Video Image Block-matching Motion Estimation Algorithm Based on Two-step Search

Wei-qi JIN(金伟其)[1], Yan CHEN(陈 艳)[1], Ling-xue WANG(王岭雪)[1], Bin LIU(刘 斌)[1], Chong-liang LIU(刘崇亮)[1], Ya-zhong SHEN(沈亚中)[2], Gui-qing ZHANG(张桂清)[2]

(1. *Key Laboratory of Photoelectronic Imaging Technology and System, Ministry of Education of China, School of Optoelectronics, Beijing Institute of Technology, Beijing* 100081, *China*; 2. *Beijing Microchemical Research Institute, Beijing* 100081, *China*)

Abstract – Aiming at the shortcoming that certain existing blocking-matching algorithms, such as full search, three-step search, and diamond search algorithms, usually can not keep a good balance between high accuracy and low computational complexity, a block-matching motion estimation algorithm based on two-step search is proposed in this paper. According to the fact that the gray values of adjacent pixels will not vary fast, the algorithm employs an interlaced search pattern in the search window to estimate the motion vector of the object-block. Simulation and actual experiments demonstrate that the proposed algorithm greatly outperforms the well-known three-step search and diamond search algorithms, no matter the motion vector is large or small. Compared with the full search algorithm, the proposed one achieves similar performance but requires much less computation, therefore, the algorithm is well qualified for real-time video image processing.

Key words* – *block-matching*; *motion estimation*; *two-step search

Manuscript Number: 1674-8042(2010)03-0228-05

dio: 10.3969/j.issn.1674-8042.2010.03.06

1 Introduction

Motion estimation is the process of overlaying two or more images of the same scene taken at different times, from different viewpoints, and/or by different sensors, and it is the key step for video encoding, color fusion, and super-resolution processing[1]. Among various motion estimation algorithms, the block-matching algorithm is widely used due to its simple and easy implementation. By exhaustively testing all the candidate blocks within the search window, Full Search (FS) algorithm gives the global Minimum Block Distortion (MBD) position which corresponds to the best matching block. However, a substantial computational load is demanded. To overcome this drawback, many fast block-matching algorithms have been developed, such as Three-Step Search (TSS), Four-Step Search (4SS), Diamond Search (DS) and so on. These fast algorithms greatly improve the search speed, but they may fall into a local optimum or mislead the search path to a wrong direction, and hence miss the optimum point[2-6]. Therefore, the block-matching motion estimation algorithm which simultaneously meets the high accuracy and low computational complexity is the current research focus. According to the analysis of the above-mentioned FS and fast block-matching algorithms, a block-matching motion estimation one based on Two-Step Search (2SS) is proposed in this paper.

2 The block-matching motion estimation

2.1 The principle of block-matching motion estimation

The principle of block-matching motion estimation is shown in Fig. 1. On selecting an interesting object-block in the current image, and assuming that each pixel in the object-block has the same rigid motion, the MBD point according to certain matching criterion in search window of the reference image is the global optimal motion vector[7].

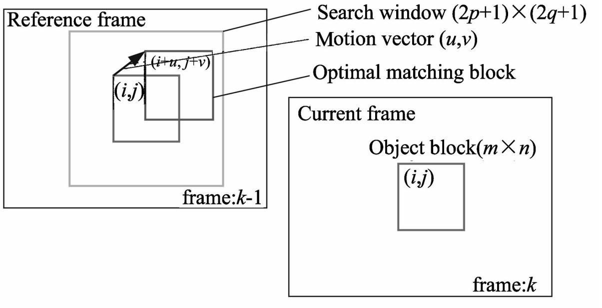

Fig.1 Principle of block-matching motion estimation

※ **Received:** 2010-06-13
Project supported: This work was supported by the Lab Open Fund of Beijing Microchemical Research Institute(P2008026EB)
Corresponding author: Wei-qi JIN(jinwq@bit.edu.cn)

2.2 Block-matching criterion

On one hand, block-matching criterion is the basis of the block similarity measurement, so it has a direct impact on the accuracy of the motion estimation. On the other hand, the block-matching criterion also affects the computational complexity of the matching algorithm to some degree. Thus, reducing the computational complexity of the block-matching criterion is an effective way to improve the speed of the motion estimation. There are three matching criterions commonly used in the motion estimation: Mean Absolute Difference (MAD), Mean Square Error (MSE), and Normalized Cross Correlation Function (NCCF)[8]. Their definitions are as follows:

1) MAD

$$\mathrm{MAD}(u,v)=\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\cdot \left| f_k(i,j)-f_{k-1}(i+u,j+v)\right|, \quad (1)$$

where (u,v) is the motion vector, f_k and f_{k-1} are the gray value of the current image and the reference image respectively, and $m\times n$ is the size of object-block. If the MAD at point (u_0,v_0) is the global minimum, this point is definitely the optimal matching point.

2) MSE

$$\mathrm{MSE}(u,v)=\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}[f_k(i,j)-f_{k-1}(i+u,j+v)]^2, \quad (2)$$

where the point with minimum MSE is the optimal matching point.

3) NCCF

$$\mathrm{NCCF}(u,v)=\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}f_k(i,j)f_{k-1}(i+u,j+v)}{[\sum_{i=1}^{m}\sum_{j=1}^{n}f_k^2(i,j)]^{2/1}[\sum_{i=1}^{m}\sum_{j=1}^{n}f_{k-1}^2(i+u,j+v)]^{1/2}}, \quad (3)$$

where the point with maximum NCCF is the optimal matching point.

When the motion estimation accuracy is pixel-level, the impact of matching criterion on the motion estimation accuracy is not significant. Compared with MSE and NCCF, there is no multiply operation in MAD, and it is simple and convenient, thus this criterion is most commonly used in the block-matching algorithm[9]. The algorithm proposed in this paper also employs MAD as the matching criterion.

2.3 Search pattern

Search pattern is critical and the most complex part of motion estimation algorithm, and it has significant impact on the accuracy and computational complexity of the algorithm. Hence, it is the main criterion of classifying the motion estimation algorithms. According to the search pattern, block-matching motion estimation algorithms fell into two categories: Full Search (FS) and fast search block-matching. In FS, the MAD of all the points in the search window of the reference image are calculated, and the position offset of point with global minimum MAD is the desired motion vector. Among all the block-matching algorithms, FS algorithm owns the highest accuracy, but it is difficult to achieve real-time processing because of substantial computational load. Thus in the real system, espcially the low-power real-time one, FS algorithm is not practical. It is usually a standard to measure the merit of other motion estimation algorithms[10].

Among the fast search block-matching algorithms, TSS and DS are the most popular. TSS employs a coarse-to-fine search pattern to estimate the motion vector, and the first search step length of TSS equals to or is slightly greater than half the maximum search step length. In each step, 9 points are checked: the center of the square and 8 boundary points of the search area. The step length is reduced by half in each step. Searching should stop when the step length is reduced to 1. Because the large search template is used in the first search step and the search mode is uniformly distributed, TSS is just suitable for estimating the large motion vector. It is quite likely to fall into a local optimum and hence deteriorates the performance especially for the small motion vector[11]. DS algorithm employs the center-biased property to estimate the motion vector, that is, the optimal motion vector which is usually located in a small area around the search center. This algorithm can maintain high search speed and accuracy for small motion vector, however, when the motion vector is large, that is, the global minimum is deviated from the search center while the block-matching error surface has local minimum near the central region. DS algorithm can not find the correct motion vector[12]. Therefore, studying a motion estimation algorithm, which not only can guarantee the search speed but also ensure the global optimum, is the urgent problem in block-matching algorithm.

3 The motion estimation algorithm based on two-step search

The Two-Step Search (2SS) algorithm is an improved fast block-matching motion estimation algorithm based on the idea of TSS and DS. Its search process is shown in Fig. 2. A 13×13 sized search window is taken for example, and the search step of 2SS is as follows:

Step 1: Search the awaiting checking points in the search window by the interlaced mode, and calculate the MAD of each checking point until the end of the search window;

Step 2: Setting the point with minimum MAD in Step1 as the center, check this point and other 8 points which surrounding it with 1 sized step length, and the point with minimum MAD is the desired motion vector.

In most cases, the gray values of adjacent pixels will not vary much, and the MAD of the points around the optimal matching point (the peak) should be next to the peak. Therefore, 2SS algorithm employs the interlaced search pattern to search the optimal matching point. Compared with FS, 2SS algorithm can achieve similar performance but requires much less computation. Assuming the size of search window is $(2p+1)\times(2q+1)$, the number of checking points in FS algorithm is $(2p+1)\times(2q+1)$, while in 2SS, the number is $(p+1)\times(q+1)+8$. When the search window is large, the computational speed of 2SS will have an obvious improvement. Compared with TSS and DS, the advantages of 2SS can be concluded that: ① The initial search length of 2SS is small, avoiding falling into a local optimum; ② The initial search area of 2SS is broad, almost covering the whole search window, and can avoid misleading the search path to a wrong direction. Therefore, 2SS algorithm can deal with both small and large motion vectors with small motion estimation error.

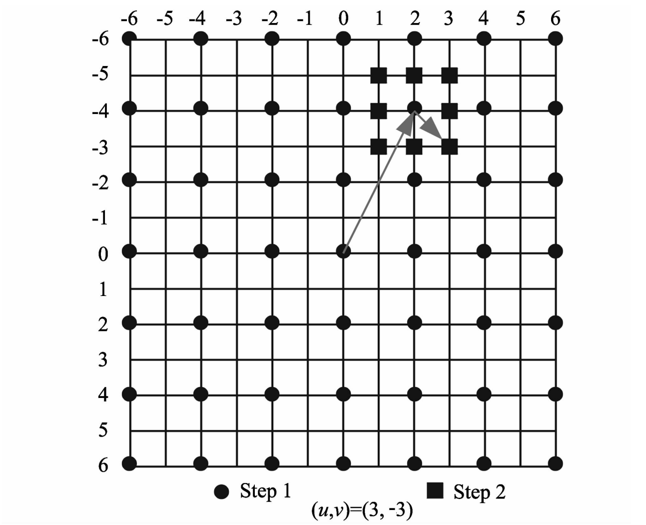

Fig. 2 Search schematic diagram of 2SS algorithm

4 Experiments and results

In order to verify the performance of 2SS algorithm, simulated and actual image sequences are tested.

Fig. 3 Image source of image sequences

4.1 The experiment of simulated image sequences

Taking Fig. 3 as the image source, several groups of image sequence are sampled with given motion vectors. Selecting the interested object-block in the current image, and estimating the motion vector by FS, 2SS, TSS and DS, respectively, the results are shown in Tab. 1. We can see that: ① Although TSS and DS algorithms require fewer search points, their estimation accuracy is limited: the initial search step of TSS is too large, making it easy to fall into a local optimum when the motion vector is small; while DS works from the origin of the search window, when the motion vector is large, it is very possible to miss the optimum point. ② FS algorithm implements global search in the whole search window, and the search result must be global optimum, but when the search window is large, there are too many points required to be checked. thus the search speed will be affected. 2SS employs the interlaced search pattern in the search window to estimate the motion vector. It inherits the high accuracy of FS, but requires much less search points.

Tab. 1 Results of the various motion estimation algorithms with different motion vectors

	Motion estimation algorithms	Estimated motion vector (pixel)	The number of search points
Actual motion vector: (3, 3) Search window: (11×11)	FS	(3,3)	121
	2SS	(3,3)	44
	TSS	(2,3)	17
	DS	(3,3)	13
Actual motion vector: (5, 5) Search window: (13×13)	FS	(5,5)	169
	2SS	(5,5)	57
	TSS	(5,4)	25
	DS	(2,2)	23
Actual mouion vector: (8, 8) Search window: (17×17)	FS	(8,8)	289
	2SS	(8,8)	89
	TSS	(8,8)	33
	DS	(2,1)	13

(Note: The motion vector is aiming to the imaging device. The right is positive in x direction, and the downward is positive in y direction.)

Taking the actual motion vector $(u,v)=(5,5)$ for example, the simulated sequence images are shown in Fig. 4(a) and Fig. 4(b), and the size of the images is 295×395. The interested object-block has been marked with a red rectangle in the current image (Fig. 4(a)). In simulation, since the scene is stationary while the imaging device is moving, there is no local motion. Thus the motion vector of the object-block can represent the movement of the whole image. Estimating the motion vector in the reference image (Fig. 4(b)) using FS, 2SS, TSS and

DS, respectively, the results are shown in Fig. 4(d)～(g). It is obvious that FS and 2SS can get the accurate motion vector, and the estimation result of TSS has slight bias, while the error of DS is obvious.

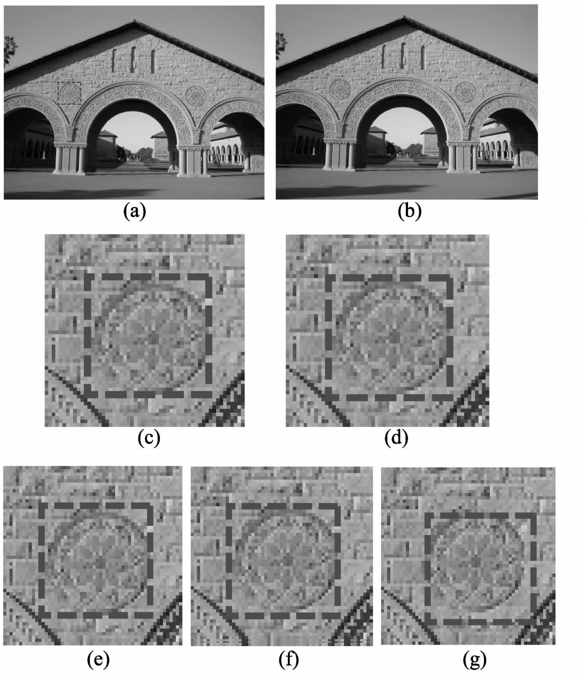

Fig. 4 Search results of FS, 2SS, TSS and DS algorithms with motion vector $(u, v) = (5,5)$: (a) Current image, (b) Reference image, (c) Object-block in current image, (d) FS search result in reference image, (e) 2SS search result in reference image, (f) TSS search result in reference image, (g) DS search result in reference image

4.2 The experiment of real image sequences

Take two actual videos containing moving objects to test the performance of 2SS. It should be explained that the estimating result is the motion vector of the moving object but not the movement of the global scene. Fig. 5 shows the current image (the 2nd frame) and the reference image (the 3rd frame) of Car video, and the size of them is 236×256. To Select the moving license plate in the current image as the object-block, the estimating results of FS, 2SS, TSS, and DS in an 11×11 sized search window are shown in Tab. 2 and Fig. 5. We can see that 2SS and DS have the same result with FS, while the result of TSS is biased. It can be concluded that when the motion vector is small, 2SS is prior to TSS.

Tab. 2 Motion estimation results of FS, 2SS, TSS and DS algorithms in car image sequences

Algorithms	Estimated motion vectors(pixel)	The number of search points
FS	(−1,3)	121
2SS	(−1,3)	44
TSS	(2,2)	17
DS	(−1,3)	21

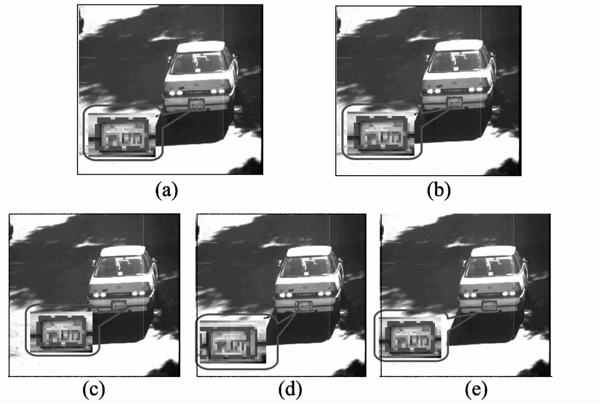

Fig. 5 Search results of FS, 2SS, TSS and DS algorithms in car image sequences: (a) Object-block in current image, (b) FS search result in reference image, (c) 2SS search result in reference image, (d) TSS search result in reference image, (e) DS search result in reference image

In order to verify the performance of 2SS for large motion vector, another experiment is carried out. To select the 5th frame and the 25th frame from People video as the current image and the reference image, face of the person marked in the current image is the object-block. The estimating results of FS, 2SS, TSS, and DS in 71×71 search window are shown in Fig. 6. We can see that 2SS has the same result as FS, and the results are consistent with the visual result, but the number of search points in 2SS is much less than FS(as shown in Tab. 3). The estimating error of DS is obvious. It verifies that 2SS is prior to DS when the motion vector is large.

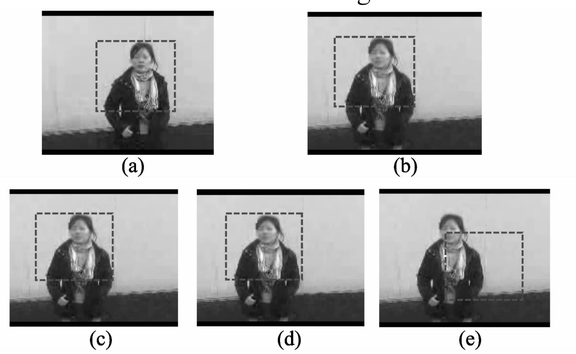

Fig. 6 Search results of FS, 2SS, TSS and DS algorithms in people image sequences: (a) Object-block in current image, (b) FS search result in reference image, (c) 2SS search result in reference image, (d) TSS search result in reference image, (e) DS search result in reference image

Tab. 3 Motion estimation results of FS, 2SS, TSS and DS algorithms in people image sequences

Algorithms	Estimated motion vectors(pixel)	The number of search points
FS	(−27, −5)	3 721
2SS	(−72, −5)	969
TSS	(−25, −6)	46
DS	(23,15)	56

5 Conclusion

Based on analyzing the characteristics of certain existing block-matching algorithms such as FS, TSS, DS and so on, a block-matching motion estimation algorithm based on two-step search is studied in this paper. The 2SS algorithm employs the interlaced search pattern in the search window to estimate the motion vector, and its small search step and wide search range ensures the search accuracy. The simulated and actual experiment results show that 2SS algorithm consistently performs well for the image sequence with wide range of motion vector. It greatly outperforms the well-known TSS and DS algorithm, and can achieve close performance compared with FS algorithm while reducing computation greatly, so 2SS is more suitable for real-time system.

The algorithm in this paper has been successfully utilized in the object-block motion estimation of our video image processing, and has achieved satisfying result. In future research, the algorithm can combine with gradient operation, optical flow estimation or image pyramid layer technology to realize high accuracy sub-pixel motion vector estimation.

References

[1] Zitova Barbara, Flusser Jan, 2003. Image registration methods: a survey. *Image vision computing*, 21: 977-1000.

[2] V. Seferidis, M. Ghanbari, 1993. General approach to block-matching motion estimation. *Optical Engineering*, 32 (7): 1464-1474.

[3] B. Furth, J. Greenberg, R. Westwater, 1997. Motion Estimations Algorithms for Video Compression. Norwell, MA: Kluwer.

[4] Zhu Shan, Kai-Kuang Ma, 2000. A new diamond search algorithm for fast block-matching motion estimation. *IEEE Transactions on Image Processing*, (9): 287-290.

[5] Man-po Lai, Wing-chung Ma, 1996. A novel four-step search algorithm for fast block motion estimation. *IEEE Trans*, *Circuit Syst*. *Video Technol*, 6: 313-317.

[6] Li-feng Huo, Zhi-guo GUI, 2007. Research of the MPEG4-based digital video compression standard. *Journal of North University of China*, 28: 131-134.

[7] Chang-zheng Zhu, Zhen-kang Shen, 2004. Improved full-search block-matching algorithm. *Infrared and Laser Engineering*, 33(4): 388-391.

[8] You-jun Xiang, Na Lei, Wei-yu Yu, et al, 2009. Research of block matching criterion for motion estimation. *Computer Science*, 36(9): 278-280.

[9] Zhi-xin Li, Jian-hua Li, Jian-dang Hou, 2007. New improved motion estimation algorithm. *Computer Engineering and Application*, 43(18): 51-52.

[10] Gao-bo Yang, Qing-song Yang, 2010. MATLAB Image/Video Processing and Application. Publishing House of Electronics Industry, Beijing.

[11] Ming Zhang, Du-yan Bi, 2007. Research progress in algorithm for block matching motion estimation. *Digital TV & Digital Video*, 31(3): 8-11.

[12] Zhao-qing Zheng, Hong-shi Sang, Xu-bang Shen, 2006. Research and progress in algorithm for block matching motion estimation. *China Integrated Circuit*, 86(7): 50-56.