SEARCH PATTERNS SWITCHING FOR MOTION ESTIMATION USING RATE OF ERROR DESCENT

Ka-Ho Ng, Lai-Man Po and Ka-Man Wong

mike.ng@student.cityu.edu.hk, eelmpo@cityu.edu.hk and kmwong@ee.cityu.edu.hk

Department of Electronic Engineering, City University of Hong Kong,

Kowloon, Hong Kong

ABSTRACT

Most of the fast motion estimation algorithms based on search-point pattern are only good at handling videos with small motions, for example, block-based gradient descent search, diamond search and hexagonal-based search. An adaptive motion estimation algorithm which can switch between search patterns for different video contents should work better than a single search pattern algorithm. In this paper, a simple classifier based on Error Descent Rate (EDR) is proposed. This classifier uses a very few number of search points to predict whether the global minimum is far away or near the center of the search window. If it is far away, a search pattern which is good at searching large motions is used. Otherwise, a pattern good at searching small motions is applied. The proposed *search patterns switching* (SPS) algorithm performs well for all kinds of video contents.

1. INTRODUCTION

Block Matching Algorithms (BMAs) have been most widely used in motion estimation for various video coding standards such as H.26X series and MPEG series. Motion Estimation (ME) is used to obtain the motion vectors (MV) that match the actual motion content. Full Search (FS) can be used to obtain the optimal MV since it searches all the possible positions and chooses the MV with minimal block distortion. Due to the huge number of search points in the searching area, ME becomes the most computational intensive part in the whole video coding process. For the requirements of real time application, the computational complexity of ME should be reduced. Many BMAs are proposed to speed up the ME process by reducing the number of search points based on two major strategies. One is coarse-to-fine search and another is to make use of the center-biased MV characteristic.

To reduce the number of search points, three-step search (3SS) [1] and 2D-logarithmic search (2DLOG) [2] are proposed which use a coarse-to-fine search approach. The coarse searches can quickly locate the rough position of the global minimum and the fine search is a high resolution search which finds the best motion vector. These methods perform well for large motion videos.

In an analysis on motion vector distribution in [5], about half of all the macro-blocks are stationary, and up to 90% of motion vectors are within the ± 2 region of the search window. To utilize this center-biased MV property, four-step search (4SS) [3], unrestricted center-biased diamond search (DS) [4], cross diamond search (CDS) [5] and block based gradient descent search (BBGDS) [6] were proposed. All of them start with a centre-biased initial search pattern. The search will finish when the minimum distortion point is at the center of this search pattern. This is known as the first-step-stop mechanism. This technique enables a substantial reduction of search points for small motion videos. Normally, these algorithms perform very well on small motion sequences. However, they are subject to quality degradation for complex motion sequences.

Non-adaptive algorithm is difficult to handle both types of videos very well. For example, with a ± 7 search window, the 3SS needs 25 search points even though it is a stationary block where CDS only needs 5 points. In contrary, CDS may need more than 25 search points for a large motion block where 3SS only needs 25 in all cases. Moreover, both large and small motions can coexist in one video sequence, or even in one frame. To handle both types of motion contents, an adaptive approach which uses a different search pattern depending on the motion content will be a better solution. In this paper, a search patterns switching (SPS) algorithm is proposed which uses a very simple Error Descent Rate (EDR) to estimate the distance between the global minimum and the center of the search window. The EDR will be discussed in detail in section 2. Section 3 describes the SPS algorithm and section 4 shows experimental results which prove that the SPS is better than other fast BMAs. The paper is ended with a short conclusion.

2. ERROR DESCENT RATE

By the unimodal error surface assumption, distortion error monotonically decreases towards the global minimum. Most fast motion estimations make use of this assumption. We can extend this by assuming that a global minimum point affects its nearby error surface more than the error surface further away from it. The effect is reflected by the decrease in the distortion error. The nearer the global minimum point is, the faster the decrease in distortion error. The rate of decrease in distortion error can be quantified by the slope of the distortion error curve.
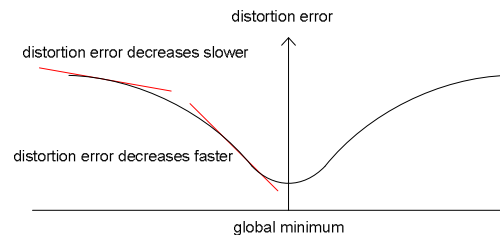

Fig. 1.   Decrease in distortion error faster near a global minimum point

Using the same assumption, we think that the position of a global minimum in a motion estimation search window affects the decrease of distortion at the search window center area. Fig. 2 shows 4 hypothetical positions of a global minimum.

We estimate the decrease of distortion near the search window center using the slope of the curve within the distance *d*. If the global minimum is at position 1, which is nearest to the search window center, the slope of the curve roughly equals *(B-A)/d*. If it is at position 2, the slope equals *(C-B)/d*, whereas at position 3 and 4, the slopes are *(D-C)/d* and *(E-D)/d* respectively.
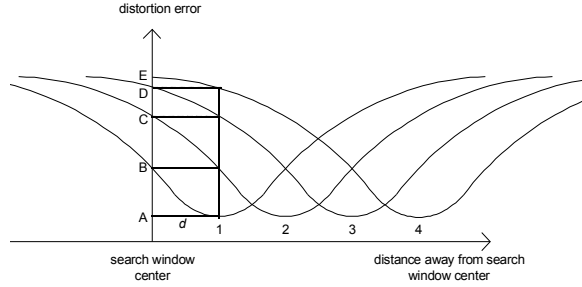
Fig. 2. The position of a global minimum affects the slope of error descent around the search window center

By comparing the slopes, we can estimate the distance of the global minimum from the search window center. This information is very important for motion estimation because we can use different search patterns for different extents of motion. This is the basis of our search patterns switching (SPS) algorithm which will be discussed in the next section.

The error descent slopes at the center of the search window can be estimated by the following method. First we find the distortion error of the search window center point A and note it as D_a. Then we calculate the distortions of its 4 adjacent points. Among the 4 points, the point with the minimum distortion is recognized as point B and its distortion is noted as D_b. This is shown in Fig. 3. The slope of error descent is then calculated as (D_b-D_a), as all of the 4 adjacent points are 1 pixel away from the center point.
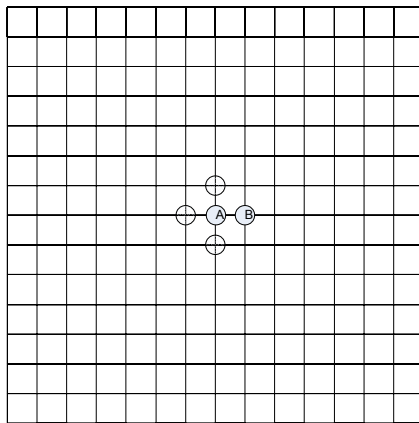

Fig. 3. Find the distortion difference between the center point and its minimum distortion neighbor

By the above method, the steepest error descent slope is used to estimate whether the global minimum is near or far away from the search window center. Fig. 4(a) and 4(b) are plots of the average value of (D_b-D_a) in Mean Absolute Error (MAE) against the Euclidean distance in pixels between the global minimum and the search window center, for the test video sequences *Stefan* and *Foreman* respectively.

We can observe an increasing tread of (D_b-D_a), for global minima close by the search window center to those further away. However it is not easy for us to make use of the values (D_b-D_a), because they are very much video content dependent. For example in *Foreman,* the average value of distortion error is lower than that in
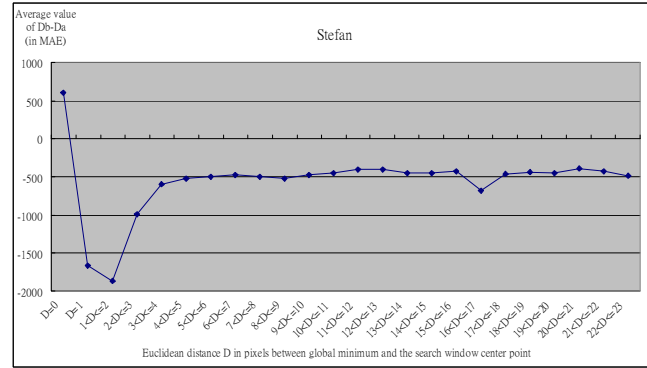
(a)
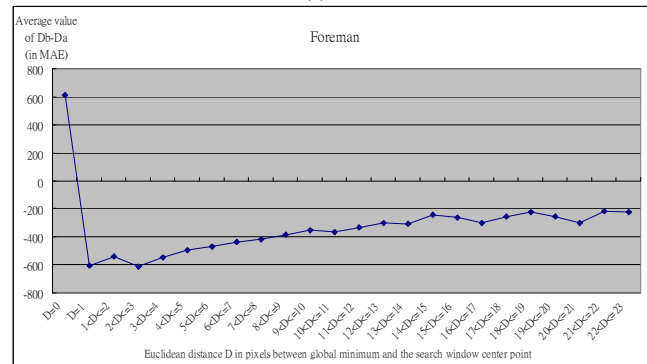
(b)

Fig. 4. Average (D_b-D_a) versus the Euclidean distance between the global minima and the search window center for 4(a) *Stefan* and 4(b) *Foreman*.

Stefan. It is because *Stefan* has complex motion contents while *foreman* has medium motion activities.

We define the Error Descent Rate (EDR) as (D_b / D_a). For example if $D_b= 60$ and $D_a=100$, the error descent rate is 0.6. In other words, the distortion error at point B is 40% less than the distortion error at search window center point A. Comparing to an error descent rate of 0.8, which is a 20% drop of distortion error, a 40% drop is surely a faster error drop than a 20% drop. We can consider the EDR as a normalized slope of error descent at the search window center. The EDR is video content independent, because it is a relative ratio between the distortion errors at different positions of a distortion error surface. Fig. 5 plots the average value of EDR against the Euclidean distance between the global minimum and the search window center, for several test video sequences using Full Search. Zero Motion Vectors (ZMVs) are omitted because the ZMVs of all the test video sequences have average EDRs very much larger than 1. From Fig. 5, we can see that the EDR, which is the normalized slope of error descent at the search window center, is a good indicator of the proximity of the global minimum to the search window center. The tread is also video-content independent. That is, all the test video sequences, which include small, medium, and large motion contents, show the same tread of increasing EDR value with increasing MV length.
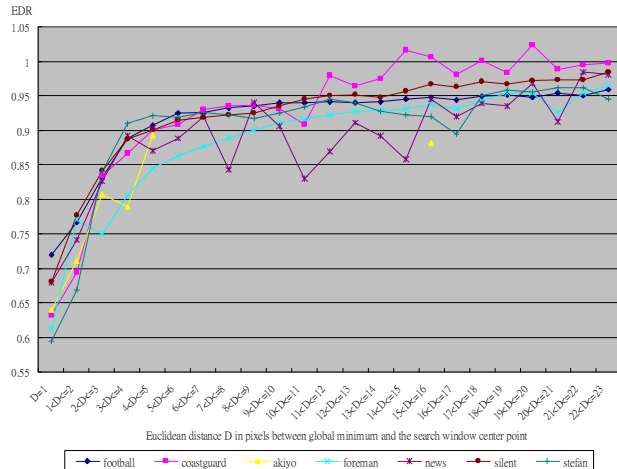
Fig. 5.   Average Error Distortion Rate (EDR) versus the Euclidean distance between the global minima and the search window center points for *Football, Coastguard, Akiyo, Foreman, News, Silent,* and S*tefan.*

We can see that most video sequences have global minima within a Euclidean distance of 2 pixels if their EDR is below 0.8. In other words, if the distortion error has a more than 20% descent at the center of the search window, the global minimum is usually near to the search window center. For global minima with Euclidean distance of more than 16 pixels from the center, they usually have an EDR >0.95, which is a gentle 5% error descent at the center. EDR is a good estimator of the distance between the global minimum and the search window center.

3. SEARCH PATTERNS SWITCHING

Based on the analysis in the previous section, we propose a search patterns switching (SPS) motion estimation algorithm. This algorithm uses the error descent rate EDR at the center of the search window to estimate whether the global minimum would be near the center or would be at the outer area of the search window. If it is estimated to be near the center, a search pattern suitable for small motion searching will be used. Otherwise a search pattern apt at searching large motions will be applied instead. The flow diagram is shown in Fig. 6.

EDR is calculated at the search window center as described in the last section. If EDR ≥ 1, all the 4 immediately adjacent points have distortion higher than the search window center point. Previous research [5] shows that a large percentage of MVs are within 1 pixel distance from the search window center, especially those in static videos. Therefore if EDR ≥ 1, the search stops and a ZMV is returned.

If EDR is higher than a threshold T, for example T=0.9 (in other words, there is a less than 10% distortion drop at the center of the search window), the global minimum has a high probability at the outer area of the search window. Three-step search (3SS), which is a search pattern apt at searching large motions, is applied. If EDR is lower than or equal to the threshold T=0.9 (in other words, there is a more than 10% distortion drop at the center of the search window), the global minimum is likely near the center of the search window. The block based gradient descent search (BBGDS), which is a search pattern apt at searching small motions, will be applied. Fig. 7(a) and 7(b) show examples of the subsequent search using 3SS and BBGDS respectively after EDR is calculated.
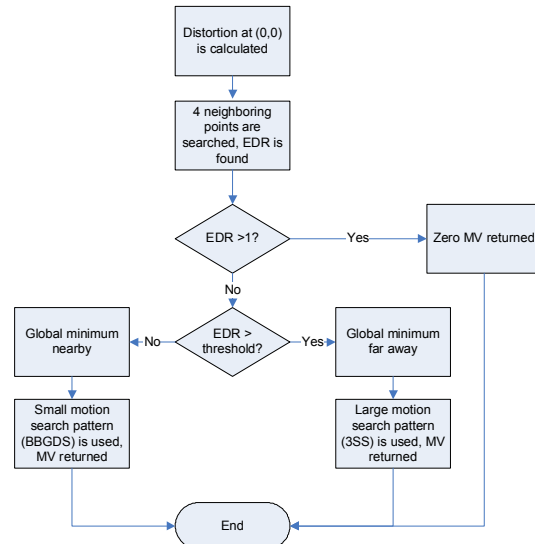
Fig. 6.   Flow diagram of the proposed search patterns switching (SPS) algorithm

(a)

(b)
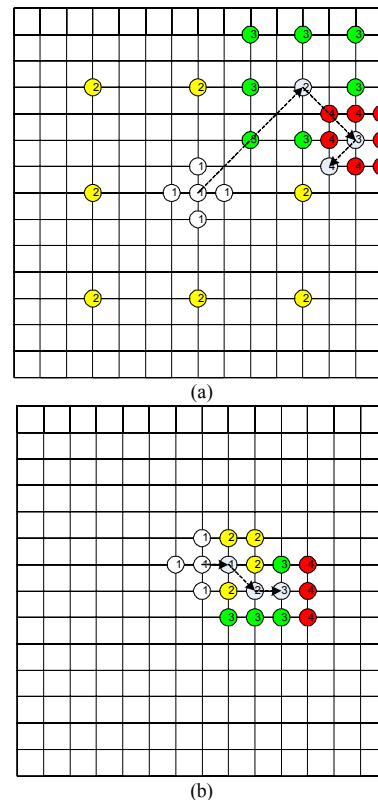
Fig. 7.   An example of 7(a) 3SS if EDR > threshold and 7(b) BBGDS if EDR $\leq$ threshold

The SPS algorithm can incorporate other small and large search combinations. For example we can use DS instead of BBGDS, 4SS instead of 3SS, etc. We denote the SPS combinations as SPS(X, Y), where X and Y are the names of the small and large search patterns respectively. Analyses and experiments show that the SPS (BBGDS, 3SS) is one of the best combinations.

4. EXPERIMENTAL RESULTS

Table 1: Performance comparison of different values of threshold T for SPS (BBGDS, 3SS)

threshold T	football PSNR (db)	# of search points	coastguard PSNR (db)	# of search points	akiyo PSNR (db)	# of search points	foreman PSNR (db)	# of search points	sean PSNR (db)	# of search points
0.600	25.075	24.019	29.858	26.130	43.011	5.571	32.241	21.749	39.419	8.030
0.650	25.078	23.821	29.859	24.062	43.011	5.521	32.248	20.814	39.420	7.952
0.700	25.083	23.576	29.859	21.389	43.014	5.471	32.258	19.788	39.420	7.859
0.750	25.087	23.226	29.860	18.645	43.015	5.421	32.270	18.722	39.420	7.766
0.800	25.099	22.747	29.866	16.402	43.015	5.377	32.284	17.660	39.419	7.674
0.850	25.114	22.047	29.877	14.472	43.018	5.328	32.295	16.635	39.420	7.563
0.900	**25.123**	**20.989**	**29.889**	**13.096**	**43.018**	**5.278**	**32.300**	**15.736**	**39.420**	**7.321**
0.950	25.025	19.832	29.864	12.310	43.020	5.231	32.243	14.984	39.416	6.700
0.975	24.871	19.296	29.837	12.044	43.022	5.203	32.138	14.710	39.414	6.215

threshold T	news PSNR (db)	# of search points	silent PSNR (db)	# of search points	stefan PSNR (db)	# of search points	mobile PSNR (db)	# of search points	container PSNR (db)	# of search points
0.600	36.672	7.863	35.617	8.678	24.246	21.879	24.256	14.191	38.449	6.070
0.650	36.674	7.706	35.618	8.534	24.247	21.431	24.256	13.379	38.449	6.070
0.700	36.681	7.512	35.624	8.366	24.250	20.896	24.255	12.617	38.449	6.070
0.750	36.693	7.262	35.628	8.158	24.251	20.302	24.256	11.860	38.449	6.070
0.800	36.702	7.009	35.632	7.919	24.249	19.565	24.255	11.120	38.449	6.070
0.850	36.706	6.765	35.640	7.657	24.231	18.575	24.256	10.403	38.449	6.067
0.900	**36.730**	**6.541**	**35.636**	**7.359**	**24.231**	**17.162**	**24.256**	**9.741**	**38.449**	**6.012**
0.950	36.694	6.307	35.593	7.099	24.036	15.236	24.254	9.108	38.449	5.744
0.975	36.673	6.178	35.528	6.968	23.770	14.141	24.252	8.799	38.449	5.488

We believe there is an optimum threshold value T for each SPS combination. Table 1 tabulates the average PSNR per frame and average number of search points used per macro-block for different test sequences, using different values of threshold T. The frames are of size CIF. The macro-blocks are of size 16x16 pixels and the search window is of size ± 16 pixels.

From the table we can see a general trend of decreasing number of search points used with the increasing values of threshold T. The higher the threshold we set, the more the blocks will be classified as small motion blocks. BBGDS will be used, which is more easily being trapped in a local minimum near the search window center instead of finding the global minimum. Although the number of search points needed is fewer, the quality degradation is severe. If a lower threshold value is set, extra number of search points is needed because the large motion search 3SS search converges slowly for small motions. We can see the threshold T=0.9 is an optimum threshold for SPS (BBGDS, 3SS). Higher than this value, not many search points can be saved but there are obvious PSNR degradations in some large motion videos.

This empirical method of finding the EDR threshold for different SPS combinations works very well, because we can find an optimum threshold value which works well in videos of a wide range of motion contents, for different SPS combinations we experimented with.

Table 2 compares the performance of SPS (BBGDS, 3SS) with threshold value T=0.9, against FS, 3SS, 4SS, hexagon-based search (HEXBS) [6], BBGDS, CDS, and DS, using the same experimental setup mentioned above.

SPS (BBGDS, 3SS) has the highest PSNR quality in *Football*, *Coastguard*, *Foreman*, and *News*, while the number of search points it uses are comparable to the faster algorithms in these sequences. It uses fewest search points in *Akiyo*, *Sean*, *News*, *Silent*, *Mobile*, and *Container*. In *Akiyo*, *Sean*, and *Mobile*, its PSNR is only around 0.02dB lower than the best ones. 4SS has the highest PSNR in *Silent* and *Stefan* among all algorithms. They are 0.246dB and 0.388 dB higher than SPS respectively. However, it needs 16.396 and 8.78 more search points than SPS.

Table 2: Performance comparison between SPS and other popular fast motion estimation algorithms

	football PSNR (db)	# of search points	coastguard PSNR (db)	# of search points	akiyo PSNR (db)	# of search points	foreman PSNR (db)	# of search points	sean PSNR (db)	# of search points
FS	25.844	858.455	30.035	868.333	43.044	868.333	32.842	868.333	39.461	868.333
2DLOG	24.905	33.000	29.397	33.000	42.919	33.000	31.920	33.000	39.314	33.000
4SS	25.055	27.478	29.756	26.015	42.939	23.240	31.998	26.289	39.332	23.444
HEX	24.661	16.144	29.785	13.043	42.817	10.342	31.420	13.709	39.296	10.476
BBGDS	24.709	20.760	29.809	13.718	43.040	8.521	32.140	16.725	39.447	8.924
CDS	24.828	21.502	29.879	15.682	43.011	8.700	32.014	16.686	39.408	9.151
DS	24.881	21.823	29.879	16.541	43.025	11.286	32.096	17.925	39.423	11.655
SPS(BBGDS,3SS)	**25.123**	**20.989**	**29.889**	**13.096**	**43.018**	**5.278**	**32.300**	**15.736**	**39.420**	**7.321**

	news PSNR (db)	# of search points	silent PSNR (db)	# of search points	stefan PSNR (db)	# of search points	mobile PSNR (db)	# of search points	container PSNR (db)	# of search points
FS	36.966	868.333	36.236	868.333	25.677	868.333	24.336	868.333	38.450	868.333
2DLOG	36.607	33.000	35.845	33.000	24.042	33.000	23.948	33.000	38.449	33.000
4SS	36.699	23.539	35.882	23.755	24.619	25.942	24.028	23.698	38.449	23.337
HEX	36.546	10.665	35.563	10.960	23.932	13.873	23.999	10.847	38.449	10.420
BBGDS	36.660	9.247	35.560	9.976	23.575	15.516	24.276	10.673	38.449	8.582
CDS	36.677	9.523	35.567	10.137	23.952	16.958	24.251	10.340	38.449	8.838
DS	36.694	11.898	35.663	12.464	24.033	17.608	24.244	12.814	38.449	11.415
SPS(BBGDS,3SS)	**36.730**	**6.541**	**35.636**	**7.359**	**24.231**	**17.162**	**24.256**	**9.741**	**38.449**	**6.012**

5. CONCLUSION

In this paper, 2 new ideas are proposed. First, it is found that the Error Descent Rate (EDR) gives a good estimation of the distance from the search window center to the global minimum. Based on this, a search patterns switching fast motion estimation algorithm (SPS) is proposed, which uses only 5 initial search points to calculate the EDR of the center of the search window. This EDR predicts whether the global minimum is near the search window center or not. If it is near, BBGDS, which is good at searching small motions, is used. Otherwise, 3SS, which is good at searching large motions, is used. SPS algorithm achieves high visual quality and is very robust.

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.,NO, L.A.,* Nov-Dec 1981, pp. G5.3.1-G.5.3.5.

[2] 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.

[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] 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.

[5] C. H. Cheung and L. M. Po, "A Novel Cross-Diamond Search Algorithm for Fast Block Motion Estimation," *IEEE Trans. on Circuits and Systems for Video Technology*, Vol.12, No. 12, pp. 1168-1177, Dec. 2002.

[6] 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. 419-422, Aug 1996.

[7] 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.