Enhanced Hexagonal Search for Fast Block Motion Estimation

Ce Zhu, Xiao Lin, Lap-Pui Chau, Lai-Man Po

***Abstract*— Fast block motion estimation normally consists of low-resolution coarse search and the following fine-resolution inner search. Most motion estimation algorithms developed attempt to speed up the coarse search without considering accelerating the focused inner search. On top of the hexagonal search method recently developed, an enhanced hexagonal search algorithm is proposed to further improve the performance in terms of reducing number of search points and distortion, where a novel fast inner search is employed by exploiting the distortion information of the evaluated points. Our experimental results substantially justify the merits of the proposed algorithm.**

***Index Terms*— Fast search algorithm, block motion estimation, hexagonal search.**

I. Introduction

IN In recent years, many fast block motion estimation algorithms have been proposed to alleviate the computation burden in a video encoder. The two-dimensional logarithmic search algorithm [1], three-step search (TSS), new three-step search (NTSS) [2], four-step search (4SS) [3], block-based gradient descent search (BBGDS) [4], simple and efficient search (SES) [5], diamond search (DS) [6-7] algorithms are amongst the class of fast search methods by reducing the number of search points in the process of block motion estimation. More recently, the hexagonal search or hexagon-based search (HEXBS) [10] has shown the significant improvement over other fast algorithms such as DS. In contrast with the DS that uses a diamond search pattern, the HEXBS adopts a hexagonal search pattern to achieve faster processing due to fewer search points being evaluated. The motion estimation process normally comprises two steps, namely, the low-resolution coarse search to identify a small area where the best motion vector is expected to lie, and the following fine-resolution inner search to select the best motion vector in the located small region. Most fast algorithms focus on speeding up the coarse search by taking various smart ways to reduce the number of search points in identifying a small area for inner search. The following focused inner search normally employs the exhaustive full search in the small area. Therefore, a fast inner search is desirable to achieve overall minimum number of search points, especially when there is no much room for further accelerating the coarse search after much effort have been invested on obtaining the fastest coarse search, while maintaining the distortion as small as possible.

A hexagonal search pattern is depicted in Fig. 1 and the inner search for the HEXBS algorithm is carried out using the shrunk hexagonal pattern covering the point 2, 4, 5, 7. In [10], a one-more-step HEXBS is also presented to cover a more pair of points of the four other points inside the hexagonal search pattern, i.e., point 1 and 3 if point 2 wins in the last step of the HEXBS algorithm, point 1 and 6 if point 4 wins, point 3 and 8 if point 5 wins, point 6 and 8 if point 7 wins. Note that a complete inner search within the large hexagon would require the evaluation of all the eight patterned points inside, excluding the center point. Hence, a more efficient inner search is desired to speed up the inner search. Here we reasonably assume that the nearer to the global minimum the smaller the distortion within a small neighborhood of the global minimum. That is, the error distortion function has monotonic characteristic statistically in a localized search area. The locally unimodal error surface assumption applies in the localized inner search, where the minimum distortion is searched within the large hexagon region, given the current minimum distortion in the center of the large hexagon and the distortions of the six checked points along the perimeter of the large hexagon. In our opinion, appropriate utilization of the surrounding information would yield a faster and yet accurate inner search.

It is also well known that the motion vector of the current block is highly correlated to those of its neighboring blocks. By simply exploiting the continuity in the motion vector field, the motion information of neighboring blocks can be utilized for prediction of a good starting point [8, 9]. Here we consider the upper and the left neighboring blocks by evaluating the two motion vectors of the two neighboring blocks compared with zero motion vector in terms of distortion. The motion vector with the smallest distortion is found to be the starting point to perform the hexagonal search, which is termed as the predictive HEXBS method. With a good prediction as the starting point, the predictive HEXBS method normally finds better motion vectors than the original HEXBS scheme that may get trapped in local optima starting from zero motion vector, thus decreasing the distortion.

The remainder of the letter is organized as follows. In the following section, a novel 6-side-based fast inner search is proposed and the experimental results showing the efficiency of the proposed fast inner search are presented in Section III. Section IV concludes the letter.

II. Fast Hexagonal Inner Search

In the original hexagonal search (HEXBS) algorithm [10], two search procedures are involved. The coarse search procedure firstly locates a region where the optimal motion vector is expected to lie, using the large hexagon search pattern consisting of 6 endpoints in Fig.1. The coarse search continues based on a gradient scheme until the center point of the hexagon has the current smallest distortion. After a hexagonal

Manuscript received September 2002; revised August 2003.

C. Zhu and L.P. Chau are with the School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore.

X. Lin is with the Institute for Infocom Research, Singapore

L.M. Po is with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong.

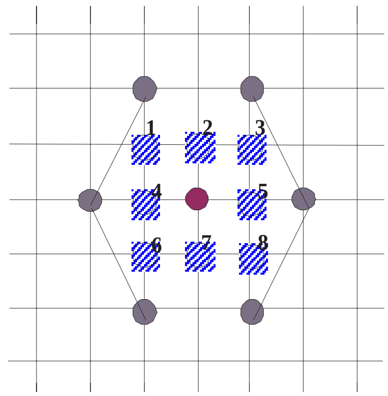


Fig. 1. The inner points in the hexagonal search pattern

area is located in the coarse search, then the following fine-resolution search looks into the small area enclosed by the large hexagon for focused inner search using the shrunk hexagon pattern, which is not a full inner search. As shown in Fig. 1, if full search is required for the inner search, eight search points inside the large hexagon will be evaluated, which is computationally inefficient. Interestingly, we find that strong correlation exists between the inner search points to be checked (i.e., the eight labeled points in Fig.1) and their surrounding checked points (here the 6 endpoints of the hexagon in Fig.1). Based on the monotonic distortion characteristic in the localized area around the global minimum, we propose to check only a portion of the inner search points that are nearer to the checked points with smaller distortions, which can save more than half of the eight search points inside. In the following we will present such an efficient inner search scheme by exploiting the distortion information of the six checked endpoints of the large hexagon. There can be several different ways to exploit the distortion information. By trial and error we find the most efficient inner search method in terms of minimizing both number of search points evaluated and the corresponding distortion. The method is referred to as 6-side-based fast inner search, which is found to be most reliable and robust among some other variants we tried in maintaining almost the same distortion as the full inner search.

A. 6-Side-Based Fast Inner Search

We consider grouping the search points in the six sides of the hexagon, resulting in 6 groups (pairs) of points, as shown in Fig. 2. For each group, we define a group distortion by summing the distortions of all the points within the group. The area near to the group with the smallest group distortion is considered as the region in which the minimum distortion is most likely to be found. Therefore we focus the inner search just in the region near to the group with the smallest group distortion. For different groups (sides) in different locations, we have different number of inner search points, as shown in Fig. 2(a) and Fig. 2(b) respectively. Two or three search points are used in the focused inner search, depending on the position of the group. Three inner points closest to the Group 2 or Group 5 will be evaluated if either has the smallest group distortion, as shown in Fig. 2(a) where Group 2 is the minimum distortion group as an example. Similarly two inner points nearest to Group 1, Group 3, Group 4 or Group 6 are to be checked if one of the groups corresponds to the smallest group distortion, as shown in Fig. 2(b) where an example for Group 1 is illustrated. Note that Group 2 and 5 are the horizontally positioned while the other four groups are in diagonal directions.

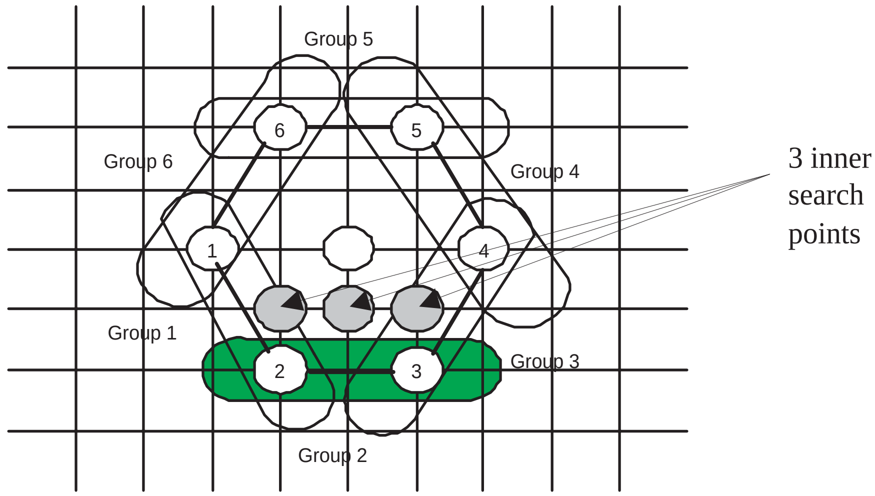
The overhead of the fast inner search is negligible, where six additions are required computationally and six memory units are needed to store the distortions corresponding to the six endpoints of the hexagon. Here we would like to highlight that our method is significantly different from the simple and efficient search (SES) algorithm [5] used to speed up the three-step search, although they share the similar philosophy in reducing the number of search points. The SES algorithm mainly speeds up the coarse search by making use of some individual distortions to determine a search quadrant in each step of the three steps based on the globally unimodal error surface assumption. In our view and observation, the globally unimodal error surface assumption is too strong to be met for most video sequences. In contrast, based on the locally unimodal error surface assumption which is statistically valid according to our experiments, our proposed method selects some portion of inner search points by taking advantage of the overall distortion information of the six checked points within a small neighborhood region. In our proposed algorithm, the group-sum distortion rather than an individual distortion is used for determining the location of the selected portion for final inner search, which has shown better performance (due to page limit, the results using an individual distortion are not included). It can be seen that the implementations for the two algorithms are substantially different.

B. Enhanced HEXBS Algorithm

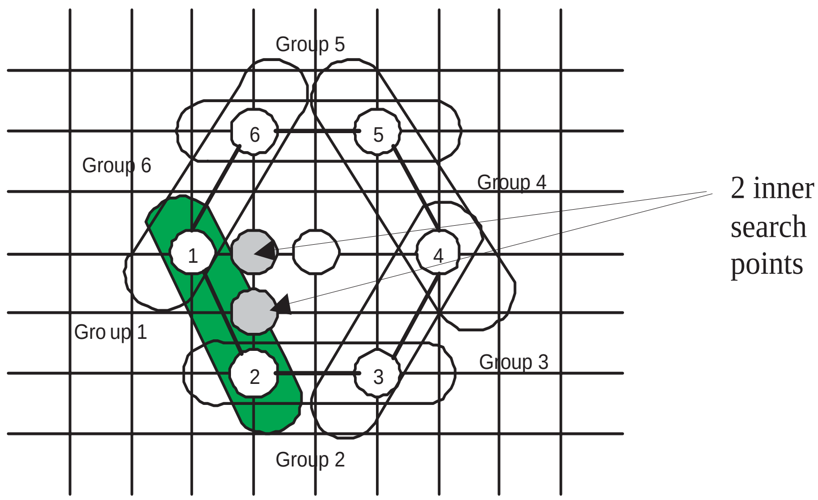
On top of the original hexagonal algorithm HEXBS, an enhanced HEXBS can be developed by incorporating the above-elaborated 6-side-based fast inner search scheme. Moreover, the predictive way of finding a good starting point using the neighboring motion vectors can also be employed in the enhanced HEXBS to reduce the number of search points in the coarse search before the inner search. Therefore, the reduction of number of search points for the enhanced HEXBS algorithm is expected to be from two aspects, i.e., one from the prediction for a good starting point using the predictive HEXBS, and the other from the proposed fast inner search. In the following experimental results, the contributions attributed to the two aspects are broken down listing the results for the predictive HEXBS (with only prediction) and the enhanced HEXBS (with prediction + fast inner search), respectively.

III. Experimental Results and Analysis

To evaluate performance of the proposed enhanced HEXBS algorithm featured with the 6-side-based fast inner search, we compare it against the original and predictive HEXBS methods in terms of number of search points and distortion. The experimental set-up is as follows: the distortion measurement of mean square error (MSE) used, block size of 16×16, and search window size of ± 15. Nine representative video



(a)



(b)

Fig. 2. (a) Three inner points nearest to Group 2 with the smallest group distortion are to be checked. If the smallest distortion group is 2 or 5, three checking points nearest to the smallest distortion group will be used in the focused inner search. (b) Two inner points nearest to Group 1 with the smallest group distortion are to be checked. If the smallest distortion group is 1,3,4 or 6, two inner points nearest to the smallest distortion group will be evaluated in the focused inner search.

sequences, "Caltrain" (512×400, 30 frames), "Coastguard" (352×240, 300 frames), "Claire" (360×288, 165 frames), "Garden" (720×480, 98 frames), "Fish" (352×240, 117 frames), "Football" (720×480, 59 frames), "Salesman" (352×288, 448 frames), "Suzie" (176×144, 149 frames) and "Tennis" (720×480, 39 frames), were used, which vary in motion content as well as frame size. An additional video sequence "Dancewolf" (720×480, 299 frames) containing large motion of horse racing is also included for testing, which was generated from a film "Dance with the wolf".

Average number of search points and average MSE values are summarized in Table 1 and Table 2 respectively, for the different algorithms and the ten different video sequences. It can be clearly seen that our proposed enhanced HEXBS has consistently achieved the fastest motion estimation in terms of number of search points with smaller MSE distortions than the original HEXBS (except "Salesman" with a slight increase in MSE) and almost same MSE distortions as the predictive HEXBS (very close to the distortions by full search). Besides, Table 3 tabulates the results of the proposed Enhanced HEXBS algorithm over the original and predictive HEXBS in terms of speed improvement rate (%SIR) and the MSE decrease percentage ($\%D_{MSE}$) [1] in average. In order to break down the improvements contributed by the prediction scheme adopted

[1]The speed improvement rate (%SIR) of Method 1 over Method 2 is defined by $\%SIR = \frac{N_2 - N_1}{N_1} \times 100\%$, where N_2 is the number of search points used in the Method 2 while N_1 is the number of search points used in the Method 1. Similarly, the distortion decrease percentage of Method 1 over Method 2 in terms of MSE can be obtained by $\%D_{MSE} = \frac{MSE_2 - MSE_1}{MSE_1} \times 100\%$, where MSE_1, MSE_1 are the distortions for Method 1 and 2, respectively.

in the Predictive HEXBS and the proposed fast inner search respectively, Table 3 also lists the speed improvement rate and the MSE decrease percentage of the predictive HEXBS over the original HEXBS for more convincing comparison.

Some Remarks On The Results

From Table 3, we can see that the proposed fast inner search consistently yields speed improvement around 20% (referring to the results of the proposed enhanced HEXBS over the predictive HEXBS), while the prediction scheme in the Predictive HEXBS yields fluctuated speed improvement from -3.03% to 30.7% for different video sequences. For the large motion video sequences such as "Dancewolf", "Garden" and "Football", the prediction scheme results in more contributions than the fast inner search in reducing number of search points. However, for the medium motion or especially low motion video sequences, the prediction scheme produces minor contribution or even negative contribution, i.e., increases number of search points slightly. For example, for "Salesman", "Suzie" and "Claire" sequences which contain low motion information with zero-biased motion vector, the prediction scheme is of no help in reducing number of search points but resulting in a slight increase in the number of search points shown in Table 1 and Table 3 (in percentage). In contrast, the proposed fast inner search contributes quite constantly in accelerating the motion estimation for different characteristics of videos, thus becoming the dominant contributor in speeding up the motion estimation for those low motion video sequences. In short, the results provided show that the proposed inner search provides typically 20% speed up while maintaining approximately the same MSE, for all the video sequences. On the other hand, the prediction provides improvement for both speed and MSE in fast motion sequences and only marginal improvement for slow motion sequences.

As shown in Fig. 2, compared with the conventional inner search in the original/predictive HEXBS algorithm, the proposed fast inner search in the Enhanced HEXBS will save two points for four sides and one point for another two sides. By uniformly averaging, $2 \times 4/6 + 1 \times 2/6 \approx 1.67$ search points will be saved if we assume each side wins with equal probability. From our experimental results, around 2 search points have been saved for the Enhanced HEXBS compared with the Predictive HEXBS. The reason behind this may lie in two aspects. One is due to the different starting points predicted from the neighboring macroblocks with different motion vectors obtained by the Predictive HEXBS and Enhance HEXBS respectively. Another is that the four sides that can help save two points in inner search win in most cases by having the smallest distortion sum.

Fig. 3 plots a frame-by-frame comparison in number of search points, for the proposed Enhanced HEXBS and the original/predictive HEXBS applied to "Suzie" and "Coastguard" sequences, respectively (due to page limit, we omit the results for some other representative sequences). The figures clearly show that the Enhanced HEXBS reduces the number of search points substantially compared with the original/predictive HEXBS, while the Enhanced DS can achieve smaller MSE than the original DS and even than the predictive DS for the two sequences "Fish" and "Suzie", shown in Table 1 and Table 3 (in percentage). In summary, in terms of number of search points as the indicator for algorithm speed, the proposed Enhanced HEXBS is evidently the fastest. On the other hand, in terms of distortion as the indicator for algorithm accuracy, the Enhanced HEXBS is also obviously better than the original HEXBS and comparable to the predictive HEXBS (very slightly worse in most cases but somewhat better in some cases). More information and observations can be read from the results listed in the tables and figures.

IV. Conclusion

In view that the whole motion estimation process comprises coarse search and finer-resolution inner search, it becomes more important to speed up the inner search when some fastest methods have been used to perform the coarse search resulting in that the coarse search requires only comparable or fewer search points compared to the full inner search, especially for those low-motion video sequences. A new fast inner search has been proposed to complement the Hexagon-based search (HEXBS) to further speed up the motion estimation and also decrease distortions, together with a simple prediction of a good starting point. In the proposed technique the final inner search is constrained to one side of the hexagon with the minimum average distortion, thus only part of the inner points will be evaluated. Experimental results have consistently shown that the resulted Enhanced HEXBS algorithm outperforms the original HEXBS remarkably in terms of search speed and distortion performance, up to 57% in both speed improvement rate and distortion decrease rate.

Acknowledgment

The authors are very grateful to H.A. Ang, C.Y. Ong and Y.L. Chua for their valuable help in programming and meticulously collating all the experimental results.

References

[1] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Communications*, vol. COM-29, pp. 1799-1808,Dec. 1981.

[2] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp. 438–443, Aug. 1994.

[3] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 313–317, June 1996.

[4] L. K. Liu and E. Feig, "A block-based gradient descent search algorithm for block motion estimation in video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 419–423, Aug. 1996.

[5] J. Y. Tham, S. Ranganath, M. Ranganath and A. A. Kassim, "A novel unrestricted center-biased diamond search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 4, pp. 369–377, Aug. 1998.

[6] 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.

[7] Jianhua Lu and M. L. Liou, "A simple and efficient search algorithm for block-matching motion estimation," *IEEE Transactions on Image Processing*, vol. 7, no. 2, pp. 429–433, April 1997.

[8] P. I. Hosur and K. K. Ma, "Motion vector field adaptive fast motion estimation," in *2nd International Conference on Information, Communications and Signal Processing (ICICS'99)*, Singapore, Dec. 1999.

TABLE I

AVERAGE NUMBER OF SEARCH POINTS PER BLOCK WITH RESPECT TO DIFFERENT METHODS AND DIFFERENT VIDEO SEQUENCES

	Caltrain	Coastguard	Claire	Dancewolf	Garden	Fish	Football	Salesman	Suzie	Tennis
HEXBS	13.233	12.853	10.368	16.018	16.165	10.807	15.99	10.512	10.236	12.887
Predictive HEXBS	12.071	11.012	10.44	13.208	12.368	10.857	13.457	10.841	10.331	12.002
Enhanced HEXBS	10.040	8.926	8.707	11.310	10.317	9.018	11.409	8.935	8.574	10.371

TABLE II

AVERAGE MSE PER PIXEL FOR DIFFERENT METHODS AND DIFFERENT VIDEO SEQUENCE

	Caltrain	Coastguard	Claire	Dancewolf	Garden	Fish	Football	Salesman	Suzie	Tennis
FS	53.225	72.918	4.744	51.005	95.916	88.251	139.264	17.529	18.125	117.117
HEXBS	60.095	79.252	5.241	59.186	162.943	98.962	216.323	18.272	20.302	151.123
Predictive HEXBS	55.425	73.799	4.815	56.563	103.649	95.085	178.166	17.933	18.956	130.279
Enhanced HEXBS	56.201	74.177	4.831	56.640	103.856	92.175	178.861	18.592	18.887	135.795

TABLE III

SPEED IMPROVEMENT RATE (%SIR) AND MSE DECREASE PERCENTAGE (%DMSE) COMPARISONS AMONG THE PROPOSED ENHANCED HEXBS, THE PREDICTIVE HEXBS, AND THE ORIGINAL HEXBS

	Caltrain	Coastguard	Claire	Dancewolf	Garden	Fish	Football	Salesman	Suzie	Tennis
The Predictive HEXBS over the Original HEXBS										
%SIR	9.63	16.72	-0.69	21.27	30.70	-0.46	18.82	-3.03	-0.92	7.37
$\%D_{MSE}$	8.43	7.39	8.85	4.64	57.21	4.08	21.42	1.89	7.10	16.00
The proposed Enhanced HEXBS over the Original HEXBS										
%SIR	31.80	44.00	19.08	41.63	56.68	19.84	40.15	17.65	19.38	24.26
$\%D_{MSE}$	6.93	6.84	8.49	4.50	56.89	7.36	20.94	-1.72	7.49	11.29
The proposed Enhanced HEXBS over the Predictive HEXBS										
%SIR	20.23	23.37	19.90	16.78	19.88	20.39	17.95	21.33	20.49	15.73
$\%D_{MSE}$	-1.38	-0.51	-0.33	-0.14	-0.20	3.16	-0.39	-3.54	0.37	-4.06

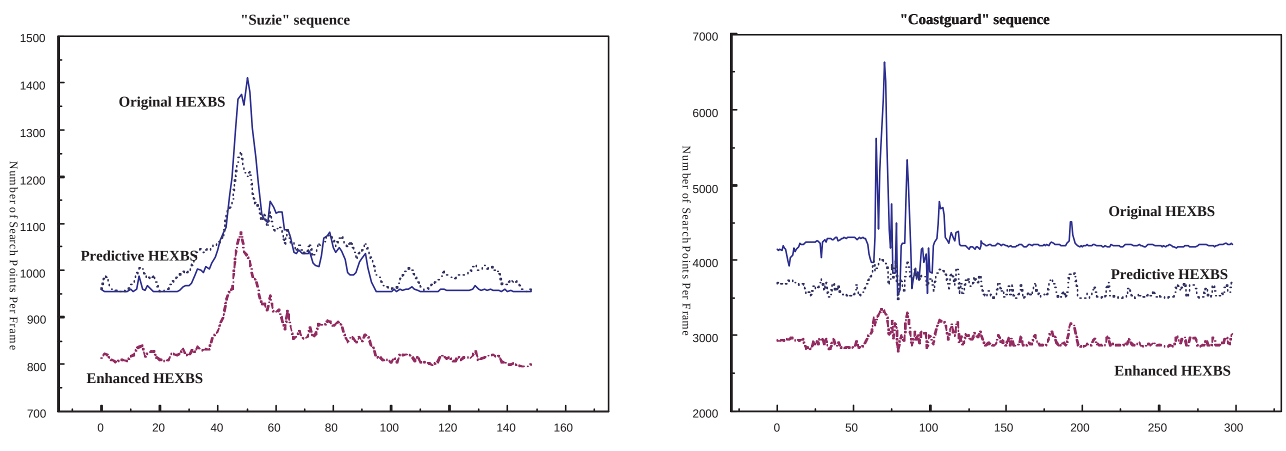


Fig. 3. Frame-by-frame comparison in number of search points for the three HEXBS algorithms applied in different video sequences.

[9] A. M. Tourapis, O. C. Au and M. L. Liou, "Predictive motion vector field adaptive search technique (PMVFAST) enhancing block based motion estimation," in *SPIE Conf. On Visual Communication and Image Processing*, pp. 883–892, Jan. 2001.

[10] C. Zhu, X. Lin, and L. P. Chau, "Hexagon-based search pattern for fast block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 349–355, May 2002.