NEW QUARTER RANDOM SEARCH MOTION ESTIMATION ALGORITHM: A GREAT QUALITY-COST TRADE-OFF

Cássio Cristani, Pargles Dall'Oglio, Marcelo Porto, Luciano Agostini.
GACI, Federal University of Pelotas – UFPel

ABSTRACT

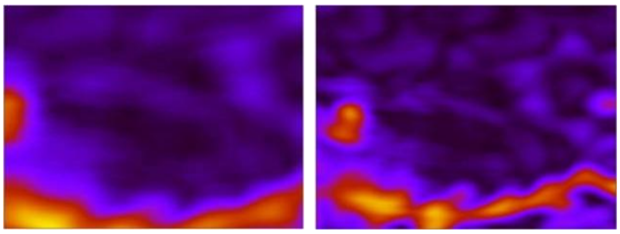
The digital videos have been very used lastly. In actual scenery, high definitions videos have gained popularity for provide better visual quality that previous format. High Definition videos (HD) generate a high data volume, making impracticable the transmission, storing and mainly processing. The HEVC standard will optimize twice compression rate to reduce efficiently the data volume of these videos. The Motion Estimation (ME) still remains as most important step to compression in encoders; however, the algorithms must take into account measures like quality and computational cost. Moreover, ME algorithms should consider the resolution features that can bias result. In this paper is show a new Motion Estimation algorithm which does not depends of standard used. The algorithm is named Quarter Random Search (QRS) and uses the randomness as way to avoid fall in the local minima. The simulation result applies to ten HD 1080p video shows that QRS algorithm is an efficient alternative order to low cost, decrease almost 140 times the computational effort in relation to Full Search algorithm. Moreover, the QRS algorithm presented great objective quality results when compared with fast algorithms.

1. INTRODUCTION

Nowadays, there are a lot of devices that uses high definition videos (HD) surrounded us. The popularity of these HD videos mainly occurs because its show the better visual quality compared to lower definitions. Increasing video definition, we are inserting and approximating pixels which represent more accurately the captured images; consequentially this representation is more complex. With this, amount of information has an expressive raise and video becomes more near of the reality.

The video coders are flexible in relation to resolution used. The emerge encoder standard High Efficiency Video Coding (HEVC) [1] seek support some resolution that H.264/AVC (state-of-art standard) not support efficiently, such as HD and greater definitions. The compression rate project to emerge standard is twice more to H.264/AVC since data volume aggregate to without compression videos greater than HD is much greater [2].

HD videos encoder process needs bigger computational effort than lower definitions, impacting directly in time. It is dominated to inter-frame coding [3]. Efficient algorithms of Motion Estimation (ME) are needed to achieve good computational cost, without depreciates video quality. However, increasing video definition, the local minima problem worsens and the obtained quality is degrades. This problem is a consequence of greedy methods. Thus, the fast algorithms lose efficiency because it is often uses greedy methods to decrease choice complexity. The Figure 1 illustrates local minima presence in HD videos – (a) 720p and (b) 1080p - in heat map form. The black regions represent a lot of sub-optimal results but only one optimal. The search aim found blackest point in this universe. It is main factor to lose efficiency of these algorithms in HD videos.



(a) 720p (b) 1080p

Figure 1: Presence of local minima in one frame: (a) 720p and (b) 1080p.

The fast algorithm uses heuristics to accelerate convergence and reduce the computational effort. However, increase the video definition performance has been decreasing. This happens because the amount of local minima largely growth and some heuristics are susceptible it, taking the algorithms conclude search early and next to center. In HD 1080p videos the cost of purely iterative algorithms, as the Diamond Search (DS) [4] and Hexagon Search (HS) [5], remains low. But the quality obtained by them has huge difference to optimal quality shown to Full Search algorithm (FS). Algorithms with N steps predefined, as Three Step Search (TSS) [6] and Four Step Search (FSS) [7]; has the same behavior. This show the impact of the video resolution because all these algorithms almost achieve same quality than FS in low definition but does cannot evolve like FS. However, the local minima problem has been disregarded in most of recently published papers, as occurs in [8] and [9] which only evaluates low definition videos for performance tests, without consider nothing HD or more definitions.

In this work is show the new algorithm named Quarter Random Search (QRS). This algorithm aim decreases the computational cost in HD 1080p videos which have a high incidence of local minima, achieve good quality and keeping compression rates. The proposed algorithm does not dependent of standard and its implementation can be applied in emerging HEVC standard. But it also should

SForum 2012 - Student Forum on Microelectronics
This work has been developed by the first author(s) in the scope of the undergraduate studies

be used in others standards. QRS algorithm uses randomness as a way of transpose local minima. The investigations made in this work evaluate Motion Estimation and Motion Compensation modules separately of the encoder. The similarity criterion used in this paper was Sum of the Absolute Difference (SAD) [10].

2. QUARTER RANDOM SEARCH

The algorithm proposed in this paper is named Quarter Random Search (QRS). It is focused in High Definition videos and can be good alternative to bigger resolutions. The QRS algorithm uses two strategies that does not produces good results in HD videos, when applies of isolated way. These strategies are randomness and Iteration pure. However, the main goal for results found to QRS is cooperation them, which allows achieve the distant regions of center and avoid fall in the local minima. The QRS has basically two stages, which will be presented sequentially for convenience, but it may be run parallel. This is important aspect for future hardware develop of the QRS algorithm.

Initial stage of the algorithm divides search area in four sectors of same size and selected one randomly. Inside selected sector are drawn *N* candidate blocks. Despite the probability of found the optimal block (global minimum) is 25% when uses one sector, if 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 might be very next to optimal block, including to heterogeneous search areas. What can be seen in black regions of the Figure 2. This occurs because even the biggest block size represents the only a little piece of moving. Still with the increase definition this pieces are less significant to moving and many blocks without relation with physical moving object can be good candidate blocks. Finally, this stage compared the *N* candidate blocks in chosen sector and applies a Small Diamond Search (SDSP) [4] as final refinement to converge to more similar regions.

The second stages apply one purely iterative algorithm at center of the search area. In this paper is used original DS since it has relatively low cost and guarantee good quality to low moving videos, where greater vectors are next of the center.

As the both stages does not generates data dependence its can be parallelized. Although not reduce the computational cost, the parallel benefit hardware implementation and multi-core processing. Thus, the performance of the QRS algorithm in real time applications is as good as classic algorithms.

Figure 2 shows the flowchart of the QRS algorithm, where the two main stages are presented. It can be seen that does not make difference by which stage is started. Still, both stages can be started parallel because processing cost aggregate in each stage is similar. The first step has random cost plus iterative refinement cost. This refinement needs less effort than second stage. The

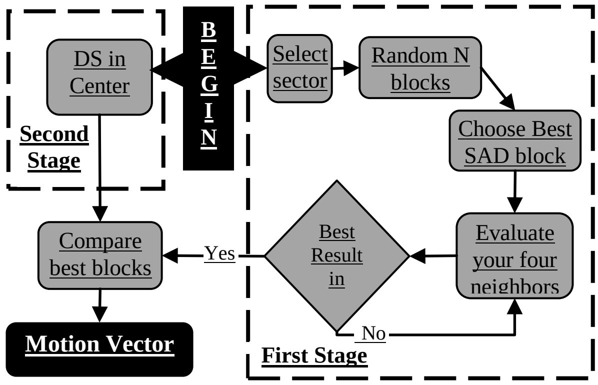


Figure 2: QRS algorithm flowchart of two stages.

second stage which uses complete DS, evaluates more blocks per iteration. Thus, the effort of each stage is almost equal, that is good for parallel. Finally, the blocks chosen in both stages are compared and the motion vector is generated for the best block.

Figure 3 illustrates a hypothetic search area and shown the two stages 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 (not illustrates) selected the best block, represented to gray block in Figure 3. After is applied a SDSP, which evaluates its four neighbors (white blocks in Figure 3), besides to explore that region with an iterative search. When finished SDSP iterative search, the best block of the first stage is know. The second stage investigates the central region of search area used to original DS. This stage is not dependent of the first one, thus, provides the better chance of convergence to other quadrants than selected. But nevertheless, this DS instance does not reaches far regions in HD 1080p videos because amount of local minima, as explained in first section, so presented more relevance to slow moving blocks.

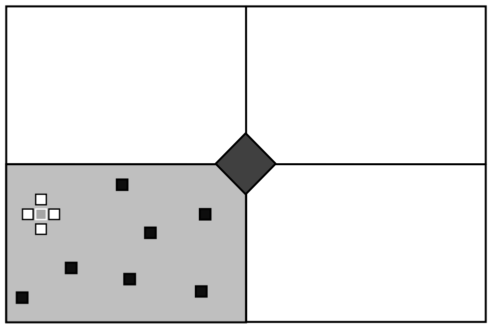


Figure 3: Example of the behavior QRS algorithm.

3. RESULTS AND COMPARISONS

The Quarter Random Search algorithm was implemented in C language. The simulation used framework produce in group us. It was developed to evaluate precisely the Motion Estimation and Compensation modules, isolately of the encoder. Therewith a specific analysis of the

Motion Estimation algorithms is held. In follow paragraphs will be analyzed QRS algorithm. It will be also compared with well-known ME algorithms.

Simulations were done for QRS algorithm has used ten HD 1080p videos [11] with different speed moving and textures. The two hundred initial 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 chosen to raise fidelity of the results. The evaluation done in [12] considers only three videos, which might tend results to video features.

Table 1 presented the QRS algorithm simulation results for each video. In this table, the quality is measured to Peak Signal Noise Ratio (PSNR), which uses decibel scale. The Computational Cost used is SAD, shown in millions. And the Waste Reduction Percentage (WRP) represents the compression rate in relation to original video data volume. From these table is possible view the size problem in the ME optimization. The three measures have lots of variations and depend of factors as the video moving and the heuristic used. Thus, improve of three together, achieve good results for both, is very difficult.

Table 1 - QRS simulations to PSNR, SAD and WRP

1080P VIDEO	PSNR (dB)	SAD ($.10^6$)	WRP (%)
Blue Sky	30,97	104,92	81,34
Man In Car	39,34	91,48	34,21
Pedestrian Area	34,23	106,17	65,80
Riverbed	26,32	118,67	53,97
Rolling Tomatoes	38,09	89,76	24,08
Rush Hour	37,16	93,85	52,35
Station2	38,16	92,92	61,62
Sun Flower	37,97	110,73	79,32
Tractor	30,47	122,81	66,80
Traffic	27,50	128,71	73,05
Average	**34,02**	**106,00**	**59,25**

In Table 2 the QRS algorithm was compared with others algorithms of the literature as: Full Search (FS) which is holder optimal quality; Diamond Search (DS), Hexagon Search (HS) which are purely iterative 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 insert some strategies and also focus in HD videos. These algorithms were also implemented in framework ours. The simulations have occurred to equal conditions for all. The variables evaluated are also PSNR, SAD and WRP, using the same measure unit. Values in table are average of the simulation to ten HD 1080 videos.

Table 2 - Motion Estimation Algorithms Comparative Results

ALGORITHM	PSNR (dB)	SAD ($.10^6$)	WRP (%)
FS	35.89	14,662.60	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 Table 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, which join some heuristics to make high quality in HD videos. When compared with the purely iterative algorithms and step algorithms, the QRS algorithm has a largely gain in quality, being at least 1 dB compared to DS and reaching 3,08dB compared to TSS.

The computational cost obtained to FS is biggest that all fast algorithms. The QRS cost is almost 140 times smaller than FS. However, analyzing only fast algorithms, can be seen that the UMH algorithm have the biggest 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 more cost but this compensates the low final quality of the DS operates HD 1080p videos. The Waste Reduction Percentage achieve to QRS algorithm is next of the best compression rates. QRS algorithm is showing a reduction in almost 60% of the video original size. The behavior of compression curve is enough equilibrate among the most algorithms.

Table 3 presents a quality comparative with ATME (Adaptive True Motion Estimation) algorithm [12] which is based in 3DRS algorithm. This algorithm optimizes the set of candidate blocks used by 3DRS through multi-objective optimization algorithm. The quality shown is best result of the simulation results contained in paper (Full version algorithm). The sequences used are 1080p definition and 50 frames per second than increase similarity between frames and local minima consequently. Analyzing Table 3, can be seen that for all videos the QRS obtained better quality, ranging from 0,45dB to 1,34dB. The computational cost was not compared since the measures are different, becomes unfair the comparisons with our results.

Table 3 - Quality comparative with work focused in HD

VIDEO	*QRS*	*ATME*	*3DRS*
Crowd Run	28,95	28.50	28.43
In To Tree	35,96	34.62	34.54
Park Joy	27,28	26.02	25.80

4. CONCLUSIONS AND FUTURE WORKS

The uses of HD videos have 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 encoders has an important role, making viable jobs as transmission, storage and mainly processing which is a strong restriction in most devices.

Motion Estimation step represents the most compression gains. Viewed only the ME module the algorithm choose influence directly the quality, compression and computational cost of coding. However, increased the video definition the amount of local minima has bigger growth which deteriorates the quality obtained to fast algorithms.

In this paper was proposed the Quarter Random Search algorithm (QRS). This algorithm is focused in HD videos, used the randomness as way of avoid local minima falls and, thus, obtained good quality. It was simulated and compared to well-know algorithms and others algorithms focused in high definition.

QRS algorithm obtained a computational cost almost 140 times less to FS algorithm with the quality decreased 1,87dB. In relation to UMH the quality was only 0,4dB less, but QRS reduce three times the computational cost. Comparing with purely iterative algorithms and steps algorithms the QRS had increase in the cost, but this is needed to growth quality. The difference of quality, however, had gains from 1dB to 3,07dB. Comparing with others algorithm which uses only three HD 1080p different videos, the proposed algorithm obtained a better quality for all videos.

5. REFERENCES

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

[2] JVT Editors (T. Wiegand, G. Sullivan, A. Luthra), Draft ITU-T Recommendation and final draft international standard of joint video specification (ITU-T Rec.H.264 |ISO/IEC 14496-10 AVC), JVT-G050r1, Geneva, May 2003.

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

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

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

[12] Cetin, M.; Hamzaoglu, I.; “An Adaptive True Motion Estimation Algorithm for Frame Rate Conversion of High Definition Video,” International Conference on Pattern Recognition, 2010, pp. 4109-4112.

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

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