Adaptive Motion Tracking Block Matching Algorithms for Video Coding

Jie-Bin Xu, Lai-Man Po, and Chok-Kwan Cheung

***Abstract*—In most block-based video coding systems, the fast block matching algorithms (BMA's) use the origin as the initial search center, which may not track the motion very well. To improve the accuracy of the fast BMA's, a new adaptive motion tracking search algorithm is proposed in this paper. Based on the spatial correlation of motion blocks, a predicted starting search point, which reflects the motion trend of the current block, is adaptively chosen. This predicted search center is found closer to the global minimum, and thus the center-biased BMA's can be used to find the motion vector more efficiently. Experimental results show that the proposed algorithm enhances the accuracy of the fast center-biased BMA's, such as the new three-step search, the four-step search, and the block-based gradient descent search, as well as reduces their computational requirement.**

***Index Terms*— Motion analysis, motion compensation, video coding.**

I. Introduction

RECENTLY, great interest has been devoted to the study of different alternatives and approaches to the problem of video compression. The high correlation between successive frames of a video sequence makes it possible to achieve high coding efficiency by reducing the temporal redundancy. Motion-compensated video coding techniques are extensively used to exploit the temporal redundancy between successive frames. The most popular motion compensation method so far has been the block-based motion estimation, which uses a block-matching algorithm (BMA) to find the best matched block from a reference frame. This approach is adopted in various video coding standards such as ITU-T H.261 [1] and MPEG-1/2 [2], [3]. If the performance in terms of prediction error is the only criterion for a BMA, full search (FS) is the best and simplest BMA. However, its computational requirement is often too high for real-time implementation. This has led to the development of many fast BMA's [4]–[13]. Some well-known and recently developed examples are the three-step search (3SS) [4], the two-dimensional-logarithm search (LOGS) [5], the new three-step search (N3SS) [7], the four-step search (4SS) [10], and the block-based gradient descent search (BBGDS) [11]. However, most of these fast hierarchical BMA's use the origin of the searching window as the initial search center and have not exploited the motion correlation of the blocks among the same image moving object. To improve the fast BMA's accuracy, the motion correlation between the neighboring blocks can be used to predict an initial search center that reflects the current block's motion trend, and then the final motion vector can be efficiently found by the center-biased BMA's such as the N3SS, 4SS, and BBGDS. Because a proper predicted initial center makes the global optimal minimum closer to the predicted search center, the center-biased BMA's should increase the chance of finding the global minimum with lower search points.

In this paper, we propose a new block-based adaptive motion tracking search algorithm for fast block motion estimation. In this algorithm, the correlation of the spatially neighboring motion vectors is considered to track the current block's motion. The neighboring motion vectors had been used as an offset vector to track the motion of the current block in [12]. In this paper, we use this information to predict the initial search center, and experimental results show that the predicted center is closer to the global minimum. Thus, center-biased BMA's such as the N3SS, 4SS, and BBGDS are used to refine the motion vector.

The rest of this paper is organized as follows. Section II discusses the interblock motion correlation. The adaptive motion tracking block matching algorithm is described in Section III. Section IV reports the simulation results, and conclusions are given in Section V.

II. Interblock Motion Correlation

Motion objects often cover many small blocks in a general moving scene, such that the motion fields of the spatial neighbor blocks may be very similar. In addition, due to the continuity of motion in the temporal direction, the motion fields of the temporal neighbor blocks may be highly correlated. In other words, the motion field of the current block can be tracked from the neighbor blocks' motion fields in the temporal or spatial direction. However, when the motion of objects changes its direction abruptly or the speed of motion is not steady, it is not effective to track the motion from the previous-frame motion fields in the neighborhood of the current block. Moreover, to keep the previous-frame motion vectors in the decoder requires a large memory buffer, which will complicate the system. Thus, we only consider the interblock spatial correlation for the motion prediction.

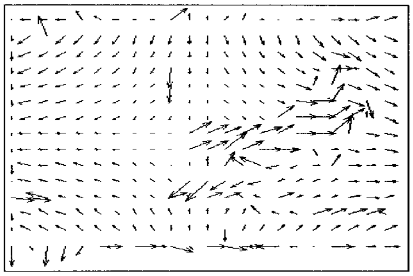
Fig. 1(a) and (b) gives an example of the motion vectors diagram and the corresponding picture for the *Tennis* sequence at frame 50, which are obtained by FS with a ± 7 search region. The scene at this moment contains fast motion objects and the camera zooming out. It can be observed that the directions and magnitudes of the motion vectors among the current block and its neighbor blocks are very similar if they are in the same object. That means spatial neighbor blocks' motion vectors are a good estimation of the current block's motion vector if we can determine that they are in the same object.

In our work, four causal neighbor blocks as shown in Fig. 2 are chosen for motion tracking. $B0$ is the current block.

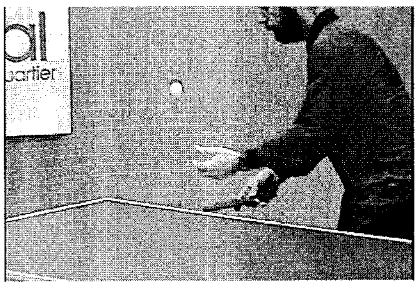
Manuscript received August 25, 1997; revised April 22, 1999. This paper was recommended by Editor-in-Chief W. Li.

The authors are with the CityU Image Processing Lab, Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong China (e-mail: jiebin@image.cityu.edu.hk; eelmpo@cityu.edu.hk).

Publisher Item Identifier S 1051-8215(99)08181-1.



(a)



(b)

Fig. 1. (a) The motion vectors diagram of the *Tennis* sequence at frame 50 and (b) the corresponding picture.

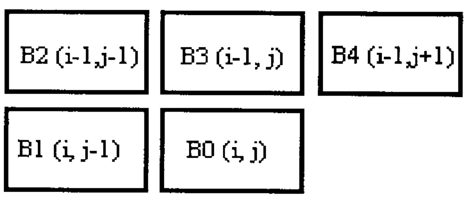
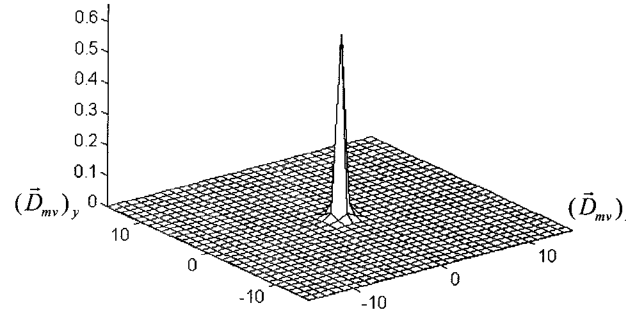


Fig. 2. Geometry of the four causal neighbor blocks.

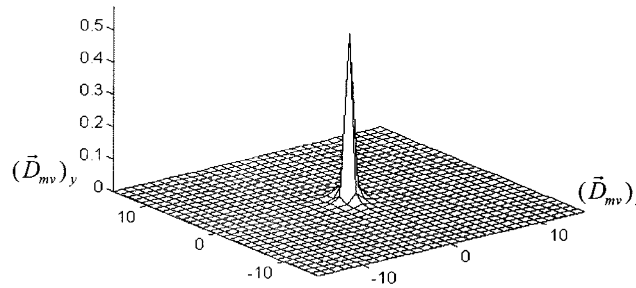
$B1$ represents the previous block in the horizontal direction, and $B2$, $B3$, and $B4$ are those in the vertical direction. The interblock motion correlation is defined by the displacement between the current block's motion vector and the mean motion vector of its four neighbors, which is formulated as

$$\vec{D}_{\rm mv} = \vec{V}_{\rm B0} - \frac{1}{4}\sum_{i=1}^{4} \vec{V}_{\rm Bi} \tag{1}$$

where $\vec{D}_{\rm mv}$ is the displacement and $\vec{V}_{\rm Bi}$, $i = 0, 1, 2, 3, 4$, are the true motion vectors corresponding to the blocks as shown in Fig. 2. When the magnitude of $\vec{D}_{\rm mv}$ is small, the current block's motion vector should highly correlate with its neighbors'. The $\vec{D}_{\rm mv}$ distributions based on the FS algorithm for the *Football* and *Tennis* sequences are shown in Fig. 3(a) and (b), respectively. The search region is ± 7 pixels in both the horizontal and the vertical directions, and the block size is 16×16. The distribution is based on the 22 400 $\vec{D}_{\rm mv}$ in each sequence, and each sequence has 80 frames with 280 blocks per frame. From these statistical data, we can find that there are nearly 90% of the blocks with the $\vec{D}_{\rm mv}$ inside the 5×5 area. As we will see in the next section, this simple measure for the interblock correlation can track the current block's motion and provide a robust initial search center prediction.



(a)



(b)

Fig. 3. Interblock motion vector correlation for the (a) *Football* and (b) *Tennis* sequences.

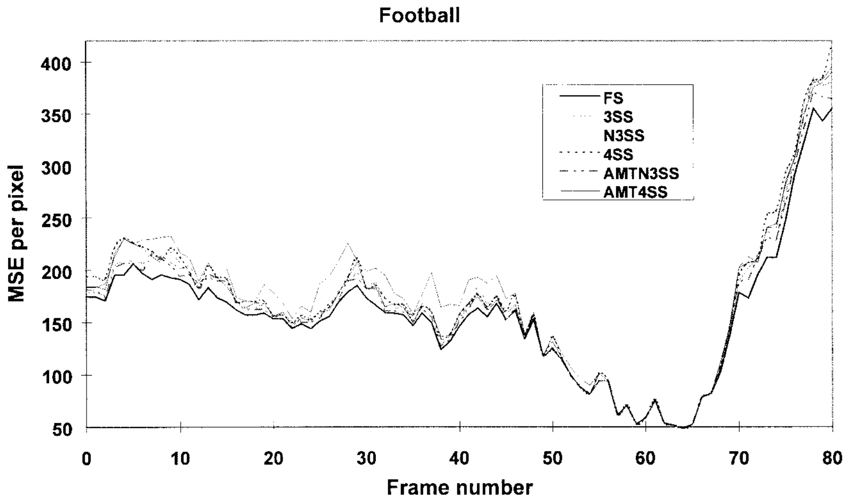
III. Adaptive Motion Tracking Block Matching Algorithm

The purpose of the motion tracking is to make the search region follow with the moving object by selecting a proper initial search center. The major advantage is that it can increase the chance of finding the true motion vector and reduce the computational requirement if the center-biased BMA's are used. It is because the halfway-stop technique of the center-biased BMA's can speed up the blocks matching with shorter distance between the starting search point and the global optimum point. More precisely, the proposed adaptive motion tracking search algorithm has two stages. The first stage is an initial search center prediction using the four causal neighbor motion vectors. The second stage is a center-biased fast BMA.

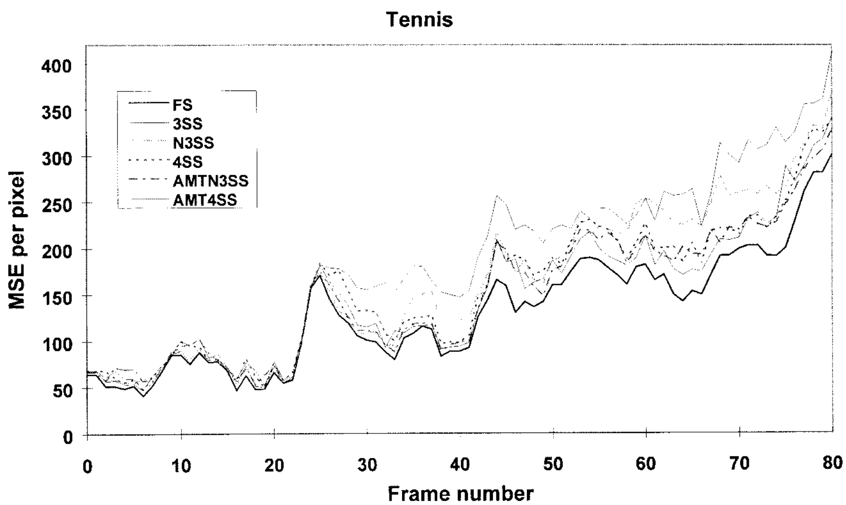
Stage 1) Determination of the Initial Search Center: The motion arising in a scene often occupies some block-based segments. Thus, to estimate the motion trend of the current block, the motion correlation between the current block and its four neighbor blocks is first determined. Let $\vec{V}_{\rm Bi}$ with $i = 0, 1, 2, 3, 4$, be the motion vectors corresponding to the blocks as shown in Fig. 2. Let $\vec{V}_{\rm init}$ be the initial search window's center from the origin of the current block. We define the mean motion vector of the four neighbor blocks as

$$\vec{V}_m = \frac{1}{4}\sum_{i=1}^{4} \vec{V}_{\rm Bi}. \tag{2}$$

When all four neighbor motion vectors are very close to $\vec{V}_m$, it seems that these blocks' motions are very similar. In this case, we assume that these blocks are within the same moving



(a)



(b)

Fig. 4. MSE comparisons of FS, 3SS, N3SS, 4SS, AMTN3SS, and AMT4SS for the (a) *Football* and (b) *Tennis* sequences.

object or in the background region; then the mean vector $\vec{V}_m$ is a good motion vector prediction. Therefore, the four neighbor motion vectors can be used to predict an initial search center $\vec{V}_p$. Otherwise, there should be no correlation, and then the origin is used as $\vec{V}_{\text{init}}$. This process can be formulated as

$$\vec{V}_{\text{init}} = \begin{cases} \vec{V}_p, & \text{if } \max_{i=1,\ldots,4} \|\vec{V}_{\text{Bi}} - \vec{V}_m\| < T \\ (0, 0), & \text{otherwise} \end{cases} \tag{3}$$

where T is a predefined displacement threshold.

Based on the observation of the motion vector distribution characteristic [7], [10], we proposed three $\vec{V}_p$ motion prediction methods as follows:

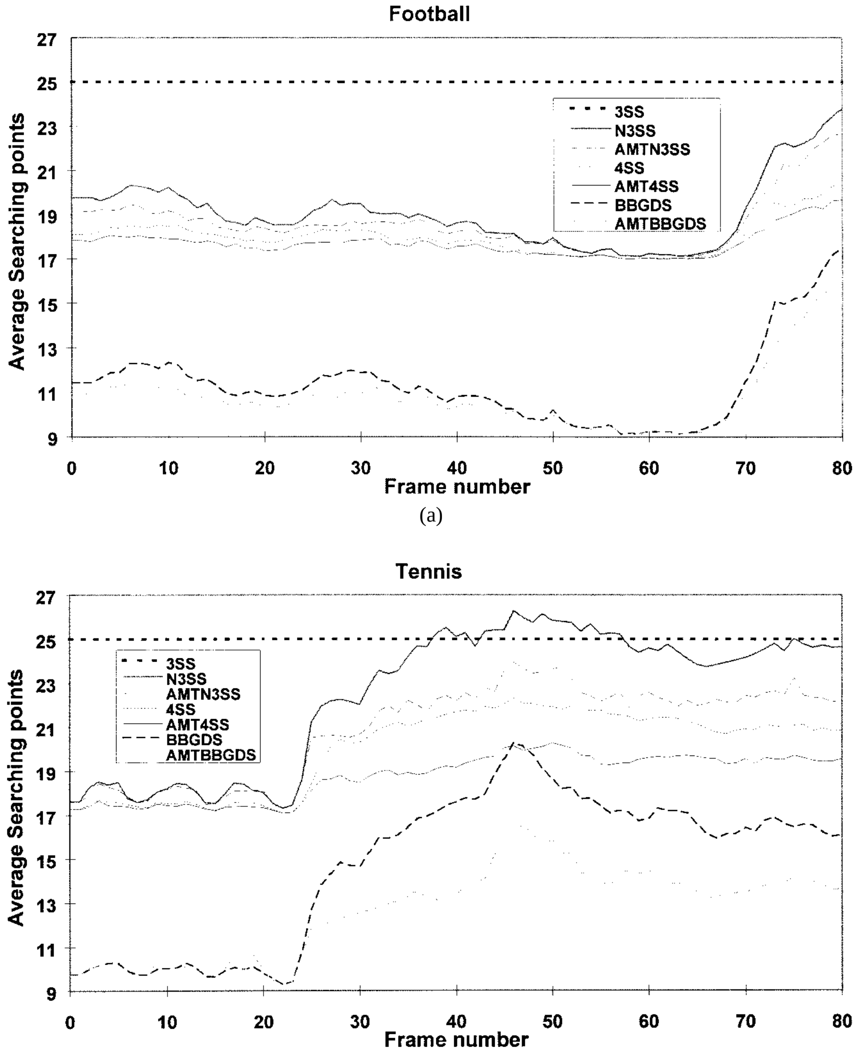
1) *center-biased prediction:* $\vec{V}_p = \arg\min_{\vec{V}_{Bi}} \|\vec{V}_{Bi}\|, \ i = 1, 2, 3, 4$;
2) *mean prediction:* $\vec{V}_p = round(\vec{V}_m)$;
3) *mean-biased prediction:* $\vec{V}_p = \arg\min_{\vec{V}_{Bi}} \|\vec{V}_{Bi} - \vec{V}_m\|, \ i = 1, 2, 3, 4$;

where $round(.)$ is the rounding of all elements of the vector.

TABLE I
AVERAGE MSE OF THE FIRST 80 FRAMES AND AVERAGE SEARCH POINTS PER MOTION VECTOR ESTIMATION FOR THE FIRST 80 FRAMES

Searching Algorithm	Football		Tennis	
	MSE	Search points	MSE	Search points
FS	156.98	225	132.12	225
3SS	176.84	25	186.21	25
N3SS	166.04	19.05	164.14	22.55
4SS	170.28	18.01	156.68	20.09
BBGDS	173.51	11.33	165.30	14.80
AMTN3SS	163.35	18.60	150.79	20.94
AMT4SS	167.19	17.72	149.97	18.75
AMTBBGDS	169.26	10.81	168.71	12.71

The center-biased prediction is accorded with the scene in which the block motion field of a real-world image sequence is usually gentle, smooth, and slowly varying. However, this method cannot track some fast movements. The mean prediction gives an accurate estimation while the assumption that those blocks within the same moving object is true. From the experiments, however, we have found that sometimes the



(a)

(b)

Fig. 5. Comparison of 3SS, N3SS, 4SS, BBGDS, AMTN3SS, AMT4SS, and AMTBBGDS on average search points per motion vector estimation versus frame number for the (a) *Football* and (b) *Tennis* sequences.

four neighbor blocks cover too large an area to track the small motion. Thus, the mean prediction may lead to larger prediction error when it fails to track the real motion. On the other hand, the mean-biased prediction selects a minimum displacement from the mean motion vector $\vec{V}_m$, which represents the object's movement. If the above assumption of those blocks within the same moving object is right, the predicted start search point is close to the real motion location. Otherwise, these blocks probably belong to different motion segments. Then the selection of a minimum displacement can preserve the center-biased distribution property of the motion field. The mean-biased method can keep a better balance of both of the cases. Experimental results also show that the mean-biased prediction provides the best results; thus, we only use this method in this paper. In addition, we do not make any prediction and use the origin as the initial search center for the first row, the first column, and the last column of each frame.

Stage 2) Refinement of the Motion Vector: After the stage one, if there is some interblock motion correction, the motion vector should be very close to the initial search window's center $\vec{V}_{\text{init}}$. Thus, center-biased fast BMA's such as N3SS, 4SS, and BBGDS are chosen to refine the final motion vector. These three algorithms are using center-biased checking points patterns in the first step, which increase the chance for finding a global minimum within the central 5×5 area.

IV. Simulation Results

In this section, we present some simulation results using the luminance component of the first 80 frames of the *Football* and *Tennis* test sequences. The *Football* sequence consists of complex motions that range from slow motion to a very fast motion. In the *Tennis* sequence, camera zooming and panning are also involved. The size of each individual frame is 360×240 pixels quantized uniformly to 8 bits. The mean absolute error (MAE) distortion function is used as the block distortion measure (BDM). The new adaptive motion tracking (AMT) search algorithms using N3SS, 4SS, and BBGDS as the second stage are named AMTN3SS, AMT4SS, and AMTBBGDS, respectively. In the second stage, the maximum displacement in the search region is ± 7 pixels in both the horizontal and the vertical directions for 16×16 block size. In our simulations, the estimated frame was formed from the original frame, and the predefined displacement threshold T of (3) is equal to five. In addition, the mean-biased prediction is used as the initial search center for all the simulations.

The statistical performance comparisons of FS, 3SS, N3SS, 4SS, BBGDS, AMTN3SS, AMT4SS, and AMTBBGDS in terms of mean-square error (MSE) between the estimated frames and the original frames are given in Table I. The MSE comparisons show that the AMTN3SS and AMT4SS achieved better performance than the original algorithms of N3SS and 4SS, respectively. The average MSE's of the first 80 frames of the two test sequences using different BMA's are shown in Fig. 4(a) and (b). (For clarity, only the results for FS, 3SS, N3SS, 4SS, ATMN3SS, and ATM4SS are shown.) The AMT improvement can be easily observed from these figures, especially in the area where fast motion is involved. On the other hand, Table I also shows that the AMTBBGDS performs better than the BBGDS for the *Football* sequence, while it is slightly degraded for the *Tennis* sequence. This small degradation is mainly due to the prediction error propagation, which is slightly more difficult to recover using BBGDS as the second stage. That is because the BBGDS is specially designed for low-bit-rate video coding applications with very center-biased motion fields. The algorithm is therefore relatively more sensitive to local minima around the initial search center. When large motion estimation error occurs in a block, it will more easily propagate to the motion estimation of the neighbor's blocks using the predicted initial center. However, this had only occurred in the *Tennis* sequence, which contains very complex motions. For most of the sequences, the ATMBBGDS always achieves performance improvement with very low average search points.

The speedup ratio of the BMA's is compared by the average search points for a motion vector estimation. The average search points per motion vector estimation for the first 80 frames are also shown in Table I. In addition, the average search points required versus the frame number for the two test sequences are shown in Fig. 5(a) and (b). These figures show that the average search points required by AMTN3SS, AMT4SS, and AMTBBGD for each frame are almost always less than N3SS, 4SS, and BBGDS, respectively. From Fig. 5, we can find that the search points needed by AMT4SS are very near to its minimum of 17 but the search points needed by N3SS and 4SS are varied with the motion content of the image sequences. This shows that the AMT search algorithm can track an optimal motion vector whether the image sequence contains fast or slow motion. In addition, the mean-biased prediction motion tracking algorithm requires only 18 additions, eight multiplications, and seven comparisons in the whole initial search center determination. This is much less than one MAE distortion computation, which requires 511 additions, 256 absolute operations, and one comparison for a 16×16 block. Thus, the proposed algorithm occupies very little computation time.

Our experiments also show that the predefined displacement threshold T is not very sensitive to the performance. When the T range from $\sqrt{10}$ to $\sqrt{40}$, the average MSE of the first 80 frames using AMT4SS varies from 149.97 to 151.26 for the *Tennis* sequence and from 168.15 to 167.04 for the *Football* sequence. The search point average varies from 18.70 to 18.90 and from 17.71 to 17.73 for the *Tennis* sequence and *Football* sequence, respectively. The variances are all less than 1%. Thus, the AMT search algorithm is very robust.

V. Conclusions

Based on the spatial interblock motion fields correlation, new adaptive motion tracking search algorithms are proposed in this paper. These algorithms exploit interblock correlation to predict the initial search center and use center-biased block matching algorithms to refine the final motion vector. Experimental results show that the mean-biased prediction AMT search algorithms combined with N3SS and 4SS effectively improved their performance in terms of mean-square error measure with lower average searching points. In addition, the determination of the initial search center using mean-biased prediction is very low; thus the overall computation requirement is always reduced. It can be expected to apply the algorithm to other BMA's as the first stage to improve the estimation accuracy of the motion vector and enlarge search area in initial motion estimation.

References

[1] "Video codec for audiovisual services at p $\times$ 64 kbit/s," ITU-T Rec. H.261, 1993.

[2] ISO/IEC 11172-2 (MPEG-1 Video), "Information technology—Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s: Video," 1993.

[3] ISO/IEC 13818-2, ITU-T H.262 (MPEG-2 Video), "Information technology—Generic coding of moving pictures and associated audio information: Video," 1995.

[4] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," in *Proc. NTC81*, New Orleans, LA, Nov. 1981, pp. G5.3.1–G5.3.5.

[5] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. COM-29, pp. 1799–1808, Dec. 1981.

[6] 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.

[7] 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.

[8] M.-J. Chen, L.-Gee Chen, and T.-D. Chiueh, "One-dimensional full search motion estimation algorithm for video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp. 504–509, Oct. 1994.

[9] L. W. Lee, J. F. Wang, J. Y. Lee, and J. D. Shie, "Dynamic search-window adjustment and interlaced search for block-matching algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, pp. 85–87, Feb. 1993.

[10] 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.

[11] 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. 419–422, Aug. 1996.

[12] C. Hsieh, P. C. Lu, J. S. Shyn, and E. H. Lu, "Motion estimation algorithm using interblock correlation," *Electron. Lett.*, vol. 26, pp. 276–277, Mar. 1990.

[13] J. Lu and M. L. Liou, "A simple and efficient search algorithm for block-matching motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 429–432, Apr. 1997.