An Architectural Study About Motion Estimation To H.264 Video Compression

Gustavo Girão, Sílvio Fernandes, Ivan Saraiva
Universidade Federal do Rio Grande do Norte
{ggirao, silvio}@lcc.ufrn.br, ivan@dimap.ufrn.br

Abstract

Vídeo Compression has been important in several multimedia areas like: digital television, videoconference, vídeo on demand among others. In this paper, a motion estimation prototype for video compression, based on the study of optimal and suboptimal algorithms, is presented.These studies aim the design space exploration of a motion estimator based on ITU-T H.264 Recommendation to the Sistema Brasileiro de Televisão Digital (SBTVD) project.

1. Introduction

A vídeo is formed by a sequence of images, called frames, which are segmented in macroblocks (MBs). The adjacent frames, usually has high similarity, which is known as temporal redundancy.

The Motion Estimation (ME) takes advantage of this redundancy to reduce que amount of bits that represents the vídeo scenes. Therefore, the ME is considered the most important part in video compression of modern patterns [1] including H.264 [2], nevertheless is computationally very expensive. There is several researches in ME algorithms once the efficiency of these algorithms determines the overvall efficiency.

The paper is organized as follows. In the next section there is a short discussion about motion estimation, followed by the presentation of the proposed prototype. The fourth section shows the synthesis results of the prototype, and finally there is the conclusion and future works discussion.

2. Motion Estimation

It is possible to construct a video sequence using a reference frame and the differences of adjacent frames.

These differences are represented by motion vectors which their coordinates are the indexes of the MBs that have the greatest similarity degree in comparison with the MBs, delimited by a search area (frame reference). The process of motion vectors calculation is called Motion Estimation [3].

The ME can be divided in two parts: the search, which consists in moving the MB inside the search area, and the similarity function, which measures how similar the MBs of adjacent frames are.

Several algorithms were proposed to balance the computational cost and efficiency (or performance) of ME. These algorithms can be classified as optimal and suboptimal. The first ones are more accurate but they are greedy, while with the suboptimal occurs the inverse. Examples of search algorithms are: exhaustive search, vector decimation**,** pixel decimation, hierarchic search, etc. Some of the similarity functions are: Simple Cross-Correlation (SCC), Normalized Cross-Correlation (NCC), Moravec, Sum of Squared Differences (SSD), Sum of Absolute Differences (SAD), etc [4].

3. Architecture

ME is computationally expensive, since the related algorithms, normally, use complex operations or are greedy algorithms, it means with high temporal complexity [5]. Futhermore, one fundamental requirement for the process is storing the frames that will be compared, so it is necessary to use memory resources and to manage them.

With that in mind, this paper presents an architecture (Figure 1) whose design is characterized by modularity.This feature seeks to evaluate the combinations between search algorithms and similarity measures. This architecture examination has been start using exhaustive search and SAD algorithms. Exhaustive search is an optimal algorithm that moves the MB sequentially through the columns of each line in a search area. The SAD makes the sum of the differences in absolute value and calculates the dissimilarity between the reference and search area MBs. Therefore, the lowest value obtained by the SAD after the exhaustive search defines the motion vector of the MB. Although they only use simple arithmetics operations, the algorithms are among the most computationally expensive that exists [5].

The motion estimator has two LPM-RAM memories destinated to the MB of current frame and search area. The memories are unidimensional and store the respective pixels sequentially. The access to the reference MB is done by sequential reading of the MB memory addresses, meanwhile to access a MB into the search area a calculation of the address must be made.

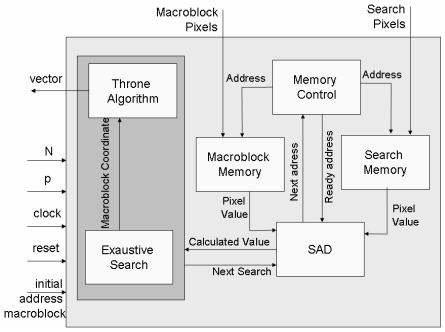

Figure 1. Motion Estimator Architecture

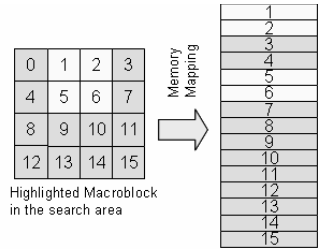

Figure 2. Search Area Memory Mapping

The computation of the address is done by the Memory Control module. This process is initiated after solicitation of the SAD module. The SAD, through its state machine, calculates the dissimilarity. After all calculations are made, the lowest value is selected by an algorithm called throne algorithm. The Exaustive Search associates this value with the coordinates of the MB from the search area which presents most similarity. These coordinates and the ones from the reference MB define the motion vector.

4. Results

The proposed architecture was implemented in VHDL and the Altera's Quartus II tool was used for simulation and attainment of the results of synthesis, shown in table 1. The synthesis results presented in this section were obtained to the EP1S10F484C5 ALTERA [6] *Stratix* family device.

These results confirm the simplicity of the used algorithms, expressed in the low consuming of logical elements and no DSP blocks were needed. The highest execution frequency was 104.78 MHz, and memory utilization matched the expected for a macroblock of 16x16 points and a search area of 64x64 points.

Table 1. Architecture Synthesis Results

Frequency (MHz)	104.78
Total Logic Elements	845 / 10,570 (7%)
DSP Blocks	0 / 48 (0%)
Total Memory bits	34,816 / 920,448 (3%)

5. Conclusion and Future Works

This paper has presented an ME architecture, which is part of H.264 encoder studies for the SBTVD project financed by FINEP [7].

The design of this architecture aims exploration of the advantages presented by several search algorithms and similarity functions, as well as their cost/benefit. Currently, the exhaustive search and the SAD similarity function are implemented, and they demonstrate efficiency in the vector calculation.

Nevertheless, the optimization of the algorithms, like a spiral exhaustive search or SAD acceleration [8] might capitalize on the architecture's resources. Besides, the synchronism analisys between the architecture modules points to the possibility of having parallel SAD modules, speeding up the process of comparing MBs. Also, a study about the utilization of larger internal memories for calculation of several motion vectors in parallel.

6. References

[1] Roma, N. F. V., *Processadores Dedicados Para Estimação de Movimento em Seqüências de Vídeo*, Dissertação de Mestrado, Lisboa, 2001.

[2] ITU, *Advanced video coding for generic audiovisualservices. Recommendation H.264.*, 2005.

[3] Richardson, I. E. G., H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia, Aberdeen, (2004).

[4] F. Dufaus and F. Moscheni, "Motion estimation techniques for digital TV: a review and a new contribution," *Proceedings of the IEEE*, vol. 83, no. 6, pp. 858-876, June 1995.

[5] Chen, T. et. al., *Single Reference Frame Multiple Current Macroblocks Scheme for Multi-Frame Motion Estimation in H.264/AVC*, Taiwan, 2005.

[6] Altera, URL: http://www.altera.com, June, 2005.

[7] FINEP, *Financiadora de Estudos e Projetos,* URL: http://www.finep.gov.br/. June, 2005.

[8] S. Vassiliadis, E. Hakkennes, S. Wong, and G. Pechanek. The Sum-Absolute-Difference Motion Estimation Accelerator. In *Proceedings of the 24th Euromicro Conference*, 2000.