Journal of Engineering and Applied Sciences 2 (9): 1361-1364, 2007
ISSN: 1816-949X

Adaptive Distortion-Based Partial Distortion Search for Fast Motion Estimation

[1]Jong-Ho Kim and [2]Byung-Gyu Kim
[1]Department of Computer Software and Engineering, University of Science and Technology, Korea
[2]Embedded SW Research Division, Electronics and Telecommunications Research Institute, Korea

Abstract: Block motion estimation with full search is computationally complex. To reduce this complexity, different methods have been proposed, including partial distortion, which can reduce the computational complexity with no loss of image quality. We propose a Distortion-based Partial Distortion Search (DPDS) based on the magnitude of distortion and adaptive update of the matching order. We calculate absolute differences for all pixels in the predicted block point. Pixels are then sorted by the amount of distortion in a descending order for the matching process, which produces a scanning map. The Sum of the Absolute Differences (SAD) of other candidate positions is then computed from this matching order. We also use an update of the scanning map by checking the increase in the number of absolute differences for the SAD value. The proposed DPDS algorithm improves the computational efficiency, compared with the original PDS scheme, because the accumulated value of the absolute pixel differences can rapidly reach the current minimum SAD value. The proposed algorithm is 4-13 times faster than the full search method with the same visual quality.

Key words: Video coding, image compression, adaptiv distribution, partial distortion, motion estimation

INTRODUCTION

Motion estimation is one of the most time-consuming parts in the block-based video codec, requiring 50-60% of the total encoding time. Various motion estimation techniques have been proposed to reduce the computing time for current video coding. The Block Matching Algorithm (BMA) is the most popular method and there are several types of block matching algorithms. One BMA method used fixed matching patterns (Zhu and Ma, 1997). A second method uses a flexible pattern and initial search position determination techniques for motion estimation using a prediction method (Nie and Ma, 2002; Byung *et al*., 2006). A third method uses successive elimination techniques with multiple resolutions (Gao *et al*., 2000). These method algorithms are fast but suffer video quality degradation caused by some limitation.

A Partial Distortion Search (PDS) (Eckart and Fogg, 1995) has also been proposed that has the same number of candidate pixel comparisons as the full search, but is faster with no video quality degradation. PDS reduces the computational complexity by early termination of SAD calculations when a partial SAD exceeds the minimum SAD during the matching process. The partial SAD of an MB is defined as:

$$SAD_p(x,y;u,v) = \sum_{j=0}^{p}\sum_{i=0}^{15} |F_t(x+l_n, y+m_n) - F_{t-1}(x+l_n+u, y+m_n+v)| \quad (1)$$

Where, $\{(l_n, m_n) \mid n = 0,...., j\times16+I,..., 256\}$ is all pixels of one Macroblock (MB) and p indicates the pixel position of the early stop.

Hui *et al*. (2005) proposed an algorithm based on the clustering characteristics of the pixel matching error with similar magnitudes. Chan *et al*. (2004) suggested representation of pixel activities using a Hilbert scan. Hong and Oh (2004) used a specified matching order for SAD calculation in which the matching order is determined by absolute pixel difference values sorted in a descending order at the center of the block.

Herein, we introduce a fast and efficient PDS algorithm to reduce the computational complexity. The proposed algorithm is called Distortion Based Partial Distortion Search (DPDS). The distortion based matching order, originally used by Hong, is also used here. The ordering method is the same but we have also used prediction. Using a motion vector prediction method, we can determine a more suitable position for a minimal SAD value and a more effective matching order than just using the center of the block. Using our method, the parameter, p value in formula (1) can be decreased. Therefore, the

Corresponding Author: Byung-Gyu Kim, Embedded SW Research Division, Electronics and Telecommunications Research Institute, Korea

proposed DPDS reduces the computational load with faster early termination of the SAD calculation.

Another advantage of the proposed DPDS is an update process for the matching order. In Hong's existing scheme, the matching order is determined at the center of the block, based on the distortion of texture. However, the comparison texture can be changed by a matching region change, creating a variation of the distortion. Thus, the accuracy of the matching order is degraded. To avoid this problem, an updating method is proposed for DPDS.

DISTORTION-BASED PARTIAL DISTORTION SEARCH ALGORITHM

Overview: The proposed DPDS consists of three steps. First, we define an initial search point to find an optimal point that has a minimal SAD value. Second, we determine a matching order and perform a SAD calculation with the given matching order. Last, a matching order update is performed for comparison based on the texture change. Using these steps, we can reduce the computational complexity.

Prediction of an initial search point: Generally, the initial position for a conventional partial distortion search is the center of the macroblock. After an initial search, the SAD value is obtained using a threshold. We can also obtain the matching order for the SAD calculation of other candidate blocks from the initial search. If we can find a more suitable initial search point with a lower SAD value, we can terminate the SAD calculation for the candidate block earlier. Therefore, the number of the absolute difference operations p of the partial SAD value can be decreased.

The accuracy of the prediction method is well known. The probability that the predicted position will be less than the SAD value is high (Yang *et al*., 2005). Some sequences have approximately 90% consentaneity. We also can obtain a faster matching following the distortion amplitude. Using the prediction of the initial search point, the proposed DPDS has less computational complexity than the method using the center of the search range as an initial search position. The median prediction method was used for the proposed algorithm. The left, left-upper, upper and right-upper blocks are used for the predictive position.

Distortion based matching order: The progressive matching order was used for the SAD calculation of a conventional partial distortion search. The proposed algorithm uses a matching order based on the magnitude of the distortions (Fig. 1). Generally, there are 256 pixel operations to determine the SAD value of one macroblock. Thus, there are 256 positions that have absolute difference values (distortions). The DPDS uses this distortion information. If a pixel that has a large distortion is considered first during the SAD calculation of the candidate blocks, the partial SAD value can be determined early to be less than a threshold SAD value as the current minimum SAD value. The positions are sorted in a decreasing order of the distortions and then are used as matching order for other candidate blocks.

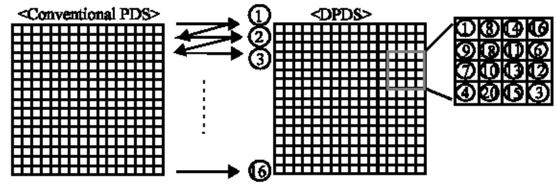

Fig. 1: The computational order of the partial distortion of one macro-block

The matching order and the threshold SAD value are first calculated at the predicted point and then the partial distortion of candidate block is compared with the current minimum SAD value. If the partial distortion of the candidate block is greater than the current minimum SAD value, the candidate block is discarded. If the partial distortion of candidate block is smaller than the current minimum SAD value, the minimum SAD value is replaced with the value of the candidate block. This comparison is repeated for all search positions according to the sort order.

Update of the matching order: The spiral scanning method was used for the candidate position transfer during the motion estimation process. The general method for determining a matching order is executed once at the center of the spiral search. This matching order is efficient for center biased pixels because the matching space has a similar texture. Therefore, the distribution characteristics of distortions are similar. However, if the candidate block position is far from this center, the change of texture for the region comparison can be enlarged by motion translation. A new texture region can be generated or an existing texture can disappear (Fig. 2).

Due to this concern, the distribution characteristics of distortions are also changed. Therefore, the matching order based on the distribution characteristics of distortions is not efficient far from the predictive center. To prevent this problem, a matching order update technique is proposed.

When the minimal SAD value is updated at the candidate pixel position, the matching order is updated

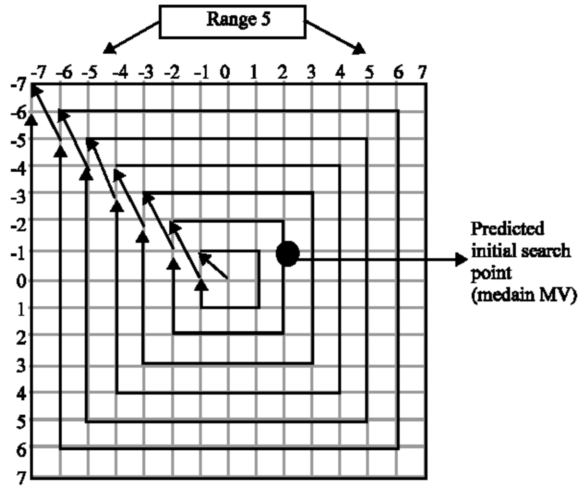

Fig. 2: Spiral search strategy to determine the position of candidate pixels

based on the new order of the distortions. However, a trade off exists between the sorting time and the reduced time caused by a new matching order. Thus, a too frequent matching order update can cause an increase in the time consumed because of the sorting time.

For many cases, most of the update occurs at positions near the center because many test sequences have center-biased motion vector characteristics. Also, many pixels near the center have similar SAD values. Therefore, unnecessary updates can occur frequently near the center.

To solve this problem, the proposed update scheme uses a range constraint. Candidate pixels near the center within the range of 5 have little texture change. Thus, we don't use an update scheme within this range. Over the range of 5, there is considerable texture change. Using this update scheme, the proposed algorithm can reduce the time required for large motion sequences.

RESULTS AND DISCUSSION

To evaluate the performance of the proposed DPDS, conventional PDS, APDS and SPDS schemes were also evaluated for comparison. We tested 100 frames for each sequence. Our search range was ± 8. The results are shown in Table 1. The proposed algorithm greatly reduced the computational complexity compared with the full search method with no loss of image quality with a speed increase of 4-13 times. Our scheme shows a small speed gain with complex motion sequences for all algorithms.

There is a large increase in the speed-up ratio for the Stephan, Bus and Coast guard sequences with fast motion. These sequences all show a moving object with relatively fast motion and a moving background image as the camera moves. Since, the proposed DPDS uses the predicted starting point and update scheme that considers the motion of sequences, our scheme is effective for sequences with large motion, compared with other algorithms.

Table 1: Speed-up ratio of each sequence

CIF	FS	PDS	APDS	SPDS	DPDS
Foreman	1.00	3.85	6.01	6.45	6.57
Bus	1.00	3.13	3.79	4.01	7.37
Stephan	1.00	3.41	4.11	4.49	5.18
City	1.00	3.17	3.77	4.19	4.3
Coast guard	1.00	4.85	6.43	6.49	7
Paris	1.00	8.91	13.66	13.1	13.85

Table 2: The average number of absolute difference calculations in one macroblock

CIF	FS	PDS	APDS	SPDS	DPDS
Foreman	256	65.33	36.78	35.29	33.18
Bus	256	80.26	59.76	57.64	28.82
Stephan	256	73.4	54.73	51.26	42.89
City	256	79.34	60.4	55.43	52.69
Coast guard	256	51.73	33.68	33.56	29.21
Paris	256	28.08	14.96	15.47	14.81

The average number of the absolute difference calculations for one Macroblock (MB) is shown in Table 2. At the candidate pixel, there are 256 Absolute Differences (AD) during a SAD calculation in a full search. One MB has 289 candidate pixels within the search range of 8. Therefore, the average number of absolute difference operations for one MB is:

$$\mathrm{AVG(AD)} = \frac{\sum_{0}^{\mathrm{CP}} \mathrm{AD}_{\mathrm{OnePixel}}}{\mathrm{CP} \times \mathrm{Total\ MB}} \tag{2}$$

CP represents the number of Candidate Pixels (CP) in one MB. The proposed algorithm has 289 candidate pixels. The proposed algorithm yields good performance for the Bus and Stephan sequences. As the motion complexity of a sequence increases, the average number of absolute difference operations also increases.

The Average number of Absolute Differences (AVG (AD)) per frame is represented as the frame moves in Fig. 3. In the Bus sequence, the proposed algorithm exhibits good performance compared with the other algorithms. The DPDS is faster than SPDS by factor of approximately 2 times. The performance improvement is especially apparent in frame numbers 1~30 and 80~90. These sections have relatively large motion comparing with the other time section. The average absolute difference values for other algorithms are increased in theses sections. But the increased rates are low for the proposed DPDS.

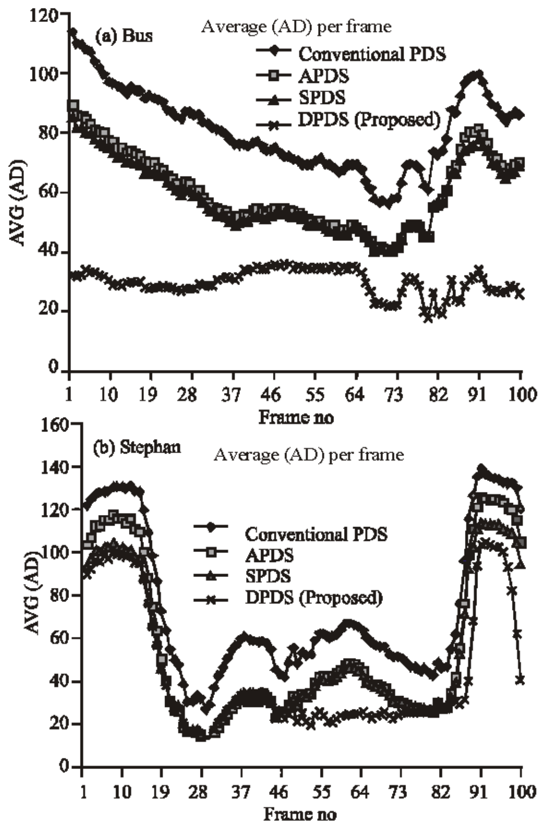

Fig. 3: The average number of the absolute difference calculations per frame

For the Stephan sequence, a similar characteristic is observed in frames 45~70 and 90~ 100. The Bus sequence has large motions but the directions of motions are simple. Therefore, the matching probabilities between the predicted minimal and the real minimal point are high. Therefore, the proposed median prediction operates properly. Good algorithm performance is shown in the large motion regions.

Our proposed matching order update scheme for large motion can reduce the computational complexity. Using an update scheme for large motion, the reordered matching positions can deal with changing textures resulting in reduced computational loads.

CONCLUSION

A distortion-based matching order was used for the proposed algorithm. Using this matching order, we can terminate the SAD calculation process early. A median prediction scheme was proposed for detection of the initial search point. With these techniques, we can determine the position that has a SAD value smaller than the center position. Early termination results in a fast algorithm using these methods. We also proposed a matching order update scheme for coping with texture changes associated with large motion. The reordered matching order according to texture changes provides a more accurate distortion order for early termination. Through comparative analysis, we have verified that the proposed algorithm reduces the computational load without loss of quality.

REFERENCES

Byung-Gyu Kim, Suk-Kyu Song and Pyoung-Soo Mah, 2006. Enhanced block motion estimation based on distortion directional search patterns. Pattern Recognition Lett., 27: 1325-1335.

Eckart, S. and C. Fogg, 1995. ISO/IEC MPEG-2 software video codec. Proc. SPIE., 2419: 100-118.

Gao, X.Q., C.J. Duanmu and C.R. Zou, 2000. A Multilevel successive elimination algorithm for block matching motion estimation. IEEE . Trans. Image Processing, 9: 501-504.

Hong, W.G. and T.M. Oh, 2004. Sorting-based partial distortion search algorithm for motion estimation. Elec. Lett., 40: 113-115.

Ko-Cheung Hui, Wan-Chi Siu and Yui-Lam Chan, New adaptive partial distortion search using clustered pixel matching error Characteristic. IEEE. Trans. Image Processing, 14: 597-607.

Libo Yang, Yu K. Jiang Li and Shipeng Li, 2005. An effective variable block-size early termination algorithm for H.264 video coding. IEEE. Trans. Circuits and Sys. Video Tech., 15: 784-788.

Yao Nie and Kai-Kuang Ma, 2002. Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation. IEEE. Trans. Image Pcessing, 11: 12.

Yui-Lam Chan, Ko-Cheung Hui and Wan-Chi Siu, 2004. Adaptive partial distortion search for block motion estimation. J. Visual Commun. Image Representation, 15: 489-506.

Zhu, S. and K.K. Ma, 1997. A new diamond search algorithm for fast block-matching motion estimation. In: Proc. 1997 Int. Conf. Inform. Commun. Signal Processing (ICICS), 1: 292-296.