CHAPTER 1

INTRODUCTION

1.1 GENERAL

Nowadays applications are demanding ever-increasing role for multimedia and surveillance contents. An enormous amount of information is stored, processed, and transmitted digitally. Users seek to enhance their personal expressions for communication through audio, images, video, and graphics information. Digital television (TV), digital versatile disk (DVD) video, high definition television (HDTV), internet video streaming, video conferencing and mobile technology have expanded the boundaries of communication systems to include a rich visual dimension. The video compression platform that support these applications must be able to manipulate data efficiently and transparently (Iain Richardson 2003) (Shen 1997). The term video compression refers to the process of reducing the amount of data required to represent a given quantity of information. The benefits of video compression are to use digital video in transmission and storage environment resourcefully. A DVD can only store a few seconds of raw video at television quality resolution and frame rate. So, DVD video storage would not be practical without compression (Ghanbari 2003) (Jack 1996).

Video compression methods can be divided into two categories. Lossless compression algorithms are characterized by an invertible encoding process that allows the compressed data to be reconstructed exactly into its
original form after decoding. In a lossless compression system, statistical redundancy is removed so that the original signal can be perfectly reconstructed. Lossy compression algorithms, on the other hand, do not guarantee exact reconstruction after decoding; the reconstructed video sequence will not be identical to original video sequence. Most practical video compression techniques are based on lossy compression, in which greater compression is achieved (Freg and Linzer 1990) (Chen et al 1977). Video compression algorithms operate by removing redundancy in the temporal, spatial and frequency domains. By removing different types of redundancy, it is possible to compress the data significantly at the expense of a certain amount of information loss (distortion). Further compression can be achieved by encoding the processed data using an entropy coding scheme such as Huffman coding or Arithmetic coding. This has led to the development of a number of key International Standards for image and video compression, including the JPEG, MPEG and H.26× series (Rao and Yip 1990) (Ahmed et al 1974).

1.2 VIDEO CODEC

The encoder/decoder pair is often described as a CODEC (enCOder/ DECoder) (Iain Richardson 2003) (Jain 1989). A video CODEC encodes a source image or video sequence into a compressed form and decodes this to produce a copy or approximation of the source sequence as shown in Figure 1.1.
The CODEC represents the original video sequence by a model (an efficient coded representation that can be used to reconstruct an approximation of the video data). Ideally, the model should represent the sequence using as few bits as possible and with as high a fidelity as possible. These two goals (compression efficiency and high quality) are usually conflicting, because a lower compressed bit rate typically produces reduced image quality at the decoder. A video encoder consists of three main functional units: a temporal model, a spatial model and an entropy encoder (Delp et al 1999) (Shen and Delp 1999) (Nasrabadi and King 1988).

Temporal model analyzes series of consecutive frames from a video sequence. The content modification of the image in a really lightly way has been recognized from one frame to another. The information's difference from one frame to the other is normally reduced to some small changes due to motion or illumination alterations in the content. This temporal redundancy in the information will allow a reduction of the data. The concept of using a reference frame to code an upcoming one and thus minimize the amount of data is the key of the inter frame predictive coding (Mitchell 2002) (Barjatya 2004). The process can be described as a two step process: first a prediction of a frame is made in base of a reference frame (in most of the cases a single temporally previous frame). Then, the difference between the actual current frame and its prediction is computed and coded. The residual frame obtained from the temporal model will be processed to reduce spatial redundancy. This step involves a transform coding to convert the image into another domain.

The transform should fulfill the following criteria:

- The data in the transform domain should be decorrelated and compact
- The transform should be reversible
- The transform should be computationally tractable.
Two most popular transforms in video Compression standards are block based discrete cosine transform (DCT) and image-based discrete wavelet transform (DWT) (Xiong et al 1999). However, the transformation makes the energy to be distributed in a more easily reducible way. The maximum of energy is concentrated at the lowest frequency components and thus the majority of coefficients have little energy. Thus, the insignificant values will be removed by applying quantization. The principle of quantization is to divide the values by a nonzero positive integer (quantization value) and round the quotient to the nearest integer. The quantized transformed coefficients are further coded to reduce the bit rate. Entropy encoding encodes the data according to the information content. Frequently occurring messages, carrying more information, are coded with longer words, while less common messages, carrying less information, are coded with shorter words (Witten et al 1987).

1.3 VIDEO COMPRESSION STANDARDS

Several international organizations have proposed standards that have found limited support from vendors. Recommendations H.120 and H.130 for video conferencing at data rates close to 2 Mbps, were proposed by the international telephone and telegraph consultative committee (CCITT) in the early 1980's. These standards were not widely accepted by manufacturers due to limited signal quality. In the late 1980s, the ITU-T (International Telecommunications Union, Telecommunication Standardization Sector) developed Recommendation H.320 (Girod et al 1997) (Chiariglione 1997) for the transmission of non-telephonic signals for video telephone systems using ISDN circuits.
1.3.1 H.261

H.261 is an ITU-T video coding standard, ratified in November 1988. Originally designed for transmission over ISDN lines on which data rates are multiples of 64 kbps. It is one member of the H.26x family of video coding standards in the domain of the ITU-T Video Coding Experts Group (VCEG). The coding algorithm was designed to be able to operate at video bit rates between 40 kbps and 2 Mbps. The standard supports two video frame sizes: CIF (352x288 luma with 176x144 chroma) and QCIF (176x144 with 88x72 chroma) using a 4:2:0 sampling scheme (Girod 1993) (Pennebaker 1996). The basic processing unit of the design is called a macroblock, and H.261 was the first standard in which the macroblock concept appeared. Each macroblock consists of a 16x16 array of luma samples and two corresponding 8x8 arrays of chroma samples and YCbCr color space (ITU-T H.261 1993).

1.3.2 H.263

The ITU Recommendation H.324, endorsed in 1995, establishes standards for low data rate multimedia communications over ordinary telephone lines, known as General Switched Telephone Network (GSTN). H.263 provides better picture quality than H.261 at comparable complexity (Bhaskaran and Constantinides 1997). No constraints on the target data rate are part of the recommendation; the equipment or the network imposes the restrictions. The color space used in H.263 is YCbCr with 4:2:0 chrominance subsampling. To achieve significant compression efficiency, temporal redundancy between pictures needs to be exploited. Therefore, adjacent pictures have high temporal correlation, especially in simple scenes typical of video conferencing applications. Intraframe and interframe coding are the basis of H.263, MPEG-1 and MPEG-2 standards. Interframe coding works well with translational motion of rigid objects. Spatial correlation between adjacent pixels is exploited using intraframe coding (ITU-T H.324 1995).
1.3.3 H.264

H.264 adds optional features to H.263 in order to broaden its range of applications, and to improve its compression performance (Mitchell et al 1996) (Girod et al 1997) (ITU-T H.263 1996). H.264 is compatible with H.263, which has functionalities such as:

- Custom picture formats: The source format (picture size) and the frame rate are negotiated during setup.

- Addition of scalability pictures:
  - Bidirectionally predictive-coded (B) pictures, as part of the temporal scalability mode. These pictures can be discarded without impacting the quality of future pictures, because they are not used to predict other pictures.
  - SNR scalable pictures: A difference frame, obtained by subtracting the original and the decoded frames, can also be encoded with finer quantization and sent to the decoder, producing an enhancement to the decoded picture. This effectively increases the subjective picture quality of the picture and consequently, increases the signal-to-noise ratio (SNR) of the video picture.
  - Spatially scalable pictures: Spatially scalable pictures are similar to SNR scalable pictures. The picture in the base layer that will be used to predict the picture in the spatial enhancement layer is interpolated by a factor of two either horizontally or vertically (1-D spatial scalability), or both horizontally and vertically (2-D spatial scalability). The decoder must be capable of handling custom picture formats, because interpolation step may produce pictures in non-standard format.
1.3.4 MPEG-1

The draft MPEG-1 standard was released in 1993. Although this is a generic video coding standard (not constrained for a specific application), it was primarily designed for storage on digital media such as CD-ROM supporting bit rates up to 1.5 Mbit/s. The standard employs a block based hybrid coding (predictive and transform coding) algorithm similar to block based video coding. This standard supports flexible picture types such as I-Pictures, P-Pictures and B-Pictures in order to provide good compression efficiency and added functionality (Pennebaker 1996). MPEG-1 includes no concept of interlace; every picture or frame contains all the lines that make up a complete image in the sequence. MPEG-1 uses Y, C_B, C_R coding and a 4:2:0 structure of color information. The luminance is coded in every pixel, but the color difference information is filtered to half the luminance resolution.

1.3.5 MPEG-2

The MPEG-2 standard was published in 1995, with a bitrate of 4 to 9Mbps. This aimed at broad variety of applications such as media storage, satellite terrestrial TV broadcasting. It builds on MPEG-1 algorithm including new tools for better quality and functionality such as interlaced video and scalable video coding for applications such as digital TV and HDTV. This is the first standard to introduce the concept of “profiles” and “levels” as means of implementing compliant decoders that support only a subset of syntax (profiles, e.g. particular set of tools) with restriction on capability (levels) such as maximum supported bitrate. Seven profiles were defined: Simple, Main, 4:2:2, SNR, Spatial, High and Multiview (ISO/IEC 13818-2 1994) (Ayloo 2006).
1.3.6 MPEG-4

In 1998, MPEG-4 was developed for obtaining flexibility, scalability, efficiency and robustness to adapt in multimedia applications (Mitchell et al 1996) (Torres and Kunt 1996) (ISO/IEC 14496-2 1998) (ISO/IEC JTC1/SC29/WG11 N2995 1999) (Sikora 1997). MPEG-4 supports object based video coding where a video scene is divided into different video objects that can be coded independently of each other; for example, foreground and background objects can be coded differently to each other. The particularities of MPEG-4 are the transformation and the VLC blocks.

The requirements for MPEG-4 can be summarized as follows:

- **Content-based interactivity:** Frame-based compression, as well as content-based access, manipulation and bit stream editing are supported. Compositing of synthetic and natural data and provisions for interactivity also are supported. Temporal random access, within certain time limits is improved.

- **Compression:** MPEG-4 provides better subjective visual quality than current and emerging standards at comparable data rates, in particular, at low data rates (64 kbps and lower). Requirements are made to support the coexistence of diverse network environments and constrained decoder resources.

- **Universal access:** Provides an array of error robustness capabilities, even under severe error conditions, such as long error bursts. Particularly targeted are low data rate applications, which would be affected the most by this type of noisy environments.
Content-based scalability: Fine granularity for quality (temporal and spatial resolution), complexity, and content scalability is achieved by MPEG-4.

1.4 EVALUATION METRICS

In order to measure the efficiency of the encoder, the commonly used metrics are compression, encoding time and visual quality.

1.4.1 Compression

The amount of compression in a video sequence is usually measured in one of two ways. The first way to observe compression is by the amount of compacting that has occurred in the sequence before and after encoding. This compacting amount is represented by a compression ratio.

\[
\text{Compression Ratio} = \frac{M}{N}
\]  

(1.1)

where \( M \) = Size of the original data and

\( N \) = Size of the compressed data

For example, a compression ratio of 2:1 means that the compressed video is twice as small as the original video. Therefore, on average, every M bits were reduced to N bits after encoding. Generally, compression is measured by observing the average bit rate of the compressed bit stream (Xiong et al 1999)(Torres and Delp 2000)(Ribas-Corbera and Neuhoff 1998).

The average bit rate is calculated in the following way:

\[
\text{Average bit rate} = \frac{\text{Total bits in video sequence}}{\text{Number of frames in sequence}} \times \text{Average frame rate} \quad \text{bits/sec}
\]  

(1.2)
1.4.2 Encoding time

For a complicated system, a direct calculation of its computational complexity can be very challenging. The computation time for an encoder is a reflection of its computational complexity. An appropriate metric for the total encoding time would measure the time span between the beginning and ending of an encoding session. Speedup is defined as

\[
\text{Speedup} = \frac{\text{Total run time for first encoding}}{\text{Total run time for second encoding}} \quad \text{Seconds}
\]  

(1.3)

1.4.3 Video quality

An accurate measurement of visual quality using a numerical metric must be based on a reliable model of the human visual system. Often, the visual quality of a reconstructed sequence is simply set to reproduce the original input sequence of the encoder. Video quality is an independent standard judged by the human eye; video fidelity is a relative standard that requires the comparison of one sequence to another. If the original, unprocessed, video sequence has a high video quality, then a high video fidelity in the reconstructed sequence translates to high video quality as well. The metric used to represent video quality is a video fidelity metric, called the average peak signal-to-noise ratio (PSNR), which is usually measured in decibels (dB). Let \( t \) be the time index denoting the location of the \( N \) frames of an input video sequence, from \( t = 0...N-1 \). Let \( f_{(i,j)} \) be the value of the pixel intensity located at coordinate \((i, j)\) in an original frame at time \( t \), where \( i = 0...X-1 \) and \( j = 0...Y-1 \). Let \( r_{(i,j)} \) be the pixel value at coordinate \((i, j)\) in the reconstructed frame at time \( t \). Then, the MSE between an original and reconstructed frame is defined as
\[ MSE(t) = \frac{1}{XY} \sum_{i=0}^{X-1} \sum_{j=0}^{Y-1} [f(i, j) - r(i, j)]^2 \]  \hspace{1cm} (1.4)

and

\[ \text{Average PSNR} = 10 \log_{10} \left( \frac{255^2}{\frac{1}{N} \sum_{t=0}^{N-1} MSE(t)} \right) \]  \hspace{1cm} (1.5)

It chooses to use the PSNR based on the results of a comparative analysis that shows it to be a relatively accurate measure of video fidelity.

1.5 SCOPE OF THE WORK

Video coding with high compression efficiency is required for many visual communication applications with limited channel bandwidth. Therefore, fast and accurate video compression technique is highly desirable to assure much reduced processing delay, while maintaining good reconstructed image quality.

Initially, set partitioning in hierarchical tree (SPIHT) algorithm based on wavelet transform evolved as the strong candidate to achieve coding performance and complex decoding with high PSNR. 3D SPIHT (three dimensional SPIHT) and LM SPIHT (low memory SPIHT) algorithms outperformed general SPIHT algorithm in improving high PSNR value for quality video compression system. However, 3D SPIHT and LM SPIHT are ineffective in maintaining high compression ratio, high gain and coding time.

Subsequently, gain is achieved through various types of coding schemes to improve the performance of video compression system. Generally, video signal has high temporal redundancies due to towering correlation between successive frames. Current systems fail to exploit the pertinent
temporal redundancy in the video frames to improve compression efficiency with less processing complexity.

Also, increasing the processing speed is a major constraint. Therefore, large number of fast block matching algorithms has been anticipated for motion estimation by limiting the number of search locations. The existing matching algorithm is not maintaining small computation requirement but also provides less matching accuracy.

Furthermore, attempts have been made to perform efficient video compression system by moving edge detection. Regular frame based approach needs to find the block position and edge types requires additional computational complexity. The current system improves coding efficiency both in terms of detection accuracy and processing speed.

Hence, in this thesis, an attempt has been made to improve the performance of video compression system with WASH tree algorithm, scalable ACC-DCT using up/down sampling approach, FWS-LMedS algorithm and moving edge detection based video compression technique.

1.6 OBJECTIVE OF THE WORK

An attempt has been made in the present work to enhance the quality of video compression system by employing different coding schemes as follows:

- To investigate the performance of video compression system using WASH tree algorithm.

- To examine the performance of the video compression system with scalable ACC-DCT using up/down sampling approach.
To study the performance of video compression system with block based motion estimation using FWS-LMedS algorithm.

To evaluate video compression system’s performance by moving edge detection based video compression using modified Kohenon mapping approach.

1.7 ORGANISATION OF THE THESIS

Chapter 1 provides an overview on video compression system. The need, scope and prime objectives of the present work and organization of the thesis are presented in this chapter.

Extensive literature related to the performance improvement of video compression system and various aspects of compression methods has been critically reviewed and presented in chapter 2. Summary of review of literature is also furnished.

Chapter 3 presents the quality improvement of video coding system by employing WASH Tree algorithm. A detailed discussion on the quality expansion with comfort of simulation results for the system employing WASH Tree algorithm based video coding are also incorporated.

Scalable ACC-DCT based video compression is presented in chapter 4. Performance analysis and discussion with the assistance of simulation results are clearly presented. Further, the system performance with up/down sampling technique is also included in this chapter.

Chapter 5 deals with Block based motion estimation using FWS-LMedS algorithm for video compression. The simulation results and performance comparison with complete discussion is concisely provided. Further, block based motion estimation using FWS-LMedS algorithm for the
system is presented along with simulation results and a meticulous comparison of performance is incorporated in this chapter.

Chapter 6 represents the system analysis with moving edge detection based video compression using modified Kohonen mapping technique. The mathematical model representing the proposed system is devised and presented. Finally a discussion on the simulation results and the performance comparison is tersely offered.

Chapter 7 concludes the thesis by emphasizing the major conjecture of the study. A summary of research contribution and the scope for future studies are also incorporated in this chapter.