PRIORITY SEARCH TECHNIQUE FOR MPEG-4 MOTION ESTIMATION OF ARBITRARILY SHAPED VIDEO OBJECT

Ko-Cheung Hui, Yui-Lam Chan and Wan-Chi Siu
Centre for Multimedia Signal Processing
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

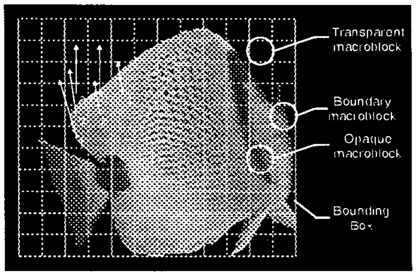
ABSTRACT

One of the main differences between the MPEG-4 video and the previously standardized video coding schemes is the support of arbitrarily shaped video objects, for which most of the existing fast motion estimation algorithms are not suitable. The conventional fast motion estimation algorithm works well for opaque macroblocks, but not the case for a boundary macroblock which contains a large number of local minima on its error surface. In this paper, we propose a fast search algorithm which incorporates the binary alpha-plane to predict accurately the motion vectors of boundary macroblocks. Besides, these accurate motion vectors can be used to develop a novel priority search algorithm which is an efficient search strategy for the remaining opaque macroblocks. Experimental results show that, when compared to the conventional methods, our approach requires a low computational complexity and provides a significant improvement in terms of accuracy in motion-compensated video object planes.

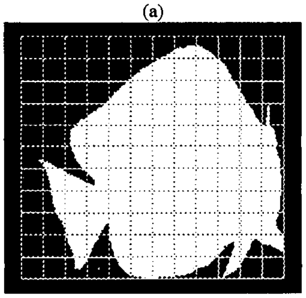
1. INTRODUCTION

MPEG-4 is the international standard which provides a coding scheme for arbitrarily shaped video objects (VOs) [1-2]. Each VO is composed of its temporal instances, video object planes (VOPs). A VOP can be fully described by texture variations and shape representation, as shown in Figure 1. The shape information can be represented as a binary alpha-plane. The alpha-plane contains information to identify the pixels which are inside an object (value of alpha-plane = 1), and the pixels which are outside the object (value of alpha-plane = 0), as depicted in Figure 1(b). MPEG-4 defines an arbitrarily shaped VOP by means of a boundary rectangle called a "bounding box". The box surrounds the VOP with the minimum number of macroblocks, as depicted in Figure 1(a). There are three kinds of macroblocks (MBs) within a bounding box: the transparent MB, the boundary MB and the opaque MB. The boundary and opaque MBs include the pixels belonging to the object, and the transparent MB lies completely outside the object area.

The basic MPEG-4 coding employs block-based motion estimation and compensation techniques to efficiently explore temporal redundancies of the video content in the separate VOPs. In general, the motion estimation and compensation techniques used can be seen as an extension of the standard MPEG block matching techniques which can process image sequences of arbitrary shapes. Most probably, the error measure criterion such as the sum of absolute differences (SAD) for block matching motion estimation is calculated on all pixels. This conventional block matching approach is applied to the opaque macroblock. But, on the boundary macroblock, the binary alpha-plane for the VOP is used to exclude the pixels of the macroblock that are outside the VOP. This forms a polygon matching instead of block matching for the motion estimation of the boundary macroblock. The SAD polygon matching criterion is calculated only for pixels associated with an alpha-plane which contains non-zero values[1-2]. However, pixels of the previous frame, which are located outside of the VOP but are still inside the bounding box of the VOP, have to be replicated by a repetitive method [1-2] for computing the SAD.



(a)



(b)

Figure 1. Representation of the VOP. (a) Image of original "Bream" VOP. (b) Binary alpha-plane of "Bream" VOP.

Motion estimation is the most computationally expensive step in MPEG1 and MPEG2 encoding. In the past, much work was reported on reducing the complexity of motion estimation [3-5]. It is likely that motion estimation will remain to be a very computationally intensive step in MPEG-4 encoding. As a result, it is highly desirable to reduce the computational requirement of motion estimation. In this paper, we propose a novel algorithm which incorporates the binary alpha-plane to accurately predict the motion vectors of boundary macroblocks such that the motion-compensated VOPs are tied more closely to the video object. Besides, these accurate motion vectors can be used to develop an efficient motion estimation algorithm for the remaining opaque macroblocks.

2. PRIORITY SEARCH ALGORITHM (PSA)

For an efficient block motion estimation, it is important to know the characteristics of different types of macroblocks. In Figure 1,

0-7803-6725-1/01/$10.00 ©2001 IEEE

three types of macroblocks are available. Their corresponding motion search strategies are summarized as follows.

- Transparent macroblocks: They are not coded and recovered at the decoder side from the shape information. Thus no motion search is required.
- Boundary macroblocks: This type of macroblock partially includes object pixels, and the polygon matching is employed to adopt arbitrarily shaped moving video objects. The human visual system is very sensitive to the poor motion-compensated prediction along the moving contours of video objects which are located on boundary macroblocks. The correct motion estimation of boundary macroblocks is critical to the development of an efficient motion estimation algorithm for arbitrarily shaped moving video objects.
- Opaque macroblocks: The opaque macroblock is coded using the conventional block matching motion estimation algorithm. Since the motion within the video object is consistent, it can be seen easily that the motion in these macroblocks correlates highly with the surrounding boundary macroblocks provided that the motion vectors in the boundary macroblocks truly represent the moving video object. For example, the opaque macroblock, MB_e, in Figure 1 highly correlates with the boundary macroblocks, MB_a, MB_b, MB_d and MB_f. Thus, the motion vectors of the boundary macroblocks play a significant role in the development of an efficient motion estimation for the opaque macroblock.

The characteristics of the different types of macroblocks described above gave us the inspiration to develop a new priority search algorithm (PSA) which performs motion estimation on all boundary macroblocks first within the bounding box of the VOP in contrast to the conventional raster-scanning approach (scanning macroblocks in the order of top-to-bottom and left-to-right). The idea behind the new search strategy is that the opaque macroblocks which are inside moving video objects highly correlate with the moving boundary macroblocks. For each opaque macroblock, if all motion vectors of its adjacent boundary macroblocks have already been computed, the current opaque macroblock can take the mean of these motion vectors as the initial center and employ a conventional fast block matching algorithm such as the block-based gradient descent search algorithm (BBGDS) to compute its motion vector for a reduction of the computational complexity. It is likely that the global minimum can be found by a local search such as using the BBGDS if the initial center is close enough to the global minimum. But if not all adjacent motion vectors of the boundary macroblocks have been computed, the computation of the motion vector of this opaque macroblock will be postponed until all motion vectors of the required adjacent boundary macroblocks are available. The advantage of this new search strategy is that it avoids unnecessary computations of the opaque macroblock so that the motion search can be conducted more efficiently.

In order to ensure the accuracy of the motion vectors of the boundary macroblocks, a full search algorithm (FSA) which evaluates the SAD at all possible locations of the search window is employed. By using the accurate motion vectors of the boundary macroblocks, the best-matched motion vectors of the opaque macroblocks can be found by the BBGDS. This PSA produces smaller motion compensation errors, and has a low computational complexity as compared with the traditional raster-scanning motion estimation. Figure 2 depicts the performance of the PSA in encoding the "Bream" video object. The figure plots the mean square errors (MSE) between the original VOP and the motion-compensated VOP of the PSA with the FSA performing on boundary macroblocks and the BBGDS on opaque macroblocks using the mean of the motion vectors of its surrounding boundary macroblocks as the initial center (PSA(FSA+BBGDS)) and compares the results with those of the full search algorithm (FSA). The results show that the MSE performance of the PSA(FSA+BBGDS) is very close to the FSA. Details on the simulation environment are described in Section 3.

As mentioned above, the accuracy of the motion information of boundary macroblocks is critical in PSA(FSA+BBGDS). Consequently, the FSA is used to ensure its accuracy. Overall, about 99% of the total search points required of the whole motion estimation process are performed for the boundary macroblocks. In order to increase the flexibility and practicability of the PSA(FSA+BBGDS), the computational burden of the motion estimation of the boundary macroblocks must be reduced. In Figure 2, we also analyse the MSE performance of the PSA by using the BBGDS for both the boundary macroblocks and the opaque macroblocks (PSA(BBGDS+BBGDS)), in which the mean of the resulting motion vectors of boundary macroblocks obtained by the BBGDS act as an initial center for performing the BBGDS on opaque macroblocks. Figure 2 shows that there is a big prediction error in the PSA(BBGDS+BBGDS) as compared with that of the FSA. This is because the probability of occurrence of the local minimum problem is more often in the boundary macroblock. This phenomenon could achieve our desire to develop a fast and efficient search algorithm for the boundary macroblock, which is described in the following section.

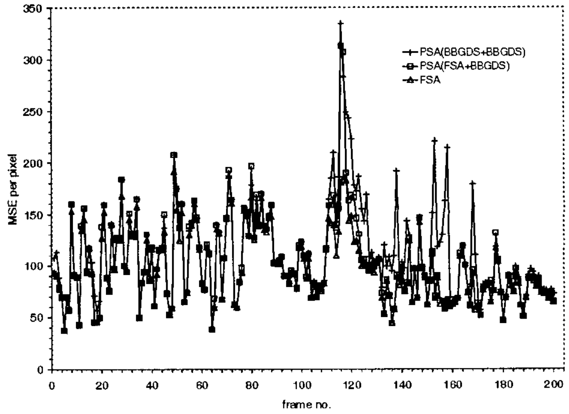


Figure 2. MSE performance of PSA for "Bream" video object.

3. RELIABLE SEARCH THROUGH BINARY ALPHA-PLANE OF THE BOUNDARY MACROBLOCK

The statistical behaviour of an error surface has a significant impact on the performance of a fast search algorithm. For the surface of the boundary macroblock as shown in Figure 3(a), it contains a large number of local minima. Almost all conventional fast algorithms have assumed explicitly or implicitly [3] that the error surface is unimodal over the search window. As a consequence, it is unlikely that conventional fast search algorithms would converge to the global minimum when performing motion estimations on the boundary macroblock. In other words, the search would easily be trapped into a local minimum. Despite the error surface exhibiting uncertainties in a large spatial scale, we can reasonably assume that it is monotonic in a small neighbourhood around the global minimum. One possible solution to prevent the problem of trapping to a local minimum is to test more of the starting points which spread across the search window. Figure 4 shows one of the starting point pattern (SPP) in which the starting points (SP) are distributed evenly over the search window. It is obvious that if the number of starting points is reduced as much as possible and the starting point is as close as possible to the true motion vector, the search algorithm becomes efficient. Hence, we have to adjust the regular SPP among the blocks so that the limited SPs have a higher chance of catching the global minimum. In this paper, we propose a binary alpha-plane assisted search algorithm (BAAS) in which the adjustment of the regular SPP generally includes a matching process for tracking a polygon shape in a video object. The proposed algorithm first estimates an initial probability of being the global minimum for each possible matching pair between the current boundary macroblock and the macroblock at the regular SPP which is updated based on the shape information. In the following, we highlight the main steps of our BAAS.

• Step 1: Adjustment of the regular SPP

In order to evaluate the similarity of arbitrary shapes between two macroblocks, we define a cost function which should have a small value or be zero only if the two macroblocks have identical shapes. A binary alpha-plane matching score (BAMS) is introduced to measure the shape similarity between a boundary macroblock of the present VOP and a macroblock with displacement *(u,v)* of the previous VOP with a block size of *NxN* pixels. Note that the BAMS can be easily implemented by a simple circuitry containing an 'XOR' logic gate and a counter.

$$BAMS(u,v)=\sum_{i}^{N-1}\sum_{j}^{N-1}BA_t(i,j)\ XOR\ BA_{t-1}(i+u,j+v) \qquad (1)$$

where $BA_t(\cdot,\cdot)$ and $BA_{t-1}(\cdot,\cdot)$ are the values of the binary alpha-plane of the current boundary macroblock at the t^{th} VOP and the reference VOP at the $(t-1)^{\text{th}}$ VOP that are to be compared respectively. The BAMS is used to determine if the polygon shapes in the two boundary macroblocks are similar. In this case, the macroblock in the regular SPP has a large probability of being closest to the global minimum. Figure 3 plots the error surfaces and the BAMS surfaces of a boundary macroblock. We have found that the correlation between these two surfaces is very high and it further ensures that the motion search algorithm can be guided by the BAMS. Thus a macroblock in the regular SPP whose BAMS is less than a pre-defined threshold, T_{BAMS}, is good enough to be an interesting SP. In other words, this SP is reserved in the updated SPP. In order to normalize thresholding, the T_{BAMS} must be proportional to the number of opaque pixels of the current macroblock. That is,

$$T_{EMS}=\alpha\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}BA_t(i,j) \qquad (2)$$

where α is a proportional constant.

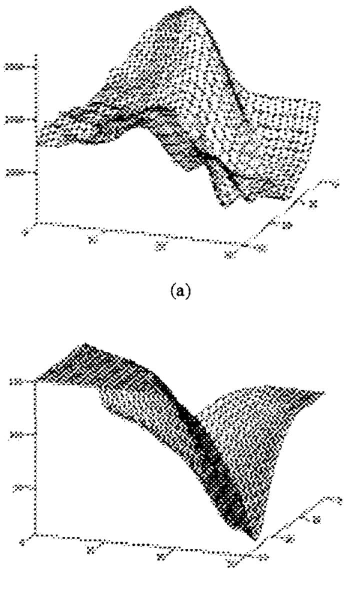


Figure 3. The relationship between error surface and the proposed BAMS surface of the boundary macroblock.

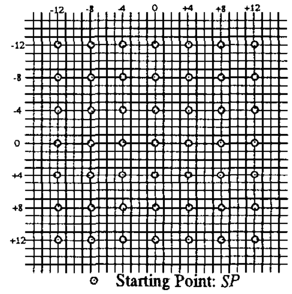


Figure 4. Regular SPP

Step 2: The formation of the final SPP

In order to reduce further the computational complexity, the updated SPP can be refined by using the image intensity. A simple way is to employ the SAD polygon matching criterion. The approach uses the SAD values to select the best matched SP as compared to other SPs in the updated SPP, and it is defined as

$$G_k=SAD_k-SAD_{\text{min\_in\_updated\_SPP}} \qquad (3)$$

where k means to cover all selected SPs of the updated SPP, except for the SP which has the smallest value of SAD in the updated SPP, and where $SAD_{min\_in\_updated\_SPP}$ and SAD_k are the smallest value in the updated SPP and the value of the SAD from the SP in the updated SPP, respectively. First, the SP with the smallest value must be reserved as the final SPP. Second, the value of G_k is used to establish the final SPP. If the value of G_k is small enough (smaller than $\beta \times SAD_{min\_in\_updated\_SPP}$, where β is also a proportional constant), it implies that the probability of this SP being the global minimum is high. In other words, this SP must be included in the final SPP; otherwise, this SP is eliminated from the updated SPP. After examining all the SPs in the updated SPP, the final SPP is formed.

Step 3: Motion vector estimation using the BBGDS

After the establishment of the final SPP, all SPs in the final SPP serves as the starting point of the BBGDS [3]. Finally, searches are conducted to find the minimum value of the SAD.

4. SIMULATION RESULTS

A series of computer simulations have been conducted to evaluate the performance of the proposed algorithm. These include the "Bream" and the "Children" video objects. The maximum allowable displacement in both the horizontal and vertical directions is *15* with a block size of 16×16. The mean square error (MSE) is used to compare the performance of the proposed algorithm with the related techniques reported in the literature. For our proposed PSA(BAAS+BBGDS) algorithm, the parameters in the formation of the final SPP α and β are set to *0.3* and *0.5*, respectively. Figure 5 compares the results of the MSE of the motion-compensated VO of the proposed algorithm together with the results of other approaches, including the FSA, BBGDS[3], PSA(FSA+BBGDS), and PSA(BBGDS+BBGDS). There is a great increase in the prediction error of the PSA(BBGDS+BBGDS) and the conventional BBGDS when compared with that of the FSA. This is because the probability of occurrence of the situation in Figure 3(a) is higher in boundary macroblocks with fast moving objects. This situation causes an unreliable stop in the search for the conventional BBGDS, and it implies that these kinds of algorithms are more easily trapped in a local minimum. However, our BAAS can resolve the problem of the local minimum by placing the checking block closest to the global minimum which is guided by the binary alpha-plane. As shown in Figure 5, the new BAAS with the PSA is significantly better than those of the PSA(BBGDS+BBGDS) and the conventional BBGDS. Also, we can see that the MSE performance of our approach is very close to that of the FSA. Furthermore, after combining all the operations of our PSA(BAAS+BBGDS), it is over *30* times as fast as the FSA. The detailed simulation results of other test sequences are summarized in Table 1.

5. CONCLUSIONS

By considering the correlation between the boundary macroblock and the opaque macroblock, a priority search algorithm (PSA) for arbitrarily shaped video objects has been proposed in this paper. We have also demonstrated that obtaining accurate motion information of boundary macroblocks is important for improvement of the performance of block motion estimation. A fast and efficient algorithm for estimating the motion vectors of boundary macroblocks is suggested, which is referred to as the binary alpha-plane assisted search (BAAS) in this paper. The binary alpha-plane is used for the adjustment of the start point patterns of the search windows such that a limited number of starting points can still provide a high chance of catching the global minimum in the boundary macroblocks. Experimental results show that our PSA coupled with the BAAS can reduce the heavy computational burden of the FSA without significantly increasing the prediction error of the motion-compensated frame. The proposed algorithm is significantly better than that of the famous BBGDS and substantially improves the accuracy of the block motion estimation for MPEG-4 video objects.

6. ACKNOWLEDGMENTS

This work is supported by the Centre for Multimedia Signal Processing, Hong Kong Polytechnic University. K.C. Hui acknowledges the research studentships provided by the University and Dr.Y.L.Chan is grateful for the support he receives from the University under its research fellowship scheme.

7. REFERENCES

[1] ISO/IEC, "MPEG-4 video verification model version 8.0," ISO/IEC JTC1/SC29/WG11 N1796, July 1997.

[2] T. Sikora, "The MPEG-4 video standard verification model," IEEE Trans. Circuits Syst. Video Technol., vol. 7, pp.19-31, Feb. 1997.

[3] L. K. Liu and E. Feig, "A block-based gradient descent search algorithm for block motion estimation in video coding," IEEE Trans. on Circuits Syst. Video Technol., vol.6, pp. 419-421, August 1996.

[4] Yui-Lam Chan and Wan-Chi Siu, "A Feature-assisted Search Strategy for Block Motion Estimation," Proceedings of the IEEE International Conference on Image Processing(ICIP'99), pp. 620-624, vol. II, 24-28 October 1999, Kobe, Japan.

[5] Yui-Lam Chan and Wan-Chi Siu, "Edge oriented Block Motion Estimation for Video Coding," IEE Proceedings: Vision, Image and Signal Processing, vol. 144, no. 3, pp. 136-144, June 1997, U.K.

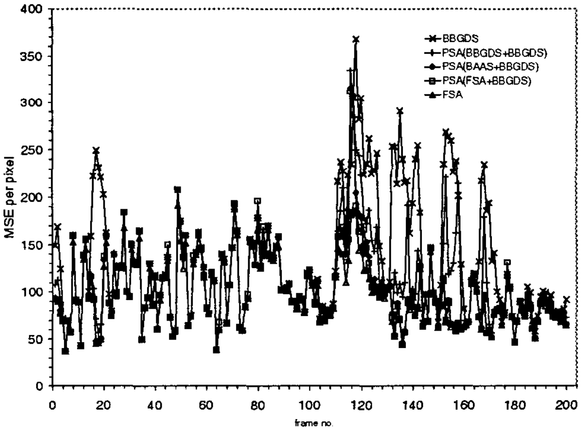


Figure 5. The MSE produce by different algorithms for "Bream" video object.

Table 1. Performance comparison of the algorithms.

Algorithms	Bream		Children	
	MSE	Complexity	MSE	Complexity
FSA	*98.5*	*100%*	*173.6*	*100%*
BBGDS	*134.9*	*1.62%*	*197.0*	*1.41%*
PSA (FSA+BBGDS)	*103.2*	*40.74%*	*175.2*	*79.52%*
PSA (BBGDS+BBGDS)	*112.5*	*1.36%*	*194.1*	*1.38%*
PSA (BAAS+BBGDS)	*103.4*	*3.24%*	*177.8*	*3.53%*