Cross Directional Rectangle Search for Fast Block-Matching Motion Estimation

Binh P. Nguyen, Trang T. T. Do
Department of Electrical and Computer Engineering
National University of Singapore, Singapore 117576
{phubinh, dothutrang}@nus.edu.sg

Abstract—**This paper proposes an efficient block-matching motion estimation algorithm, called cross directional rectangle search (CDRS), which can be applied to a wide range of video applications. The algorithm relies on the hypothesis of monotonic block distortion surface and the cross center-biased characteristic of motion vector probability distribution which has been used in literature. A cross pattern in the initial step and one of four possible directional rectangle search patterns in the next step are repeatedly used to find the motion vectors. The algorithm can achieve a faster computation speed with similar distortion performance compared to some other well-known fast block-matching motion estimation algorithms.**

Keywords-motion estimation; block-matching; cross center-biased; video coding; CDRS.

I. INTRODUCTION

In video coding, motion estimation is very important since it is used to efficiently reduce the temporal redundancy information between successive frames, thus improving the compression ratio. Among existing motion estimation methods, block-matching algorithms (BMAs) are widely used because of their simplicity and effectiveness. Block-matching aims to find, within a search window, the best-matched block from the previous frame based on a block distortion measure or other matching criteria. The displacement of the best-matched block is described as a motion vector to the block in the current frame. The algorithm that can find the best-matched block is the full search (FS) method, which evaluates all the candidate blocks within the search window. However, the high computational cost of FS limits its application in practice. In order to reduce the heavy computational complexity of FS, many fast BMAs have been proposed. Most fast BMAs exploit different search patterns to reduce the search points when finding the best-matched block, such as square patterns in the three-step search (TSS) [1], new three-step search (NTSS) [2], efficient three-step search (E3SS) [3], four-step search (4SS) [4], and block-based gradient descent search (BBGDS) [5]; cross patterns in the cross search algorithm (CSA) [6] and two-dimensional logarithmic search (TDLS) [7]; diamond patterns in diamond search (DS) [8] and cross-diamond search (CDS) [9]; and hexagon patterns in hexagon-based search (HEXBS) [10]. The primary assumption of these fast BMAs is that the block distortion decreases monotonically when the search position moves toward the minimum distortion point. Hence, it is not necessary to check all the search points in the search window since the best-matched position can be found by following the changing trend of the distortion.

In this paper, we propose a novel BMA, called cross directional rectangle search (CDRS), which can achieve substantial speed improvement over popular fast BMAs with similar distortion performance.

The rest of the paper is organized as follows. The new cross directional rectangle search algorithm is described in Section II. Section III presents the experimental results and the comparisons of our proposed algorithm with several popular fast BMAs. The paper concludes in Section IV.

II. CROSS DIRECTIONAL RECTANGLE SEARCH

Firstly, our algorithm is developed based on the cross center-biased characteristic of motion vector probability distribution which has been reported and widely used in many studies [6], [8-9]. It means that most motion vectors are very close to zero motion and the probability of the motion vector in the axis-aligned directions is larger than the other directions. Thus, the best search strategy is to search from the center of the search window and the nearest neighbors along the two axes. Secondly, the proposed algorithm also relies on the hypothesis of monotonic block distortion surface. This implies that the minimum distortion point can be found along the directions of the distortion from the higher points to the lowest point.

The proposed CDRS algorithm uses one cross pattern and four directional rectangle search patterns as depicted in Fig. 1. The algorithm comprises the following steps:

- Step 1: Apply the cross pattern at the center of the search window. Evaluate the matching distortion of all search points in this pattern. If the minimum matching distortion point (MMDP) occurs at the center, the search process stops and the motion vector is found at the center; otherwise, go to step 2.
- Step 2: Based on the position of the MMDP found in step 1, choose a proper directional rectangle search pattern accordingly to find the new MMDP. If the location of the MMDP remains unchanged, the search process stops with the motion vector found at the MMDP. If not, go back to step 1.

As seen from Fig. 1, five search points are processed in step 2. If step 2 is performed, there are five more points to be checked. From step 3 onwards, only a few points in the search patterns are evaluated since some of them have already been checked. Fig. 2 is an example of CDRS in which the motion vector (2, 2) is determined after processing 17 search points in 4 steps.

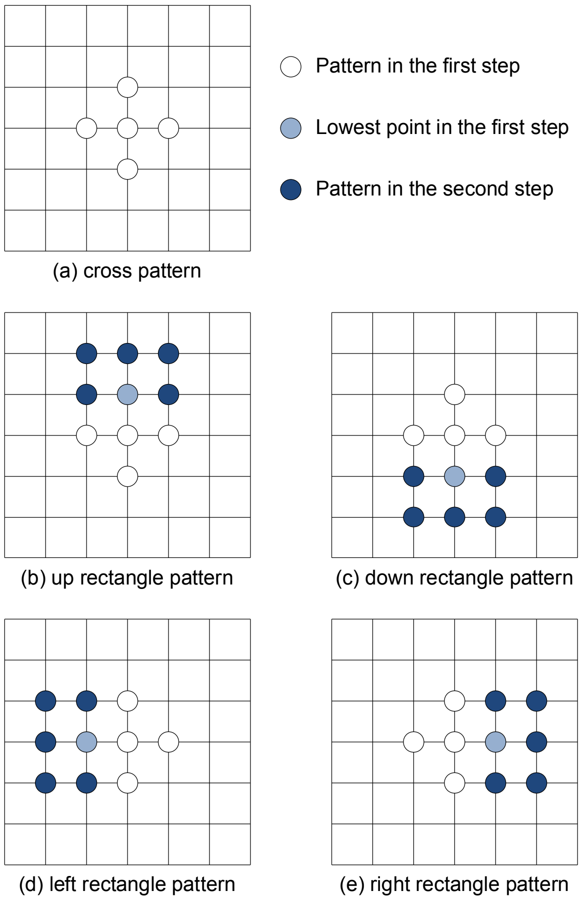

Figure 1. Search patterns of CDRS.

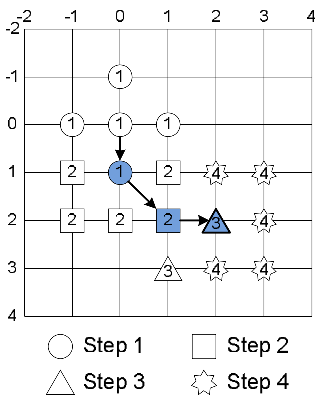

Figure 2. An example of CDRS.

III. EXPERIMENTAL RESULTS

In our simulations, mean square error (MSE) was used as the block distortion measure with block size of 16×16 and the search window size of 15×15 pixels. This means that each component of the motion vector ranges from -7 to $+7$. Twelve video sequences with difference motion speeds, frame sizes, frame rates, and lengths were used in the experiments. The descriptions of these datasets are given in Table I and II.

In addition to FS, five fast BMAs including NTSS, E3SS, DS, CDS, and HEXBS were implemented to compare their performances with the proposed algorithm. Three criteria were used for the comparisons: the average number of search points per block, the speed-up ratio with respect to FS, and the average of peak signal to noise ratio (PSNR) between predicted frames and original frames. Table III, IV and V summary the experimental results. As can be seen from these tables, the proposed CDRS out performs other algorithms in term of speed (measured by the number of search points) when it was applied to small and normal motion sequences, such as "Boat", "Dancing", "Island", and "Tourists". For large motion sequences, e.g. "Dolphins", "HorseRiding", and "Running", CDRS achieved the comparative computation speed with the fastest one, HEXBS. In all cases, CDRS produced the similar or above average PSNR compared to other fast BMAs in the experiments.

Fig. 3 illustrates the frame by frame comparison of PSNR and the average number of search points per block when applying NTSS, E3SS, DS, CDS, HEXBS and CDRS to the "Tourists" sequence. It clearly demonstrates the robustness of the proposed CDRS to other algorithms in term of speed and fidelity.

TABLE I. SPECIFICATIONS OF THE TESTING VIDEO SEQUENCES

Sequence	Frame size (pixels)	Frame rate (fps)	Number of frame
Anelka	1024 × 576	25.00	300
Baby	640 × 480	30.00	300
Boat	384 × 288	25.00	200
Coral	512 × 288	25.00	300
Dancing	528 × 304	25.00	300
Dolphins	384 × 288	25.00	300
Drogba	640 × 368	29.97	300
HorseRiding	608 × 256	23.98	95
Island	384 × 288	25.00	100
Running	576 × 240	25.00	300
Tourists	384 × 288	25.00	180
Waves	384 × 288	25.00	300

TABLE II. CHARACTERISTICS OF THE TESTING VIDEO SEQUENCES

Sequence	Motion content	Background changing	Camera zooming	Camera panning
Anelka	Fast	Normal	Slow	Slow
Baby	Normal	No	No	No
Boat	Slow	Slow	No	No
Coral	Slow	Slow	Slow	Slow
Dancing	Fast	Slow	No	Slow
Dolphins	Fast	Fast	No	Fast
Drogba	Fast	Normal	Slow	Slow
HorseRiding	Fast	Fast	No	Fast
Island	Slow	Slow	Slow	Slow
Running	Fast	Fast	Slow	Normal
Tourists	Normal	Normal	Normal	No
Waves	Fast	Normal	No	Normal

IV. CONCLUSION

We have proposed an efficient block-matching motion estimation algorithm which can be applied to a wide range of video applications. The algorithm uses one cross pattern and four directional rectangle search patterns to exploit the cross center-biased characteristic of motion vector probability distribution and the hypothesis of monotonic block distortion surface. Experimental results showed that our method can achieve a faster computation speed with similar distortion performance compared to other well-known BMAs, e.g., NTSS, E3SS, DS, CDS, and HEXBS.

TABLE III. AVERAGE NUMBER OF SEARCH POINTS PER BLOCK

	Anelka	*Baby*	*Boat*	*Coral*	*Dancing*	*Dolphins*	*Drogba*	*HorseRiding*	*Island*	*Running*	*Tourists*	*Waves*
FS	215.97	212.91	205.04	207.11	207.90	205.04	210.83	206.67	205.04	205.53	205.04	205.04
NTSS	22.83	20.45	18.11	19.70	18.89	28.05	20.03	28.60	19.32	26.62	19.73	24.05
E3SS	20.14	17.10	14.74	16.68	15.52	25.23	16.78	25.64	16.46	24.38	16.99	22.08
DS	17.96	15.60	13.94	15.28	14.62	24.07	15.46	23.79	15.28	22.02	15.40	19.90
CDS	18.47	13.80	10.83	13.20	11.89	27.90	13.66	29.03	13.90	25.87	13.74	22.59
HEXBS	13.92	12.36	11.34	12.14	11.74	16.07	12.26	16.65	12.33	15.84	12.27	14.73
CDRS	12.33	9.38	7.91	9.66	8.77	21.05	9.54	20.56	9.50	17.64	9.72	15.12

TABLE IV. SPEED-UP RATIO WITH RESPECT TO FS

	Anelka	*Baby*	*Boat*	*Coral*	*Dancing*	*Dolphins*	*Drogba*	*HorseRiding*	*Island*	*Running*	*Tourists*	*Waves*
FS	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
NTSS	9.46	10.41	11.32	10.51	11.01	7.31	10.53	7.23	10.61	7.72	10.39	8.53
E3SS	10.72	12.45	13.91	12.42	13.40	8.13	12.56	8.06	12.46	8.43	12.07	9.29
DS	12.03	13.65	14.71	13.55	14.22	8.52	13.64	8.69	13.42	9.33	13.31	10.30
CDS	11.69	15.43	18.93	15.69	17.49	7.35	15.43	7.12	14.75	7.94	14.92	9.08
HEXBS	15.52	17.23	18.08	17.06	17.71	**12.76**	17.20	**12.41**	16.63	**12.98**	16.71	**13.92**
CDRS	**17.52**	**22.70**	**25.92**	**21.44**	**23.71**	9.74	**22.10**	10.05	**21.58**	11.65	**21.09**	13.56

TABLE V. AVERAGE PSNR PER FRAME

	Anelka	*Baby*	*Boat*	*Coral*	*Dancing*	*Dolphins*	*Drogba*	*HorseRiding*	*Island*	*Running*	*Tourists*	*Waves*
FS	32.76	35.94	38.81	31.65	30.89	25.86	33.26	23.83	34.75	29.41	24.84	31.99
NTSS	32.40	35.84	38.73	31.59	30.74	25.63	33.18	23.54	34.69	28.76	24.74	31.27
E3SS	32.56	35.82	38.64	31.58	30.74	25.66	33.19	23.57	34.63	28.84	24.71	31.64
DS	32.58	35.84	38.64	31.62	30.71	25.49	33.13	23.25	34.72	28.51	24.77	31.56
CDS	32.30	35.74	38.20	31.19	30.60	25.39	33.06	23.19	34.19	28.26	24.37	31.47
HEXBS	32.19	35.67	38.12	31.17	30.54	25.38	32.96	23.13	33.22	28.20	24.40	31.14
CDRS	32.47	35.76	38.64	31.55	30.56	25.34	33.03	23.00	34.64	28.25	24.67	31.38

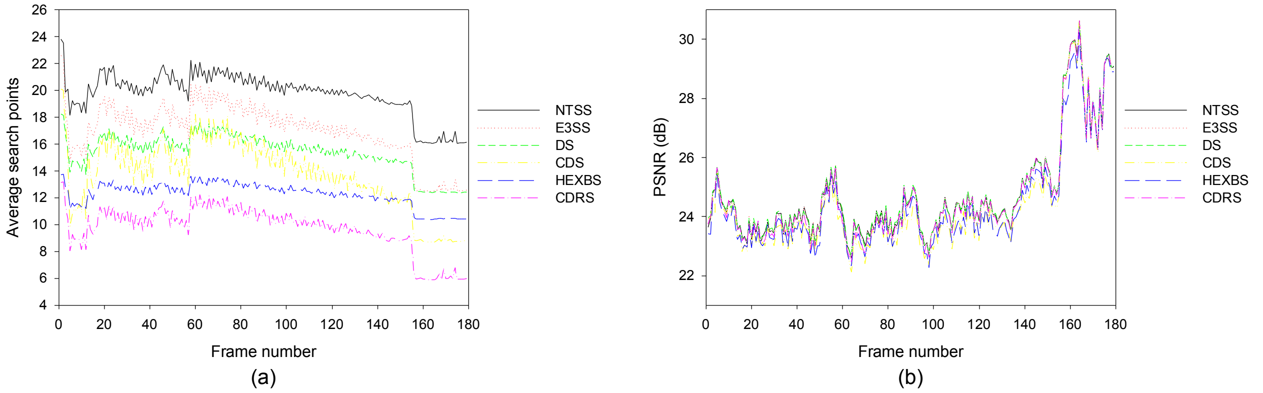

Figure 3. Frame by frame comparisons for "Tourists" sequence: (a) Average search points per block; (b) PSNR per frame.

REFERENCES

[1] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," in Proceedings of the National Telecommunications Conference, 1981, pp. G5.3.1–G5.3.5.

[2] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 4, pp. 438–442, 1994.

[3] X. Jing and L.-P. Chau, "An efficient three-step search algorithm for block motion estimation," IEEE Transactions on Multimedia, vol. 6, pp. 435–438, 2004.

[4] L. M. Po, and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, pp. 313–317, 1996.

[5] L.-K. Liu and E. Feig, "A block-based gradient descent search algorithm for block motion estimation in video coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, pp. 419–422, 1996.

[6] M. Ghanbari, "The cross-search algorithm for motion estimation," IEEE Transactions on Communications, vol. 38, pp. 950–953, 1990.

[7] J. Jain and A. Jain, "Displacement measurement and its application in interframe image coding," IEEE Transactions on Communications, vol. 29, pp. 1799–1808, 1981.

[8] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block matching motion estimation," in Proceedings of the International Conference on Information, Communications and Signal Processing 1997 (ICICS'97), 1997, vol. 1, pp. 292–296.

[9] C.-H. Cheung and L.-M. Po, "A novel cross-diamond search algorithm for fast block motion estimation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 12, pp. 1168–1177, 2002.

[10] C. Zhu, X. Lin, and L.-P. Chau, "Hexagon-based search pattern for fast block motion estimation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 12, pp. 349–355, 2002.