Fast Motion Estimation for Frame-Rate Conversion

Justy W.C. Wong*, Oscar C. Au**, Peter H.W. Wong***

Department of Electrical and Electronic Engineering
Hong Kong University of Science and Technology

Clear Water Bay, Hong Kong, China.

Tel: +852 2358-7053 Fax: +852 2358-1485

Email: *eejusty@ee.ust.hk, **eeau@ee.ust.hk, ***eepeter@ee.ust.hk

Abstract

Different multimedia applications and transmission channels require different resolution, frame rates/structures and bitrate, there is often a need to transcode the stored compressed video to suit the needs of these various applications. This paper is concerned about fast motion estimation for frame rate/structure conversion. In this paper, we proposed several novel algorithms that exploit the correlation of the motion vectors in the original video and those in the transcoded video. We achieve a much higher quality than existing fast search algorithms with much lower complexity.

1 Introduction

In many networked multimedia services such as web TV, video-on-demand and video-conferencing, most if not all stored video is only available in compressed form due to the huge storage size of video and limited available bandwidth. As different applications and transmission channels require different resolution, frame rates/structures and bit rates, there is often a need to transcode the stored compressed video to suit the needs of these various applications. This paper is concerned about fast motion estimation for frame rate/structure conversion. It is well known that motion estimation is very computationally expensive. Though traditional fast motion searches exist, it is possible to exploit the correlation of the motion vectors in the original video and those in the transcoded video for considerably better performance. In particular, we will address the situation shown in Figure 1 in which a video in IPPP format is to be converted to the IBP format. In this case, frame k is converted from P-frame to B-frame and frame $k+1$, a P-frame original predicted with respect to frame k , is to be predicted from frame $k-1$.

One often encounters large compressed video files on the Internet or Web TV but hesitates whether to spend a significant amount of time and money to download such large files. A fast playback preview version of the video can give an overview of the video for the user and the file size can be reduced significantly. This preview video should still be compressed with the same standard method as the original compressed video such that the same video decoder can be used to view it. One simple way of fast playback is to discard some frames and then re-compress it again but it may cause the change in the frame type structure. For a MPEG bitstream, it has I-frames (intra-frame), P-frames (predictive-frame) and B-frames (interpolated-frame). As shown in Figure 1, all P- and B-frames are predicted from I- and P-frames (anchor frames). So the simplest way of getting a fast playback MPEG bitstream is to discard the B-frames because they are not referenced by any other frames. The frame type structure is changed from IBBP to IPPP format. For efficient coding, this IPPP format can be further converted to IBP or IBBP format. And again, full recompression of the IBBP format video is a computation intensive process due to the motion estimation. Moreover, when transcode a compressed video, some frames might be skipped if the bitrate is too high or for frame rate conversion. Again, it will affect the frame structure of the original compressed bitstream. This situation is similar to the previous problem and we can combine two frame type conversion cases to estimate this problem.

In this paper, we propose several fast motion estimation algorithms for frame rate conversion. In section 2, we will propose two algorithms for P-to-B frame conversion. In section 3, we will discuss the case of P-to-P frame conversion. Finally, we will show the simulation results.

2 Motion for P-to-B frame Conversion (frame k)

It can be observed in Figure 2 that the backward motion vectors should be highly correlated with the original forward motion vectors. In [2], it was proposed to predict the reverse motion vectors from the forward motion vectors. We will apply the idea here. Consider a macro block $MB_k(x,y)$ at location (x,y) . The backward motion vector of $MB_k(x,y)$ should be similar in magnitude to the forward motion vector of $MB_{k+1}(x,y)$ though with opposite sign. Let $v_{k,B}(x,y)$ be the backward motion vector at location (x,y) found by full search for frame k (a B-frame in the IBP format) and let $v_{k+1,P}(x,y)$ be the original forward motion vector at location (x,y) of frame $k+1$ (a P-frame in the IPPP format). Let $v'_{k,B}(x,y)$ be the predicted version of $v_{k,B}(x,y)$ to be computed. We propose to use the negative value of $v_{k+1,P}(x,y)$ as $v'_{k,B}(x,y)$, i.e.,

$$v'_{k,B}(x,y) = -v_{k+1,P}(x,y) \quad \text{for all } x,y. \quad (1)$$

However, the $v_{k+1,P}(x,y)$ is not available when frame $k+1$ is an I-frame ($k+1=\text{GOP}$). In this situation, we use the negative value of $v_{k,P}(x,y)$ as the $v'_{k,B}(x,y)$ which is a poorer prediction than $-v_{k+1,P}(x,y)$. So the equation (1) becomes

$$v'_{k,B}(x,y) = \begin{cases} -v_{k+1,P}(x,y) & \text{for } k+1 \neq \text{GOP}, \\ -v_{k,P}(x,y) & \text{otherwise.} \end{cases} \quad (2)$$

and we called this algorithm P2B.

P2B can give a reasonably good prediction of $v_{k,B}(x,y)$. However, there are situations such as shown in Figure 3 in which it can be poor. In Figure 3, $v_{k,B}(x,y)$ is not related to $v_{k+1,P}(x,y)$ but is actually more related to the motion vector $v_{k+1,P}(x+16,y)$ of the neighboring macroblock, $MB_{k+1}(x+16,y)$. This suggests that the original motion vectors of the neighboring macroblocks can possibly be good prediction. Here we modify P2B by using the $v'_{k,B}(x,y)$ found by P2B and the $v'_{k,B}$ from the eight neighboring macroblocks, which are $v'_{k,B}(x-16,y-16)$, $v'_{k,B}(x,y-16)$, $v'_{k,B}(x+16,y-16)$, $v'_{k,B}(x-16,y)$, $v'_{k,B}(x+16,y)$, $v'_{k,B}(x-16,y+16)$, $v'_{k,B}(x,y+16)$ and $v'_{k,B}(x+16,y+16)$. We compute the MAD of these nine candidate motion vectors and find the best one. We called this modification the P2B-with-Search (P2BS). In order to get a better prediction, an additional half-pixel search can be performed and we called this P2BS-LS (P2BS with local search).

3 Motion for P-to-P frame Conversion (frame k+1)

Let $v'_{k+1,P}(x,y)$ be the motion vectors of the transcoded P-frame $k+1$ with respect to the frame $k-1$. It can be observed in Figure 4 that $v'_{k+1,P}(x,y)$ is close to $v_{k,P}(x',y') + v_{k+1,P}(x,y)$ for some (x',y') . When tracing the object, the predicted block for frame $k+1$ will usually overlap with four macroblocks in frame k . In this example, as the overlapping region of the predicted block and the macroblock $MB_{k,P}(x,y-16)$ is the largest, the object in $MB_{k+1,P}(x,y)$ is more likely matched to $MB_{k,P}(x,y-16)$ than $MB_{k,P}(x,y)$. Let (x',y') be the position of the macroblock in frame k that gives the largest overlapping region with the predicted block associated with $v_{k+1,P}(x,y)$. In [3], forward dominant vector selection (FDVS) is proposed to use $v'_{k,P+1}(x,y) = v_{k,P}(x',y') + v_{k+1,P}(x,y)$. From our simulation results, FDVS can be poor in terms of PSNR when compared with FS. To improve the performance FDVS, [3] performed a refinement search with search area of ± 2 pixels performed. However, this refinement search increases the complexity sharply.

Here we propose to use the original motion vectors $v_{k,P}$ of the four macroblocks in the frame k which overlaps with the predicted block associated with $v_{k+1,P}(x,y)$. With the four candidate vectors, we compute the MAD and choose the one with minimum MAD. We called this algorithm P2P with

search (P2PS). To get better prediction, an additional half-pixel search can be performed and we called it P2PS-LS.

So far, the discussion is on changing a IPPP structure to an IBP structure. Actually, the proposed algorithms P2B, P2BS, P2BS-LS, P2PS, P2PS-LS can be used for frame-rate reduction also.

4 Simulation Results

We tested our algorithms against full search (FS) and three-step-search (3SS) by converting several MPEG I video sequence ("Football", "Table Tennis", "Miss America" and "Salesman") with SIF resolution (352×240) and a GOP of 30 frames from the IPPP frame structure in each GOP into IBP frame structure. The search window is ± 7 . The performance measure is the peak-signal-to-noise-ratio (PSNR) between the motion compensated frames and the original frames.

Some results of P-to-B conversion are shown in Table 1 & 2 and Figures 1 & 2. From the graphs, we can see that the performance of P2BS-LS is much better than 3SS and can be very close to full search for all test sequences with much fewer average search points.

Some results of P-to-P conversion are shown in Table 3 & 4 and Figure 3 & 4. From the tables, FVDS has higher PSNR than 3SS in "Table Tennis" and "Miss America", but lower in "Football" and "Salesman". The refinement of FVDS (FVDS-R) improves much in PSNR but it needs about many (23) average search points in all cases. On the other hand, the proposed P2PS has similar PSNR as FVDS-R with much lower complexity. The local search of P2PS-LS can significantly improve the PSNR by 0.8dB in the cases of "Table Tennis" and "Salesman", making the final PSNR very close to that of FS-14. Actually, P2PS-LS has higher PSNR than FS-7 while requiring much fewer search point than FS-7 and FS-14. Moreover, P2PS-LS has higher PSNR than FVDS-R by about 0.3 to 0.6 dB and with only 1/3 of the average points of FVDS-R.

5 Conclusions

In this paper, we proposed two algorithms the P2B and P2BS for the estimation of backward motion vectors for the P-to-B frame conversion. We also proposed the P2PS for the estimation of motion vectors for the P-to-P frame conversion. The performance of the proposed algorithms is much better than the 3-Step-Search with much lower computation load. The proposed algorithms can achieve various quality and complexity tradeoff. In particular, the P2BS-LS and P2PS-LS have close to optimal performance with very small computation requirement. Moreover, the proposed algorithms can be combined and used for frame-rate reduction also.

6 Acknowledgement

This is funded in part by the RGC grant HKUST6009/97E and by an HKTIIT grant.

7 References

- [1] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion-Compensated Interframe Coding for Video Conferencing," *Proc. of Nat. Telecommun. Conf.*, pp. G5.3.1-5.3.5, Nov./Dec. 1981.
- [2] S. J. Wee, "Reversing Motion Vector Fields", *Proc. of 1998 IEEE Int. Conf. Image Processing*, Chicago, USA, Oct. 1998.
- [3] J. Youn, M. T. Sun, C. W. Lin, "Motion Vector Refinement for High-Performance Transcoding", *IEEE Transactions on Multimedia*, pp. 30-40, vol. 1, no. 1, March 1999.

Acknowledgement

This paper is funded partially by an RGC grant and an HKTIIT grant.

	Football		Table Tennis	
	PSNR (dB)	Avg. Search pt	PSNR (dB)	Avg. Search pt
3 SS	24.39	25	25.60	25
3 SS (half pel)	25.15	25+8	26.82	25+8
P2B	23.96	–	27.43	–
P2BS	24.92	5.44	28.93	3.40
P2BS-LS	25.20	12.86	29.13	9.28
FS-7	25.50	225+8	29.33	225+8

Table 1. Average PSNR (in dB) of the predicted frame using different algorithms for the backward prediction

	Salesman		Miss America	
	PSNR (dB)	Avg. Search pt	PSNR (dB)	Avg. Search pt
3 SS	35.39	25	38.54	25
3 SS (half pel)	36.00	25+8	39.91	25+8
P2B	35.57	–	39.03	–
P2BS	36.00	2.55	39.56	2.34
P2BS-LS	36.12	8.93	40.31	9.76
FS-7	36.17	225+8	40.34	225+8

Table 2. Average PSNR (in dB) of the predicted frame using different algorithms for the backward prediction

	Football		Table Tennis	
	PSNR (dB)	Avg. Search pt	PSNR (dB)	Avg. Search pt
3 SS-14	23.02	33	23.53	33
3 SS (half pel)	23.61	33+8	24.31	33+8
FVDS	22.45	–	25.59	–
FVDS-R	23.06	23.35	25.96	23.02
P2PS	23.12	2.14	25.95	1.35
P2PS-LS	23.55	8.43	26.71	7.54
FS-7	23.23	225+8	26.61	225+8
FS-14	24.37	841+8	27.07	841+8

Table 3. Average PSNR (in dB) of the predicted frame using different algorithms for the P frames

	Salesman		Miss America	
	PSNR (dB)	Avg. Search pt	PSNR (dB)	Avg. Search pt
3 SS-14	36.75	33	37.76	33
3 SS (half pel)	37.56	33+8	38.94	33+8
FVDS	35.88	–	37.95	–
FVDS-R	37.16	23.04	38.66	23.09
P2PS	36.68	0.65	38.18	0.81
P2PS-LS	37.47	8.06	39.25	8.24
FS-7	37.80	225+8	39.55	225+8
FS-14	37.85	841+8	39.58	841+8

Table 4. Average PSNR (in dB) of the predicted frame using different algorithms for the P frames

Figure 1 Missing motion information due to a change in frame structure

Figure 2. The motion estimation of frame k & $k+1$ with: (a) P to P-frame forward prediction (b) B to P-frame backward prediction

Figure 3. The motion estimation of frame k & $k+1$ with: (a) P-to-P frame forward prediction (b) B-to-P frame backward prediction

Figure 4 The forward motion estimation of frame $k-1$, k & $k+1$ with: (a) original P-frame forward prediction (b) re-encoded P-frame forward prediction

Figure 2. (a) PSNR of predicted B-frames using different algorithms; (b) PSNR of full search minus that of P2BS-LS and 3SS for "Table Tennis"

Figure 3. (a) PSNR of predicted B-frames using different algorithms; (b) PSNR of full search minus that of P2BS-LS and 3SS for "Miss America"

(a)

(b)

Figure 4. (a) PSNR of predicted P-frames using different algorithms; (b) PSNR of FS-14 minus that of P2PS-LS and 3SS for "Table Tennis"

(a)

(b)

Figure 5. PSNR of predicted P-frames (a) using different algorithms; (b) PSNR difference of FS-14 minus P2PS-LS and 3SS for the sequence "Miss America"