Novel Cross-Diamond-Hexagonal Search Algorithms for Fast Block Motion Estimation

Chun-Ho Cheung, *Member, IEEE,* and Lai-Man Po, *Member, IEEE*

Abstract—**In this paper, we propose two cross-diamond-hexagonal search (CDHS) algorithms, which differ from each other by their sizes of hexagonal search patterns. These algorithms basically employ two cross-shaped search patterns consecutively in the very beginning steps and switch using diamond-shaped patterns. To further reduce the checking points, two pairs of hexagonal search patterns are proposed in conjunction with candidates found located at diamond corners. Experimental results show that the proposed CDHSs perform faster than the diamond search (DS) by about 144% and the cross-diamond search (CDS) by about 73%, whereas similar prediction quality is still maintained.**

Index Terms—**Cross-center-biased property, cross-diamond-hexagonal search, motion estimation.**

I. INTRODUCTION

BLOCK-MATCHING motion estimation is inextricably part of today's video coding techniques and standards, such as ISO/IEC MPEG-1, 2, 4 [1]–[3], ITU-T H.261 [4], H.263 [5], and the emerging H.264 [6], [7]. Video frames are first divided into macroblocks. The displacement of these macroblocks from reference frame are reckoned and coded together with the residual frames using the generic hybrid predictive/transform coding framework [1]–[7]. In the last 20 years, many fast algorithms were proposed to pursue low computational complexity consumed by the full search (FS) algorithm. To reduce the exhaustive checking of candidate motion vectors (MV), fast block-matching algorithms (BMA) with different block-matching strategies and their corresponding search patterns with various sizes and shapes have enormous impact on both search speed and accuracy. The typical example is the three-step search (3SS) [8], which employs rectangular search patterns with different sizes. Others like 2D-logarithmic search (2DLOG) [9] and orthogonal search (OSA) [10] algorithms performed searching in either orthogonal or linear direction. These fast algorithms result in speed improvement, however, with quality varied amongst the nature of video sequences, especially for those possessing high motion content. Afterwards, the exploitation of center-biased property by new three-step search (N3SS) [11], four-step search (4SS) [12], and block-based gradient descent search (BBGDS) [13] algorithms increased the searching speed significantly by taking the nature of most real-world sequences into account. Meanwhile, they still maintained the prediction quality comparable with the FS's. Until the unprecedented suggestion of unrestricted search steps and nonrectangular search patterns, such as the diamond search (DS) [14]–[16] and hexagon-based search (HEXBS) [17] algorithms, they required much fewer checking points, in contrast to algorithms with limited steps. Although these fast BMAs may result in sub-optimal solution because of traps by local minima, they were usually employed in practical implementations due to their simplicity and regularity of data access, for instance 3SS [18].

Besides the shape that candidate blocks taken for matching, the size of search patterns employed in these fast algorithms gave us an insight to the search strategies. Search pattern located at the central 3×3 grid is advantageous to efficiently tackle blocks with very small motion or quasistationary. The initial step using 3×3 grid in N3SS and BBGDS is a typical example favorable for videoconferencing, like "Miss America" and "akiyo", in which more than 80% of blocks are found quasistationary. The larger central 5×5 search pattern, for example, used in 4SS, DS and HEXBS, can provide faster searching speed than N3SS and BBGDS. Moreover, they give consistently better motion estimates and directions due to larger size. Another relief of reducing checking points is to have successive search patterns overlapped as much as possible. For example, 4SS and DS requires three or five extra checking points while HEXBS requires consistently three extra points in advancing step. Recently, cross-diamond search (CDS) [19] algorithm exploits a more dominant cross-center-biased property in most real-world sequences. A more advanced idea using similar starting pattern, but with dynamic size and adaptive feature, can be found in [20]. Experimental result in [19] basically compares most fast BMAs without prediction feature. It shows that CDS generally requires about two to five fewer points than DS. The speedup gain using CDS over DS is reported up to 40% with similar or even better quality than DS's. In this paper, we propose two novel cross-diamond-hexagonal search (CDHS) algorithms by employing a smaller cross-shaped pattern before the first step of CDS and replacing the diamond-shaped pattern with hexagonal search patterns (HSP) in subsequent steps. The rest of the paper is organized as follows. Section II presents the proposed search patterns used in CDHSs. In Section III, the algorithmic flow of the CDHSs will be described and followed with a theoretical analysis on the gain in search points by using CDHSs over CDS and DS. Section IV reports some significant experimental results. Lastly, the conclusions are given in Section V.

Manuscript received October 9, 2002; revised November 29, 2003. This work was supported by a grant from City University of Hong Kong, Hong Kong SAR (Project no. 7001385). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chang Wen Chen.

The authors are with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong SAR, China (e-mail: terence@ieee.org; eelmpo@cityu.edu.hk).

Digital Object Identifier 10.1109/TMM.2004.840609

TABLE I
VIDEO SEQUENCES USED FOR ANALYSIS

Frame Format	Sequences (No. of frames)
CIF (352×288)	Miss America (80), sales (80), Claire (80)
SIF (352×240)	tennis (80), garden (80), football (80)
CCIR601 (720×486)	tennis (40), garden (40), football (40)
QCIF(176×144)	carphone (382), Claire (494), foreman (400), grandma (870), missA (150), trevor (150), sales (449), Susie (150), mthr_dotr (961)
MPEG-4 (A) QCIF	Akiyo (300), container (300), sean(300), hall_monitor (300), mother_daughter(300)
MPEG-4 (B) QCIF	coastguard (300), foreman (300), silent (300)
MPEG-4 (C) QCIF	mobile (300), Stefan (300), table (300)
MPEG-4 (E) QCIF	children (300), destruct (300), td (300), Weather (300)

TABLE II
DOMINANT CROSS-CENTER-BIASED DISTRIBUTION

	$P(+)$	$P(\Diamond)$	$P(\square)$
QCIF ($w = 7$)			
$p = 0$	76.51	76.51	76.51
$p = 1$	88.30	88.30	90.25
$p = 2$	90.26	92.21	93.36
CIF/SIF ($w = 7$)			
$p = 0$	45.44	45.44	45.44
$p = 1$	69.09	69.09	71.85
$p = 2$	74.76	77.52	81.80
CCIR601 ($w = 15$)			
$p = 0$	23.34	23.34	23.34
$p = 1$	34.74	34.74	35.57
$p = 2$	46.79	47.62	51.85

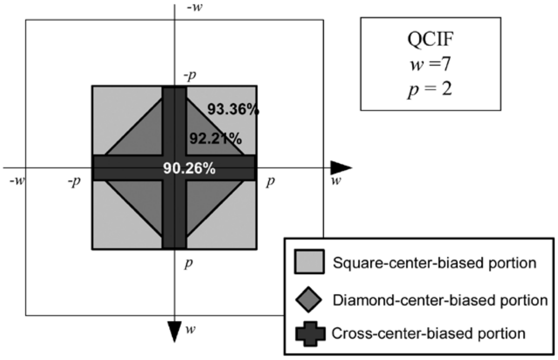
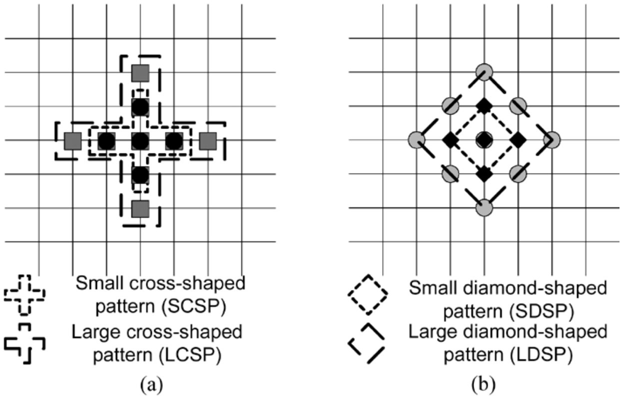


Fig. 1. Over 97% of blocks are found possessing cross-center-biased property in the central 5 × 5 diamond-zone when using QCIF sequences, i.e., $P(+)/P(\Diamond) = 90.26\%/92.21\%$ (over 96% for CIF/SIF and 98% for CCIR601).



(a) (b)

Fig. 2. Search patterns used in CDHS algorithms.

II. SEARCH PATTERN USED IN CDHSS

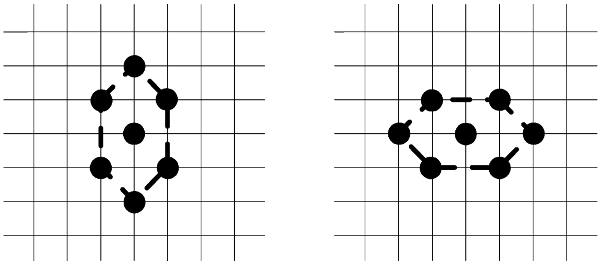
A. *Cross-Center-Biased Motion Vector Distribution*

In addition to [19], the average motion vector probability (MVP) distributions taken from 31 sequences, listed in Table I, in CIF/SIF/CCIR601 formats are analyzed. These sequences consist of different motion contents, from gentle to vigorous motion activities. The MVPs are resulted from simulations using FS with mean absolute difference (MAD) as the block distortion measure (BDM) inside a search area w, typically $|w|$ is set to 7 for QCIF/CIF/SIF and to 15 for CCIR601 sequences. We define $P(+)$, $P(\Diamond)$ and $P(\square)$ as the probabilities in which motion vectors are found within the cross $(+)$, diamond $(\Diamond)$ and square $(\square)$ region, respectively.

In Fig. 1, over 93% of motion vectors are found within the central 5×5 area when using QCIF sequences. This is a square-center-biased (SCB) distribution indicating most of real-world sequences move gently, smoothly and slowly, and can be regarded as quasistationary. Within this square region, about 90% and 92% motion vectors are found located in the cross-center-biased (CCB) and diamond-center-biased (DCB) portions. Amongst these three shaped-distributions, we can see CCB distribution is the most dominant. Moreover such cross-biased behavior maintains over 93% (and 83% for CIF/SIF) at boundaries of search area. Similar center-biased properties distributed as a cross, diamond and square shape are found in CIF/SIF and CCIR601 formats. Table II shows clearly the CCB occupies the most proportion at different displacement p, $p \in w$, especially when $1 \leq p \leq 2$. Thus, a small cross-shaped pattern (SCSP) is employed in the very beginning of the proposed algorithms, prior to the large cross-shaped pattern (LCSP) used in CDS. These two cross-shaped patterns, as shown in Fig. 2(a), can work more efficiently on finding small motion vectors than diamond-shaped ones within the central 5×5 area in the first two steps. Afterwards, the proposed search algorithms will use the large diamond-shaped pattern (LDSP) for better performance on large motion vectors and possibly stop the search using a small diamond-shaped pattern (SDSP) if the previous minimum BDM found in the diamond center. These two diamond-shaped patterns are shown in Fig. 2(b). However, our proposed algorithms work differently from DS's or HEXBS's as we propose to replace the LDSP with a large hexagonal search pattern (LHSP), shown in Fig. 3(a)–(d), when the minimum BDM located at any diamond corner.

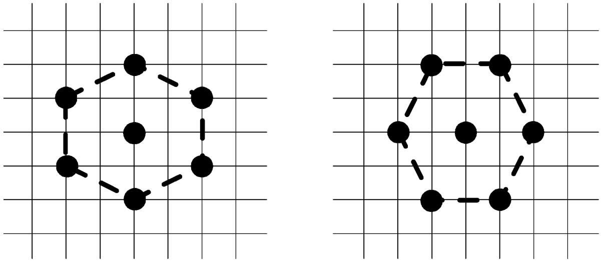
B. *Hexagonal Search Patterns*

Translation, zooming, pan and tilt motions are usually found in video. This block movement will easily fall on the diamond corners if DS is employed for coding. There is still a room to reduce two fewer points between successive steps of DS by switching the LDSP into HSPs. In this part, the probability of a search path hit any diamond corners using DS is tabulated in Table III. For example, we can see the sequences "garden" and "coastguard" have respectively over 66% and 80% diamond-



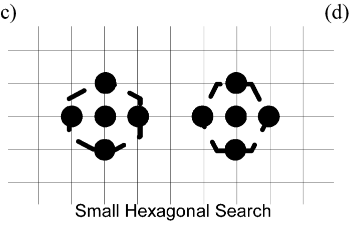
Vertical Flat Hexagonal Search Pattern (VF-HSP)
(a)

Horizontal Flat Hexagonal Search Pattern (HF-HSP)
(b)



Vertical Thick Hexagonal Search Pattern (VT-HSP)
(c)

Horizontal Thick Hexagonal Search Pattern (HT-HSP)
(d)



Small Hexagonal Search Pattern (SHSP)
(e)

Fig. 3. Two pairs of LHSPs, flat and thick, proposed in CDHS algorithms.

TABLE III
PROBABILITY THAT A DIAMOND-CORNER HIT EXISTS IN THE SEARCH PATH

Sequence name	CIF/SIF (%)		QCIF (%)	
	DS	DS+SCSP	DS	DS+SCSP
garden	66.33	47.39	93.42[a]	87.93[a]
tennis	43.02	36.43	39.85[a]	34.71[a]
football	14.77	12.28	51.84[a]	42.34[a]
missA	42.22	13.03	5.53	1.77
coastguard	80.99	66.44	31.94	4.61
stefan	56.52	49.46	52.79	41.86
foreman	39.22	33.41	25.53	21.66
mother_daughter	14.75	8.84	5.06	2.41

[a] using sequences in CCIR601 dimension.

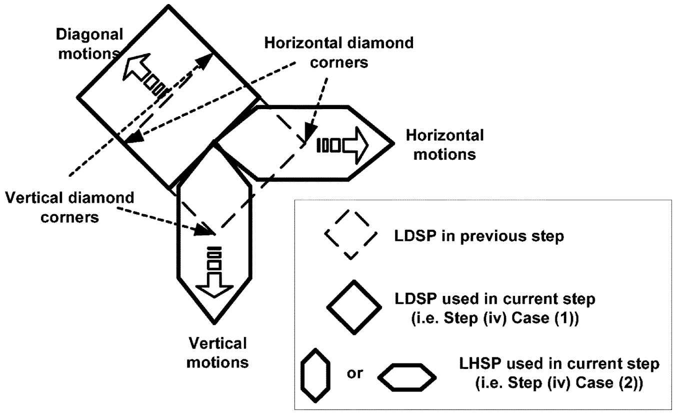
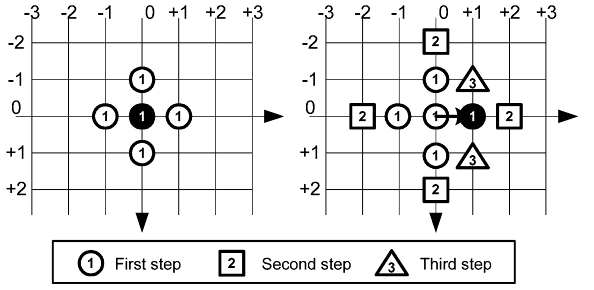


Fig. 4. Search patterns switched for different directions.

corner hit if using DS. This phenomenon still maintains at a relatively high ratio if using smaller dimension of QCIF sequences, such as coastguard (32%) and Stefan (53%). It is trivial that larger sequences, i.e., CCIR601, provide more chances to get diamond-corner hits, for example, garden (93%) and football (52%). However, it is not always true as motion direction may be "diversified" into other directions as the frame dimension increases, i.e., falling onto the diamond faces. This can be indicated by the decrease from about 43% to 40% when changing the sequence "tennis" from CIF to CCIR601 format. In addition, Table III also indicates the diamond-corner hits will be dropped if the smaller SCSP is used to allow earlier halfway-stop. The use of SCSP is applied in prior to the CDS algorithm, i.e., labeled as "DS+SCSP," and will be described clearly in Section III. From Table III, the probabilities of the search path hitting a diamond corner are generally dropped due to the earlier halfway-stop condition. With exploiting the cross-center-biased property, the drop may be tremendous, e.g., coastguard (QCIF). Nevertheless, there is still a big room to consider a switch into another search pattern, such as hexagonal patterns, in order to reduce the extra checking points in advancing from a diamond corner, especially for some rich motion clips with motions oriented orthogonally.

Fig. 3 proposes two pairs of LHSPs. Each pair consists of two orientations: vertical and horizontal. These two pairs are different in their sizes. Fig. 3(a) and (b) are *flat* version while Fig. 3(c) and (d) are *thick* version. A small hexagonal search pattern (SHSP) is also shown in Fig. 3(e), which is used as the final step. Checking points spanned in the SHSP is physically the same as SCSP and SDSP and proposed for the sake of completeness after pattern switched. Based on the search strategy of DS, any LHSP shown in Fig. 3(a)–(d) and the SHSP can be formed a new fast BMA, i.e., resulting in four *hexagonal search algorithms*. Typically, Fig. 3(d) is equivalent to the search pattern used in HEXBS. Due to space limitation, we summarize the performance of these hexagonal search algorithms though they give similar result. The thick ones advance the search with bigger steps and result in faster search speed than the flat ones, but make sacrifice for quality. Conversely, the flat ones give better quality, as they require more steps and thus more points. From the orientation point of view, horizontal ones endeavor to trigger sequences with relatively higher horizontal motion content, such as "coastguard" and "garden." In short, it is *a priori* problem that we cannot know the nature of a sequence in advance. Therefore, a diamond-shaped pattern, i.e., LDSP, plays an important role in our proposed CDHSs. Fig. 4 shows three possible directions that the minimum BDM found located in the previous LDSP. Either a pair of LHSPs will be consistently used throughout the search. If the *flat* pair is used, i.e., Fig. 3(a) and (b), we name the proposed algorithm as CDHS-*F*. Similarly, we denote CDHS-*T* as the one that employs *thick* LHSPs, i.e., Fig. 3(c) and (d). The flow of the CDHSs is now described in more detail in the following section.



(a) (b)

Fig. 5. Examples of CDHSs. Two small motion search paths.

III. PROPOSED CROSS-DIAMOND-HEXAGONAL SEARCH

A. Flow of the Proposed CDHSs

The proposed CDHS algorithms differ from DS, HEXBS, and CDS by performing a highly cross-center-biased search with SCSP in the first step. In addition, the search may involve up to two different patterns: diamond-shaped LDSP and hexagonal pair LHSP. The common strategy amongst them is employing a halfway-stop technique. The following summarizes the CDHS algorithms.

Step (i) Starting: A minimum BDM point is found from the five checking points of SCSP at the center of search area. If the minimum BDM occurs at SCSP center, the search stops. [This is called first-step-stop as shown in Fig. 5(a).]

Step (ii) Large Cross Searching: The four outermost points of the central LCSP are evaluated, i.e., the four candidates at $(\pm 2, 0)$ and $(0, \pm 2)$. This step guides another possible correct direction for the subsequent steps.

Step (iii) Half-diamond Searching: Two additional points of the central LDSP closest to the current minimum BDM of the central LCSP are checked, i.e., two of the four candidate points located at $(\pm 1, \pm 1)$. If the minimum BDM found in previous steps is at any endpoint of SCSP, i.e., $(\pm 1, 0)$ or $(0, \pm 1)$, and the new minimum BDM found in this step still coincides with this point, the search stops. [This is called third-step-stop, as shown in Fig. 5(b).]

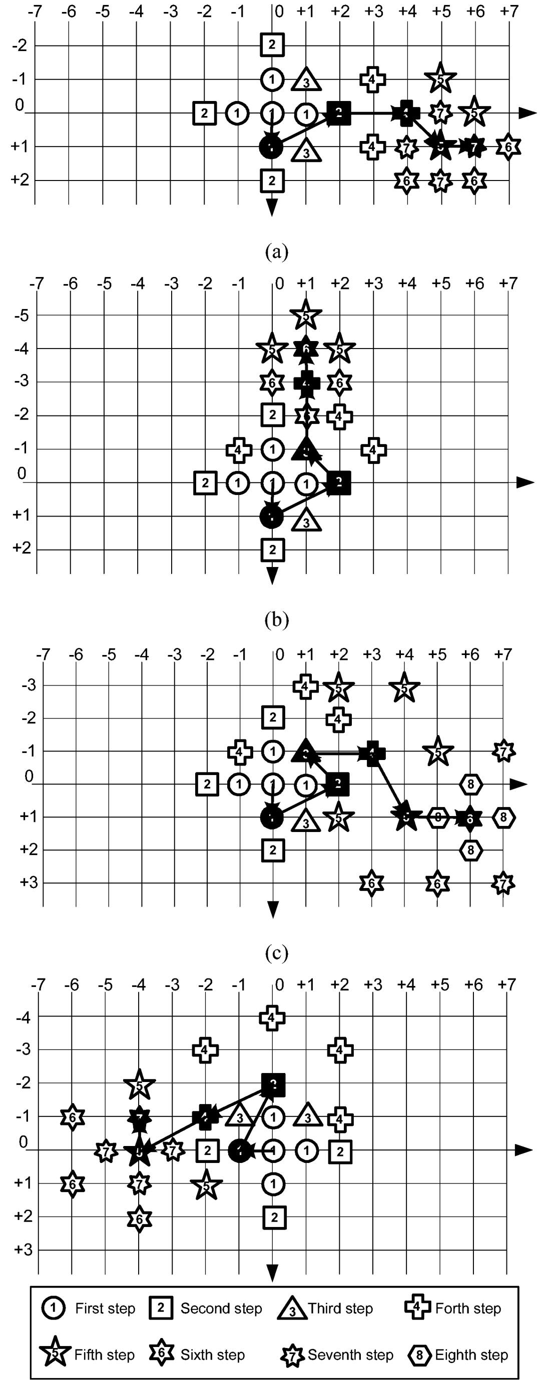
Step (iv) Searching:

– *Case (1):* If LDSP is used in previous step and the minimum BDM is found located at any point on diamond edge, a new LDSP is formed by repositioning the previous minimum BDM point as the center of LDSP.

– *Case (2):* If LDSP is used in previous step and the minimum BDM is found located at either of the horizontal (vertical) diamond corners, a new horizontal (vertical) LHSP is formed by repositioning the previous minimum BDM as the center of LHSP.

– *Case (3):* Otherwise, a new LHSP of the same shape is formed by repositioning the previous minimum BDM as the center of LHSP.

For any case above (LDSP → LDSP, LDSP → LHSP, or LHSP → LHSP), three new checking points[1] are evaluated.



(d)

Fig. 6. Each candidate point is marked with the corresponding step number, in which only one is found to be the minimum BDM point (filled). Examples of unrestricted search path of CDHS-F for (a) $\mathrm{MV}(+6, +1)$ and (b) $\mathrm{MV}(+1, -4)$, and search path of CDHS-T for (c) $\mathrm{MV}(+6, +1)$ and (d) $\mathrm{MV}(-4, -1)$.

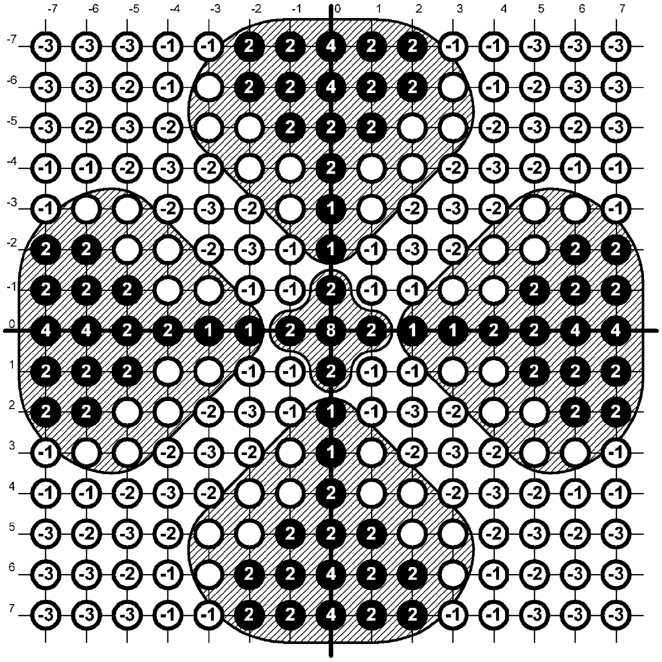
If the new minimum BDM point is still at the center of the newly formed LDSP or LHSP, go to the final Step (v) (Ending). Otherwise, this step is repeated again.

Step (v) Ending: With the minimum BDM point in the previous step as the center, a new SDSP is formed if LDSP is used in previous step; otherwise, a SHSP is employed instead. Identify the new minimum BDM point, which is the final motion vector, from the four new candidate points[2] in SDSP or SHSP.

---

[1] In CDHS-T, an exceptional case of not using three extra new points exists in the transitional step from LDSP to LHSP. There are five new search points in conjunction of newly formed LHSP (HT-HSP or VT-HSP) if the previous minimum BDM is found at the diamond corner. Afterwards, the search keeps advancing with three extra points, as shown in Fig. 6(c) – step 5 and Fig. 6(d) – step 4.

[2] Some points have been checked in previous step(s).



	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
-7	-3	-3	-3	-1	-1	2	2	4	2	2	-1	-1	-3	-3	-3
-6	-3	-3	-2	-1		2	2	4	2	2		-1	-2	-3	-3
-5	-3	-2	-3	-2			2	2	2			-2	-3	-2	-3
-4	-1	-1	-2	-3	-2			2			-2	-3	-2	-1	-1
-3	-1			-2	-3	-2		1		-2	-3	-2			-1
-2	2	2			-2	-3	-1	1	-1	-3	-2			2	2
-1	2	2	2			-1	-1	2	-1	-1			2	2	2
0	4	4	2	2	1	1	2	8	2	1	1	2	2	4	4
1	2	2	2			-1	-1	2	-1	-1			2	2	2
2	2	2			-2	-3	-1	1	-1	-3	-2			2	2
3	-1			-2	-3	-2		1		-2	-3	-2			-1
4	-1	-1	-2	-3	-2			2			-2	-3	-2	-1	-1
5	-3	-2	-3	-2			2	2	2			-2	-3	-2	-3
6	-3	-3	-2	-1		2	2	4	2	2		-1	-2	-3	-3
7	-3	-3	-3	-1	-1	2	2	4	2	2	-1	-1	-3	-3	-3

Fig. 7. Gain in search points using CDHS-F over DS.

Two halfway-stop examples for small motion paths are shown in Fig. 5 and four typical search paths using CDHS-F and CDHS-T are shown in Fig. 6. In Fig. 6(a), the search replaces LDSP pattern with HF-HSP and gives final motion vector at $\mathrm{MV}(+6,+1)$. Similarly, another example using CDHS-F with final $\mathrm{MV}(+1,-4)$ is shown in Fig. 6(b), in which the search initially follows diagonal direction using LDSP and turn into vertical direction using VF-HSP at a vertical diamond corner. Both proposed CDHS-F and CDHS-T will stop and truncate the search pattern at boundaries of search area. For instance, Fig. 6(c) shows an example of CDHS-T with HT-HSP truncated at seventh step and the final $\mathrm{MV}(+6,+1)$ is identified by the minimum BDM point, which coincides at the center of patterns used in sixth and eighth step. Another example using CDHS-T with final $\mathrm{MV}(-4,-1)$ is shown in Fig. 6(d). It is noted that the proposed CDHS-T requires two more checking points in the transitional step [Step (iv), Case (2)] when switching into LHSP, as shown in Fig. 6(c) and (d).

B. Theoretical Analysis

The maximum theoretical speedup of the proposed CDHSs is about 20.5–45.0 times with window size of 7, as only five and 11 points are checked, depicted in Fig. 5(a) and (b). After utilizing cross-center-biased (CCB) characteristics, CDHSs can further reduce the computations by checking three more points consistently in advancing steps. The gain in number of search point N_s can be calculated similarly [19] by

$$\text{Gain in } N_s = \sum_{p \in w} n_p P_p. \tag{1}$$

TABLE IV
AVERAGE GAIN IN N_s PER BLOCK USING CDHSS OVER OTHER BMAS

	CIF/SIF			QCIF		
Gain in N_s	4SS	DS	CDS	4SS	DS	CDS
CDHS-T	6.04	3.18	1.97	9.50	5.98	3.17
CDHS-F	6.42	3.56	2.36	9.64	6.13	3.31

TABLE V
PERFORMANCE COMPARISONS OF CDHSS ON SEQUENCE "SALES" (CIF)

BMA	N_S	Speedup	MAD	Distance[a]	Prob.[b]
FS	204.283	1.000	2.864	0.000	100.000
3SS	23.212	8.801	2.916	0.246	94.710
N3SS	16.935	12.063	2.872	0.076	98.020
4SS	16.206	12.605	2.908	0.207	95.440
DS	13.019	15.692	2.907	0.215	95.120
CDS	9.495	21.514	2.874	0.083	97.830
CDHS-T	**6.963**	**29.337**	**2.876**	**0.086**	**97.550**
CDHS-F	**6.927**	**29.492**	**2.876**	**0.086**	**97.720**

[a] the average distance from the true MV;
[b] the probability of finding the true MV, which is obtained using FS.

TABLE VI
AVERAGE SPEED IMPROVEMENT RATE (SIR) (%) AND AVERAGE MAD CHANGED (%) OF CDHS-F OVER DS AND OVER CDS

Sequence	CDHS-F OVER DS (%)		CDHS-F over CDS (%)	
	SIR	MAD	SIR	MAD
Miss America[a]	57.7 / 91.0	**-2.7** / 0.3	13.3 / 43.8	0.1 / **0.0**
sales[a]	87.9 / 142.8	**-1.1** / 0.1	37.1 / 74.9	0.1 / **0.0**
Claire[a]	131.7/ 136.5	0.2 / 0.1	66.6 / 71.1	**0.0 / 0.0**
akiyo[a]	**143.7 / 145.9**	**0.0 / 0.0**	**72.9 / 76.6**	**0.0 / 0.0**
container[a]	140.9 / 145.9	0.1 / **0.0**	71.2 / 76.7	0.1 / **0.0**
mother_daughter[a]	77.5 / 114.2	0.5 / 0.2	38.4 / 59.0	0.3 / 0.1
coastguard[a]	*18.0* / 40.2	0.3 / 0.2	*13.0* / 12.1	0.3 / 0.1
foreman[a]	32.3 / 63.4	3.9 / 0.3	21.4 / 29.4	3.5 / 0.2
silent[a]	100.3 / 117.6	1.7 / 0.7	53.0 / 62.6	1.2 / 0.4
mobile[a]	41.7 / 119.2	**-0.1 / 0.0**	9.9 / 60.9	**0.0 / 0.0**
table[a]	64.7 / 56.0	2.5 / 1.4	37.8 / 28.4	1.6 / 0.8
stefan[a]	29.6 / 35.7	2.4 / 2.2	24.3 / 23.1	1.7 / 1.4
garden[b]	22.8 / *15.7*	**-0.1** / 3.9	10.1 / 20.7	0.4 / 3.7
football[b]	74.5 / 31.9	1.1 / 3.8	39.9 / 24.8	0.7 / 3.4
tennis[b]	29.2 / 47.0	4.1 / 2.9	22.3 /32.2	3.2 / 2.4

[a] $\mathrm{X/Y}$ = SIR using CIF/SIR using QCIF sequence.
[b] $\mathrm{X/Y}$ = SIR using SIF/SIR using CCIR601 sequence.

Therefore, the average gain using CDHS-F over DS with window size of 7 is 3.56 search points per block,[3] by (1) and Fig. 7. Table IV summarizes all the theoretical gains of CDHSs over 4SS, DS and CDS. It shows that the gain over 4SS is

---

[3] In detail, the gain in Ns by CDHS-F over

$$\begin{aligned}\mathrm{DS} &= 0.4544(8) + 0.2641\left(\frac{4}{8}\right) + 0.0952\left(-\frac{16}{16}\right)\\ &\quad + 0.0776\left(-\frac{24}{24}\right) + 0.0327\left(-\frac{20}{32}\right) + 0.0226\left(-\frac{4}{40}\right)\\ &\quad + 0.0223\left(\frac{12}{48}\right) + 0.0268\left(-\frac{28}{56}\right)\\ &= 3.56 \text{ search points per block.}\end{aligned}$$

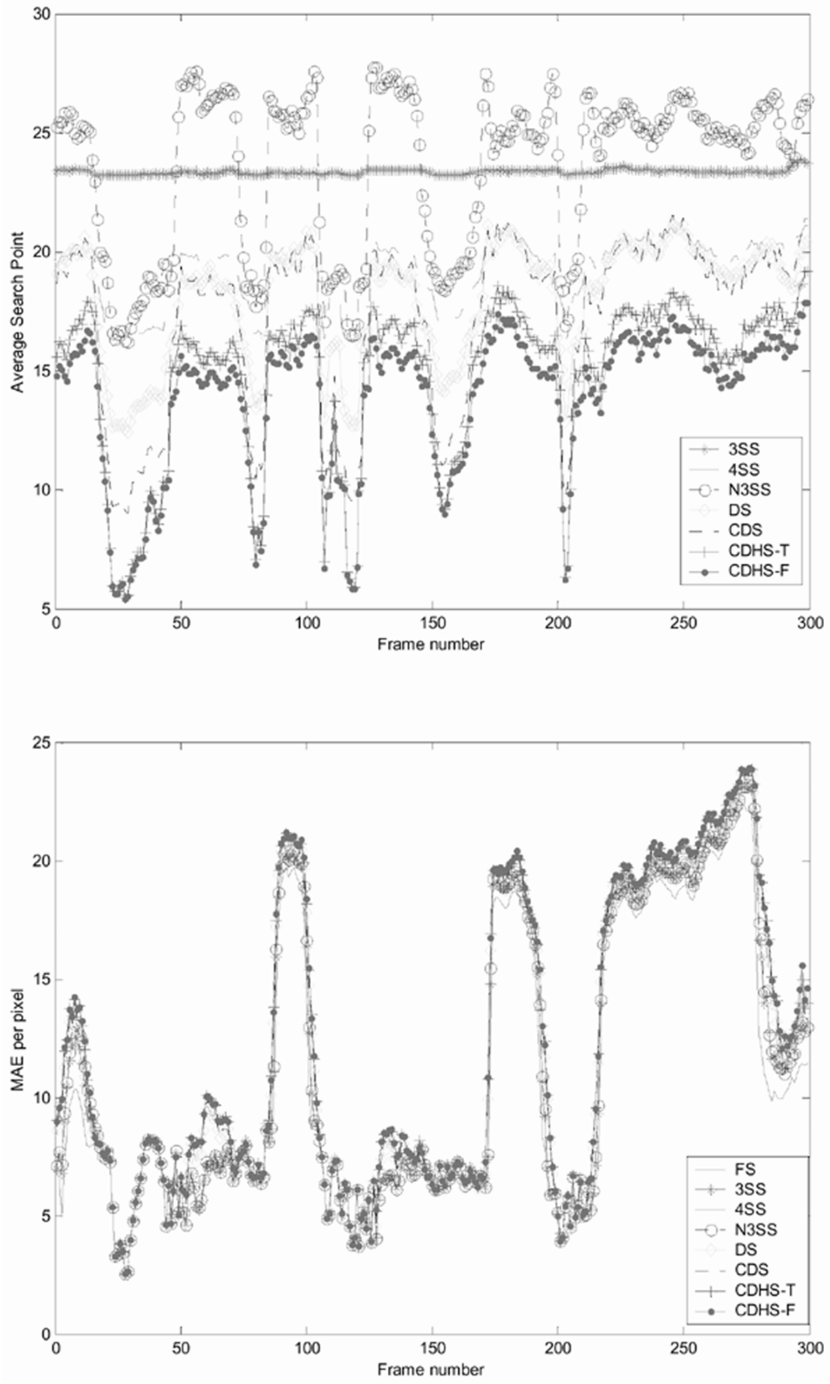


Fig. 8. Frame-wise performance comparison between different BMAs on MPEG-4 high-motion CIF sequence "stefan" by average number of search points and MAE performance per pixel.

larger than that of DS, and is larger than CDS's. It also shows that CDHS-F generally checks fewer points than CDHS-T, i.e., number of candidate points checked: CDHS-F $<$ CDHS-T $<$ CDS $<$ DS $<$ 4SS.

IV. EXPERIMENTAL RESULTS

The proposed CDHS algorithms are simulated using the luminance of the testing sequences listed in Table I. Block size of 16 and sum of absolute difference (SAD) as the BDM are used. Window size is set to ± 7 for SIF/CIF/QCIF sequences and ± 15 for CCIR601s. For CCIR601s, half-pel motion estimation is also applied. Table V shows that CDHS-F gives the fastest speed amongst all BMAs with similar or better in both quality and accuracy using sequence "sales."

We can also see CDHS-F performs slightly faster than CDHS-T. As space is limited here, Table VI summarizes the performance of CDHS-F over DS and CDS when using different sequences listed in Table I. It shows that the speedup gain using CDHS-F over DS is up to about 144% and gain over CDS is about 73%. In addition, CDHS-F can also provide even better prediction quality, indicated by negative values, e.g., 2.7% lesser MAD error than DS's. Fig. 8 plots the average number of points used and the MAD per pixel frame-by-frame using the vigorous sequence "stefan." We can see that the points used by CDHSs are generally the fewest and the quality still be maintained very close to other fast BMAs and FS's.

V. CONCLUSION

In this paper, we propose two fast cross-diamond-hexagonal search algorithms, namely CDHS-F and CDHS-T. They differ from each other in their hexagonal search pattern sizes. Both are suggested to firstly employ a small cross-shaped pattern to explore the cross-center-biased property of most real-world sequences. They then typically perform block matching as in DS, but switch using hexagonal search pattern in advancing steps. Experimental results show that our proposed CDHSs typically outperform other fast BMAs. In particular, they improve searching speed by about 144% faster than diamond search algorithm and about 73% faster than the cross-diamond search.

REFERENCES

[1] "Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s—Part 2: Video," JTC1/SC29/WG11, ISO/IEC 11 172-2 (MPEG-1 Video), 1993.

[2] "Generic Coding of Moving Pictures and Associated Audio Information—Part 2: Video," ITU-T and ISO/IEC JTC 1, ITU Rec. H.262 – ISO/IEC 13 818-2 (MPEG-2 Video), 1994.

[3] "Information Technology—Coding of Audio Visual Objects—Part 2 Visual," JTC1/SC29/WG11, ISO/IEC 14 469-2 (MPEG-4 Visual), 2000.

[4] "Video Codec for Audiovisual Services at $\mathrm{p} \times 64\ \mathrm{kbit/s}$," ITU-T SG15, ITU-T Rec. H.261, 2 ed., 1993.

[5] "Video Coding for Low Bit Rate Communication," ITU-T SG16, ITU-T Rec. H.263, 3rd ed., 2000.

[6] H. Schwarz and T. Wiegand, "The emerging JVT/H.26L video coding standard," in *Proc. of IBC 2002*, Amsterdam, The Netherlands, Sep. 2002.

[7] "Draft ITU-T Rec. and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264-ISO/IEC 14 496-10 AVC)," Joint Video Team (JVT) of ITU-T and ISO/IEC JTC1, Geneva, JVT of ISO/IEC MPEG and ITU-T VCEG, JVT-G050r1, 2003.

[8] 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 1981, pp. G5.3.1–G5.3.5.

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

[10] A. Puri, H. M. Hang, and D. L. Schilling, "An efficient block matching algorithm for motion compensated coding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1987, pp. 1063–1066.

[11] 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.

[12] 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, Jun. 1996.

[13] 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, no. 4, pp. 419–423, Aug. 1996.

[14] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," in *Proc. 1997 Int. Conf. Information, Communications and Signal Processing (ICICS)*, vol. 1, Sep. 9–12, 1997, pp. 292–296.

[15] 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.

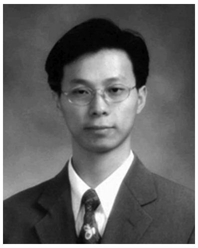
[16] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Trans. Image Process.*, vol. 9, no. 2, pp. 287–290, Feb. 2000.

[17] C. Zhu, X. Lin, L. P. Chau, K. P. Lim, H. A. Ang, and C. Y. Ong, "A novel hexagon-based search algorithm for fast block motion estimation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, May 2001.

[18] "CCITT SGXV, "Description of reference model 8 (RM8)"," in *Document 525, Working Party XV/4, Specialists Group on Coding for Visual Technology*, Jun. 1989.

[19] C. H. Cheung and L. M. Po, "A novel cross-diamond search algorithm for fast block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 12, pp. 1168–1177, Dec. 2002.

[20] Y. Nie and K. K. Ma, "Adaptive rood pattern search for fast block-matching motion estimation," *IEEE Trans. Image Process.*, vol. 11, no. 12, pp. 1442–1449, Dec. 2002.



Chun-Ho Cheung (M'03) received the B.Eng. (Hons) degree in computer engineering and the Ph.D degree in electronic engineering in 1996 and 2002, respectively, both from the City University of Hong Kong.

He joined the Hong Kong Institute of Technology in 2003, where he is currently Associate Professor and working with the Department of Computing and Information Technology. His research interests are in image compression, motion estimation/compensation and image retrievals.

Dr. Cheung was awarded the Student Prize and Best Student Scholarship from the Hong Kong Institute of Engineers, and the Team Concept Group of Scholarship, all in 1995.

Lai-Man Po (M'92) received the B.Sc. and Ph.D. degrees, both in electronic engineering, from the the City University of Hong Kong (CUHK) in 1988 and 1991, respectively.

Since November 1991, he has been with the Department of Electronic Engineering, CUHK, and is currently Associate Professor. His research interests are in vector quantization, motion estimation and image/video compression.

Dr. Po was awarded First Prize (1988) in the Paper Contest for Students and Non-corporate Members organized by the Institute of Electronics and Radio Engineers of Hong Kong, and a Postgraduate Fellowship of the Sir Edward Youde Memorial Council (1988 to 1991).