Comparison between Block-based and Pixel-based Temporal Interpolation for Video Coding

Chi Wah Tang* and Oscar C. Au**

Department of Electrical and Electronic Engineering
The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong

Email: Tommy.CW.Tang@hkt.cwplc.com*, eeau@ee.ust.hk**

Abstract

In very low bit rate (VLBR) video conferencing and videotelephony applications, the bit rate of sending video sequences must be kept low due to the limited channel bandwidth. To satisfy the VLBR requirement, temporal subsampling is a simple technique which may be combined with other compression schemes to achieve very large compression ratio. A previous proposed block-based motion compensated temporal interpolation (MCTI) [1-3] can interpolate frames with reasonably good visual quality but also with some blocking artifacts. In this paper, the block based MCTI is compared with a pixel-based approach. While the pixel-based approach has good performance in some test cases, it is highly computationally intensive making it rather impractical. It is also found to perform poorly in some sequences.

1. Introduction

In video conferencing and videotelephony applications, the bit rate of sending video sequences must be kept low due to the limited channel bandwidth. To satisfy the very-low-bit-rate (VLBR) requirement of the Public Switch Telephone Network (PSTN), temporal sub-sampling is a simple technique, which may be combined with other compression schemes to achieve very large compression ratio. A previous proposed motion compensated temporal interpolation (MCTI) [1-3], a block-based approach, can interpolate frames with reasonably good visual quality but also with some blocking artifacts.

In block-based motion estimation, all the pixels within the same block are assumed to belong to the same object, which undergoes translational motion. It is also assumed that no camera zooming occurs. However, in reality, there may be more than one object within a block and the objects may move in different directions. In that case, one motion vector per block will not be sufficient to describe the multiple object motions. This results in blocking artifacts in MCTI.

However, in a pixel-based motion model, each pixel has its own motion vector. This can describe multiple object motions within any block. With no block structures in the pixel-based optic flow model, there should be no blocking artifacts.

Fig. 1 Motion field in block-(left) and pixel-based (right) model

Figure 1 shows the pixel-by-pixel motion field of the block-based and pixel-based models for a block containing two objects. In the block-based model on the left, all the pixels within the same block are described by the same motion vector although the two objects are moving in different directions. In the pixel-based model on the right, each pixel has its own motion vector. Therefore, multiple objects can be described by the pixel-based model accurately. Actually, deformation can also be adequately described by the pixel-based model because the pixel motion within an object can be different. It appears that pixel-based motion model may be vastly superior to the block-based motion model.

In this paper, we perform a comparative study between the block-based and pixel-based temporal interpolation. We will compare the visual quality and PSNR of the interpolated frames using both approaches together with their computation requirements.

2. Temporal Interpolation Algorithms

2.1 Motion Compensated Temporal Interpolation (MCTI) [1-3]

For any k , the goal is to generate a frame to be inserted between the $(k-1)^{th}$ and k^{th} received frames so that object motions would appear to be smoother. In MCTI, the $(k-1)^{th}$ frame, the k^{th} frame and the inserted frame are divided into blocks of size $N \times N$. Forward and backward block-based motion estimation are performed to find the motion vector. Then the appropriate motion vectors of each block are added to a candidate motion vector lists of the blocks in the inserted frame. For each block, the candidate motion vector with the largest overlapping area is chosen and the inserted frame is then generated by motion compensation.

2.2 Temporal Interpolation using Optic Flow (TIOF)

Temporal Interpolation using Optic Flow (TIOF) applies a typical optic flow [4] model to temporal interpolation. Similar to MCTI, for any k , our goal is to generate a frame to be inserted between the $(k-1)^{th}$ and k^{th} received frames so that object motion would be smoother. This time, instead of block-based motion model, we use optic flow, which is a pixel-based motion model, for interpolation. Forward and backward motion vectors for each pixel are found iteratively. Each pixel in the inserted frame is generated by motion compensation, which is similar to that in MCTI.

3. Simulation Results and Discussion

The two algorithms, block-based MCTI and pixel-based TIOF, are simulated using the luminance component of "Miss America" and "Salesman" sequences which are in CIF (352×288) format. In analyzing the simulation results, besides subjective visual quality, we will compare the quality by comparing their peak-signal-to-noise-ratio (PSNR). For the sake of simplicity, the generated frame is interpolated from the original $(k-1)^{th}$ frame and the original $(k+1)^{th}$ frame.

Figure 2 shows the plot of PSNR against frame number for the "Miss America" sequence interpolated temporally using Temporal Interpolation using Optic Flow (TIOF) and Motion Compensated Temporal Interpolation (MCTI). The average PSNR for TIOF and MCTI are shown in Table 1.

	TIOF	MCTI
Average PSNR	38.32dB	37.44dB

Table 1 Average PSNR of "Miss America"

Fig. 2 Comparison of PSNR of TIOF and MCTI ("Miss America")

From the plot in Figure 2, we can see that the PSNR of TIOF is significantly higher than that of MCTI in most of the frames. On the average, the PSNR of TIOF is almost 1dB higher than that of MCTI in our simulation, as can be seen in Table 1. For the "Miss America" sequence, the pixel-based TIOF gives better visual quality than the block-based MCTI.

Shown in Figure 3 is the original frame 7 of the "Miss America" sequence, a difficult frame to interpolate. In this portion of the sequence, the eyes of Miss America is opened in frame #6 and then closed in frame #8. In the actual frame #7, the eyes are half closed.

Fig. 3 Original frame 7 of "Miss America"

Fig. 4 Interpolated frame 7 using TIOF (37.86dB)

Fig. 5 Error map of frame 7 using TIOF

Fig. 6 Interpolated frame 7 using MCTI (36.35dB)

Fig. 7 Error map of frame 7 using MCTI

Figures 4 and 6 shows the interpolated frames using TIOF and MCTI respectively. The corresponding error images are shown in Figures 5 and 7 respectively.

Both TIOF and MCTI generate the eye by blurring it. As expected, TIOF succeeds in removing the blocking artifacts of MCTI. According to the error map, we can see that, for both TIOF and MCTI, the error are mainly due to the moving parts, the eyes in particular. Note that the region of error for the eyes is larger in MCTI than that in TIOF. This is probably due to the disadvantage of block-based approach as mentioned before.

In block-based approach, all blocks are assumed to undergo linear translational motion. If the object inside a block is deforming, such as the eyes of Miss America, such that the assumption no longer holds, the block-based MCTI cannot interpolate properly. However, when TIOF is used, the pixel-based model can describe motion of individual pixels and hence the motion of the eyes can be better described.

In block-based MCTI, all the objects in a block are assumed to have the same motion. But, not all objects inside a block undergo the same motion in real situations. As a result, using MCTI, there are errors in the collar of Miss America where the neck moves but her T-shirt remains stationary. These errors are absent when the pixel-based TIOF is used.

So far, TIOF appears to be better than MCTI. However, pixel-based TIOF does not perform well for all the cases. When the simulation is performed on the "Salesman" sequence, the pixel-based TIOF has poorer result than the block-based MCTI. This can be seen from the results below.

Figure 8 shows the plots of PSNR against frame number for the "Salesman" sequence using TIOF and MCTI. The average PSNR for TIOF and MCTI are shown in Table 2.

	TIOF	MCTI
Average PSNR	34.75dB	34.84dB

Table 2 Average PSNR of "Salesman" sequence

Fig. 8 Comparison of PSNR using TIOF and MCTI ("Salesman")

Unlike the simulation using the "Miss America" sequence, the PSNR of TIOF on "Salesman" in Figure 8 is lower than that of MCTI in most of the frames. This is confirmed by the lower average PSNR of TIOF in Table 2.

Shown in Fig. 9 is the original frame 13 of the "Salesman" sequence, a very difficult frame to interpolate. In this portion of the sequence, the left hand of the salesman undergoes rapid rotational motion, while the right hand, together with the box, undergoes rapid deformation. With non-translational motion, both TIOF and MCTI may have problems.

Fig. 9 Original frame 13 of "Salesman"

Fig. 10 Interpolated frame 13 using TIOF (32.81dB)

Fig. 11 Error map of frame 13 using TIOF

Fig. 12 Interpolated frame 13 using MCTI (33.70dB)

Fig. 13 Error map of frame 13 using MCTI

Similar to the "Miss America" sequence, according to the error map, the error for both TIOF and MCTI are mainly due to the moving parts, the left hand and the box in the right hand in particular.

In the MCTI generated Figure 12, lots of blocking artifacts occur around the left hand, the right hand and the box. Most of the block-based motion field are wrong in these areas and thus blocking artifacts result. In the TIOF generated Figure 10, the error can be easily seen from the image directly. Both the left hand and the box in the right hand do not interpolated well using TIOF. From the pixel-based motion field, it was found that most of the motion vectors are wrong due to the inability of the motion estimation algorithm to track fast and non-translational motion. As a result, there are dot errors (salt-and-pepper artifacts) which are due to the wrong motion vectors for the pixels in these two regions. This suggests that the pixel-based

interpolation is very sensitive to the error of the motion vector, probably even more sensitive than block-based MCTI because the resulting PSNR of TIOF is lower than MCTI.

Errors in motion lead to blocking artifacts in the block-based MCTI, and salt-and-pepper artifacts in the pixel-based TIOF. The two kinds of artifacts are different but are both annoying.

In terms of computation, MCTI requires a lot of computation, in the order of 10^8 per frame. This makes it less than practical and thus fast algorithms are investigated in [1]. On the other hand, TIOF requires even more computation than MCTI. It is in the order of 10^9 , which makes it highly impractical.

Considering the varying performance and high computation requirement of pixel-based approach, we find that it is more beneficial to pursue the block-based approach, to improve its performance and reduce its computation requirement.

4. Conclusions

In this paper, we study the pros and cons of temporal interpolation using block-based MCTI and pixel-based TIOF to generate the skipped frame in temporally subsampled video sequence. Simulation results suggest that the pixel-based TIOF does not introduce the blocking artifacts as block-based MCTI does. However, the pixel-based TIOF has its unique salt-and-pepper artifacts which is annoying. A serious drawback of the pixel-based approach is the extremely huge computational requirement which makes it impractical. We conclude that the block-based approach is more suitable for temporal interpolation than the pixel-based approach.

5. Acknowledgement

This work was supported by RGC Grant HKUST695/95E.

6. References

- [1] C.K. Wong, O.C. Au, "Fast Motion Compensated Temporal Interpolation for Video", *Proc. SPIE: Visual Communication and Image Processing*, Vol.2, pp.1108-1118, May 1995.
- [2] C.K. Wong, O.C. Au, "Modified Motion Compensated Temporal Interpolation for Very-low-bit-rate Video", *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Vol.4, pp.2327-2330, May 1996.
- [3] C.K. Wong, O.C. Au and C.W. Tang, "Motion Compensated Temporal Interpolation with Overlapping", *Proc. IEEE Int. Sym. Circuits and Systems*, Vol.2, pp.608-611, May 1996.
- [4] Ajit Singh, *Optic Flow Computation*, IEEE Computer Society Press, Ch. 5, pp.55-86, 1991.