Fast Motion Estimation in HEVC Inter Coding: An Overview of Recent Advances

Yongfei Zhang[1, 2*], Chao Zhang[1], Rui Fan[3]
[1] Beijing Key Lab of Digital Media, School of Computer Science and Engineering, Beihang University, Beijing, China, 100191
[2] State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China, 100191
[3] China Academy of Electronic and Information Technology, Beijing, China, 100041
*Corresponding Author: Yongfei Zhang (E-mail: yfzhang@buaa.edu.cn Tel: +86-1082314108)

***Abstract*—High Efficiency Video Coding (HEVC), the latest video coding standard, is becoming popular due to its excellent coding performance, in particular in the case of high-resolution video applications. However, the significant gain in performance is achieved at the cost of substantially higher encoding complexity than its precedent H.264/AVC, in which motion estimation (ME) is one of the most time-consuming parts that effectively removes temporal redundancy. During the development, especially after the release of H.265/HEVC, plenty of fast ME algorithms have been developed to reduce the motion estimation complexity for better application of HEVC into practical real-time video applications. In this review, we provide a comprehensive review of the state-of-the-art fast ME algorithms for HEVC inter coding, for both integer-pixel and fractional-pixel ME algorithms. In all, this review paper provides a comprehensive review of the recent advances of ME for HEVC inter frame coding and hopefully it may provide valuable leads for the improvement, implementation and applications of HEVC inter-prediction as well as for the ongoing development of the next generation video coding standard.**

Index Terms— HEVC, Inter Coding, Motion Estimation.

I. INTRODUCTION

High Efficiency Video Coding (HEVC)[1], the latest video compression standard developed by the joint collaborative team on video coding (JCTVC), can significantly improve the coding performance compared its predecessor H.264/AVC[2], which is however achieved at a much improved computational cost of up to 2-10 times higher computational complexity, which makes it quite difficult to apply in real-time video applications [3-5].

Considering the high coding efficiency and pervasive applications of HEVC, low-complexity thus fast HEVC encoder is urgently needed and a great amount of fast algorithms have been developed to reduce the high complexity of HEVC and for better application of HEVC into practical real-time video applications [6-9].

Since motion estimation (ME) is one of the most time-consuming parts in HEVC, plenty of fast ME algorithms have been developed in the literature to reduce the computational complexity of motion estimation thus the video codec [10, 11].

In this paper, we present a comprehensive survey of the fast motion estimation algorithms in inter-frame coding in HEVC, including both integer-pixel and fractional-pixel fast ME algorithms. Hopefully, it might help researchers to better cope with the latest Call for Proposals (CfPs) for the next generation video coding standard beyond H.265/HEVC [12].

The rest of this paper is organized as follows. Section II provides a brief overview of the inter-frame coding, with special focus on ME in HEVC. In Section III and IV, the fast integer-pixel and fractional-pixel ME solutions are reviewed. Section V forecast the research trends and concludes the paper.

II. MOTION ESTIMATION IN HEVC

Among all the coding techniques employed in HEVC, inter-frame coding, represented by Motion Estimation and Compensation (MEC), is one of the most important parts of video compression and the major contributor to compression efficiency. It is effective for finding the best matched block in the reference frames to reduce temporal redundancy, the major redundancy in video compression, between successive frames. Then, only the Motion Vector (MV), generated by ME and representing the displacement between the best matched block and the current prediction block, and the residual after Motion Compensation (MC), instead of the original video pixels, need to be encoded and stored or transmitted.

The entire ME process is made up of three coarse-to-fine procedures, namely, MV prediction, integer-pixel ME and fractional-pixel ME. First, MV prediction predicts the start search position for the following motion search by utilizing the neighboring motion information. In HEVC, Advanced Motion Vector Prediction (AMVP), a new and effective technology that predicts the starting search position by referencing the motion vector (MV) information of spatial and temporal motion vector candidates, is adopted, which derives several most probable candidates based on data from adjacent PBs and the reference picture. The displacement between the starting search position and the current coding PU is called a predictive motion vector (PMV). HEVC also introduces a merge mode to derive the motion information from spatially or temporally neighboring blocks [1].

The second step is integer-pixel motion estimation, which is conducted using appropriate search strategies from the starting search position related to PMV until the best integer-pixel search position is obtained. Block matching algorithm (BMA) is the most popular search algorithm for ME because it is

This work was partially supported by the National Key R&D Program of China (Grant No.2016YFC0801001), the NSFC Key Project (No. 61632001) and the National Natural Science Foundation of China (No. 61772054). This paper is partially done when Rui Fan were with Beijing Key Lab of Digital Media, School of Computer Science and Engineering, Beihang University, Beijing, China, 100191.

simple to implement but also performs reasonably. The basic idea of BMA is that the frame is divided into fixed-size blocks (PUs in HEVC). The most matched block within a search window in the reference frame is obtained based on the rate-distortion cost (RDCost), which is measured in *Eqn*. (1) and (2) as follows

$$RDCost(\boldsymbol{mv}, \lambda_{motion}) = SAD(s, r(\boldsymbol{mv})) + \lambda_{motion} R(\boldsymbol{mv} - \boldsymbol{pmv}) \quad (1)$$

$$SAD(s, r(\boldsymbol{mv})) = \sum_{i=1}^{W} \sum_{j=1}^{H} |s(i,j) - c(i-x, j-y)| \quad (2)$$

where $\boldsymbol{mv} = (mv_x, mv_y)$ is the MV of current PU, $\boldsymbol{pmv}$ is the predictive motion vector (PMV) and λ_{motion} is the Lagrange multiplier related to the quantization parameter. $R(\boldsymbol{mv}\text{-}\boldsymbol{pmv})$ represents the number of bits for coding the difference between motion vector $\boldsymbol{mv}$ and predictive motion vector $\boldsymbol{pmv}$ based on a look-up table. SAD is the distortion between the current block s and the reference block r determined by $\boldsymbol{mv}$, which is a measurement of distortion in the process of integer-pixel ME. In *Eqn*. (2), $s(i, j)$ is the pixel value at position (i, j) in the current frame; $c(i\text{-}x, j\text{-}y)$ represents the pixel value at position $(i\text{-}x, j\text{-}y)$ in the reference frame. W and H denote the width and height of the block, respectively.

The earliest and most straightforward full search (FS) strategy traverses all the positions in the search window and obtains the optimal MV with the minimum *RDCost* through the most exhaustive computation. Although FS provides the best quality amongst various ME algorithms, its computational complexity is very high and can involve as much as 40-80% of the total encoding time.

To address this drawback and achieve a balanced point between the coding performance and computational complexity, test zone search (TZS) is implemented as the build-in fast search mechanism (FSM) in the HEVC test model (HM) [13]. First, the start search position is determined by checking the PMV and zero motion. As a second step, a diamond search pattern or square search pattern is implemented, and an additional raster search is performed when the difference between the obtained motion vector and start position is too large. In the last step, an extra diamond search or square search is performed as a refinement search until the best search position is picked. Although TZS reduces ME complexity to a much greater extent than FS, the computational complexity is still huge for real-time systems because there are too many search points. To further reduce the complexity of TZS, plenty number of fast integral ME algorithms have been developed in the references, which will be reviewed in Section III.

Third, to further reduce the prediction residual, fractional-pixel motion estimation is implemented around the optimal integer-pixel position to obtain the final best-matched fractional-pixel position as the last step. Similar to integral ME, the optimal fractional pixel search position around the best integer-pixel search position is also determined according to the rate-distortion cost. In HEVC, quarter-sample precision is used for the MVs in luma component, and 7-tap or 8-tap filters are used for interpolation of fractional sample positions (compared to six-tap filtering of half-sample positions followed by linear interpolation for quarter-sample positions in H.264/AVC). The fractional sample interpolation process for the chroma components is similar to the one for the luma component, except that the number of filter taps is 4 and the fractional accuracy is 1/8 for the usual 4:2:0 chroma format case (where, in H.264/MPEG-4 AVC, only two-tap bilinear filtering was applied). Thanks to the well-designed, more complicated 8-tap luma sample and 4-tap chroma sample DCT-based interpolation filter coefficients, HEVC tends to improve the total encoding performance by more than 10% compared with the state-of-the-art video coding standard, H.264/AVC [2], which also simultaneously brings significant computational complexity. According to the computational complexity analysis of motion estimation in [14], fractional pixel motion estimation accounts for approximately 60%~80% of the computational complexity of the entire motion estimation process because it involves numerous DCT-based interpolation filter operations and a large number of rate-distortion calculations. It was shown in [15] that the average number of pixel accesses, multiplications and additions during the interpolation process in HEVC is almost twice that in H.264/AVC. This higher computational complexity makes fractional pixel motion estimation too slow for real-time video applications. Consequently, plenty of fast algorithms have been proposed to accelerate the fractional pixel motion estimation process in HEVC, which will be reviewed in Section IV.

III. Fast Integer-pixel Motion Estimation

Fast integer-pixel motion estimation has drawn great attentions due to at least the following two reasons. First, the integer-pixel ME is very time-consuming and might occupy 40-80% of the total encoding time. Second, the accuracy of the integer-pixel ME has a large influence on the performance of the subsequent fractional pixel ME [15].

As mentioned in Section II, TZSearch [4], as the default fast integer pixel ME algorithm of the HEVC standard, improves the encoding speed by nearly 100 times compared to the full search algorithm. Even so, the high computational complexity is still a bottleneck for real-time applications. In recent years, reducing the computational complexity of integer pixel motion estimation has become research hotspot. According to the emphasis, the research can be broadly divided into three categories, namely, search pattern design, search window decision and early termination strategies [15].

A. Search Pattern Design Algorithms

Different search patterns has been designed in many fast block-matching algorithms, for example, three step search (TSS) [16], four step search (FSS) [17], diamond search (DS) [18], hexagon search (HEXBS) [19], unsymmetrical cross multi hexagon search (UMHexagonS) [20], etc. These search patterns can greatly reduce the computational complexity of the integer pixel motion estimation, but sometimes these methods are easy to fall into local optimum and lead to performance degradation.

In order to achieve better balance between coding performance and computational complexity, Parmar [21] and Jeong [22] presented pentagon pattern and rotating pentagon pattern respectively. On the other hand, Yang [23] proposes a directional search algorithm with a square pattern. A quadratic pattern is introduced in [24], which utilizes the sum of absolute difference distribution to predict the start search point of each step for obtaining the optimal matching location by a coarse-fine order.

However, the motion characteristics of different video regions vary greatly and we expect the regions with complex motion should consume more computing resources to ensure video quality, while simple or stationary region motion estimation should be done as quickly as possible to find the optimal matching location. So, using same search pattern to the videos with different motion characteristics may be difficult to get a good tradeoff between performance and coding efficiency. To address this problem, a motion classification-based search pattern design algorithm was proposed [15]. By exploring the motion relationship of neighboring blocks and the coding cost characteristic, the Prediction Units (PUs), the basic unit of ME, are first categorized into one of three classes, namely, motion-smooth PUs, motion-medium PUs and motion-complex PUs. Then, different search strategies are carefully designed for PUs of each class according to their respective motion and content characteristics. Experimental results shown that the motion classification-based search pattern design algorithm outperforms state-of-the-art fast ME algorithms in terms of both coding performance and complexity reduction.

B. Search Window Decision Algorithms

The algorithms in this category concentrate on reducing the search window size, i.e., the number of search points by dynamically adjusting the size of the moving search window, thus reducing the computational complexity of the overall coding.

As we know, the search window size for H.264/AVC is 16 and it is extended to 64 in HEVC. HEVC Reference software HM exploits the dynamic search window algorithm to adjust the range of integer pixel motion estimation by calculating the temporal distance between the reference frame and the current encoding frame. Ko [25] found that horizontal and vertical component of motion vector differences (MVDs) approximately satisfy the Laplace distribution respectively, then proposed a search window design based on the hitting probability of MVDs to get an adaptive search range. However, Dai [26] has found that the distribution of MVD in [25] is more similar to Cauchy distribution through a large number of experiments, which improves the prediction accuracy of the search range and improves the encoding performance. Shen [27] divides the motion into three kinds, namely homogeneous-motion, normal-motion and complex-motion, and then uses the MVD distribution information of adjacent blocks to predict their search range respectively. [25-27] utilize the MVD information of spatio-temporal blocks to predict the current block's search window size, but the correlation between adjacent blocks' MVD information is not very high that cannot guarantee the accuracy of block's search range. Liao [28] proposes to use the MVD information of the parent CU to predict the current block's search window size and to establish a linear relationship between the size of the search window and the MVD of CTU.

To sum up, these algorithms can't always give the right prediction of search window size which leads to degradation of encoding performance particularly when the global optimum search location is not within the range of search window.

C. Early termination strategies

The early termination strategy achieves acceleration on encoding by terminating all or part of the integer pixel motion estimation process in advance. These algorithms can be further divided into two classes of sub methods.

The idea of the first class is that if the encoding performance of the current search location is acceptable in the course of search process, then the subsequent search process will be terminated. [29-33] propose to compute the cost threshold using the encoding information of spatio-temporal adjacent blocks and determine whether the coding performance is acceptable through the relationship between the current encoding cost and the threshold. Pan [34] proposes that if the motion vector predictor (MVP) of the parent node is equal to 0, then the integer pixel motion estimation of the child node can terminate early and directly uses the MVP predicted by advanced motion vector prediction (AMVP). But this method is easy to significantly reduce the encoding performance because of the inaccurate threshold setting.

The early termination strategies in the second class reduce the computational complexity by skipping the motion search in the impossible location of the motion search process and directly search the position of the subsequent position. Typical algorithms include successive elimination algorithm (SEA) [35], multilevel SEA [36], global SEA [37], and confidence interval based algorithm [11, 38]. These algorithms exploit the look-up table to determine whether the current search position will get worse encoding performance on the premise of not directly calculating the current coding cost, then skip to the subsequent search process. As a result, this kind of algorithms can effectively reduce the huge complexity cost of the full search algorithm, but it is not suitable for the simple search patterns, because it needs to collect the coded frame and reference frame information before the frame encoding, which brings a neglected extra computation.

The fast integral-pixel ME algorithms are listed in Table I for clearer presentation.

As far as we can see, a good fast integer pixel motion estimation algorithm must be content-adaptive with using any of above methods. The integer pixel motion estimation [39-44] based on region classification achieve a better tradeoff between coding performance and coding speed by designing different methods for different regions of the video. Li [44] proposes a fast integer pixel motion estimation method which is adopted by HEVC standard with an excellent performance, but it still has the problem of misjudging the motion characteristic. A precise content-adaptive fast integer pixel motion estimation algorithm is expected.

TABLE I. PERFORMANCE OF FAST INTEGER-PIXEL MOTION ESTIMATION ALGORITHMS

Emphasis	Properties	Author, Year, Reference	Anchor	Configuration	Reported Performance			
					BD-Rate (%)	BD-PSNR(dB)	△ ME Time (%)	△Encoder Time (%)
Search pattern design	Pentagon pattern	Parmar2014 [21]	HM 14.0	search range: 64 maximum CU partition depth: 4	0.358	-0.00399 (△ PSNR-Y, dB)	22.295	14.506
		Jeong 2015 [22]	HM 14.0	search range: 64 maximum CU partition depth: 4	0.2464	-0.0078	≈18	-
	Square pattern	Yang 2014 [23]	HM 8.0	search range: 64 GOP size: 4	0.146	-0.006	60.80	11.16
	Quadratic pattern	Gao 2015 [24]	HM 14.0	search range: 64 maximum CU size: 64 maximum CU partition depth: 4	1.2	-0.06	-	≈56
	Motion Classification -based patterns	Fan 2017 [15]	HM16.0	LDP, RA	0.17/1.12	-	82.47[1]	12.47/ 20.25
Search window decision	Follow cauchy distribution	Dai 2012 [26]	FS with non-adaptive SR in HM 3.0	LDP(C_{prob}: 98.4/98.8/98/99.2)	0.4/0.4/0.3/0.2	-	-	≈95
	Reference spatio-temporal information	Liao 2015 [28]	HM 10.1	-	-	-0.067	81.4(average accessed pixel, %)	-
Early termination strategy	Reference spatio-temporal information	Nalluri2015 [31]	FS/TZSearch in HM 16.0	Avg LDP,LDB,RA	0.705/0.429	-0.038/-0.032	98.113/43.181	84.982/23.247
		Medhat 2016 [32]	FS in X265	Search range:64 QP: 28	0.89(△ Bit rate,%)	-0.0105 ((△ PSNR, dB)	-	45.92
	Use MVP	Pan 2016 [34]	TZSearch in HM 12.0	LD/RA	0.55/0.86	-0.020/-0.034	20.12/18.52	15.04/12.29
	Based on confidence interval	Hu 2014 [11]	FSM in HM 12.1	LD	0.97(BD-Rate-Y, %)	-	69.76	13.05
		Hu 2013 [38]	HM 7.0	LD	-	-0.0241 (△ PSNR-Y, dB)	73.49	16.71
Overall	Adopted by standard	Li 2015 [44]	TZSearch in HM 10.0	Threshold:10	0.5(BD-Rate-Y, %)	-	-	49

[1] The ME time saving is measured as average number of search points (*ASP*) for one ME [15], as compared to that of TZS in HM.

IV. FAST FRACTIONAL-PIXEL MOTION ESTIMATION

Fractional-pixel motion estimation aims at finding the minimum rate distortion cost at the 49 fractional pixel locations around the optimal integer pixel location to further reduce the prediction residual and improve the coding performance. Fast fractional-pixel ME algorithms can be roughly divided into two categories, namely fast interpolation algorithms and interpolation-free algorithms.

A. *Fast Interpolation Algorithms*

The fast interpolation is to speed up fractional-pixel motion estimation by skipping the search process at fractional-pixel locations where the rate distortion cost may be poor.

H.264/AVC standard Test model JM and HEVC Standard Test model HM use a coarse-to-fine fractional-pixel motion estimation strategy which can be split into two steps. The first step is to search the integer pixel motion estimation position and the 8 1/2 pixel locations around it, then find the location with the minimum cost of the rate distortion. The second step with this location as the center of the search around the 8 1/4 pixel position from which to find out the minimum cost of the rate distortion as the final optimal fractional-pixel motion estimation position. Although the above method can effectively reduce the computational complexity of fractional-pixel motion estimation without significantly affecting the video quality, solving the remaining amount of calculation is still a difficulty for real-time applications. In [45-47], different search patterns or search strategies are designed for 1/2 pixel and 1/4 pixel motion estimation in order to simplify the search process. [48, 49] predict the starting search position of the fractional-pixel motion estimation then performs a fine search near the position and finally finds the optimal fractional-pixel motion estimation position. [50, 51] predicts one to three optimal fractional-pixel candidate positions by the estimated rate distortion cost of the surrounding adjacent integer pixels and interpolates the candidate position and calculates the cost of the rate distortion, then chose the location with the minimum cost of the rate distortion as the optimal fractional-pixel motion estimation position. [52, 53] classify the current encoded PU according to spatio-temporal information or current PU pixel motion estimation rate distortion cost, and limits the search accuracy of fractional-pixel motion estimation of different classes of PU.

Although the proposed method can save some computational complexity in fractional-pixel motion estimation, the computation complexity is still high because of the large amount of interpolation and rate distortion cost estimation in fractional-pixel motion estimation.

B. *Interpolation-free Algorithms*

The interpolation-free algorithms aim to directly estimate the exact optimal fractional-pixel position without the time-consuming fractional-pixel interpolation and rate-distortion cost calculation after the integer pixel motion estimation. On one hand, since the time-consuming fractional-pixel interpolation and rate-distortion cost calculation after the integer pixel motion estimation is skipped, these fast algorithms can achieve much more time saving as compared to those algorithms in the previous category, about 90% more specifically. On the other hand, since the computational complexity of the prediction process is not high and the complexity proportion of the fractional-pixel interpolation and the rate distortion cost calculation is fixed in the overall coding complexity, these algorithms bring similar encoding speed up, as will be shown in TABLE II.

The main idea of the method is to use the distribution of the adjacent integer pixel rate distortion cost to predict the optimal fractional-pixel position. Although the local optimal problem is a common problem in the estimation of the integer pixel motion since the fractional-pixel motion estimation is carried out only within an integer pixel range the singularity of the error surface is obvious [54]. For this reason the error surface model is used to reflect the rate distortion cost distribution within the fractional-pixel range and to estimate the optimal fractional-pixel motion estimation position. Typical model includes the 5-terms model in *Eqn*. (3) [54, 55], the 6-terms model in *Eqn*. (4) [56, 57] and the 9-terms models in *Eqn*. (5) [58, 59].

$$f_5(x,y) = Ax^2 + By^2 + Cx + Dy + E \quad (3)$$

$$f_6(x,y) = Ax^2 + Bxy + Cy^2 + Dx + Ey + F \quad (4)$$

$$f_9(x,y) = Ax^2y^2 + Bx^2y + Cxy^2 + Dx^2 + Exy + Fy^2 + Gx + Hy + I \quad (5)$$

where the coefficients A, B, ..., I in *Eqn*. (3)-(5) are the parameters used to describe the error surface model, which can be obtained by curve fitting the rate distortion cost of the integer pixel motion estimation. In the following, we assume that the optimal pixel motion estimation position is P0 (0, 0), and the adjacent 8 integer pixel positions are P1(0, -1), P2(-1, 0), P3(0, 1), P4(1, 0), P5(-1, -1), P6(-1, 1), P7(1, 1), and P8(1, -1).

The optimal fractional-pixel position of the 5-terms model is easy to obtain because the 5-terms model can be decomposed independently into the quadratic polynomial of x and y. The coefficients A, B, C, D and E can be calculated from the rate-distortion cost of the estimated integer pixel motion estimation positions P0, P1, P2, P3 and P4. [54, 55] use the 5-terms model to predict the fractional-pixel motion estimation position as the starting search point, and exploit this point as the center for fine search until the optimal fractional-pixel motion estimation position is found. In order to improve the prediction accuracy of the 5-terms model, the *xy* term is introduced in the 6-terms model [56, 57] as the rotation factor, which can more accurately describe the error surface. In addition to the rate distortion cost of P0-P4, it is also necessary to select the rate distortion cost of a position from P5-P8 to calculate the coefficient of the 6-terms model. Dikbas [58] proposes a complete system model (CSM) to determine which of the P5-P8 is better suited for calculating model coefficients. To further improve the prediction accuracy of the 6-terms model, a 9-terms model is proposed to describe the error surface in more detail, resulting in higher encoding performance.

However, the use of the above parabolic equation to describe the error surface may meet a problem that a great influence will happen to the prediction accuracy of Parabolic Equations when an exceptional condition appear in one or several surrounding integer pixel position. To solve this problem, Suh [59] propose a method of calculating the horizontal coordinate x and the vertical coordinate y of the optimal fractional-pixel position independently and the optimal fractional-pixel position can be obtained by only three shift operations and four comparisons. But the computational process is too simplified, which results in greater coding performance loss.

In order to further improve the prediction accuracy of the optimal fractional-pixel position, Zhang [60] propose an algorithm to fit the valley curve of the error surface and then approximate the location with the minimum cost of the rate distortion on the error cost surface through the minimum location of the valley curve of the error surface. P6, P2, P5 and P3, P0, P1 and P7, P4, P8 are calculated respectively for the rate distortion cost of the integer pixel motion estimation to form the minimum position of the quadratic curve, then use these three positions to approximate the valley curve of the error surface and estimate the minimum location of the error surface. Although the method of estimating the optimal fractional-pixel position achieves better coding performance than the method of directly fitting error surface, there are some differences in the optimal integer pixel motion estimation positions between different integer pixel motion estimation algorithms, and the rate distortion cost of the surrounding integer pixel motion estimation are not always monotonically decreasing. In the case of the anomaly that may exist in the process of calculating the optimal fractional-pixel position for[60], Dai [61] proposes a method of modifying the anomaly of the intermediate result and improves the fractional-pixel motion estimation method in [60]. In [60] the quadratic curve may satisfy a convex function or a linear function, therefore the minimum value is limited to the range of -1 to 1 in [62] and the coding performance is further improved.

Although the interpolation-free algorithms in [60-62] can reduce the computational complexity of fractional pixel motion estimation to a great extent, there are still some problems. First, the three local minimum points are more

TABLE II. PERFORMANCE OF FAST FRACTIONAL-PIXEL MOTION ESTIMATION ALGORITHMS

Emphasis	Properties	Author, Year, Reference	Anchor	Configuration	Reported Performance			
					BD-Rate (%)	BD-PSNR(dB)	△ FME Time (%)	△Encoder Time (%)
Fast interpolation	Addptive pattern decision	Sotetsumoto 2013 [45]	HM 9.0	QP:20,24,28 Frame #: 30	3.119 (△Bit rate, kbit/s)[1]	-0.018 (△PSNR-Y, dB)[2]	51.334[3]	-
	Candidate position selection	Dai 2012 [50]	HM 3.0	Lowdelay-loco	-	-0.033 (△PSNR, dB)[4]	-	54.065[5]
		Dai 2012 [51]	HM 3.0	Lowdelay-loco	17.51 (△Bit rate, kbit/s)[6]	-0.00083 (△PSNR, dB)[7]	-	45.52[8]
	Reference spatio-temporal information	Jia 2016 [53]	HM 14.0	RA	0.02 （△Bit rate,%）	-0.01 (△ PSNR, dB)	40.86	24.56
Interpolation-free algorithms	error cost surface fitting	Zhang 2010 [60]	HM16.0[9]	LDP	4.30	-	~91	-
	Intermediate result correction	Dai 2013 [61]	HM 6.0	LDP	4.00	-	-	-
			HM16.0[9]	LDP	3.31	-	~91	-
	Precise search range	Zuo 2015 [62]	HM 11.0	LDP	3.4	-	91.1	-
			HM16.0[9]	LDP	2.95	-	~91	-
	multidirectional parabolic prediction	Fan 2017 [15]	HM16.0	LDP	2.42	-	~90	-

1. Averaged difference between the proposed bit rate and HM 9.0 bit rate in TABLE III of [45].
2. Averaged difference between the proposed PSNR and HM 9.0 PSNR in TABLE III of [45].
3. Average value of reduction rate in TABLE III of [45].
4. Average value of proposed △ PSNR in TABLE I of [50].
5. Average value of proposed reduced time in TABLE I of [50].
6. Averaged difference between the proposed bit rate and hierarchical search bit rate in TABLE I of [51].
7. Averaged difference between the proposed PSNR and hierarchical search PSNR in TABLE I of [51].
8. Hierarchical search total encoding minus proposed total encoding time and divided by hierarchical search total encoding in TABLE I of [51], and then seek the average.
9. This result is from Ref. [15] for fair comparison.

precise when they are derived in a direction perpendicular to the valley bottom trend of the error surface. However, the valley bottom trend of the error surface does not simply vary along the x-axis or y-axis direction; consequently, the horizontal and vertical directions supported in the above methods are insufficient to accommodate situations with other directions. Second, in most cases, the implicit assumption that the three minimum points are located along one straight line [60-62] does not hold, because the valley bottom of the error surface might not match a linear variation. Third, the determination criterion for the horizontal and vertical direction parabolas obtained by comparing the coding cost gradient is not accurate. Moreover, the high prediction accuracy of AMVP can be used to improve the encoding performance, but these above algorithms have not considered it. To overcome these problems, a multidirectional parabolic prediction-based interpolation-free fractional pixel motion estimation scheme is proposed in [15]. First, a multidirectional fractional pixel motion estimation with four different directional prediction patterns are proposed to better accommodate different valley bottom trends of the error surface. Then the valley bottom curve is decomposed by passing the three minimum points with two projection parabolas that better fit the distribution of the valley bottom curve.

The fast fractional-pixel ME algorithms are listed in Table II for clearer presentation and comparison.

In summary, the method of predicting the optimal fractional-pixels using the adjacent integer pixel rate distortion cost, among which [15, 60-62] are the state-of-the-art algorithms, can reduce much more coding complexity as compared to the algorithms in the first categories. However, the performance loss is relatively higher. How to accurately model the shape of error surface and improve the prediction accuracy might be one of the future focuses in interpolation-free fast fractional-pixel motion estimation algorithms.

V. CONCLUSIONS

In this paper, we presented a comprehensive review on the recent progress of motion estimation in inter coding of the new H.265/HEVC video coding standard. More specifically, after a review of the ME in HEVC, the recently developed fast algorithms on both the integral-pixel and fractional-pixel ME were reviewed and analyzed. Within each category, the algorithms were further subdivided into sub-categories according to mechanisms, and compared in terms of pros, cons, coding efficiency and coding complexity.

Through such a comprehensive review of the recent advances of fast algorithms of ME in the inter frame coding in

HEVC, hopefully it would provide valuable leads for the improvement of the ME. What's more important, at the current critical time of developing the next generation video coding standard beyond HEVC, further research addressing the problem of high efficiency low-complexity motion estimation are expected to achieve new breakthrough and make great impact for the ongoing development of the next generation video coding standard.

REFERENCES

[1] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 13, no. 7, pp. 560-576, 2003.

[3] F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC Complexity and Implementation Analysis," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 22, no. 12, pp. 1685-1696, Dec. 2012.

[4] J. Stankowski, D. Karwowski, K. Klimaszewski, K. Wegner, O. Stankiewicz, and T. Grajek, "Analysis of the complexity of the HEVC motion estimation," *International Conference on Systems, Signals And Image Processing, (IWSSIP)*, 2016.

[5] J. R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the Coding Efficiency of Video Coding Standards-Including High Efficiency Video Coding (HEVC)," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 22, no. 12, pp. 1669-1684, 2012.

[6] Y. Zhang, H. Wang, and Z. Li, "Fast Coding Unit Depth Decision Algorithm for Interframe Coding in HEVC," *IEEE Data Compression Conference (DCC)*, 2013.

[7] I. Zupancic, S. G. Blasi, E. Peixoto, and E. Izquierdo, "Inter-Prediction Optimizations for Video Coding Using Adaptive Coding Unit Visiting Order," *IEEE Transactions on Multimedia,* vol. 18, no. 9, pp. 1677-1690, Sep. 2016.

[8] Y. Zhang, Z. Li, and B. Li, "Gradient-Based Fast Decision for Intra Prediction in Hevc," *IEEE Visual Communications and Image Processing (VCIP 2012),* pp. 1-6, 2012.

[9] W. Xiao, B. Li, J. Z. Xu, G. M. Shi, and F. Wu, "HEVC Encoding Optimization Using Multicore CPUs and GPUs," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 25, no. 11, pp. 1830-1843, Nov. 2015.

[10] R. Fan, Y. Zhang, and B. Li, "Motion Classification-Based Fast Motion Estimation for High-Efficiency Video Coding," *IEEE Transactions on Multimedia,* vol. 19, no. 5, pp. 893-907, May. 2017.

[11] N. Hu and E. H. Yang, "Fast Motion Estimation Based on Confidence Interval," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 24, no. 8, pp. 1310-1322, Aug. 2014.

[12] (2018, Apr.). *Versatile Video Coding*. Available: https://mpeg.chiariglione.org/standards/mpeg-i/versatilevideo-coding

[13] I. K. Kim, K. McCann, K. Sugimoto, B. Bross, and W. J. Han, "High Efficiency Video Coding (HEVC) Test Model 10 (HM 10) Encoder Description," *Joint Collaborative Team on Video Coding meeting, JCTVC-L1002,* 2013.

[14] H. Lv, R. Wang, X. Xie, H. Jia, and W. Gao, "A comparison of fractional-pel interpolation filters in HEVC and H.264/AVC," *IEEE Visual Communications and Image Processing(VCIP)*, 2012.

[15] R. Fan, Y. Zhang, B. Li, and G. Wang, "Multidirectional parabolic prediction-based interpolation-free sub-pixel motion estimation," *Signal Processing: Image Communication,* vol. 53, pp. 123-134, Apr. 2017.

[16] H. Chung-Lin and H. Chao-Yuen, "A new motion compensation method for image sequence coding using hierarchical grid interpolation," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 4, no. 1, pp. 42-52, Jan. 1994.

[17] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 6, no. 3, pp. 313-317, Mar. 1996.

[18] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Transactions on Image Processing,* vol. 9, no. 2, pp. 287-290, Feb. 2000.

[19] C. Zhu, X. Lin, and L. P. Chau, "Hexagon-based search pattern for fast block motion estimation," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 12, no. 5, pp. 349-355, May. 2002.

[20] C. H. Cheung and L. M. Po, "Novel cross-diamond-hexagonal search algorithms for fast block motion estimation," *IEEE Transactions on Multimedia,* vol. 7, no. 1, pp. 16-22, Feb. 2005.

[21] N. Parmar and M. H. Sunwoo, "Enhanced Test Zone Search Motion Estimation Algorithm for HEVC," *2014 International Soc Design Conference (Isocc)*, 2014.

[22] J. H. Jeong, N. Parmar, and M. H. Sunwoo, "Enhanced Test Zone Search Algorithm With Rotating Pentagon Search," *International Soc Design Conference (ISOCC)*, 2015.

[23] S. H. Yang, J. Z. Jiang, and H. J. Yang, "Fast motion estimation for HEVC with directional search," *Electronics Letters,* vol. 50, no. 9, pp. 673-674, Apr. 2014.

[24] L. F. Gao, S. F. Dong, W. M. Wang, R. G. Wang, and W. Gao, "A Novel Integer-Pixel Motion Estimation Algorithm Based on Quadratic Prediction," *IEEE International Conference on Image Processing (ICIP)*, 2015.

[25] Y. H. Ko, H. S. Kang, and S. W. Lee, "Adaptive Search Range Motion Estimation Using Neighboring Motion Vector Differences," *IEEE Transactions on Consumer Electronics,* vol. 57, no. 2, pp. 726-730, May. 2011.

[26] W. Dai, O. C. Au, S. J. Li, L. Sun, and R. B. Zou, "Adaptive Search Range Algorithm Based on Cauchy Distribution," *IEEE International Conference on Visual Communications And Image Processing (VCIP)*, 2012.

[27] L. Q. Shen and Z. Y. Zhang, "Content-Adaptive Motion Estimation Algorithm for Coarse-Grain SVC," *IEEE Transactions on Image Processing,* vol. 21, no. 5, pp. 2582-2591, May. 2012.

[28] Z. T. Liao and C. A. Shen, "A Novel Search Window Selection Scheme for the Motion Estimation of HEVC Systems," *International Soc Design Conference (ISOCC)*, 2015.

[29] Y. Ismail, J. B. McNeely, M. Shaaban, H. Mahmoud, and M. A. Bayoumi, "Fast Motion Estimation System Using Dynamic Models for H.264/AVC Video Coding," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 22, no. 1, pp. 28-42, Jan. 2012.

[30] Y. H. Moon, K. S. Yoon, S. T. Park, and I. H. Shin, "A New Fast Encoding Algorithm Based on an Efficient Motion Estimation Process for the Scalable Video Coding Standard," *IEEE Transactions on Multimedia,* vol. 15, no. 3, pp. 477-484, Apr. 2013.

[31] P. Nalluri, L. N. Alves, and A. Navarro, "Complexity Reduction Methods for Fast Motion Estimation in HEVC," *Signal Processing-Image Communication,* vol. 39, pp. 280-292, Nov. 2015.

[32] A. Medhat, A. Shalaby, M. S. Sayed, M. Elsabrouty, and F. Mehdipour, "Adaptive low-complexity motion estimation algorithm for high efficiency video coding encoder," *IET Image Processing,* vol. 10, no. 6, pp. 438-447, Jun. 2016.

[33] J. Luo, X. H. Yang, and L. H. Liu, "A fast motion estimation algorithm based on adaptive pattern and search priority," *Multimedia Tools And Applications,* vol. 74, no. 24, pp. 11821-11836, Dec. 2015.

[34] Z. Q. Pan, J. J. Lei, Y. Zhang, X. M. Sun, and S. Kwong, "Fast Motion Estimation Based on Content Property for Low-Complexity H.265/HEVC Encoder," *IEEE Transactions on Broadcasting,* vol. 62, no. 3, pp. 675-684, Sep. 2016.

[35] W. Li and E. Salari, "Successive Elimination Algorithm for Motion Estimation," *IEEE Transactions on Image Processing,* vol. 4, no. 1, pp. 105-107, Jan. 1995.

[36] X. Q. Gao, C. J. Duanmu, and C. R. Zou, "A Multilevel Successive Elimination Algorithm for Block Matching Motion Estimation," *IEEE Transactions on Image Processing,* vol. 9, no. 3, pp. 501-504, Mar. 2000.

[37] Y. W. Huang, S. Y. Chien, B. Y. Hsieh, and L. G. Chen, "Global Elimination Algorithm and Architecture Design for Fast Block Matching Motion Estimation," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 14, no. 6, pp. 898-907, Jun. 2004.

[38] N. Hu and E. H. Yang, "Confidence Interval Based Motion Estimation," *IEEE International Conference on Image Processing(ICIP)*, 2013.

[39] D. Zhang, G. Cao, and X. Gu, "Improved motion estimation based on motion region identification," *International Conference on Systems and Informatics*, 2012.

[40] M. C. Chi, C. H. Yeh, and M. J. Chen, "Robust Region-of-Interest Determination Based on User Attention Model Through Visual Rhythm Analysis," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 19, no. 7, pp. 1025-1038, Jul. 2009.

[41] C. L. Guo and L. M. Zhang, "A Novel Multiresolution Spatiotemporal Saliency Detection Model and Its Applications in Image and Video Compression," *IEEE Transactions on Image Processing,* vol. 19, no. 1, pp. 185-198, Jan. 2010.

[42] L. L. Lin, I. C. Wey, and J. H. Ding, "Fast predictive motion estimation algorithm with adaptive search mode based on motion type classification," *Signal Image And Video Processing,* vol. 10, no. 1, pp. 171-180, Jan. 2016.

[43] Y. Zhang and T. Z. Shen, "Motion information based adaptive block classification for fast motion estimation," *International Conference on Neural Networks and Signal Processing*, 2008.

[44] X. F. Li, R. G. Wang, X. L. Cui, and W. M. Wang, "Context-Adaptive Fast Motion Estimation of HEVC," *IEEE International Symposium on Circuits And Systems (ISCAS)*, 2015.

[45] T. Sotetsumoto, T. Song, and T. Shimamoto, "Low complexity algorithm for sub-pixel motion estimation of HEVC," *IEEE International Conference on Signal Processing, Communication and Computing*, 2013.

[46] N. T. Ta and J. R. Choi, "High Performance Fractional Motion Estimation in H.264/AVC Based on One-step Algorithm and 8×4 Element Block Processing," *Signal Processing-Image Communication,* vol. 26, no. 2, pp. 85-92, 2011.

[47] P. R. Hill and D. R. Bull, "Sub-pixel motion estimation using kernel methods," *Signal Processing-Image Communication,* vol. 25, no. 4, pp. 268-275, Apr. 2010.

[48] Z. B. Chen, C. Du, J. H. Wang, and Y. He, "PPFPS - A paraboloid prediction based fractional pixel search strategy for H.26L," *IEEE International Symposium on Circuits And Systems(ISCAS)*, 2002.

[49] C. Du, Y. He, and J. L. Zheng, "PPHPS: A parabolic prediction-based, fast half-pixel search algorithm for very low bit-rate moving-picture coding," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 13, no. 6, pp. 514-518, Jun. 2003.

[50] W. Dai, O. C. Au, C. Pang, L. Sun, R. B. Zou, and S. J. Li, "A Novel Fast Two Step Sub-Pixel Motion Estimation Algorithm In HEVC," *IEEE International Conference on Acoustics, Speech And Signal Processing (ICASSP)*, 2012.

[51] W. Dai, O. C. Au, S. J. Li, L. Sun, and R. B. Zou, "Fast Sub-Pixel Motion Estimation with Simplified Modeling in HEVC," *IEEE International Symposium on Circuits And Systems(ISCAS)*, 2012.

[52] H. Lee, B. Jung, J. Jung, and B. Jeon, "Fast subpel motion estimation for H.264/advanced video coding with an adaptive motion vector accuracy decision," *Optical Engineering,* vol. 51, no. 11, Nov. 2012.

[53] S. Jia, W. P. Ding, Y. H. Shi, and B. C. Yin, "A Fast Sub-Pixel Motion Estimation Algorithm For HEVC," *IEEE International Symposium on Circuits And Systems(ISCAS)*, 2016.

[54] J. F. Chang and J. J. Leou, "A quadratic prediction based fractional-pixel motion estimation algorithm for H.264," *IEEE International Symposium on Multimedia(ISM)*, 2005.

[55] J. F. Chang and I. J. Leou, "A quadratic prediction based fractional-pixel motion estimation algorithm for H.264," *Journal Of Visual Communication And Image Representation,* vol. 17, no. 5, pp. 1074-1089, Oct. 2006.

[56] T. K. Chiew, J. T. H. Chung-How, D. R. Bull, and C. N. Canagarajah, "Interpolation-free subpixel refinement for block-based motion estimation," *IEEE International Conference on Visual Communications And Image Processing*, 2004.

[57] P. R. Hill, T. K. Chiew, D. R. Bull, and C. N. Canagarajah, "Interpolation free subpixel accuracy motion estimation," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 16, no. 12, pp. 1519-1526, Dec. 2006.

[58] S. Dikbas, T. Arici, and Y. Altunbasak, "Fast Motion Estimation With Interpolation-Free Sub-Sample Accuracy," *IEEE Transactions on Circuits And Systems for Video Technology,* vol. 20, no. 7, pp. 1047-1051, Jul. 2010.

[59] J. W. Suh, J. Cho, and J. Jeong, "Model-based quarter-pixel motion estimation with low computational complexity," *Electronics Letters,* vol. 45, no. 12, pp. 618-619, Jun. 2009.

[60] Q. Zhang, Y. Y. Dai, and C. C. J. Kuo, "Direct Techniques for Optimal Sub-Pixel Motion Accuracy Estimation and Position Prediction," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 20, no. 12, pp. 1735-1744, Dec. 2010.

[61] W. Dai, O. C. Au, W. J. Zhu, W. Hu, P. F. Wan, and J. L. Li, "A Robust Interpolation-Free Approach for Sub-Pixel Accuracy Motion Estimation," *IEEE International Conference on Image Processing(ICIP)*, 2013.

[62] X. G. Zuo and L. Yu, "A Novel Interpolation-free Scheme for Fractional Pixel Motion Estimation," *IEEE Picture Coding Symposium (PCS) with Packet Video Workshop (PV)*, 2015.