Motion Estimation and Inter Prediction Mode Selection in HEVC

AHMAD ASGHAR, MUHAMMAD ATIQ, RAI AMMAD KHAN, NADEEM A. KHAN
Department of Electrical Engineering, SBASSE
Lahore University of Management Science, Lahore
Pakistan
{12060004, 12060009, 13100205, nkhan}@lums.edu.pk

Abstract: - Motion estimation is one of the most computationally challenging parts of any video encoder. An estimated 50% to 70% encoder complexity is due to motion estimation. Recent video coding standards like H.264/MPEG-4(AVC) and HEVC include a sophisticated motion estimation algorithm called Test Zone Search. This paper examines the algorithms designed to perform Coding Unit type and size selection in HEVC and the proposed improvements to the Test Zone Search algorithm with respect to HEVC while also suggesting further improvements.

Key-Words: - Inter prediction, coding unit, depth, motion estimation, test zone search, H.264/MPEG-4(AVC), HEVC.

1 Introduction

Motion estimation (ME) and motion compensation is seen as one of the most important methods of exploiting redundancy in motion pictures. Its importance is so high that 50% to 70% of encoder complexity is dedicated to the motion estimation process. However, as we move towards higher resolution videos, computational complexity is becoming a bigger concern. This is why motion estimation is seen as a major savings area in terms of computational expense. Recent improvements in computer architecture mean that new machines offer higher parallel processing capabilities which can be used to process video frames in “tiles”. This concept was introduced in the latest video coding standard to be released, the HEVC. This reduced the computation load on a single processor making the video coding process more robust.

However, in HEVC and the previous video coding standard H.264/MPEG-4(AVC), motion estimation using multiple reference frames was also introduced which added to the complexity of the motion estimation process. While it provided the ability to improve PSNR, it also added extra computational cost.

The most important aspect of this holistic approach is that it combines the advantages provided by each approach. In the best case scenarios this may lead to a total improvement of over 50% in some instances in terms of BD-Data Rate and also provide improvement in BD-PSNR.

With evolution of video coding techniques, motion estimation has changed significantly. The motion estimation technique incorporated into H.264/MPEG-4(AVC) was a variant of the test zone search algorithm called enhanced predictive zonal search (EPZS). This was primarily because of the fact that older motion estimation methods required improvements to manage the predictive methodology offered by H.264/MPEG-4(AVC) video coding standard. H.264 introduced the option of using multiple reference pictures for motion prediction. In such a scenario EPZS and PMVFAST were the motion estimation techniques that provided the closest results to full search for motion estimation in multi-field predictive environment.

The motivation for writing this paper comes from the fact that although there have been efforts to cover different aspects of the motion estimation process, very few studies have proposed a complete algorithm for the motion estimation process. This paper discusses the methodologies employed to cost optimize different portions of the motion estimation process. It singles out the most promising proposals and suggests a comprehensive algorithm that covers the major aspects of motion estimation process from coding unit depth selection to locating the best match. Section II of the paper presents an overview of the motion estimation techniques employed in H.264/MPEG-4(AVC) and test releases of HEVC. Section III contains a review of the best approaches that cover the different parts of the motion estimation process. Section IV compares different motion estimation techniques proposed for HEVC. Section V presents complete algorithm for motion estimation in an HEVC encoder. It should be mentioned here that this paper presents a holistic view of the motion estimation process in HEVC

ISBN: 978-960-474-330-8

with focus on different motion estimation techniques for inter predicted coding units.

2 Motion Estimation in HEVC

Complexity of motion estimation has increased further as the pixel density of motion pictures has increased. Along with that HEVC has introduced the concept of coding unit trees which increase the complexity of the motion estimation process. The idea of coding units is a novel concept in the field of video coding. Before that, analysis was performed on units that generally had a fixed size. H.264/MPEG-4 (AVC) was the first to introduce variable block sizes that ranged from 16x16 to 4x4. While it gave the advantage of selecting a unit size that could optimize the encoding time, the largest block size available was still the macro-blocks. HEVC introduced hyper-blocks (64x64) with the option of coding the block in its entirety or breaking it down into further Prediction Units (PUs) and Transform Units (TUs). The introduction of such transform trees has provided encoder designers with the advantage of managing computational resources efficiently. Introduction of coding trees has also reduced the data bit rate by nearly 50% for high pixel density video sequences.

Sub-pixel motion estimation techniques were also a focal point in H.264/MPEG-4 (AVC) whereby a new method for search refinement was introduced such that once a correlating block for the current block was found; a sub-pixel level estimation was performed to find the exact match of current block. Sub-pixel approach is data intensive because it ensures maximum PSNR. The significance of sub-pixel approach in terms of HEVC remains high because of the fact that HEVC build on some concepts adopted from H.264/MPEG-4 (AVC) like variable block sizes and variable PSNR.

These developments are the reason motion estimation process must now include the selection of coding unit type selection and coding unit depth selection because a non-optimized approach to coding unit selection will lead to coding unit sizes that will be either too big or too small meaning the motion estimation process will not be able to perform as well as it is expected to. For these reasons, this paper proposes a complete approach to motion estimation including coding unit depth selection and motion estimation/refinement stages.

3 Review of Approaches

Motion estimation techniques for HEVC have been discussed in greater detail in several recent publications. Coding units were introduced as a new feature in HEVC and CU depth and type selection have been the point of focus in many publications since the concept of CU was introduced. Motion estimation techniques have been around much longer and therefore have been more widely discussed but here some of the most well cited approaches are discussed with focus on HEVC.

3.1 Coding Unit Selection Approaches

The algorithms discussed in following section target two different but extremely important aspects of Coding Units. One is the selection of prediction type for a coding unit and the other is the depth to which a coding unit should be broken before it can be encoded.

3.1.1 Coding Unit Depth Selection

Coding unit depth selection is the second important aspect of coding unit based video encoding. HEVC reference software introduced coding units that ranged from 64x64 to 8x8 distributed into different coding depths. This provided a very clear advantage of selecting the coding unit size that would fit the coding mode and the level of video quality required. Coding depth selection is therefore an extremely important part of the overall video coding process. Different researchers have proposed algorithms for selection of coding unit depth for HEVC. The algorithms include coding unit depth selection based on Bayesian decision rules [2], coding unit depth selection based on weighted SVM [3] and coding unit depth selection based on coding depths of adjacent and preceding collocated units [4]. While each algorithm provides significant improvement in terms of data rate saving by optimizing the coding unit depth selection process, a tradeoff with coding and computational complexity must be considered. In this regard, the decision to select coding unit depth based hierarchical fast algorithm was made because it adds less computational complexity to the coding unit creation process than the other two algorithms.

A comparison of the empirical results of the three algorithms presented in table (1) also suggests that this might be the better option among the three algorithms. Results show that the hierarchical fast coding unit depth selection algorithm [4] adds the least amount of data overhead when averaged over several sequences and it provides a saving of over 40% processing time. The weighted SVM algorithm [3] does provide more saving in terms of time but the data overhead is greater compared to the hierarchical algorithm.

Algorithm	ΔBits	ΔTime
Hierarchical	0.01%	41.59%
Bayesian	1.88%	41.40%
Weighted SVM	1.29%	44.83%

Table 1: Comparison of Coding Unit Depth Selection Algorithms

Analysis of the hierarchical fast coding unit depth selection shows why the algorithm is better suited for robust encoding. According to the algorithm if a certain coding unit with a coding depth that is rarely used comes up, it can be skipped. The following threshold is used to make this decision for a certain coding unit:

$$\frac{Num_{depth\,(x)}}{Num_{depth\,(x+1)\times 4}} < Th1$$

Where $Num_{depth(x)}$ is the number of coding units with depth '*x*' where '*x*' can be from 1 to 4 and $Num_{depth(x+1)}$ is the number of coding units one depth lower. Th1 is the threshold that, once exceeded will mean that depth '*x*' or lower will not be used for coding unit size.

However, if a coding unit depth that was previously skipped becomes more frequent, it cannot be skipped any longer. This decision is made on the basis of a second threshold proposed by the authors given as:

$$\frac{Num_{depth\,(x)}}{Num_{depth\,(x+1)\times 4}} > Th2$$

where '*Th2*' is the threshold used to decide when to re-open coding unit depth '*x*' for usage. According to the authors, '*Th1*' and '*Th2*' are decided based on empirical evaluations.

Coding unit depth selection is a highly important feature because if the coding unit depth is too small, we may receive a comparative gain in terms of PSNR but we will suffer in terms of data rate. Therefore, there is a tradeoff between the two. This algorithm makes the decision of avoiding a certain coding unit depth based on the co-located coding units and the coding units around it. The drawback is that we limit the coding unit depth until a predefined threshold is met.

3.1.2 Coding Unit Type Selection

The HEVC reference software HM developed by the joint team of experts from ITU-T and ISO contains a fast algorithm for the selection of the type of encoding a coding unit will under-go. The decision for this algorithm is based on the resolution of an object offered within a picture frame. For very high resolution images or parts of images intra coding is preferred. In the coding mode selection flow for HEVC encoders, the first picture of a video sequence and the first picture at each clean random access point into a video sequence is coded using only intra-picture prediction that uses some prediction of data spatially from region-to-region within the same picture, but has no dependence on other pictures [1]. For all remaining pictures of a sequence or between random access points, inter-picture temporally predictive coding modes are typically used for most blocks.

Intra-picture coding has been further complicated by the provisioning of 35 coding directions made available in the HEVC reference software which means decisions regarding intra-coding have to be made carefully because in high-resolution video sequences any mismatch in coding unit size or prediction can lead to large drops in PSNR and increase in data bit rate.

The decision regarding coding mode selection is therefore based on the fact that intra prediction is preferred whenever maintaining high PSNR becomes more important than ensuring high data compression. Inter-prediction mode and skip mode are preferred where substantial bit rate savings can be ensured with slight degradation in PSNR. HEVC reference software HM has covered this aspect in detail and the algorithm used for mode selection is very robust. While inter-prediction mode is used where motion is detected, there are often areas, or in the case of HEVC coding units, where the motion is too little for inter-prediction to be the optimal solution. In such cases skip mode is employed in which the coding unit is not taken through the process of encoding and is simply skipped while the decoder is informed about the action so that it can use the data from a previous frame to construct that coding unit.

Different coding unit depth selection algorithms also refer to the selection of coding unit type [2] [3] [4]. According to the hierarchical fast coding unit depth selection algorithm [4], this decision is made simply on the basis of the RD-cost of encoding a prediction unit (PU) using inter prediction coding and intra picture coding. If either of the two comes on top, then that coding mode is used for the selected PU. The coding unit depth selection algorithm based on Bayesian probabilities [2] considers mode selection as a part of the process and involves it as a feature in the optimized solution process. It looks at the coding mode and depth which have the least cost in terms of computation and data overhead. This is similar to the approach in the previous algorithm with the exception that it takes coding mode into account when making depth decision and not after depth decision has been made. The weighted SVM algorithm [3] uses the same approach as the

Bayesian probabilities algorithm i.e. it considers mode selection as a part of the depth selection process and decides the mode and depth based on the cost of selection. A direct comparison between the two approaches for coding mode selection i.e. after depth selection process and within depth selection process is difficult because none of the selected algorithms focuses on the cost of mode selection process on its own. However, an analysis of the two approaches would suggest that if coding unit type selection decision is made while coding depth is also decided, it may lead to an increase in data rate and encoder complexity whereas deciding coding unit type once coding unit depth has been decided means that the encoder complexity is not increased.

The algorithm for hierarchical coding unit depth selection [4] suggests that coding unit type selection may also be made part of the process. Once the decision regarding whether or not to skip a certain coding depth or not is made, the algorithm then calculates the cost of using intra picture coding over inter picture coding. If the cost of using intra picture coding comes out to be comparable to or lower than inter picture coding the intra picture coding is preferred and the encoding process moves in that direction. However, if the RD-cost of intra picture coding is much higher, then inter picture coding is preferred and the process moves on to motion estimation.

3.2 Search Strategies for Best Match in HEVC

While efficient processes for coding unit size and coding mode selections can remarkably improve the performance of an encoder, motion estimation still remains a vital aspect for data saving. Several algorithms have been put forward to improve the process of motion estimation that focus on two separate approaches for motion estimation. The two alternate motion estimation algorithm types proposed for HEVC are sub-pixel motion estimation algorithms and zonal search algorithms.

Considering the significant difference between the two approaches it is important to understand why they are being discussed together. In context with the proposal for a holistic motion estimation approach, it was important to consider every possible methodology employed previously or proposed after HEVC reference software was released.

3.2.1 Zonal Search Motion Estimation

The motion estimation process used in HEVC reference software is Zonal Search algorithm. This algorithm was also the preferred choice for H.264/MPEG-4 (AVC) reference software due to its holistic approach to find the best match. The original zonal search algorithm was based on square search grids for locating the best match. Improvements were later proposed to use diamond search grids instead of square grids based on empirical results [7]. Further improvements proposed the use of hexagonal search grids instead of diamond or square search grids because of reduced computations with minimum loss of PSNR [8]. Latest improvements propose the use of rotating alternate hexagons based on hexagonal search grids [9].

Because rotating hexagonal grids is a direct extension of hexagonal grids a comparison between diamond search grids and rotating hexagonal search grids was performed and the results show that using rotating hexagonal search grids is more efficient in terms of computation time and cost. These results have been listed in table (2). In the table, TZS refers to the general zonal search algorithm. TZS(d) is test zone search with diamond grid, TZS(h) is test zone search with hexagonal grids, TZS(rh1) and TZS(rh2) are test zone search with rotating hexagon 1 and 2 respectively and TZS(et-rh1) is test zone search with rotating hexagon 1 with early termination.

Algorithms	BD-PSNR	BD-Bit Rate	Δ ME Time (%)	Δ Encoding Time (%)
TZS(d) vs. TZS(h)	0.01	-0.293	-	-
TZS(d) vs. TZS(rh1)	0.0013	-0.0199	14.81	10.37
TZS(d) vs. TZS(rh2)	0.0017	-0.0276	-	-
TZS(d) vs. TZS(et-rh1)	0.001	-0.012	53.15	37.81

Table 2: Comparison of different zonal search algorithms

Experimental results suggest that for an extremely small degradation in PSNR and slight increase in overhead data rate, significant gains can be achieved in motion estimation time and encoding time when moving from test zone search with diamond grids to test zone search with rotating hexagons and early termination. The methodology behind this significant gain is discussed further on.

The proposals for changing zonal search algorithm focus on replacing the search grid of the original TZS algorithm from square or diamond search grids to hexagonal search grids. Initial proposal [8] was to use static hexagonal search which was improved further [9] to replace static hexagons with rotating hexagonal search grids.

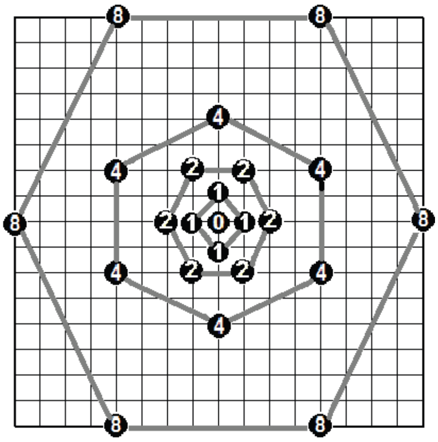

Fig. 1: Rotating Hexagonal Search Pattern for Motion Estimation with stride length 8

The authors suggest that instead of using diamond search grids (fig. 2(a)) that had to compare a total of 52 separate points, hexagonal search grids (fig. 2(b)) should be used which have 6 search points per grid and a total of 40 comparisons. This would lead to a straight forward saving of 23% computational cost compared to TZS with diamond search at a cost of 0.01dB average loss in PSNR. Further saving was achieved by employing a strategy where the algorithm would move from grid search to raster search as soon as the search went to a grid size that was larger than the initially defined threshold for starting raster scan. The authors initially gave two different versions of hexagonal grids, horizontal hexagon search grids and vertical hexagon search grids. The authors found that horizontal hexagon search grids were more sensitive to lateral motion than vertical motion whereas vertical hexagons performed vice versa.

Improvements proposed later to the initial algorithm introduced rotating hexagons (fig, 3) [9] where a horizontal hexagon would be followed by a vertical hexagon and so on until the inner grid of 4 points was reached. This helped in getting the best out of both hexagonal grid patterns. They also proposed an adaptive threshold for early termination of the search algorithm. As per the proposal, the algorithm would calculate a certain threshold based on the number of packet units and coding unit depth as given by the following formula:

$$Th(d,p) = (2^{d+p}) \sum_{i=1}^{N(d,p)} \frac{cost(i)}{N(d,p)}$$

where *N(d,p)* represents the total number of coding units in the first frame of GOP for the depth '*d*' and partition size '*p*', and cost is the SAD. Once this threshold is exceeded the algorithm will move to the refinement process using the point where distortion cost was minimum within the searched area.

For the refinement process, the algorithm again employed rotating hexagonal grids around the best match point and reduced the search grid size by half until the global minima was found. The results for the proposed algorithms were very encouraging, providing a saving of nearly 53% in terms of computational complexity while only suffering a loss of 0.001dB PSNR and 0.012% increase in coding rate compared to TZS with diamond search. Results of comparison are shown in table (2).

3.2 Sub-Pixel Motion Estimation

Sub-pixel motion estimation algorithms are popular because of the level of resolution and accuracy they ensure. The major advantage as mentioned before is in terms of PSNR because of sub-pixel level estimation. In H.264/MPEG-4 (AVC) sub-pixel level estimation could be taken to as low as 1/8th pixel size but here it is assumed that sub-pixel estimation is carried out only to ½ pixel depth. Several proposals have been put forward for sub-pixel motion estimation process all of which target the minimization of RD-cost function for SAD at a particular pixel.

Comparison between different algorithms shows that the main difference comes when evaluating the computational cost of each algorithm. Of the studied algorithm one algorithm proposes a two-step approach based on "rough" and then "precise" estimation using 5 and 6 search points [5] whereas the other algorithm finds global minima using a two-step approach and all nine points surrounding the actual point [6]. In terms of computational efficiency, the nine point search algorithm presents a better option compared to the coarse-fine approach.

One way to distinguish between the two algorithms is to look at the empirical results from the respective papers. The issue here is that HEVC provides a wide array of options including different quality factors for a video sequence. Also the authors of the two papers do not present a comparison with a common approach which makes direct comparison difficult. And finally the two algorithms use different video sequences for benchmarking which makes it even more difficult. However, a rough comparison is given in table (3). It can be seen from the comparison of the two that although the rough strategy based ME algorithm provides marginally better PSNR, there is a big gap in the data rate and search points per partition. Results suggest that for a marginal loss of quality, nine point search algorithm can provide significant saving in data rate and computation.

Method	PSNR (dB)	Bit Rate (kb/s)	Search points per partition
Rough Strategy Based Sub-pixel ME	37.44	3858.808	2.6767
Nine Point Search Based Sub-pixel ME	36.895	2191.635	1.5925

Table 3: Comparison of sub-pixel motion estimation algorithms

Based on the above results, nine point sub-pixel motion estimation algorithm was found to be better suited for practical applications. An analysis of the algorithm shows why. The authors state that to find the exact best match using a cost function, the best results were obtained when a nine point cost function was employed. If the designers wanted to employ a less computationally expensive function, they could use a six or five point cost function where the parameter values could be calculated using heuristic techniques. The authors suggest a modified version for the five point cost function that, for only a little additional computational overhead compared to the standard five point cost function, could deliver more accurate results. They suggested that the nine points should be considered two separate set of point where the points on the x and y axis i.e, 0 degrees are in the first set and the points at an angle of 45 degrees in the second set. The algorithm would first find the local minima for set one and then the local minima for set two. The algorithm would then find the global minima by taking the average of the two points. The authors justified their proposal by providing evidence that the global minima thus found is very close to the actual global minima of the error surface.

4 Towards a Highly Efficient Approach for Motion Estimation in HEVC

In Section 3 main advantages and disadvantages of each algorithm are discussed. These include Coding Unit size selection, Coding Mode selection in a Prediction Unit, sub-pixel search to find best matched pixel and zonal search algorithm for motion estimation. In this section we propose an algorithm that will help improve not only PSNR but also reduce data rates by using the best of all techniques

We propose an algorithm that starts by finding the optimal coding unit depth and coding mode for each prediction unit. The approach for finding the coding unit depth and the coding mode will be hierarchical fast coding unit depth selection [2]. The algorithm then proceeds to find the best search start point for Test Zone Search using the same approach as the one that is used to find the best matched sub-pixel using nine point search algorithm except for the fact that here the unit will be block rather than pixel.. As discussed previously, TZS with rotating hexagonal grids only considers 5 points for search initiation. This can degrade the PSNR when compared against full search algorithm. We propose a combination of the two strategies such that nine point search will act as the launch board for TZS by helping it find the best search initiation block. This will help us achieve a gain in PSNR while saving the computational complexity that would have been incurred if full search algorithm or TZS with diamond search algorithm were to have been employed.

Overall the proposed algorithm will provide PSNR improvement in motion estimation phase, data rate saving in motion estimation state and computational time saving over the entire algorithm.

4 Conclusion

The computational cost of motion estimation can be quite high compared to other parts of the encoder. This paper discusses the techniques which are proposed and employed in order to compensate that cost. Because of its approach to motion estimation for HEVC, this paper also considers coding unit selection as one of the vital parts of the motion estimation process. It presents a review of the most promising techniques for coding unit depth selection and motion estimation and finds that while each approach may have some short comings but when combined together, they can provide a holistic methodology for the motion estimation process in HEVC. This can be seen in the proposed algorithm and it is expected that the algorithm will give a gain of over 50% compared to existing techniques in terms of BD-Rate while also providing gains in terms of BD-PSNR.

Acknowledgment

The authors would like to acknowledge the contribution of Dr. Nadeem Khan whose guidance was pivotal for the successful completion of this project.

References:

[1] T. Wiegand, B. Bross, W.J. Han, J.-R. Ohm, and G. J. Sullivan, Working Draft 3 of High Efficiency Video Coding, JCTVC-E603, JCTVC of ISO/IEC and ITU-T. Geneva, Swiss, Mar. 2011.

[2] Xiaolin Shen, Lu Yu, Jie Chen "Fast Coding Unit Size Selection for HEVC based on Bayesian Decision Rule" IEEE 2012 Picture Coding Symposium.

[3] Xiaolin Shen and Lu Yu "CU Splitting Early Termination Based on Weighted SVM" EURASIP Journal on Image and Video Processing 2013

[4] Jie Leng, Lei Sun, Takeshi Ikenaga and Shinichi Sakaida "Content Based Hierarchical Fast Coding Unit Decision Algorithm For HEVC", International Conference on Multimedia and Signal Processing 2011.

[5] Weiyao Lin, Krit Panusopone, David M. Baylon, Ming-Ting Sun, Zhenzhong Chen and Hongxiang Li "A Fast Sub-Pixel Motion Estimation Algorithm for H.264/AVC Video Coding" IEEE Transactions on Circuits and Systems for Video Technology, vol. 21, no. 2, February 2011

[6] Wei Dai, Oscar C. Au, Sijin Li, Lin Sun, Ruobing Zou, "Fast Sub-Pixel Motion Estimation with Simplified Modeling in HEVC", IEEE International Symposium on Circuits and Systems (ISCAS), 2012.

[7] Tourapis, A.M., Au, O.C. and Liou, M.L. "New Results on Zonal Based Motion Estimation Algorithms-Advanced Predictive Diamond Zonal Search" ISCAS 2001

[8] N.Purnachand, L.Nero Alves and A.Navarro, "Improvements to TZ search motion estimation algorithm for multiview video coding", IEEE IWSSIP 2012, Vienna Apr 2012.

[9] N.Purnachand, Luis Nero Alves and Antonio Navarro, "Fast Motion Estimation Algorithm for HEVC", IEEE Second International Conference on Consumer Electronics - Berlin (ICCE-Berlin), 2012