모션 속도와 다중 초기 중심점 예측에 기반한 빠른 비디오 모션 추정 알고리즘

(Fast Video Motion Estimation Algorithm Based on Motion Speed and Multiple Initial Center Points Prediction)

팽 소 호 † (Sha-Hu Peng) 유 잠 멜 † (Khairul Muzzammil Saipullah)

윤 병 춘 † (Byung-Choon Yun) 김 덕 환 †† (Deok-Hwan Kim)

요 약 본 논문은 모션 속도와 다수의 초기 중심점에 기반한 빠른 모션 추정 알고리즘을 제안한다. 제안한 방법은 시공간적 이웃 모션 벡터들에 의해 초기 탐색점을 예측한다. 모션벡터를 빠르게 구하기 위하여 모션 속도와 예측된 초기 중심점들에 기반한 동적인 탐색 패턴이 이용된다. 제안한 방법은 시공간 정보와 동적 탐색 패턴을 이용하여 탐색 속도를 매우 빠르게 할 뿐만 아니라 양질의 영상화질을 유지할 수 있다. 실험결과를 통해 제안한 방법이 완전탐색, 새로운 삼단계 탐색, 사단계 탐색 방법들과 비교하여 검색시간을 줄이면서 PSNR 관점에서 양질의 영상화질을 제공함을 알 수 있다.

키워드 : 모션 추정, 모션 속도, 다중 초기 중심점

· 이 논문은 인하대학교 연구비 지원에 의하여 연구되었음

· 이 논문은 2010 한국컴퓨터종합학술대회에서 '모션속도와 다양한 초기의 중앙적 예측에 기반한 빠른 비디오 모션 추정'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 인하대학교 전자공학과
pengshaohu@iesl.inha.ac.kr
mellore@iesl.inha.ac.kr
bchyun@iesl.inha.ac.kr

†† 종신회원 : 인하대학교 전자공학과 교수
deokhwan@inha.ac.kr
(Corresponding author임)

논문접수 : 2010년 8월 9일
심사완료 : 2010년 10월 22일

Copyright©2010 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨팅의 실제 및 레터 제16권 제12호(2010.12)

Abstract This paper proposes a fast motion estimation algorithm based on motion speed and multiple initial center points. The proposed method predicts initial search points by means of the spatio-temporal neighboring motion vectors. A dynamic search pattern based on the motion speed and the predicted initial center points is proposed to quickly obtain the motion vector. Due to the usage of the spatio-temporal information and the dynamic search pattern, the proposed method greatly accelerates the search speed while keeping a good predicted image quality. Experimental results show that the proposed method has a good predicted image quality in terms of PSNR with less searching time comparing with the Full Search, New Three-Step Search, and Four-Step Search.

Key words : Motion estimation, Motion speed, Multiple initial center points

1. Introduction

Due to the ability of removing temporal redundancies between successive image frames, motion estimation and compensation play key roles in video encoding standards, such as MPEG-2 [1], H.264 [2]. The most popular motion estimation technique is the block matching motion algorithm (BMA), which obtains the motion vector (MV) by separating the current frame into non-overlapping square blocks and searching the best matching block within a defined search window in the reference frame for each current block.

Full Search (FS) is the optimal block matching algorithm that can find the MV by exhaustively searching all locations inside a defined search window. It is simple and easy to be implemented in hardware. But on the other hand, the large computation of this algorithm makes it difficult to be applied in real-time video compression. To overcome this problem, many efficient fast algorithms have been proposed in the past decades. Some well-known methods are Three-Step Search (TSS) [3], New Three-Step Search (NTSS) [4], Four-Step Search (FSS) [5], etc. These methods can reduce the computation by only searching some significant points inside the defined search window while keeping a good error performance. However, these methods fail in taking account of the spatio-temporal dependencies that exist between the MVs. Algorithms that use the spatial or temporal information have been proposed in these years [6-9].

Recently, H. Nisar et al. [10] proposed a fast motion estimation algorithm based on multiple initial point prediction. The algorithm predicts initial points and constructs a dynamic search pattern for each block by means of the spatio-temporal neighboring MVs.

This paper proposes a new fast motion estimation algorithm based on motion speed and multiple initial center point prediction. The proposed method takes advantage of the spatio-temporal neighboring MVs to determine the initial points similar to [10]. However, unlike [10], a more reasonable strategy is proposed to get the initial center points. Moreover, a motion speed is estimated by means of the spatio-temporal MVs and thereafter a simple dynamic search pattern is defined to search the MV based on the motion speed and the initial center points. Due to the new strategy of initial center point estimation and the usage of the motion speed, the proposed method achieves fast search speed and good predicted image quality.

The remainder of this paper is organized as: the new strategy of initial center point prediction is presented in Section 2. Section 3 describes the search pattern based on the motion speed and the initial center points. Experimental results and conclusion are presented in Section 4 and Section 5, respectively.

2. Initial center point prediction

To estimate the MV for each block, choosing proper initial center points to start the search is critical to fast computation and optimal matching. Since the spatio-temporal neighboring blocks usually have similar MVs as the current block does, the spatio-temporal neighboring MVs can be used to predict the initial center points. There are three spatial neighboring MVs (*S1*, *S2*, *S3*) and nine temporal neighboring MVs (*T0*, *T1*,..., *T8*) for initial point prediction. In this paper, the spatial left block and the spatial top block are used together since these blocks are often associated with the same objects and have similar MVs as that of the current block. The spatial left block is not always correlated with the current block and is often unavailable for left margin blocks. So we have chosen spatial top block as H. Nisar et al. [10] did to compensate when the spatial left block is not available. In the temporal domain, the MVs (*T0*, *T1*, ..., *T8*) are less stable than those in the spatial domain since these temporal MVs are also affected by the frame rate. Therefore, we only choose one temporal MV (*T0*) for prediction. Both the spatial and temporal MVs are shown in Fig. 1.

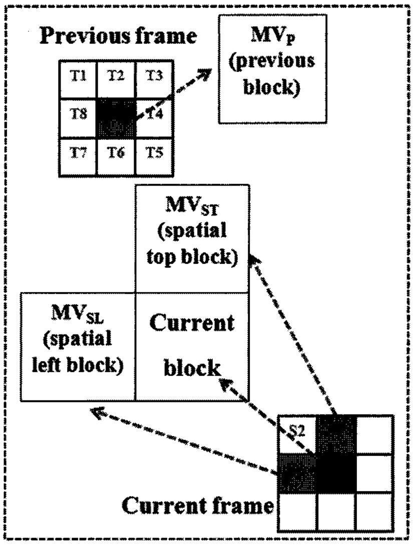


Fig. 1 The spatial and temporal MVs

With the properties of the MV_P, MV_{ST} and MV_{SL}, this paper proposes to take advantage of these MVs and use a new strategy to predict the initial center points. Different from H. Nisar's algorithm, three cases are determined to generate the initial center points by means of the directions of the MV_P, MV_{ST} and MV_{SL}. Furthermore, instead of separating the Cartesian coordinate into four quadrants to determine whether combining two motion vectors, this paper proposes to consider the angles between the motion directions of the motion vectors. The new strategy is described as follows:

Case 1: if MV_P, MV_{ST} and MV_{SL} are all equal to 0, then the current block is predicted into a stationary block.

Case 2: if the angle between MV_{ST} and MV_{SL} is less that a threshold T_A, then MV_{ST} and MV_{SL} are combined into one MV, which can be denoted as follows:

$$MV_{STL} = \frac{1}{2}(MV_{ST} + MV_{SL}) \quad (1)$$

1) If the angle between MV_{STL} and MV_P is less than T_A, one initial center point is generated, as shows in Equation (2). This situation can be shown by Fig. 2(a).

$$MV = \alpha * MV_{STL} + \beta * MV_P \quad (2)$$

2) If the angle between MV_{STL} and MV_P is greater than T_A, two initial center points (P_1 and P_2) are generated, where P_1 = MV_{STL}, P_2=MV_P. It can be shown by Fig. 2(b).

Finally, a motion speed is estimated for "case 2" by: MS = $||MV_{STL} - MV_P||$

Case 3: if the angle between MV_{ST} and MV_{SL} is greater that T_A, the angle (A_{STP}) between MV_P and MV_{ST}, and the angle (A_{SLP}) between MV_P and MV_{SL} are calculated.

1) If ($A_{STP} < A_{SLP}$) and ($A_{STP} < T_A$), MV_P and MV_{ST} are combined as follows:

$$MV_{STP} = \alpha * MV_{ST} + \beta * MV_P \quad (3)$$

Two initial center points are generated, which are: P_1 = MV_{STP}, P_2=MV_{SL}. It can be shown in Fig. 3(a). The motion speed is estimated as: MS = $||MV_{ST} - MV_P||$.

2) If ($A_{SLP} < A_{STP}$) and ($A_{SLP} < T_A$), MV_P and MV_{SL} are combined as follows:

$$MV_{SLP} = \alpha * MV_{SL} + \beta * MV_P \quad (4)$$

Two initial center points are generated, which are: P_1= MV_{SLP}, P_2=MV_{ST}. It can be shown in Fig. 3(b). The motion speed is estimated as: MS = $||MV_{SL} - MV_P||$.

3) If both A_{STP} and A_{SLP} are greater than T_A, three initial center points are generated, which are P_1 = MV_{SL}, P_2=MV_{ST}, and P_3=MV_P. As shown in Fig. 3(c).

3. The search pattern

After the initial center points are predicted, a dynamic search pattern that based on the motion speed and the predicted points is proposed to obtain the motion vector. The sum of absolute difference (SAD) is used as block distortion measure. As mentioned earlier, there are three cases of the initial center points. Therefore, the search pattern makes use of these initial center points and the estimated motion speed to obtain the MV quickly. It is described as follows.

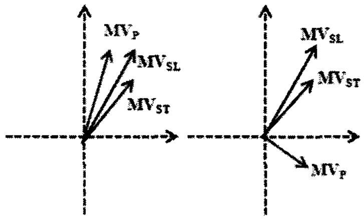


Fig. 2 The motion directions of case 2

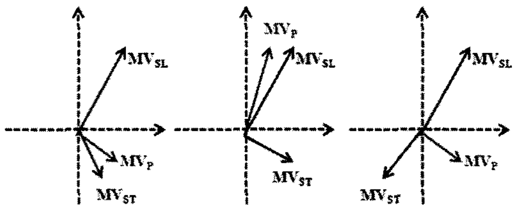
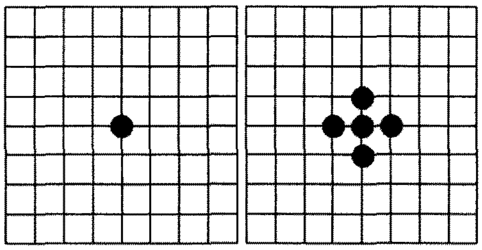


Fig. 3 The motion directions of case 3

3.1 Search pattern for stationary block

1. If SAD is less than a threshold T_{SAD}, the initial center point is predicted to be the MV. As shown in Fig. 4(a).
2. If SAD is greater than T_{SAD}, then a small rood pattern is applied. The point that with minimum SAD is determined as the MV. As shown in Fig. 4(b).



(a) (b)

Fig. 4 Search pattern for stationary block

3.2 Search pattern for one initial center point

1. If MS is less than a threshold T_{MS}, a small rood search pattern is applied and the point with the minimum SAD is determined as MV. As shown in Fig. 5(a)
2. If MS is greater than T_{MS}, a large rood search

pattern is firstly applied to find the minimum SAD point. Then it follows a small rood search pattern and the point with minimum SAD is determined as the MV. As shown in Fig. 5(b).

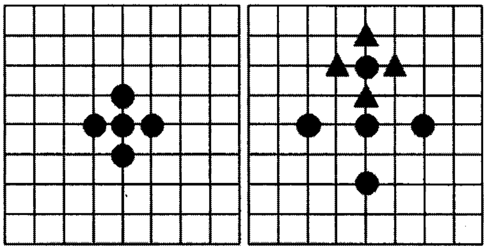


Fig. 5 The search pattern for one initial center point

3.3 Search pattern for two initial center points

In this case, the SADs of the two initial center points are calculated and the one with minimum SAD is selected as the new search center point. Finally, the search pattern in section 3.2 is applied to obtain the MV.

3.4 Search pattern for three initial center points

Similar to section 3.3, the SADs of the three initial center points are calculated and the one with minimum SAD is selected as the new search center point. Then a large rood search pattern is applied to find out the minimum SAD point. Finally, a small rood search pattern is used to obtain the MV. As shown in Fig. 5(b).

4. Experimental results

To test the performance of the proposed method, the PSNR and the search time of the proposed algorithm were calculated to compare with those of the Full Search, New Three-Step Search, Four-Step Search and H.Nisar's (the one proposed by H. Nisar [10]). Four video sequences were used for the experiments. They were "Suzie (150 frames)", "Miss_am (150 frames)", "Foreman (400 frames)" and "Carphone (382 frames)", which were obtained from [11]. The size of each frame of these videos was 176-by-144 pixels and the block size was set to 15-by-15. In the following experiments, α and β were set to 0.65 and 0.35, respectively. T_A, T_{SAD} and T_{MS} were set to 45 degrees, 512 and 1.5 respectively. Table 1 shows the experimental environment.

Table 1 the experimental environment

CPU:	Dual core(TM) 2.4 GHz
Operation System:	Windows Xp
Memory:	2 GB
Programming Tool:	VC++ 2005 and OpenCV

Table 2 shows the average PSNR (peak signal-to-noise ratio) of the video of "Suzie", "Miss_am", "Foreman" and "Carphone". Table 3 shows the average time consumption of these video sequences. As we can see from both tables; the Full Search achieved optimal PSNR whereas took much more time than the other algorithms. FSS and NTSS achieved similar PSNR and similar time consumption. H. Nisar's algorithm and the proposed method achieved the fastest searching speed while keeping good performance comparing with others. In the case of "Miss_am" sequence, since most content of this video sequence is static and the speed of those motion blocks is slow, the proposed method achieved high PSNR and fast search speed owning to its simple search pattern and new prediction strategy. For the case of complicated background, fast motion sequence (the "Carphone" video), the PSNR of the method is degraded. Due to the fast motion and complicated background, it is difficult to predict the initial point and obtain a precise MV. As we can observe from Table.2, the performance values of all the methods are degraded in this case. In these experiments, there were totally 320 blocks for testing, with 183 block corresponding to Case 1, 122 blocks corresponding to Case 2 and 15 blocks corresponding to Case 3. Most of the blocks were corresponded to Case 1 and Case 2. Since the proposed method used a simple search pattern and predicted the initial points based on the spatio-temporal MVs, it could achieve high performance in Case 1 and Case 2 and thus could obtain good searching speed and maintain good PSNR. Sum up from table 2 and 3, the average PSNR of the four sequences of the proposed method is about 0.6dB higher than that of H. Nisar's algorithm. Moreover, the proposed method reduces about 5% of average searching time comparing with H. Nisar's algorithm for the four video sequences.

Table 2 the average PSNRs of the video sequences

	FS	FSS	NTSS	H. Nisar's	Proposed
Foreman	31.75	31.16	30.97	29.74	30.97
Carphone	30.42	30.26	30.24	29.51	29.65
Suzie	34.52	34.36	34.20	31.70	32.65
Miss_am	38.33	38.23	39.32	38.17	38.18

Table 3 the average time consumption of the video sequences (ms)

	FS	FSS	NTSS	H. Nisar's	Proposed
Foreman	14.721	1.916	1.969	0.357	0.307
Carphone	14.742	1.833	1.862	0.336	0.338
Suzie	14.722	1.858	1.703	0.279	0.276
Miss_am	14.734	1.711	1.840	0.152	0.148

5. Conclusion

A fast motion estimation algorithm based on motion speed and multiple initial center points is proposed in this paper. The proposed method predicts the initial center points and estimates the motion speed by means of the directions of the spatio-temporal neighboring motion vectors. According to the predicted initial center points and the motion speed, a dynamic search pattern is then proposed to obtain the motion vector quickly. For the slow motion blocks, a small rood search pattern is applied, while for those fast motion blocks, a large rood search pattern following a small rood search pattern is adopted. Due to the new strategy of initial center point prediction and the dynamic search pattern, the proposed method achieves good predicted image quality and speeds up the search process.

References

[1] ISO/IEC JTC// SC29/WG11 moving picture experts group, MPEG2 test model 4, 1993.

[2] T.Wiegand, G.Sullivan, A.Luthra, Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-TRec.H. 264|ISO/IEC 14496-10 AVC), 2003.

[3] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," in *Proc. NTC 81*, pp.C9.6.1-9.6.5, 1981.

[4] R. Li, B. Zeng, M.L. Liou, A new three step search algorithm for block motion estimation, *IEEE Trans. Circuits Syst. Video Technol.*, pp.438-442, 1994.

[5] L.M. Po, W.C. Ma, A novel four-step search algorithm for fast block motion estimation, *IEEE Trans. Circuits Syst. Video Technol.*, pp.313-317, 1996.

[6] D. Turga, T. Chen, Estimation and mode decision for spatially correlated motion sequences, *IEEE Trans. CSVT*, pp.1098-1107, 2001.

[7] K.R. Namuduri, Motion estimation using spatio-temporal contextual information, *IEEE Trans. Circuits Syst. Video Technol.*, pp.1111-1115, 2004.

[8] H. Nisar, T.S. Choi, An adaptive block motion estimation algorithm based on spatio temporal correlation, Digest of Technical Papers, *International Conference on Consumer Electronics*, pp.393-394, January 2006.

[9] H. Nisar, T.S. Choi, Fast motion estimation algorithm based on spatio-temporal correlation and direction of motion vectors, *Electron. Lett.*, pp.1384-1385, 2006.

[10] H. Nisar, T. S. Choi, Multiple initial point prediction based search pattern selection for fast motion estimation, Pattern Recognition, pp.475-486, 2009.

[11] http://www.softage.ru/products/video-codec/demo-page/