.

MPEG video is specifically designed for the compression of the video sequences, which is nothing but a group of pictures. Except for the special case of a scene change these pictures tend to be quite similar from one to the next. The compression techniques used by the MPEG take

advantage of this similarity of one picture to the next in a sequence. Compression techniques that use this information from other pictures in the sequence are called interframe techniques. In case of a scene change the interframe technique doesnot work and the compression model should look for redundancy within a picture and such a technique is called as intraframe technique.

MPEG uses the lossy compression technique which effectively leaves out the video information which is imperceptible to the human eye. With video compression the more you compress a frame or data stream the more losses occur. The trick with digital compression is to balance the compression ratio with the resulting image quality.

(c) Mpeg video layers:

MPEG video is broken up into a hierarchy of layers to help with error handling, random search and editing and synchronization, for example with an audio bitstream. From the top level the first layer is known as the video sequence layer. The second layer down is the Group of pictures layer which is composed of one or more groups of I frames (intra) or P and B frames (non-intra) pictures that constitute the clip .

The third layer down is the picture layer itself and the next layer beneath is called the slice layer. Each slice is a contiguous sequence of raster ordered macroblocks. Each slice consists of macroblocks which are 16 X 16 arrays of luminance pixels with two 8 X 8 arrays of associated chrominance.

The macroblocks can be further divided into distinct 8 X 8 blocks for further processing such as transform coding. Each of these layers has its own unique 32 bit start code. The term intra coding refers to the fact that the various loseless and lossy compression techniques are performed relative to information that is contained only within the current frame. In other words no temporal processing is performed outside of the current frame.

A video sequence is simply a series of pictures taken at closely spaced intervals in time. Except for the special case of a scene change, these pictures tend to be quite similar from one to the next. Intuitively a compression system ought to be able to take advantage of this similarity.

The compression techniques used by MPEG take advantage of this similarity or predictability from one picture to the next in a sequence. Compression techniques that use information from other pictures in the sequence are called interframe techniques. When a scene change occurs and sometimes for other reasons, interframe compression does not work and in those cases the compression model should take advantage of the similarity of a given region of a picture to immediately adjacent areas of the same picture. Such techniques that use the information from a single picture is called intraframe techniques.

These two compression techniques, inter frame and intra frame are at the heart of the MPEG video compression algorithm. The outermost layer of an mpeg video stream is the video sequence layer. Except for certain critical timing information in the mpeg systems layer, an mpeg video

bitstream is completely self contained and is independent of other video bitstreams.

Each video sequence is divided into one or more groups of pictures and each group of pictures is composed of one or more pictures of three different types, I, P and B. I pictures (intra coded pictures are coded independantly, entirely without reference to other pictures . P and B pictures are compressed by coding the differences between the picture and reference I or P pictures, thereby exploiting the similarities from one picture to the next . P pictures (predictive coded pictures) obtain predictions from temporally preceding I or P pictures in the sequences whereas the B pictures (bidirectionally predictive coded pictures) obtain predictions from the nearest preceding I or P pictures in the sequence. Different regions of B pictures may use different predictions and may predict from preceding, upcoming or both pictures or neither. Similarly P pictures may also predict from preceding pictures or use no prediction. If no prediction is used, that region of the picture is coded by the intra techniques.

In a closed group of pictures P and B pictures are predicted only from other pictures in the group of pictures; in an open group of pictures the prediction may be from outside the group of pictures.

(d) Display and coding order:

Since mpeg sometimes uses the information from future pictures in the sequence, the coding order; the order in which compressed pictures

are found in the bitstreams, is not the same as the display order, the order in which the pictures are presented to the viewer. The coding order is the order in which the pictures should be decoded by the decoder.

(e) Macroblock:

The basic building block of a mpeg picture is the macroblock. The macroblock consists of a 16 X 16 sample array of luminance (grayscale) samples together with one 8X 8 block of samples for each of two chrominance (colour) components. The 16 X16 array of luminance is actually composed of four 8 X 8 blocks of samples and these 8 X 8 blocks are the units of data that are fed to the compression models.

(f) **Slice:**

The mpeg picture is not simply an array of macroblocks, however it is composed of slices, where each slice is a contiguous sequence of macroblocks in a raster scan order, starting at a specific address or position in the picture specified in the slice header. This slice structure, among the other things, allows for great flexibility in signaling changes in some of the coding parameters. This is needed both to optimize quality for a given bitrate and to control that bitrate.

(g) The discrete cosine transform in mpeg:

At the heart of both inter and intra coding in mpeg is the discrete cosine transform (DCT). The DCT has certain properties that simplify coding models and make the coding efficient in terms of perceptual quality

measures. Basically, the DCT is a method of decomposing a block of data into a weighted sum of spatial frequencies. Each of these spatial frequency patterns has a corresponding coefficient, the amplitude needed to represent the contribution of that spatial frequency pattern in the block of data being analyzed. In other words, each spatial frequency pattern is multiplied by its coefficient and the resulting 64 8 X 8 amplitude arrays are summed, each pixel separately to reconstruct the 8 X 8 block.

If only the low frequency DCT coefficients are nonzero, the data in the block vary slowly with position .If high frequencies are present, the block intensity changes rapidly from pixel to pixel. The qualitative behavior can be seen where the 64 2-D patterns that make up the 8 X 8 DCT are seen to range from absolutely flat–the term that gives the average or DC value of the 8 X 8 block of pixels to a checkerboard with very large intensity changes from one pixel to next and at each axis of the plot the variation is one dimensional .

(h) Quantization:

When the DCT is computed for a block of pixels, it is desirable to represent the coefficients of high spatial frequencies with less precision. This is done by a process called quantization.

A DCT coefficient is quantized by dividing it by a non-zero positive integer called a quantization value and rounding the quotient, the quantized DCT coefficient to the nearest integer. The bigger the quantization value the less precise is the quantized coefficient. Lower precision coefficients can be transmitted to a decoder with a fewer bits. The use of

large quantization values for high spatial frequencies allows the encoder to selectively discard high frequency activity that the human eye cannot readily perceive. The DCT and the visually weighted quantization of the DCT are key parts of the mpeg coding system.

A lower resolution is used for the chrominance blocks because the human eye resolves high spatial frequencies in luminance better than in chrominance.

The DCT turns out to have several advantages. For intra coding the coefficients are independent of one another. This makes it possible to design a relatively simple algorithm called a coding model. However the coding performance is actually influenced much more profoundly by the visually weighted quantization.

In non-intra coding, coding the difference between the current picture and a picture already transmitted, the DCT does not greatly improve the decorrelation, since the differential signal obtained by subtracting the prediction from a similar picture is already fairly well decorrelated. However quantization is still a powerful compression for controlling bitrate, even if decorrelation is not improved very much by the DCT.

(i) Motion compensation:

If there is motion in the sequence, a better prediction is often obtained by coding differences relative to areas that are shifted with respect to the area

being coded, a process known as motion compensation. The process is determining the motion vectors is called motion estimation.

The motion vectors describing the direction and amount of motion of the macroblocks are transmitted to the decoder as a part of the bitstream. The decoder then knows which area of the reference picture was used for each prediction, and sums the decoded difference with this motion compensated prediction to obtain the output.

A key element in mpeg inter compression is motion compensation. In inter compression the pixels in a region of a reference picture are used to predict pixels in a region of the current picture. Differences between the reference picture and the current picture are then coded to whatever occuracy is affordable at the desired bitrate. If there is motion some parts of the image used for prediction may be shifted relative to the current image and motion compensation is used to minimize the effects of these shifts.

(j) Motion in picture sequences:

The pictures in moving picture sequences often have large regions in which very small changes occur from one picture to the next in the sequence .The picture to picture correlation in these regions is very high and any reasonable video compression scheme should take advantage of this. Obviously, there are usually some changes from picture to picture and these changes are caused by a number of mechanisms like noise, scene lighting and motion. If the only difference from one scene to the next is random noise, perhaps from video amplifiers, the most logical choice for prediction

of a given region would be region with the same spatial coordinates in the reference picture. Since we are dealing with lossy compression in which distortions too small to be visible can be ignored, small differences due to noise can be often ignored and the region can be copied from the reference picture without modification. Only a few bits are required to tell the decoder that there are no changes in this region.

Relative change from one picture to the next can be caused by camera motion such as panning or vibration and even by instabilities in video synchronization. Relative motion within a picture gives rise to many interesting technical problems. For example consider the relatively simple case of rigid body motion of an object in a picture. The motion of a rigid body has six degrees of freedom, three for spatial translations and three for rotational motions.

De-formable bodies are less easily analyzed; multiple objects, camera motion and scene lighting changes introduce further complications. In a video sequence the scene is projected through a lens onto a 2-D array of sensors and captured at discrete time intervals. Real motion becomes apparent motion in which the eye perceives motion as it views the sequence of still pictures .

In each still picture the motion can be described by a 2-D array of motion vectors that give displacements relative to a reference picture in the sequence. If good values of these displacements are available, pixels in the reference picture can be used to predict the current picture.

Since pictures are rectangular arrays of pixels it is convenient to describe motion vectors in terms of horizontal and vertical displacements. Hopefully the displacements are those that give the best match between a given region in the current picture and the corresponding displaced region in the reference picture. The main technical issues in using the motion vectors are the precision of the motion vectors, the size of the regions assigned to a single motion vector and the criteria to select the best motion vector value. The decoder simply decodes the motion vectors; the processing to determine the motion vector values is done entirely in the encoder.

The size of the region assigned to a single motion vector determines the number of motion vectors that are needed to describe the motion .In mpeg the size of the region is the macroblock and the same values for the motion displacements are used for every pel in the macroblock. An mpeg-1 macroblock has four luminance blocks and chrominance blocks; while the same motion displacement information is used for all the blocks, the actual displacement must be scaled to reflect

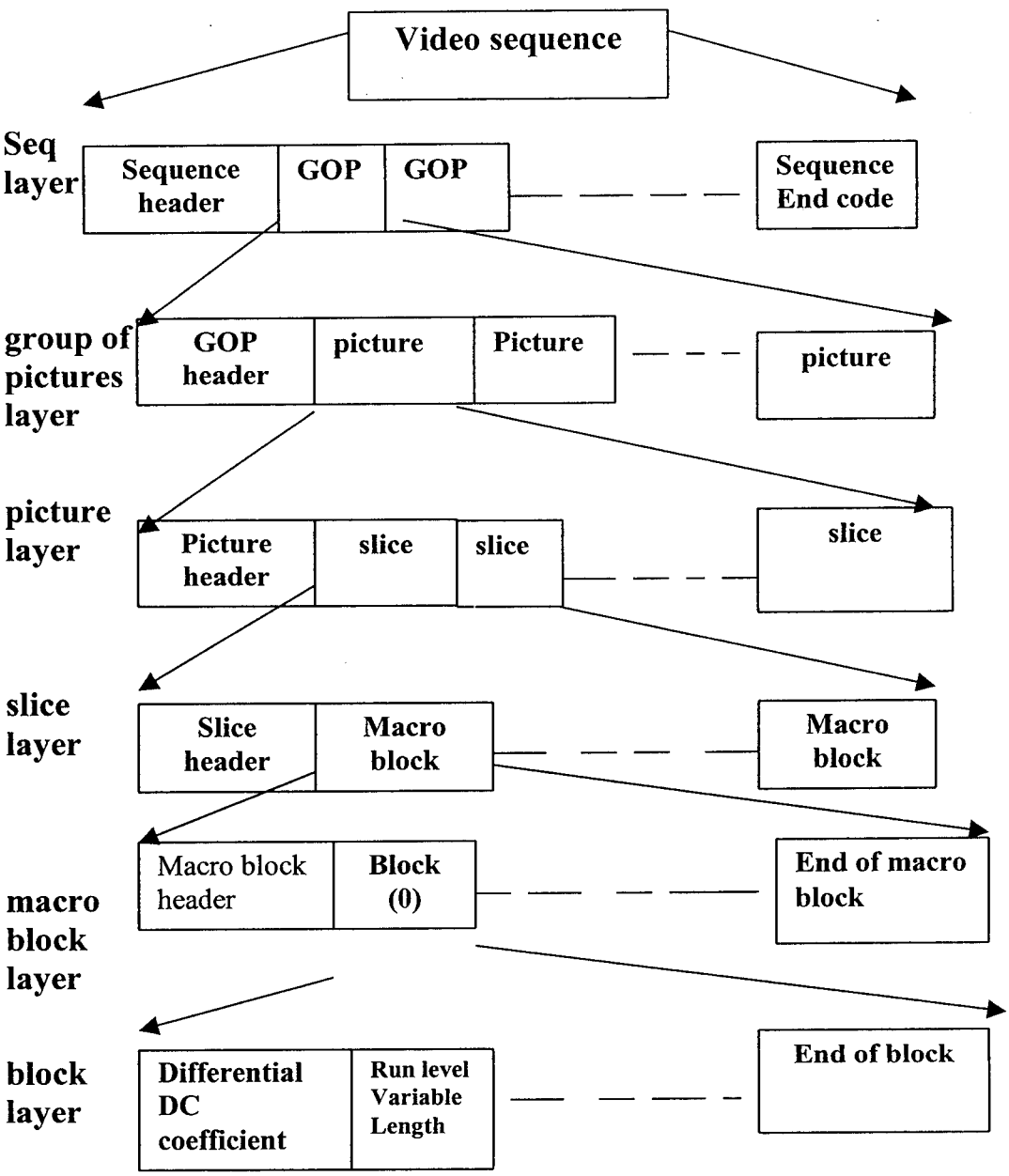
CHAPTER 3: VIDEO SYNTAX

(3.1) VIDEO SEQUENCE OVERVIEW

The mpeg video bitstream consists of six syntactical layers. The system layer provides a wrapper around the video sequence. The system layer performs the synchronization and multiplexing of the audio and video bitstreams into an integrated datastream.

A video sequence always starts with a sequence header. The sequence header is followed by atleast one or more groups of pictures (GOP) and ends with a sequence end code. Additional sequence headers may appear between any groups of pictures within the video sequence. The optional nature of the second sequence header and the extra GOP's is indicated. Most of the parameters in these optional additional sequence headers must remain unchanged from their values recorded in the first sequence header. The addition of extra sequence headers may be to assist in random access playback or video editing.

fig : The layers of a video stream



A group of pictures always start with a group of picture header and is followed by atleast one picture. Each picture in the group of pictures has a picture header followed by one or more slices .In turn each slice is made up of a slice header and one or more groups of DCT blocks called macroblocks. The first slice starts in the upper left corner of the picture and the last slice ends in the lower right corner.

The macroblocks is a group of six 8X8 DCT blocks, four blocks contain luminance samples and two contain chrominance samples. Each macroblock starts with a macroblock header containing information about which the DCT blocks are actually coded.

DCT blocks are coded as intra or non-intra. If an intra block is coded, the difference between the DC coefficient and the prediction is coded first .The AC coefficients are then coded using the variable length codes (VLC) for the run level pairs until an end of block terminates the block. If a non-intra bock is coded, DC and AC coefficients are coded together using the same run level VLC codes.

The four picture types in mpeg are: I pictures (intra coded),P pictures (non-intra or predicitive coded),B pictures (bi-directionally predictive coded) and D pictures (DC coded pictures). I pictures are coded independantly of other pictures. P pictures are coded with respect to a preceding picture and B pictures are coded with respect to both preceding and succeeding pictures. D pictures contain only DC coefficient information and therefore an end of macroblock data element is used to complete each macroblock.

Start codes:

The sequence header, group of pictures header and slice header all start with a unique byte aligned 32 bit patterns called start codes. Other start codes are defined for system use, user data and error tagging. The start codes all have a 3byte prefix of 23 zero bits followed by a 1 bit; the final byte then identifies the particular start code.

The byte alignment of start codes is achieved by stuffing as many zero as needed to get to byte alignment. Zero bytes may then be stuffed if desired. Zero byte stuffing before the start codes is one of the mechanisms used to avoid decoder buffer overflow.

CHAPTER 4: CODING SCHEME

(4.1) INTRODUCTION TO IMAGE TRANSFORMS

Let f(x) be a continous function of a real variable x. The Fourier transform of f(x), denoted by F(u) is defined by the equation,

$$F(u) = \int_{-\infty}^{\infty} f(x) .\exp (-j2\Pi ux) .dx \quad \rightarrow \quad 1$$

where $j = \sqrt{-1}$.

Given F(u), f(x) can be obtained by using the inverse fourier transfom ,

$$F^{-1} (F(u)) = f(x) = \int_{-\infty}^{\infty} f(u) .\exp (j2\Pi ux) .du \quad \rightarrow \quad 2$$

Equations 1 and 2 called the fourier transform pair, exist if f(x) is continous and integrable and f(u) is integrable. These conditions are always satisfied in practice. Fourier transform of a real function is generally complex.

$$\text{ie. } F(u) = R(u) + j\, I(u) \quad \rightarrow \quad 3$$

where R(u) and I (u) are the real and imaginary components. It is often convenient to express equation 3 in the exponential form ie ,

$$F(u) = |F(u)|\, e^{j\Phi(u)} \qquad \rightarrow \qquad 4$$

where

$$|F(u)| = \sqrt{\left(R^2(u) + I^2(u) \right)}$$

and $\Phi(u) = \tan^{-1}\left(I(u)/R(u) \right)$.

The magnitude function $|F(u)|$ is called the Fourier spectrum of f(x) and $\Phi(u)$ is its phase angle. The square of the spectrum, $P(u) = |F(u)|^2 = \left(R^2(u) + I^2(u) \right)$ is commonly referred to as the power spectrum of f(x). The variable u appearing in the Fourier transform is often called the frequency variable. This name arises from the expression of the exponential term, $e^{-j2\prod ux}$ using the Euler's formula in the form

$$e^{-j2\prod ux} = \cos 2\prod ux - j \sin 2\prod ux \qquad \rightarrow \qquad 5$$

Interpreting the integral n the equation 1 is a limit summation of discrete terms makes evident that F(u) is composed of an infinite sum of sine and cosine terms and that each value of u determines frequency of the corresponding sine cosine pair.

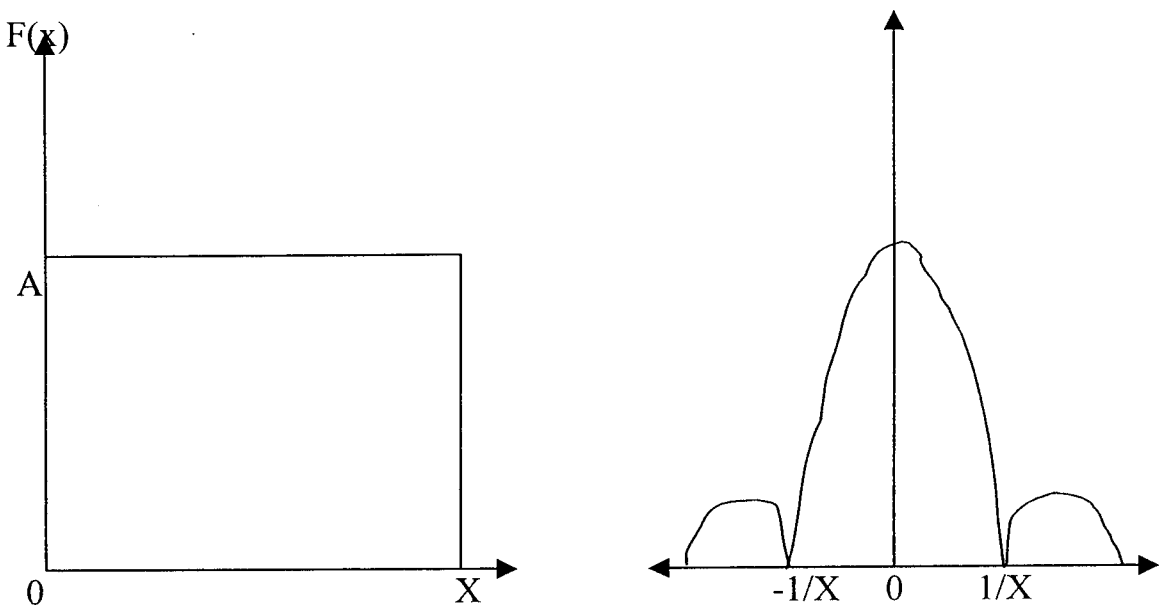


fig : A simple function and its Fourier spectrum

Discrete Cosine Transform :

Suppose that a continous function f(x) is discretized into a sequence { $f(x_0)$, $f(x_0 + \Delta x)$, , $f(x_0 + 2\Delta x)$, ..., $f(x_0 + (N-1)\Delta x)$} By taking N samples Δx units apart. To do so f(x) is defined as $F(x) = f(x_0 + \Delta x)$, where x now assumes the discrete values 0,1,2..N-1. In other words the sequence denotes any N uniformly spaced samples from a corresponding continous function.

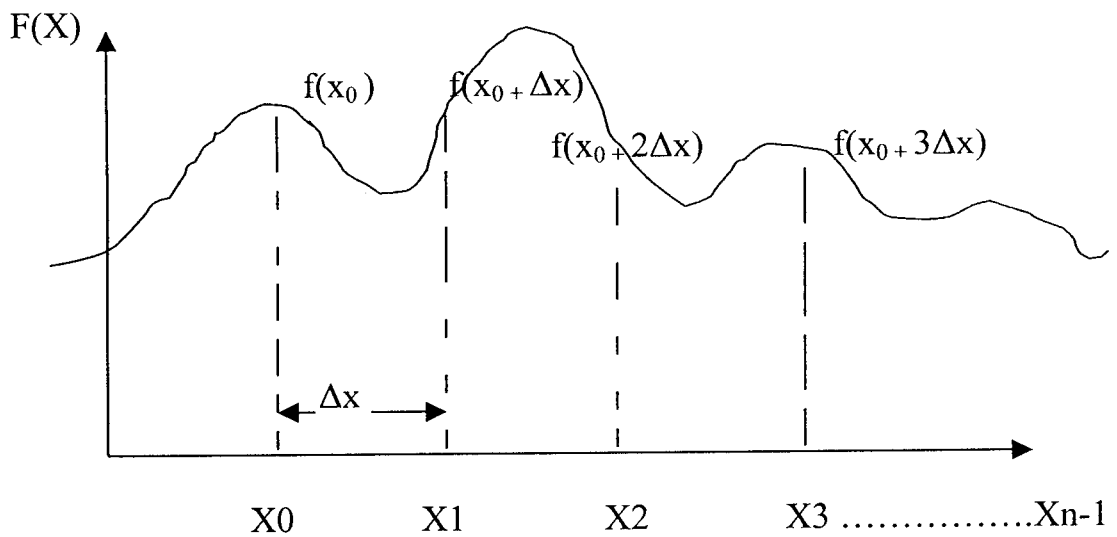


Fig : Sampling a continous function

The discrete Fourier transform that applies to the sampled fundtion is given by

$$F(u) = 1/N \sum_{X=0}^{N-1} f(x) .e^{(-j2\Pi ux/N)} \quad \text{for } u=0,1,2,\ldots N-1. \rightarrow \quad 6$$

and

$$F(x) = \sum_{U=0}^{N-1} f(u) .e^{(j2\Pi ux/N)} \quad \text{for } x=0,1,2,\ldots N-1. \rightarrow \quad 7$$

The values u=0,1,2....N in the DFT correspond to samples of the continous transform at values 0, Δu, $2\Delta u$ ... N-1 Δu.

The fast fourier transform :

The number of complex multiplications and additions required to implement equation 6 is proportional to N^2 ie for each of the N values of u, expansion of the summation requires N complex multiplication of f(x) by $e^{(j2\prod ux/N)}$ the summation requires N complex multiplications of f(x) by $e^{(-j2\prod ux/N)}$ and N-1 additions of the results.

Proper decomposition of the equation 6 can make the number of multiplication and addition operations proportional to N $\log_2$ N. The decomposition procedure is called the FFT algorithm. The reduction is proportionally from N^2 to N $\log_2$ N operations represents a significant saving in the compuational effort.

FFT algorithm :

This FFT algorithm is based on the successive doubling method.for convenience the FFT equation is expressed in the form

N-1

$$F(u) = 1/N \sum_{X=0} f(x)\, W_n^{ux} \quad \rightarrow \quad 8$$

Where $W_N = e^{(-j2\prod ux/N)}$

and N is assumed to be of the form $N=2^n$ $\rightarrow$ 9

where n is a positive integer.

Hence N can be expresssed as $N=2M$ $\rightarrow$ 10

where M is also a positive integer.

Substitiution of equation 8 in equation 10 yields

$$F(u) = 1/2M \sum_{X=0}^{2M-1} f(x)\, W_{2M}^{ux} \quad \rightarrow \quad 11$$

Defining $$F_{even}(u) = 1/M \sum_{X=0}^{M-1} f(2x)\, W_{2M}^{ux} \quad \rightarrow \quad 12$$

$$F_{odd}(u) = 1/M \sum_{x=0}^{M-1} f(2x)\, W_{2M}^{ux} \quad \rightarrow \quad 13$$

for u=0,1,2,...M-1 results in the reduction of the equation 11 to

$$F(u+M) = 1/2\ [f_{even}(u) - F_{odd}(u)\, W_{2M}^{u} \quad \rightarrow \quad 14$$

An N point transform can be compted by dividing the original expression into two parts .

A comparison of direct FT and FFT for different values of N :

N	N^2 direct FT	FFT $N \log_2 N$	Computational advantage $N/ \log_2 N$
2	4	2	2.00
4	16	8	2.00
6	64	24	2.67
8	256	64	4.00
16	1024	160	6.40

A video picture normally has relatively complex variations in signal amplitude as a function of distance across the screen. It is however possible to express this complex variation as a sum of simple oscillatory sine or cosine waveforms that have a general behavior. The sine or cosine waveforms must have the right spatial frequencies and amplitudes in order for the sum to exactly match the signal variations.

When the waveforms are cosine functions, the summation is a cosine transform and the waveforms that make up these transforms are called the basis functions.more formally the expression of a set of eight samples f(x), as a sum of eight cosine basis functions is

$$F(x) = \sum_{\mu=0}^{7} c(\mu)/2 \; F(\mu) \cos [(2x+1) \mu \prod + 16] \longrightarrow \quad 15$$

and $c(\mu)$ is given by $c(\mu) = 1/\sqrt{2}$ if $\mu = 0$ and $c(\mu) = 1$ if $\mu > 0$.

X is the displacement along the row of samples.

μ is the index determining the spatial frequency of the cosine transform.

$F(\mu)$ is the DCT coefficient for that spatial frequency.

The cosine basis functions always finish the interval at a full or half-cycle as they satisfy the condition of orthogonality. The additive constant in (2x+1) simply shifts the sample points so that they are symmetric around the centre of the N point interval.

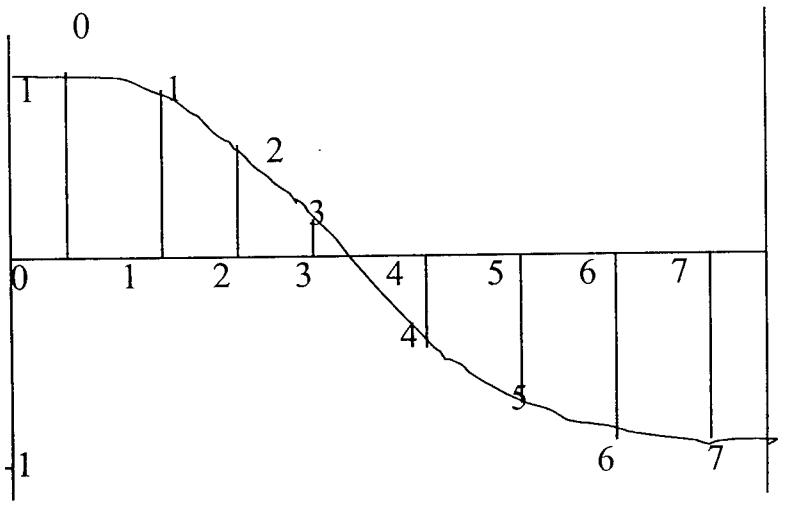


fig : Sampled cosine waveform

Transform selection :

Transform coding systems based on DCT, DFT and many other transforms have been constructed and studied. The choice of a particular transform in a given application depends on the amount of reconstruction error that can be tolerated and the computational resources available.

DFT holds the edge over the other transforms as it provides a good compromise between the information packing ability and the computational complexity. A significant factor affecting the transform coding error and the complexity in computation is the subimage size. In most applications images are divided so that the redundancy between adjacent subimages is reduced to some acceptable level so that n is an integer power of 2 where n is the

subimage dimension. Another key advantage of the DCT is that the boundaries between the subimages are visible unlike the DFT where it is discontinous.

(4.2) PROCESSING STEPS FOR THE DCT BASED CODING

Each compression scheme has its strengths and weaknesses because the codecs you choose will determine how good the images will look and how smooth the images will flow. Once a codec saves the non-redundant data, it statistically processes the arrangements of the pixels in the image. The statistical arrangements are then coded and the codec then uses a variety of compression methods.

In the case of MPEG, discrete cosine transform (DCT)is used. A frame is divided into blocks, usually 8 X 8 pixels each and transform mechanisms work on each of the individual blocks. DCT converts pixel intensities into a frequency based equivalent .The result is a transform yield which is a series of numbers that represent every fine detail in a pixel block

Compression then comes by eliminating the representative numbers after a certain point in the series .The consequence is compressed video with a loss of fine detail, but the level of detail is not detectable. And a major fact is that MPEG can attain compression ratios of upto 180to1.

8X8 FDCT and IDCT:

At the input to the encoder, source image samples are grouped into 8X8blocks and input to the FDCT [Forward Discrete cosine Transform]. At the output of the decoder IDCT [Inverse Discrete cosine Transform] outputs the 8X8 sample blocks to form the reconstructed image .The following are the mathematical definitions of the FDCT and the IDCT respectively.

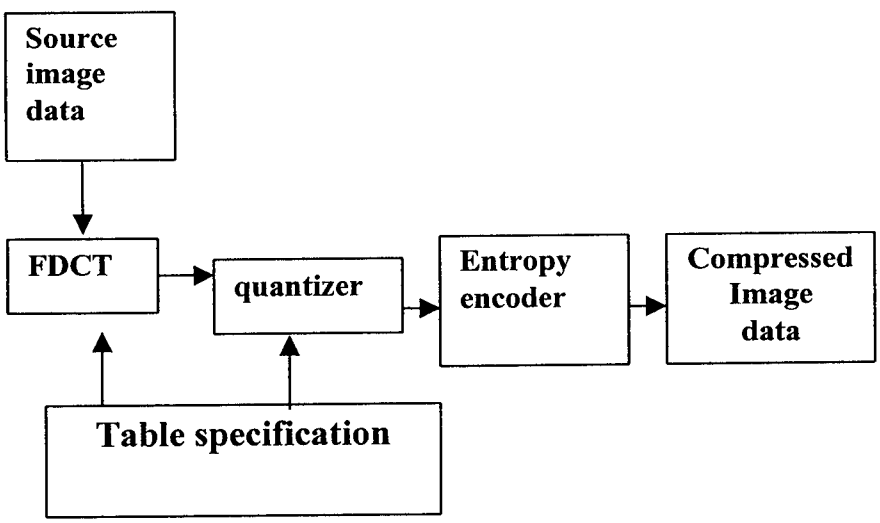


Fig : DCT based encoder

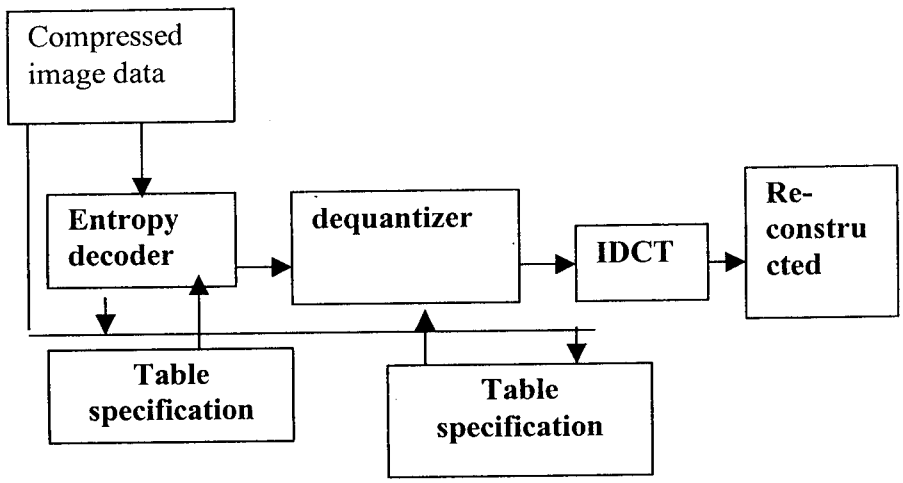


Fig : DCT based decoder

The following equations are the idealized mathematical definitions of the 8X8 FDCT and IDCT :

$$F(u,v) = (1/4)c(u)c(v)\left[\sum_{X=0}^{7}\sum_{y=0}^{7} f(x,y)*\cos[((2x+1)u\pi/16]\cos[((2y+1)v\pi/16)]\right] \rightarrow 16$$

$$F(u,v) = (1/4)\left[\sum_{X=0}^{7}\sum_{y=0}^{7} c(u)c(v)f(x,y)\cos[((2x+1)u\pi/16]\cos[((2y+1)v\pi 16)]\right] \rightarrow 17$$

Where $c(u), c(v) = 1/\sqrt{2}$ **for** $u,v = 0$; **1 otherwise .**

The DCT is related to the discrete Fourier transform .The FDCT can be viewed as a harmonic analyzer and the IDCT as a harmonic synthesizer. Each 8X8 block of source image samples is effectively a 64 point discrete signal which is a function of two spatial functions x and y. The discrete Fourier Transform provides the decomposition of the input into N periodic components. The DFT of a discrete time signal x(n) is defined as

$$X(k) = (1/N) \sum_{n=0}^{N-1} x(n) W_N^{nk} \quad ; k=0,1,\ldots N-1.$$

where $W_N = e^{(-2j \pi nk)/N.}$

The set of frequency samples which defines the spectrum x(k), are given on a frequency axis whose discrete frequency locations are given by $f_k = k(F/N)$ k=0,1,2...N-1. The frequency resolution of the DFT is equal to the frequency increment F/N and the frequency response is determined by applying a component exponential signal and evaluating the DFT bin output response as the frequency is varied.

The FDCT takes such a signal as input and decomposes it into 64 orthogonal basis signals. Each contains one of the 64 unique 2D spatial frequencies which comprise the input signal spectrum. The output of the

FDCT is the set of 64 basis signal amplitudes or DCT coefficients whose values are uniquely determined by the particular 64 point input signal.

The DCT coefficient values can thus be regarded as the relative amounts of the 2D spatial frequencies contained in the 64 point input signal. The coefficient with 0 frequency is called the DC coefficient and the remaining 63 coefficients are called the AC coefficients.

As the sample values typically vary slowly from point to point across an image the FDCT processing step lays the foundation for achieving data compression by concentrating most of the signal in lower spatial frequencies. For a typical 8X8 block from a typical source image most of the spatial frequencies have 0 or near 0 amplitude and need not be encoded.

At the decoder the IDCT reverses this processing step. It takes the 64 DCT coefficients and reconstructs a 64point output signal by summing the basis signals. Mathematically DCT is one to one mapping of 64 point vectors between the image and the frequency domains. If the FDCT and IDCT could be computed with perfect occuracy and if the DCT coefficients were not quantized the original 64 point signal could be exactly recovered .In principle the DCT introduces no loss to the source image samples; it merely transforms them into a domain in which they can be more efficiently encoded. A fundamental property is that FDCT and IDCT equations contain transcendental functions and no physical implementation can compute them with perfect occuracy.

Quantization:

After output from the FDCT each of the 64 DCT coefficients is uniformly quantized in conjunction with a 64 element quantization table which must be specified as an user input to the encoder within the range of 1 – 255. The purpose of this process is to achieve further compression by representing the DCT coefficients with no greater precision that is necessary to achieve the desired image quality.

Stated in other words the goal of this processing step is to discard information which is not visually significant. Quantization is a many to one mapping and therefore is fundamentally lossy. It is the principle source of losiness in a DCT based system.

It is defined as the division of each DCT coefficient by its corresponding quantizer step size followed by rounding to the nearest integer. The equation is

$$\mathbf{Fq(u,v) = integer\ round\ [\ F(u,v) / Q\ (u,v)\ ]}$$

Dequantization is the reverse function which is simply removing the normalisation by multiplying by the stepsize which returns the result to the IDCT.

$$\mathbf{Fq'(u,v) = Fq(u,v) * Q\ (u,v).}$$

DC coding and zig zag sequence :

After quantisation the DC coefficient is treated separately from the rest as it is a measure of the average value of the 64 image samples and they contain a significant fraction of the total image energy.

Finally all the quantized coefficients are ordered into the zigzag sequence to facilitate entropy coding by placing the low frequency coefficients,before high frequency coefficients. Entropy coding achieves compression by encoding the quantized DCT coefficients more compactly based on their statistical characteristics.

CHAPTER 5: IMPLEMENTATION

Video sequence layer:

The syntax for the video sequence function is given below.

```
/*   pseudocode for  video sequence function   */
   video_sequence(){
            next_start_code ();  /* find next start code */
        do {sequence_header();        / *sequence header  */
             do{ group_of_pictures ();
               } while(nextbits(32)==group_start_code);
       } while (nextbits(32)== sequence_start_code);
                    sequence_end_code(32);
        }  /* end video sequence function */
```

The first procedure in the above function locates the start code for the video sequence. The bitstream is defined to be a video sequence and the start code of the video sequence is the sequence header code. A sequence header is always followed by at least one group of pictures.

More than one sequence header can occur in a video sequence. A group of pictures must follow a sequence header and the final step of the sequence_header (). Process positions the bitstream pointer at the group of the picture start code. After each group of pictures, however if the next start code is for a group of pictures, processing of group of pictures continues.

```
/*  pseudocode for  sequence header function */

    sequence_header (){
sequence_header _ code(32);
horizontal_size(12);
vertical_size(12);
pel_aspect_ratio(4);
picture_rate(4);
bit_rate(18);
marker bit(1);
 buffersize (10);
 load intra quantizer matrix(1); /* load for intar quantizer */
 if  (load intra quantizer matrix)
      intra quantizer matrix [0..63];
load nonintra quantizer matrix(1);/* load for non intra */
 if(load nonintra quantizer matrix)
      nonintra quantizer matrix [0..63];
 next start code();
  if(nextbits(32)==extension start code)
    {  extension start code (32);
  while (nextbits (24)!= start code prefix )
    {
      sequence extension data(8); /* byte of data */
     }
    next start code();
      }
    If ( nextbits(32)== data start code)
```

```
{ data start code (32);
 while (nextbits(24)!= start code prefix)
{  data(8);
}
   next start code ();
}
}     /*  end of      sequence header function  */
```

Group of pictures layer:

The Group of pictures layer starts with nine required data elements. These may be followed by optional extension and user data before the picture layer is coded. In the group of pictures function a 25 bit time code follows the 32 bit group start code. This time code contains six data elements.

```
/*    pseudocode for the group of pictures     function  */

group of pictures (){
      group start code(32);
      time code(25);
      next start code ();
      If ( nextbits(32)==extension start code){
       Extension  start code(32);
       While (nextbits != start code prefix)
        { group extension data (8);  /* byte of data */
          }
```

```
next start code();
 }
 if(nextbits (32) ==data start code){
   start data code (32);
     while(nextbits(24)!= start code prefix ){
     user data(8);
      }
  next start code ();
    }
  do{
       picture (); /* decode picture */
   }while ( nextbits(32)==picture start code )
    }

/* end of the group of pictures function */
```

picture layer :

The picture layer is where the picture coding type is signalled and the forward and backward motion vector elements are established that define the scale and precision of the motion vectors.

```
/* pseudo code for picture function */

  picture (){
    picture start code(32);
```

```
        temporal reference(10);
         picture coding type(3);
        if(picture coding type==2)|| (3)){
         /*  if p or b picture need forward motion vector */
         forward f code(3); /*  range of the motion vector */
        }
if(picture coding type== (3)){/* if  b picture need backward motion vector */
         backward f code(3);
          }
/* include extra information if any  */
         next start code ();
          }
            do{
        slice();
      }  while nextbits(32)== slice start  code)
   }
/*  end of picture function   */
```

Slice layer:

```
          /*  pseudo code for the slice function */
         slice(){
        slice start code(32);
        while(nextbits(2)=='1'){
        /*  extra slice information */
        }
      extra bit slice (1);
```

```
do{
  macroblock (); /* process a macroblock */
} while (nextbits (23)!=0)
next start code();
  }
```

Macroblock layer :

The macro block header provides information about the position of the macroblock relative to the position of the macroblock just coded. It also codes the motion vectors for the macroblock and identifies the blocks in the macroblocks that are coded.

```
/*   pseudocode for the macroblock function */

    macroblock(){
    if(motion forward ){
     motion horizontal forward code ();
      if(forward f !=1)&&horizontal forward code !=0)
   /* variable length coding for macroblock address */
     }
if(motion backward ){
     motion horizontal backward code ();
      if(backward f !=1)&&horizontal backward code !=0)
   /* variable length coding for macroblock address */
     }
```

```
    if (picture coding type ==4 ) /*  d picture */
}     /*     end of macroblock();  */
```

Block layer:

This is the lowest layer of the video sequence and it is in this layer that the actual 8X8 blocks are coded. Note that the coding depends on whether the block is a chrominance or a luminance block and whether the block is intra or non-intra.

```
/*     the pseudocode for the block function  */

      block (){
       if (ith block coded ) {
        if(intra coded macroblock){
         if ( luminance  block ){
           variable length code for luminance size)
         }   /* end if luminance block  */
      else
       {       /*  vlc for cr or cb size */
        } /* end else  chrominance block */
        } /* end if intra coded */
       else
        {  dct coeff first ();
          }  /* end not intra coded macro block */
      if(picture coding type)!= "d type ");
        {
```

```
        while ! end of block )
            dct coeff (3-28);
        end of block();

}  /* end if not d picture */

}  /* end if block I coded */

} /* end  block function */
```

These pseudo codes gives the outline of the basic set of operations done on a digital video stream .The actual coding is done in c, which has a syntax more or less similar to the pseudo code syntax. The decoder is made to run in the GUI environment of visual C++.

CHAPTER 6: EXPERIMENTAL RESULTS.

The mpeg standard is primarily intended to process video at what is known as source input format (SIF). That is 352 X 240 at 30 frames/second. The decoder converts one or more mpeg–1 and mpeg–2 video bitstreams and converts them to uncompressed video. Since mpeg-2 is forward compatible with mpeg-1 the decoder is capable of decoding mpeg-1 sequences.

TEST VIDEO CLIPS:

(1) Space shuttle launch

Resolution : 352 X 240
Size : 632 KB
Frame rate : 30 frames /s
Length : 10 seconds
Pixel depth : 16

(2) A tour into a house

Resolution : 352 X 240
Size :1.05 MB
Frame rate : 30 frames/s
Length : 22 seconds
Pixel depth : 16

The decoder is capable of decoding all the mpeg-1 bitstreams except the D –picture sequences. It is robust against the stream syntax errors.The performance of this decoder can be easily understood when making a comparitive study with a standard mpeg decoder like the Windows media player.

The above test video clips where chosen to test the decoder and they are among the testclips provided by official source of the MPEG organisation. There is no special criteria to chose the test video clips, any video clip with the resolution equal to the SIF specification can be chosen.

The first clip namely the space shuttle launch which runs for a period of 10 seconds in the media player took 5 more seconds on the decoder .In the same way the house tour clip also took about 8 seconds more than the normal 22 seconds in the media player. Moreover the quality of the image was also comparitively poor. The control of various parameters of the image was not easy. Thus the loophole may turn out to be the speed mismatch which results in the buffer extremities. The parameters in the display function need to be varied in order to improve the image quality.

The quality of decompressed signal is measured by three elements. These elements are the number of displayable colours, The number of pixels per frame ie resolution and the number of frames per seconds. Each of these elements can be traded off for another and all of them can be traded for

better transmission rates. However it is impossible to combine all of them for the quality of the decompressed video.

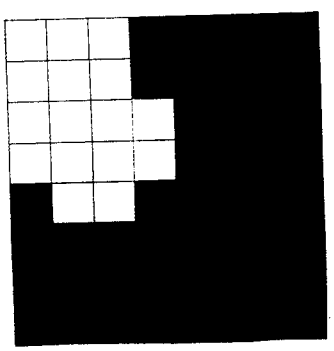
The decoder's ability to provide better quality image can be done by improving the display parameters. Quantization which is the fundamental source of stripping off the redundant image can be treated better to obtain the better image. The quantization value may be the target to reduce the amount of data to be chopped off from the source data thereby increasing the quality of the image, with a tradeoff with the compression ratio. The efficiency of the decoder turns out to be one third lesser than the standard decoder. The efficiency can be improved considerably in the future by acting upon the various parameters and converging them to an optimal value.

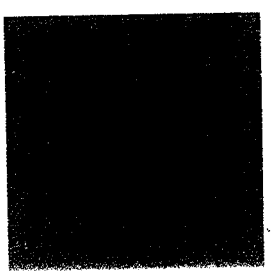
The following figure gives the results in a DCT based lossy compression method, where the decompressed image corresponding to the number of DCT coefficients chosen is given.

Original image (saturn)

16 DCT coefficients selected :

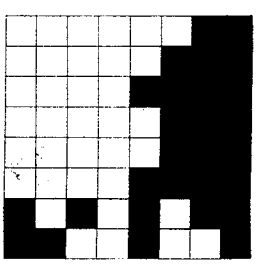
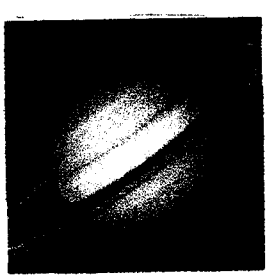
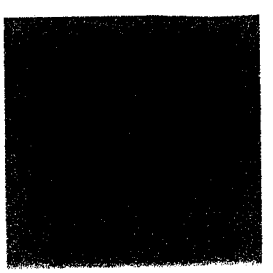
DCT coefficients reconstructed image error image





36 DCT coefficients selected :

DCT coefficients reconstructed image error image

CHAPTER 7: CONCLUSION.

The main purpose of this decoder is to emphasize on the correct implementation of the standard and to demonstrate a sample implementation of the decoder. There is a very high scope of improving the decoder as it is a very simple structured decoder. There is a wide possibility of adding many more features thereby improving the decoder to perform at a much more efficient manner. The decoder is not optimized for speed, although some parts like the IDCT are based on relatively fast algorithms.

Since this project is being focussed on the decoding end there may be mismatch in the buffering requirements which is normally described taking into account both the transmission as well as the reception of the video .So the mismatch may result in either buffer overflow or the underflow. All the drawbacks follow the standard difficulties faced by a decoder of any form to decode the mpeg video stream which is in its early stages. Even though this decoder is capable of supporting the high bitrates of mpeg -2 it is unable to perform at the higher resolution options.

CHAPTER 8: BIBLIOGRAPHY.

1. D. J. DeFatta, J. G. Lucas, W. S. Hodgkiss, " Digital Signal Processing ", Wiley, 1995.

2. R. C. Gonzalez, R. E. Woods, " Digital Image Processing ", Addison-Wesley, 2000.

3. J. L. Mitchell, W. B. Pennebaker, C. E. Fogg, D. J. LeGall, " MPEG video compression standard ", Chapman & Hall, 1996.

4. E. Balagurusamy, " Programming in ANSI C ", Tata McGraw Hill, 2000.

5. Gregory. K. Wallace, " The JPEG still picture compression standard ", communications of the ACM, April 1991, vol 31. No.4.

6. Chad Hogg, " Introductions to MPEG ", MPEG Software Simulation Group,April 1992.

7. Clay. M. Thomson, " Image processing user's guide ", The Math works Inc., Jan 1995.

CHAPTER 9: APPENDIX.

Differences between mpeg-1 and mpeg-2 video:

(Extract from . J. L. Mitchell, W. B. Pennebaker, C. E. Fogg, D. J. LeGall, " MPEG video compression standard ", Chapman & Hall, 1996.)

All currently defined mpeg-2 profiles and levels require decoders to handle mpeg-1 constrained parameters bitstreams within their level capabilities. For most data elements mpeg-2 is a direct subset of mpeg-1. However, some mpeg-1 data elements do not have a direct equivalent in mpeg-1.

The IDCT (Inverse Discrete Cosine Transform) mismatch is handled differently in mpeg-1 and mpeg-2. mpeg-1 makes each non-zero co-efficient odd after inverse quantization. mpeg-2 adjusts only one of the co-efficient (the highest vertical and horizontal frequency)if the sum of the inverse quantized co-efficients is even.

Mpeg-1 pictures have the chrominance samples horizontally and vertically positioned in the center of a group of four luminance samples. Mpeg-2 co-allocates the chrominance samples on the luminance samples.

Mpeg-1 has always a fixed picture rate.mpeg-2 has a low delay mode in which a big picture can take more than the normal single picture time inferred from the picture rate.

Mpeg-1 codes the four bit pixel aspect ratio (this parameter gives the information about the pixel shape) in the sequence header. Mpeg-2 specifies the display aspect ratio in the data element aspect ratio information. The frame size and display size are used to derive the pel aspect ratio.

Mpeg-1 allows D-pictures whereas mpeg-2 does not allow those pictures except within mpeg-1 constrained parameter bitstreams. Mpeg-2 also restricts the horizontal size to 720. This is the largest possible horizontal size an mpeg-2 picture can have.