An Efficient Search Strategy for Block Motion Estimation Using Image Features

Yui-Lam Chan and Wan-Chi Siu, *Senior Member, IEEE*

***Abstract*—Block motion estimation using the exhaustive full search is computationally intensive. Fast search algorithms offered in the past tend to reduce the amount of computation by limiting the number of locations to be searched. Nearly all of these algorithms rely on this assumption: the MAD distortion function increases monotonically as the search location moves away from the global minimum. Essentially, this assumption requires that the MAD error surface be unimodal over the search window. Unfortunately, this is usually not true in real-world video signals. However, we can reasonably assume that it is monotonic in a small neighborhood around the global minimum. Consequently, one simple strategy, but perhaps the most efficient and reliable, is to place the checking point as close as possible to the global minimum. In this paper, some image features are suggested to locate the initial search points. Such a guided scheme is based on the location of certain feature points. After applying a feature detecting process to each frame to extract a set of feature points as matching primitives, we have extensively studied the statistical behavior of these matching primitives, and found that they are highly correlated with the MAD error surface of real-world motion vectors. These correlation characteristics are extremely useful for fast search algorithms. The results are robust and the implementation could be very efficient.**

A beautiful point of our approach is that the proposed search algorithm can work together with other block motion estimation algorithms. Results of our experiment on applying the present approach to the block-based gradient descent search algorithm (BBGDS), the diamond search algorithm (DS) and our previously proposed edge-oriented block motion estimation show that the proposed search strategy is able to strengthen these searching algorithms. As compared to the conventional approach, the new algorithm, through the extraction of image features, is more robust, produces smaller motion compensation errors, and has simple computational complexity.

***Index Terms*—Block matching algorithm, image features extraction, motion estimation, motion vector.**

I. Introduction

MOTION estimation plays an important role in today's video coding and processing systems, because motion vectors are critical information for temporal redundancy reduction [1]–[4]. Due to their simplicity and the coding efficiency of the motion vectors, the most successful technique for motion estimation is perhaps the block matching algorithm (BMA) [5]–[23]. To do this, a current frame is first partitioned into two-dimensional (2-D) small blocks and a search window in the reference frame is defined. Then, each block of the current frame is compared against all blocks of a previous frame within the search window. The motion vector is finally obtained by searching for the minimum point on an error surface that is composed of the block matching distortion over all candidate motion vectors within the search window.

Nevertheless, motion estimation, even using the block matching technique in which the mean absolute difference (MAD) is employed to measure the block matching distortion, still requires enormous computations. In particular, it has often become a bottleneck problem in real-time applications if the conventional full search algorithm (FSA) is used. To illustrate this situation, we consider a typical MPEG-1 application [24]: Suppose that a 15×15 search window is used for a video sequence with a source intermediate format (SIF) at a rate of 30 frame per second. It is found that the full search motion estimation using MAD block matching distortion requires about 1.2 billion (integer) additions and 0.57 billion comparisons per second! In fact, this amount of computations can take up to 75% of the computing power of the whole encoding system. For future high-definition television (HDTV) applications, much larger search windows will have to be employed. This gives rise to a computational demand that is several orders of magnitudes higher than that for the MPEG-1 applications [25]. Clearly, the full search motion estimation is unlikely to be implemented in real-time video coding applications, and determination of a way to speed up block motion estimation with negligible performance degradation has become an important research topic for quite some time.

During the past two decades, many fast search algorithms, which reduce the computation time by searching only a subset of the eligible candidate blocks, have been developed. These fast block motion estimation algorithms include the n-step hierarchical search algorithm (n-SHS) [16], the 2-D logarithmic search algorithm [17], the conjugate directional search algorithm (CDS) [18], the new three-step search algorithm [19], the block-based gradient descent search algorithm (BBGDS) [20], the diamond search algorithm (DS) [21] and many other variations. These algorithms reduce the number of computations required by calculating the MAD matching criterion at locations coarsely spread over the search window according to some pattern and then repeating the procedure with finer resolution around the location with the minimum MAD found from the preceding step. Obviously, how the initial search pattern is to be

Manuscript received April 22, 1999; revised April 6, 2001. This work was supported by the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, Hong Kong Polytechnic University. Y. L. Chan was supported by the University under its Research Fellowship Scheme. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Steven D. Blostein.

Y.-L. Chan and W.-C. Siu are with the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: enwcsiu@polyu.edu.hk).

Publisher Item Identifier S 1057-7149(01)06035-3.

selected becomes the most crucial task. Nearly all existing fast search algorithms rely on this assumption: the MAD distortion function increases monotonically as the search location moves away from the global minimum [17]. Obviously, this assumption essentially requires that the MAD error surface be unimodal over the search window, which leads to employ a uniform pattern. Unfortunately, this is usually not true in real-world video signals. As a consequence, the minimum MAD found by these methods is frequently higher than that is produced by the FSA. This is one of the reasons why the FSA is still widely used in most high-quality video codec. The adaptive multiple-candidate hierarchical search (AMCHS) [12], [15] that is used to overcome this drawback adaptively by keeping more than one winner for the next step of motion estimation. The genetic motion search algorithm (GMS) [22] attempts to overcome this problem by first choosing a random search point and then using an algorithm similar to the genetic processes of mutation and evolution to find the global minimum of the matching distortion. However, the number of operations required by the other fast search algorithms outlined above is approximately 15% of that required by the GMS. In this paper, a reliable search strategy that employs the guide of some image features is proposed to locate one of the initial search points as close as possible to the true motion vector so that the chance of catching the true motion vector is maximized by using some image features. In addition, it is found that the proposed strategy is very suitable for combining with other block motion estimation algorithms to form an efficient algorithm.

In Section II of this paper, we present an in-depth study of the correlation between the MAD error surface and some image features, and then formulate a reliable search strategy for block motion estimation, together with an analysis of the algorithm's complexity and some of the simulation results. Based on the studies in Section II, we apply the search strategy to our feature-oriented block motion estimation algorithm to further improve its overall performance. Finally, a conclusion is drawn in Section IV.

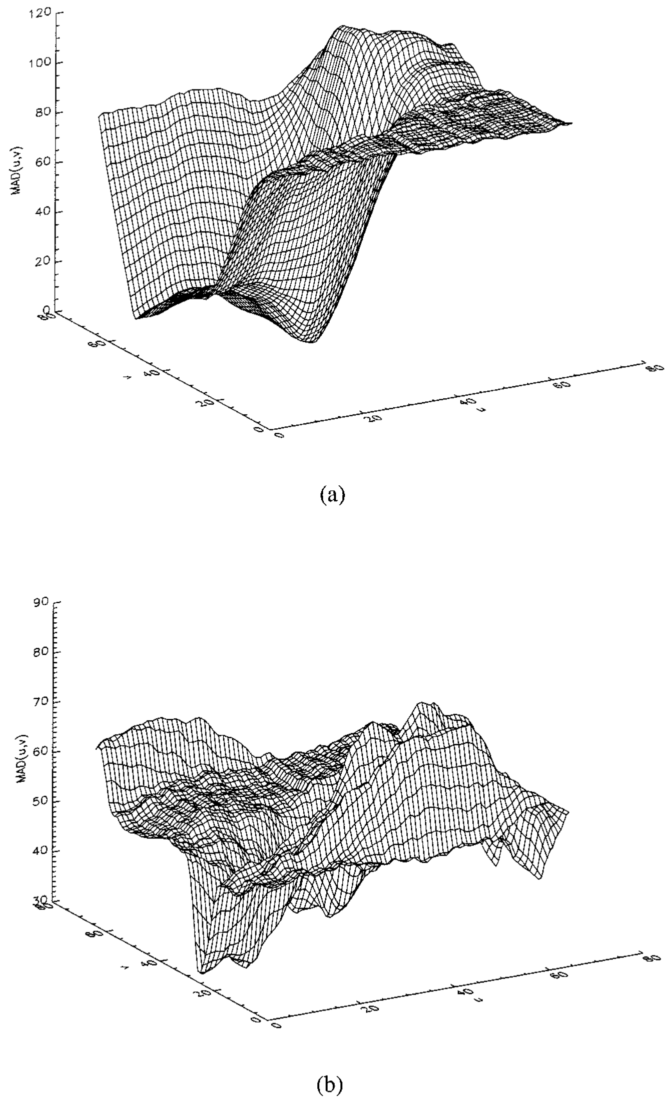


Fig. 1. MAD error surface for two different blocks.

II. Framework of an Edge-Assisted Search Strategy for Motion Estimation Algorithm

A. Statistical Behavior of the Error Surface

Suppose that the maximum motion in the vertical and horizontal directions is $\pm W$. There are thus $(2W+1)^2$ candidates in total to be checked if the full search algorithm is used, each corresponding to a checking point in the search window. The MAD values that result from these checking points form an error surface

$$\mathrm{MAD}(u,v)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|I_t(i,j)-I_{t-1}(i+u,j+v)\right| \quad (1)$$

where the block size is taken as $N \times N$, (u,v) denotes the location of the candidate motion vector, and $I_t(\cdot,\cdot)$ and $I_{t-1}(\cdot,\cdot)$ refer to the blocks in the current frame(tth frame) and in the reference frame ($(t-1)$th frame) that are to be compared.

The statistical behavior of the MAD error surface has a significant impact on the performance of a fast search algorithm for block motion estimation. For the surface as shown in Fig. 1(a), the MAD error decreases monotonically as the search location moves toward the global minimum value. This implies that simple fast search algorithms such as the n-step hierarchical search algorithm [16], the block-based gradient decent search algorithm [20] and the diamond search algorithm [21] require a small number of searches to determine the global optimum for this block. For the surface as shown in Fig. 1(b), it contains a large number of local minima. Almost all conventional fast algorithms have explicitly or implicitly assumed [17] that the error surface is unimodal over the search window. As a consequence, it is unlikely that the above-described fast search algorithms would converge to the global minimum. In other words, the search would easily be trapped into a local minimum.

B. Reliable Search Through Image Features

The guided strategy presented in this paper intends to strengthen the conventional fast search algorithms. Without loss of generality, we employ the block-based gradient descent search algorithm (BBGDS) [20] as an example to illustrate the problem of the conventional fast search algorithm. Let us recall that in the first step of the BBGDS, a search is carried out only around

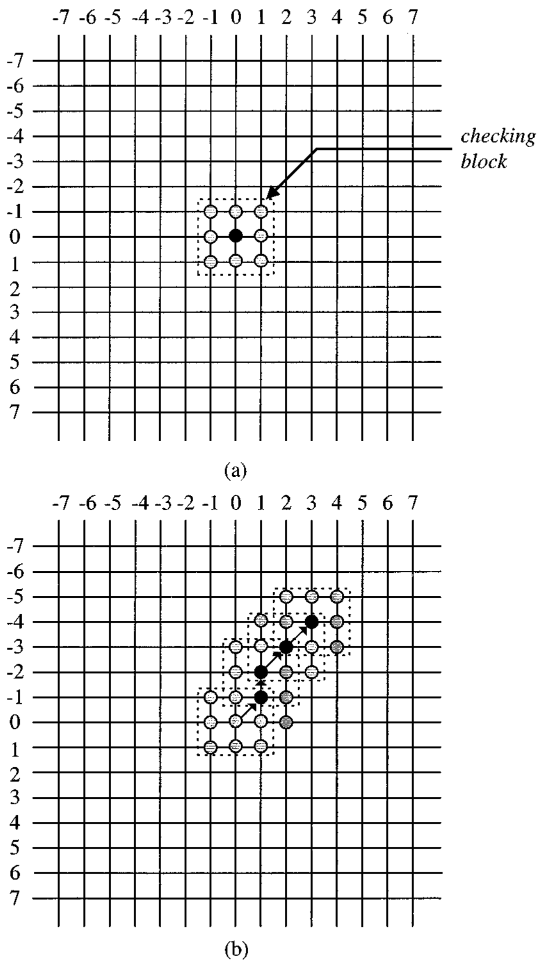


Fig. 2. (a) First step of the BBGDS and (b) example of the BBGDS search procedure, where motion vector $(3, -4)$ is found.

the center checking point as shown in Fig. 2(a). If the optimum is found at the center, the procedure stops. Otherwise, further search is conducted around the point where the minimum has just been found. The procedure continues until the winning point becomes a center point of the checking block (3×3 checking points) or when the checking block hits the boundary of the predefined search range [20]. The procedure is illustrated in Fig. 2, where the motion vector $(3, -4)$ is found. Of course, the BBGDS relies on the assumption that the MAD measure decreases monotonically as the search point moves closer to the optimal point. It can easily be trapped into the local minimum in cases where the error surface looks like the one in Fig. 1(b).

Let us use Fig. 3 to give a clearer account of this phenomenon. Fig. 3(a) shows a nonunimodal surface due to many reasons such as the aperture problem, the textured (periodical) local image content, the inconsistent block segmentation of moving object and background, the luminance change between frames, etc. In the first step of the BBGDS, the center point in the checking block wins. It stops the search process and a local minimum will be found. However, it is seen that the global minimum is located at the far side of the winning point and the MAD value of the winning point is significantly larger than that of the global minimum. This will degrade the quality of the motion-compensated frame.

Although the error surface exhibits uncertainties in large spatial scale, we can reasonably assume that it is monotonic in a small neighborhood around the global minimum [19]. In the existence of local minima, one simple strategy, but perhaps the most efficient and reliable, is to place the checking block as close as possible to the global minimum, as depicted in Fig. 3(b). If the initial checking block is close enough to the global minimum, it will be very likely able to find the global minimum through a local search. One possible solution is to test more starting points that spread across the search window. Fig. 4 shows one of the starting point pattern (SPP) in which the starting points (SPs) are distributed evenly across the search window. However, it is inefficient to use such a large amount of the starting points in this regular SPP. Consider a search window with a MAD surface as shown in Fig. 1(a), it is wasteful to use all of the starting points as shown in Fig. 4. 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 position of the true motion vector, the search algorithm becomes more efficient.

Let us suppose that the global minimum point is C_g, and a starting point close to the global minimum is C_c, as depicted in Fig. 3(b). These points play an important role in the development of a reliable search, which can be further seen from the following discussion. In a hierarchical fast search scheme, the global minimum C_g can be reached only if a local search is conducted around C_g. By reducing the number of sampling points in the regular SSP, we can save computation time as compared to the conventional methods. Thus, the number of SPs must be minimized under the condition that the remaining SPs must cover point C_g in doing the local search. This requires that at least one of the nearby starting points of C_g, such as C_c, be kept, after performing the adjustment of the regular SSP, so that the local search around the limited starting point can cover C_g.

Although the objective of the SPP design is straightforward, it becomes nontrivial when one tries to design a universal SPP for a sequence of frames such that each individual search will have the minimal amount of search points in order to obtain the global optimal solution. Usually, such a universal solution does not exist. Hence, we have to perform an adjustment of the regular SPP such that the limited SPs have a high enough chance of catching the global minimum. In this paper, we propose a feature-assisted search algorithm. The adjustment of the regular SPP is a primitive-based approach which generally includes a matching process for tracking the feature primitives from frame to frame in a sequence of images. This is the main theme of this paper. The proposed algorithm first estimates an initial probability of being the global minimum of each possible matching pair between the current block and the block at the regular SPP. Then the regular SPP is updated based on certain criteria, such as the feature similarity. Many features, such as the edge points, corner points and segmented regions, can be used as matching primitives. This approach produces good results if the extraction of primitives is nearly perfect. In the meantime, the performance of a fast algorithm is also judged by its speed-up ratio and robustness. For hardware implementation, regularity is also an important factor to be considered. Hence, tradeoffs have to be made carefully among these competing factors.

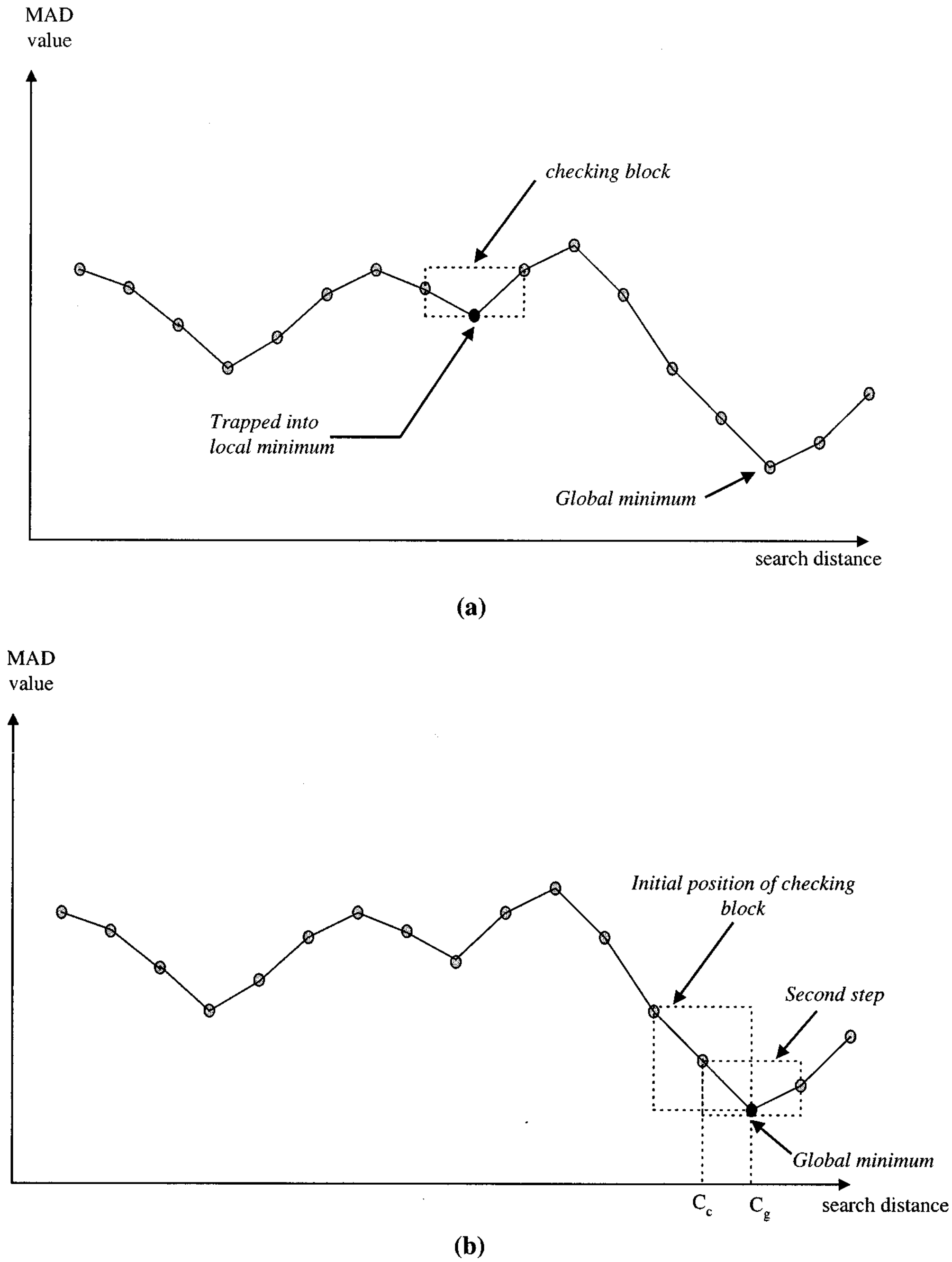


Fig. 3. Nonumimodal error surface tested by a checking block. (a) The checking block starts at the origin and a false checking results, hence a local minimum is found. (b) If the initial checking block is close enough to the global minimum, the global minimum can be successfully found.

C. Edge-Assisted Search Algorithm

In [23], we have shown that edge features can greatly improve the accuracy of motion estimation. Also, the availability of VLSI edge detection chips [26] makes the possibility of using edges in motion estimation quite realistic and potentially rewarding. In this paper, we try to employ the edge feature as an example to illustrate our proposed feature-assisted search approach and we will refer to it as the edge-assisted search algorithm (EAS). A flowchart of the proposed approach is shown in Fig. 5 and the realization procedure of the EAS is as follows.

• ***Step 1: Image Preprocessing by Edge Detection***

An efficient extraction of the edge information is an important aspect of our proposed EAS. In choosing an edge detection algorithm, one must consider its speed and precision. Prior to locating the directional derivatives, the image must be smoothed, so that ripples, spikes and high frequency noises from the image are removed. A simple smoothing process using the mean, that performs equally-weighted smoothing using a square window with the size of 5×5, has been employed here. By using the data reuse technique, some computations can be saved, as described in further details in a latter part of the paper. The edge detection algorithm is then applied to the smoothed current frame. For the sake of simplicity, the 3×3 Sobel gradient convolution masks

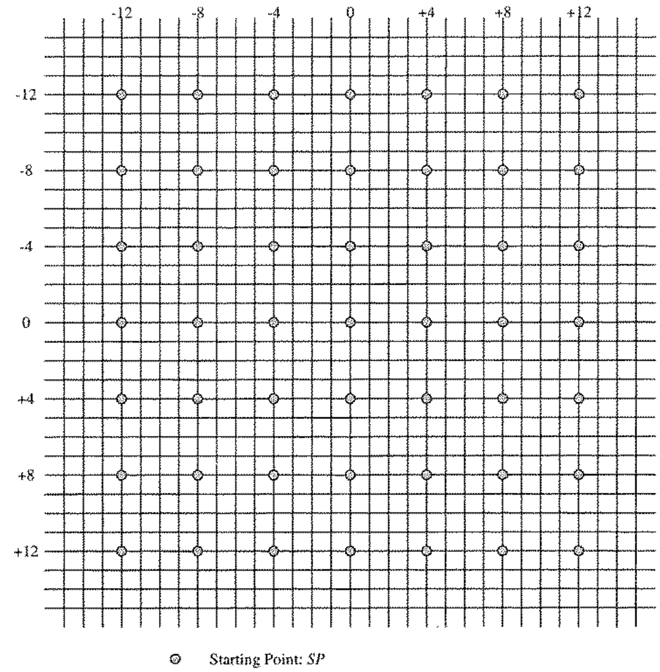


Fig. 4. Regular SPP: SPs distribute evenly across the search window.

have been used

$$g_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad g_y = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}. \tag{2}$$

The edge frame $S_t(i,j)$ is given by

$$S_t(i,j) = |I'_t(i,j) * g_x| + |I'_t(i,j) * g_y| \tag{3}$$

where $I'_t(i,j)$ is the smoothed image with the same dimension as the original image.

The binary edge frame, $B_t(i,j)$, is defined by

$$B_t(i,j) = \begin{cases} 1 & \text{if } S_t(i,j) > T_e \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where T_e is a predefined threshold.

The block is considered as having an edge if $\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} B_t(i,j) \geq T_{\text{count}}$ is satisfied, where T_{count} is the threshold of edge count. This is done in order to prevent false determination of edge blocks due to the presence of noise and consequently it can guarantee that motion vectors obtained in the EAS are more reliable. If the block is classified as an edge block, the search of this block proceeds to step 2; otherwise one of the conventional fast search algorithms such as the BBGDS [20] or the DS [21] is applied.

• ***Step 2: Adjustment of the Regular SPP***

Once the image feature description is determined, a matching evaluation function is required to show the degree of similarity between two descriptions. Usually the similarity between two descriptions is defined in the form of a cost function or a distance function, where these costs are expected to be minimized and are zero only if both descriptions are identical.

Binary edge points can be selected as image features for block matching, and details of such an approach can be found in reference [23] for locating the best fit of edge points from two different blocks based on distance minimization. Note that this straightforward approach to determine the distance function by computing the binary edge matching is very time-consuming because it considers all binary edge points across the reference image pixel by pixel. An efficient technique that reduces this running time is highly desirable. In this paper, we propose to reduce the computational burden by introducing the edge

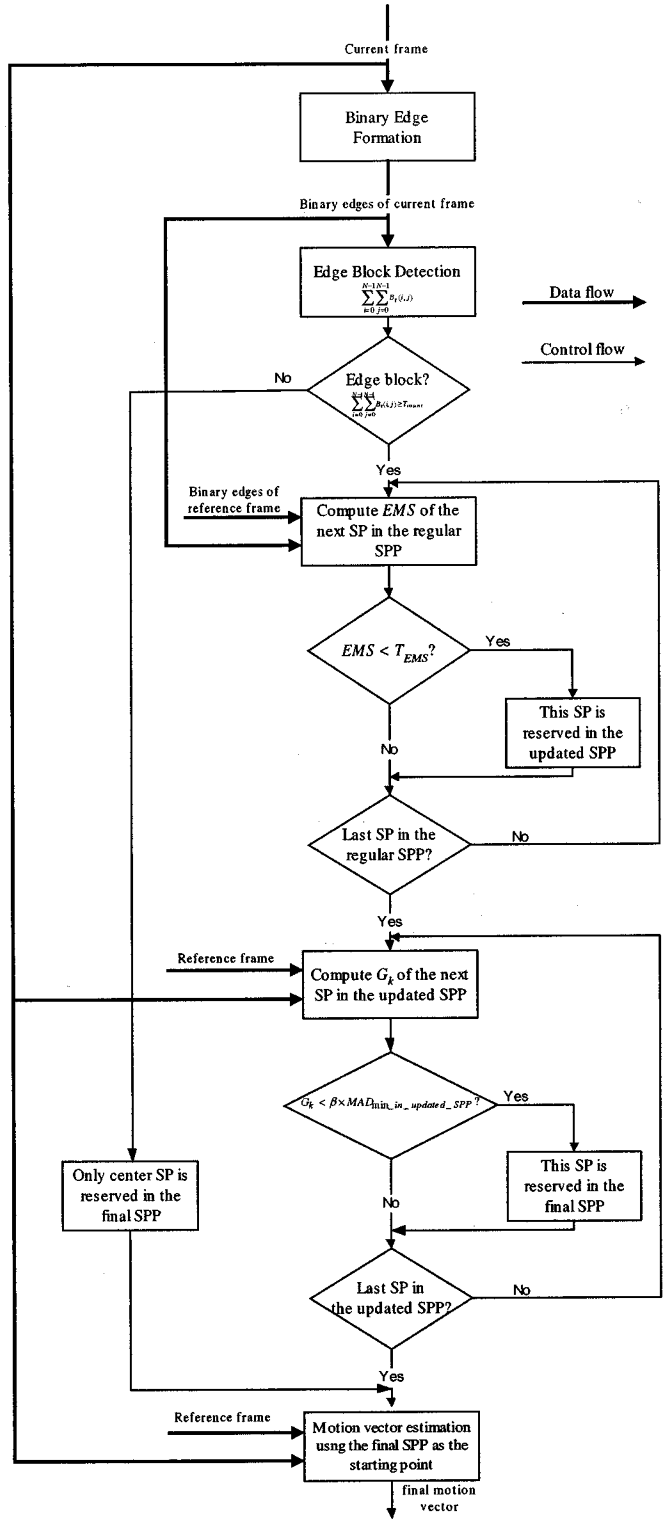


Fig. 5. Flowchart of the proposed EAS.

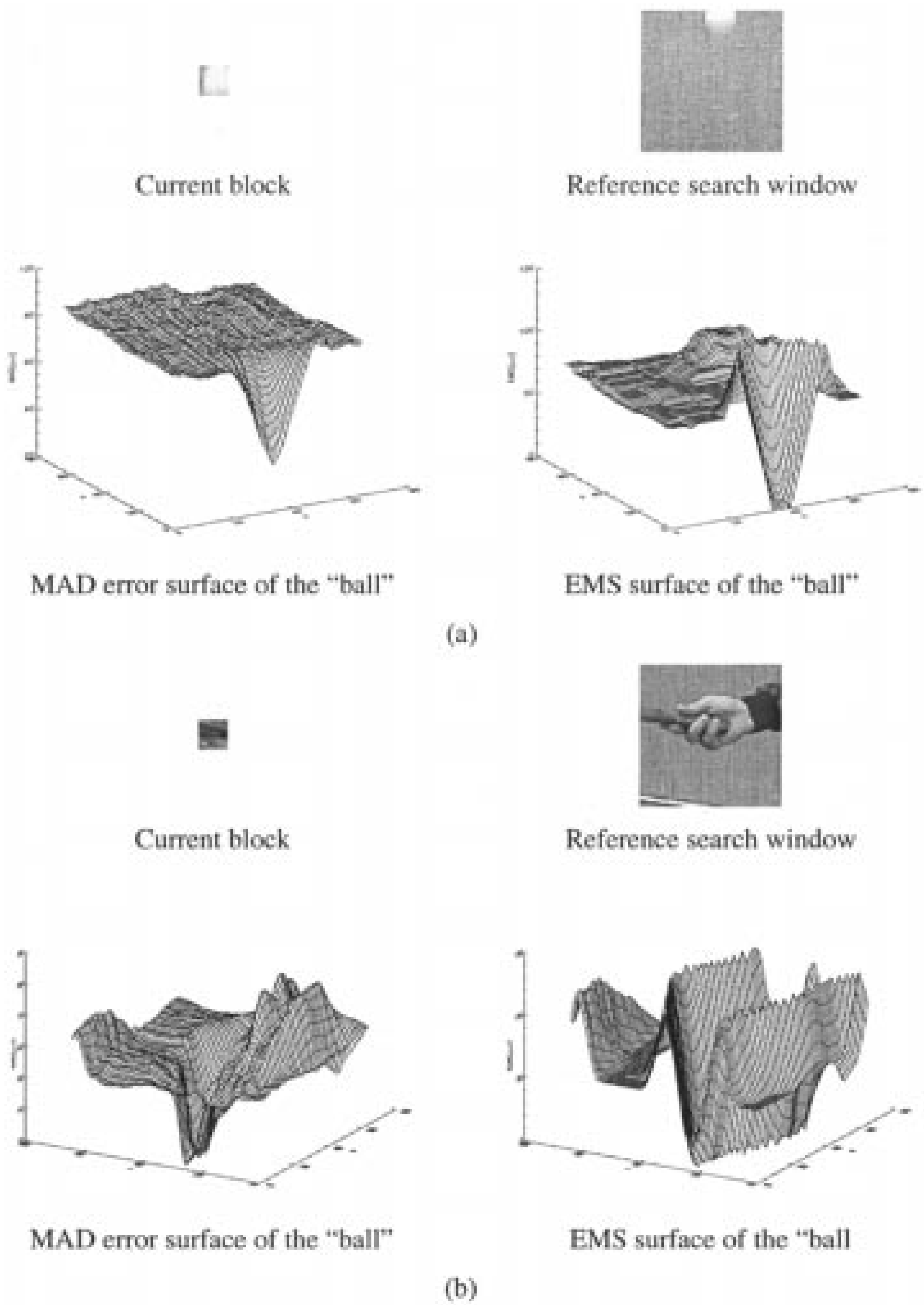


Fig. 6. Relationship between the MAD error surface and the proposed EMS surface.

matching score (EMS), as the cost function to adjust the regular SPP and to avoid the pixel-wise comparison. The EMS is defined as the difference between the sum of edge points of each block

$$\mathrm{EMS}(u,v) = \left| \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} B_t(i,j) - \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} B_{t-1}(i+u, j+v) \right| \tag{5}$$

where (u, v) denotes the location of the possible motion vector, and $B_t(\cdot,\cdot)$ and $B_{t-1}(\cdot,\cdot)$ refer to the binary edge blocks of the current frame(tth frame) and of the reference frame ($(t-1)$th frame) that is to be compared.

The adjustment of the regular SPP is based on the measure of how large the probability of being the global minimum of each possible matching pair between the current block and the block at the regular SPP is. The EMS is to consider that if the numbers of edge points in two blocks are similar, the block in the regular SPP has a large probability of being closest to the global minimum. In Fig. 6, the MAD surfaces and the EMS surfaces of two different blocks containing a ball or a racket moving against a background are plotted. We have found that the correlation between these two surfaces is very high and this further ensures that the motion search algorithm can be guided by the EMS. Thus a block in the regular SPP whose EMS is less than a pre-defined threshold, T_{EMS}, will be considered suitable for an interesting SP. In other words, this SP is reserved in the SPP. In the follow of our discussion, we refer the modified SPP to as the updated SPP. In order to normalize the thresholding, the T_{EMS} is proportional to the number of edge points of the current block. That is

$$T_{\mathrm{EMS}} = \alpha \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} B_t(i,j) \tag{6}$$

where α is a proportional constant.

• ***Step 3: 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 MAD matching criterion. Selection of the best matched SP as compared to other SPs in the updated SPP is based upon the MAD values, and it is defined as

$$G_k = \mathrm{MAD}_k - \mathrm{MAD}_{\mathrm{min\_in\_updated\_SPP}} \tag{7}$$

where k means to cover all selected SPs of the updated SPP, except the SP with the smallest value of MAD in the updated SPP, where $\mathrm{MAD}_{\mathrm{min\_in\_updated\_SPP}}$ and MAD_k are the smallest value in the updated SPP and the value of the MAD from the SP in the updated SPP, respectively. First, the SP with the smallest value must be reserved in 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 $\beta \times \mathrm{MAD}_{\mathrm{min\_in\_updated\_SPP}}$, 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, the SP is eliminated from the updated SPP. After examining all SPs in the updated SPP, the final SPP is formed.

• ***Step 4: Motion Vector Estimation Using the Final SSP as the Starting Point***

After the establishment of the final SPP, all SPs in the final SPP serve as the starting point of one of the conventional fast searching algorithms such as the BBGDS [20] or DS [21]. Finally, a search is conducted to find the minimum value of MAD.

D. Simulation Results

1) Quality Comparison: A series of computer simulations have been conducted to evaluate the performance of the proposed EAS. These include the "Table Tennis" and the "Football" sequences in SIF format. 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) between the estimated frame $\hat{I}_t(i,j)$ and the original frame $I_t(i,j)$, is used to compare the performance of the proposed algorithm with the related techniques in the literature. The MSE is defined as

$$\mathrm{MSE} = \frac{\sum_{i=0}^{L-1}\sum_{j=0}^{P-1}[I_t(i,j) - \hat{I}_t(i,j)]^2}{P \times L} \tag{8}$$

where $P \times L$ is the size of the image.

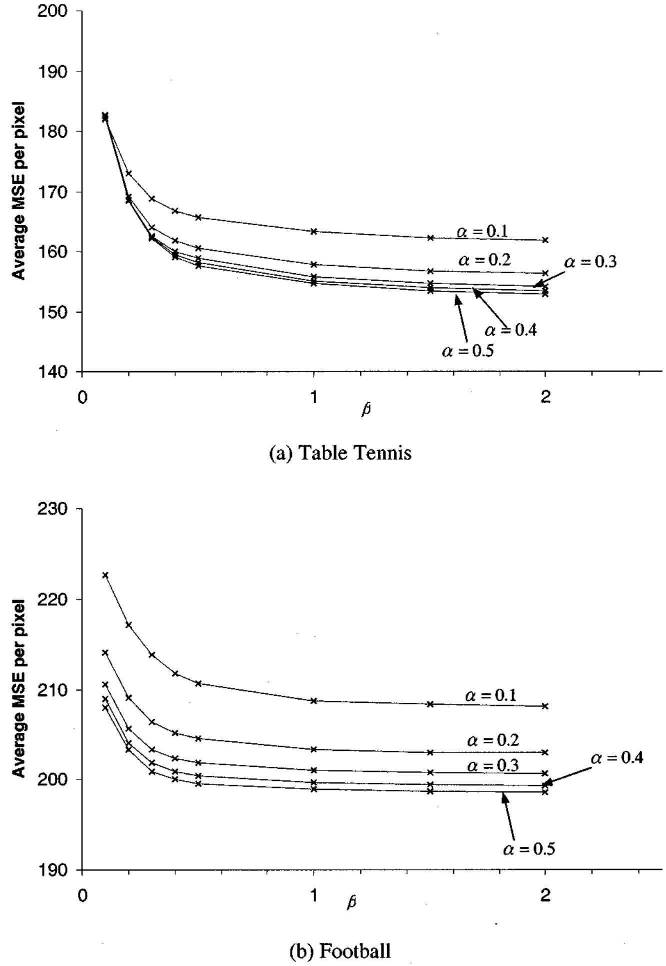


Fig. 7. MSE performance of the BBGDS with the EAS, plotted as a function of α and β. Each line represents the MSE performance for a fixed α under the variation of β. (a) Table Tennis and (b) Football.

Since our proposed EAS can work together with the conventional fast search algorithms, the BBGDS [20] and the DS [21] are employed to illustrate the performance of the EAS. To obtain the binary edge information, the T_e and T_{count} of the edge detection are set to 40 and 16, respectively. Consider Figs. 7 and 8, which plot the MSE performance and the number of search points required of the BBGDS with the EAS for all possible choices of α and β, which are used to control the number of SPs in the updated SPP and the final SPP respectively. These experimental results indicate that, for a fixed α, increasing β results in a roughly linear increase in the number of SPs and it improves the MSE performance. However, the MSE performance does not improve further when β is beyond 1 in both the "Table Tennis" and "Football" sequences. This indicates that some SPs are wasted when $\beta > 1$. Also, as α increases, the MSE performance improves as shown in Fig. 7. When α is beyond 0.3, the MSE performance does not improve significantly. A similar result is obtained when the DS is used instead of the BBGDS, as depicted in Figs. 9 and 10. Thus, by considering the tradeoff between the computational requirement and the quality of most image sequences, we have set the proportional constants, α and β, to 0.3 and 0.5, respectively for the rest of the comparison. The conventional methods for performance

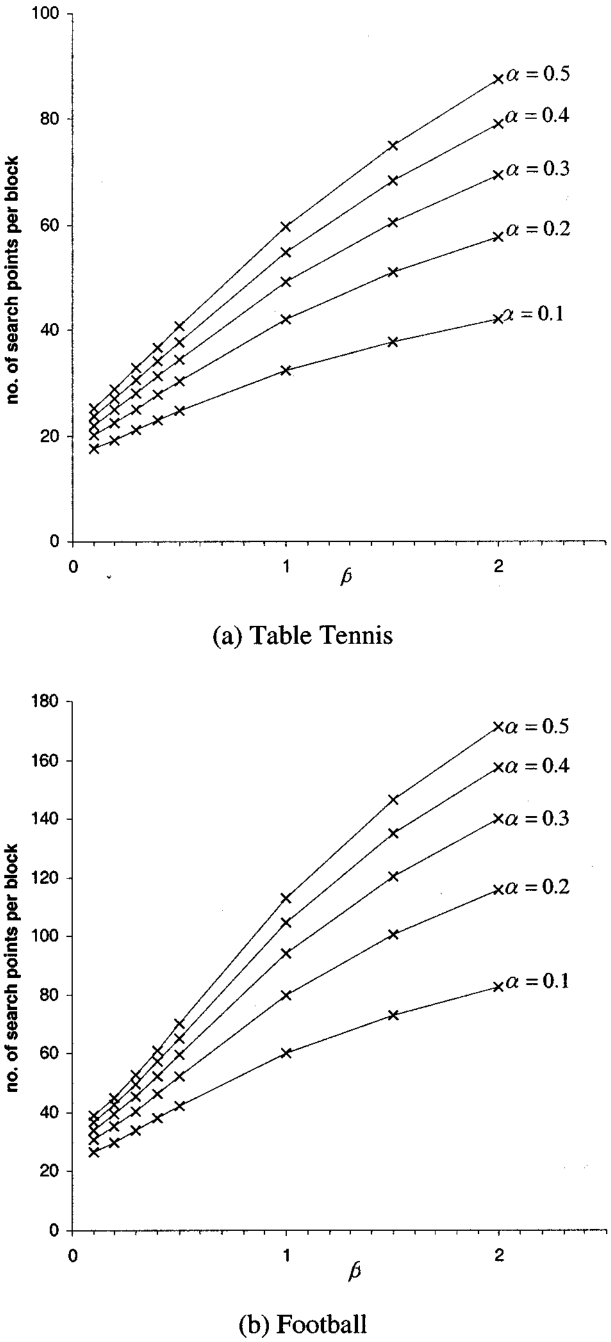


Fig. 8. Number of search points required per frame of the BBGDS with the EAS, plotted as a function of α and β. Each line represents the search point requirement for a fixed α under the variation of β. (a) Table Tennis and (b) Football.

comparison are the full search algorithm (FSA), the n-step hierarchical search algorithm(n-SHS) [16], the block-based gradient descent search algorithm (BBGDS) [20], and the Diamond search algorithm (DS) [21].

In the following subsection, we will analyze the MSE performance of the BBGDS and the DS with the help of our proposed EAS, as compared to the FSA, the n-SHS, the BBGDS without the EAS and the DS without the EAS. Figs. 11 and 12 show that there is a big prediction error in the n-SHS, the conventional BBGDS and the conventional DS as compared with that of the

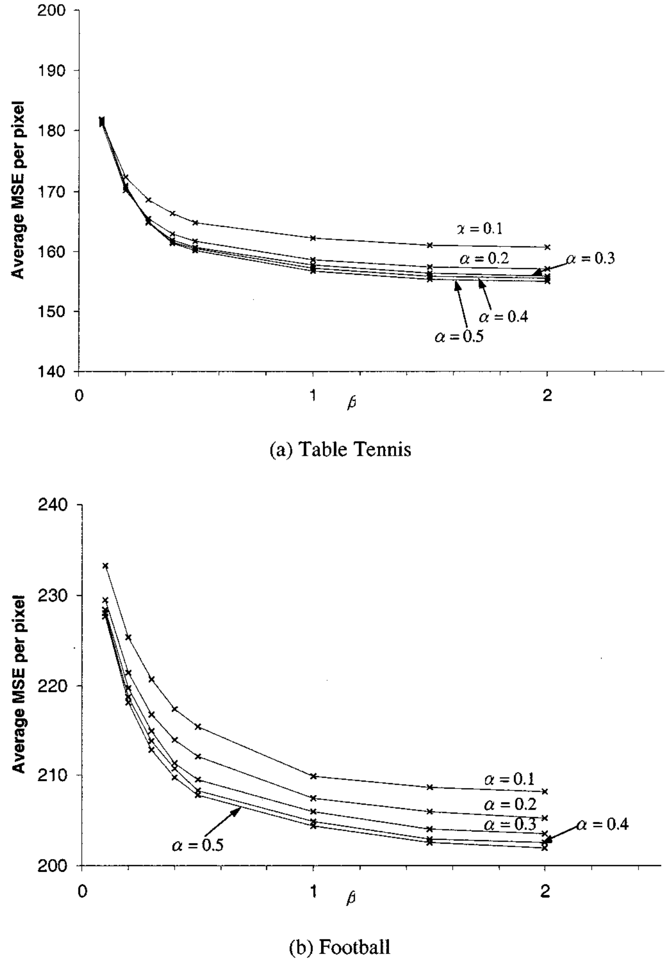


Fig. 9. MSE performance of the DS with the EAS, plotted as a function of α and β. Each line represents the MSE performance for a fixed α under the variation of β. (a) Table Tennis and (b) Football.

FSA. This is because the probability of occurrence of the situation as shown in Fig. 3(a) is more often in fast moving sequences. In this situation, an inappropriate choice was made in the early steps of the n-SHS. The unreliable stop in the search for the conventional BBGDS and DS implies that these types of algorithms are more easily trapped in a local minimum. However, our new EAS can resolve the problem of the local minimum by placing the checking block closest to the global minimum which is guided by the edge features. As shown in Figs. 11 and 12, the new EAS can strengthen both the BBGDS and DS, which are significantly better than that of the n-SHS, the conventional BBGDS and the DS, and their MSE performance are very close to the FSA.

2) Complexity Analysis and Discussion: In the following, the computational complexities of our proposed EAS are compared with those of the conventional algorithms including the FSA, the n-SHS, the BBGDS and the DS. In general, several factors need to be taken into account in comparing the cost associated with various algorithms. These factors include speed, chip area and power, and they can usually be traded with each other depending upon the architecture to be used, hence comparing the costs associated with various algorithms is not an easy task. Nevertheless, it is possible to choose simple ways to define

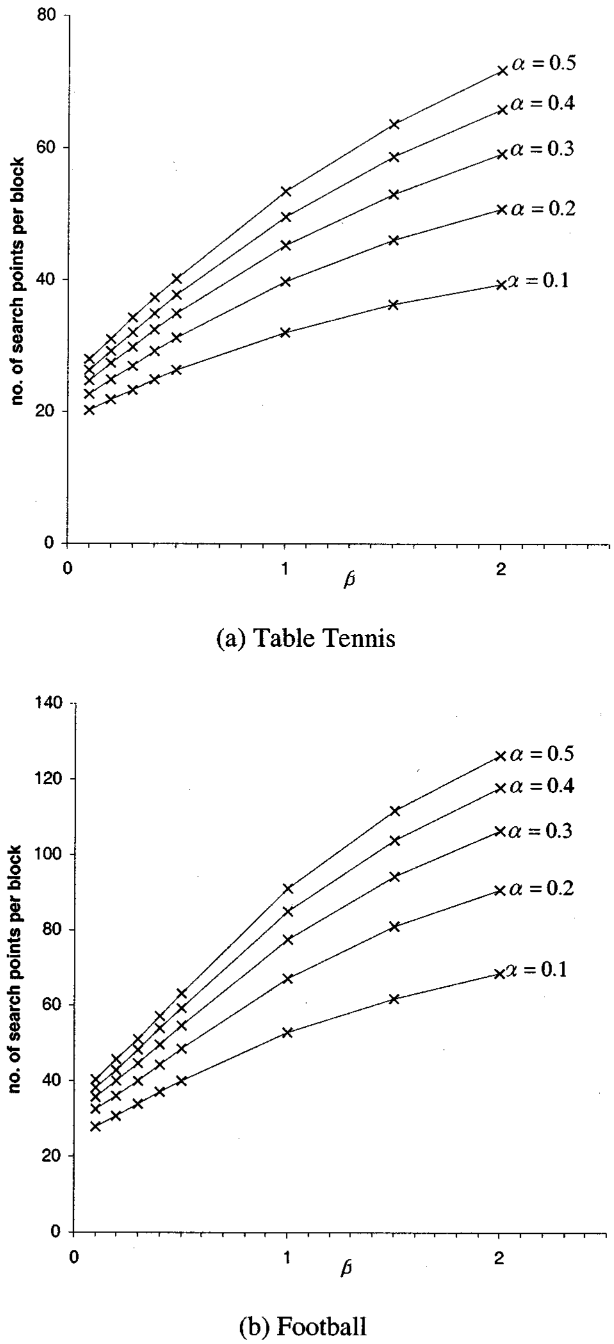


Fig. 10. Number of search points required per frame of the DS with the EAS, plotted as a function of α and β. Each line represents the search point requirement for a fixed α under the variation of β. (a) Table Tennis and (b) Football.

complexity. The fixed-point implementation of the proposed EAS is now compared with that of the FSA, that of the n-SHS, that of the BBGDS and that of the DS. The matching criterion as shown in (1) requires 2-D operations, i.e., $255(16 \times 16 - 1)$ additions, $256(16 \times 16)$ subtractions, and $256(16 \times 16)$ absolute conversions per search point are needed. Therefore, for its 961 search points in the search range of the FSA, each block requires 245 055 additions, 246 016 subtractions, and 246 016 absolute conversions. For its 33 search points of the n-SHS, each block requires 8415 additions, 8448 subtractions, and 8448 absolute. For the BBGDS, the DS and the search algorithms with the help of our proposed EAS, only the average number of search points per block for the entire sequence are reported. For the conventional BBGDS and DS, the number of search points required depends on whether the stop criterion is fulfilled. When our proposed EAS is used to strengthen the search algorithms, less search points are required for smooth region within the frame, whereas more search points are needed in regions containing many edges and motions, as they require more search points using the edge block matching criterion.

Apart from calculating the matching criterion, the formation of the final SPP is the major overhead of our proposed EAS. Its computational complexity is now examined. The formation of the final SSP consists of three parts: smoothing of the frame, taking directional derivatives of the frame, and adjusting the regular SPP. Step 3 in the EAS is not considered as overhead since its major computation has been taken into account in the required search points as previously mentioned. Smoothing simply involves taking the average of the input with the 5×5 window, and fast calculation for smoothing an image is developed in [23]. By exploiting the maximal data reusage, the computation for smoothing can be reduced drastically. The input image is smoothed in the j-direction and the sum of image pixels along j-direction $V(i,j)$ is calculated, which is defined as

$$V(i,j) = \sum_{h=-2}^{2} I(i, j+h) \tag{9}$$

where $I(i,j)$ is the input image at spatial location (i,j).

From Fig. 13, it is obvious that the difference between $V(i,j)$ and $V(i,j+1)$ is only two pixels and can be expressed as

$$V(i,j+1) = V(i,j) - I(i,j-2) + I(i,j+3). \tag{10}$$

Suppose the image size is $P \times L$, to obtain the summation of image pixels $V(i,j)$ for $j = 0$, requires $4P$ additions. Other $V(i,j)s$ can be obtained by using (10), requiring $2(L-5)$ additions. In the same way, if the above procedure is performed on the i-direction with $V(i,j)$ as the input, the smoothed image is obtained. From the above discussion, the total number of additions required per $N \times N$ block for the smoothing process of an image is

$$A_s = N^2 \times \frac{P[4 + 2(L-5)] + L[4 + 2(P-5)]}{PL}. \tag{11}$$

To detect edges, eleven additions, two absolute conversions and two multiplications by two are required to compute the $S_n(i,j)$ as defined in (2). Note that multiplication by two can be achieved by a simple shift, and the cost of this multiplication could therefore be neglected.

To adjust the regular SPP, 49 times of the EMS calculation and selection process, as shown in (5) and (6), respectively, are required in the case of the edge block. Since the calculation of the number of edge points within each block is performed by the binary counter, it can be neglected as compared with the operational effort of additions and subtractions. Thus, 49 subtractions and multiplications per block are required to adjust the regular SPP. Combining all of these, Table I shows a comparison of the operational requirements of the BBGDS with the help of the

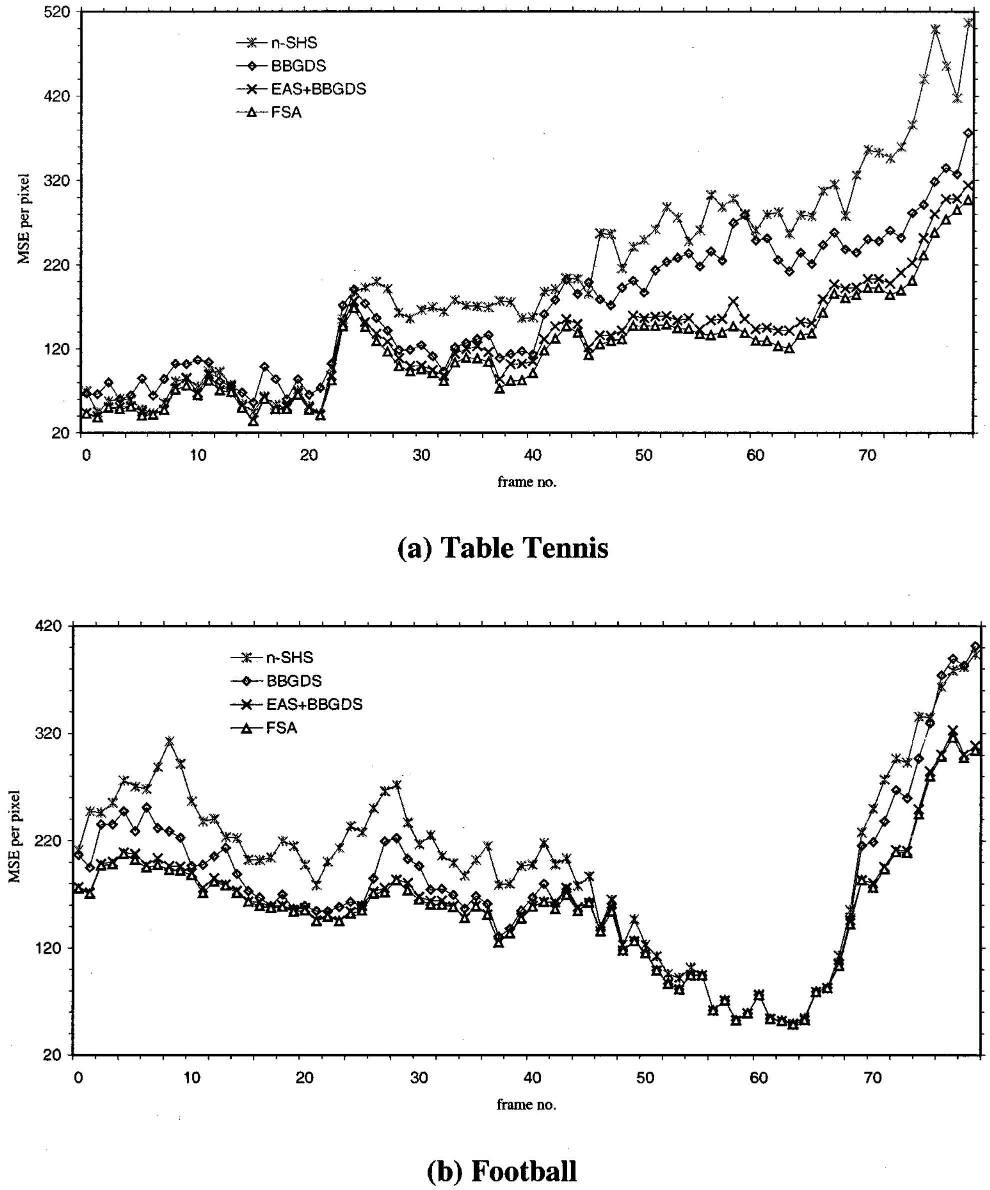


Fig. 11. Performance using the EAS to strengthen the BBGDS. (a) "Table Tennis" and (b) "Football."

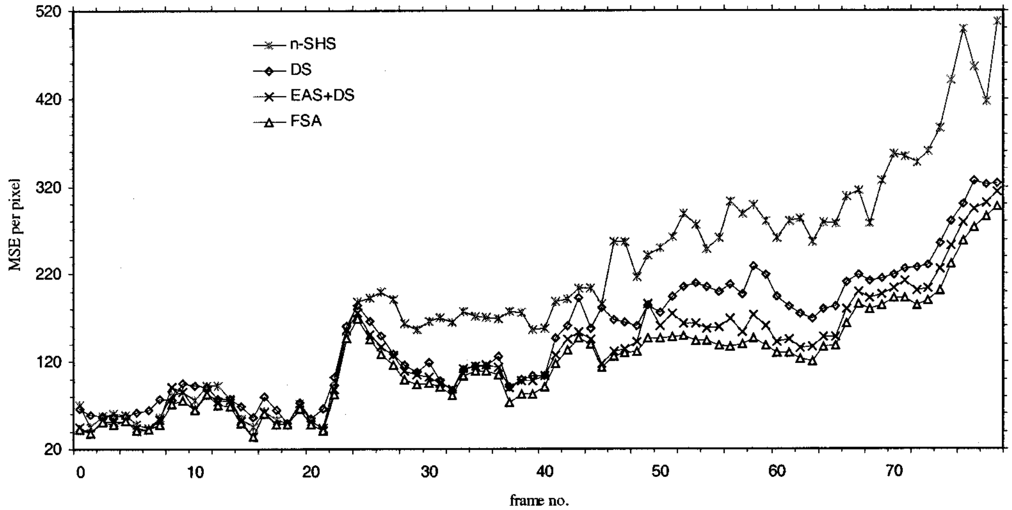
EAS, the DS with the help of the EAS, the BBGDS, the DS, the n-SHS and the FSA. The table shows that the algorithms with the proposed EAS require more computational effort as compared with the n-SHS, the BBGDS and the DS. It is due to the fact that the EAS can avoid the serious local minimum problem of the n-SHS, the BBGDS and the DS by involving a reasonable number of starting points, which have a high degree of similarity of edge points between the current block and the block in the search window. However, it has a speed-up of over 15 times as compared with the FSA.

III. New Edge-Oriented Block Motion Estimation Using EAS

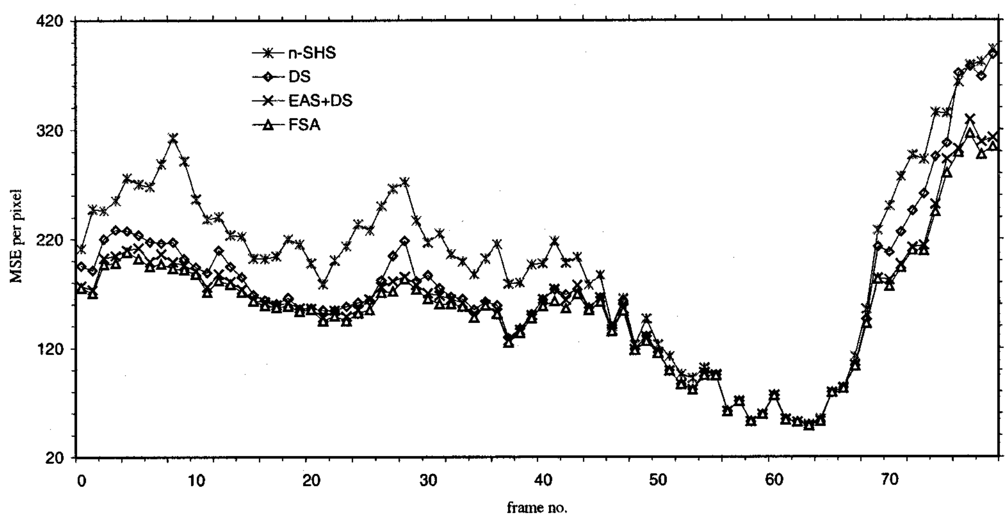
A. Description of the Technique

As mentioned in (1), the block matching algorithm is based on the matching of pixel intensities in 2-D blocks between two frames, and it is known as an intensity-based block matching algorithm. Despite its successful applications, it suffers from several drawbacks. One of the problems is that blocks located on the boundaries of moving objects are not estimated accurately. It causes poor motion-compensated prediction along the moving edges to which the human visual system is very sensitive. By considering the characteristics of block motions for typical image sequences, we suggested an intelligent classifier [23] to classify blocks into three categories which perform different modes of motion estimation.

- Type 1: Blocks in a still area or with a background of very slow motion. Minimal motion search is required. These blocks might have motion vectors which correspond to the previous frame.
- Type 2: Blocks that contain the boundary between the moving objects and the still area. In some cases, the intensity-based block motion estimation works well and obtains the true motion vector of the moving object. However, this is not the case if a large portion of pixels in the block is



(a) Table Tennis



(b) Football

Fig. 12. Performance using the EAS to strengthen the DS. (a) "Table Tennis" and (b) "Football."

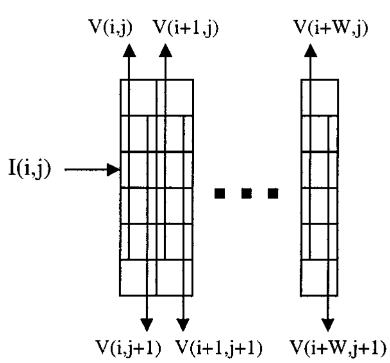


Fig. 13. Illustration of the data reuse of image smoothing.

occupied by the still area. Thus the motion vector dominated by the effect of the still area is unavoidably selected. Consequently, the edge would be lost in the motion-compensated frame. This situation supports the use of an edge matching criterion to replace the intensity-based matching criterion of block motion estimation, as defined in (12). The idea of edge matching is to try to track the corresponding edge in the previous frame, so that better moving edges can be predicted in the compensated frame. In this matching method, the true motion vectors of moving objects could be computed

$$\mathrm{EMAD}(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} |S_t(i,j) - S_{t-1}(i+u, j+v)|. \quad (12)$$

- Type 3: Blocks in a moving object on which the intensity-based block motion compensation works well. It can be seen that the motion in these blocks correlate highly

TABLE I
COMPARISON OF COMPUTATIONAL COMPLEXITY AND AVERAGE MSE FOR VARIOUS ALGORITHMS

	Additions/ Subtractions	Absolute Conversions	Multiplications	Average MSE
		Table Tennis		
FSA	491,071	246,016	—	146.74
n-SHS	16,863	8,448	—	237.66
BBGDS	7,016	3514	—	196.72
BBGDS + EAS	21,497	9,354	19	158.82
DS	8,190	4,103	—	178.86
DS + EAS	21,642	9,436	19	160.62
		Football		
FSA	491,071	246,016	—	190.70
n-SHS	16,863	8,448	—	239.6
BBGDS	10,629	5325	—	256.57
BBGDS + EAS	34329	15,782	19	201.81
DS	11,227	5,624	—	245.02
DS + EAS	31,800	14,522	19	209.87

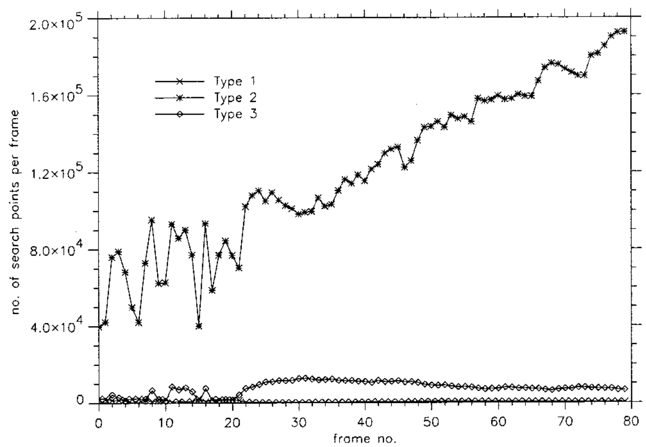


Fig. 14. Number of search points required per frame for different types of blocks in the EOBMA.

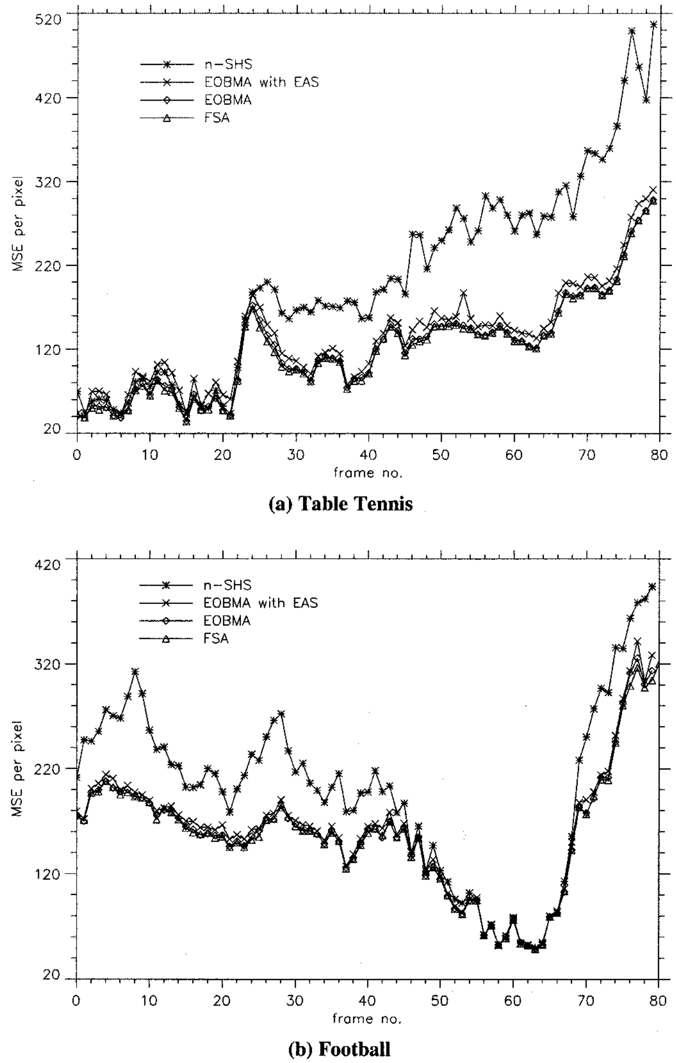


Fig. 15. MSE produced by different algorithms for video sequences (a) "Table Tennis" and (b) "Football."

with the surrounding edge blocks, provided that the motion vectors in the edge blocks truly represent the motion objects. Thus, the motion vectors of the edge blocks (type 2) are significant enough to develop an efficient motion estimation of blocks which are classified as type 3. If there are adjacent blocks for which the motion vectors have already been computed, the current block can make use of these motion vectors as the initial centers and employ the intensity-based block matching with a smaller displacement to compute its motion vector. But if no adjacent block has its motion vector computed, the computation of the motion vector of this block will be postponed until the motion vectors of the required adjacent blocks are available. The advantage of this motion estimation technique is that unnecessary computations are avoided so that the motion search can be conducted efficiently.

The experimental results in [23] show that the edge-oriented block matching algorithm (EOMBA) provides a better motion-compensated prediction along the moving edges in comparison with the traditional intensity-based block motion estimation methods. It is visually important to human perception. As mentioned before, the accuracy of the motion vectors of a type 2 block is critical in the EOBMA. Consequently, the FSA with a large search window is used to ensure its accuracy. Fig. 14 shows that 95% of the total search points required of the whole motion estimation process are performed for type 2 blocks. In order to increase the flexibility and practicability of the EOBMA, the computational burden of the motion estimation of type 2 blocks must be reduced.

TABLE II
COMPARISON OF COMPUTATIONAL COMPLEXITY AND AVERAGE MSE FOR VARIOUS ALGORITHMS

	Additions/ subtractions	Absolute Conversions	Multiplications	Average MSE
		Table Tennis		
FSA	491,071	246,016	—	146.74
n-SHS	16,863	8,448	—	237.66
EOBMA	200,410	98,995	—	152.53
EOBMA + EAS	64,984	31,139	19	167.24
		Football		
FSA	491,071	246,016	—	190.70
n-SHS	16,863	8,448	—	239.6
EOBMA	351,947	174,912	—	210.00
EOBMA + EAS	155,890	76,682	19	219.85

Since the ideas behind both the EOBMA and EAS have been developed according to the image edges, the proposed EAS is

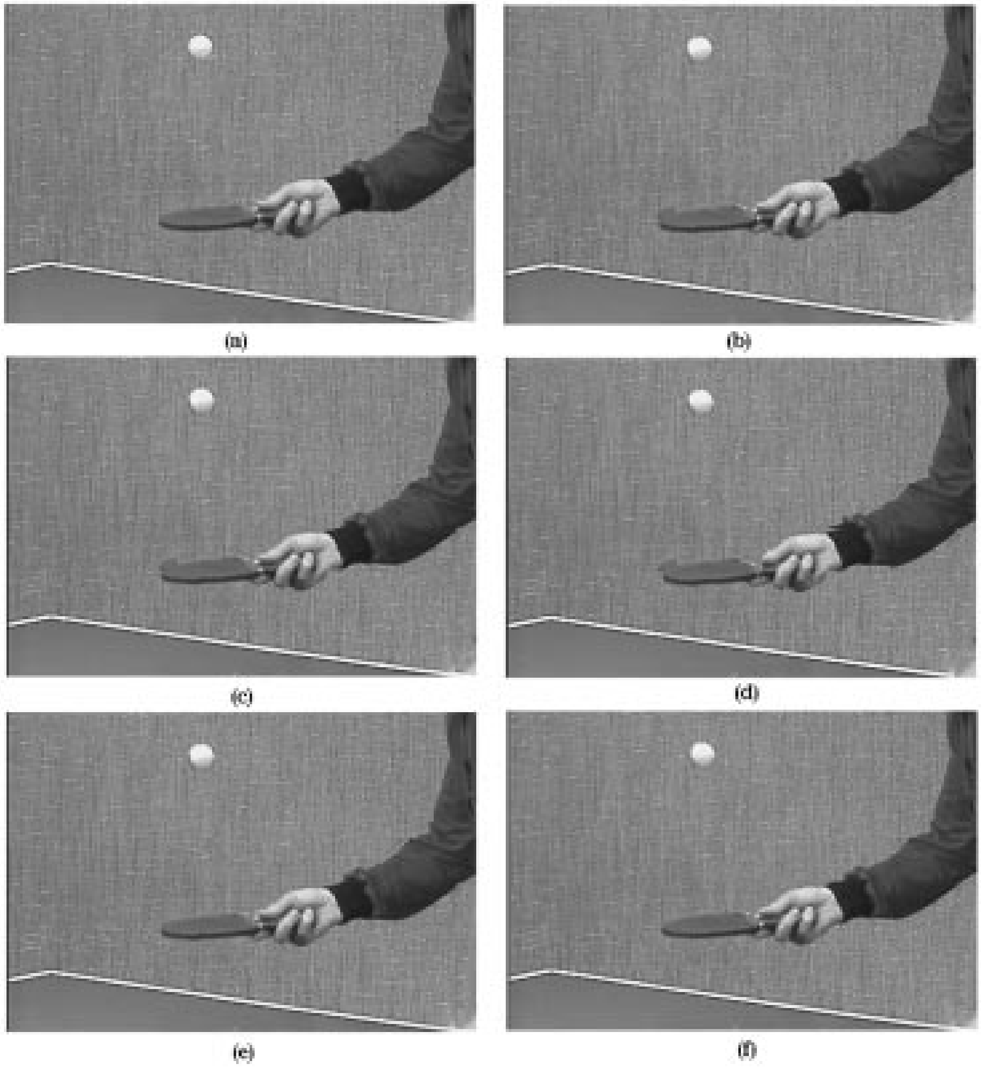


Fig. 16. Motion-compensated frame of the "Table Tennis" sequence. (a) Original frame, (b) FSA, (c) BBGDS with the proposed EAS, (d) n-SHS, (e) EOBMA, and (f) EOBMA with the proposed EAS.

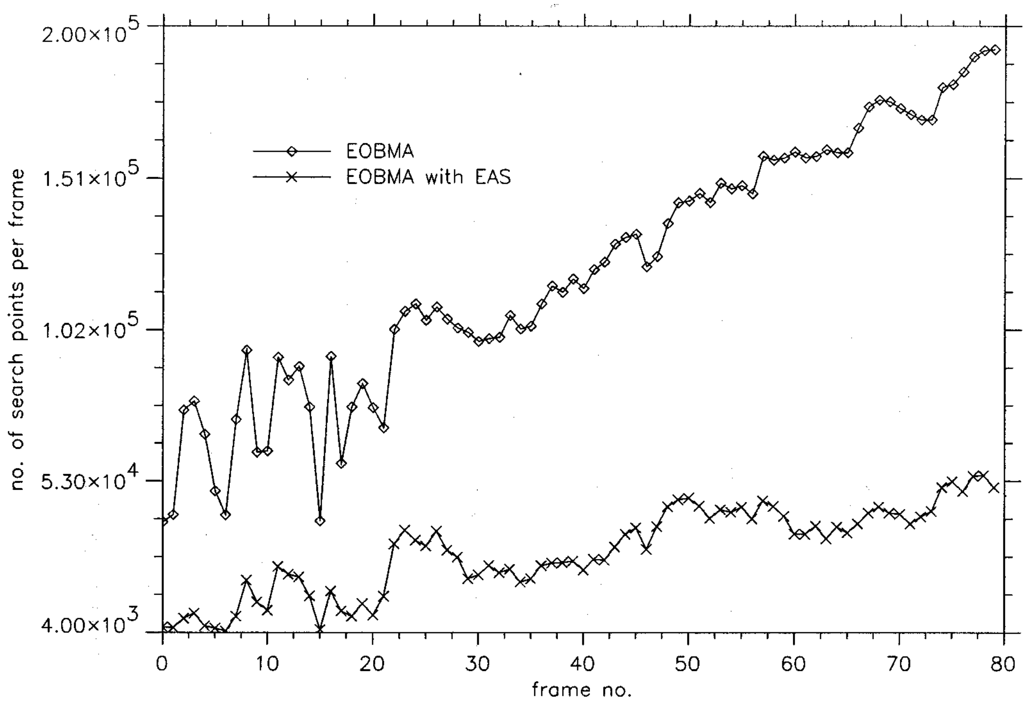
a possible solution when combined with the EOBMA. In the following simulation results and analysis, instead of the FSA, our EAS is used in the motion estimation of type 2 blocks in the EOBMA. Some encouraging results will be shown.

B. Simulation Results

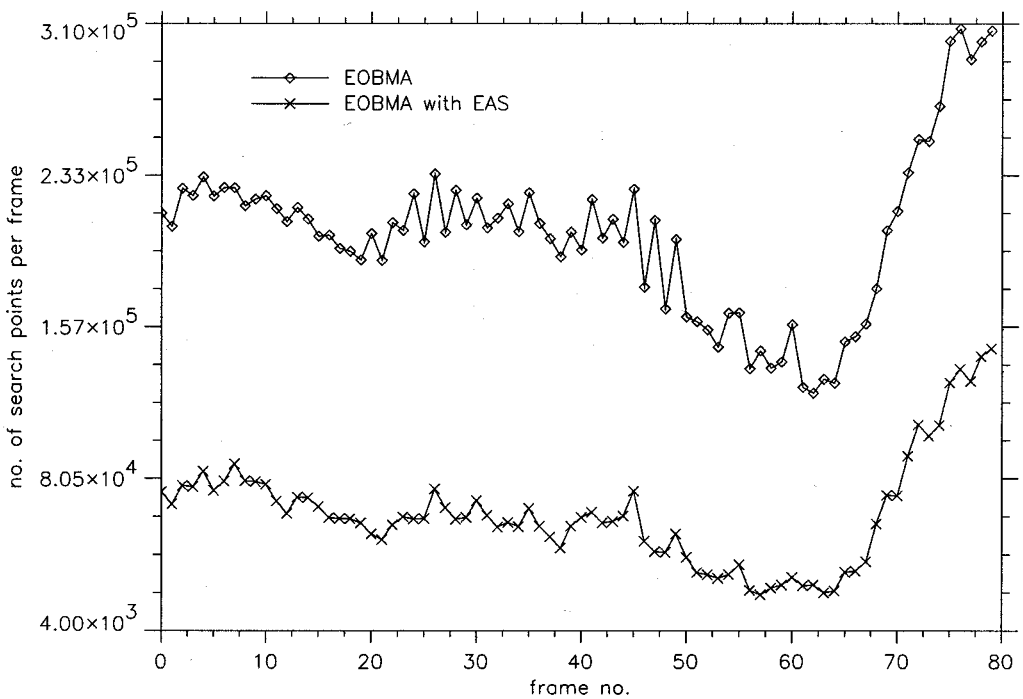
In this section, we are going to give experimental results on the performance of the proposed EAS when applied to the EOBMA from the viewpoint of the computational complexity as well as the accuracy of the estimated motions. Again, the "Football" and the "Table Tennis" image sequences with the size in SIF format have been used as test sequences for comparison purposes. These sequences have various motion characteristics such as camera panning, zooming, and motion of human body.

Fig. 15 shows the results using the MSE for the motion-compensated frame by using the EOBMA, the EOBMA with the help of the EAS, the n-SHS and the FSA. It can be seen from this figure that both edge-oriented block motion estimation algorithms outperform the n-SHS for the "Table Tennis" and the "Football" sequences. Also, Table II shows the average MSE of various algorithms for both image sequences. As it is seen, our proposed EOBMA with and without the help of the EAS, compares favorably with the traditional intensity-based FSA in terms MSE performance.

Although the MSE measure has a good physical and theoretical basis, it could correlate poorly with the subjectively judged distortion of an image. This is mainly due to the fact that the human visual system does not process images in a point-by-point fashion, but extracts certain spatial, temporal, and chromatic features. Thus, the MSE measure cannot reflect image characteristics such as edge fidelity, image contrast, and other similar characteristics [27], [28], and hence subjective quality is also a very important measure. Edge-oriented algorithms aim to obtain more accurate motion prediction along the moving edges, to which the human visual system is very sensitive. Fig. 16 shows the motion-compensated frames of the "Table Tennis" sequence produced by the EOBMA, the EOBMA with the EAS, the BBGDS with the EAS, the FSA and the n-SHS. Fig. 16(b), (c) and (d) show the incorrect prediction of the racket along



(a) Table Tennis



(b) Football

Fig. 17. Number of search points required per frame for type 2 block in the EOBMA with and without EAS.

the edge produced by the traditional FSA, the BBGDS with the EAS and n-SHS, while the edge of the racket can be preserved by using the edge-oriented block motion estimation algorithms, the EOBMA with and without the help of the EAS, as shown in Fig. 16(e) and (f) respectively. This visual result indicates that the proposed EAS technique when applied to the EOBMA is successful and is able to preserve the advantage of the EOBMA. It is even better than the FSA and the BBGDS with the EAS in terms of subjective view, though the objective measure appears to be a little inferior. Since our algorithm reduces the number of edge mismatches in the prediction frame, it could also contain less high frequency information in the prediction error frame, so that the number of bits required to code the DCT coefficients is reduced. Besides, it can remove the most visually disturbing artifacts, therefore the frame produced by it seems virtually error free. In low-bit-rate applications as to which there is insufficient bandwidth to reconstruct the prediction error adequately, the artifact produced by the motion estimation can remain in the final decoded frame of the traditional method. However, our new edge-oriented block motion estimation algorithms can achieve a good subjective quality, as shown in Fig. 16. Table II and Fig. 17 show that the EAS proposed in this paper can significantly reduce the computational complexity of type 2 blocks of the EOBMA and thus the overall complexity is dramatically reduced. This implies that the new EAS renders more useful our previously proposed EOBMA.

IV. Conclusions

In this paper, a fast search algorithm for block motion estimation has been proposed. The proposed algorithm generally includes a matching process to track edge primitives from frame to frame in a sequence of images, hence we can consider it as an edge-assisted search algorithm, EAS. Edge features have been used for the adjustment of the start point patterns (SPP) of search windows such that a limited number of starting points can still provide a high enough chance of catching the global minimum. First, this method estimates an initial probability of being the global minimum of each possible matching pair between the current block and the block at the SPP. The SPP is then updated based on the EMS which is introduced to consider the degree of similarity of edge points between two blocks. We have demonstrated that the correlation between the EMS and the true motion vector is very high and it can ensure that the motion search algorithm can be guided by the EMS. We have tested the proposed EAS using a number of image sequences, including the “Table Tennis” and the “Football,” and found that it can reduce the heavy computational burden of the full search algorithm without significantly increasing the prediction error of the motion-compensated frame. The EAS is significantly better than those of the widely known search algorithms such as the n-SHS [16], the BBGDS [20] and the DS [21], and shows great improvement in the accuracy of the block motion estimation.

The proposed EAS can also work in conjunction with the EOBMA [23]. Since both of the algorithms are developed based on image features, they can take advantages of each other. For example, the EAS can reduce the computational burden of the EOBMA. Thus, it is able to enhance the flexibility and practicability of the EOBMA. Experimental results show that the EOBMA with the help of the EAS has comparable mean square error performance as compared to the traditional intensity-based block matching algorithm using the exhaustive full search, while it is a significant improvement over the n-step hierarchical search algorithm. However, the poor prediction along the moving edges, which is very annoying around moving objects, is substantially reduced by our proposed algorithm. In addition, since edges are more closely tied to physical features in a scene, when compared to individual pixel intensities, accurate moving edges are likely to be useful in other processing parts of a video compression system.

However, because the EAS is not highly regular, hardware implementation is difficult. On the other hand, as general-purpose processors are becoming more and more powerful, software encoding will likely be possible, and it is the trend of the future development of video processing. As a concluding remark, we believe that the results of our work will certainly be useful for the future development of software codecs.

References

[1] A. N. Netravali and J. O. Limb, “Picture coding: A review,” *Proc. IEEE*, vol. 68, pp. 366–406, Mar. 1980.

[2] A. K. Jain, “Image data compression: A review,” *Proc. IEEE*, vol. 69, pp. 349–389, Mar. 1981.

[3] H. G. Musmann, P. Pirsch, and H. J. Grallert, “Advances in picture coding,” *Proc. IEEE*, vol. PROC-73, pp. 523–548, Apr. 1985.

[4] F. Dufaux and F. Moscheni, “Motion estimation techniques for digital TV: A review and a new contribution,” *Proc. IEEE*, vol. 83, pp. 858–876, June 1995.

[5] M. Bierling, “Displacement estimation by hierarchical block matching,” *Proc. SPIE*, vol. 1001, pp. 942–951, 1988.

[6] H. Gharavi and M. Mills, “Blockmatching motion estimation algorithms—new results,” *IEEE Trans. Circuits Syst.*, vol. 37, pp. 649–651, May 1990.

[7] M. Ghanbari, “The cross-search algorithm for motion estimation,” *IEEE Trans. Commun.*, vol. 38, pp. 950–953, July 1990.

[8] L. G. Chen, W. T. Chen, Y. S. Jehng, and T. D. Chiueh, “An efficient parallel motion estimation algorithm for digital image processing,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 1, pp. 378–385, Dec. 1991.

[9] B. Liu and A. Zaccarin, “New fast algorithms for the estimation of block motion vectors,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, pp. 148–157, Apr. 1993.

[10] H. M. Jong, L. G. Chen, and T. D. Chiueh, “Accuracy improvement and cost reduction of 3-step search block matching algorithm for video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp. 88–90, Feb. 1994.

[11] M. J. Chen, L. G. Chen, and T. D. Chiueh, “One-dimensional full search motion estimation algorithm for video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp. 378–385, Oct. 1994.

[12] Y. L. Chan and W. C. Siu, “Adaptive multiple-candidate hierarchical search for block matching algorithm,” *Electron. Lett.*, vol. 31, no. 19, pp. 1637–1639, Sept. 1995.

[13] ——, “New adaptive pixel decimation for block motion vector estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 113–118, Feb. 1996.

[14] 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.

[15] Y. L. Chan and W. C. Siu, “On block motion estimation using a novel search strategy for an improved adaptive pixel decimation,” *J. Vis. Commun. Image Represent.*, vol. 9, no. 2, pp. 139–154, June 1998.

[16] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, “Motion compensated interframe coding for video conferencing,” in *Proc. Nat. Telecommunications Conf.*, Nov./Dec. 1981, pp. G.5.3.1–G.5.3.5.

[17] J. R. Jain and A. K. Jain, “Displacement measurement and it’s applications in interframe coding,” *IEEE Trans. Commun.*, vol. COMM-29, pp. 1799–1808, July 1981.

[18] R. Srinivasan and K. R. Rao, “Predictive coding based on efficient motion estimation,” *IEEE Trans. Commun.*, vol. COMM-33, pp. 1011–1015, Sept. 1985.

[19] 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. 438–442, Aug. 1994.

[20] 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, pp. 419–421, Aug. 1996.

[21] 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.

[22] K. H.-K. Chow and M. L. Liou, “Genetic motion search algorithm for video compression,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, pp. 440–445, Dec. 1993.

[23] Y. L. Chan and W. C. Siu, "Edge oriented block motion estimation for video coding," *Proc. Inst. Elect. Eng.*, vol. 144, no. 3, pp. 136–144, June 1997.

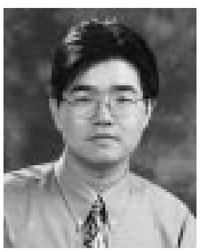
[24] *Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1,5 Mbit/s—Part 2: Video*, 1993.

[25] *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video*, 1996.

[26] C. Lee, F. V. M. Catthoor, and H. J. De Man, "An efficient ASIC architecture for a time edge detection," *IEEE Trans. Circuits Syst.*, vol. 36, no. 10, pp. 1350–1360, 1989.

[27] J. O. Limb, "Distortion criteria of the human viewer," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, pp. 778–793, Dec. 1979.

[28] D. J. Granrath, "The role of human visual models in image processing," *Proc. IEEE*, vol. 69, pp. 552–560, May 1981.



Yui-Lam Chan received the B.Eng.(Hons.) degree (with First Class Honors) and the Ph.D. degree from the Hong Kong Polytechnic University, Hong Kong, in 1993 and 1997, respectively.

He joined the Hong Kong Polytechnic University in 1997 and is now a Research Fellow in the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering. He has published more than 20 research papers in various international journals and conferences. His research interests include multimedia technologies, signal processing, image and video compression, video transcoding, video conferencing, and digital TV.

Dr. Chan is the recipient of more than ten significant prizes, scholarships, and fellowships for his outstanding academic achievements, such as the Champion of the Varsity Competition in Electronic Design in 1993 and the Sir Edward Youde Memorial Fellowship and the Croucher Foundation Scholarships.

Wan-Chi Siu (S'77–M'77–SM'90) received the Associateship degree from The Hong Kong Polytechnic University (formerly the Hong Kong Polytechnic), the M.Phil. degree from The Chinese University of Hong Kong, and the Ph.D. degree from Imperial College of Science, Technology and Medicine, London, U.K. in 1975, 1977, and 1984, respectively.

He was with The Chinese University of Hong Kong from 1975 to 1980. He joined The Hong Kong Polytechnic University as a Lecturer in 1980 and has been Chair Professor since 1992. He took up administrative duties as Associate Dean of the Engineering Faculty from 1992 to 1994, and was Head of the Department of Electronic and Information Engineering from 1994 to 2000. He has been Director of the Centre for Multimedia Signal Processing and Dean of Engineering Faculty of the same university since September 1998 and September 2000, respectively. He has published more than 200 research papers, and his research interests include digital signal processing, fast computational algorithms, transforms, image and video coding, and computational aspects of pattern recognition and neural networks. He is a member of the editorial board of the *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology* and the *EURASIP Journal on Applied Signal Processing*, and an overseas member of the editorial board of the *IEE Review*.

Dr. Siu was a Guest Editor of a special issue of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II published in May 1998. He was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II from 1995 to 1997. He was General Chairman of the International Symposium on Neural Networks, Image and Speech Processing (ISSIPNN'94), and a Co-Chair of the Technical Program Committee of the IEEE International Symposium on Circuits and Systems (ISCAS'97) which were held in Hong Kong in April 1994 and June 1997, respectively. He is now the General Chair of the 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMVSP'2001) and the 2003 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'2003) which will be held in Hong Kong. From 1991 to 1995, he was a member of the Physical Sciences and Engineering Panel of the Research Grants Council (RGC), Hong Kong Government, and in 1994, he chaired the first Engineering and Information Technology Panel to assess the research quality of 19 Cost Centers (Departments) from all universities in Hong Kong. He is an Honorary Professor of the South China University of Technology. He was the keynote speaker of several conferences and has received many awards, including the Best Paper Award (ICNNSP'95) in 1995, the Distinguished Presenter Award of the PolyU in 1997, and the IEEE Third Millennium Award for Outstanding Achievements and Contributions in 2000. He is a Chartered Engineer, a Fellow of the IEE and the HKIE, and has been listed in *Marquis Who's Who in the World*, *Marquis Who's Who in Science and Engineering*, and other citation biographies.