A New Motion Estimation Algorithm of Low Computational Cost and Focusing on Full HD Videos

Cássio Cristani, Pargles Dall'Oglio, Marcelo Porto, Luciano Agostini
{crcristani, pwdalloglio, msporto, agostini}@inf.ufpel.edu.br

Group of Archietectures and Integrated Circuits - GACI
Federal University of Pelotas - UFPEL

Abstract

In this work a new Motion Estimation (ME) algorithm is presented. The Quarter Random Search (QRS) as is called has focus on high definition videos (HD), achieve good commitment between quality and computational cost. The problems inherent with HD videos have its origins at amount extremely high of local minima. Fast algorithms are directly affected by local minima, degrading visual quality and presenting higher quality losses in comparison with lower resolution videos. Thus, quality difference to Full Search (FS) (no affected) increase and few solutions can avoid local minima falls. The main goal of proposed algorithm is to handle efficiently with HD 1080p resolutions, keeping good quality and reducing computational cost in comparison with FS algorithm. The number of local minima in the motion estimation process is directly related to the video resolution, so for resolutions higher than HD 1080p, this problem becomes bigger. In this way, QRS contributes to high and ultra-high definition motion estimation.

1. Introduction

Nowadays, there are a lot of devices that uses high definition (HD) videos. The popularity of these HD videos mainly happens because its show a better visual quality compared to lower definitions. As each image of video has more pixels, the discretization is bigger and the amount of information increases significantly. However, it leads the videos near more and more of the real quality.

The video coders are flexible about the used resolution. The emerge encoder standard HEVC [1] seek support some resolution that state-of-the-art standard H.264/AVC not prioritized, such as Full HD (1920x1080 pixels). Compression rate projected to HEVC is twice more of the H.264/AVC since data volume aggregated to HD videos and upper definitions, without compression, is extremely greater [2].

The encoder process of HD videos needs of bigger computational cost and it is dominated to inter-frame coding [3]. Efficient algorithms of Motion Estimation (ME) with commitment mainly between cost and quality are needed to improve this coding module. The results obtain in ME can change during after coding since follow steps has tools which deal other features of the reconstructed frame. Fig. 1 illustrates the occurrence of local minima in a HD 1080p video, in a heat map form. The black regions represent better quality than red regions. Local minima can be seen as the resistance on search to found better blocks in comparison with already analyzer blocks. Increased video resolution a same region becomes more homogeneous, difficult the search and increased also the amount of local minima.

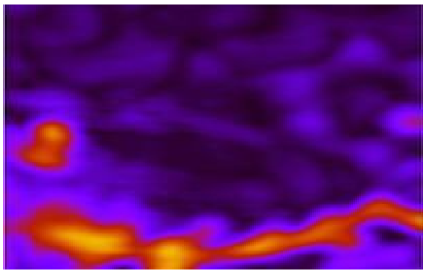

Fig. 1 – Local minima incidence on a frame.

The fast algorithm uses heuristics to accelerate convergence and reduce the computational effort. However, fast algorithms lose efficiency because they are susceptible to local minima fall. Simple heuristics are good alternative for low definition, but in HD best block can be distant of center. Thus, search is conclude early and next to center, that is detrimental to quality. In HD 1080p videos the cost of iterative purely algorithms, as the Diamond Search (DS) [4] and Hexagon Search (HS) [5], remains low. But the quality obtained by them had huge difference to optimal quality shown to Full Search algorithm (FS). This happen because they not transpose local minima hurdling. Algorithms with *N* steps predefined, as Three Step Search (TSS) [6] and Four Step Search (FSS) [7], has a worst behavior. In just published papers the local minima aren't being considered for performance tests, as occurs in [8] and [9]. Algorithms evaluate in low definition videos cannot be compares directly with high definition. Search window populated with a lot of local minima must be considers since this goal strongly affect algorithms susceptible to local minima.

This work presents the new ME algorithm named Quarter Random Search (QRS), focused on efficiency for HD 1080p. This algorithm aim decreases the computational cost in HD 1080p videos and to avoid local minima in greater resolution videos, keeping good quality and computational cost. The proposed algorithm isn't dependent of coding standard. The investigations made in this work are Motion Estimation and Compensation modules isolated, thus, the evaluation remains valid for process of all standards. QRS algorithm uses randomness as a main way of transpose the local minima. The similarity criterion used in this paper was Sum of the Absolute Difference (SAD) [10].

2. Quarter Random Search Algorithm

The algorithm proposed in this paper is named of Quarter Random Search (QRS). It is focused in High Definition videos, but may be good alternative higher resolutions. The QRS algorithm uses two strategies which not produces good results in HD videos, when applied individually. However, the main goal to QRS efficiency is the cooperation among these strategies, enabling thus achieve more distant candidate blocks from center. The QRS has basically two stages, presented sequentially, but it may be run parallel and not depend of which start.

Initial stage of the algorithm divides the search area in four sectors of same size and one is random selected. Inside the selected sector are randomly chooses *N* candidate blocks. Despite the probability of found the optimal block (global minimum) is 25% when one sector is used, compared with balanced random of *N* candidate blocks in each sector, this enough decrease the first stage cost. In other hand, each sector has good blocks that may be very next to optimal block, including to homogeneous search areas. This can be seen in black regions of the Fig. 2. This occurs because even biggest block represents a little piece of moving. Still with the increase definition these pieces are less significant to moving and including blocks without physical moving relation may be good candidate blocks. The randomness is a fast way to reach distant regions. It is efficient form adopted for us to avoid local minima falls. Finally, first stage compares the *N* candidate blocks, selecting the best. At this block is done a final refinement. The SDSP pattern [4] is applies aiming converge until more similar regions.

The second stage applies an iterative purely algorithm on the center of search area. In this work is used to original DS algorithm since it has low cost and guarantee good quality to low moving videos. This stage is responsible to generate from 50-60% of vectors. These vectors are near the center and easily can be found.

As the both stages no generate data dependence theirs may be parallel. Although no reduce the computational cost, the parallelism benefits hardware implementation and multi-core processing. Thus, the QRS algorithm performance can be equal or until better than classic algorithms, even considering real time applications.

Fig. 2 shows the flowchart of the QRS algorithm. The two main stages are describing them. In illustration can be seen that hasn't difference by which stage is started. Still both stages can be started parallel since the processing cost aggregate to each stage is similar. The first stage should make random choose, but its iterative refinement is less costly. The second stage which uses DS evaluates more blocks. Thus, good parallel may be extract of these features. Finally, after stages selected its block is done a comparison and the motion vector is generated for the best block.

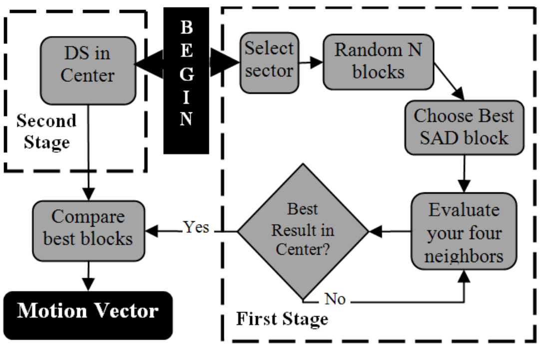

Fig. 2 – Block diagram of QRS algorithm.

Fig. 3 illustrates a hypothetically search area and shown the operation of the QRS algorithm. In this example, hatched sector was selected; it can be viewed as the 3rd mathematical quadrant. Inside were randomly selected *N* candidate blocks, this example has $N = 8$. A comparator selected the best block (represented to gray block in Fig. 3) and it is apply a Small Diamond Search Pattern (SDSP) [4], which evaluates its four neighbors (white blocks in Fig. 3). The region of best block is exposed to iterative steps of this pattern. When finished SDSP iteration search, the best block of the first stage is know. The second stage investigates the central region of search area used the original DS. This stage isn't dependent of the first one, thus, provides the better chance of convergence to other quadrants. But nevertheless, this DS instance doesn't reaches far regions in HD 1080p videos because amount of local minima, so presented more relevance slow moving blocks.

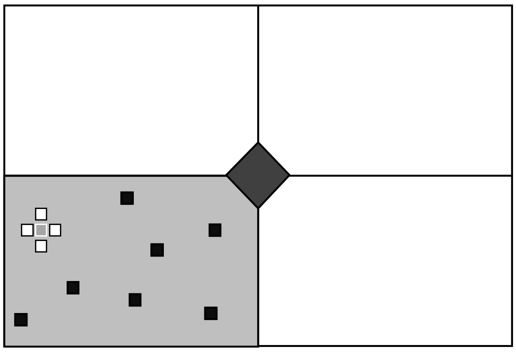

Fig. 1 – Hypothetical running to a frame with $N = 8$

3. Results and Comparisons

The Quarter Random Search algorithm was implemented in C language. The simulation used a framework developed in our group to provide a precise evaluation of the Motion Estimation and Motion Compensation modules isolated from the other modules of the video coder. Therewith a specific analysis of ME algorithms is held. In follow paragraphs the QRS algorithm will be evaluated and compared with some well-know ME algorithms.

Simulations were done for QRS algorithm using ten HD 1080p videos [11] with different motion activity. The initial two hundred frames of each video were process using block size of 16x16 pixels and search area size with range [-48,+48] from central block. The large video sample was used to raise results fidelity. The evaluation done in [12] consider three videos, this may tend results to video features.

Tab. 1 presented the simulation results for each video. In this Tab., the quality is measured to Peak Signal Noise Ratio (PSNR), which uses decibel scale. Computational Cost used is SAD, shown in millions. And the waste reduction percentage (WRP) represented the compression rate in relation as original size video. From these data is possible view the size problem in the ME optimization. The variations of threes variables is greatest and depends of factors as the video moving, the heuristic used. However, improve of three together, achieve good results for both, is very difficult.

In Tab. 1 the QRS algorithm is compared with others algorithms presents in the literature as: Full Search (FS) which is holder optimal quality; Diamond Search (DS), Hexagon Search (HS) which are iterative purely algorithms; Tree Step Search (TSS) and Four Step Search (FSS) which uses predefined steps but suffer to lack of final refinement; and the Uneven Multi Hexagon Search (UMH) [12] which is other algorithms focusing in HD videos. These algorithms were also implemented in framework previous describe and the simulations occur to equal conditions for all. The quality is measured to Peak Signal Noise Ratio (PSNR), which uses decibel scale. Computational Cost used is SAD, shown in millions. And the waste reduction percentage (WRP) represented the compression rate in relation as original size video. The challenge

Tab.1 – Motion Estimation Algorithms Comparative Results.

ALGORITHM	PSNR (dB)	SAD ($.10^6$)	WRP (%)
FS	35.89	14.662,6	64,34
UMH	34.42	311.96	67,64
QRS	**34,02**	**106,00**	**59,25**
DS	33.02	48.07	55,72
HS	32.79	32.52	54,58
FSS	32.40	58.03	53,63
TSS	30.94	43.51	44,79

In Tab. 2 the algorithms are disposed to quality sort. The proposed algorithm achieves 3rd best quality, losing 1,87dB in relation to the FS that present optimal result. It still loses 0,4dB in relation to the UMH that uses some heuristics to make high quality in HD videos. When compared with the iterative purely algorithms and step algorithms, the QRS algorithm has a largely gain in quality, being 1 dB compared to DS and reaching 3,08dB compared to TSS.

The computational cost obtained for all fast algorithms is less lot of FS cost. The QRS cost is almost 140 times smaller than FS. However, analyzing only fast algorithms, can be seen that the UMH algorithm have the greatest cost to get quality its. In relation of them, QRS has cost three times smaller to only 0,4dB of lose quality. Compared QRS to DS, it have almost twice cost but this compensates the low efficiency of the DS into HD 1080p videos. The waste reduction percentage achieve to QRS algorithm leads next of the best compression rates, showing a reduction in almost 60% of the video original size. The behavior of compression curve is practically linear for all algorithms.

4. Conclusions

The use of HD videos has increased in market world. These videos were supported initially to high definition televisions (HDTV), but dedicated hardware possible the supported to several devices, including mobile phone. Thereunto, the video encoders are very important, making viable jobs as transmission, storage and mainly processing which is a strong restriction in most of the devices.

The Motion Estimation step provides the higher compression gains in the current video coders. The ME algorithm has direct influence in the quality, compression and computational cost of the video coding. However, with the increase in the video definition the amount of local minima has bigger growth which deteriorates the quality obtained by fast algorithms.

In this paper the Quarter Random Search algorithm (QRS) was presented. This algorithm is focused in HD videos and uses the randomness as way of avoid local minima falls. Thus, it can achieve better quality than traditional fast ME algorithms. The QRS algorithm was simulated and compared to well-know algorithms and others algorithms focused in high definition.

QRS algorithm obtained a computational cost almost 140 times lower them FS algorithm with a quality decrease of 1,87dB. In relation to UMH algorithm, the quality was only 0,4dB lower, however the QRS can reduce in three times the computational cost. Comparing with iterative purely algorithms and steps algorithms the QRS had increase in the cost, but this is acceptable due to the achieved growth in the quality.

5. References

[1] JCT. Working Draft 3 of High-Efficiency Video Coding. JCTVC-E603, 2011.

[2] ISO/IEC-JTC1/SC29/WG11, "Joint Call for Proposals on Video Compression Technology," ed. Kyoto, Japan: Video Coding Experts Group (VCEG), 39th Meeting, 2010.

[3] Puri, A., et al., Video Coding Using the H.264/MPEG-4 AVC Compression Standard. *Elsevier Signal Processing: Image Communication*, [S.l.], n. 19, p.793–849, 2004.

[4] Zhu, S., MA, K., A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation. In: IEEE Transactions on Image Processing, Vol. 9, No. 2, 2000, pp. 287-290.

[5] C. Zhu, X. Lin and L. Chau, “Hexagon-Based Search Patten for Fast Block Motion estimation.” IEEE Transactions on Circuts and Systems for Video Technology, pp.349-355, May 2002.

[6] Jing, X., Chau, L, An efficient three-step search algorithm for Block motion estimation. IEEE Transactions on Multimedia, Vol. 6, No. 3, 2004, pp. 435-438.

[7] Tasdizen, O., et al., Dynamically Variable Step Search Motion Estimation Algorithm and a Dynamically Reconfigurable Hardware for Its Implementation. IEEE Transactions on Consumer Electronics, Vol. 55, No. 3, 2009, pp. 1645-1653.

[8] Byun, J., Choi, J., Kim, J., A Fast Multi-Reference Frame Motion Estimation Algorithm. *IEEE Transactions on Consumer Electronics*, Vol. 56, No. 3, 2010, p. 1911-1917.

[9] Xie, L., Huang C., Chen B., UMHexagonS Search Algorithm for Fast Motion Estimation. In: *International Conference on Computer Research and Development*, 2011, pp. 483 – 487.

[10] Ghanbari, M. Standard Codecs: Image Compression to Advanced Video Coding. United Kingdom: The Institution of Electrical Engineers, 2003.

[11] Xiph.org: Test media, available at <http://media.xiph.org/video/derf/>, May, 2011.

[12] X.Yi, J.zhang, and N. Ling, "Improved and simplified fast motion estimation for JM," JVT-P021, July 2005