# New DCT-Domain Transcoding using Split and Merge Technique

*Kai-Tat Fung and Wan-Chi Siu*

Centre for Multimedia Signal Processing
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

## ABSTRACT

For the conventional downscaling video transcoder, the video server would be to first decompress the video, perform the downscaling operation in the pixel domain, and then recompress it. This is computational intensive. However, it is difficult to perform video downscaling in the DCT-domain since the prediction errors of each frame are computed from its immediate past higher resolution frames. Besides, the motion vector need to resample due to the size of the video is changed. Due to the mismatch of the resampled motion vector with the incoming DCT coefficients, the video transcoder need to recalculate the new DCT coefficient with lower resolution in pixel domain; this can create undesirable complexity as well as introduce re-encoding error. In this paper, we propose a new architecture to obtain the new DCT coefficients and the new motion vector by reuse the incoming motion vector and DCT coefficients. Since our proposed transcoder is mainly performing in DCT domain, low computational complexity can be achieved as well as the re-encoding can be reduced. Experimental result show that our proposed video downscaling transcoder can lead to significant computational savings as well as provide a high video quality compared to the conventional approach.

## 1. Introduction

Transcoding techniques are becoming an important role for a video server to provide quality support services to heterogeneous clients or transmission channels[1-7]. It is in this scenario that the video server should have capability of performing transcoding using different transcoding approaches to converting a previously compressed video bitstream into a lower bitrate bitstream without modifying its original structure according to the client devices (e.g. mobile communication) in terms of its computational complexity and bandwidth constraint. Several transcoding techniques are proposed recently such as requantization[1], frame rate reduction[2] and video downscaling[3-5] to reduce the bitrate of the incoming bitstream. In this paper, we will focus on the technique for video downscaling. One straightforward approach for implementing transcoding is to cascade a decoder and an encoder [1], commonly known as pixel-domain transcoding. To downscale the encoded video produced by the current video compression standards such as MPEG, H.261 or H.263 which employ motion compensated prediction to exploit the temporal redundancy to achieve a lower bitrate. The conventional approach needs to decompress the video and performs downscaling of the video in pixel domain. Then new motion vectors and DCT coefficients for this downscaled video are needed to recompute inside a transcoder. This involves a high computational complexity, memory, and long delay on a video server to generate the downscaled video. One of the simplest approach to reduce the computational complexity may be take the average of the four motion vectors associated with the four macroblocks and halve it so that the resulting motion vector can be associated with the 16x16 macroblock of the downscaled by two video. The motion vectors obtained in this manner are not optimal[3]. As a consequence, some information reusing approaches [3-5] such as adaptive motion vector resampling [3] for downscaled videos were used to provide a computational efficient solution to recompose the new motion vector. However, these approaches only deal with the problem of motion re-estimation during the transcoding process, the DCT coefficients with lower resolution are required to be recomputed. Due to the mismatch of the resultant motion vector with the incoming DCT coefficients, the video transcoder has to recalculate the new DCT coefficients with lower resolution in pixel domain; this can create undesirable complexity as well as introduce re-encoding error. The effect of re-encoding errors is depicted in Figure 1 where the "Table Tennis" sequence was transcoded at quarter of the incoming frame size. This figure shows that re-encoding errors lead to a significant degradation in the picture quality.

Figure 1. The re-encoding error introduced by the video downscaling transcoder using AMVR system.

Motivated by this, we propose a new architecture to obtain the new DCT coefficients and new motion vector by reusing the incoming motion vector and DCT coefficients to avoid the complexity and the quality degradation arising from pixel-domain transcoding. In addition, an adaptive feedback control scheme is proposed, which can adaptively control the re-encoding errors due to transcoding and avoid unnecessary operation. Due to the transcoding operation mainly in DCT domain, the computational complexity and re-encoding error can be reduced significantly. As a result, our proposed video downscaling transcoder which has an architecture of low-complexity can provide a better transcoded sequence.

## 2. Low Complexity and High quality Video Downscaling in DCT Domain Transcoding

In this section, we present a new DCT-based video downscaling transcoding architecture. The new architecture has focus on the following areas:

2.1 Transcoding the overlapping region of MC macroblocks in DCT domain using the minimum distance motion vector with unchange video resolution (MVMD)

2.2 Adaptive transcoding of the non-overlapping region MC macroblocks boundary

0-7803-7750-8/03/$17.00 ©2003 IEEE

2.3 Reconstruction of the new prediction errors and adaptive re-encoding error control architecture

The architecture of the proposed transcoder is shown in Figure 2. The input bitstream is first parsed with a variable-length decoder to extract the header information, coding mode, motion vectors and quantized DCT coefficients for each macroblock. Each macroblock is then manipulated independently. Switches, $SW_1$ is employed to pass the reconstructed quantized DCT coefficients to the DCT-domain downsampling operator, for the transformed and quantized residual signal depending on the coding mode originally used in the front encoder for the current macroblock being processed. For non-MC macroblocks or well-align case, the incoming prediction errors in the DCT-domain is directly downsampled in DCT domain. Hence, low computational complexity can be achieved and the quality degradation introduced using the pixel domain approach can be avoided.

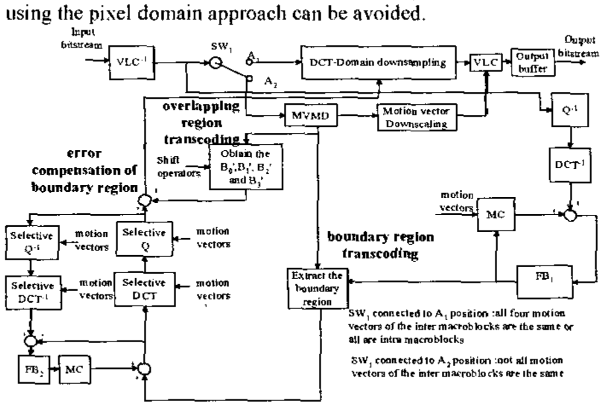



Figure 2. Architecture of a possible DCT-based video downscaling video transcoder.

When the motion vectors are not well aligned, directly downsampling in DCT domain cannot be achieved since the incoming prediction errors will be mismatch with the reconstructed new motion vector. The major difficulties to transcode these MC macroblocks is that re-encoding errors will be generated due to the re-encoding process of the new DCT coefficients, which introduces quality degradation in the transcoded sequence. Also, high computational complexity is required. Motivated by this, our proposed architecture transcodes the new prediction errors mainly in DCT domain by reusing the incoming DCT coefficients and motion vectors. By calculate the new motion vector with the minimum distance (MVMD) among the four incoming motion vectors as shown in Figure 3, the overlapping region between the incoming DCT coefficients and the targeted new DCT coefficients can be reused. In other words, full inverse DCT, forward DCT, quantization and requantization are not required in the overlapping region. Therefore, the video quality degradation in the overlapping region can be avoided and low computational complexity can be achieved. For the boundary region, adaptive DCT, adaptive IDCT, adaptive quantization and adaptive requantization are used to calculate the DCT coefficients in the boundary region. The frame buffer, FB2, is also proposed to feedback the re-encoding error introduced in the boundary region of the macroblock. Hence, the re-encoding error introduced in the boundary region can be controlled more dynamically without introducing any redundant operations.

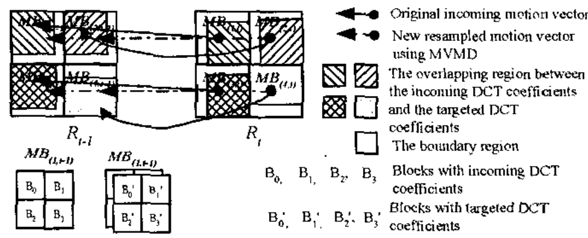



Figure 3. Diagram showing the possibility of avoiding video quality degradation in overlapping regions.

### 2.1 Transcoding the overlapping region of MC macroblocks in DCT domain using the minimum distance motion vector with unchange video resolution (MVMD)

In this paper, we split the MC macroblocks in two regions: overlapping region and boundary region as shown in Figure 3. In the overlapping region transcoding, we propose a new minimum distance motion vector and the shifted operator to compute the DCT coefficients in overlapping region in DCT domain to achieve low computational complexity and avoid the re-encoding error.

Figure 3 shows the scenario that the macroblock is pointing to the macroblock in the previous frame. The prediction error obtained from the incoming bitstream are $B_0$, $B_1$,...$B_{15}$ as shown in Figure 4a, where $B_0$, $B_1$, $B_2$, and $B_3$ have one motion vector. And the other blocks have other three different motion vectors. Since our proposed new resampled motion vector before downscaling is different from the incoming motion vectors as shown in Figure 3, the new prediction error $B_0'$, $B_1'$, $B_2'$ and $B_3'$ reference to the previous frame as shown in Figure 3 is needed to obtain in order to employ the downscaling of DCT coefficients in DCT domain. Our proposed resampled motion vector can be obtained as described in the following equations:

$$\text{Cost\_f}(x) = (x-x_1)^2 + (x-x_2)^2 + (x-x_3)^2 + (x-x_4)^2 \quad (1)$$

$$\text{Cost\_f}(y) = (y-y_1)^2 + (y-y_2)^2 + (y-y_3)^2 + (y-y_4)^2 \quad (2)$$

where x and y are resampled motion vector in horizontal and vertical direction respectively and $(x_1,y_1)$,...$(x_4,y_4)$ are the motion vectors from the incoming bitstream as shown in Figure 3. Note that if all the incoming motion vectors are the same, i.e. $x_1=x_2=x_3=x_4$ and $y_1=y_2=y_3=y_4$, the new motion vector will be $(x_1,y_1)$. Otherwise, we need to minimize the cost function in both vertical and horizontal direction as shown below:

$$\frac{d}{dx}\text{Cost\_f}(x) = -2(x-x_1) - 2(x-x_2) - 2(x-x_3) - 2(x-x_4) \quad (3)$$

$$\frac{d}{dy}\text{Cost\_f}(y) = -2(y-y_1) - 2(y-y_2) - 2(y-y_3) - 2(y-y_4) \quad (4)$$

Set the derivatives to zero, we have

$$x = \frac{x_1+x_2+x_3+x_4}{4},\ y = \frac{y_1+y_2+y_3+y_4}{4} \quad (5)$$

Then the new prediction error $B_0'$, $B_1'$, $B_2'$ and $B_3'$ can employs this resampled motion vector (x,y) with minimun distance ($dx=x-x_i$ and $dy=y-y_i$) from the original motion vector $(x_i,y_i)$to avoid the re-encoding error in the overlapping region if $B_0'$, $B_1'$, $B_2'$ and $B_3'$ can be obtained in DCT domain directly from $B_0$, $B_1$, $B_2$, $B_3$, $B_4$, $B_5$, $B_8$, $B_{10}$ and $B_{12}$ as shown in Figure 4a without performing requantization process during transcoding.

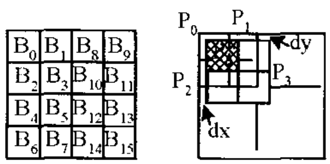
Since the shifted version of the original motion vector is used, the corresponding prediction error in the overlapping region can be obtained without performing full re-encoding process. If the other three motion vectors have the same direction (i.e. well-aligned), the whole new prediction ($B_0$', $B_1$', $B_2$' and $B_3$') can be obtained in the DCT domain. In other words, no re-encoding error will be introduced. Otherwise, decomposition of the overlapping region and boundary region are needed.

The major idea to obtain the DCT coefficients in overlapping region is to represent it as an addition of horizontally and/or vertically displaced anchor blocks. Then the DCT values of the $B_0$', $B_1$', $B_2$' and $B_3$' is constructed using the pre-computed DCT values of the shifting matrics.

Consider $P_0$' be the targeted block of interest in pixel domain as shown in Figure 4b, $P_0$, $P_1$, $P_2$ and $P_3$ are the four neighboring blocks in pixel domain from which $P_0$' is derived, and the new resampled motion vector obtained using minimum distance derivation approach is (x,y)with a minimum distance ($dx=x-x_i$ and $dy=y-y_i$) from the original one. The shaded regions in $P_0$, $P_1$, $P_2$ and $P_3$ are moved by dx and dy. Then $P_0$' can be represent in the following equation.

$$P_0' = \sum_{i=0}^{3} S_{i1} P_i S_{i2} \qquad (6)$$

where $S_{ij}$ are matrics tabulated in table 1.




(4a) (4b)

Figure 4a) Incoming DCT coefficients of four macroblocks. 4b) The overlapping region and the boundary region of $MB_{t-1}$.

Table 1. The shifting operators.

| Sub-block | $P_0$ | $P_1$ | $P_2$ | $P_3$ |
|---|---|---|---|---|
| $S_{i1}$ | $\begin{bmatrix} 0 & I_{hd} \\ 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & I_{hd} \\ 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 0 \\ I_{hb} & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 0 \\ I_{hb} & 0 \end{bmatrix}$ |
| $S_{i2}$ | $\begin{bmatrix} 0 & 0 \\ I_{wd} & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & I_{wb} \\ 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 0 \\ I_{wd} & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & I_{wb} \\ 0 & 0 \end{bmatrix}$ |

Each I is an identity matrix of size hb=dy or hd=8-dy or wb=dx or wd=8-dx. The pre-multiplication shifts the sub-block of interest horizontally while post-multiplication shifts the sub-block vertically. Four possible locations of the sub-block of interest: upper-left, upper-right, lower-left and lower-right are shown in Table 1.

Define the 2D-DCT of an 8x8 block A as

$$DCT(A) = \hat{A} = TAT^t \qquad (7)$$

where T is the 8x8 DCT matrix with entries t(i,j) given by

$$t(i,j) = \frac{1}{2} k(i) \cos\frac{(2j+1)i\pi}{16} \qquad (8)$$

where i represents the row index and j represents column index and

$$k(i) = \begin{cases} \frac{1}{\sqrt{2}}, & i=0 \\ 1, & otherwise. \end{cases} \qquad (9)$$

Using the properties of DCT,

$$DCT(CD)=DCT(C)DCT(D) \qquad (10)$$

where C and D are 8x8 matrices.

From eq(6), we have,

$$DCT(P_0') = \sum_{i=0}^{3} DCT(S_{i1}) DCT(P_i) DCT(S_{i2}) \qquad (11)$$

$$\Rightarrow B_0' = \sum_{i=0}^{3} DCT(S_{i1})(B_i) DCT(S_{i2}) \qquad (12)$$

Note that the DCT of the $S_{i1}$ and $S_{i2}$ can be pre-computed and $B_i$ can be extracted from the incoming bitstream, so the re-encoding and high computational complexity can be avoided. If the macroblocks $MB_{(1,t)}$, $MB_{(2,t)}$, $MB_{(3,t)}$ and $MB_{(4,t)}$ have the same motion vector, direct downsampling of DCT coefficients can be applied in the DCT domain. Otherwise, decomposition of the overlapping region and boundary region is needed as shown in Figure 3.

Using the properties of DCT,

$$DCT(A+B)=DCT(A)+DCT(B) \qquad (13)$$

where A and B represent the pixels in overlapping region and boundary region with size 8x8 respectively. We can split the block $B_1$',$B_2$' and $B_3$'in two regions: overlapping region and boundary region as shown in Figure 3.

Then the overlapping region of $B_2$' can be obtained by considering the boundary $B_4$ and $B_5$ with zero DCT coefficients value as shown in Figure 3. Similarly, the overlapping region of $B_1$' and $B_3$' can be obtained by considering the boundary $B_8$ and $B_{10}$ with zero DCT coefficients value and the boundary $B_5$, $B_{10}$ and $B_{12}$ with zero DCT value as shown in Figure 3. Then the DCT value of the overlapping region A can be obtained. Note that the DCT value of $B_1$', $B_2$' and $B_3$' is not completed since the DCT value in boundary region B is not considered as described in equation (13). In the following, the DCT value of boundary region B in 8x8 is need to calculate it separately by using 1-D inverse DCT, motion compensation, forward 1D-DCT and requantization as shown in Figure 3. In this process, re-encoding error cannot be avoided due to requantization.

### 2.2 Adaptive transcoding of the non-overlapping region MC macroblocks boundary

For the transcoding of MC boundary, the boundary region is extracted and selective quantization and selective 1D-DCT of the quantized DCT coefficients of $MB_{(1,t)}$, $MB_{(2,t)}$, $MB_{(3,t)}$ and $MB_{(4,t)}$ are only needed to perform in the boundary as shown in Figure 3. Note that each macroblock is composed of four *8x8* blocks in video coding standards, and the DCT and quantization operations are performed on units of *8x8* blocks. When processing $MB_{(1,t)}$, $MB_{(2,t)}$, $MB_{(3,t)}$ and $MB_{(4,t)}$, only their corresponding *8x8* blocks, which have pixels overlapping with $MB_t$ boundary, are subject to the selective inverse 1D-DCT computation. Hence, the partially inverse 1D-DCT is only performed in the boundary, so the motion vector $mv_{t-1}$ is needed as an input to the adaptive 1D-DCT module to control which rows or columns are needed to perform the 1D-DCT as shown in Figure 2. In most cases, this approach significantly reduces the required number of columns or rows DCT compare with the 2D-DCT approach.

Therefore, the new DCT coefficients of the boundary region can be obtained using partially transformed by putting the zero values in the overlapping region and perform adaptive quantization to achieve low computational complexity.

### 2.3 Reconstruction of the new prediction errors and adaptive re-encoding error control architecture

After DCT coefficients of region B are obtained, the $B_2$', $B_3$'and $B_1$' can be reconstructed by adding DCT(A) and DCT(B) together using equation (13). In Figure 2, the newly quantized DCT coefficient of a MC macroblock can then be perform downscaling of DCT coefficients in DCT domain as described in Figure 3. For well-aligned case, the downscaling of the incoming DCT coefficients can be performed in DCT domain directly. Conversely, requantization is performed for the formation of new DCT coefficients in the macroblock boundary in not well-aligned case, which will introduce additional re-encoding errors.

Note that the re-encoding error only introduced in the boundary region as shown in Figure 3. However, these errors will degrade the quality of the reconstructed frame. Since each P-frame is used as a reference frame for the following P-frame, quality degradation propagates to later frames in a cumulative manner. Thus, a feedback loop is suggested as shown in Figure 2 to compensate for the re-encoding errors introduced in the boundary region. The adaptive forward and inverse 1D-DCT and adaptive quantization pairs in the feedback loop are mainly responsible for minimizing re-encoding errors. For these MC macroblocks, the quantized DCT coefficients are adaptive inversely quantized and adaptive inverse 1D-DCT is then performed. The re-encoding error introduced in the boundary region can be obtained by subtracts the original signal and the recovered signal after quantization. This re-encoding error is then stored in $FB_2$ and feedback to the latter frames to avoid the accumulation of re-encoding error.

After the reconstruction of the new prediction errors, the DCT domain downsampling will be performed and the resampled motion vector will be downscaled. After downscaling process, variable length encoding is applied. Then the output data is stored into output buffer for transmission.

## 3. Experimental Results

Results of the simulations are used to compare the performance of a reference transcoder which is a conventional pixel-domain transcoder (CPDT) by employing align-to-average weighting (AAW), align-to-best weighting (ABW) or adaptive motion vector resampling (AMVR) to resample an downscaled motion vector from the incoming motion vectors of the four macroblocks. In the front encoder, the first frame was encoded as intraframe (I-frame), and the remaining frames were encoded as interframes (P-frames). Picture-coding modes was preserved during transcoding. We show that our proposed DCT-based transcoders outperform CPDT+AAW, CPDT+ABW and CPDT+AMVR in all cases as shown in Table 2. The results are more significant for the low motion activity sequences because our proposed DCT-based transcoder should not introduce any re-encoding errors in the overlapping region since the transcoding is performed in DCT domain. Therefore, significant improvement in MC regions can be achieved about 1.25-1.83dB as compared with the conventional video downscaling transcoder. Also, our proposed transcoders have a speed-up of about 1.5-2.3 times faster than that of the conventional transcoder. This is because we can achieve significant computational savings while maintaining good video quality on the overlapping region of MC macroblocks in these sequences without performing full decoding and re-encoding process.

Table 2. Average PSNR of the proposed transcoders, where the frame rate of the incoming bitstream was 30 frames/s. The front encoder for encoding "Salesman", "Miss_America", "Hall", "Tennis", "Football" and "Flower" was MPEG2 TMN5 [8].

| Sequences | Input bitrate | Average PSNR difference as compared with CPDT+AAW for MC macroblock transcoding. | | |
|---|---|---|---|---|
| | | CPDT+ABW | AMVR[3] | DCT+MVMD |
| Salesman | 512k | 0.06 | 0.41 | 1.83 |
| Miss_America | 512k | 0.09 | 0.39 | 1.74 |
| Hall | 512k | 0.11 | 0.42 | 1.62 |
| Tennis | 3M | 0.12 | 0.43 | 1.45 |
| Football | 3M | 0.27 | 0.56 | 1.25 |
| Flower | 3M | 0.08 | 0.38 | 1.31 |

## 4. Conclusion

In this paper, we have proposed a new architecture for a low-complexity and high quality video downscaling transcoder to solve the problem of MC macroblcoks transcoding. Its low complexity is achieved by: 1) Re-use the DCT coefficients for macroblocks coded with motion compensation to deactivate most of the complex modules of the transcoder and 2) an adaptive MC macroblock boundary transcoding and error compensation using selective inverse 1D-DCT, 1D-DCT, quantization and inverse quantization for motion-compensated macroblocks. Furthermore, we have also shown that transcoding the overlapping region of MC macroblocks in DCT domain using the minimum distance motion vector with unchange video resolution, an adaptive transcoding of the non-overlapping region MC macroblocks boundary and reconstruction of the new prediction errors and adaptive re-encoding error control architecture can reduces significantly the re-encoding errors due to transcoding. The overall performance of the proposed architecture produces a better picture quality than the conventional video downscaling transcoder at the same reduced bitrates.

## 5. Acknowledgements

This work is supported by the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, Hong Kong Polytechnic University. K.T. Fung acknowledges the research studentships provided by the same University.

## 6. References

[1] Jeongnam Youn, Ming-Ting Sun and Chia-Wen Lin, "Motion vector refinement for high-performance transcoding," IEEE Trans. Multimedia, vol. 1, pp. 30-40, March 1999.

[2] K. T. Fung, Y. L. Chan and W. C. Siu, "New architecture for dynamic frame-skipping transcoder," IEEE Trans. Image Processing, vol.11, pp. 886-900, August 2002.

[3] Bo Shen, Ishwar K. Sethi and Bhaskaran Vasudev, "Adaptive motion-vector resampling for compressed video downscaling," IEEE Transactions on circuit and systems for video technology, vol.9, no.6, September 1999.

[4] Tap-Peng Tan; Haiwei Sun; YongQing Liang, "On the methods and applications of arbitrarily downsizing video transcoding," in Proc. IEEE International Conference on Multimedia and Expo, vol.1, pp.609-612, 2002

[5] Zhijun Lei; Georganas, "H.263 video transcoding for spatial resolution downscaling," in Proc. International Conference on *N.D.* Information Technology: Coding and Computing, pp. 425-430, 2002.

[6] J. Chalidabhongse and C.-C. Jay Kuo, "Fast motion vector estimation using multiresolution-spatio-temporal correlations," IEEE Trans. Circuits Syst. Video Technol., vol. 7, pp. 477-488, June 1997.

[7] Shanableh, T.; Ghanbari, M., "Heterogeneous video transcoding to lower spatio-temporal resolutions and different encoding formats," IEEE Transactions on Multimedia, vol.2, pp.101-110, June 2000.

[8] ISO/IEC, "Test Model 5, TM5, ISO/IEC JTC/ SC29/ WG11/ N0400, MPEG93/457,"Apr. 1993