A Novel Flatted Hexagon Search Pattern for Fast Block Motion Estimation

Thou-Ho (Chao-Ho) Chen and Yi-Fan Li
Department of Electronic Engineering, National Kaohsiung University of Applied Sciences,
Kaohsiung, Taiwan 807, R.O.C.

ABSTRACT

A speed-quality effective block-matching algorithm (BMA) based on the flatted-hexagon search (FHS) pattern is developed for motion estimation. The basic idea behind the proposed algorithm is that the covering range of a search pattern should be enlarged as horizontal as possible to find the optimal motion vector quickly due to the fact that the probability of horizontal-biased motions is larger than that of vertical-biased motions is most real-world image sequences. This reveals that both the searching-speed and matching-probability can be improved if the shape of a search pattern is flatted horizontally. Experimental results show that the proposed FHS is superior to other famous BMAs, such as the three-step search (3SS), diamond search (DS) and hexagonal-based search (HEXBS), in terms of speed-probability product improvement rate corresponding to the full search (FS).

1. INTRODUCTION

Motion estimation can make the interframe coding to achieve a very high compression ratio, when compared to the intraframe coding, by exploiting the heavy temporary redundancy between successive frames. Among various motion estimation techniques, the block-matching algorithm (BMA) is the most attractive method for the current international video compression standards including H.261, H263, MPEG-1, MPEG-2, MPEG-4 [1], because of its effectiveness and simplicity for implementations. However, the matching process of finding the optimal still involves a large amount of calculations, e.x., the most accurate approach, called the full search (FS) method which requires to evaluate all candidate blocks, can consume at least 63% of the computational power for the MPEG-4 encoder. To reduce the intensive computational complexity with a tolerable distortion, many fast block-matching algorithms were developed [2-9].

Among the above suboptimal methods, both concepts of the search pattern's attributes and initial searching range directs the developmental process of these algorithms. By taking advantage of the characteristics of the center-biased motion vector distribution existed in most real-world image sequences, the new three-step search (NTSS) [3], four-step search (4SS) [4] and block-based gradient descent search (BBGDS) [5] perform better than the three-step search (3SS) [2], where these four search patterns are square-shaped. Based on a practical compact-shaped pattern with fewer checking points, a diamond-search (DS) algorithm [6][7] not only improves the searching speed but also reduce the chances of being trapped in local optimal compared to those four algorithms. For most high-resolution (e.x., 720×480) image sequences, the hexagon-based search (HEXBS) algorithm [8] utilizes a hexagon-shaped pattern with only 7 checking points to achieve substantial speed improvement over the DS algorithm with similar distortion performance. Nevertheless, the matching-probability performance will degrade with the decreasing resolution in the format of video. To obtain a faster searching speed than the DS algorithm while maintaining similar search quality, the cross-diamond search (CDS) algorithm [9] employs a cross-search pattern at the initial step to exploit the characteristics of the center-biased motion vector distribution very efficient, followed by halfway-stop technique, and large/small diamond search patterns in the subsequent steps. Although, various search patterns and processes at different steps will make the CDS algorithm to be complicated on realization, especially for VLSI implementation due to its favoritism of regularity.

In this paper, we propose a novel fast block motion estimation with a novel flatted-hexagon search (FHS). The introduction of flatted-hexagon search pattern will make the FHS algorithm has a better speed-probability performance than other BMAs, especially for low-resolution image sequences (e.x., 352×240 or 352×288), which both speed and matching probability must be considered. Section II analyzes attributes of search patterns used in most real-world image sequences. The FHS algorithm will be described in Section III, and then simulation results and comparisons with several fast BMAs for five standard image sequences are given in Section IV. Conclusions are made in the final section.

2. ANALYSIS OF SEARCH PATTERNS

The previous researches [2-9] reveal that both shape and size of a search pattern can influence the searching speed and quality significantly. In point of shape independly, compactness is aimed to consider all possible searching directions for tracking the optimal motion vector that has the least matching error. The hexagon is more compact than the diamond, as shown in Fig.1 (a) and (b), which in turn is better than the square. The pattern's size will affect the probability of the best match and also the moving speed of the search pattern. Small-size pattern usually makes the searching to be trapped into a local minimal-error point, especially for those image sequences with large motion content, which may also implies the high-resolution format. On the other hand, a large search pattern is most likely to result in misleading of the searching direction that may frequently either delay the searching time or even miss the optimal one, especially for the video with small-motion content or low-resolution format. It should be pointed out that the quantity of checking points at each step will have the similar influence on speed and quality as the pattern's shape and size. Basically speaking, fewer checking points in every step can speed up the search but suffer a larger distortion, and oppositely more checking points with provide a better quality performance of block-matching.

Among fast BMAs with center-biased searching, the diamond-shaped search algorithm [6][7] performs better than those square-shaped methods [2][5], because its medium-size compact-shape pattern can find large motion vectors with 9 checking points to increase the search speed under a certain quality. Also, the unrestricted search process minimizes the distortion caused by the local-optimal trapping problem. The hexagon pattern used in [8] has a more compact (i.e, circle-approximated) shape with, larger size and less checking points (7 checking points) than the above diamond pattern. By using such a search pattern, the HEXBS algorithm has a

0-7803-8554-3/04/$20.00 ©2004 IEEE.

speed-up improvement over the DS algorithm mainly owing to the contribution of less checking points. But, the joinment of fewer checking points and larger size for the search pattern will make the HEXBS suffer a degradation on the probability of finding true motion vectors. This leads that the HEXBS can maintain a similar matching probability as the DS for the most high-resolution image sequences (e.x., CCIR601, 720×486) but only for some certain low-resolution image sequences (e.x., SIF, 352×240 or CIF, 352×288). In other words, the HEXBS has a significant matching-probability degradation for most low-resolution image sequences, when compared to the DS.

Based on an advanced analysis on the distributions of motion vector probabilities in the most real-world image sequences, it is clear that the probability of horizontal-biased motions is significantly larger than that of vertical-biased motions. Table 1 lists probability distributions of horizontal-biased and vertical-biased motions in five well-known image sequences with various motion contents for a search window ±7. In the table, horizontal-biased and vertical-biased constituents are defined as motion vectors, in which the angle between the motion vector itself and the horizontal and vertical line, respectively, is equal or small than 30°. For the video-conferencing sequence, the "Salesman" bears a very high center-biased motion vector distribution, i.e., a small-motion content, with a very low H/V (horizontal/vertical) probability ratio of 17.94/14.17. For the medium-motion sequence, the "Foreman" has a high H/V ratio of 30.17/22.62. For large-motion content, the "Football" sequence has a low H/V ratio of 27.14/17.05 but the "Garden" sequence captured by panning the camera with translation is a very high horizontal-motion-biased sequence with a H/V ratio of 92.78/2.12, while the "Tennis" sequence with camera zooming has a very low H/V ratio of 26.07/20.50.

The above analysis reveals that the covering range of a search pattern should be enlarged as horizontal as possible in order to find the optimal motion vector quickly. This motivates that both speed and probability performance in block-matching process may be improved if the shape of a search pattern is flatted horizontally. A hexagon becomes the most attractive search-pattern to be flatted, since it has a more compact form than other search patterns reported previously. Therefore, a flatted-hexagon search pattern used in the block-matching algorithm for motion-vector estimation is proposed and described in the following section. By the same deduction, the hexagon search pattern may also be flatted vertically for the certain image sequences with massive vertical-biased motion contents.

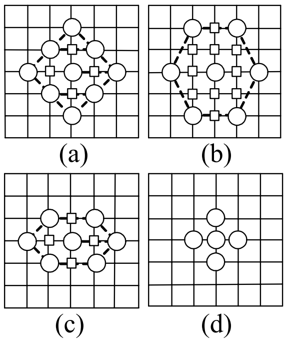


Fig. 1 shapes of various search patterns:
(a) diamond pattern;
(b) hexagon pattern;
(c) flatted-hexagon pattern;
(d) ending cross pattern
○ : checking point
□ : unchecked point

3. THE FLATTED HEXAGON SEARCH ALGORITHM

The flatted-hexagon pattern can be viewed as that a hexagon pattern [8] is flatted horizontally or that both top and bottom checking points of a diamond pattern [5][6] are removed. Besides, the flatted-hexagon search pattern is composed of seven checking points with the center surrounded by six endpoints of the flatted hexagon as similar to the hexagon search pattern, as described in Fig. 1 (c). Hence, this makes the searching process oh the FHS algorithm is almost the same as the HEXBS algorithm. That is, the flatted hexagon search pattern keeps advancing with the center moving to any of the six endpoints and whichever endpoint the center of the search pattern moves to, there are always three new endpoints introduced and the other three endpoint and the original center point are overlapped, as shown in Fig. 2 (c). A cross pattern, as shown in Fig. 1 (d) that is also adopted by the HEXBS and DS, is finally need in the focused inner search.

Firstly, a minimum block distortion measure (BDM) is obtained. By calculating the 7 search-points of the flatted-hexagon pattern which is located at the center of search windows, as shown in Fig. 2 (a). If the minimum BDM is found at the central checking point, the search will switch to use the cross pattern including new 4 search points around that center for ending the search, plotted in Fig. 2 (b). Then, one with minimum BDM among there 5 checking points will be the optimal solution for motion vector estimation. Otherwise, the flatted-hexagon pattern moves toward one endpoint with a minimum BDM and then the search continues with the same flatted-hexagon pattern centered at that minimum BDM point in two normal forms of Fig. 2 (c). In such two normal searches each needs only 3 new checking points.

The proposed FHS algorithm can be summarized in the following three steps.

Step 1 (Starting):

A flatted-hexagon pattern is set as the center (0,0) of the search windows. If the minimum BDM point is found to be at the center of the search pattern, go to Step 3 (Ending); otherwise, go to Step2 (Searching).

Step 2 (Searching):

A new flatted-hexagon pattern is formed and centered at the point which has a minimum BDM in the previous search step. Three new search points are compared together with those 5 overlapped points checked in the previous step. If the minimum BDM point again occurred at the center of the search pattern, go to Step 3(Ending); otherwise, repeat this step continuously.

Step 3 (Ending):

A cross pattern is set at the minimal BDM point calculated in the previous step. Four new candidate points covered by the cross pattern are compared with the previous minimal BDM located at the center of the cross pattern to generate a final optimal solution, i.e., the estimated motion vector. Based on the above searching strategy, the total number of search points per block will be deduced as

$$N_{FHS}\ (mv_x\ ,\ mv_y) = 7+3\times n+4 \qquad (1)$$

Where $(mv_x\ ,\ mv_y)$ is the final motion vector estimated and n is the number of Step 2 executed. Note that equation (1) is the same as the HEXBS but different searching depths. Fig.3 demonstrates a search path strategy which results in a motion vector (4,-1) with 17 checking points.

4. EXPERIMENTAL RESULTS AND COMPARISONS

For the purpose of comparison, four previous BMAs and the proposed FHS algorithm are simulated by using the luminance component of five popular sequences with various types: “Salesman” (CIF 499 frames),”Foreman” (CIF 300 frames),”Garden” (SIF 115 frames),”Tennis” (SIF 67 frames) and “Football” (SIF 125 frames). Evaluation with such five representative sequences in terms of MAD (mean absolute distortion) used as the BDM, matching probability (i.e., the probability of finding the true motion vector) and number of checking points required is described in Table 2. For each sequence, the search is performed at a block size of 16×16 within a window of size ±7. For making a tradeoff between speed and probability improvement compared to the FS method, a measure of speed-probability product is introduced into the table. The *SP* product is defined as follows:

$$SP = (N_{FS} / N_{way}) \times (P_{way} / P_{FS})$$

Where N_{FS} and N_{way} denote the number of checking points required for the FS and another searching way, respectively while P_{FS} and P_{way} mean the matching probability of the FS and another way. Both ratio of N_{FS}/N_{way} and P_{way}/P_{FS} also imply the rates of speed and matching probability improvement compared to the FS, respectively. In other words, the value of *SP* can be viewed as a rate of speed-probability improvement over the FS. Thus, the larger the *SP* value, the more the improvement can be achieved.

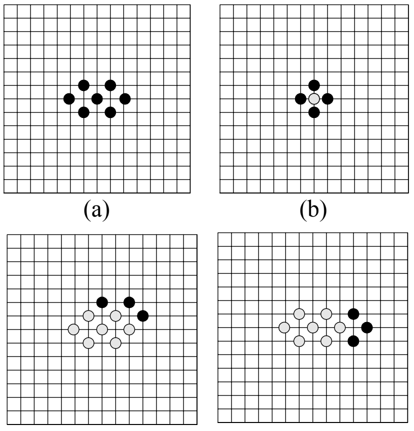


Fig. 2 Various forms of the flatted-hexagon search pattern: (a) starting search-points; (b) ending search-points; (c) search points of two normal searches.
●: the required checking points

In Table 2, it is clear that the FHS is superior to other four previous BMAs when the *SP* improvement is considered. Especially for “Garden” sequence, its large quantity of horizontal-motions makes the FHS bear a higher *SP* value than other BMAs. For a highly center-biased image sequence with small-motion content, “Salesman”, the FHS has the highest matching probability and even fewer checking points required than the DS and about the same number of checking points as the HEXBS. For medium- and large-motion sequences with a moderate horizontal-motion probability, such as “Foreman”, “Tennis” and “Football”, the FHS always achieves substantial probability improvement over the HEXBS with a little speed degradation and a significant speed improvement over the DS with a moderate probability decrement. Based on this feature, the FHS can provide a higher value of *SP* than other BMAs.

Fig. 4 (a) and (b) plot the average search points per block and MAD difference corresponding to the FS, respectively, in a way of frame-by-frame comparison for the “Garden” sequence. These curves demonstrate that the proposed FHS is superior to the DS but a similar to the HEXBS in terms of number of search points used and gives a similar distortion error as the DS but a large improvement over the HEXBS. It should be noted that, the *SP* improvement rate of FHS will decrease with the increasing resolution of the video format due to its small size of search pattern.

5. CONCLUSIONS

The paper presents a novel flatted-hexagon search algorithm for block motion estimation. The flatted-shape of search pattern makes the proposed FHS algorithm to have a searching speed only slightly smaller than that of HEXBS but significantly larger than the DS, and a matching probability larger than that of HEXBS but slightly smaller than the DS. For most real-world video data, experimental results reveal that the FHS algorithm will be more attractive than others. When both the searching speed and matching probability are concurrently considered.

6. REFERENCES

[1] K. R. Rao and J. J. Hwang, *Techniques and Standards for Image, Video, and Audio Coding*. New Jersey: Prentice Hall, Inc., 1996.

[2] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, “Motion compensated interframe coding for video conferencing,” in *Proc. National Telecommunications Conf.*, New Orleans, LA, Nov. 1981, pp. G5.3.1-G5.3.5.

[3] 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.

[4] 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.

[5]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.

[6] 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, pp. 369-377, Aug. 1998.

[7] S. Zhu and K. K. Ma, “A new diamond search algorithm for fast block-matching motion estimation,” *IEEE Trans. Image Processing*, vol. 9, pp. 287-290, Feb.2000.

[8] Ge Zhu, Xiao Lin, and Lap-Pui Chau, “Hexagon-Based Search Pattern for Fast Block Motion Estimation”, IEEE Transaction on Cir. Sys. Viedo Tech., Vol. 12, No. 5, May 2002.”

[9] C. H. Cheung and L. M. Po,” A Novel Cross-Diamond Search Algorithm for Fast Block Motion Estimation”, Transaction on Cir. Sys. Viedo Tech., Vol. 12, No. 12, Dec. 2002.”

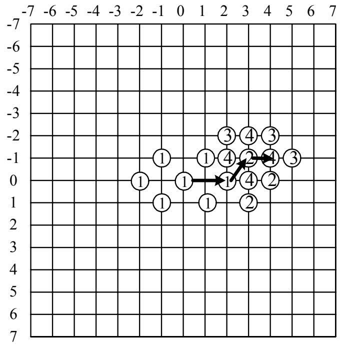
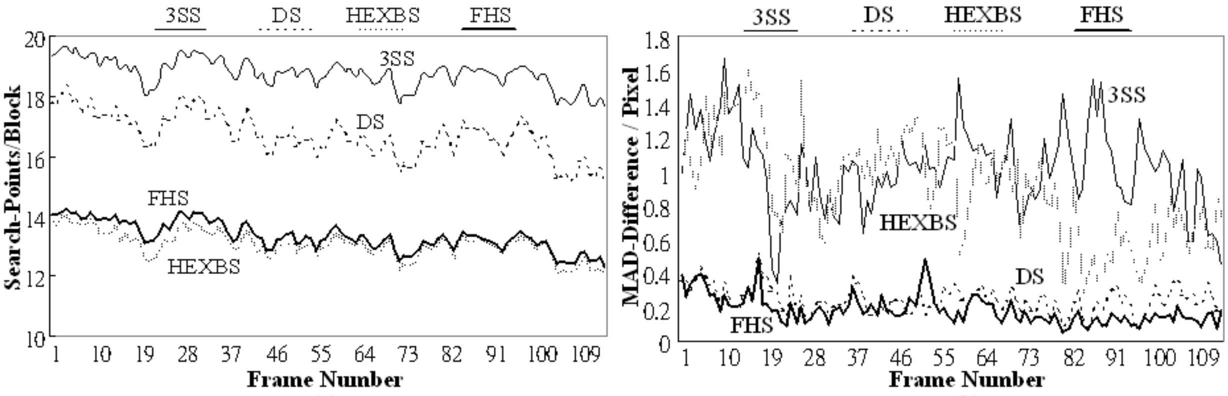


Fig. 3
A realizing example of FHS, in which the final motion vector (4,-1) is found with 17 checking points.
① Starting
② Searching
③ Searching
④ Ending



(a) (b)

Fig. 4 Comparison of 3SS, DS, HEXBS and FHS for "Garden" sequence in terms of: (a) the average number of search point per block; and (b) the average MAD difference per pixel corresponding to the FS.

Table 1. Probability distributions of horizontal- and vertical-biased motions

	Salesman	Foreman	Garden	Tennis	Football
Horizontal(%) probabilities	17.94	30.17	92.78	26.07	27.14
Vertical (%) probabilities	14.17	22.61	2.12	20.50	17.05

* Horizontal and vertical constituents are defined as motion vectors with $\theta_h \leq 30^0$ and $\theta_v \leq 30^0$, respectively.

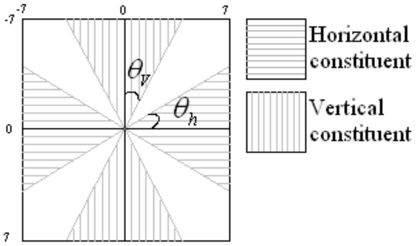


Table 2. Simulation results of FHS and other fast BMAs

Salesman 352x288 449frames

BMAS	MAD	Probability(%)	Points	SP
FS	2.773	100.000	204.283	1
3SS	2.824	94.641	23.212	8.329
DS	2.813	95.227	12.890	15.090
HEXBS	2.825	94.287	10.564	18.232
FHS	**2.784**	**96.589**	**10.637**	**18.548**

Garden 352x240 115frames

BMAS	MAD	Probability(%)	Points	SP
FS	8.413	100.000	202.048	1
3SS	9.431	83.519	23.212	7.269
DS	8.656	92.894	16.712	11.230
HEXBS	9.322	81.470	13.069	12.594
FHS	**8.595**	**93.460**	**13.333**	**14.162**

Foreman 352x288 300frames

BMAS	MAD	Probability(%)	Points	SP
FS	4.144	100.000	204.283	1
3SS	4.391	83.299	23.334	7.292
DS	4.372	87.349	17.237	10.352
HEXBS	4.617	69.329	12.956	10.931
FHS	**4.552**	**81.890**	**14.036**	**11.918**

Tennis 352x240 67frames

BMAS	MAD	Probability(%)	Points	SP
FS	4.809	100.000	202.048	1
3SS	5.862	70.615	23.436	6.087
DS	5.045	91.129	16.309	9.672
HEXBS	5.438	75.238	12.891	11.792
FHS	5.338	**84.954**	**13.415**	**12.794**

Football 352x240 125frames

BMAS	MAD	Probability(%)	Points	SP
FS	10.065	100.000	202.048	1
3SS	10.534	88.599	23.117	7.743
DS	10.513	91.307	15.968	11.552
HEXBS	10.822	79.029	12.387	12.890
FHS	**10.688**	**87.878**	**13.151**	**13.501**