Fast Sub-Pixel Motion Estimation with Simplified Modeling in HEVC

Wei Dai, Oscar C. Au, Sijin Li, Lin Sun, Ruobing Zou
Department of Electronic and Computer Engineering
HKUST, Clear Water Bay, Kowloon, Hong Kong
{weidai, eeau, sliae, lsunece, zouruobing}@ust.hk

***Abstract*—Motion estimation (ME) is one of the key elements in video coding standard which eliminates the temporal redundancies between successive frames. In recent international video coding standards, sub-pixel ME is proposed for its excellent coding performance. Compared with integer-pixel ME, sub-pixel ME needs interpolation to get the value in sub-pixel position. Also, Hadamard transform will be applied in order to achieve better performance. Therefore, it is becoming more and more critical to develop fast sub-pixel ME algorithms. In this paper, a novel fast sub-pixel ME algorithm is proposed which makes full use of 8 neighboring integer-pixel points. This algorithm models the error surface in sub-pixel position by a second order function with five parameters two times to predict the best sub-pixel position. Experimental results show that the proposed method can reduce the complexity significantly with negligible quality degradation.**

I. INTRODUCTION

Motion estimation (ME) is one of the most important parts in video coding standard which is dedicated to achieve high coding performance by reducing temporal redundancies. Compared with other ME methods, block-based matching algorithm (BMA) attracts widely attention because a complex motion can be analyzed as a translational motion and its performance is acceptable compared to other complex motion models. The most simple and straightforward way to find the best position is the full search (FS) algorithm which checks every position among all the candidates and chooses the best one based on the rate-distortion (RD) performance. Although the FS approach can reach the global minimum, the computational complexity incurred is usually unaffordable. To relax the computational burden, the ME process is often divided into two steps: First, integer-pixel points within a search range are checked to find the best integer-pixel position. Then, sub-pixel points around the best integer-pixel position are examined to select the best sub-pixel position.

In order to accelerate the ME process, a lot of fast ME algorithms have been proposed. Typically, these algorithms can be classified into two categories: fast integer-pixel ME algorithm and fast sub-pixel ME algorithm. Algorithms like three step search (TSS) [1], new three step search (NTSS) [2], PMVFAST [3], E-PMVFAST [4] all belong to the first category which speeds up the process of integer-pixel ME. Generally speaking, fast integer-pixel ME will be terminated within 10 points search. However, for traditional hierarchical sub-pixel ME, 16 points are needed for quarter-pixel accuracy. Besides, interpolation is required to get the value for those sub-pixel positions, which means the computational complexity of sub-pixel ME becomes comparable to that of fast integer-pixel ME.

One way to do fast sub-pixel ME is to model the error surface in the locality of the best integer-pixel point. Jing-Fu Chang *et al.* [5] modeled the error surface as a unimodal and proposed a second order function with five parameters to approximate the error surface. The best sub-pixel position was obtained by minimizing the second order function followed with some refinement process. In [6], the error surface was approximated by a function with six parameters, and the best sub-pixel position was found through a simple 4-connected gradient descent search. Salih Dikbas *et al.* [7] introduced a function with nine parameters to model the error surface, and the best sub-pixel position was located by finding the minimum position of the function.

The rest of this paper is organized as follows. In Section II, the existing research on fast sub-pixel ME is reviewed and a novel fast sub-pixel ME algorithm is proposed based on the previous study. To evaluate the performance of the proposed algorithm, it is compared with several representative methods, and the experimental results are shown in Section III. Finally, Section IV concludes the paper.

II. PROPOSED FAST SUB-PIXEL ME ALGORITHM

Several surface models have been used to approximate the error surface in sub-pixel ME process, including the 9-term, 6-term and 5-term error models. Mathematically, they can be written as follows:

$$\begin{aligned} f_9(x,y) &= Ax^2y^2 + Bx^2y + Cxy^2 + Dx^2 \\ &\quad + Exy + Fy^2 + Gx + Hy + I \qquad (1) \\ f_6(x,y) &= Ax^2 + Bxy + Cy^2 + Dx + Ey + F \qquad (2) \\ f_5(x,y) &= Ax^2 + Bx + Cy^2 + Dy + E, \qquad (3) \end{aligned}$$

where parameters $A, B, \ldots, I$ are estimated by fitting the RD cost of integer-pixel points on the given models. The RD cost is defined as:

$$RDCost = SAD(m,n) + \lambda R(m,n), \qquad (4)$$

where $SAD(m,n)$ is the SAD value with respect to the motion vector (m,n), $R(m,n)$ is the cost of the motion vector and

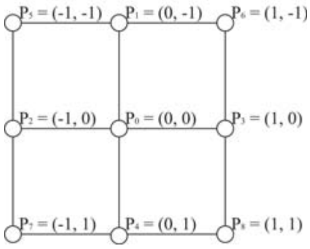

Fig. 1. Illustration of the locations of 9 integer-pixel points.

λ is the *Lagrange multiplier*. The locations of 9 integer-pixel points are given in Fig. 1. Note that the contour of surface model (2) corresponds to a rotated $2-D$ ellipse while the contour of (3) corresponds to a simple ellipse whose axes align well with the x and y axes.

In (1)(2) and (3), different number of parameters are used, and all the parameters can be obtained with only addition and bit shift operations [5]-[7]. Nine points are required to model the error surface in (1), so the minimization of (1) is quite complicated. For (2), it uses P_0, P_1, P_2, P_3, P_4 and one additional point from P_5, P_6, P_7, P_8 to approximate the error surface, which means criteria is needed to select one point out of those four points. Moreover, the minimum positions of (1) and (2) are not guaranteed to be within the $(-1,1)\times(-1,1)$ area. Therefore, some techniques are adopted to find the minimum position such as exhaustive search or gradient descent search [6], but both of those methods need a large number of multiplications which is very time consuming. In contrast, (3) is very simple which only needs five points to model the error surface. Thus, the minimization is very easy and the location of the minimum position can be calculated as:

$$\begin{cases} x_{min} = -B/2A \\ y_{min} = -D/2C. \end{cases} \tag{5}$$

Because more information can provide better prediction, so in this paper, all the 8 neighboring integer-pixel points are considered. In order to make full use of the information in 8 neighboring points without too many multiplications, a novel fast sub-pixel ME algorithm is proposed which utilizes the properties of (3) to simplify the modeling and minimization process. First, 8 neighboring integer-pixel points are divided into two groups. Then, the points in each group together with the best integer-pixel point are used to model the error surface. Two minimum positions are obtained and checked based on the RD cost. Moreover, a point lies between the previous two minimum positions will be checked for further refinement. The flowchart of the algorithm is illustrated in Fig. 2 and details are discussed below.

A. 0° Approximation

The first step of the algorithm is the same as [5] which use five points P_0, P_1, P_2, P_3, P_4 to model the error surface:

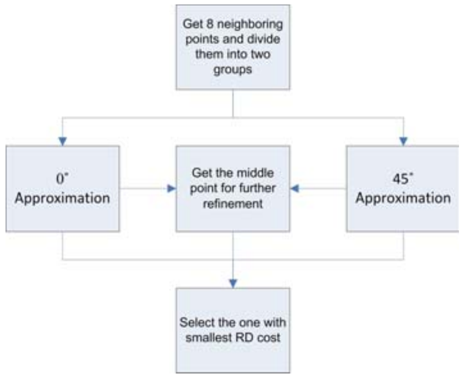

Fig. 2. Flowchart of the proposed algorithm.

$$\begin{cases} f_5(P_0) = E_0 \\ f_5(P_2) = A_0 - B_0 + E_0 \\ f_5(P_3) = A_0 + B_0 + E_0 \\ f_5(P_1) = C_0 - D_0 + E_0 \\ f_5(P_4) = C_0 + D_0 + E_0, \end{cases} \tag{6}$$

where A_0, B_0, C_0, D_0, E_0 can be solved with only addition and bit shift operations [5]. The first predicted best sub-pixel position can be obtained by:

$$\begin{cases} x_{0^\circ} = -B_0/2A_0 \\ y_{0^\circ} = -D_0/2C_0. \end{cases} \tag{7}$$

Let $Quantize(x)$ be the operation of quantizing x to the nearest sub-pixel position with predefined accuracy. First, $(x_{0^\circ}, y_{0^\circ})$ is quantized to the predefined accuracy:

$$\begin{cases} x_{0^\circ}^Q = Quantize(x_{0^\circ}) \\ y_{0^\circ}^Q = Quantize(y_{0^\circ}). \end{cases} \tag{8}$$

B. 45° Approximation

The second step of the algorithm is to model the error surface by another four points P_5, P_6, P_7, P_8 together with the best integer-pixel point P_0. A new second order function is introduced:

$$\begin{aligned} f_5'(x,y) =& A_{45}(x+y)^2 + B_{45}(x+y) \\ &+ C_{45}(x-y)^2 + D_{45}(x-y) + E_{45}. \end{aligned} \tag{9}$$

Note that the contour of (9) is an ellipse whose axes align well with rotated x and y axes by 45°. For simplicity, we rotate the $x-y$ plane by 45° and map each point to the rotated plane according to Fig. 3. The new function becomes similar as it in the previous step:

$$\begin{aligned} f_5'^T(x^T, y^T) =& A_{45}^T x^{T2} + B_{45}^T x^T \\ &+ C_{45}^T y^{T2} + D_{45}^T y^T + E_{45}^T. \end{aligned} \tag{10}$$

Mathematically, this transformation can be written as:

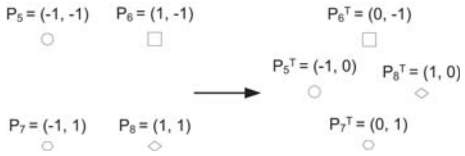

Fig. 3. Illustration of point correspondences before and after transform.

$$\begin{cases} x^T = (x+y)/2 \\ y^T = (y-x)/2. \end{cases} \tag{11}$$

The minimum position of (10) is:

$$\begin{cases} x^T_{45^\circ} = -B^T_{45}/2A^T_{45} \\ y^T_{45^\circ} = -D^T_{45}/2C^T_{45}. \end{cases} \tag{12}$$

After the minimum position in the transformed space is obtained, the corresponding point in the original space can be calculated by the inverse transformation of (11):

$$\begin{cases} x_{45^\circ} = x^T_{45^\circ} - y^T_{45^\circ} \\ y_{45^\circ} = x^T_{45^\circ} + y^T_{45^\circ}. \end{cases} \tag{13}$$

Also quantization is applied to the point in the original space with predefined accuracy:

$$\begin{cases} x^Q_{45^\circ} = Quantize(x_{45^\circ}) \\ y^Q_{45^\circ} = Quantize(y_{45^\circ}). \end{cases} \tag{14}$$

C. Refinement Process

In the previous two steps, the error surface is modeled by two second order functions. The contours of those two functions are two ellipses. One ellipse's axes align well with the x and y axes and another is a $45°$ rotation of the first one. But the error surface for real situations will not just happen in these two cases. Actually, the rotation angle of the real ellipse will be any number between $[0°, 90°]$. So further refinement procedure should be taken to get a better prediction.

In Fig. 4, an example of a rotated ellipse is used to show the relationship between the global minimum position and two positions which are obtained in the previous two steps. It can be observed that the real best sub-pixel position lies somewhere between $(x_{0^\circ}, y_{0^\circ})$ and $(x_{45^\circ}, y_{45^\circ})$. By extensively experimental observation, this is also true for most of the real situations. Furthermore, taking quantization error into consideration, the point lies in the middle of $(x_{0^\circ}, y_{0^\circ})$ and $(x_{45^\circ}, y_{45^\circ})$ is chosen to be the refined sub-pixel position.

$$\begin{cases} x_{mid} = (x_{0^\circ} + x_{45^\circ})/2 \\ y_{mid} = (y_{0^\circ} + y_{45^\circ})/2. \end{cases} \tag{15}$$

We denote (x^Q_{mid}, y^Q_{mid}) as the quantized position of (x_{mid}, y_{mid}).

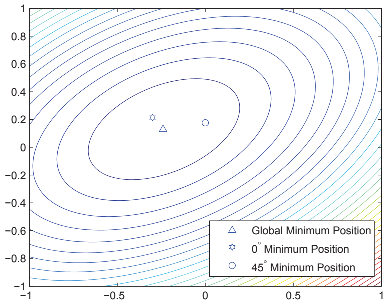

Fig. 4. Illustration of relationship between global minimum point and two predicted points in the previous steps with arbitrary θ.

D. Compare and Get the Optimal Position

Finally, comparison is taken among $(x^Q_{0^\circ}, y^Q_{0^\circ})$, $(x^Q_{45^\circ}, y^Q_{45^\circ})$ and (x^Q_{mid}, y^Q_{mid}). The one with minimum RD cost will be chosen as the final motion vector.

III. Experiment Results

The proposed algorithm has been implemented on the latest HEVC reference software HM3.0, and the encoder is set to be lowdelay-loco mode. To evaluate the performance of proposed algorithm, hierarchical search is chosen to be the anchor algorithm. Moreover, method in [5] (FPME) is selected as the representative algorithm of fast sub-pixel ME. The reason to choose FPME is that FPME and the proposed method are quite similar. The major difference is: FPME uses only four neighboring integer-pixel points and performs the refined search iteratively after the prediction. However, the proposed method uses eight neighboring integer-pixel points and does the approximation twice followed by one single refinement process.

Three sequences: ParkScene (1080p), Vidyo1 (720p) and BasketballDrill (WVGA) are tested with four QP values. PSNR, bit rate, total encoding time and average sub-pixel search points per partition (SP/PT) are measured. Experimental results in Table I show that the proposed method achieves significant encoding time reduction over the other two methods at the expense of negligible coding performance degradation. Moreover, SP/PT shows that the proposed algorithm can greatly reduce the number of sub-pixel search points compared to the conventional fast sub-pixel ME algorithm. Less than 2 points are required by the proposed algorithm on the average while usually more than 6 points are needed for FPME. This means that our algorithm can predict the best sub-pixel position successfully with a high probability.

IV. Conclusion

In this paper, a new fast sub-pixel ME algorithm is presented. The proposed algorithm makes full use of 8 neighboring points' information and approximates the true error surface twice by a second order function with five parameters. Two predicted positions are obtained by $0°$ approximation and $45°$

TABLE I

COMPARISON OF THE PROPOSED METHOD WITH FPME AND HIERARCHICAL SEARCH METHOD.

Sequence Name	QP	Method	PSNR (dB)	Bit Rate (kb/s)	Total Encoding Time (s)	SP/PT*
ParkScene (1080P)	22	Hierarchical Search	39.82	8900.73	13605.20	16.00
		FPME	39.82	8927.60	11818.94	6.65
		Proposed	39.82	8975.08	9621.90	1.88
	27	Hierarchical Search	36.91	3589.12	12201.56	16.00
		FPME	36.91	3602.00	10388.30	6.53
		Proposed	36.90	3619.66	8142.70	1.69
	32	Hierarchical Search	34.14	1515.94	11398.79	16.00
		FPME	34.14	1520.86	9507.15	6.29
		Proposed	34.14	1527.35	7260.01	1.55
	37	Hierarchical Search	31.58	647.42	10906.23	16.00
		FPME	31.57	647.89	8874.71	5.97
		Proposed	31.57	650.30	6700.02	1.46
Vidyo1 (720P)	22	Hierarchical Search	43.25	2235.76	20823.83	16.00
		FPME	43.25	2246.58	9084.29	6.06
		Proposed	43.24	2261.42	7146.52	1.59
	27	Hierarchical Search	40.97	823.36	19757.09	16.00
		FPME	40.96	827.79	8572.87	5.89
		Proposed	40.96	830.96	6374.80	1.37
	32	Hierarchical Search	38.48	391.19	18890.51	16.00
		FPME	38.48	392.76	8049.89	5.66
		Proposed	38.48	393.48	6006.04	1.27
	37	Hierarchical Search	35.75	208.81	17477.18	16.00
		FPME	35.75	208.71	7685.68	5.49
		Proposed	35.74	209.74	5895.88	1.30
BasketballDrill (WVGA)	22	Hierarchical Search	39.98	4318.73	5367.79	16.00
		FPME	39.98	4325.86	4447.41	6.85
		Proposed	39.98	4351.99	3760.12	2.02
	27	Hierarchical Search	36.72	2029.13	4777.00	16.00
		FPME	36.72	2034.07	3916.23	6.86
		Proposed	36.71	2043.51	3177.00	1.84
	32	Hierarchical Search	33.87	959.53	4468.40	16.00
		FPME	33.86	961.43	3644.42	6.67
		Proposed	33.86	963.63	2800.99	1.65
	37	Hierarchical Search	31.37	469.77	4234.31	16.00
		FPME	31.36	470.90	3308.05	6.21
		Proposed	31.34	472.50	2532.96	1.49

* Average sub-pixel search points per partition.

approximation. The middle point of $(x_{0^\circ}, y_{0^\circ})$ and $(x_{45^\circ}, y_{45^\circ})$ is chosen to be the refined position. The point with the minimum RD cost among those three sub-pixel points will be selected as the final motion vector. Experimental results show that the proposed algorithm greatly reduces the number of average search points while maintaining the coding performance compared to the conventional hierarchical search algorithm.

V. ACKNOWLEDGEMENT

This work has been supported in part by the Research Grants Council (GRF Project no. 610210) and the Hong Kong Applied Science and Technology Research Institute Project (ART/093CP).

REFERENCES

[1] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," in *Proc. Nat. Telecommun. Conf.*, New Orleans, LA, Nov. 29ĺCDec. 3 1981, pp. G5.3.1ĺCG5.3.5.

[2] R. Li, B. Zeng, and M. Liou, "A new three-step search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no. 4, pp. 438ĺC442, Aug. 1994.

[3] A. M. Tourapis, O. C. Au, and M. L. Liou, "Predictive Motion Vector Field Adaptive Search Technique (PMVFAST)," in *ISO/IEC JTC1/SC29/WG11 MPEG2000, Noordwijkerhout*, NL, March 2000.

[4] Hoi-Ming Wong, O. C. Au, Chi-Wang Ho and Shu-Kei Yip, "Enhanced predictive motion vector field adaptive search technique (E-PMVFAST)-based on future MV prediction," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, 2005, pp. 4 pp.

[5] Jing-Fu Chang and Jin-Jang Leou, "A quadratic prediction based fractional-pixel motion estimation algorithm for H.264," in *Multimedia, Seventh IEEE International Symposium on*, 2005, pp. 8 pp.

[6] Hill, P.R.; Chiew, T.K.; Bull, D.R.; Canagarajah, C.N.; , "Interpolation Free Subpixel Accuracy Motion Estimation," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol.16, no.12, pp.1519-1526, Dec. 2006

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