Digital Video Transcoding

JUN XIN, MEMBER, IEEE, CHIA-WEN LIN, SENIOR MEMBER, IEEE, AND MING-TING SUN, FELLOW, IEEE

Invited Paper

Video transcoding, due to its high practical values for a wide range of networked video applications, has become an active research topic. In this paper, we outline the technical issues and research results related to video transcoding. We also discuss techniques for reducing the complexity, and techniques for improving the video quality, by exploiting the information extracted from the input video bit stream.

Keywords—*Error resilience, motion estimation, rate control, transcoding architecture, video transcoding.*

I. INTRODUCTION

Video transcoding is the operation of converting a video from one format into another format. A format is defined by such characteristics as the bit rate, frame rate, spatial resolution, coding syntax, and content, as shown in Fig. 1.

One of the earliest applications of transcoding is to adapt the bit rate of a precompressed video stream to a channel bandwidth. For example, a TV program may be originally compressed at a high bit rate for studio applications, but later needs to be transmitted over a channel at a much lower bit rate.

In universal multimedia access [1], different terminals may have different accesses to the Internet, including local access network (LAN), digital subscriber line (DSL), cable, wireless networks, integrated services digital network (ISDN), and dial-up. The different access networks have different channel characteristics such as bandwidths, bit error rates, and packet loss rates. At the users' end, network appliances including handheld computers, personal digital assistants (PDAs), set-top boxes, and smart cellular phones are slated to replace personal computers as the dominant terminals for accessing the Internet. These network terminals vary significantly in resources such as computing power and display capability. To flexibly deliver multimedia data to users with different available resources, access networks, and interests, the multimedia content may need to be adapted dynamically according to the usage environment [2]. Transcoding is one of the key technologies to fulfill this challenging task. Transcoding is also useful for content adaptation for peer-to-peer networking over shared multihop communication links [3].

There are many other transcoding applications besides the universal multimedia access. In statistical multiplexing [4], multiple variable-bit-rate video streams are multiplexed together to achieve the statistical multiplexing gain. When the aggregated bit rate exceeds the channel bandwidth, a transcoder can be used to adapt the bit rates of the video streams to ensure that the aggregated bit rate always satisfies the channel bandwidth constraint. A transcoder can also be used to insert new information including company logos, watermarks, as well as error-resilience features into a compressed video stream. Transcoding techniques are also shown useful for supporting VCR trick modes, i.e., fast forward, reverse play, etc., for on-demand video applications [8]–[10]. In addition, object-based transcoding techniques are discussed in [7] for adaptive video content delivery. A general utility-based framework is introduced in [11] to formulate some transcoding and adaptation issues as resource-constrained utility maximization problems. In [12], a utility-function prediction is performed using automatic feature extraction and regression for MPEG-4 video transcoding. Several rate-distortion models for transcoding optimization are introduced in [13] to facilitate the selection of transcoding methods under a rate constraint. Envisioning the need of transcoding, the emerging MPEG-7 standard [5], which standardizes a framework for describing audiovisual contents, has defined "transcoding hints" to facilitate the transcoding of compressed video contents [6], [7].

Dynamic change of coding parameters such as bit rates, frame rates, and spatial resolutions could also be achieved

Manuscript received January 16, 2004; revised July 9, 2004.

J. Xin is with Mitsubishi Electric Research Laboratories, Cambridge, MA 02139 USA (e-mail: jxin@merl.com).

C.-W. Lin is with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 612, Taiwan, R.O.C. (e-mail: cwlin@cs.ccu.edu.tw).

M.-T. Sun is with the Department of Electrical Engineering, University of Washington, Seattle, WA 98195 USA (e-mail: sun@ee.washington.edu).

Digital Object Identifier 10.1109/JPROC.2004.839620

0018-9219/$20.00 © 2005 IEEE

Amazon/DivX
Exhibit 1007
Page 1

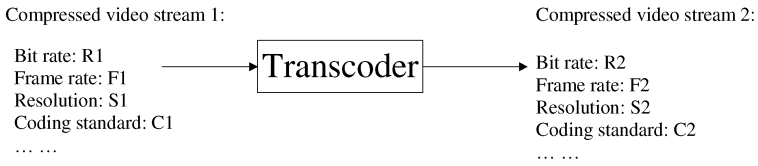

Fig. 1. Format conversion using a video transcoder.

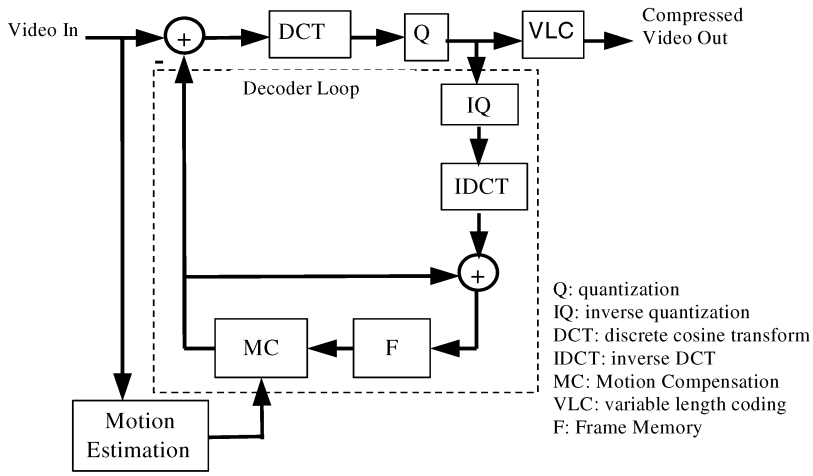

Fig. 2. Block diagram of a standard video encoder.

to a limited extent by scalable coding [14], [15]. However, in the current scalable video coding standards, the enhancement layers are generated by coding the prediction residuals between the original video and the base-layer video. In many applications, the network bandwidth may fluctuate wildly with time. Therefore, it may be difficult to set the base-layer bit rate. If the base-layer bit rate is set low, the base-layer video quality will be relatively low and the overall video quality degradation may be severe, since the prediction becomes less effective. On the other hand, if the base-layer bit rate is set high, the base-layer video may not get through the network completely. In general, the achievable quality of scalable coding is significantly lower than that of nonscalable coding. In addition, scalable video coding demands additional complexities at both encoders and decoders. The inherent weaknesses of scalable coding have kept it from being widely deployed in practical applications. Nevertheless, scalable coding is still an active research area. A new coding standard is being developed to overcome these drawbacks [16]. With these problems addressed, the scalable coding schemes may be more suitable for streaming video applications when a large number of users require different levels of format adaptations, since less computation is involved.

In this paper, the input to the transcoder is a compressed video produced by a standard video encoder. Current major video coding standards—MPEG-1 [17], MPEG-2 [18], MPEG-4 [19], H.263 [20], and the emerging H.264 [21] all use hybrid discrete cosine transform (DCT) and block-based motion compensation (MC) schemes. Note that H.264 uses an integer transform that approximates DCT. A block diagram of the standard video encoders is shown in Fig. 2. MC reduces the temporal redundancy. DCT reduces the spatial redundancy and achieves energy compaction. Quantization is performed to achieve higher compression ratio. Variable-length coding (VLC) is applied after the quantization to reduce the remaining redundancy. A decoder is embedded in the encoder to reconstruct video frames, which are stored in the frame memory for prediction of future frames.

A straightforward realization of a transcoder to cascade a decoder and an encoder: the decoder decodes the compressed input video and the encoder reencodes the decoded video into the target format. It is computationally very expensive. Therefore, reducing the complexity of the straightforward decoder-encoder implementation is a major driving force behind many research activities on transcoding.

What makes transcoding different from video encoding is that the transcoding has access to many coding parameters and statistics that can be easily obtained from the input compressed video stream. They may be used not only to simplify the computation, but also to improve the video quality. The transcoding can be considered as a special two-pass encoding: the "first-pass" encoding produces the input compressed video stream, and the "second-pass" encoding in the transcoder can use the information obtained from the first-pass to do a better encoding. Therefore, it is possible for the transcoder to achieve better video quality than the straightforward implementation, where the encoding is single pass. The challenge of the research on transcoding is then how to intelligently utilize the coding statistics and parameters extracted from the input to achieve the best possible video quality and the lowest possible computational complexity.

In this paper, we discuss the issues and research results related to the transcoding of video streams compressed using

Amazon/DivX
Exhibit 1007
Page 2

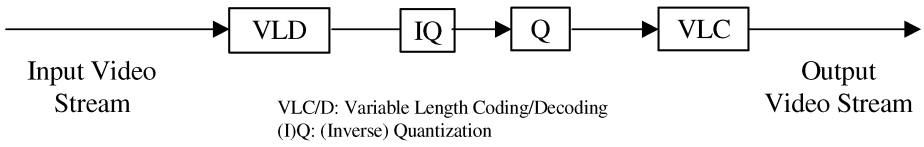

Fig. 3. Transcoding using requantization.

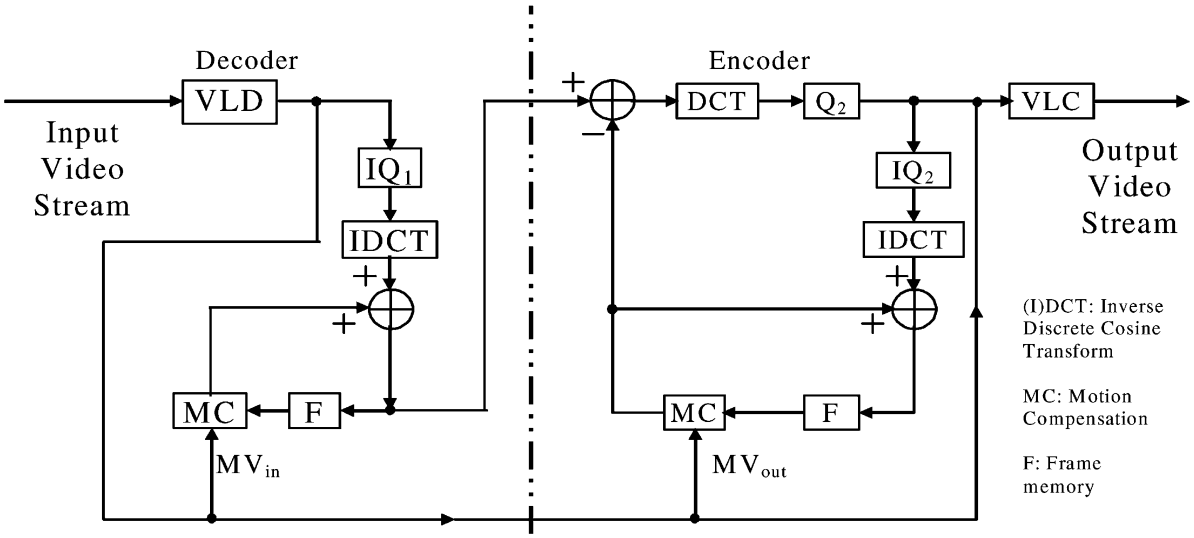

Fig. 4. CPDT architecture.

the hybrid MC/DCT schemes. An overview of transcoding architectures and techniques has been given in [22], which presents many of the fundamentals in this area. This paper is intended to provide a more in-depth view of architectures and techniques, and cover such topics as quality optimization, complexity reduction techniques, and related applications such as logo and watermark insertion.

The remainder of this paper is organized as follows. In Section II, we first review the transcoding techniques used for the bit-rate reduction. In Section III, we discuss the transcoding techniques for spatial and temporal resolution reductions. Section IV discusses the issues associated with the standards conversion. Section V addresses the transcoding quality optimization. Section VI discusses the transcoding for information insertion. Finally, Section VII concludes this paper.

II. Bit-Rate Transcoding

Generally, there exist three transcoding architectures for the bit-rate transcoding: open-loop transcoders [23]–[25], cascaded pixel-domain transcoders (CPDTs) [24], [26] and DCT-domain transcoders (DDTs) [27]–[29]. The open-loop architectures include selective transmission [23], [24], where the high-frequency DCT coefficients are discarded, and requantization [24], [25], where the DCT coefficients are requantized. Fig. 3 shows a requantization transcoder. The open-loop transcoders are computationally efficient, since they operate directly on the DCT coefficients. However, they suffer from the drift problem.

The drift problem is explained as follows. A video picture is predicted from its reference pictures and only the prediction errors are coded. For the decoder to work properly, the reference pictures reconstructed and stored in the decoder predictor must be same as those in the encoder predictor. The open-loop transcoders change the prediction errors and, therefore, make the reference pictures in the decoder predictor different from those in the encoder predictor. The differences accumulate and cause the video quality to deteriorate with time until an intrapicture is reached. The error accumulation caused by the encoder/decoder predictor mismatch is called drift and it may cause severe degradation to the video quality [26], [30]. It should be noted that in the following discussions, many transcoder architectures are not strictly drift free. However, the degree of the video quality degradation caused by the drift varies with architectures. In addition, the drift will be terminated by an intrapicture. In the applications where the number of coded pictures between two consecutive intrapictures is small and the quality degradation caused by the drift is acceptable, these architectures, although not drift free, can still be quite useful due to the potentially lower cost in terms of computation and required frame memory.

Fig. 4 illustrates the drift-free CPDT [24], a concatenation of a decoder and a simplified encoder. Rather than performing the full-scale motion estimation, as in a stand-alone video encoder, the encoder reuses the motion vectors along with other information extracted from the input video bitstream. Thus, the motion estimation, which usually accounts for 60%–70% of the encoder computation [31], is omitted. To

Amazon/DivX
Exhibit 1007
Page 3

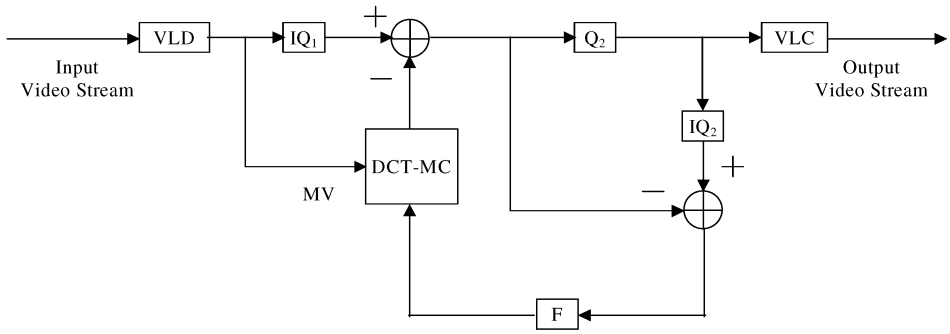

Fig. 5. SDDT architecture.

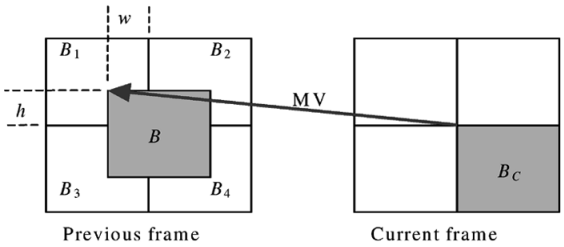

Fig. 6. DCT-MC.

address another major source of computational complexity, DCT, in [26], the end-of-block locations of the output video are predicted using those of the input video, and only partial DCT is performed.

The simplified DCT-domain transcoder (SDDT) is derived based on the assumption that DCT, IDCT, and MC are linear operations. It was first derived in [27] and [28], and then further simplified in [29], as shown in Fig. 5. SDDT eliminates the DCT/IDCT and reduces the number of frame buffers by half. The DCT-domain MC (DCT-MC) [32] is the major computation-intensive operation in SDDT. As shown in Fig. 6, the goal is to compute the DCT coefficients of the target DCT block B from the coefficients of its four overlapping DCT blocks, B_1–B_4. To speed up the DCT-MC, several fast schemes have been proposed. In [33], the DCT-MC is simplified through a matrix decomposition. A fast algorithm utilizing shared information in a macroblock (MB) is proposed in [34]. In [35], the fact that the energy of a DCT block is concentrated in its low frequency coefficients is exploited to perform an efficient approximation of the MC operation. Another approach is to separate the MC into two one-dimensional (1-D) operations [36], which are further simplified by a lookup table scheme [37].

With these fast algorithms, SDDT may require less computation and memory than CPDT. However, the linearity assumptions on which the derivation is based are not strictly true, since there are clipping functions performed in the video encoder/decoder, and rounding operations performed in the interpolation for fractional-pixel MC. The failed assumptions may cause drift in the transcoded video. Detailed analyses of the causes and impacts of drift can be found in [30].

In addition, SDDT can only be applied to bit-rate transcoding, since it assumes that the frame memories in the encoder and the decoder have the same spatial/temporal resolution and the output video uses the same frame coding types, motion vectors and coding modes as the input video. In contrast, CPDT enjoys the flexibility to allow changes in these coding parameters.

The cascaded DCT-domain transcoder (CDDT) [38], shown in Fig. 7, can be used for spatial/temporal resolution downscaling and other coding-parameter changes. However, compared to SDDT, its flexibility is achieved using additional DCT-MC and frame memory, which results in a significantly higher cost in computation and storage. It is thus often adopted for downscaling applications, where the encoder-side DCT-MC and memory will not cost much, since the encoder operates at a reduced resolution [39], [40].

Table 1 compares the computational complexity of five different MPEG-2 transcoders: CPDT, SDDT, CDDT, CPDT using full-scale full-search motion reestimation (DEC-ENC1), and CPDT using three-step fast search motion reestimation (DEC-ENC2). Two 300-frame CIF (352×288) test sequences, "Foreman" and "Mobile & Calendar," coded with two group-of-pictures (GOP) structures, (15, 1) (i.e., IPPP...) and GOP $= (15, 3)$ (i.e., IBBP...), are used in the simulations. The experiments are performed on a Pentium-IV 1.8-GHz PC. The two DCT-domain transcoders are significantly faster than the pixel-domain transcoders based on our implementations. It should be noted that the speed comparison might depend on the specific implementations of the architectures. There exist different low-level optimization methods that can be used to reduce the computational complexities of CPDT, SDDT, and CDDT. For example, the method in [90] can significantly speed up CPDT, while the methods in [29] and [33]–[35] can speed up SDDT and CDDT.

Our experiments show that the pixel-domain transcoders (except DEC-ENC2) usually have better peak signal-to-noise ratio (PSNR) performance as illustrated in Figs. 8 and 9. In Fig. 8, the test video is first encoded at QP (Quantization Parameter) $= 7$ and GOP $= (15, 3)$, and the bit rate is then reduced using the five transcoders by transcoding at QP $= 10, 12, 15, 17, 20$, respectively. In Fig. 9, the incoming video is transcoded at QP $= 11$ for five different GOP sizes (GOP $= (N, 3)$,

Amazon/DivX
Exhibit 1007
Page 4

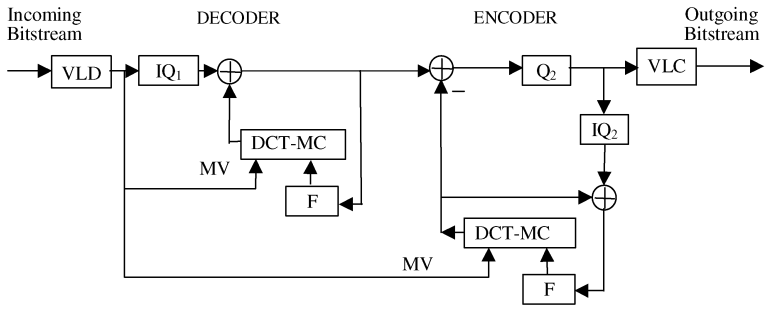

Fig. 7. CDDT architecture.

Table 1
Runtime Complexity Comparison of Five Different Transcoders. The Video Sequences Are Encoded at QP $= 7$, and Then Transcoded at QP $= 15$

	Foreman		Mobile & Calendar	
	IPPP…	IBBP…	IPPP…	IBBP…
CPDT	7.0 fps	7.1 fps	6.6 fps	6.7 fps
SDDT	23.2 fps	17.1 fps	23.0 fps	17.5 fps
CDDT	14.1 fps	9.7 fps	15.7 fps	10.7 fps
DEC-ENC1	2.4 fps	2.2 fps	2.4 fps	2.1 fps
DEC-ENC2	4.2 fps	3.8 fps	4.4 fps	3.6 fps

for $N = 15, 30, 60, 90, 120$, and 240, respectively). The number associated with each operation point indicates the bit rate generated. We can observe from Figs. 8 and 9 that the drift caused by the two DCT-domain transcoders is not serious for small GOP sizes. However, the performance degradation, especially for SDDT, can become rather significant with large GOP sizes. Such large GOP sizes may be used in applications such as networked video streaming and wireless video that demand high coding efficiency.

III. Spatial and Temporal Transcoding

The heterogeneity of communication networks and network access terminals often demand the conversion of compressed video not only in the bit rates, but also in the spatial/temporal resolutions. One of the challenging tasks in spatial/temporal transcoding is how to efficiently reestimate (or map) the target motion vectors from the input motion vectors.

Many works on the motion reestimation for spatial transcoding consider the simple case of $2:1$ downscaling. Fig. 10 illustrates a case of the motion-mapping problem, where the input MBs have four motion vectors while the target output MB has a single motion vector. Several strategies have been proposed to compose the target motion vector using the input motion vectors. One strategy is to randomly choose from the four input motion vectors [41], [42]. The weighted average taking into account the prediction error is presented in [43]. Different methods are compared in [31] and [41], including median, majority, average, and random selection. The median method is shown to achieve the best performance. The work in [44] selects the motion vector using a likelihood score based on the statistical characteristics of the MBs associated with the best matching motion vectors. Note that the motion vectors formed by the above algorithms need to be downscaled to the target spatial resolution.

Recent works extend these strategies to tackle the transcoding of arbitrary down-sampling ratio [45], [46] by taking care of the unequal contributions of related input motion vectors. When the down-sampling ratio is large and one target MB is down-sampled from a number of input MBs, the motion vectors of the input MBs are more likely to be inconsistent. A multicandidate approach is proposed in [47] to address this issue. The transcoding of interlaced video is discussed in details in [46], where the motion mapping is further complicated by various types of frame and field motion vectors.

For transcoding with temporal resolution changes, due to the frame dropping, one has to derive a new set of motion vectors that do not exist in the input video. This issue is addressed in [48], where a technique called forward dominant vector selection (FDVS) is proposed. The FDVS scheme is illustrated in Fig. 11. The best-match area referenced by the motion vector of the current MB overlaps with at most four MBs in its reference frame. The motion vector of the MB with the largest overlapping portion is called the dominant motion vector and is selected for composing the target motion vector. This process is repeated for all the dropped frames and the final target motion vector is formed by adding all the dominant motion vectors together, followed by a motion vector refinement. In [49], the dominant motion vector is selected based on the activity of the overlapping MBs, instead of the area as in FDVS. Another method, telescopic vector composition (TVC) [31], accumulates all motion vectors of the current MB's colocated MBs in the dropped frames and adds the resulting motion vector to the current MB's motion vector. For typical videos with small motion vectors, TVC can achieve similar performance as FDVS. It is shown [31] that a ± 2-pixel refinement around the composed motion vector can achieve similar performance as the full-scale full-search motion reestimation. For the spatial resolution reduction, typically a half-pixel refinement is enough to achieve a good quality [31], [46]. For the temporal resolution reduction, as the number of skipped frames increases, more refinement may be desirable. The refinement range may be dynamically decided based on the motion vector magnitudes and the number of skipped frames [50]. In [48], the refinement range is determined based

Amazon/DivX
Exhibit 1007
Page 5

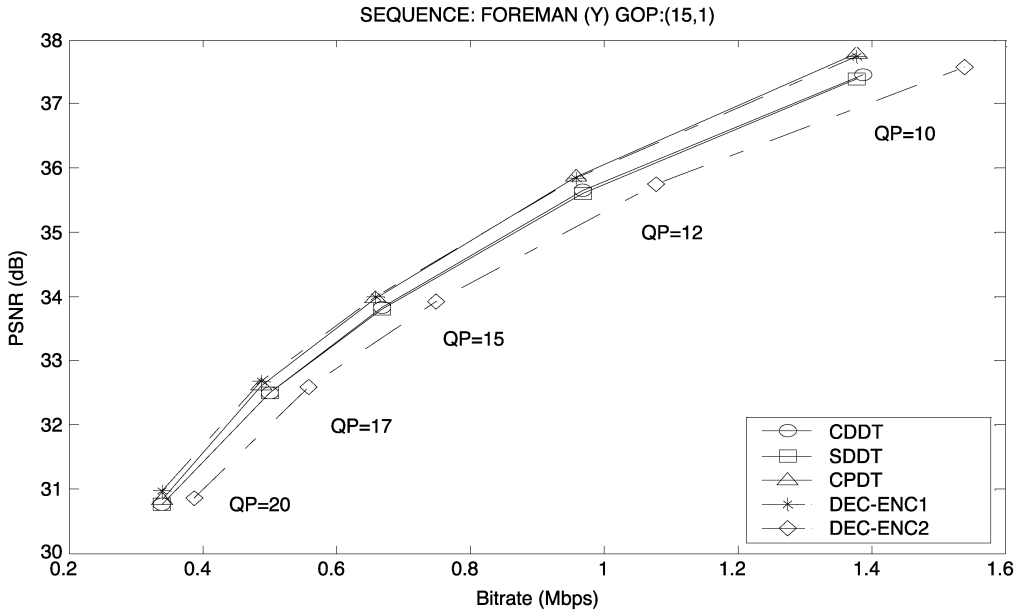

Fig. 8. Performance comparison of average PSNR for five transcoders. The "Foreman" sequence is encoded at QP $= 7$, and then transcoded at different QP's, respectively. GOP $= (15, 1)$.

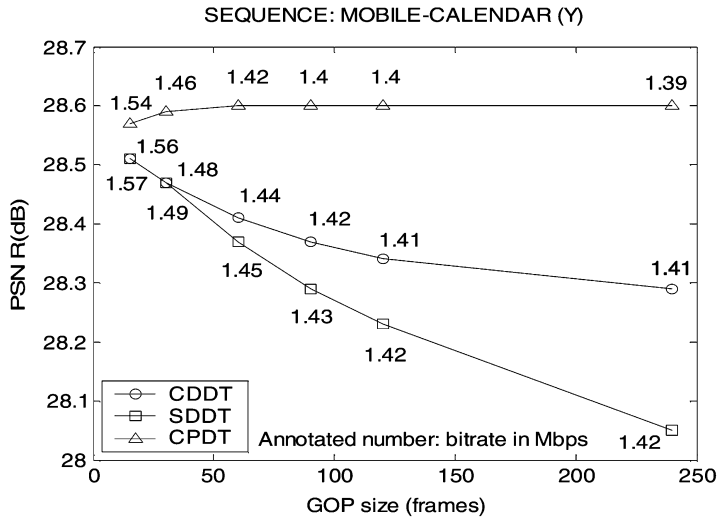

Fig. 9. Performance comparison of average PSNR for CPDT, SDDT, and CDDT for different GOP sizes. The "Mobile & Calendar" sequence is encoded at QP $= 5$ using size different GOP sizes, and then transcoded at QP $= 11$.

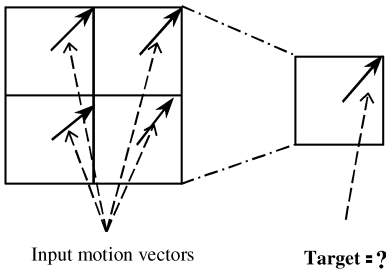

Fig. 10. In $2:1$ spatial transcoding, the target motion vector(s) for that MB is highly correlated with the four input motion vectors.

on the input/output quantization scales and the prediction errors. In [51], a fast motion vector refinement scheme is proposed for the DCT-domain spatial downscaling, where the speedup is achieved through exploiting the redundancies in the DCT-MC computations of two adjacent checkpoints.

Due to the spatial/temporal resolution reduction, the drift problem is usually significant in open-loop transcoding architectures [52]. Therefore, the drift-free CPDT architecture is more favorable in terms of quality. In [53] and [54], the drift in spatial transcoding is analyzed. Based on the analyses, several drift-compensation architectures providing different levels of complexity-quality tradeoffs are proposed. In-depth analyses and performance comparisons of these alternative architectures and CPDT are provided in [55].

A DCT-domain architecture is proposed in [56] for temporal transcoding, where the reencoding errors are reduced using the direct addition of DCT coefficients and signals

Amazon/DivX
Exhibit 1007
Page 6

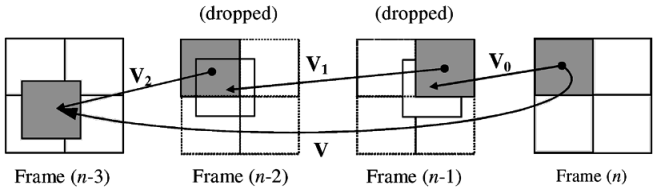

Fig. 11. FDVS. $\mathbf{V}_1$ and $\mathbf{V}_2$ are dominant motion vectors.

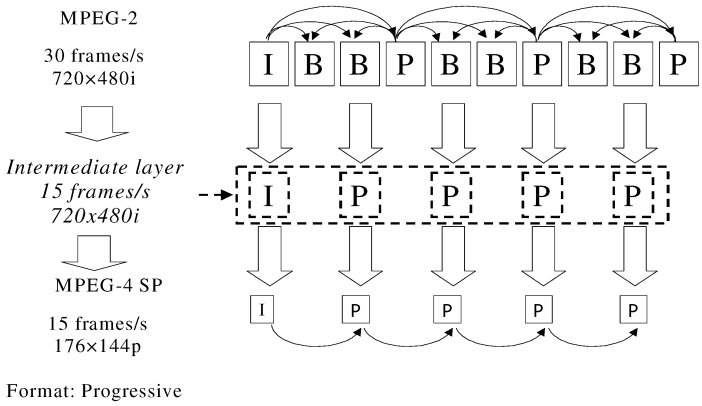

Fig. 12. Example of MPEG-2 to MPEG-4 simple profile transcoding.

from an error compensation feedback loop. In [57], a hybrid DCT/pixel-domain transcoder architecture is proposed for video downscaling. It contains a DCT-domain decoder followed by a pixel-domain encoder, where a modified DCT-domain inverse transformation and down-sampling method is developed to convert a DCT block into a downscaled pixel block.

IV. Standards Transcoding

In many applications, video coded in one coding standard (e.g., MPEG-2) may need to be converted to another standard (e.g., MPEG-4) besides the changes in bit rate and resolution. In what follows, we use two examples to illustrate how the information obtained from the input video sequence may be used to help the standards transcoding process.

A. MPEG-2 to MPEG-4 Simple Profile (SP) Transcoding

MPEG-4 SP is aimed at low-complexity and low-bit-rate video applications. Compared to MPEG-2 video, it does not support B-frames and interlaced video. In addition, it usually operates at lower spatial resolutions and frame rates than MPEG-2 video. Fig. 12 illustrates a typical scenario: an interlaced MPEG-2 video of 720×480 resolution and at 30 frames/s is transcoded to the progressive MPEG-4 SP of 176×144 resolution at 15 frames/s. It involves conversion of video formats and frame coding types besides the spatial and temporal resolution conversions. The new challenge is that the motion vectors of an incoming video frame may not use the same reference frame as the target frame.

The motion reestimation problem in the case of frame type conversion is first discussed in [31], where the target motion vector is chosen from several candidate motion vectors that are formed by using the motion information from current and adjacent frames. The work in [58] and [59] introduces an intermediate, virtual layer of video, which has the same frame rate and frame type as the target video and the same spatial resolution as the input video. The motion reestimation process consists of two steps. In the first step, one or more intermediate motion vectors are formed for each MB in the intermediate video frame using the motion information of the input video. In the second step, these motion vectors of the intermediate-layer video are used to compose the motion vectors for the target video. This step also takes care of the mismatch of the motion vector types caused by the interlaced input and the progressive output. Effectively, the first step handles the frame-rate reduction and the frame-type conversion, and the second step deals with the spatial-resolution reduction and the interlaced-to-progressive processing. This two-step process has low complexity, since all operations are performed on the motion vectors and, therefore, the computationally expensive block matching is not needed.

B. MPEG-2 to MPEG-4 Advanced Simple Profile (ASP) Transcoding

Aiming at providing high quality video coding, MPEG-4 ASP incorporates several new coding tools. One of the tools is global MC (GMC), which can improve the coding performance for scenes with global motions [60]. No previous video coding standards, including MPEG-2, support GMC. Therefore, in the transcoding of MPEG-2 to MPEG-4 ASP, global motion (GM) parameters may be estimated to take advantage of this tool. The estimation is referred to as global motion estimation (GME). Direct GME methods operate in the pixel-domain [61], [62]. They are computationally expensive due to the iterative processes in the nonlinear esti-

Amazon/DivX
Exhibit 1007
Page 7

mations and the number of pixels involved when the general perspective model is used.

A much more efficient algorithm is presented in [63] by performing the GME based on the input MB motion vectors, instead of on estimating the pixel-wise motion vector for each decoded pixel. The motion vectors in the input video stream are obtained from the block matching motion estimation process. They contain GM information plus local motion and noise due to the inaccurate block-matching process. The local motion and the block-matching noise are modeled as a Gaussian distribution with zero mean. The GM parameters are obtained by iteratively minimizing the fitting error between the input motion vectors and the sampled motion vectors generated from the estimated motion model using the Newton–Raphson method with outlier rejections. This compressed-domain GME is shown to be fast (only requires less than 0.2% of the processing compared to the pixel-domain GME implemented in the MPEG-4 reference software), robust, and give accurate results, so the computationally expensive pixel-domain GME can be saved.

C. Transcoding Between Other Standards

The recently developed scalable coding standard MPEG-4 fine granularity scalability (FGS) [15] has attracted interests of transcoding between FGS and single-layer video. An efficient architecture is derived in [64] for transcoding the FGS video to a single-layer video. In [65], it is pointed out that at the same bit rate, the transcoding from FGS to single-layer yields better video quality than simply truncating the FGS bit-stream. Various methods for transcoding a single-layer MPEG stream to an FGS stream, both with the open-loop and closed-loop structures, are investigated in [66].

The syntax translation between different formats with minimal quality loss is addressed in [67], where mapping techniques of the syntactic and semantic elements of H.263 and MPEG-4 video are presented. Other format transcoding techniques not covered by this section can be found in [68]–[70].

V. Transcoding Quality Optimization

As mentioned earlier, transcoding can be considered as the second pass of a two-pass video encoding process where the input video bit-stream is considered as the result of the first pass. Many useful statistics, such as the quantization step sizes, coding modes, coded bits of each MB and frame, and motion vectors, can be easily obtained from the input video bit-stream to help the second pass encoding. Therefore, it is possible for the transcoder to achieve better video quality than the direct one-pass encoding using the original source. Although, in the transcoding, the video is encoded twice, the degradation in the first encoding pass may be negligible compared to the degradation in the second encoding pass. In what follows, we discuss the technologies related to the video quality optimization: requantization, rate control, and mode decision.

A. Requantization

Quantization is the only operation in current video coding standards that introduces quality loss. Video coding standards specify the representation levels of the quantization, not the decision levels. Not considering other constraints, the optimal quantizer simply maps an input value to its nearest representation level [71]. In [72], optimal requantization for transcoding MPEG-2 intraframes (I-frames) in the probabilistic sense are proposed based on two principles: minimization of the MSE (MMSE) cost function and *maximum a posteriori* (MAP). Both methods require the knowledge of the original quantization method and the original DCT coefficients distribution, which may be carried in the input video stream as the user data or may be estimated from the input video with minor additional complexity. Methods for estimating model parameters for this purpose are discussed in [73]. The MMSE and MAP requantization strategies are shown to achieve improved performance compared to those designed for a video encoder. The requantization distortion is especially significant for certain ratios between output and input quantization scales [72], [74]. This leads to the selective requantization scheme [74] that simply avoids those critical input/output quantization-scale ratios in the decision of quantization scales. In [75], the quantization is optimized jointly with deblocking.

B. Rate Control

Rate control determines quantization parameters and is responsible for maintaining consistent video quality while satisfying bandwidth, delay, and memory constraints. Generally, all rate-control algorithms designed for video coding are applicable to transcoding, for instance, MPEG-2 TM5 [76]. In practice, however, functional limitations need to be considered. For example, a rate-control algorithm usually needs to know the GOP configuration. However, in transcoding, the output GOP configuration is often determined by the input one, since transcoding typically does not change the frame coding types in order to keep complexity low. In real-time transcoding, the input GOP configuration is usually unknown to the transcoder. For these applications, in order to use the GOP-based rate-control algorithms, the input GOP configuration needs to be estimated. One solution is to predict the current GOP configuration based on the configuration of the immediately preceding GOP and adjust the rate-control algorithm accordingly when new GOP parameters are detected [77]. Another solution is to simply scale the bits of each frame according to the rate conversion ratio [78]. In [79], in addition to scaling the input frame bit number, relatively more bits are allocated to I-frames, since I-frames need a larger portion of the available bit budget to achieve a consistent quality with other frames [46], [59], [79], especially when the target bit rate is low.

For applications where the delay of a GOP time is allowed such that the coding statistics of the current input GOP can be collected before transcoding, improved rate control can be achieved by making use of the coding statistics [46]. A rate-control algorithm allocates bits to frames proportional to

Amazon/DivX
Exhibit 1007
Page 8

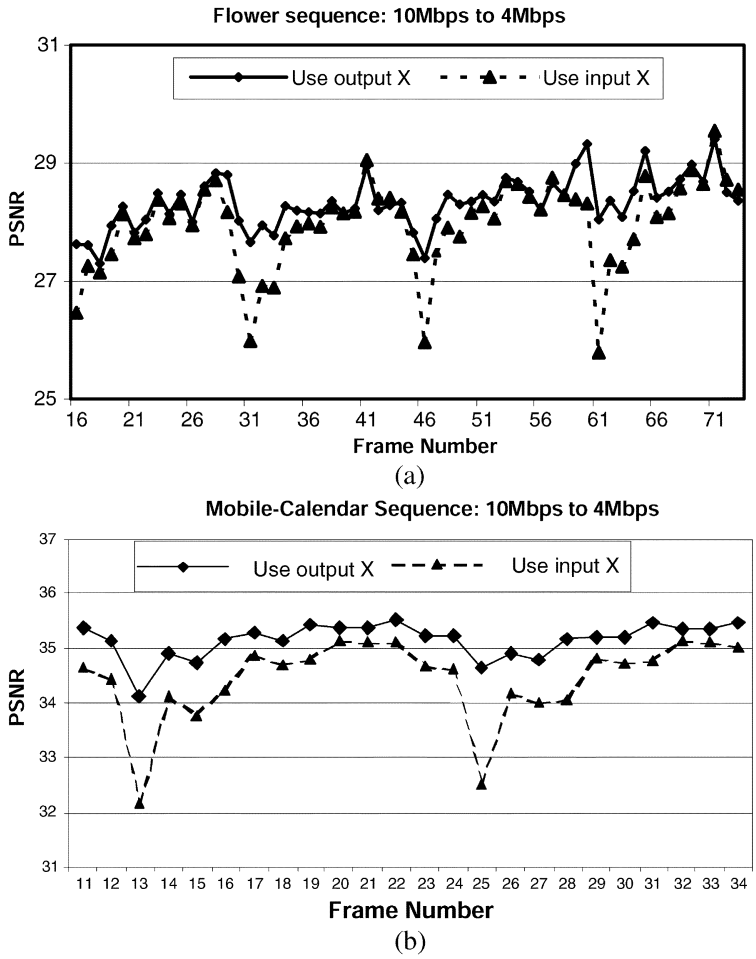

Fig. 13. Performance of the bit-allocation using output or input complexity (X).

their complexities. In video coding, however, the current and future frame complexities are usually unknown prior to encoding. Rate-control algorithms designed for encoding, such as MPEG-2 TM5, based on the stationary assumption, estimate the complexity of the current frame using the complexity of the previous frame of the same type. It is well known that this estimation is poor when the stationary assumption fails. In transcoding, intuitively it is possible to compute the frame complexities from the input bit-stream (since the quantization step sizes and the number of bits information are available), and then use these complexities for the bit allocation in transcoding. The approach of scaling down the input frame bits is one special case of such algorithms where the number of bits is taken as the complexity measure. It is found in [46] and [80] that complexity measures depend on the coding bit rate. Therefore, the complexity measures calculated from the input video bit-stream at the input bit rate may not be suitable to directly serve as the complexity measure for coding the frames at the output bit rate. Instead, the correlations between the complexity measures of the input and output videos are utilized to provide a more accurate estimation of the output frame complexity, which leads to improved bit-allocation and video quality as shown in Fig. 13, where the transcoder input and output are both MPEG-2, the bit rate is reduced from 10 to 4 Mb/s. The above techniques can be adapted to the joint transcoding of multiple preencoded video streams (statistical multiplexing) [4], [81], [82].

MB-layer rate control adjusts the quantization parameters based on the encoder buffer feedback and is particularly desirable for low-delay transcoding. Research works on this topic can be found in [59], [79], and [83]. In [7], a joint rate-control scheme taking into account the various spatio-temporal tradeoffs among all objects in a scene for MPEG-4 object-based transcoding is proposed. Dynamic rate-control algorithms tailored for multipoint video conferencing are discussed in [84]–[86].

C. Mode Decision

There are various levels of mode decisions, including MB-level, frame-level, and object-level. Rate-distortion optimized mode decision techniques are explained in details in [87]. These techniques are also applicable to transcoding. However, suboptimal but simple mode decision strategies are often desirable in complexity-constrained transcoding. In bit-rate transcoding, typically the modes of the input video are reused by the transcoder [28].

Amazon/DivX
Exhibit 1007
Page 9

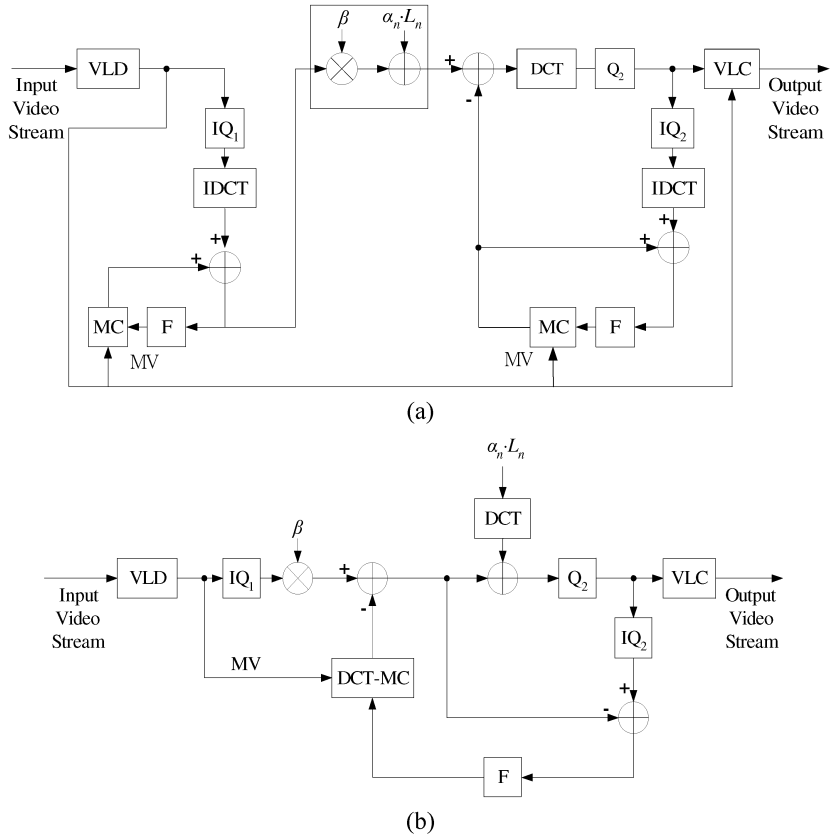

Fig. 14. Logo insertion in: (a) the pixel domain and (b) the DCT domain.

In spatial transcoding, the MB-type (inter/intra) decision usually follows the majority of the input MB-types [31], [46]. Various heuristic strategies are discussed in [54] to perform the decision for the open-loop architectures. MB prediction-mode decision techniques, including the frame/field prediction for interlaced transcoding, are discussed in [46]. A good overview of MB-level mode decision techniques can be found in [22].

Dynamic frame skipping based on the accumulated magnitude of motion vectors is proposed in [50] and [86]. In [7], strategies are proposed to drop less relevant objects if the scene has been coded as a set of objects. In [6], it is demonstrated that transcoding hints of MPEG-7 are valuable in improving mode decisions at various levels.

VI. Information Insertion Transcoding

In general, any operation that changes the content of a compressed video stream may be regarded as transcoding. In this section, we discuss two information insertion examples.

A. Logo/Watermark Insertion

For copyright protection, video watermarks and company logos can be inserted into the compressed video stream [88]. In the pixel-domain transcoders, the logo insertion can be implemented as illustrated in Fig. 14(a)

$$W_n = \alpha_n \cdot L_n + \beta \cdot I_n$$

where L_n, I_n, and W_n are the pixel values of the logo signal, the decoded picture, and the logo-inserted picture for frame n, respectively. α_n and β are the scaling factors for frame n controlling the intensity of the logo in order to provide uniform visibility [89], [90]. Efficient architectures performing this operation in the compressed-domain as illustrated in Fig. 14(b) are proposed in [88] and [90]. These architectures realize the same function as their pixel-domain counterpart does.

Fig. 15 shows a logo and a sample picture after logo insertion in the DCT domain [90]. The logo is inserted into an MPEG-2 encoded bit-stream whose bit rate is reduced from 8 to 4 Mb/s. The approach of [89] inserts the information in a simple, open-looped manner, where only the area affected by the inserted information needs to be modified. However, it is subject to drift. The insertion of new information may affect the optimality of the existing coding parameters of the affected picture area, including the motion vectors and coding modes as discussed in [91], where techniques are discussed to modify these coding parameters.

B. Error-Resilience Transcoding

In practical applications where video contents are compressed and stored for future delivery, the encoding process is typically performed without enough prior knowledge about the channel characteristics of network hops between the encoder and the decoder. In addition, the heterogeneity

Amazon/DivX
Exhibit 1007
Page 10

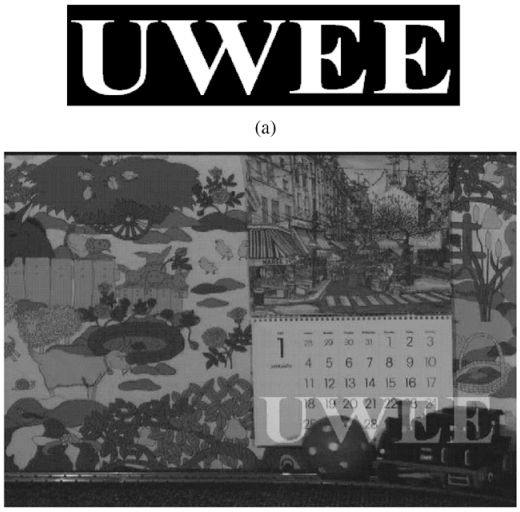

Fig. 15. DCT-domain logo insertion ($\alpha_n = 0.2, \beta = 0.8$). (a) Logo. (b) Logo-inserted sample picture.

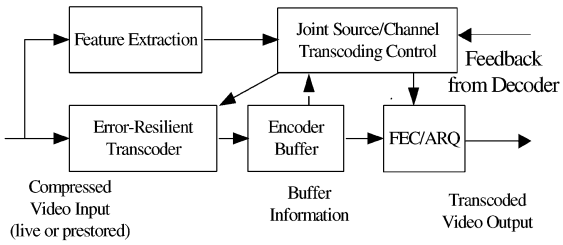

Fig. 16. System framework of error-resilience video transcoder.

of client networks also makes the encoder difficult to adapt the video contents to a wide degree of different channel conditions, especially for wireless client terminals. To overcome these problems, a video transcoder can be placed in a network node (e.g., mobile switch/base-station, proxy server, and video gateway) connected to a high-loss network (e.g., wireless network or highly congested network) to insert error-resilience features into the video bit-stream to achieve robust video transmission over wireless channels.

Fig. 16 shows a typical example of error-resilience transcoder with feedback [92]–[94]. The transcoder first extracts the video features (e.g., locations of video data which are likely to result in more serious error propagation if lost) from the incoming bit-stream as well as estimates the client channel conditions according to the feedback channel statistics. The extracted features and the estimated channel conditions are then used to determine the error resilience policy that guides the joint allocation of source/channel coding resources. The features of video contents can also be precomputed in the front-end encoding process and sent to the transcoder as auxiliary data to assist the transcoding. Commonly used error-resilience source coding tools [95] include data partitioning, synchronization marker, reversible variable length codes (RVLC), error-resilience entropy coding (EREC), multiple-description coding (MDC), reference frame selection (RFS), adaptive intra refresh (AIR), and so on. Forward error correction (FEC) and automatic retransmission request (ARQ) are two major schemes for channel protection.

An error-resilience MPEG-2 transcoding scheme based on EREC is proposed in [96]. In this method, the incoming bit-stream is reordered without adding redundancy such that longer VLC blocks fill up the spaces left by shorter blocks in a number of VLC blocks that form a fixed-length EREC frame. Such fixed-length EREC frames of VLC codes are then used as synchronization units, where only one EREC frame, rather than all the codes between two synchronization markers, will be dropped should any VLC code in the EREC frame be corrupted due to transmission errors. In [92], the authors propose a rate-distortion framework with analytical models that characterize the error propagation of a corrupted video bit-stream subjected to bit errors. These models are then used to guide the use of spatial and temporal localization tools: synchronization markers and intrarefresh, respectively, to compute the optimal bit-allocation among spatial error-resilience, temporal error-resilience, and the source rate. The work in [93] proposes an error-resilience transcoder for general packet radio service (GPRS) mobile-access networks with the transcoding process performed at a video proxy that can be located at the edge of two or more networks. Two error-resilience tools: the AIR and RFS methods with feedback control signaling (FCS) are used adaptively to reduce error effects, while preserving the transmission rate adaptation feature of the video transcoders. In [97], a rate-distortion optimized GOP-based bit-allocation scheme is proposed based on models accounting for the interframe dependence in both video source requantization and error propagation of motion compensated video. In [94], the authors propose a multiple-description FEC (MD-FEC)-based transcoding scheme, which uses the $(N, i, N - i + 1)$ Reed–Solomon erasure-correction block code to protect the ith layer of an N-layer scalable video. The multiple-description packetization method is specially designed to allow the ith layer to be decodable when i or more descriptions arrive at the decoder. The scheme in [98] proposes to implement an ARQ proxy at the base station of a wireless communication system for handling ARQ requests and tracking errors to reduce retransmission delays as well as to enhance the error resilience. The ARQ proxy resends important lost packets (e.g., packets with header information and motion vectors) detected through the retransmission requests from wireless client terminals, while dropping less important packets (e.g., packets carrying DCT coefficients) to meet the bandwidth limit. A transcoder is used to compensate for the mismatch error between the front-end video encoder and the client decoders caused by the dropped packets.

VII. Conclusion

In this paper, we provide an overview of issues related to transcoding, including transcoding applications, transcoder architectures, techniques for reducing the computation, and techniques for improving the video quality. Transcoding is an active research topic. The challenge is how to use the

Amazon/DivX
Exhibit 1007
Page 11

information from the input video bit-stream to reduce the complexity of a transcoder and improve the quality of the output video. Video transcoding is also related to the research activities on compressed-domain video processing, since the incoming video stream to a transcoder is in the compressed-domain. As new video coding standards continue to be developed, the need for transcoding will continue to exist. For example, currently, MPEG-2 video is widely used in digital TV, DVD, and HDTV applications. However, newer coding standards such as H.264/MPEG-4 AVC have been developed which perform much better than MPEG-2. A transcoder will be useful for solving format incompatibility problems for universal multimedia access.

ACKNOWLEDGMENT

The authors would like to thank Dr. A. Vetro and Prof. S.-F. Chang for their valuable suggestions to improve this manuscript.

REFERENCES

[1] R. Mohan, J. R. Smith, and C.-S. Li, "Adapting multimedia Internet content for universal access," *IEEE Trans. Multimedia*, vol. 1, no. 1, pp. 104–114, Mar. 1999.

[2] S.-F. Chang and A. Vetro, "Video adaptation: Concepts, technologies, and open issues," *Proc. IEEE*, vol. 93, no. 1, pp. 148–158, Jan. 2005.

[3] Y. Wu, A. Vetro, H. Sun, and S. Y. Kung, "Intelligent multi-hop video communications," presented at the IEEE Pacific-Rim Conf. Multimedia, Beijing, China, 2001.

[4] J. Xin, M. T. Sun, and K.-S. Kan, "Bit allocation for joint transcoding of multiple MPEG coded video streams," in *Proc. IEEE Int. Conf. Multimedia and Expo*, 2001, pp. 8–11.

[5] "Information technology—Multimedia content description interface—Part 5: Multimedia description schemes (MPEG-7 MDS)," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC CD 15 938-5, 1st ed., 2001.

[6] P. M. Kuhn, T. Suzuki, and A. Vetro, "MPEG-7 transcoding hints for reduced complexity and improved quality," presented at the International Packet Video Workshop 2001, Kyongju, Korea.

[7] A. Vetro, H. Sun, and Y. Wang, "Object-based transcoding for adaptable video content delivery," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 387–401, Mar. 2001.

[8] S. Wee, "Reversing motion vector fields," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, 1998, pp. 209–212.

[9] S. Wee and B. Vasudev, "Compressed-domain reverse play of MPEG video streams," in *Proc. SPIE Int. Symp. Voice, Video, and Data Communications*, 1998, pp. 237–248.

[10] Y.-P. Tan, Y.-Q. Liang, and J. Yu, "Video transcoding for fast forward/reverse video playback," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 2002, pp. 713–716.

[11] S.-F. Chang, "Optimal video adaptation and skimming using a utility-based framework," presented at the Tyrrhenian Int. Workshop Digital Communications, Capri Island, Italy, 2002.

[12] J.-G. Kim, Y. Wang, and S.-F. Chang, "Content-based utility function prediction for real-time MPEG-4 video transcoding," in *Proc. IEEE Int. Conf. Image Processing*, 2003, pp. 189–192.

[13] P. Yin, A. Vetro, and B. Liu, "Rate-distortion models for video transcoding," in *Proc. SPIE Conf. Image Video Communications Processing*, 2003, pp. 479–488.

[14] M. Ghanbari, "Two-layer coding of video signals for VBR networks," *IEEE J. Sel. Areas Commun.*, vol. 7, no. 5, pp. 771–781, Jun. 1989.

[15] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 301–317, Mar. 2001.

[16] "Applications and requirements for scalable video coding," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC/JTC1/SC29/WG11/N5540, Mar. 2003.

[17] "Information technology—Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s—Part 2: Video (MPEG-1 video)," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 11 172-2, 1st ed., 1993.

[18] "Information technology—Generic coding of moving pictures and associated audio information: Video (MPEG-2 video)," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 13 818-2, 2nd ed., 2000.

[19] "Coding of audio-visual objects—Part 2: Visual (MPEG-4 video)," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 14 496-2:2001, 2nd ed., 2001.

[20] "Video coding for low bit rate communication," Int. Telecommun. Union-Telecommun. (ITU-T), Geneva, Switzerland, Recommendation H.263, 1998.

[21] "Draft text of final draft international standard for advanced video coding," Int. Telecommun. Union-Telecommun. (ITU-T), Geneva, Switzerland, Recommendation H.264 (draft), Mar. 2003.

[22] A. Vetro, C. Christopulos, and H. Sun, "Video transcoding architectures and techniques: An overview," *IEEE Signal Process. Mag.*, vol. 20, no. 2, pp. 18–29, Mar. 2003.

[23] A. Eleftheriadis and D. Anastassiou, "Constrained and general dynamic rate shaping of compressed digital video," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1995, pp. 396–399.

[24] H. Sun, W. Kwok, and J. W. Zdepski, "Architectures for MPEG compressed bitstream scaling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 2, pp. 191–199, Apr. 1996.

[25] Y. Nakajima, H. Hori, and T. Kanoh, "Rate conversion of MPEG coded video by re-quantization process," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1995, pp. 408–411.

[26] J. Youn, M.-T. Sun, and J. Xin, "Video transcoder architectures for bit rate scaling of H.263 bit streams," in *Proc. ACM Multimedia*, Nov. 1999, pp. 243–250.

[27] D. G. Morrison, M. E. Nilson, and M. Ghanbari, "Reduction of the bit-rate of compressed video while in its coded form," in *Proc. 6th Int. Workshop Packet Video*, 1994, pp. D17.1–D17.4.

[28] G. Keesman, R. Hellinghuizen, F. Hoeksema, and G. Heideman, "Transcoding of MPEG bitstreams," *Signal Process. Image Commun.*, vol. 8, no. 6, pp. 481–500, Sep. 1996.

[29] P. A. A. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bitrate reduction of MPEG-2 bit streams," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 8, pp. 953–967, Dec. 1998.

[30] J. Youn and M.-T. Sun, "Video transcoding with H.263 bit streams," *J. Visual Commun. Image Represent.*, vol. 11, pp. 385–404, Dec. 2000.

[31] T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding to lower spatial-temporal resolutions and different encoding formats," *IEEE Trans. Multimedia*, vol. 2, no. 2, pp. 101–110, Jun. 2000.

[32] S.-F. Chang and D. G. Messerschmitt, "Manipulation and compositing of MC-DCT compressed video," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 1, pp. 1–11, Jan. 1995.

[33] N. Merhav and V. Bhaskaran, "Fast algorithms for DCT-domain image downsampling and for inverse motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 3, pp. 468–476, Jun. 1997.

[34] J. Song and B.-L. Yeo, "A fast algorithm for DCT-domain inverse motion compensation based on shared information in a macroblock," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 5, pp. 767–775, Aug. 2000.

[35] C.-W. Lin and Y.-R. Lee, "Fast algorithms for DCT-domain video transcoding," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 2001, pp. 421–424.

[36] S. Acharya and B. Smith, "Compressed domain transcoding of MPEG," in *Proc. IEEE Int. Conf. Multimedia Computing and Systems*, 1998, pp. 295–304.

[37] S. Liu and A. C. Bovik, "Local bandwidth constrained fast inverse motion compensation for DCT-domain video transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 5, pp. 309–319, May 2000.

[38] W. Zhu, K. Yang, and M. Beacken, "CIF-to-QCIF video bitstream down-conversion in the DCT domain," *Bell Labs. Tech. J.*, vol. 3, no. 3, pp. 21–29, Jul.–Sep. 1998.

[39] R. Dugad and N. Ahuja, "A fast scheme for image size change in the compressed domain," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 11, no. 4, pp. 461–474, Apr. 2001.

[40] Y.-R. Lee, C.-W. Lin, and Y.-W. Chen, "Computation reduction in cascaded DCT-domain video downscaling transcoding," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2003, pp. 860–863.

[41] N. Bjork and C. Christopoulos, "Transcoder architecture for video coding," *IEEE Trans. Consum. Electron.*, vol. 44, no. 1, pp. 88–98, Feb. 1998.

Amazon/DivX
Exhibit 1007
Page 12

[42] S. Wee, J. G. Apostolopoulos, and N. Feamster, "Field-to-frame transcoding with spatial and temporal downsampling," in *Proc. IEEE Int. Conf. Image Processing*, vol. 4, 1999, pp. 271–275.
[43] B. Shen, I. K. Ishwar, and V. Bhaskaran, "Adaptive motion-vector re-sampling for compressed video downscaling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 6, pp. 929–936, Sep. 1999.
[44] S.-H. Jang and N. S. Jayant, "An adaptive nonlinear motion vector resampling algorithm for down-scaling video transcoding," in *Proc. IEEE Conf. Multimedia and Expo*, vol. 2, 2003, pp. 229–232.
[45] G. Shen, B. Zeng, Y.-Q. Zhang, and M.-L. Liou, "Transcoder with arbitrarily resizing capability," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 5, 2001, pp. 25–28.
[46] J. Xin, M.-T. Sun, B. S. Choi, and K. W. Chun, "An HDTV to SDTV spatial transcoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 998–1008, Nov. 2002.
[47] J. Xin, M.-T. Sun, and T.-D. Wu, "Motion vector composition for MPEG-2 to MPEG-4 video transcoding," in *Proc. Workshop and Exhibition MPEG-4*, 2002, pp. 9–12.
[48] J. Youn, M.-T. Sun, and C.-W. Lin, "Motion vector refinement for high-performance transcoding," *IEEE Trans. Multimedia*, vol. 1, no. 1, pp. 30–40, Mar. 1999.
[49] M.-J. Chen, M.-C. Chu, and C.-W. Pan, "Efficient motion estimation algorithm for reduced frame-rate video transcoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 4, pp. 269–275, Apr. 2002.
[50] J.-N. Hwang, T.-D. Wu, and C.-W. Lin, "Dynamic frame-skipping in video transcoding," in *Proc. IEEE Workshop Multimedia Signal Processing*, 1998, pp. 616–621.
[51] K.-D. Seo and J.-K. Kim, "Motion vector refinement for video downsampling in the DCT domain," *IEEE Signal Process. Lett.*, vol. 9, no. 11, pp. 356–359, Nov. 2002.
[52] P. Yin, M. Wu, and B. Liu, "Video transcoding by reducing spatial resolution," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, Sep. 2000, pp. 972–975.
[53] P. Yin, A. Vetro, H. Sun, and B. Liu, "Drift compensation architectures and techniques for reduced resolution transcoding," *Proc. SPIE, Visual Commun. Image Process.*, vol. 4671, pp. 180–191, Jan. 2002.
[54] P. Yin, A. Vetro, B. Liu, and H. Sun, "Drift compensation for reduced spatial resolution transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 11, pp. 1009–1020, Nov. 2002.
[55] A. Vetro, T. Hata, N. Kuwahara, H. Kalva, and S. Sekiguchi, "Complexity-quality analysis of transcoding architectures for reduced spatial resolution," *IEEE Trans. Consum. Electron.*, vol. 48, no. 3, pp. 515–521, Aug. 2002.
[56] K.-T. Fung, Y.-L. Chan, and W.-C. Siu, "New architecture for dynamic frame-skipping transcoder," *IEEE Trans. Image Process.*, vol. 11, no. 8, pp. 886–900, Aug. 2002.
[57] T. Shanableh and M. Ghanbari, "Hybrid DCT/pixel domain architecture for heterogeneous video transcoding," *Signal Process. Image Commun.*, vol. 18, no. 8, pp. 601–620, Sep. 2003.
[58] J. Xin, M.-T. Sun, and K. Chun, "Motion re-estimation for MPEG-2 to MPEG-4 simple profile transcoding," presented at the Int. Packet Video Workshop, Pittsburgh, PA, 2002.
[59] J. Xin, "Improved standard-conforming video transcoding techniques," Ph.D. dissertation, Univ. Washington, Seattle, 2002.
[60] H. Jozawa, K. Kamikura, A. Sagata, H. Kotera, and H. Watanabe, "Two-stage motion compensation using adaptive global MC and local affine MC," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 7, no. 1, pp. 75–85, Feb. 1997.
[61] "MPEG-4 video verification model version 18.0," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC JTC1/SC29/WG11, Jan. 2001.
[62] F. Dufaux and J. Konrad, "Efficient, robust and fast global motion estimation for video coding," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 497–501, Mar. 2003.
[63] Y. Su, M.-T. Sun, and V. Hsu, "Global motion estimation from coarsely sampled motion vector field and the applications," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 2, 2003, pp. 628–631.
[64] Y.-C. Lin, C.-N. Wang, T. Chiang, A. Vetro, and H. Sun, "Efficient FGS-to-single layer transcoding," in *Proc. IEEE Int. Conf. Consumer Electronics*, 2002, pp. 134–135.
[65] Y.-P. Tan and Y.-Q. Liang, "Methods and need for transcoding MPEG-4 fine granularity scalability video," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 4, 2002, pp. 719–722.
[66] E. Barrau, "MPEG video transcoding to a fine-granular scalable format," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 2002, pp. 717–720.
[67] S. Dogan, A. H. Sadka, and A. M. Kondoz, "Efficient MPEG-4/H.263 video transcoder for interoperability between heterogeneous multimedia networks," *Electron. Lett.*, vol. 35, no. 11, pp. 863–864, May 1999.
[68] H. Kato, H. Yanagihara, Y. Nakajima, and Y. Hatori, "A fast motion estimation algorithm for DV to MPEG-2 conversion," in *Proc. IEEE Int. Conf. Consumer Electronics*, 2002, pp. 140–141.
[69] N. Feamster and S. Wee, "An MPEG-2 to H.263 transcoder," presented at the SPIE Int. Symp. Voice, Video, and Data Communication Conf., Boston, MA, 1999.
[70] J.-L. Wu, S.-J. Huang, Y.-M. Huang, C.-T. Hsu, and J. Shiu, "An efficient JPEG to MPEG-1 transcoding algorithm," *IEEE Trans. Consum. Electron.*, vol. 42, no. 3, pp. 447–457, Aug. 1996.
[71] A. Gersho and R. Gray, *Vector Quantization and Signal Compression*. Norwell, MA: Kluwer, 1991.
[72] O. Werner, "Requantization for transcoding of MPEG-2 intraframes," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 179–191, Feb. 1999.
[73] Z. Guo, O. C. Au, and K. B. Letaief, "Parameter estimation for image/video transcoding," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 2, 2000, pp. 269–272.
[74] H. Sorial, W. Lynch, and A. Vincent, "Selective requantization for transcoding of MPEG compressed video," in *Proc. IEEE Int. Conf. Multimedia and Expo.*, 2000, pp. 217–220.
[75] B. Shen, "Efficient deblocking and optimal quantizer selection for video transcoding," in *Proc. IEEE Int. Conf. Image Processing*, 2003, pp. 193–196.
[76] "Test Model 5," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC JTC1/SC29/WG11, N0400, Apr. 1993.
[77] L. Wang, A. Luthra, and B. Eifrig, "Rate control for MPEG transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 2, pp. 222–234, Feb. 2001.
[78] P. N. Tudor and O. H. Werner, "Real-time transcoding of MPEG-2 video bit streams," in *Proc. IEE Int. Broadcasting Conv.*, 1997, pp. 286–301.
[79] P. Assunção and M. Ghanbari, "Buffer analysis and control in CBR video transcoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 1, pp. 83–92, Feb. 2000.
[80] J. Xin, M.-T. Sun, and K. W. Chun, "Bit allocation for transcoding of pre-encoded video streams," *Proc. SPIE: Visual Commun. Image Process.*, vol. 4671, pp. 164–171, Jan. 2002.
[81] I. Koo, P. Nasiopoulos, and R. Ward, "Joint MPEG-2 coding for multi-program broadcasting of pre-recorded video," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 4, 1999, pp. 2227–2230.
[82] H. Sorial, W. E. Lynch, and A. Vincent, "Joint transcoding of multiple MPEG video bitstreams," in *Proc. IEEE Int. Symp. Circuits Systems*, 1999, pp. 251–254.
[83] H. Kasai, T. Hanamura, W. Kamayama, and H. Tominaga, "Rate control scheme for low-delay MPEG-2 video transcoder," in *Proc. IEEE Int. Conf. Image Processing*, 2000, pp. 964–967.
[84] M.-T. Sun, T.-D. Wu, and J.-N. Hwang, "Dynamic bit allocation in video combining for multipoint video conferencing," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 45, no. 5, pp. 644–648, May 1998.
[85] T.-D. Wu and J.-N. Hwang, "Dynamic bit rate conversion in multipoint video transcoding," in *Proc. IEEE Int. Conf. Image Processing*, 1999, pp. 817–821.
[86] C.-W. Lin, Y.-C. Chen, and M-.T. Sun, "Dynamic region of interest transcoding for multipoint video conferencing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 10, pp. 982–992, Oct. 2003.
[87] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.
[88] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Process.*, vol. 66, no. 3, pp. 283–301, May 1998.
[89] J. Meng and S. F. Chang, "Embedding visible video watermarks in the compressed domain," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 1998, pp. 474–477.
[90] J. Youn, J. Xin, and M.-T. Sun, "Fast video transcoding architectures for networked multimedia applications," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 4, 2000, pp. 25–28.
[91] K. Panusopone, X. Chen, and F. Ling, "Logo insertion in MPEG transcoder," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 2, 2001, pp. 981–984.
[92] G. de los Reyes, A. R. Reibman, S.-F. Chang, and J. C.-I. Chuang, "Error-resilient transcoding for video over wireless channels," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1063–1074, Jun. 2000.

Amazon/DivX
Exhibit 1007
Page 13

[93] S. Dogan, A. Cellatoglu, M. Uyguroglu, A. H. Sadka, and A. M. Kondoz, "Error-resilient video transcoding for robust internetwork communications using GPRS," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 453–564, Jun. 2002.

[94] R. Puri, K.-W. Lee, K. Ramchandran, and V. Bhargavan, "An integrated source transcoding and congestion control paradigm for video streaming in the internet," *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 18–32, Mar. 2001.

[95] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[96] R. Swann and N. Kingsbury, "Transcoding of MPEG-II for enhanced resilience to transmission errors," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, 1996, pp. 813–816.

[97] M. Xia, A. Vetro, B. Liu, and H. Sun, "Rate-distortion optimized bit allocation for error resilient video transcoding," in *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 3, 2004, pp. III-945–III-948.

[98] T.-C. Wang, H.-C. Fang, and L.-G. Chen, "Low delay and error robust wireless video transmission for video communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 12, pp. 1049–1058, Dec. 2002.

Jun Xin (Member, IEEE) received the B.S. degree in electrical engineering from Southeast University, Nanjing, China, in 1993, the M.S. degree in electrical engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 1996, and the Ph.D. degree in electrical engineering from the University of Washington, Seattle, WA, in 2002.

From 1996 to 1998, he was a Software Engineer with Motorola-ICT Joint R&D Laboratory, Beijing. From January 2003 to August 2003, he was a Senior Software Engineer with Broadware Technologies, Cupertino, CA. Since 2003, he has been with Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, as a Member of Technical Staff. His research interests include digital video processing and multimedia communication.

Chia-Wen Lin (Senior Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, R.O.C., in 1992 and 2000, respectively.

He was a Section Manager of the Customer Premise Equipment (CPE) and Access Technologies Department, Computer and Communications Research Laboratories (CCL), Industrial Technology Research Institute (ITRI), Taiwan, R.O.C. From April 2000 to August 2000, he was a Visiting Research Scholar with Information Processing Laboratory, Department of Electrical Engineering, University of Washington, Seattle. In August 2000, he joined the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan, R.O.C., where he is currently an Assistant Professor. From July 2002 to August 2002, he was also a Visiting Professor with Microsoft Research Asia, Beijing, China. He has authored or coauthored over 50 technical papers. He holds ten patents with more pending. His research interests include video coding and networked multimedia technologies.

Dr. Lin was the recipient of the 2000 Research Achievement Award presented by ITRI. He was also the recipient of the 2000 and 2001 Best Ph.D. Thesis Awards presented by the Acer Foundation and the Ministry of Education, Taiwan, R.O.C., respectively.

Ming-Ting Sun (Fellow, IEEE) received the B.S. degree from National Taiwan University, Taipei, in 1976, the M.S. degree from the University of Texas, Arlington, TX, in 1981, and the Ph.D. degree from the University of California, Los Angeles, in 1985, all in electrical engineering.

He was the Director of the Video Signal Processing Research Group at Bellcore (now Telcordia). He joined the University of Washington, Seattle,in August 1996 where he is now a Professor. He holds nine patents and has published over 140 technical papers, including ten book chapters in the area of video technology.

Prof. Sun received an Award of Excellence from Bellcore for his work on the digital subscriber line in 1987. He received the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (TCSVT) Best Paper Award in 1993 and an IEEE Circuits and Systems Society Golden Jubilee Medal in 2000. He was a Distinguished Lecturer of the Circuits and Systems Society from 2000 to 2001. He was the General Cochair of the Visual Communications and Image Processing 2000 Conference. From 1988 to 1991, he was the Chairman of the IEEE Circuits and Systems Standards Committee and established the IEEE Inverse Discrete Cosine Transform Standard. He was the Editor-in-Chief of TCSVT from 1995 to 1997 and was the Editor-in-Chief of the IEEE TRANSACTIONS ON MULTIMEDIA (TMM) from 1999 to 2001.

Amazon/DivX
Exhibit 1007
Page 14