Low-Complexity and High-Quality Frame-Skipping Transcoder for Continuous Presence Multipoint Video Conferencing

Kai-Tat Fung, Yui-Lam Chan, and Wan-Chi Siu, *Senior Member, IEEE*

***Abstract*—This paper presents a new frame-skipping transcoding approach for video combiners in multipoint video conferencing. Transcoding is regarded as a process of converting a previously compressed video bitstream into a lower bitrate bitstream. A high transcoding ratio may result in an unacceptable picture quality when the incoming video bitstream is transcoded with the full frame rate. Frame skipping is often used as an efficient scheme to allocate more bits to representative frames, so that an acceptable quality for each frame can be maintained. However, the skipped frame must be decompressed completely, and should act as the reference frame to the nonskipped frame for reconstruction. The newly quantized DCT coefficients of prediction error need to be recomputed for the nonskipped frame with reference to the previous nonskipped frame; this can create an undesirable complexity in the real time application as well as introduce re-encoding error. A new frame-skipping transcoding architecture for improved picture quality and reduced complexity is proposed. The proposed architecture is mainly performed on the discrete cosine transform (DCT) domain to achieve a low complexity transcoder. It is observed that the re-encoding error is avoided at the frame-skipping transcoder when the strategy of direct summation of DCT coefficients is employed. By using the proposed frame-skipping transcoder and dynamically allocating more frames to the active participants in video combining, we are able to make more uniform peak signal-to-noise ratio (PSNR) performance of the subsequences and the video qualities of the active subsequences can be improved significantly.**

***Index Terms*—Compressed-domain processing, frame skipping, video compression, video conferencing, video transcoding.**

I. Introduction

With the advance of video compression and networking technologies, multipoint video conferencing is becoming more and more popular [1]–[10]. In multipoint video conferencing, the conference participants are connected to a multipoint control unit (MCU) which receives video signals from several different participants, and then processes and transmits them to all participants. Multipoint video conferencing can be either "switched presence" or the "continuous presence." A typical switched presence MCU [11], [12] permits the selection of a particular video signal from one participant for transmission to all participants. Switched presence MCU generally does not require the processing of video signals to generate a combined video signal and therefore is relatively simple to implement. However, only one participant can be seen at a given time. Continuous presence mode [1]–[4] consists of a video combiner which combines the multiple coded video bitstreams from the conference participants into a single coded video bitstream and sends it back to the conference participants for decoding and presentation. Each participant in a continuous presence conference can then view one or more of the other participants in real time.

There are two possible approaches to implement a video combiner for continuous presence multipoint video conferencing. The first approach is coded-domain combining [1], [2]. This technique modifies the headers of individual coded bitstreams from conference participants, multiplexes bitstreams, and generates new headers to produce a combined video bitstream conforming to the video coding standard. For example, a QCIF combiner was proposed in [1] which concatenates four H.261 bitstreams coded in QCIF picture format (176×144 pixels) into a single H.261 bitstream coded in CIF picture format (352×288 pixels). Since the coded-domain combiner only needs to perform the multiplexing and header-modification functions in concatenating the video bitstreams, the implementation complexity is very low. Also, since it does not need to decode and re-encode the video sequence, it does not introduce any quality degradation. However, the coded-domain combiner requires an asymmetric network channel between the participant and the MCU because the video bitrate from the MCU to the participants is four times that from the participants to the MCU. This asymmetric requirement is not supported by most networks.

The second approach to video combining is based on the transcoding technique [3], [4]. This type of video combiner decodes each coded video bitstream, combines the decoded video in the pixel domain, and re-encodes the combined video at the transmission channel rate. Transcoding is a very practical approach for video combining in multipoint video conferencing over a symmetrical wide-area network. However, the computational complexity is inevitably increased since the individual video bitstream needs to be decoded and the combined video signal needs to be encoded. This intrinsic double-encoding process will also introduce additional degradation.

Manuscript received February 13, 2002; revised May 15, 2002. This work was supported by the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, Hong Kong Polytechnic University. The associate editor coordinating the review of this paper and approving it for publication was Dr. Hong-Yuan Mark Liao.

The authors are with the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: enwcsiu@polyu.edu.hk; enktfung@eie.polyu.edu.hk; enylchan@polyu.edu.hk).

Digital Object Identifier 10.1109/TMM.2003.819761

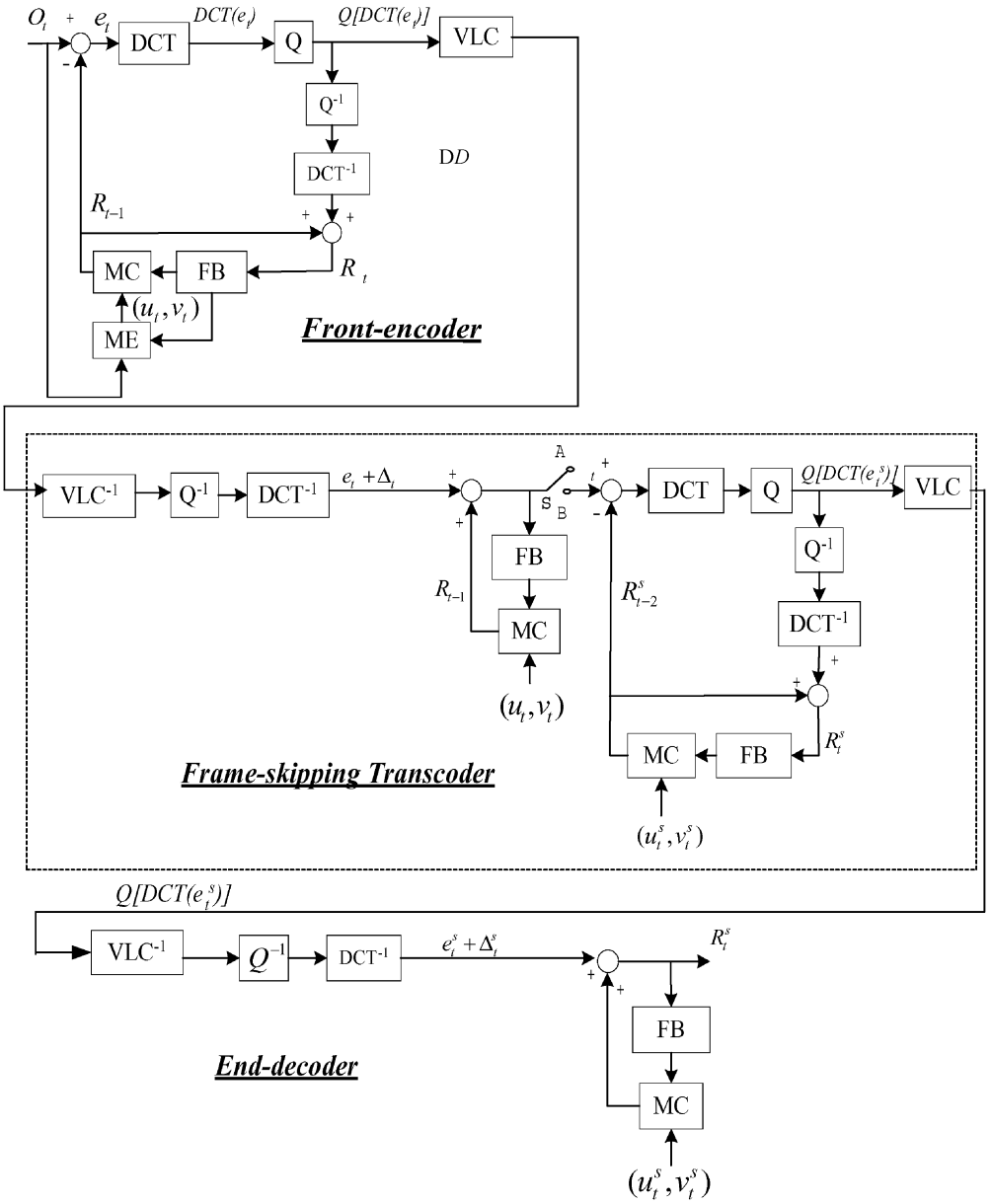

Fig. 1. Frame-skipping transcoder in pixel-domain.

In recent years, discrete cosine transform (DCT) domain transcoding was introduced [13]–[15], under which the incoming video bitstream is partially decoded to form the DCT coefficients and downscaled by the requantization of the DCT coefficients. Since DCT-domain transcoding is carried out in the coded domain where complete decoding and re-encoding are not required, the processing complexity is significantly reduced. The problem, however, with this approach is that the requantization error will accumulate frame by frame, and prediction memory mismatch at the decoder will cause poor video quality. This phenomenon is called "drift" degradation which often results in an unacceptable video quality. Several techniques for eliminating "drift" degradation have been proposed [16]–[18]. In [16], [17], the requantization error is stored in a frame buffer and is fed back to the requantizer to compensate for the requantization error introduced in the previous frame. A simple drift-free MPEG-2 video transcoder has also been proposed [18], in which various modes of motion compensation defined in MPEG-2 are implemented in the DCT domain. Thus, the DCT-domain approach is very attractive for video combining in multipoint video conferencing.

However, it is impossible to achieve the desired output bitrate by performing only the requantization. In other words, if the bandwidth of the outgoing channel is not enough to

allocate bits with requantization, frame skipping is a good strategy for controlling the bitrate and maintaining the picture quality within an acceptable level. It is difficult to perform frame skipping in the DCT-domain since the prediction error of each frame is computed from its immediate past frames. This means that the incoming quantized DCT coefficients of the residual signal are no longer valid because they refer to the frames which have been dropped. All motion vectors and predicted errors must be computed again for the nonskipped frame which references the previous nonskipped frame. This can create an undesirable complexity in real time applications as well as introduce re-encoding errors. In this paper, we provide a computationally efficient solution to perform frame skipping in a transcoder, mainly in the DCT-domain, to avoid the complexity and the re-encoding error arising from pixel-domain transcoding. A new system architecture for continuous presence multipoint videoconferencing based on the proposed low-complexity and high-quality frame-skipping transcoder is developed. Simulation results are presented to show the performance improvement realized by our proposed architecture.

The organization of this paper is as follows. Section II of this paper presents an in-depth study of the re-encoding error in the frame-skipping transcoder. The proposed frame-skipping transcoder is then described in Section III. Section IV presents the system architecture of the proposed continuous presence in a multipoint videoconference. Simulation results are presented in Section V. Finally, some conclusive remarks are provided in Section VI.

II. Frame-Skipping Transcoding

Fig. 1 shows the structure of a frame-skipping transcoder in pixel-domain [19]–[21]. At the front encoder, the motion vector, mv_t, for a macroblock with $N \times N$ pixels in frame O_t, the current frame, is computed [22]–[26] by searching for the best matched macroblock within a search window S in the previously reconstructed frame, R_{t-1}, and it is obtained as follows:

$$mv_t = (u_t, v_t) = \arg \min_{(m,n)\in S} SAD(m,n) \tag{1}$$

$$SAD(m,n) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} |O_t(i,j) - R_{t-1}(i+m, j+n)| \tag{2}$$

where m and n are the horizontal and vertical components of the displacement of a matching macroblock, $O_t(i,j)$ and $R_{t-1}(i,j)$ represent a pixel in O_t and R_{t-1}, respectively.

In transcoding the compressed video bitstream, the output bitrate is lower than the input bitrate. As a result, the outgoing frame rate in the transcoder by cascading a decoder and an encoder, as depicted in Fig. 1, is usually much lower than the incoming frame rate. Hence switch S is used to control the desired frame rate of the transcoder. Table I summaries the operating modes of the frame-skipping transcoder.

Assume that frame $t-1$, R_{t-1}, is skipped. However, R_{t-1} is required to act as the reference frame for the reconstruction of frame t, R_t, such that

$$R_t(i,j) = R_{t-1}(i+u_t, j+v_t) + e_t(i,j) + \Delta_t(i,j) \tag{3}$$

TABLE I
Switch Position for Different Modes of Frame Skipping

Frame skipping mode	*S* Position
Skipped frame	*A*
Non-skipped frame	*B*

where $\Delta_t(i,j)$ represents the reconstruction error of the current frame in the front-encoder due to the quantization, and $e_t(i,j)$ is the residual signal between the current frame and the motion-compensated frame:

$$e_t(i,j) = O_t(i,j) - R_{t-1}(i+u_t, j+v_t). \tag{4}$$

Substituting (4) into (3), we obtain the expression for R_t:

$$R_t(i,j) = O_t(i,j) + \Delta_t(i,j). \tag{5}$$

In the transcoder, an optimized motion vector for the outgoing bitstream can be obtained by applying the motion estimation such that

$$(u_t^s, v_t^s) = \arg \min_{(m,n)\in S} SAD^s(m,n) \tag{6}$$

$$SAD^s(m,n) = \sum_{i}^{M}\sum_{j}^{N} |R_t(i,j) - R_{t-2}^s(i+m, j+n)| \tag{7}$$

where $R_{t-2}^s(i,j)$ denotes a reconstructed pixel in the previous nonskipped reference frame. The superscript "s" is used to denote the symbol after performing the frame-skipping transcoder. Although the optimized motion vector can be obtained by a new motion estimation, it is not desirable because of its high computational complexity. Reuse of the incoming motion vectors has been widely accepted because it is considered to be almost as good as performing a new full-scale motion estimation and was assumed in many transcoder architectures [20], [21]. Thus, we assume that the new motion vector is (u_t^s, v_t^s). Hence, the reconstructed pixel in the current frame after the end-decoder is

$$R_t^s(i,j) = R_{t-2}^s(i+u_t^s, j+v_t^s) + e_t^s(i,j) + \Delta_t^s(i,j) \tag{8}$$

where $e_t^s(i,j) = R_t(i,j) - R_{t-2}^s(i+u_t^s, j+v_t^s)$ and represents the requanization error due to the re-encoding in the transcoder. Hence, we have

$$R_t^s(i,j) = R_t(i,j) + \Delta_t^s(i,j). \tag{9}$$

This equation implies that the reconstructed quality of the nonskipped frame deviates from the input sequence to the transcoder, R_t. An additional error, Δ_t^s, is introduced. Re-encoding of the current frame involves a recomputation of the residual signal between the current frame and the nonskipped reference frame. Note that frame $t-2$ acts as the reference instead of frame $t-1$, since frame $t-1$ does not exist after frame skipping. The newly quantized DCT-domain data are then recomputed by means of the DCT and quantization processes. This re-encoding procedure leads to error Δ_t^s. The effect of the re-encoding error is depicted in Fig. 2 where the

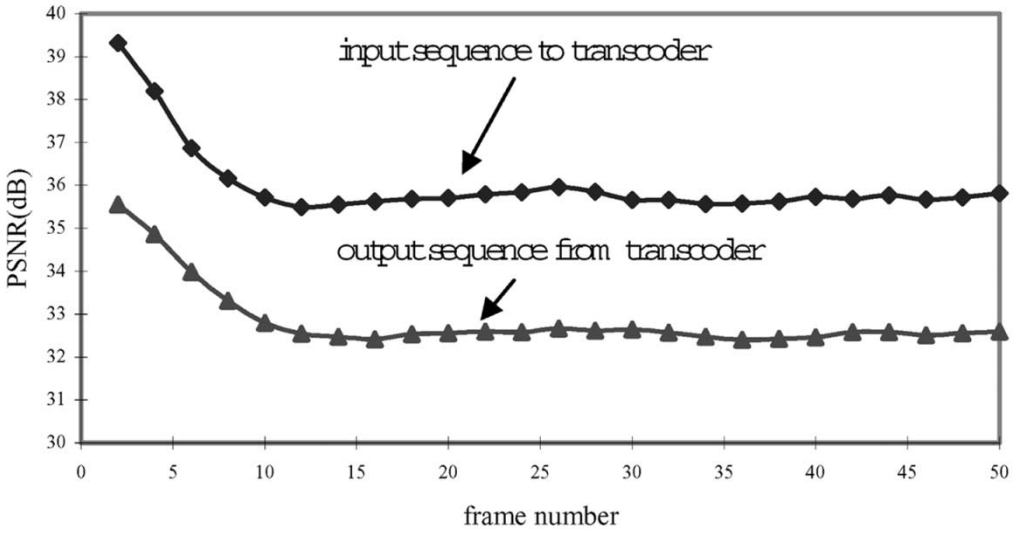

Fig. 2. Quality degradation of frame-skipping transcoder for the "Salesman" sequence.

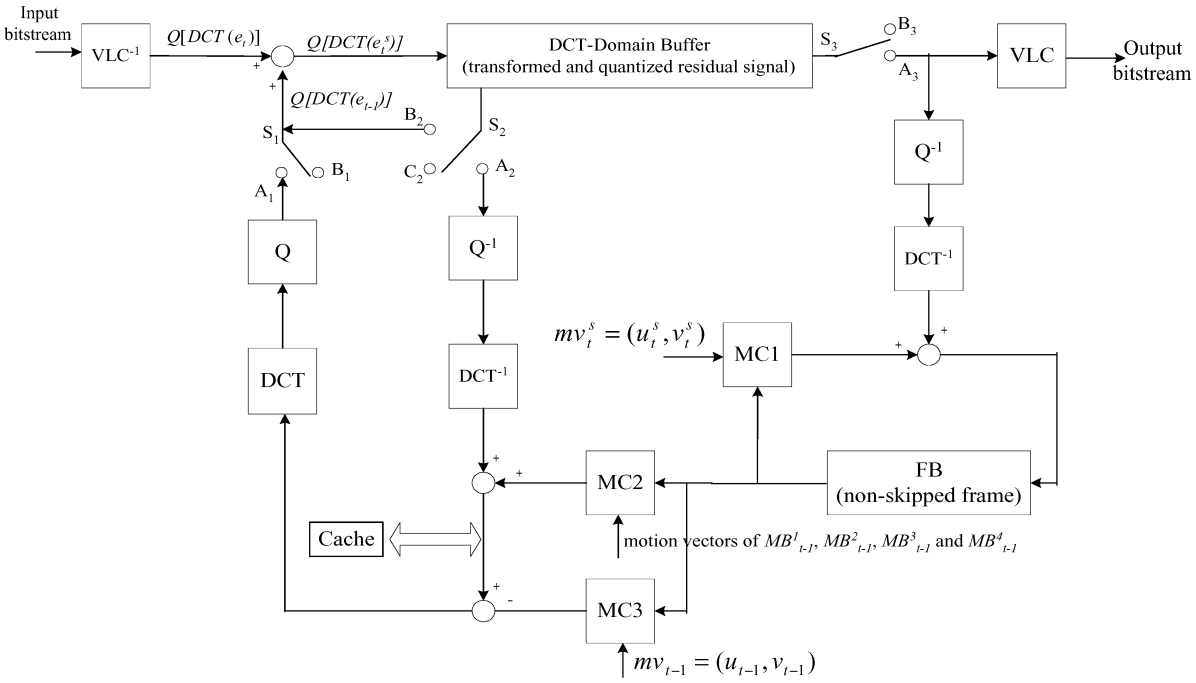

Fig. 3. Proposed frame-skipping transcoder.

"Salesman" sequence was transcoded at half of the incoming frame-rate. In Fig. 2, the peak signal-to-noise ratio (PSNR) of the frame-skipping pictures is plotted to compare with that of the same pictures directly using a decoder without a transcoder. This figure shows that the re-encoding error leads to a drop in picture quality of about 3.5 dB on average, which is a significant degradation. Details on the simulation environment and coding parameters used in the simulation are given in Section V.

III. Low-Complexity Frame-Skipping for High Performance Video Transcoding

Fig. 3 shows the architecture of the proposed transcoder. 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, $Q[DCT(e_t)]$. Each macroblock is then manipulated independently. Two switches, $S1$ and $S2$, are employed to update the DCT-domain buffer for the transformed and quantized residual signal depending on the coding mode originally used at the front encoder for the current macroblock being processed. The switch positions for different coding modes are shown in Table II. When the macroblock is not motion compensated, the previous residual signal in the DCT-domain is directly fed back from the DCT-domain buffer to the adder, and the sum of the input residual signal and the previous residual signal in the DCT-domain is updated in the buffer. Note that all operations are performed in the DCT-domain, thus the complexity of the

frame-skipping transcoder is reduced. Also, quality degradation of the transcoder introduced by Δ_t^s is avoided. When the motion compensation mode is used, modules for motion compensation, DCT, inverse DCT, quantization, and inverse quantization are used to update the DCT-domain buffer. The advantages of this DCT-domain buffer arrangement and the details of our method are described in Sections III-A–C. Note that switch $S3$ is used to control the frame rate and refresh the frame buffer (FB) which stores the current nonskipped frame. The current nonskipped frame is obtained by adding the decoded residual error to the motion-compensated frame which is computed through the motion compensation(MC1) of the previous nonskipped frame in FB. Table III shows the frame-skipping modes of our proposed transcoder.

TABLE II
DIFFERENT CODING MODES FOR SWITCHES S1 AND S2

Coding mode	*S1* Position	*S2* Position
No MC	B_1	B_2
MC	A_1	A_2

TABLE III
SWITCH POSITIONS FOR DIFFERENT FRAME-SKIPPING MODES OF OUR PROPOSED TRANSCODER

Frame-skipping mode	*S3* Position
Skipped frame	B_3
Non-skipped frame	A_3

A. Direct Summation of DCT Coefficients for Macroblock Without Motion Compensation (Non-MC Macroblock)

For non-MC macroblocks, the direct summation of DCT coefficients is employed such that the DCT transform pair and motion compensation operations are not needed. For typical video conferencing, most of the video signals are included in non-MC macroblocks and hence the complexity reduction realized by using the direct summation is significant. In Fig. 4, the situation for which one frame is dropped is illustrated. We assume that MB_t represents the current macroblock and MB_{t-1} represents the best matching macroblock to MB_t. Since MB_t is a non-MC macroblock, the spatial position of MB_{t-1} is the same as that of MB_t, and MB_{t-2} represents the best matching macroblock to MB_{t-1}. Since R_{t-1} is dropped, for MB_t, we need to compute a motion vector, (u_t^s, V_t^s), and the prediction error in the DCT-domain, $Q[DCT(e_t^s)]$, by using R_{t-2} as a reference. Since the motion vector in MB_t is zero, then

$$(u_t^s, v_t^s) = (u_{t-1}, v_{t-1}). \tag{10}$$

Re-encoding can lead to an additional error; however, this can be avoided if $Q[DCT(e_t^s)]$ is computed in the DCT-domain. In Fig. 4, the pixels in MB_t can be reconstructed by performing the inverse quantization and inverse DCT of $Q[DCT(e_t)]$ and summing this residual signal to pixels in MB_{t-1} which can be similarly reconstructed by performing inverse quantization and inverse DCT of $Q[DCT(e_{t-1})]$ and summing this residual signal to pixels in the corresponding MB_{t-2}. The reconstructed macroblocks MB_t and MB_{t-1} are given by

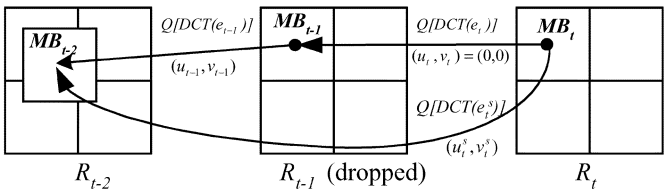

Fig. 4. Residual signal recomputation of frame skipping for non-MC macroblocks.

$$MB_t = MB_{t-1} + e_t + \Delta_t \tag{11}$$

and

$$MB_{t-1} = MB_{t-2} + e_{t-1} + \Delta_{t-1}. \tag{12}$$

Note that Δ_t and Δ_{t-1} represent the quantization errors of MB_t and MB_{t-1} in the front-encoder. Substituting (12) into (11), (13) is obtained:

$$e_t^s = (e_t + \Delta_t) + (e_{t-1} + \Delta_{t-1}) \tag{13}$$

where $e_t^s = MB_t - MB_{t-2}$. This is the prediction error between the current macroblock and its corresponding reference macroblock. By applying the DCT for e_t^s and taking into account the linearity of DCT, we obtain the expression of e_t^s in the DCT-domain:

$$DCT(e_t^s) = DCT(e_t + \Delta_t) + DCT(e_{t-1} + \Delta_{t-1}). \tag{14}$$

Then the newly quantized DCT coefficients of the prediction error are given by

$$\begin{aligned} Q[DCT(e_t^s)] &= Q[DCT(e_t + \Delta_t) + DCT(e_{t-1} + \Delta_{t-1})] \\ &= \text{rounding} \\ &\quad \times \left[\frac{DCT(e_t + \Delta_t) + DCT(e_{t-1} + \Delta_{t-1})}{q}\right] \end{aligned} \tag{15}$$

where q is the quantization step-size in the front encoder. Note that, in general, quantization is not a linear operation because of the integer truncation. However, Δ_t and Δ_{t-1} are introduced due to the quantization in the front encoder such that

$$\Delta_t = DCT^{-1}\left[q \times \text{rounding}\left(\frac{DCT(e_t)}{q}\right)\right] - e_t \tag{16}$$

and

$$\Delta_{t-1} = DCT^{-1}\left[q \times \text{rounding}\left(\frac{DCT(e_{t-1})}{q}\right)\right] - e_{t-1}. \tag{17}$$

From (16) and (17), we have

$$DCT(e_t + \Delta_t) = q \times \text{rounding}\left(\frac{DCT(e_t)}{q}\right) \tag{18}$$

and

$$DCT(e_{t-1} + \Delta_{t-1}) = q \times \text{rounding}\left(\frac{DCT(e_{t-1})}{q}\right). \tag{19}$$

Equations (18) and (19) show that $DCT(e_t + \Delta_t)$ and $DCT(e_{t-1} + \Delta_{t-1})$ are divisible by q which is the quantization step-size provided by the front encoder. Thus, if the quantization step-size is not altered in the transcoder, (15) can be written as

$$\begin{aligned} Q[DCT(e_t^s)] &= \text{rounding}\left[\frac{DCT(e_t+\Delta_t)}{q}\right] \\ &\quad + \text{rounding}\left[\frac{DCT(e_{t-1}+\Delta_{t-1})}{q}\right] \\ &= Q[DCT(e_t+\Delta_t)] \\ &\quad + Q[DCT(e_{t-1}+\Delta_{t-1})]. \end{aligned} \tag{20}$$

Rewriting (18) and (19), we have

$$Q[DCT(e_t)] = \frac{DCT(e_t+\Delta_t)}{q} \tag{21}$$

and

$$Q[DCT(e_{t-1})] = \frac{DCT(e_{t-1}+\Delta_{t-1})}{q}. \tag{22}$$

Again, $DCT(e_t+\Delta_t)$ and $DCT(e_{t-1}+\Delta_{t-1})$ are divisible by q, (21) and (22) can be written as

$$\begin{aligned} Q[DCT(e_t)] &= \text{rounding}\left[\frac{DCT(e_t+\Delta_t)}{q}\right] \\ &= Q[DCT(e_t+\Delta_t)] \end{aligned} \tag{23}$$

and

$$\begin{aligned} Q[DCT(e_{t-1})] &= \text{rounding}\left[\frac{DCT(e_{t-1}+\Delta_{t-1})}{q}\right] \\ &= Q[DCT(e_{t-1}+\Delta_{t-1})]. \end{aligned} \tag{24}$$

From (20), (23), and (24), we obtain the final expression of the prediction error in the quantized DCT-domain by using R_{t-2} as a reference:

$$Q[DCT(e_t^s)] = Q[DCT(e_t)] + Q[DCT(e_{t-1})] \tag{25}$$

Equation (25) implies that coefficients $Q[DCT(e_t^s)]$ of the newly quantized DCT can be computed in the DCT-domain by summing directly the quantized DCT coefficients between the data in the DCT-domain buffer and the incoming DCT coefficients, whilst the updated DCT coefficients are stored in the DCT-domain buffer, as depicted in Fig. 3, when switches S_1 and S_2 are connected to B_1 and B_2 respectively. Since it is not necessary to perform motion compensation, DCT, quantization, inverse DCT, and inverse quantization, the complexity is reduced. Furthermore, since requantization is not necessary for non-MC macroblock, the quality degradation of the transcoder introduced by Δ_t^s is also avoided. Fig. 5 shows the distribution of the coding modes of a typical sequence, the "salesman." It is clear that over 95% of the macroblocks are coded without motion compensation. By using the direct summation of DCT coefficients for non-MC macroblocks, the computational complexity involved in processing these macroblocks can be reduced significantly and the additional re-encoding error can be avoided.

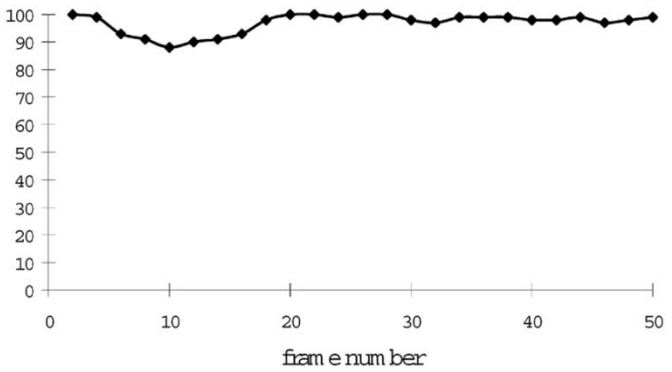

Fig. 5. Distribution of coding modes for "salesman" sequence.

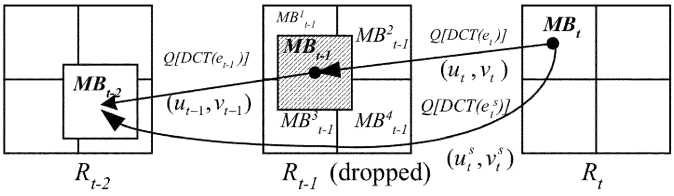

Fig. 6. Residual signal recomputation of frame-skipping for MC macroblocks.

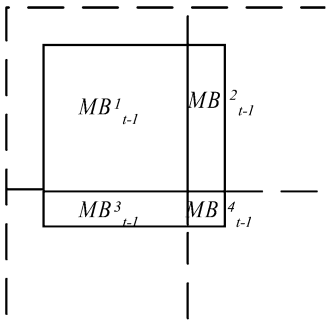

Fig. 7. Composition of MB_{t-1}.

B. DCT-Domain Buffer Updating for Motion-Compensated Macroblock (MC Macroblock)

For MC macroblocks, direct summation cannot be employed when it is not well aligned on a macroblock boundary, as shown in Fig. 6. In other words, $Q[DCT(e_{t-1})]$ of MB_{t-1}, which is defined as the prediction error between MB_{t-1} and MB_{t-2} in Fig. 6, is not available from the incoming bitstream. It is possible to use the motion vectors and quantized DCT coefficients of four neighboring macroblocks, MB_{t-1}, MB^1_{t-1}, MB^2_{t-1}, MB^3_{t-1}, and MB^4_{t-1}, to compute $Q[DCT(e_{t-1})]$. First, we need to find the prediction error, e_{t-1}. Actually, e_{t-1} is equal to the difference of the reconstructed pixel in MB_{t-1} and the corresponding pixel in MB_{t-2} of the previous nonskipped frame stored in the frame buffer (FB) as depicted in Fig. 3. In order to obtain e_{t-1}, the motion vector of MB_{t-1}, (u_{t-1}, v_{t-1}) as depicted in Fig. 6, with reference to the best matching macroblock, MB_{t-2}, in R_{t-2} is required. Again, MB_{t-1} is not on a macroblock boundary; it is possible to use the bilinear interpolation from motion vectors mv_{t-1,MB_1}, mv_{t-1,MB_2}, mv_{t-1,MB_3}, and mv_{t-1,MB_4} of the four neighboring macroblocks, MB_{t-1}, MB^1_{t-1}, MB^2_{t-1}, MB^3_{t-1}, and MB^4_{t-1}, to come up with an approximation of (u_{t-1}, v_{t-1}) [27]. However, the bilinear interpolation of motion vectors has

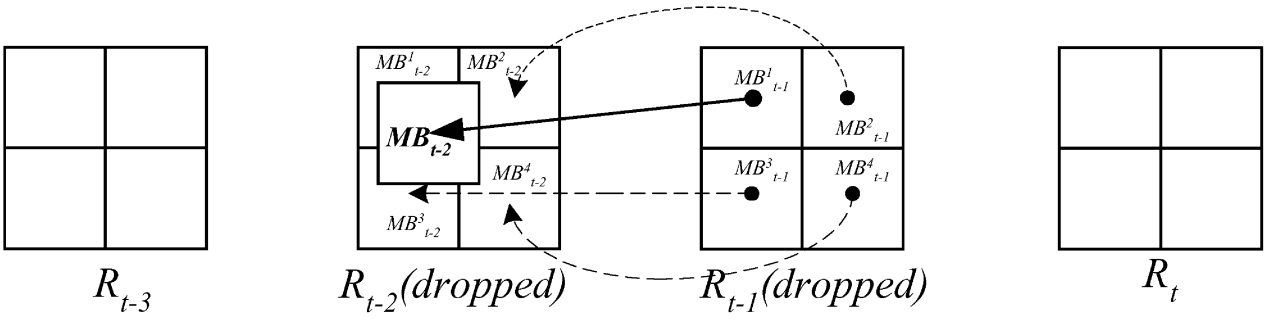

Fig. 8. Multiple frame skipping of our proposed transcoder.

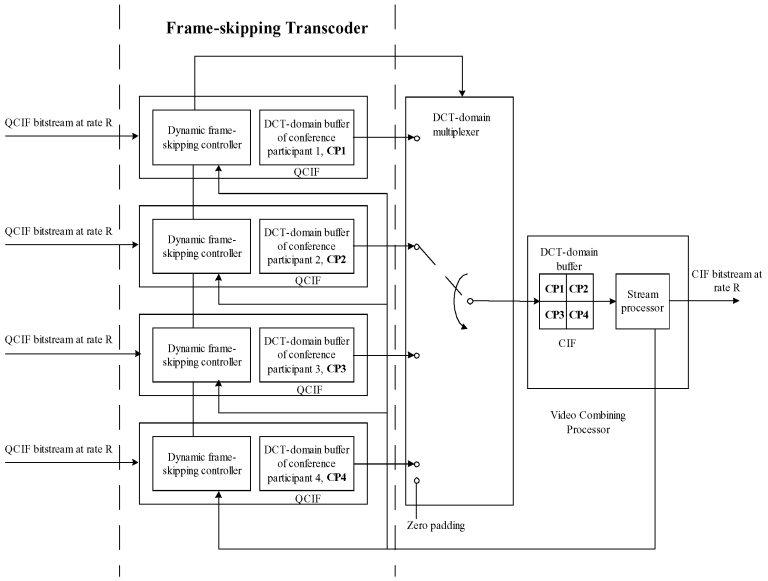

Fig. 9. System architecture for video combiner using the frame-skipping transcoder.

TABLE IV
AVERAGE PSNR WITH RESPECT TO THE ORIGINAL SEQUENCE OF OUR PROPOSED TRANSCODER AS COMPARED WITH PIXEL-DOMAIN TRANSCODER BY FORCING MOTION VECTOR TO ZERO IN THE FRONT-ENCODER. THE FRAME-RATE OF INCOMING BITSTREAM IS 30 FRAMES/S WHICH ARE THEN TRANSCODED TO 15 FRAMES/S

Sequences	Input bitrate	pixel-domain transcoder Average PSNR	Our proposed transcoder Average PSNR	Input sequence to the transcoder Average PSNR
Salesman	64k	33.35	35.12	35.12
	128k	36.51	38.33	38.33
News	64k	33.73	35.27	35.27
	128k	36.83	38.79	38.79
Hall	64k	36.65	36.86	36.86
	128k	38.71	39.09	39.09

several drawbacks. For example, it leads to inaccuracy of the resultant motion vector because the area covered by the four macroblocks may be too divergent and too large to be described by a single motion vector. Thus, the forward dominant vector selection (FDVS) method is used to select one dominant motion vector from four neighboring macroblocks [20], [21]. A dominant motion vector is defined as the motion vector carried by a dominant macroblock. The dominant macroblock is the macroblock that has the largest overlapped segment with MB_{t-1}.

Hence, inverse quantization and inverse DCT of the quantized DCT coefficients of MB^1_{t-1}, MB^2_{t-1}, MB^3_{t-1}, and MB^4_{t-1} are performed to obtain their corresponding prediction errors in the pixel-domain. MB_{t-1} is composed of four components as shown in Fig. 7. Fig. 3 depicts that each segment of the reconstructed pixels in MB_{t-1} can be obtained by summing its prediction errors and its motion-compensated segment of the previous nonskipped frame stored in FB, in which this motion-compensated segment is computed through MC2. e_{t-1} can be obtained by computing the difference

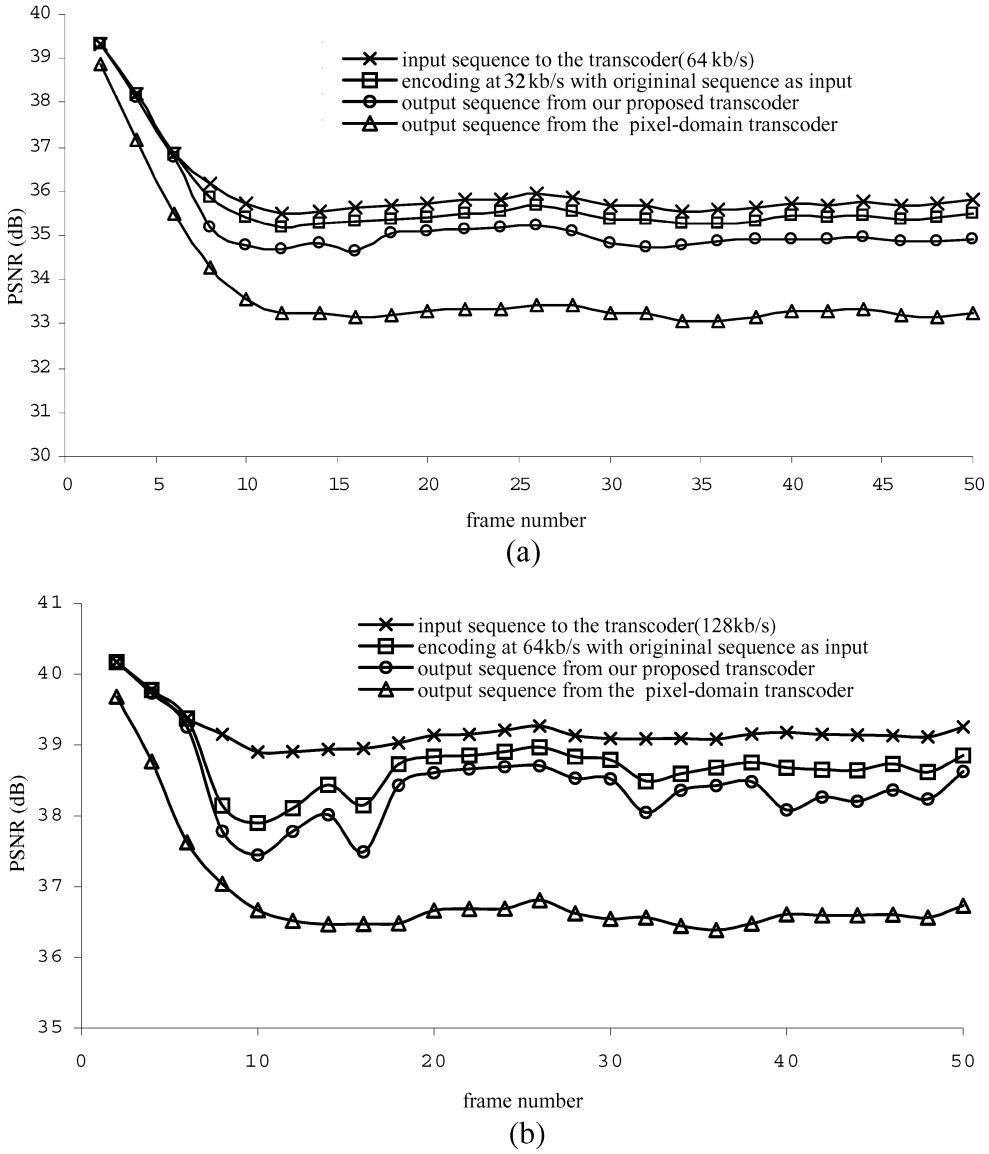

Fig. 10. Performance of the proposed transcoder of "Salesman" sequence encoded at (a) 64 kb/s with 30 frames/s, and then transcoded to 32 kb/s with 15 frames/s; (b) 128 Kb/s with 30 frames/s, which are then transcoded to 64 kb/s with 15 frames/s.

between the reconstructed pixels in MB_{t-1} and their corresponding motion-compensated pixels in MB_{t-2} which is obtained through MC3, and it is then transformed and quantized to $Q[DCT(e_{t-1})]$. Fig. 6 shows that the newly quantized DCT coefficients of $MB_t, Q[DCT(e_t^s)]$, can be computed by summing the newly computed $Q[DCT(e_{t-1})]$ and the incoming $Q[DCT(e_t)]$ and it is quite similar to that of the non-MC macroblock as mentioned in (20), except for the formation of $Q[DCT(e_{t-1})]$. For non-MC macroblocks, $Q[DCT(e_{t-1})]$ is available from the incoming bitstreams. On the other hand, requantization is performed for the formation of $Q[DCT(e_{t-1})]$ in MC macroblocks, which will introduce additional re-encoding error Δ_{t-1}^s such that the reconstructed frame after the end-encoder becomes

$$R_t^s = R_t + \Delta_{t-1}^s. \tag{26}$$

Note that, as compared with Δ_t^s in (9), Δ_{t-1}^s is the re-encoding error due to frame $t-1$ instead of frame t.

In order to reduce the implementation complexity of the MC macroblock, a cache subsystem is added to our proposed transcoder, as depicted in Fig. 3. Since motion compensation of multiple macroblocks may require the same pixel data, a cache subsystem is implemented to reduce redundant inverse quantization, inverse DCT, and motion compensation computations to speed up the performance for MC macroblocks in the re-encoding process. The cache subsystem is composed of a frame buffer and a control unit. A control unit is used to detect the new region referenced by MC macroblocks and provide the information to the frame buffer to store the corresponding reconstructed pixels in R_{t-1}.

C. Multiple Frame-Skipping in our Proposed Transcoder

Another advantage of the proposed frame-skipping transcoder is that when multiple frames are dropped, it can be processed in the forward order, thus eliminating the multiple DCT-domain buffers that are needed to store the

TABLE V
PERFORMANCE OF THE PROPOSED TRANSCODER. THE FRAME-RATE OF INCOMING BITSTREAM IS 30 FRAMES/S WHICH ARE THEN TRANSCODED TO 15 FRAMES/S

Sequences	Input bitrate	pixel-domain transcoder			Our proposed transcoder using BiVS			Our proposed transcoder using FDVS		
		MC region	Non-MC region	All region	MC region	Non-MC region	All region	MC region	Non-MC region	All region
Salesman	64k	30.25	34.31	33.77	31.02	36.24	35.41	31.45	36.32	35.47
	128k	33.58	37.14	36.85	33.73	39.20	38.51	34.11	39.28	38.62
News	64k	30.44	34.66	34.03	30.91	36.38	35.43	31.37	36.49	35.55
	128k	34.07	37.47	37.13	34.08	39.69	38.86	34.56	39.79	38.97
Hall	64k	36.14	37.07	36.93	36.34	37.66	36.88	36.86	37.79	37.06
	128k	38.3	39.43	38.94	37.85	40.11	39.12	38.36	40.23	39.27

TABLE VI
PERFORMANCE OF THE PROPOSED TRANSCODER. THE FRAME-RATE OF INCOMING BIT STREAM IS 30 FRAMES/S WHICH ARE THEN TRANSCODED TO 10 FRAMES/S

Sequences	Input bitrate	pixel-domain transcoder			Our proposed transcoder using BiVS			Our proposed transcoder using FDVS		
		MC region	Non-MC region	All region	MC region	Non-MC region	All region	MC region	Non-MC region	All region
Salesman	64k	29.85	34.12	33.45	30.41	35.78	35.20	30.89	35.89	35.30
	128k	33.59	37.03	36.70	33.20	38.86	38.34	33.68	38.95	38.44
News	64k	30.04	34.49	33.69	30.27	36.09	35.13	30.82	36.23	35.3
	128k	34.11	37.34	36.93	34.71	30.18	38.47	35.23	39.36	38.62
Hall	64k	34.2	36.37	35.96	36.15	37.56	36.78	36.74	37.75	37.01
	128k	38.2	38.81	38.52	37.60	39.51	38.67	38.21	39.72	38.91

TABLE VII
SPEED-UP RATIO OF THE PROPOSED TRANSCODER AS COMPARED WITH THE PIXEL-DOMAIN TRANSCODER. THE FRAME-RATE OF INCOMING BIT STREAM IS 30 FRAMES/S WHICH ARE THEN TRANSCODED TO 15 FRAMES/S

Sequences	Input bitrate	Speed-up ratio
Salesman	64k	6.75
	128k	7.64
News	64k	6.08
	128k	6.68
Hall	64k	3.57
	128k	3.96

incoming quantized DCT coefficients of all dropped frames. Fig. 8 shows a scenario in which two frames are dropped. When R_{t-2} is dropped, we store the DCT coefficients of its prediction errors in the DCT-domain buffer. The stored DCT coefficients of prediction errors will be used to update the DCT coefficients of prediction errors for the next dropped frame. This means that when R_{t-1} is dropped, our proposed scheme updates the DCT coefficients of prediction errors for each macroblock according to its coding mode. For example, macroblocks, MB^2_{t-1}, MB^3_{t-1}, and MB^4_{t-1}, in R_{t-1} are coded without motion compensation. From (25), the DCT coefficients of prediction errors in the DCT-domain buffer are added to the corresponding incoming prediction errors of the macroblock in R_{t-1}. The buffer is then updated with the new DCT coefficients. In Fig. 8, MB^1_{t-1} is an MC macroblock. It is necessary to perform the re-encoding of the macroblock pointed by MB^1_{t-1} and then add the corresponding incoming DCT-coefficients to form the updated data in the DCT-domain buffer. By using our proposed scheme, only one DCT-domain buffer is needed for all the dropped frames. The flexibility of multiple frame-skipping provides the fundamental framework for dynamic frame-skipping, which is used in multipoint video conferencing.

IV. DYNAMIC FRAME ALLOCATION FOR VIDEO COMBINING IN MULTIPOINT CONFERENCING

In a multipoint video conferencing system, usually only one or two participants are active at any given time [3]. The active conferees need higher frame rates to produce a better video quality as well as to present a smoother motion. Fig. 9 shows the proposed system architecture for video combiners in multipoint video conferencing. Our approach for video combining is based on frame-skipping transcoding which primarily consists of the DCT-domain approach without the requirement of the asymmetric network channels. Thus, the original video sequence can be encoded by fully utilizing the available channel bandwidth. Up to four QCIF video bitstreams are received by the video combiner from the conference participants. Each QCIF bitstream is processed by our proposed frame-skipping transcoder. The main function of a frame-skipping transcoder is frame-rate reduction. Note that the output frame rates from

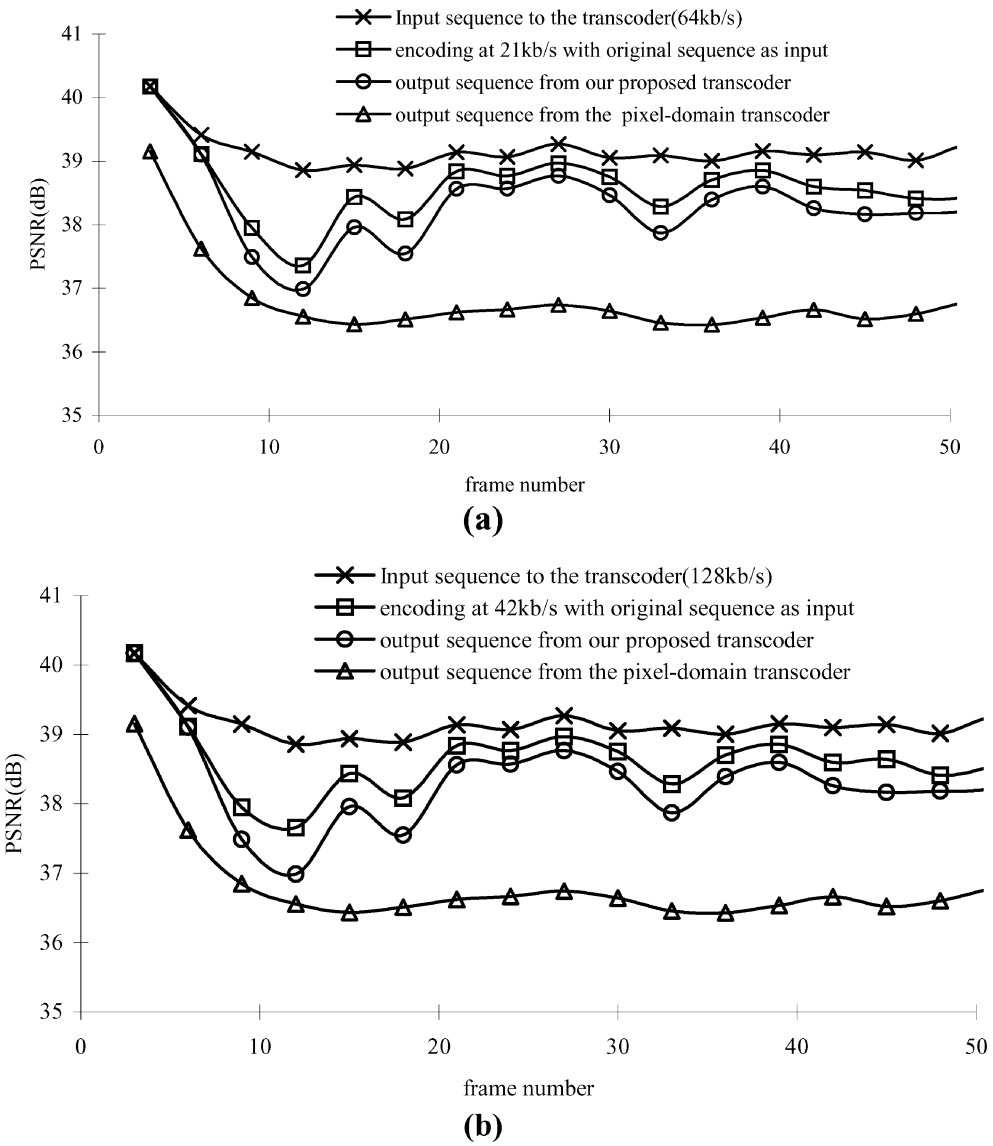

Fig. 11. Performance of the proposed transcoder of "Salesman" sequence encoded at (a) 64 Kb/s with 30 frames/s, and then transcoded to 21 kb/s with 10 frames/s; (b) 128 kb/s with 30 frames/s, which are then transcoded to 42 kb/s with 10 frames/s.

TABLE VIII
SPEED-UP RATIO OF THE PROPOSED TRANSCODER AS COMPARED WITH THE PIXEL-DOMAIN TRANSCODER. THE FRAME-RATE OF INCOMING BIT STREAM IS 30 FRAMES/S WHICH ARE THEN TRANSCODED TO 10 FRAMES/S

Sequences	Input bitrate	Speed-up ratio
Salesman	64k	9.78
	128k	11.82
News	64k	7.68
	128k	8.84
Hall	64k	4.38
	128k	4.69

the four transcoders are not constant, and they are not necessarily equal to one another.

In the transcoder, the frame-skipping controller can dynamically distribute the encoded frames to each subsequence by considering their motion activities so that the quality of the transcoded video can be improved. The frame rate required to transcode a subsequence is highly related to its motion activity [27]–[29]. Thus, it is necessary to regulate the frame rate of each transcoder according to the motion activity in the current frame (MA_t) of the subsequence. To obtain a quantitative measure for MA_t, we use the accumulated magnitudes of all of the motion vectors estimated for the macroblocks in the current frame [3], [27], i.e.,

$$MA_t = \sum_{i=1}^{N} |(u_t^s)_i| + |(v_t^s)_i| \tag{27}$$

where N is the total number of macroblocks in the current frame, and $(u_t^s)_i$ and $(v_t^s)_i$ are the horizontal and vertical components of the motion vector of the ith macroblock, which uses the previous nonskipped frame as a reference.

If the value of MA_t after a nonskipped frame exceeds the predefined threshold, T_{MA}, the incoming frame should be kept. It is interesting to note that T_{MA} is set according to the outgoing

Fig. 12. Encoded frame 194 of the four conferee's videos, which are received by the MCU.

bit rate of the video combiner, but this is not the focus of this paper. By adaptively adjusting the frame rate of each subsequence according to the MA_t, the proposed architecture can allocate more frames for a subsequence with high motion activity and less frames for a subsequence with low motion activity.

Our proposed transcoder updates the quantized DCT coefficients of the current frame in the DCT-domain buffer for each QCIF subsequence. At the beginning of the formation of a combining sequence, a decision is made as to which DCT-domain buffers be included in the new buffer by the dynamic frame-skipping controllers. After the current frame of the subsequence is selected by the multiplexer, the DCT coefficients in the corresponding DCT-domain buffer are copied to the video combining processor's buffer in CIF format. The quantized DCT coefficients of each conference participant only need to be mapped according to Fig. 9. Hence, the DCT-domain buffer of the conference participant 1, *CP1*, is mapped to the first quadrant of the combined picture, the DCT-domain buffer of the conference participant 2, *CP2*, is mapped to the second quadrant, etc. If the current frame of the subsequence is skipped, the corresponding quadrant is filled with zero value. Following the assembly of the new DCT buffer in the video combining processor, the data in the buffer are coded in compressed bitstream by the stream processor.

V. Simulation Results

In this section, we present some simulation results. A series of computer simulations were conducted to evaluate the overall efficiency of the proposed frame-skipping transcoder. The performance of the proposed video combiner for multipoint continuous presence video conferencing is also presented as follows.

A. Performance of the Frame-skipping Transcoder

To evaluate the overall efficiency of the proposed frame-skipping transcoding approach, all test sequences in QCIF (176×144) format were encoded at high bitrate (64 kb/s and 128 kb/s) using a fixed quantization parameter. For the front encoder, the first frame was coded as an intraframe (I-frame), and the remaining frames were encoded as interframes (P-frames). These picture-coding modes were preserved during the transcoding. In the following, the peak signal-to-noise ratio (PSNR) of the transcoded sequence was measured against the original sequence.

The first experiment is used to demonstrate the performance of direct summation of DCT coefficients for non-MC macroblocks. The motion vector was set to zero in the front encoder such that the technique of direct summation of DCT coefficients can be used in all macroblocks for the proposed frame-skipping transcoder. In this case, requantization is not required for the proposed transcoder and the video quality is degraded slightly for the incoming bitstream. The speed-up of our proposed frame-skipping transcoder can be obtained by the following evaluations. Consider a video sequence which is transcoded from 30 frames/s to 15 frames/s. The computational requirement of the proposed transcoder includes mainly the direct summation of all DCT coefficients (29 frames $\times$ 99 macroblocks $\times$ 256 operations for QCIF video). This requires a sum of 734 976 operations. In pixel-domain transcoder, some algorithms have to be used to compute the DCT and IDCT. Reference [30] is a possible choice, which requires 41 operations to compute an 8-length one-dimensional (1-D) DCT/IDCT. Then, the computational requirement of the pixel-domain transcoding approach [20] involves the calculation of inverse quantization (29 frames $\times$ 99 macroblocks $\times$ 256 operations), IDCT (99 frames $\times$ 99 macroblocks $\times$ 4 blocks $\times$ 16 1-D IDCT $\times$ 41 operations), prediction errors (29 frames $\times$ 99 macroblocks $\times$ 256 operations), DCT (14 frames $\times$ 99 macroblocks $\times$ 4 blocks $\times$ 16 1-D DCT $\times$ 41 operations) and requantization (14 frames $\times$ 99 macroblocks $\times$ 256 operations). This requires a sum of 12 995 136 operations. Thus, the speed-up factor of the proposed transcoder as compared with the pixel-domain transcoder is about 17.68.

Note that the computation of DCT, prediction errors and requantization processes are required only for the nonskipped frames in the pixel-domain transcoder. On the other hand, it is not necessary to perform direct summation of all DCT coefficients in our proposed transcoder since some DCT coefficients are zero. Only a few DCT coefficients are needed to be found in practical situation. In fact, the speed-up factor of the proposed transcoder has been found to be 37 as compared to the pixel-domain approach when all trivial operations, such as the multiplication of one, addition of zero, etc. are neglected. Our proposed transcoder has also no PSNR degradation as compared to the input sequence to the transcoder, as shown in Table IV. This is due to the fact that direct summation of DCT coefficients will not lead to re-encoding errors.

In a practical situation, it is not possible to restrict the motion estimation of the front encoder. The PSNR performance of the proposed frame-skipping transcoder using forward dominant vector selection (FDVS) [20], [21] and bilinear interpolation vector selection (BiVS) [27] for composing an outgoing motion vector is shown in Table V and Table VI, in which the frames are temporally dropped by a factor of 1 and 2. It is shown that FDVS is better than BiVS in all cases. Therefore, only FDVS is used in the following discussions. The PSNR performance

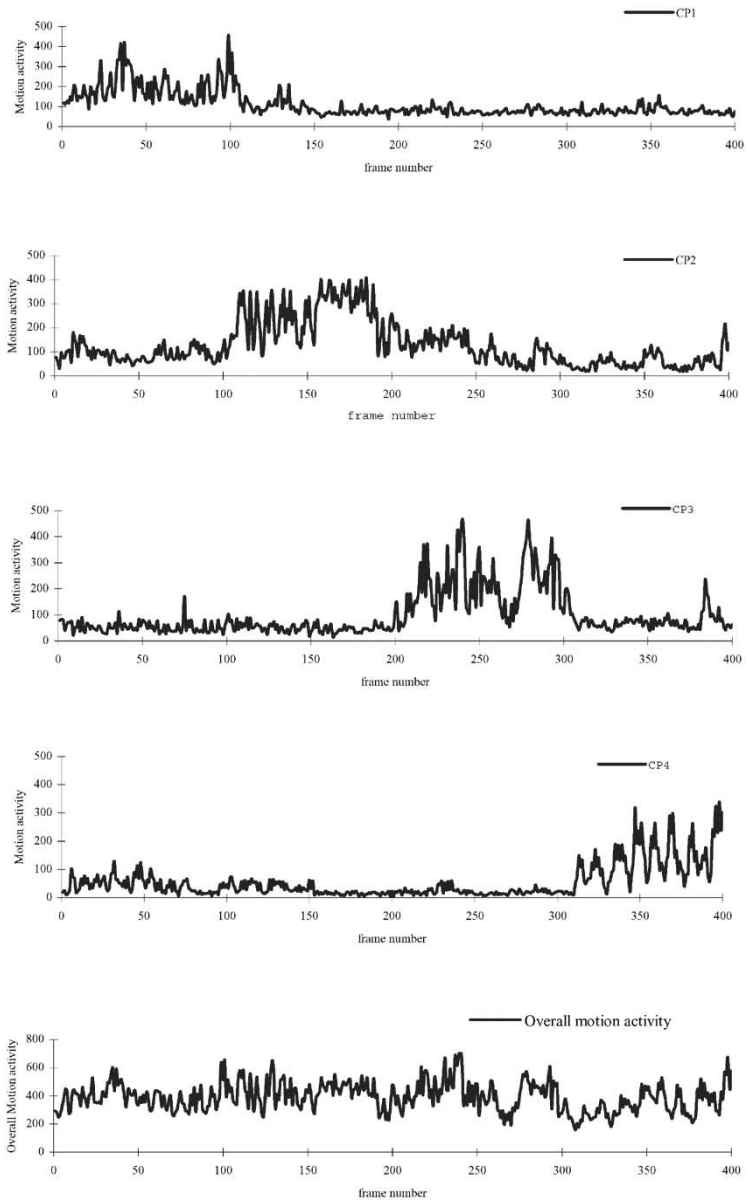

Fig. 13. Motion activity of a multipoint videoconference.

of the proposed frame-skipping transcoder for the "Salesman" sequence is also shown in Fig. 10. At the front encoder, the original test sequence "Salesman" was encoded at 64 kb/s and 128 kb/s in Fig. 10(a) and (b), respectively, and then transcoded into 32 kb/s and 64 kb/s at half of the incoming frame rate. As shown in Fig. 10, the proposed transcoder outperforms the pixel-domain transcoder. The PSNR performance is also close to the encoding at transcoded bitrate with the original sequence as the input since the re-encoding error is significantly reduced. Table VII shows that it has a speed-up of about seven times faster than that of the pixel-domain transcoder for the "Salesman" sequence. This is because the probability of the macroblock coded without motion compensation happens more frequently in typical sequences, and this type of macroblock should not introduce any re-encoding error due to the direct summation of DCT coefficients. Compared with the pixel-domain approach, the re-encoding error introduced is significant since it suffers quantization error in all macroblocks. In Fig. 10, the performance of our proposed transcoder is very close to the input sequence to the transcoder in the first few frames since most of the macroblocks are non-MC macroblocks which result from the technique of direct summation of DCT coefficients to avoid the quality degradation. Since the number of MC-macroblocks become significant after the seventh frame, this results in the quality degradation. Note that re-encoding error is introduced in all macroblocks and is accumulative in the pixel-domain approach. As shown in Table V, the average PSNR performance of the non-MC macroblock in the proposed transcoder is significantly better than that of the pixel-domain transcoder. Thus, we can achieve significant computational savings while maintaining a good video quality for these macroblocks. On the other hand, our proposed transcoder also shows an improvement with respect to the MC macroblock, as depicted in Table V. Furthermore, the cache system in the transcoder can reduce the computational burden of re-encoding the motion-compensated macroblocks. All these advantages

combined gives rise to significant computational saving as well as quality improvement. These demonstrate the effectiveness of the proposed frame-skipping transcoder. The simulation results of other test sequences are summarized in Tables V and VII.

In order to illustrate the effects of the proposed frame-skipping transcoder with multiple frame dropping, Tables VI, VIII and Fig. 11 set forth the results of the frame-skipping transcoding for which the frames are temporally dropped by a factor of two. The results appear to be similar to that of the above. But it is quite apparent that the pixel-domain transcoder gives the worst performance, and our proposed transcoder provides a significant improvement. Also the computational complexity is reduced remarkably.

B. Performance of Continuous Presence Video Conferencing System

For our simulation, we recorded a four-point video conferencing session. Each conferee's video was encoded into a QCIF format at 128 kb/s, as shown in Fig. 12. The four video sequences were transcoded and combined into a CIF format. We then selected segments of the combined sequence to form a 400-frame video sequence in which the first person was most active in the first 100 frames, the second person was most active in the second 100 frames, and so on. Out of 100 frames, the first one is coded as an I frame while all other 99 frames are coded as P-frames. The motion activities for the four conference participants and the combined video sequence are shown in Fig. 13. In Fig. 13, the top four curves correspond to the four participants in the upper left, upper right, lower left and lower right corners, respectively. The bottom curve represents the motion activity of the combined video sequence. Fig. 13 shows that although there were short periods of time when multiple participants were active, only one participant was active during most of the time, while other participants were relatively inactive. The overall motion activity of the combined video sequence is relatively random. This indicates that multipoint video conferencing is a suitable environment for dynamic allocation of the encoding frames to each participant.

In the following discussions, we will analyze the performance of the proposed video combiner for continuous presence multipoint video conferencing system as compared to the pixel-domain combiner with dynamic frame-skipping (PDCOMB-DFS) [27]. Since the active participants need higher frame-rates to produce an acceptable video quality while the inactive participants only need lower frame-rates to produce an acceptable quality, we used dynamic frame allocation in both the proposed video combiner and the PDCOMB-DFS to distribute the encoded frames into each subsequence according to the motion activities. Inevitably, this improvement is made by sacrificing a certain amount of quality of the motion inactive periods. Fig. 14 shows the PSNR performance of the conference participants for different video combining approaches. For example, the participant is most active from frame 0 to frame 100 in Fig. 14(a) (as shown in the motion activity plot in Fig. 13). This active period is transcoded more frequently following the motion activities of the subsequence, therefore the videos displayed on the receiver are smoother. It can be seen from Fig. 14 that the PDCOMB-DFS loses due to the re-encoding process and the proposed video combiner offers

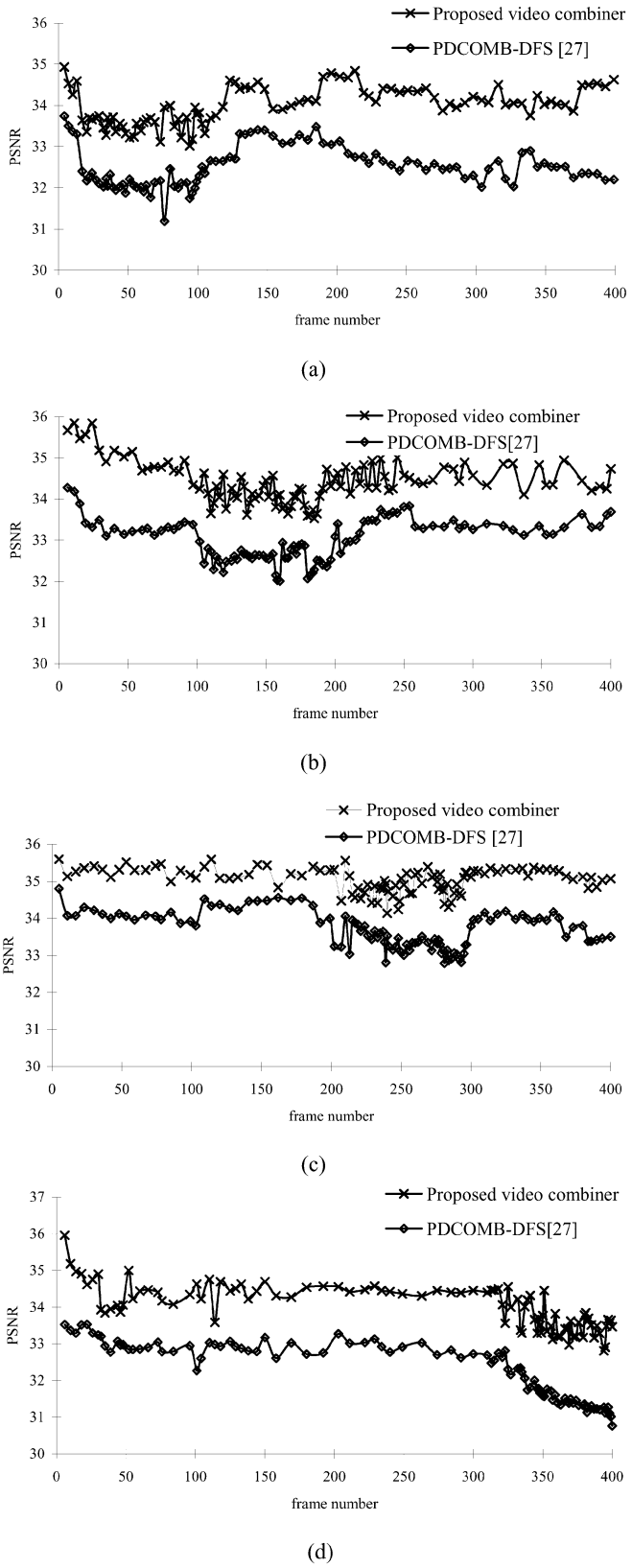

Fig. 14. PSNR performance of a conference participant who is most active (a) between frame 0 and frame 100, (b) between frame 101 and frame 200, (c) between frame 201 and frame 300, and (d) between frame 301 and frame 400.

Fig. 15. Frame 194 of the combined video sequence using (a) PDCOMB-DFS [27]. (b) Our video combiner using the proposed frame-skipping transcoder. The active conference participant is at the upper right corner.

a much better quality as compared to the PDCOMB-DFS. The gain can be as high as 1.5–2.0 dB for both the active and nonactive periods due to the high efficiency of our proposed frame-skipping transcoder. A frame (194th frame) in the combined sequence is shown in Fig. 15. It can be seen that the video quality of the active participant (at the upper right corner) with the proposed video combiner is much better than that with the PDCOMB-DFS. In Fig. 14(d), we observe that from frame 300 to frame 400, the PSNR of the PDCOMB-DFS drops significantly. This is because each nonskipped frame is used as a reference frame of the following nonskipped frame; quality degradation propagates to later frames in a cumulative manner. However, our proposed video combiner suffers less error accumulation as compared to the PDCOMB-DFS since the proposed direct summation of DCT coefficients can be applied to non-MC macroblocks to reduce re-encoding errors. Furthermore, due to the dynamic frame allocation based on the motion activities, the degradation of video quality of the inactive participants is not very visible and this is also supported by Figs. 12 and 15. Table IX shows the PSNR of each participant of all 400 frames of the video sequence at 128 kb/s using different video combiners. The diagonal values indicate more active motion in different time slots of individual conference participants. The table shows that, by using the proposed video combiner, the PSNR's of all conference participants are greatly improved as compared to the PDCOMB-DFS during both the active and nonactive periods. In a practical multipoint video conferencing system, active participants should be given most attention, and the video quality of these active participants is particularly important. The proposed video combiner can achieve this goal significantly.

VI. Conclusions

This paper proposes a low-complexity and high quality frame-skipping transcoder. Its low complexity is achieved by using 1) a direct summation of the DCT coefficients for macroblocks coded without motion compensation to deactivate most complex modules of the transcoder, and 2) a cache subsystem for motion-compensated macroblocks to reduce redundant IDCT and inverse quantization. We have also shown that a direct summation of the DCT coefficients can eliminate the re-encoding error. Furthermore, our proposed frame-skipping transcoder can be processed in an novel arrangement when multiple frames are dropped. Thus, only one DCT-domain buffer is needed to store the updated DCT coefficients of all dropped frames. Overall, the proposed frame-skipping transcoder produces a better picture quality than the pixel-domain approach at the same reduced bitrates.

In the proposed transcoder, we assume that there is no change in quantization step-size. It is possible that the DCT coefficients are requantized with a larger quantization step-size than that originally used at the front encoder in order to provide a more flexible way for bit-rate reduction. The problem with this approach is that requantization error will be accumulated due to this change of quantization step-size. In this case the possible prediction memory mismatch at the decoder will cause "drift" degradation [16], [17]. The problem may be alleviated by using a reconstruction loop mentioned in [16], [17]. In this approach, the requantization error is stored in a frame buffer and is fed back to the requantizer to correct the requantization error introduced in the previous frames. This can be done by integrating both spatial and temporal transcoding techniques in DCT-domain, which is a fruitful direction for further research.

We have also integrated our proposed frame-skipping transcoder into a new video combining architecture for continuous presence multipoint video conferencing. In multipoint video conferencing, usually only one or two participants are active at any given time. When the frame skipping transcoding approach is used, the frame rate of coding a subsequence needed to achieve a certain quality level depends very much on its motion activity, using the frame-skipping transcoding approach. We can achieve a better video quality by dynamic frame allocation based on the motion activities of the subsequences. Since re-encoding is minimized in our frame-skipping transcoder, the proposed architecture provides a better performance than a video combiner in terms of quality and complexity. However, many problems still remain to be investigated. For example, it would be desirable to design an efficient global frame allocation algorithm which can guarantee a reasonable suboptimality frame-rate for all subsequences in

TABLE IX
AVERAGE PSNR'S OF THE COMBINED VIDEO SEQUENCE

	Frame 1 – 100		Frame 101 - 200		Frame 201 – 300		Frame 301 - 400	
	A	B	A	B	A	B	A	B
1st conference participant (most active during frame 1 –100)	**32.21**	**33.65**	32.99	34.15	32.60	34.30	32.41	34.19
2nd conference participant (most active during frame 101 –200)	33.42	35.07	**32.55**	**34.07**	33.39	34.55	33.36	34.50
3rd conference participant (most active during frame 201 –300)	34.11	35.31	34.31	35.24	**33.32**	**34.84**	33.85	35.20
4th conference participant (most active during frame 301 –400)	33.08	34.48	32.84	34.43	32.91	34.44	**31.66**	**33.64**

A - PDCOMB-DFS [27].
B - Proposed video combiner.

order to fulfill the desired output bitrate of the video combiner. Nevertheless, it can be seen that a video combiner using the frame-skipping transcoding approach is able to provide a new and viable continuous presence multipoint video conferencing service in the near future.

ACKNOWLEDGMENT

K. T. Fung acknowledges the research studentships provided by the Hong Kong Polytechnic University. The authors would like to thank the anonymous reviewers for their helpful comments.

REFERENCES

[1] S.-M. Lei, T.-C. Chen, and M.-T. Sun, "Video bridging based on H.261 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp. 425–437, Aug. 1994.
[2] M.-T. Sun, A. C. Loui, and T.-C. Chen, "A coded-domain video combiner for multipoint continuouspresence video conferencing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 855–863, Dec. 1997.
[3] M.-T. Sun, T.-D. Wu, and J.-N. Hwang, "Dynamic bit allocation in video combining for multipoint conferencing," *IEEE Trans. Circuits Syst II*, vol. 45, pp. 644–648, May 1998.
[4] C.-W. Lin, T.-J. Liou, and Y.-C. Chen, "Dynamicbit rate control in multipoint video transcoding," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 2, May 28–31, 2000, pp. 17–20.
[5] "Video Codecs for Audiovisual Services at $p \times 64$ kb/s," ITU-T Study Group XV, Rec. H.261, 1992.
[6] "Video Coding for Low Bitrate Communication," ITU-T Study Group XV, Rec. H.263, 1997.
[7] L. Chiariglione, "The development of an integrated audiovisual coding standard: MPEG," *Proc. IEEE*, vol. 83, pp. 151–157, Feb. 1995.
[8] "Information Technology-Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1,5 Mbit/s-Part 2: Video,", ISO/IEC 11 172-2, 1993.
[9] "Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Video,", ISO/IEC 13 818-2, 1996.
[10] H. J. Stuttgen, "Network evolution and multimedia communication," *IEEE Multimedia*, vol. 2, pp. 42–59, Fall 1995.
[11] "Multipointcontrol Units for Audiovisual Systems Using Digital Channels up to 2 Mb/s," ITU-T Study Group XV, Rec. H.231, 1992.
[12] "Procedures for Establishing Communication Between Three or More Audiovisual Terminals Using Digital Channels up to 2 Mb/s," ITU-T Study Group XV, Rec. H.243, 1992.
[13] H. Sun, W. Kwok, and J. W. Zdepski, "Architectures for MPEG compressed bitstream scaling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 191–199, Apr. 1996.
[14] Y. Nakajima, H. Hori, and T. Kanoh, "Rate conversion of MPEG coded video by re-quantization process," in *IEEE Int. Conf. Image Processing, ICIP95*, vol. 3, Washington, DC, Oct. 1995, pp. 408–411.
[15] P. Assuncao and M. Ghanbari, "Post-processing of MPEG2 coded video for transmission at lower bit rates," in *IEEE Int. Conf. Acoustics, Speech, and Signal Processing, ICASSP'96*, vol. 4, Atlanta, GA, May 1996, pp. 1998–2001.
[16] M. Yong, Q.-F. Zhu, and V. Eyuboglu, "VBR transport of CBR encoded video over ATM networks," in *Proc. 6th Int. Workshop Packet Video*, Portland, OR, Sept. 1994, pp. D18.1–D18.4.
[17] D. G. Morrison, M. E. Nilsson, and M. Ghanbari, "Reduction of the bitrate of compressed video while in its coded form," in *Proc. 6th Int. Workshop Packet Video*, Portland, OR, Sept., pp. D17.1–D17.4.
[18] P. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit-rate reduction of MPEG-2 bit streams," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, Dec. 1998.
[19] G. Keeman, R. Hellinghuizen, F. Hoeksema, and G.Heideman, "Transcoding of MPEG-2 bitstreams," *Signal Process.: Image Commun.*, vol. 8, pp. 481–500, Sept. 1996.
[20] J. Youn, M.-T. Sun, and C.-W. Lin, "Motion vector refinement for high-performance transcoding," *IEEE Trans. Multimedia*, vol. 1, pp. 30–40, Mar. 1999.
[21] ——, "Motion estimation for high performance transcoding," *IEEE Trans. Consumer Electron.*, vol. 44, pp. 649–658, Aug. 1998.
[22] Y. L. Chan and W. C. Siu, "New adaptive pixel decimation for block motion vector estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 113–118, Feb. 1996.
[23] ——, "Edge oriented block motion estimation for video coding," *Proc. Inst. Elect. Eng., Vis., Image, Signal Process.*, vol. 144, no. 3, pp. 136–144, June.
[24] ——, "On block motion estimation using a novel search strategy for an improved adaptive pixel decimation," *J. Vis. Commun. Image Represent.*, vol. 9, no. 2, pp. 139–154, June 1998.
[25] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, "A novel unrestricted center-biased diamond search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 369–377, Aug. 1998.
[26] F.-H. Cheng and S.-N. Sun, "New fast efficient two-step search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 977–983, Oct. 1999.
[27] J.-N. Hwang, T.-D. Wu, and C.-W. Lin, "Dynamic frame-skipping in video transcoding," in *Proc. IEEE Second Workshop on Multimedia Signal Processing*, 1998, pp. 616–621.
[28] S. H. Kwok, W. C. Siu, and A. G. Constantinides, "Adaptive temporal decimation algorithm with dynamic time window," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 104–111, Feb. 1998.
[29] ——, "A scaleable and adaptive temporal segmentation algorithm for video coding," *Graph. Models Image Process.*, vol. 59, pp. 128–138, May 1997.
[30] P. Yip and K. R. Rao, "The decimation-in-frequency algorithms for a family of discrete sine and cosine transforms," *Circuits, Syst., Signal Process.*, pp. 4–19, 1988.

Kai-Tat Fung received the B.Eng. (Hons.) and the M.Phil. degrees in 1998 and 2001, respectively, from the Hong Kong Polytechnic University (HKPU), where he is currently pursuing the Ph.D. degree. His research interests include video transcoding, video conferencing application, image and video technology, audio compression, and blind signal separation.

Yui-Lam Chan received the B.Eng. degree (with first class honors) and the Ph.D. degree from the Hong Kong Polytechnic University (HKPU) in 1993 and 1997, respectively.

He joined HKPU in 1997 and is now an Assistant Professor in the Centre for Multimedia Signal Processing and the Department of Electronic and Information Engineering. He has published over 20 research papers in various international journals and conferences. His research interests include multimedia technologies, signal processing, image and video compression, video transcoding, video conferencing, and digital TV.

Dr. Chan was the recipient of more than ten famous prizes, scholarships, and fellowships for his outstanding academic achievements, such as the Champion of the Varsity Competition in Electronic Design in 1993, and was a recipient of the Sir Edward Youde Memorial Fellowship and the Croucher Foundation Scholarships.

Wan-Chi Siu (M'77–SM'90) received the Associateship from The Hong Kong Polytechnic University (HKPU), formerly called the Hong Kong Polytechnic, the M.Phil. degree from The Chinese University of Hong Kong (CUHK), and the Ph.D. degree from Imperial College of Science, Technology & Medicine, London, U.K., in 1975, 1977, and 1984, respectively.

He was with CUHK between 1975 and 1980. He then joined HKPU as a Lecturer in 1980 and became Chair Professor in 1992. He was Head of Department of the Electronic and Information Engineering Department and subsequently Dean of Engineering Faculty between 1994 and 2002. He is now Director of the Centre for Multimedia Signal Processing, HKPU. He has published over 200 research papers, and his research interests include digital signal processing, fast computational algorithms, transforms, image and video coding, and computational aspects of pattern recognition, and neural networks. He is a member of the Editorial Board of the *Journal of VLSI Signal Processing Systems* for Signal, Image and Video Technology and the *EURASIP Journal on Applied Signal Processing*.

Prof. Siu was a Guest Editor of a special issue of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—PART II, in May 1998, and was also an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—PART II between 1995 and 1997. He was the general chair or the technical program chair of a number of international conferences. In particular, he was the Technical Program Chair of the IEEE International Symposium on Circuits and Systems (ISCAS'97) and the General Chair of the 2001 International Symposium on Intelligent Multimedia, Video & Speech Processing (ISIMP'2001), which were held in Hong Kong in June 1997 and May 2001, respectively. He was the General Chair of the 2003 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'2003), held in Hong Kong. Between 1991 and 1995, he was a member of the Physical Sciences and Engineering Panel of the Research Grants Council (RGC), Hong Kong Government, and in 1994, he chaired the first Engineering and Information Technology Panel to assess the research quality of 19 Cost Centers (departments) from all universities in Hong Kong. He is a Chartered Engineer and a Fellow of the IEE and the HKIE.