Chapter 1

Introduction

A video signal is represented by a sequence of image frames. The frame rate of a video signal is defined as the number of frames produced or transmitted per second. For typical consumer applications such as mobile video communication, 30 frames per second (fps) is adequate. Also, it could be as high as 60 fps for very high end applications or as low as 10 fps or 15 fps for video conferencing applications. Applications based on video signal involve a huge amount of data whose storage and communication poses a difficult problem. On the other hand, as the time difference between any two successive frames of a video signal is very short (ranging from 1/10 second to 1/60 second, depending upon the frame rate), the contents of the neighboring frames are quite similar. Therefore, significant temporal redundancy is present in any video signal, which may be eliminated by video compression technology so that transmission and storage of video information become more efficient.

This introductory chapter starts off with highlighting the need to develop an efficient design for low power Motion Estimation (ME) module for a video encoding system. Among all the modules required for video compression, the ME module involves the highest computational intensity, and hence consumes the largest amount of power [1]. Power consumption is very critical for the present-day video applications such as portable videophone and digital camcorder, which are typically operated by battery power [2]. After a brief back-

ground on the need for video compression, the motivation of the present work is clearly stated. The chapter ends by giving an outline of the present thesis.

1.1 Need for Video Compression

With continuous progress of communication and memory circuitry, and consequent decrease in cost per transmitted bit and cost per stored bit, it is perhaps not immediately obvious why video compression is necessary. Video compression has two important benefits. First, video compression makes it possible to use digital video in transmission and storage environments that would not otherwise support uncompressed video. For example, even at low frame rates and/or small frame size, the current internet throughput rates are not sufficient to handle uncompressed video in real time [3]. Moreover, a Digital Versatile Disk (DVD) can store only a few seconds of uncompressed video at television-quality resolution and frame rate, and so DVD-Video storage would not be practical without video and audio compression. Second, more efficient use of transmission and storage resources are possible by video compression techniques. For an available fixed bit-rate transmission channel, it is always more practicable to send a high-resolution compressed video or multiple compressed video channels rather than to send a single, low-resolution, uncompressed video signal. Therefore, in spite of ongoing advances in storage and transmission capacity, video compression is likely to be an essential component of video encoding system for many years to come.

The main principle behind video compression is the exploitation of redundancy (both spatial and temporal) of data and the deficiencies of the human visual system [4]. The compression which results in loss of any information is termed as the lossy compression. On the other hand, there is no loss of information in lossless compression. Most practical video compression techniques are based on lossy compression, in which more compression is achieved with the penalty that the decoded signal is not identical to the original. The principal goal of video compression is to minimize the weights on the files and to maximize

the quality of the reconstructed video signal. As already mentioned, there is a very high amount of temporal redundancy present in a given video sequence. The change in the scene contents between the frames is due to the motion of the objects present in the frame. The principal idea behind achieving reduction of temporal redundancy is to encode a frame first and to treat it as the reference frame and to predict the consecutive frames based on this reference frame. The prediction step is termed as the motion estimation. Motion Compensation (MC) follows ME. Both motion estimation and motion compensation form the heart of a video encoding system. In order to facilitate the interoperability between compression at the video source and decompression at the destination, there is a demand for an international standard for the coding methods. As a result, modern video coding standards like MPEG-4 and H.264/AVC have been developed [5].

It is clear from the above discussions that motion estimation exploits the temporal redundancy present between successive frames and tries to remove the same in an efficient way. Motion estimation, which is computationally very intensive, involves about 80% of the total computational power of the encoder [1]. Block Matching Algorithm (BMA) is one of the most efficient and popular techniques for minimizing the temporal redundancy present between frames due to its simplicity [6, 7, 8]. In this technique, the current frame is divided into blocks of size 16×16 pixels known as the Macroblocks (MB), and for each MB in the current frame one searches for the best matched MB (within a search range) in the reference frame. Motion Vector (MV) is defined as the displacement between the MB in the current frame and the best matched one in the reference frame. In full search method, one tries to find the best match by searching within the entire search window in the reference frame which is formed by the given search range [9]. However, since this method typically computes the MVs for all the possible MBs located in the search window, the computational cost is very high. Hence, a large number of fast search algorithms have been developed [10, 11, 12, 13, 14, 15]. These fast search algorithms reduce the computational time as well as the hardware overhead to a considerable extent. However, a major drawback associated with these fast algorithms is that very often the search may be trapped

in some local minima, thereby producing suboptimal results.

1.2 Motivation Behind the Present Work

The present day video compression standards like H. 264 can realize a compression ratio of the order of 200:1, whereas the previous standards like H.261 used to provide compression ratio of the order of 98:1 only [16] for the same video quality. Therefore, the modern video coding standards provide a substantial amount of improvement in the coding efficiency compared to the previous standards. The coding gain calls for hardware systems of higher complexity. Motion estimation has been identified as the main source of power consumption in the video encoding systems. Although motion estimation based on BMA entails simple arithmetic computations, it involves a huge amount of memory access which involves considerable power consumption and also affects the overall speed of operation [17]. Consequently, there are not many existing implementations which take care of low power consumption and high speed of operation at the same time. Design of an architecture for ME, which can take care of the high memory bandwidth requirement, is crucial for an efficient video encoding system. This is what has been attempted in the present work - to design hardware structure for ME with low power requirements. Several techniques have been adopted in this work to design ME architectures with low area, low power consumption and high speed of operation to meet the real time requirements for the modern video coding/processing applications. One of them is to use fast search techniques for ME instead of the full search strategy. Due to computational regularity, motion estimation based on full search is generally preferred for Very Large Scale Integration (VLSI) implementation. Although the computational time of these fast search algorithms is much smaller than that required of the full search algorithm, the search data flow for these fast search algorithms is irregular which makes the memory access mechanism more complex than that of the full search algorithm. Moreover unlike the full search algorithm, the processing order of these fast search algorithms is not predefined but dynamic, which makes the controlling unit much more complicated than that of the full search algorithm. These factors pose con-

siderable challenges for designing VLSI architectures for fast search ME algorithms. Many techniques have been applied to reduce the overall power consumption of the ME modules by reducing the required memory bandwidth. This is achieved by reducing the redundant data access for different search locations. Additionally, highly parallel architectures have been developed to increase the overall speed of operation for the ME process. The work recorded in this thesis has therefore focused on the design and development of fast ME architectures characterized by high processing speed, low power, and low area making it suitable for portable video application devices that are typically operated by battery power and involve real time operation. Thus the major directions in which the present work has progressed are as follows:

- Design a power efficient ME architecture for fast ME based on unconstrained center biased diamond search algorithm
- Implement the Fast Two Step Search (2-SS) algorithm and related architecture for half-pixel ME
- Develop a new algorithm and architecture for quarter-pixel ME based on the statistical distribution of MVs at quarter-pixel locations
- Conceive a new ME method by combining fast search techniques with 1BT and its implementation on hardware
- Design, analyze, and implement an ME technique based on pixel truncation and adaptive search pattern.

1.3 Major Contributions of the Present Work

In the research work embodied in this thesis, effort has been directed towards development of low power architectures for implementing fast ME algorithms, which can be applied for real-time video compression with high video quality but low cost. In the present

thesis, fast and efficient ME algorithms have also been developed. A brief account of the work described in the present thesis is given as follows.

The first component of the work is concerned with developing high-performance architecture for the Unrestricted Center Biased Diamond Search (UCBDS) algorithm. In the present architecture, external memory access has been considerably reduced by adopting an intelligent data arrangement. As a consequence, power consumption is low for this design which can suit real time low bit rate video applications.

The present work next deals with sub-pixel ME which involves two steps, namely Half-Pixel Motion Estimation (HPME) and Quarter-Pixel Motion Estimation (QPME). The proposed HPME architecture is based on the fast two step search algorithm [69] that serves to reduce the number of search points at half-pixel locations. The proposed design implements HPME for variable block sizes as prescribed by modern video coding standards such as H.264.

This thesis also proposes a QPME algorithm which explores the most likely quarter-pixel locations while skipping the unlikely ones. It also proposes the corresponding QPME architecture which, in conjunction with the HPME architecture, can serve as part of a complete low-power H.264 video coding system. Results of implementing the above architectures demonstrate area and power reduction compared to the conventional sub-pixel ME structure while maintaining the desired quality of the reconstructed image. The proposed sub-pixel ME architecture is deemed appropriate for real time processing of HDTV frames.

The work recorded in this thesis goes on to address implementation of fast ME architectures based on One-Bit Transformation (1BT). The 1BT based ME starts with converting an image frame with 8 bits/pixel representation into a binary frame with 1 bit/pixel format. Next, while carrying out ME on these binary frames, Boolean XOR operation is used to find the number of non-matching points, which replaces the Sum of Absolute Differences (SAD) as the matching criterion. The power efficient architectures proposed in this thesis use Diamond Search (DS) algorithm to perform fast ME on 1BT frames. By taking the

advantage of high speed, the proposed architectures can be made to run at a reduced clock frequency subject to a constraint on the frame size and frame rate. FPGA implementations of the architectures reveal that the average power dissipation can be substantially reduced by lowering the clock frequency.

The present work also proposes a variable block size ME algorithm which is executed in two stages. At the first stage, ME is performed using pixel truncation, while at the second stage, ME is performed using full pixel resolution with an adaptive search pattern. For the sake of comparison, the proposed Fast Two Stage Search (F2SS) algorithm as well as the conventional full search algorithm has been applied on different benchmark video sequences. In the architecture proposed to implement the above algorithm, reduction of power is ensured by introducing a novel memory management scheme.

1.4 Outline of the Dissertation

The present thesis has been organized as follows:

Chapter 1: The objectives and scope of the thesis and the motivation for the work are presented in this chapter. It also specifies the overall contributions made in the proposed work.

Chapter 2: This chapter presents a survey on diverse research activities related to the existing algorithms and architectures for integer and sub-pixel ME, different matching criteria, one bit transformation based binary ME algorithms and ME based on pixel truncation.

Chapter 3: This chapter describes in depth the implementation of an ME architecture based on UCBDS algorithm. The synthesis results for the proposed architecture have also been included.

Chapter 4: An architecture for realizing the HPME algorithm is proposed in this chapter. A fast algorithm and its architecture for implementing QPME have also been described in this chapter.

Chapter 5: In this chapter, VLSI architectures have been proposed for the implementation

of fast ME based on a combination of DS and 1BT.

Chapter 6: A two stage fast algorithm for VBSME based on pixel truncation has been proposed in this chapter. A suitable low-power architecture for implementing the proposed ME algorithm has also been described in this chapter.

Chapter 7: This chapter summarizes the overall contributions and results of the work recorded in the thesis, and identifies some future research directions in the relevant fields.