WIPE PRODUCTION IN MPEG-2 COMPRESSED VIDEO

W.A.C. Fernando, C.N. Canagarajah, D. R. Bull

Image Communications Group, Centre for Communications Research, University of Bristol,
Merchant Ventures Building, Woodland Road, Bristol BS8 1UB, United Kingdom.
Voice - +44-117-954-5198, Fax - +44-117-954-5206
Email - W.A.C.Fernando@bristol.ac.uk

ABSTRACT

With the increased role of technology in video production, several types of complex video special effects editing have begun to appear. In this paper we consider wiping special effects editing in MPEG-2 [1] compressed video without full frame decompression and motion estimation. We estimated the DCT coefficients and use these coefficients together with the existing motion vectors to produce these special effects editing in compressed domain. Results show that both objective and subjective quality of the edited video in compressed domain closely follows the quality of the video edited in uncompressed domain at the same bit rate.

1. INTRODUCTION

Video special effects are needed to enhance the quality of the video production. Most special effects can be divided into three major categories: dissolving, fading and wiping. All these special effects are used to produce a gradual scene change between two scenes. However, these video editing tools are designed for spatial domain processing. The large channel bandwidth and memory requirements for the transmission and storage of image and video necessitate the use of video compression techniques [1,2]. Hence, the visual data in multimedia databases is expected to be stored mostly in the compressed form. Therefore, a typical desktop video editing system must first convert the compressed domain representation to a spatial domain representation and then perform the editing function on the spatial domain data. This increases the overall computational complexity of the editing process. In order to avoid the unnecessary decompression operations and recompression processes, it is efficient to edit the image and video in the compressed format itself.

Smith [3] showed how the algebraic operation of pixel-wise scalar addition and multiplication can be done in DCT compressed domain. He used these operations on JPEG [4] images to implement two common video transformations: dissolving and sub-titling. Author argued that scalar addition and multiplication could be implemented on quantised matrices. However due to the non-linear behaviour of the mapping function, many problems were introduced with this scheme [3]. Another limitation of this scheme is the problems associated with extending this scheme for video compression standards such as MPEG-2. Shen proposed DC-only fade-out operation for MPEG-2 compressed video [5]. This algorithm is proposed under the crude assumption that fade-out is viewed as a reduction of picture brightness. However, this is a poor approximation of the actual fade-out operation in video production. We proposed fade-out and fade-in special effects editing in MPEG-2 compressed video without full frame decompression and motion estimation [6]. In this paper we present a novel technique for wipe editing and production in compressed video without full frame decompression and re-compression.

Rest of the paper is organised as follows. We present briefly in section 2 related mathematical models for compressed domain video editing. Proposed schemes for video special editing are presented in section 3. Experimental results are given in section 4. Finally, section 5 presents the conclusions and future work.

2. VIDEO EDITING IN COMPRESSED DOMAIN

2.1 Mathematical representation of wiping

Let,

A(n)   *M*x*N* vector representing the pixel values in frame "*n*" in a video sequence A.

B(n)   *M*x*N* vector representing the pixel values in frame "*n*" in a video sequence B.

P(n)   *M*x*N* vector representing the wiping transition. (elements of *P(n)* are either "*1*" or "*0*" always)

W(n)   *M*x*N* vector representing the pixel values in frame "*n*" in a video sequence composed from video sequence A and B with wiping.

Consider a video sequence of length S_E having wipe transition from W_S to W_E. Then, $W(n)$ can be described as in Equation (1).

$$W(n) = \begin{cases} A(n) & n < W_S \\ P(n) \otimes A(n) + \overline{P}(n) \otimes B(n) & W_S \leq n \leq W_E \\ B(n) & W_E < n \leq S_E \end{cases} \quad (1)$$

where, "$\otimes$" denotes element by element matrix multiplication and matrix P generates the wiping pattern.

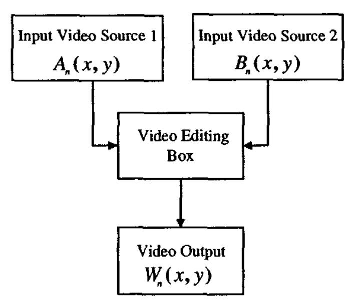

Figure 1: Uncompressed domain video editor

Figure 1 illustrates a typical uncompressed domain wipe editor. The function of the editing box is to generate *W(n)* according to the

0-7803-6297-7/00/$10.00 © 2000 IEEE

wipe generating matrix *P(n)*. When both $A_n(x,y)$ and $B_n(x,y)$ are compressed, this operation cannot be done in compressed domain without some additional processing. A straightforward way to performing wiping on compressed sequence would be to decompress the sequence, apply wiping operation in space (time) domain and recompress the video. Figure 2 presents this model. Within this loop, costly IDCT, DCT and motion estimation operations make it difficult for real time implementations. Here, we propose an alternative scheme to do wiping directly in compressed domain with minimum decompression.

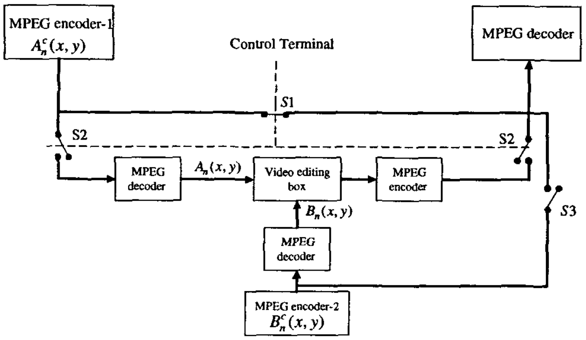

Figure 2: Conventional compressed domain video editor (A_n^c – Compressed signal of A_n and B_n^c – Compressed signal of B_n)

2.2 DCT coefficients estimation

In this section we show how DCT coefficients are estimated for inter-coded frames in MPEG-2 video.

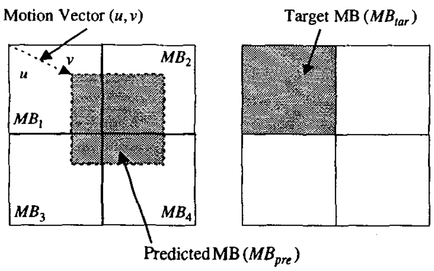

Figure 3: Graphical representation of four neighbouring MBs and motion vectors

$$DCT(MB_{vir}) = \sum_{i=1}^{4} DCT(S_{i1})DCT(MB_i)DCT(S_{i2}) \qquad (2)$$

A graphical representation of four neighbouring MBs and motion vectors are shown in Figure 3. DCT coefficients in inter-coded frames can be extracted from an intra-coded frame. The computation of the DCT coefficients of a image block at a new arbitrary position can be obtained from the DCT coefficients of the four original neighbouring blocks by pre-multiplying and post-multiplying with appropriate matrices (S_{i1} and S_{i2}) [7]. Equation 2 describes this process of DCT coefficients evaluation of a motion-compensated MB. There are four possible locations of the sub-block of interest (with reference to MB_i): upper-left, upper-right, lower-left and lower-right. These locations define S_{i1} and S_{i2} matrices as tabulated in Table 1. Parameters h_i and w_i are the height and width of the overlap of MB_{pre} with MB_i. For a particular MB, these two parameters can be evaluated from its motion vector (u,v). Therefore, DCT coefficients can be evaluated for the predicted MB using Equation (2). DCT coefficients of the error term are readily available for MPEG-2 compressed video. Finally, DCT coefficients of the target MB (MB_{tar}) are calculated by adding the DCT coefficients of the predicted MB and the DCT coefficients of the predicted error signal.

Sub-block	Position	S_{i1}	S_{i2}
MB_1	lower right	$\begin{bmatrix} 0 & I_{h_1} \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ I_{w_1} & 0 \end{bmatrix}$
MB_2	lower left	$\begin{bmatrix} 0 & I_{h_2} \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & I_{w_2} \\ 0 & 0 \end{bmatrix}$
MB_3	upper right	$\begin{bmatrix} 0 & 0 \\ I_{h_3} & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ I_{w_3} & 0 \end{bmatrix}$
MB_4	upper left	$\begin{bmatrix} 0 & 0 \\ I_{h_4} & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & I_{w_4} \\ 0 & 0 \end{bmatrix}$

Table 1: S_{i1} and S_{i2} matrices

3. PROPOSED VIDEO EDITING SYSTEM

Normally, motion estimation process bears 60-70% of computational complexity of a typical MPEG video encoder. Therefore, in order to keep the computational complexity of the video editing process at a reasonable level, it is desirable to reuse the motion vectors as much as possible without re-computing them. In this proposed scheme, we use existing motion vectors and DCT coefficients to generate different wiping patterns.

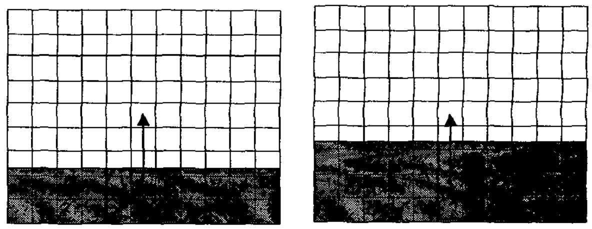

Figure 4-a: Vertical wiping with picture A and B in uncompressed domain (Frame "*n-3*" and Frame "*n*")

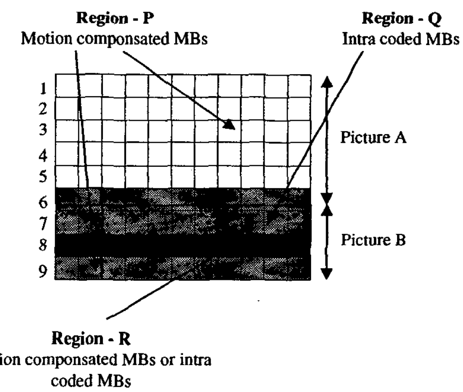

Figure 4-b: Vertical wiping with picture A and B in compressed domain (Frame "*n*")

Consider Figure 4 to illustrate the operations for forward vertical wiping production. In Figure 4 each small rectangle represents a 16x16 MB. An intermediate step of this wiping operation is shown in Figure 4-a. Assume that frame "*n-3*" is an I/P frame whereas frame "*n*" is a P-frame. Figure 4-b presents how the MBs are coded in compressed domain for this wiping pattern. Rows one to five and nine (**region-P**) in frame "*n*" can be motion compensated from frame "*n-3*". MBs in row eight (**region-R**) either can be intra-coded or motion compensated depending on the magnitude of motion vectors. If any motion vector points to raw seven, these MBs are intra-coded. Otherwise MBs are motion compensated. Rows six and seven (**region-Q**) are intra-coded as they cannot be motion compensated. Motion vectors and relevent DFD signals are extracted from the two sequences accordingly. DCT values are calculated using the technique explained in section 2.2. It should be noted that MBs in row six are composed with both picture A and picture B. DCT of these MBs can be calculated as shown in Equation (3). Similar set of Equations can be derived for other wiping patterns as well. Appendix shows some of the Equations for common wiping patterns.

$$DCT(MB_{resultant}) = DCT(\zeta)DCT(MB_A) + DCT(\vartheta)DCT(MB_B) \qquad (3)$$

where,

$$\zeta = \begin{bmatrix} I_{h_i} & 0 \\ 0 & 0 \end{bmatrix}, \qquad \vartheta = \begin{bmatrix} 0 & 0 \\ 0 & I_{8-h_i} \end{bmatrix}$$

Parameter h_i is dependent on wiping speed, which is selected by the user.

Similar arguments can be followed for other frames as well. Using this approach, we can produce forward vertical wiping in compressed domain. Vertical backward, horizontal forward, horizontal backward, barn-door forward, barn-door backward, box-wipe backward, box-wipe forward can also be generated in a similar fashion.

3.1 Implementation of the proposed video editing system

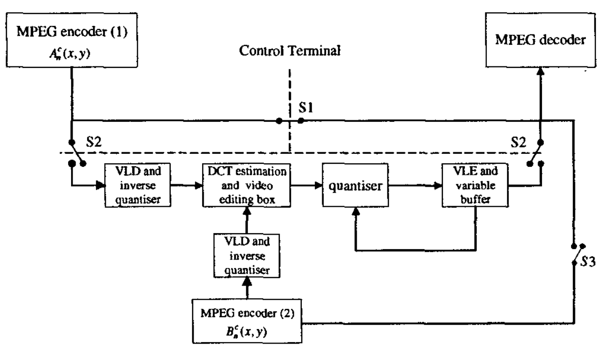

Figure 5: Proposed compressed domain video editor

Figure 5 shows the proposed compressed domain video editor. Two MPEG compressed video streams are applied to the editor and these two streams are partially decoded with variable length decoder (VLD) and the inverse quantiser. With this data, we estimate DCT coefficients for each frame. Then these coefficients and motion vectors are applied to the video editing box to process the parameters as explained previously. Finally, all MBs are quantised to suit the channel bit rate. The quantisation is achieved through the feedback loop from the buffer as in normal MPEG-2 video encoder.

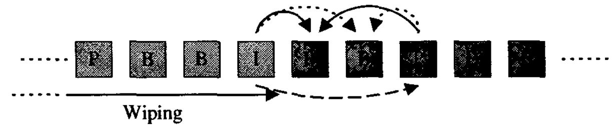

Figure 6-a: Wiping ends at I/P-frame

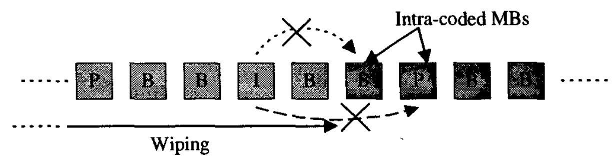

Figure 6-b: Wiping ends at first B-frame

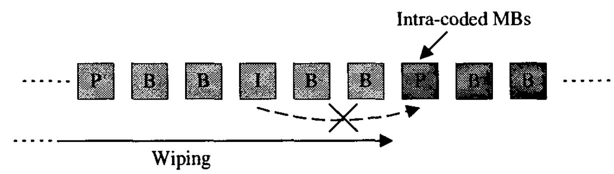

Figure 6-c: Wiping ends at second B-frame

When user needs to switch the video into a different sequence through any special effects operation, S2 switch will be closed and S1 switch will be opened through control terminal. The wiping pattern and the length of the wiping operation can be controlled by the user. There are twelve locations, where wiping can be terminated within a GOP (12,3) structure. It should be noted that the last frame in the wiping operation is an unaltered picture from the second sequence, $B_n(x,y)$. Therefore, if the last frame of the wiping process is an I-picture or a P-picture (originally) as shown in Figure 6-a, then no further processing is required for following frames to synchronise with the sequence. Depending on whether the last frame is a B1- picture or B2-picture (Figure 6-b and 6-c) some MBs are needed to be intra-coded in two or one future pictures respectively. After the wipe operation, S2 switch will be opened and S3 switch will be closed.

4. RESULTS

Here we are presenting some simulations results for wiping video special effects performed using the proposed scheme. We used MPEG-2 video streams as input to the proposed compressed domain video editor and fixed the channel bit rate. We compared the performance of the proposed algorithm with the conventional scheme (Figure 2). PSNR of the luminance signal is considered to measure the objective quality of the edited video using these two techniques. We considered the signal $W_n(x,y)$ (output at the uncompressed video editor – Figure 1) as the reference signal for all PSNR calculations. Figure 7 shows the comparison for a forward vertical wiping special effect editing in compressed domain. It shows that PSNR of the proposed scheme closely follows the PSNR of the conventional scheme. The drop in average PSNR is very small. This is due to existing motion vectors can be used very effectively in wipe editing. Performance comparison of the subjective quality of the wipe edited video is presented in Figure 8. Subjective quality of the proposed scheme is again similar to the conventional scheme. We tested this proposed scheme for common wiping patterns with different wipe rates and different MPEG-2 video bit streams and observed similar set of results. Some of these results are summarised in Table (2).

Results show that the average PSNR of the proposed scheme closely follows the average PSNR of the conventional scheme. The small drop in average PSNR of the proposed scheme is acceptable when we consider the significant reduction in computational complexity. Conventional scheme is computationally very expensive compared to the proposed scheme and make it difficult for real time implementations. Therefore, these special effects editing can be done in compressed domain itself without full frame decompression and motion re-estimation.

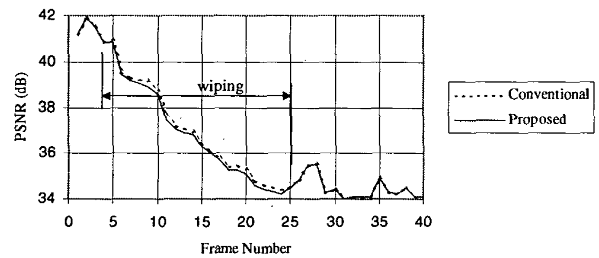

Figure 7: Performance comparison for wipe production

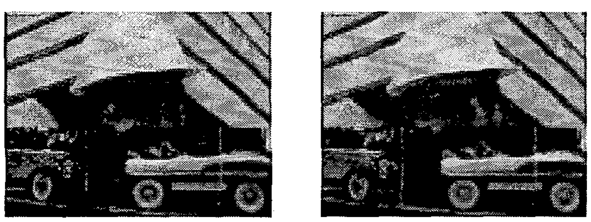

Figure 8: Frame-12 of the wipe operation
(from left, conventional and proposed)

Description	Wiping Rate	Average PSNR (dB)	
		Conventional	Proposed
Vertical (F)	10	41.0	40.2
Horizontal(F)	12	40.1	39.6
Barn-doors(B)	20	37.6	37.1
Horizontal(F)	20	47.8	47.4
Vertical (B)	30	47.4	46.8
Horizontal(F)	10	45.8	45.2
Barn-doors(F)	15	48.4	47.7

Table 2: Summarised results (F-forward, B-Backward)

5. CONCLUSIONS

In this paper we have developed an algorithm for most common wipe editing operations in MPEG-2 compressed domain itself without full frame decompression and motion re-estimation. DCT coefficients are estimated and used these coefficients together with the existing motion vectors to produce wipe special effects in compressed domain. We also made allowances for various end points in the MPEG sequence (i.e. I, P or B frames). Results show that the subjective and objective quality of the edited video with the proposed scheme closely follows the quality of the edited video with the conventional method. Unlike the conventional scheme, proposed scheme is computationally inexpensive and makes it possible for real time implementations. Future work is required to extend this work for more complex wipe patterns such as circular and star wiping in compressed domain.

ACKNOWLEDGEMENTS

First author would like to express his gratitude and sincere appreciation to the University of Bristol and CVCP for providing financial support for this work.

APPENDIX

Wiping pattern	$DCT(MB_{\text{resultant}})$
Vertical	$DCT(\zeta)DCT(MB_A)+DCT(\vartheta)DCT(MB_B)$
Horizontal	$DCT(MB_A)DCT(\zeta)+DCT(MB_B)DCT(\vartheta)$
Barn-door (Vertical)	$DCT(\zeta_1)DCT(MB_A)+DCT(\vartheta_1)DCT(MB_B)$
Barn-door (Horizontal)	$DCT(MB_A)DCT(\zeta_1)+DCT(MB_B)DCT(\vartheta_1)$
Box	$DCT(\zeta_1)DCT(MB_A)DCT(\zeta_1)+DCT(\vartheta_1)DCT(MB_B)+$ $DCT(\zeta_1)DCT(MB_A)DCT(\vartheta_1)$

where $\zeta=\begin{bmatrix} I_{h_i} & 0 \\ 0 & 0 \end{bmatrix}, \vartheta=\begin{bmatrix} 0 & 0 \\ 0 & I_{8-h_i} \end{bmatrix}, \zeta_1=\begin{bmatrix} 0 & 0 & 0 \\ 0 & I_{h_i} & 0 \\ 0 & 0 & 0 \end{bmatrix},$

$$\vartheta_1=\begin{bmatrix} I_{g_i} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & I_{8-h_i-g_i} \end{bmatrix}$$

REFERENCES

1. Sikora, T., 'MPEG Digital Video-Coding Standards', *IEEE Signal Processing Magazine*, pp. 82-100, September 1997.
2. ITU-T, 'Draft H.263', May 1996.
3. Smith, B.C., Rowe, L.A., 'A New Family of Algorithms for Manipulating Compressed Images', *IEEE Computer Graphics and Applications*, pp. 34-42, September 1993.
4. Wallace, G.K., 'The JPEG Still Picture Compression Standard', *Communications of the ACM*, Vol. 34, pp. 30-44, April 1991.
5. Shen, B., 'Fast Fade-out Operation on MPEG Video', *Proceedings of IEEE International Conference on Image Processing*, CD Paper Number - MP10.01, 1998.
6. Fernando, W.A.C., Canagarajah, C.N. and Bull, D. R., 'Video Special Effects Editing in MPEG-2 Compressed Video', *Proceedings of International Symposium on Circuits and Systems, Volume IV-Image and Video Processing*, Paper Number 651, 2000.
7. Chang, S.F., Messerschmitt, D.G., 'Manipulation and Composting of MC-DCT Compressed Video', *IEEE Journal of Selected Areas in Communications*, vol.13, pp.1-11, January 1995.