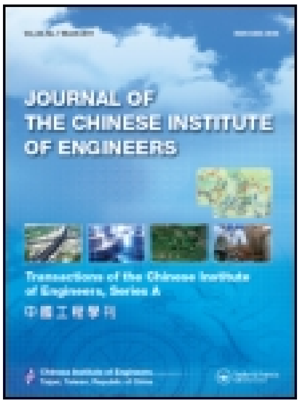

Journal of the Chinese Institute of Engineers
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tcie20

An efficient FGS to mpeg-1/2/4 single layer transcoder with R-D optimized multi-layer streaming technique for video quality improvement

Shih-Hao Wang [a] , Wei-Lin Chen [a] & Tihao Chiang [b]
[a] Institute of Electronics , National Chiao Tung University , Hsinchu 300, Taiwan
[b] Institute of Electronics , National Chiao Tung University , Hsinchu 300, Taiwan Phone: 886-3-5731558 Fax: 886-3-5731558 E-mail:
Published online: 04 Mar 2011.

To cite this article: Shih-Hao Wang , Wei-Lin Chen & Tihao Chiang (2007) An efficient FGS to mpeg-1/2/4 single layer transcoder with R-D optimized multi-layer streaming technique for video quality improvement, Journal of the Chinese Institute of Engineers, 30:6, 1059-1070, DOI: 10.1080/02533839.2007.9671332

To link to this article: http://dx.doi.org/10.1080/02533839.2007.9671332

Journal of the Chinese Institute of Engineers, Vol. 30, No. 6, pp. 1059-1070 (2007)

AN EFFICIENT FGS TO MPEG-1/2/4 SINGLE LAYER TRANSCODER WITH R-D OPTIMIZED MULTI-LAYER STREAMING TECHNIQUE FOR VIDEO QUALITY IMPROVEMENT

Shih-Hao Wang, Wei-Lin Chen, and Tihao Chiang*

ABSTRACT

The need for interoperability poses a challenge with the widespread use of multiple video formats. A Multi Layer Streaming-Simplified DCT Domain Transcoder (MLS-SDDT) is proposed to achieve efficient transcoding and video quality improvement. However, an efficient architecture is needed to handle drifting errors. The MLS-SDDT algorithm addresses these issues by proposing: (1) an FGS compatible Simplified DCT Domain Transcoder (FGS-SDDT) architecture for MPEG-1/2/4 single layer transcoding, and (2) an R-D optimized multi-layer streaming technique for video quality improvement. To support efficient transcoding from FGS to MPEG-1/2/4, the FGS-SDDT adopts low complexity Simplified DCT Domain Transcoder (SDDT) architecture as an FGS compatible SDDT. To resolve the serious error-drifting problem from SDDT, especially for heterogeneous transcoding, a mathematical analysis of the error-drifting problem is provided and a solution by multi-layer streaming is adopted. The multi-layer streaming approach pre-computes the errors according to the mathematical analysis and encodes the errors as the second enhancement layer for error compensation. For different network conditions, an R-D optimized model is used to improve the bit allocation in the two enhancement layers for better R-D performance. Our experiments show that the MLS-SDDT can deliver 1.4~7.0 dB PSNR improvement for MPEG-1 and 1.9~8.6 dB improvement for MPEG-2 as compared to the SDDT. For FGS to MPEG-4 single layer transcoding, the MLS-SDDT can achieve minor PSNR loss (0.0 ~ 0.4 dB) with lower computational complexity as compared to the SDDT approach.

Key Words: transcoding, MPEG-4 FGS, MPEG-1/2/4, multi-layer streaming, R-D optimization.

I. INTRODUCTION

Multi-format video transcoding demand has increased with the popularity of Internet video. In MPEG-21, Universal Multimedia Access (UMA) has provided a platform for multimedia content delivery which can be accessed by anyone anywhere (Mohan *et al*., 1999). Due to numerous video bitstream formats such as MPEG-1/2/4, interoperability becomes difficult. Thus, a video transcoder which is capable of converting source video formats to target video formats has increased importance.

Transcoding architectures can be roughly categorized into four types as shown in Table 1 (Xin *et al*., 2005). The first type is Decoder-Encoder (DEC-ENC) transcoding architecture that cascades a full decoder and a full encoder. The second type is Cascaded Pixel Domain Transcoding (CPDT) that is a simplified DEC-ENC architecture with motion vectors (MV) and reuse (Bjork and Christopoulos, 1998; Shanableh and Ghanbari, 2000; Shen *et al*., 1999; Sun *et al*., 1996;

*Corresponding author. (Tel: 886-3-5731558; Fax: 886-3-5731791; Email: tchiang@mail.nctu.edu.tw)

The authors are with the Institute of Electronics, National Chiao Tung University, Hsinchu 300, Taiwan.

Table 1 The comparison of four commonly used transcoding architectures

Type	DEC-ENC	CPDT	CDDT	SDDT
Features	Full encoder and decoder	Reuse of MV information	Transcoding in DCT domain	Single MC architecture
Operational domain	Pixel	Pixel	DCT	DCT
Speedup[1]	1.0	2.72 ~ 3.23	4.41 ~ 6.54	7.77 ~ 9.67
PSNR loss[1]	0.0	0.0 ~ 0.1	0.2 ~ 0.3	0.2 ~ 0.3
Error drifting source	No	No	Arithmetic	Arithmetic + prediction

[1]As compared to the DEC-ENC (Xin *et al*., 2005)

Youn *et al*., 1999). The third type is Cascaded DCT Domain Transcoding (CDDT) that converts the data operations of CPDT from pixel domain to DCT domain (Assuncao and Ghanbari, 1998; Chang and Messerschmitt, 1995; Merhav and Bhaskaran, 1997; Song and Yeo, 2000). One feature is that motion compensation (MC) is replaced with a new MC algorithm in the DCT domain to save unnecessary DCT and IDCT operations. The fourth type is Simplified DCT Domain Transcoding (SDDT) that adopts the single MC architecture in CDDT to share this MC for the encoder and decoder (Assuncao and Ghanbari, 1998; Keesman *et al*, 1996; Morrison *et al*, 1996). Among these four architectures, DEC-ENC and CPDT have the greatest flexibilities and smallest error-drifting problems, but suffer from huge computational complexity. SDDT has the smallest computational complexity, but it suffers from minor PSNR loss for homogeneous transcoding, and a serious error-drifting problem due to the prediction mismatches from using the same MC for heterogeneous transcoding.

In this paper, a Multi Layer Streaming-Simplified DCT Domain Transcoder (MLS-SDDT) is proposed to achieve efficient transcoding and video quality improvement. The MLS-SDDT contains the FGS-SDDT to achieve efficient FGS to MPEG-1/2/4 transcoding. For multiple formats video transcoding, FGS is selected as the bitstream format at the server end because its bitstream can be truncated at any point to match the network bandwidth. MPEG-1, MPEG-2 and MPEG-4 are selected as the target bitstream formats at the client end as they have been widely used in consumer terminals such as VCD players, DVD players, cell phones, etc. To enable efficient transcoding of FGS to single layer MPEG-1/2/4, the FGS-SDDT adopts the SDDT architecture which is a very low complexity transcoder with 7.77 ~ 9.67 times speedup as compared to the DEC-ENC (Xin *et al*., 2005).

To resolve the serious error-drifting problem for heterogeneous transcoding (FGS to MPEG-1 and FGS to MPEG-2) using SDDT, we mathematically analyze the error-drifting sources including arithmetic and prediction mismatches. Since the prediction mismatch is the major error-drifting source, we propose a multi-layer streaming technique for the error compensation. The multi-layer streaming approach reproduces the prediction mismatch and encodes it as the second enhancement layer at the server end. For different network conditions, a Rate-Distortion (R-D) optimized model is used to improve the bit allocation in two enhancement layers for better R-D performance. Our experiments show that the R-D optimized multi-layer streaming can deliver 3.6~9.6dB PSNR improvement as compared to the SDDT for heterogeneous transcoding.

The contributions of this work include:

- Efficient FGS-SDDT architecture for FGS to MPEG-1/2/4 single layer transcoding: We develop our multi-format transcoder based on the low complexity of SDDT architecture. To support the FGS bitstream in the SDDT architecture, we derive a new transcoding architecture called FGS-SDDT that integrates FGS decoder into the SDDT.
- Significant R-D performance improvement over the SDDT: The SDDT sacrifices R-D performance for low computational complexity. To resolve this issue, we mathematically analyze the error-drifting problem and propose a multi-layer streaming technique with a new enhancement layer for mismatch compensation. For better R-D performance under limited channel bandwidth, an R-D model is used to re-arrange the bit allocation for the two enhancement layers so as to deliver better R-D performance. Our experiments show the R-D optimized multi-layer streaming can deliver 1.9-8.6 dB improvement as compared to the SDDT for heterogeneous transcoding.

The remainder of this paper is organized as follows. In Section II, we present the architecture of FGS-SDDT for the FGS to MPEG-1/2/4 single-layer transcoding and the mathematical analysis for the error-drifting problem. According to the analysis, a multi-layer streaming technique and an R-D model are proposed in Section III to resolve the error-drifting problem and improve the video quality. In Section IV, the experiments show significant PSNR improvement over SDDT. Section V gives concluding remarks and ideas for future work.

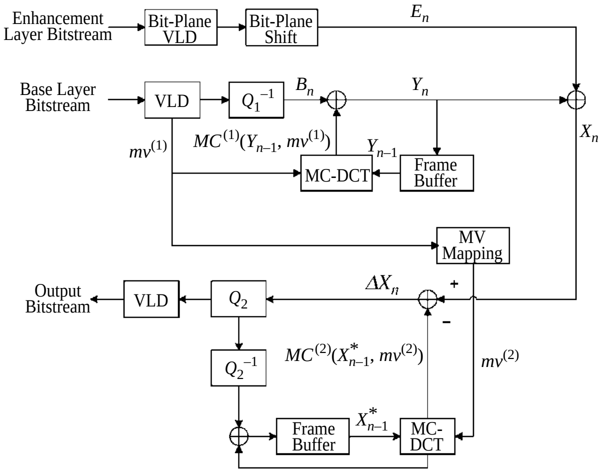

Fig. 1 FGS compatible CDDT transcoding architecture

II. EFFICIENT FGS TO MPEG-1/2/4 SINGLE LAYER TRANSCODER

1. FGS-SDDT Architecture

The FGS-SDDT approach is an FGS compatible SDDT architecture. To meet the low complexity requirement, the FGS-SDDT is developed based on the SDDT architecture. For FGS support, we integrate the SDDT decoder with an FGS decoder to make FGS-SDDT. Thus, the FGS-SDDT extends the single-layer to single-layer SDDT architecture to a multi-layer to single-layer transcoding architecture.

Like SDDT, the FGS-SDDT is a single MC architecture in the DCT domain for FGS multi-layer to MPEG-1/2/4 single layer transcoding. To survey prior works related to FGS multi-layer to single layer transcoding, Lin *et al*. (2002) proposed a single MC architecture for MPEG-4 in the pixel domain, but this architecture is incompatible with our DCT domain architecture. However, it provides a good reference for the integration of the FGS decoder. To develop our own FGS-SDDT architecture, we use similar derivations to build our own FGS-SDDT architecture in the DCT domain.

The derivations for the FGS-SDDT start from CDDT architecture with two MCs operations. Fig. 1 shows the functional block diagram for the FGS-to-single-layer CDDT. By following step by step derivations, we can construct our own single MC architecture from the two-MC architecture. The derivation steps for FGS-SDDT are as follows. The related symbol definitions throughout this paper are summarized in Table 2.

1). The reconstructed frame in FGS decoder X_n denoted as

Table 2 Symbol definitions

Symbols	Meaning
X_n	n-th decoded frame at decoder-loop
X_n^*	n-th reconstructed frame at encoder-loop
Y_n	n-th reconstructed frame at decoder-loop
ΔX_n	n-th residue frame at the encoder-loop
B_n	n-th base-layer residue frame
E_n	n-th enhancement-layer residue frame
d_a	Arithmetic error
d_p	Prediction error
R_E	Bitrate for first enhancement layer
R_ε	Bitrate for second enhancement layer
α	Bit allocation parameter
α_{opt}	Rate-distortion optimized bit allocation parameter
$MC^{(i)}(\cdot)$	Motion compensation (i = 0 for decoder and i = 1 for encoder)
$Q(\cdot)$, $IQ(\cdot)$	Quantization and inverse quantization.
$mv^{(i)}$	Motion vectors (i = 0 for decoder and i = 1 for encoder)
$D(\cdot)$	Distortion function
R_{BL}	Target transmission bitrate for base layer
R_{EL}	Target transmission bitrate for enhancement layer
R	Target transmission bitrate ($R_{BL} + R_{EL}$)

$$X_n = Y_n + E_n = (B_n + MC^{(1)}(Y_{n-1}, mv^{(1)})) + E_n \quad (1)$$

is composed of base layer reconstructed frame Y_n and the enhancement layer E_n. The base layer reconstructed frame Y_n is the sum of the motion compensated prediction $MC^{(1)}(Y_{n-1}, mv^{(1)})$ and the DCT coded residue B_n.

2). The residue ΔX_n to be entropy coded in the encoder is denoted as

$$\begin{aligned} \Delta X_n &= X_n - MC^{(2)}(X^*_{n-1}, mv^{(2)}) \\ &= (B_n + MC^{(1)}(Y_{n-1}, mv^{(1)}) + E_n) \\ &\quad - MC^{(2)}(X^*_{n-1}, mv^{(2)}) \end{aligned} \quad (2)$$

which is the difference between the reconstructed frame in the decoder loop X_n and the motion compensated prediction in the encoder $MC^{(2)}(X^*_{n-1}, mv^{(2)})$. Here, X^*_{n-1} is the reconstructed frame in the encoder and $mv^{(2)}$ represents re-mapped motion vectors. To substitute X_n in Eq. (2) using Eq. (1), we can further derive a new relationship for ΔX_n which is constructed by two motion compensated frames in both encoder and decoder.

3). To build a single MC architecture for FGS-SDDT, we need to merge the two MCs in Eq. (2). To make the merge possible, we re-write Eq. (2) which has two MCs as

$$\begin{aligned}\Delta X_n &= ((B_n + E_n) + (MC^{(1)}(Y_{n-1}, mv^{(1)}) \\ &\quad - MC^{(2)}(X^*_{n-1}, mv^{(2)}))) \\ &\quad + (MC^{(2)}(Y_{n-1}, mv^{(2)}) \\ &\quad - MC^{(2)}(Y_{n-1}, mv^{(2)})) \\ &= ((B_n + E_n) + (MC^{(2)}(Y_{n-1}, mv^{(2)}) \\ &\quad - MC^{(2)}(X^*_{n-1}, mv^{(2)}))) \\ &\quad + (MC^{(1)}(Y_{n-1}, mv^{(1)}) \\ &\quad - MC^{(2)}(Y_{n-1}, mv^{(2)})) \qquad (3)\end{aligned}$$

that shares the same MC with an additional error term $(MC^{(1)}(Y_{n-1}, mv^{(1)}) - (MC^{(2)}(Y_{n-1}, mv^{(2)})$. This error term can be eliminated under the linearity assumption of $mv^{(1)} = mv^{(2)}$ and $MC^{(1)}(X, mv^{(1)}) = MC^{(2)}(X, mv^{(2)})$. This assumption is true for homogeneous transcoding, but may not be true for heterogeneous transcoding. We will discuss this issue in the next Section. Under the linearity assumption, Eq. (3) can be further simplified as

$$\begin{aligned}&\Delta X_n \\ &= B_n + MC^{(2)}(Y_{n-1} - X^*_{n-1}, mv^{(2)}) + E_n \\ &= B_n + MC^{(2)}((X_{n-1} - E_{n-1}) - X^*_{n-1}, mv^{(2)}) + E_n \\ &= B_n + MC^{(2)}((X_{n-1} - X^*_{n-1}) - E_{n-1}, mv^{(2)}) + E_n \\ &= B_n + MC^{(2)}((\Delta X_{n-1} - \Delta X^*_{n-1}) - E_{n-1}, mv^{(2)}) + E_n \qquad (4)\end{aligned}$$

by merging the two MCs to be a single MC architecture. In this single MC architecture, we are not able to obtain both reconstructed frames Y_{n-1} and X^*_{n-1} at the same time. So, Y_{n-1} is further replaced by other available data in single MC architecture to be a new form of motion compensation from the error image $(\Delta X_{n-1} - \Delta X^*_{n-1}) - E_{n-1}$. The error image contains two components. The first component is the mismatch between quantized and un-quantized residue via Q_2 in Fig. 2. The second component is the removal of the effects from the enhancement layer to avoid reconstruction mismatches, because $mv^{(1)}$ is for the base layer only, while $mv^{(2)}$ is for reconstructed images with the base layer and the enhancement layer.

Figure 2 shows the FGS-SDDT architecture based on the derivations in Eq. (4). With this architecture, we can transcode an FGS bitstream to any format, MPEG-1, MPEG-2 or MPEG-4 bitstream by replacing the entropy coding (VLC) and quantization (Q_2) modules for the appropriate target format. This architecture also preserves the same complexity as compared to the SDDT. As compared to the CDDT in Fig. 1, it saves one MC and one frame buffer.

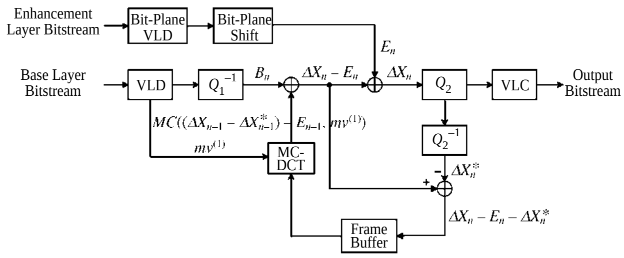

Fig. 2 FGS-SDDT transcoding architecture

2. Analysis of the Error-Drifting Problem

The FGS-SDDT is developed from SDDT, so it suffers from the same error-drifting problems as SDDT. The error-drifting problem is from the serious mismatch of MC prediction between two MCs in heterogeneous transcoding. For homogeneous transcoding, the two MC operations are the same, so this effect can be eliminated.

To analyze the error-drifting problem, the errors come from two possible sources, prediction error and arithmetic error. The prediction error is the mismatch of motion compensated predictions from the two different MC operations. In Eq. (3), we eliminate the error term $MC^{(1)}(Y_{n-1}, mv^{(1)}) - MC^{(2)}(Y_{n-1}, mv^{(2)})$ under the linearity assumption of $mv^{(1)} = mv^{(2)}$ and $MC^{(1)}(X, mv^{(1)}) = MC^{(2)}(X, mv^{(2)})$. This assumption is true for homogeneous transcoding since the two MCs operations are the same and we do not need MV re-mapping for $mv^{(2)}$. However, it will introduce errors for heterogeneous transcoding since $mv^{(1)}$ may be different from $mv^{(2)}$ and $MC^{(1)}(X, mv^{(1)})$ may be different from $MC^{(2)}(X, mv^{(2)})$. The inequality happens in our case that $MC^{(1)}(X, mv^{(1)})$ in the MPEG-4 standard supports four MV and unrestricted MV (UMV), but $MC^{(2)}(X, mv^{(2)})$ in MPEG-1 and 2 standards do not support such advanced motion prediction techniques. This error term is referred to as d_p where $d_p = MC^{(1)}(Y_{n-1}, mv^{(1)}) - MC^{(2)}(Y_{n-1}, mv^{(2)})$.

The other mismatch source is arithmetic errors. This type of error comes from arithmetic operations such as rounding, saturation, or fixed point approximation for floating point operations. These errors happen in computation of the term $MC^{(2)}((\Delta X_{n-1} - \Delta X^*_{n-1}) - E_{n-1}, mv^{(2)})$ in Eq. (4), and are referred to as d_a. This type of error happens for both homogeneous and heterogeneous transcoding, but typically this error has minor or negligible effects as compared to d_p.

In prior studies, solutions were proposed such as intra refreshment (Bjork and Christopoulos, 1998; Shanableh and Ghanbari, 2000; Shen *et al.*, 1999; Youn *et al.*, 1999) or MV refinement (Yin *et al.*, 2002) to stop error propagation. However, similar solutions can passively reduce the impact of error propagation, but cannot effectively stop the generation of errors. Since the major error-drifting problem is from prediction error (d_p), a solution to stop prediction errors is provided.

III. R-D OPTIMIZED MULTI-LAYER STREAMING

1. Error Compensation with Multi-Layer Streaming

The multi-layer streaming technique is used to resolve the error-drifting problem in heterogeneous transcoding. To analyze prediction error d_p, we find $MC^{(1)}(X, mv^{(1)})$ and $MC^{(2)}(X, mv^{(2)})$ are not available simultaneously in our single MC architecture. The $mv^{(2)}$ may also be different from $mv^{(1)}$ since MPEG-1 and MPEG-2 do not support four MV or UMV. So, our solution is to pre-compute d_p and encode it as the second enhancement layer in the FGS server for error compensation in heterogeneous transcoding. In homogeneous transcoding, d_p can be eliminated and thus the weighting for the second enhancement layer is set to zero.

The generation of a second enhancement layer relies on two assumptions, $MC^{(1)}(X, mv) = MC^{(2)}(X, mv)$ and the $mv^{(2)}$ re-mapping process known by the FGS server. The first assumption is true in our applications due to the same bi-linear pixel interpolation process for sub-pel motion search. The second assumption is also true because the server is aware of the $mv^{(2)}$ re-mapping process. So, the second enhancement layer which contains d_p is derived as

$$d_p = MC^{(1)}(Y_{n-1}, mv^{(1)}) - MC^{(1)}(Y_{n-1}, mv^{(2)}). \quad (5)$$

Figure 3 shows the MLS-SDDT architecture with the multi-layer streaming technique. Compared to the FGS-SDDT shown in Fig. 2, the MLS-SDDT contains an additional enhancement layer (also called error layer) for error compensation. With this auxiliary layer as side information, we can compensate for the prediction error d_p by bitplane decoding of the error layer bitstream. To consider unlimited channel bandwidth, we can fully resolve the error-drifting problem caused by the prediction error. In this MLS-SDDT, the advantage of a single frame buffer and a single MC in SDDT is still preserved for efficient transcoding, but it adds a bitplane VLD module to decode the error layer bitstream for mismatch compensation. The bitplane VLD module costs an additional complexity of 5-10% depending on the sequences and bitrates so the effect on the overall complexity is minor.

2. R-D Optimized Data Transmission

An R-D model is adopted for better bit allocation

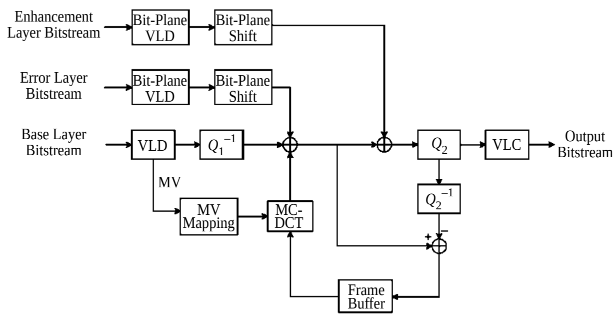

Fig. 3 MLS-SDDT transcoding architecture

of the two enhancement layers with limited channel bandwidth. To model the bit allocation problem, the criterion for the R-D model is denoted as

$$\alpha_{\text{opt}} = \arg \min_{\alpha \in [0,1]} D(\alpha) \text{ given } R. \quad (6)$$

Given the target bitrate R, we can find the minimal distortion $D(\alpha)$ under the bit allocation parameter α. The parameter α is defined as a mixed ratio of bits for first enhancement layer (R_{EL1}) over bits for second enhancement layer (R_{EL2}) and is denoted as

$$\alpha = \frac{R_{EL1}}{R_{EL}} = \frac{R_{EL1}}{R_{EL1} + R_{EL2}}. \quad (7)$$

To build the relationship between R_{EL} and α_{opt}, a statistical method is adopted by observing various sequences and bitrates. We simulate various combinations of R_{EL} and α as shown in Fig. 4. The range for R_{EL} is set below 2560 kilo-bps (kbps) with an interval of 256 kbps. The parameter α ranges from 0 to 1 with a step size of 0.05. We test four cases with base layer bitrates (R_{BL}) of 256, 512, 1024, and 2048 kbps. Four sequences including Foreman, Akiyo, Mobile, and Stefan in CIF format are used for the experiments with GOP structure of $N = 15$ and $M = 1$ (i.e., IPPP...). The distortion is measured by mean square error (MSE). In Fig. 4, the R-D curves modeled by various bitrates and α are plotted as dash lines. For brevity, only the results for $R_{BL} = 256$ kbps are shown. In Fig. 4, the best R-D curve is the one with the smallest MSE, and this curve is plotted as a solid line. Fig. 5 plots the optimized α values under different R_{EL} for the sequences in Fig. 4, and the statistically optimized curve is the average of the optimized α values for each sequence as shown in Fig. 6. From Fig. 6, we can find a monotonically increasing behavior for the statistically optimized curve where small α is preferred at lower bitrate and larger α is preferred at higher bitrate. The α is large because $R_{EL1} >> R_{EL2}$ at higher bitrate while α is small because $R_{EL1} \le R_{EL2}$ at lower bitrate.

To approximate the statistically optimized R-D

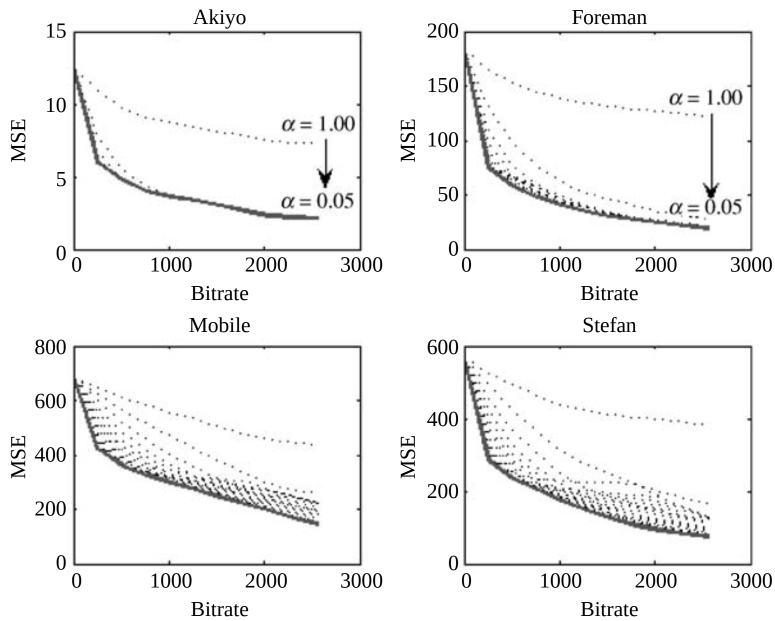

Fig. 4 R-D curves modeled by various bit allocation parameter α for R_{BL} = 256 kbps. The solid curve is the best R-D curve

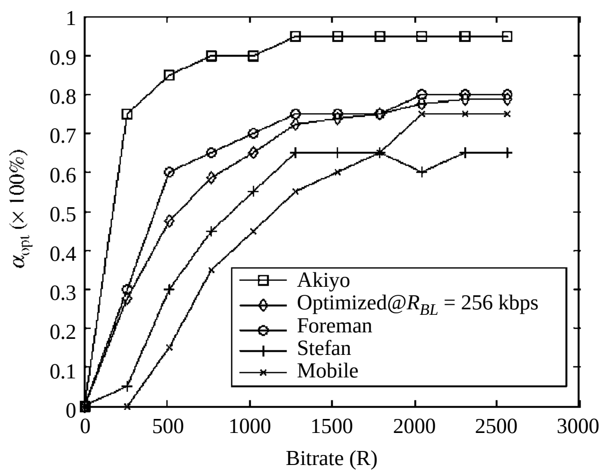

Fig. 5 The optimized bit allocation parameter α_{opt} by averaging the best α from each individual sequence for R_{BL} = 256 kbps

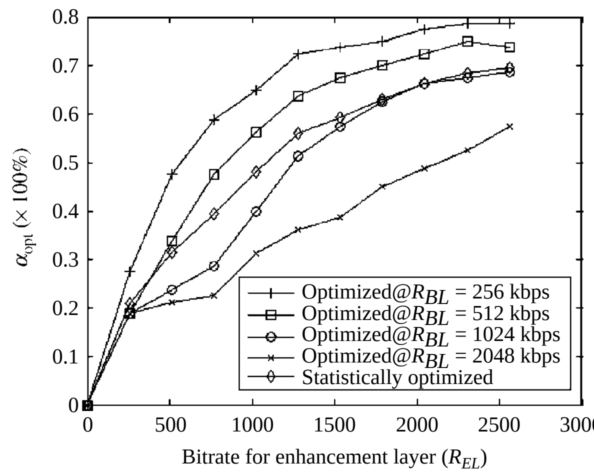

Fig. 6 Statistically optimized curve from four individually optimized curves for different base layer bitrates

curves with monotonically increasing properties as shown in Fig. 6, we examine four curve models including linear, power-law, quadratic, and exponential models as shown in Fig. 7. Table 3 shows the root mean square errors (RMSE) for approximation. There are a total of four sequences used for this test, but only 2 sequences are shown in Table 3 for brevity. From this Table, the power-law model with the smallest RMSE is selected.

The power-law curve is formulated as

$$\alpha_{\text{opt}} = aR^{b} + c, \tag{9}$$

where (a, b, c) is a set of model parameters. Then, we apply this power-law model to the statistically optimized curve as shown in Fig. 6 for parameter calculation, and plot the fitting curves as shown in Fig. 8. The set of parameter (a, b, c) = (0.3476, 0.1857, -0.7764) is adopted in our experiments of FGS transcoding to MPEG-1, MPEG-2 single layer.

IV. EXPERIMENTAL RESULTS

1. Rate-Distortion Performance

Three types of transcoding experiments including (1) MPEG-4 FGS to MPEG-1 (2) MPEG-4 FGS to MPEG-2 Main Profile, and (3) MPEG-4 FGS to

Table 3 Four models for R-D curves approximation. The distortion is measured by root mean square errors (RMSE)

RMSE	Linear	Power-Law	Quadratic	Exponential
Akiyo	0.04110	0.01178	0.01929	0.04230
Foreman	0.09279	0.01757	0.06005	0.10010

Table 4 The PSNR loss for MLS-SDDT as compared to the CPDT for FGS to MPEG-1 transcoding

PSNR Loss (dB)		CPDT	SDDT	MLS-SDDT	$\Delta PSNR_{(SDDT,\ MLS\text{-}SDDT)}$
Foreman	256 kbps	0.0	-5.3	-0.6	+4.7
	512 kbps	0.0	-6.5	-0.3	+6.2
	1024 kbps	0.0	-7.2	-0.4	+6.8
	2048 kbps	0.0	-7.4	-0.4	+7.0
Mobile	256 kbps	0.0	-2.9	-0.3	+2.6
	512 kbps	0.0	-3.4	-0.5	+2.9
	1024 kbps	0.0	-4.3	-0.8	+3.5
	2048 kbps	0.0	-4.5	-0.9	+3.6
Stefan	256 kbps	0.0	-4.9	-1.1	+3.8
	512 kbps	0.0	-5.7	-1.0	+4.7
	1024 kbps	0.0	-6.2	-0.8	+5.4
	2048 kbps	0.0	-6.8	-1.0	+5.8
Akiyo	256 kbps	0.0	-1.6	-0.1	+1.5
	512 kbps	0.0	-1.5	-0.1	+1.4
	1024 kbps	0.0	-1.5	-0.0	+1.5
	2048 kbps	0.0	-1.6	-0.2	+1.4

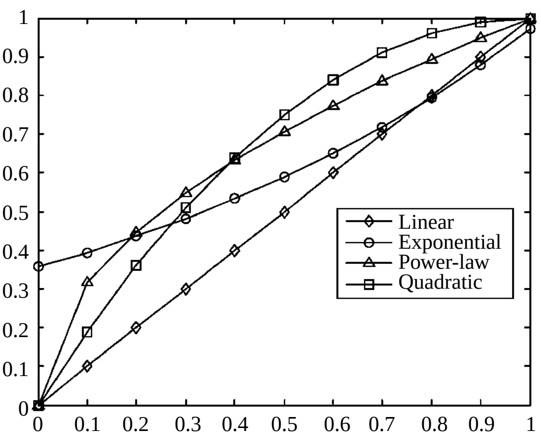

Fig. 7 Behaviors of four curve models

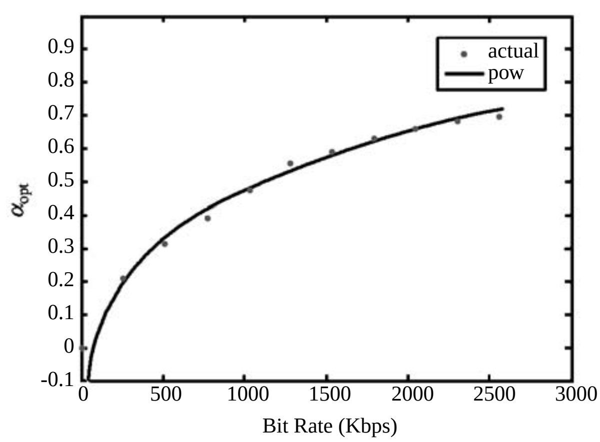

Fig. 8 Approximation of statistically optimized curve using power-law model

MPEG-4 Simple Profile are tested with the MLS-SDDT architecture. Four commonly used MPEG sequences including Foreman, Akiyo, Mobile, and Stefan in CIF format are used for the test. The bitstream is encoded with GOP structure of N = 15, M = 1 (IPPPP). To evaluate the effects of different base layer bitrates (R_{BL}), four R_{BL} including 256, 512, 1024, and 2048 kbps are adopted in the tests.

(i) MPEG-4 FGS to MPEG-1

Table 4 shows the comparison results of R-D performance for the three types of transcoding architectures, including CPDT, SDDT, and the MLS-SDDT. The CPDT approach is the reference upper bound for PSNR evaluation, and the SDDT is the reference lower bound. From this Table, the MLS-SDDT

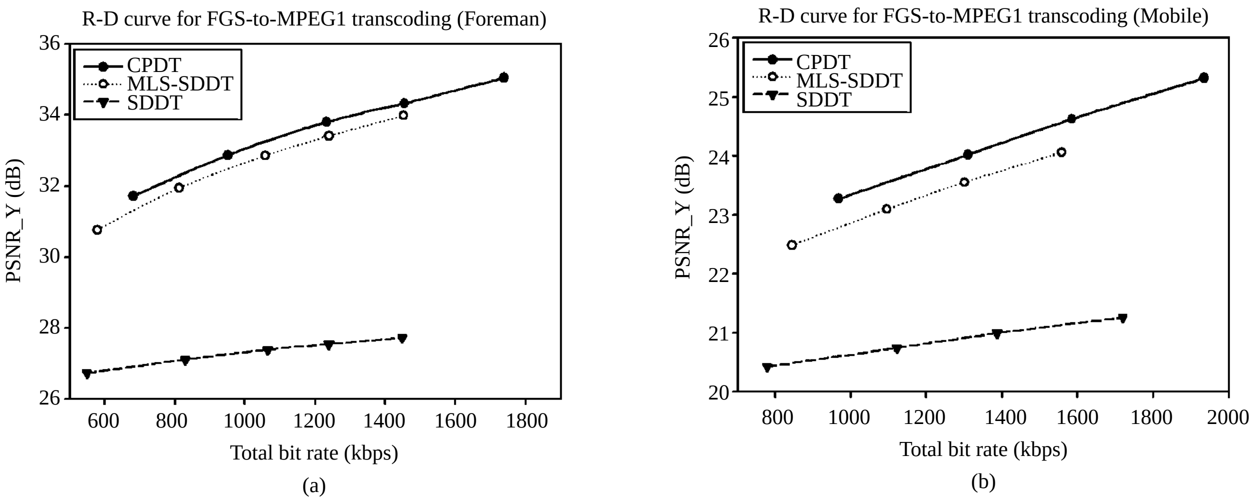

Fig. 9 R-D performance comparison for FGS-to-MPEG-1 transcoding with R_{BL} = 512 kbps. (a) Foreman (b) Mobile

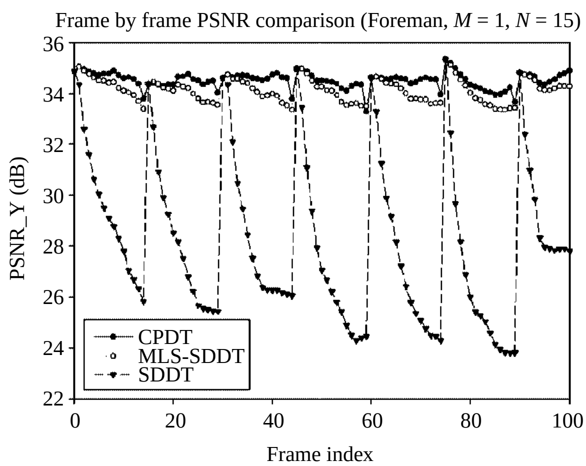

Fig. 10 Frame by frame comparison for FGS-to-MPEG-1 transcoding with GOP of M=1, N = 15 on Foreman sequence

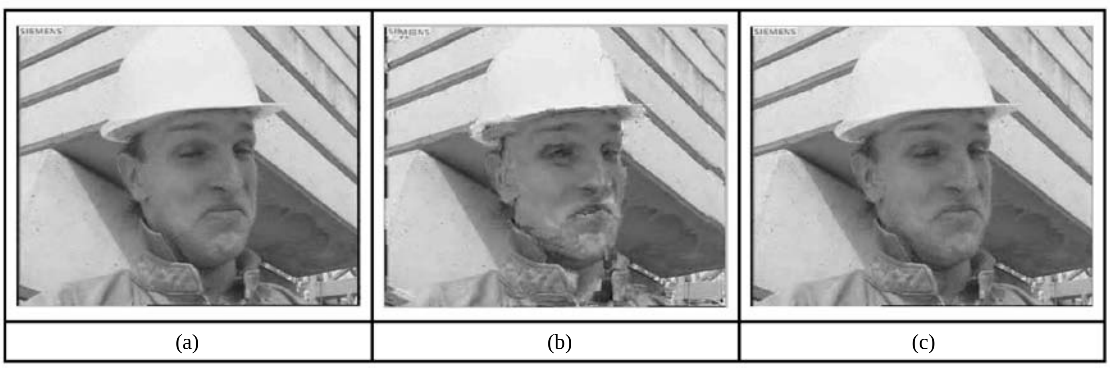

Fig. 11 Visual quality comparison for FGS-to-MPEG-1 transcoding at frame index of 73. (a) CPDT (b) SDDT (c) MLS-SDDT

can provide 1.4 to 7.0 dB PSNR improvement as compared to SDDT architecture for different sequences and different base layer bitrates. As compared to CPDT architecture, the quality loss for MLS-SDDT is 0.3 to 1.1 dB.

Figure 9 shows the R-D curves for Foreman sequence with R_{BL} = 512 kbps. From this Figure, significant PSNR improvement is observed as compared to the SDDT. Fig. 10 shows the frame by frame PSNR comparison, and Fig. 11 shows the visual quality

Table 5 The PSNR loss for MLS-SDDT as compared to the CPDT for FGS to MPEG-2 transcoding

PSNR Loss (dB)		CPDT	SDDT	MLS-SDDT	$\Delta PSNR_{(SDDT,\ MLS\text{-}SDDT)}$
Foreman	256 kbps	0.0	-6.5	-0.6	+5.9
	512 kbps	0.0	-8.2	-0.8	+7.4
	1024 kbps	0.0	-9.2	-1.0	+8.2
	2048 kbps	0.0	-9.6	-1.0	+8.6
Mobile	256 kbps	0.0	-4.0	-0.6	+3.4
	512 kbps	0.0	-4.6	-0.8	+3.8
	1024 kbps	0.0	-5.0	-0.7	+4.3
	2048 kbps	0.0	-5.8	-1.2	+4.6
Stefan	256 kbps	0.0	-6.6	-1.2	+5.4
	512 kbps	0.0	-7.3	-1.2	+6.1
	1024 kbps	0.0	-8.2	-1.1	+7.1
	2048 kbps	0.0	-8.7	-1.5	+7.2
Akiyo	256 kbps	0.0	-3.6	-1.2	+2.4
	512 kbps	0.0	-3.4	-1.1	+2.3
	1024 kbps	0.0	-3.3	-1.2	+2.1
	2048 kbps	0.0	-3.5	-1.6	+1.9

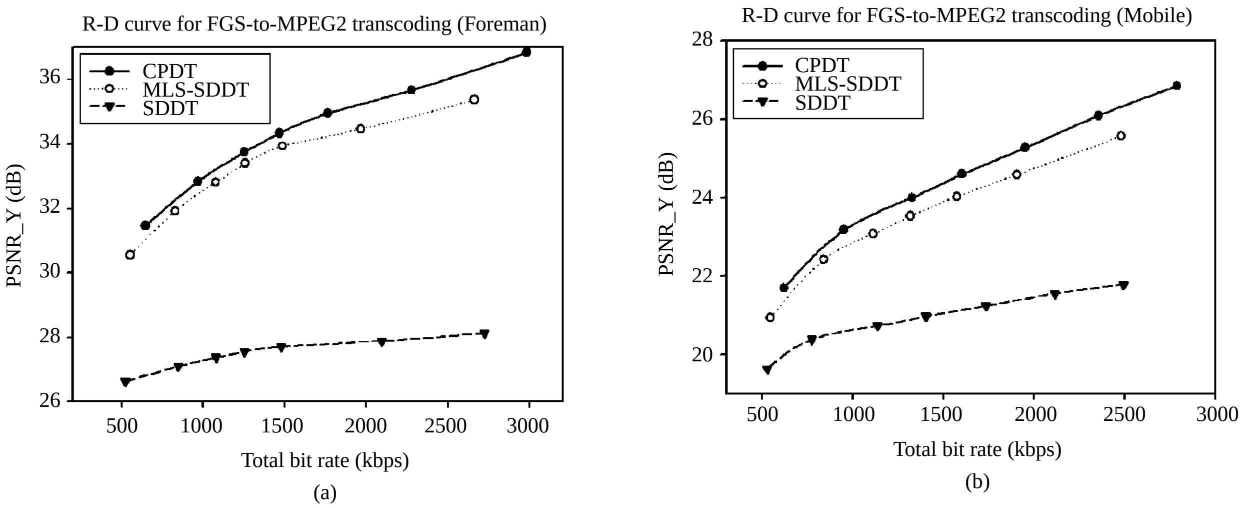

Fig. 12 R-D performance comparison for FGS-to-MPEG-2 transcoding with R_{BL} = 512 kbps. (a) Foreman (b) Mobile

comparison. From Fig. 10 and Fig. 11, the significant PSNR and quality improvement for the MLS-SDDT as compared to SDDT can be observed.

(ii) MPEG-4 FGS to MPEG-2 Main Profile

Table 5 shows the comparison results of R-D performance for the three types of transcoding architectures, including CPDT, SDDT, and the MLS-SDDT. Similar to the preceding subsection, CPDT is the reference upper bound and SDDT is the reference lower bound. From this Table, the MLS-SDDT can provide 1.9 to 8.6 dB PSNR improvement as compared to SDDT architecture. As compared to CPDT architecture, the quality loss for MLS-SDDT is 0.6 to 1.6 dB.

Figure 12 shows the R-D curves for Foreman sequence with R_{BL} = 512 kbps. From this Figure, we can find significant PSNR improvement from the SDDT architecture. Fig. 13 shows the frame by frame PSNR comparison and Fig. 14 shows the visual quality comparison. From Fig. 13 and Fig. 14, the significant PSNR and quality improvement for the MLS-SDDT as compared to SDDT can be observed.

(iii) MPEG-4 FGS to MPEG-4 Simple Profile

Table 6 shows the comparison results of R-D performance for three transcoding architectures, CPDT, Lin *et al*.'s work (2002), and the MLS-SDDT. CPDT is the reference upper bound for performance evaluation of the transcoders. Lin *et al*.'s work (2002)

Table 6 The PSNR loss for MLS-SDDT as compared to the CPDT for FGS to MPEG-4 Simple Profile transcoding

PSNR Loss (dB)		CPDT	Lin *et al*. (2002)	MLS-SDDT
Foreman	256 kbps	0.0	+0.0	+0.0
	512 kbps	0.0	+0.0	-0.2
	1024 kbps	0.0	-0.1	-0.4
	2048 kbps	0.0	+0.1	-0.2
Mobile	256 kbps	0.0	+0.0	+0.2
	512 kbps	0.0	+0.0	+0.2
	1024 kbps	0.0	+0.1	+0.1
	2048 kbps	0.0	+0.0	+0.0
Stefan	256 kbps	0.0	+0.0	+0.1
	512 kbps	0.0	+0.0	+0.0
	1024 kbps	0.0	+0.0	+0.0
	2048 kbps	0.0	+0.2	+0.0
Akiyo	256 kbps	0.0	+0.1	+0.0
	512 kbps	0.0	+0.0	+0.0
	1024 kbps	0.0	+0.0	+0.0
	2048 kbps	0.0	+0.0	-0.4

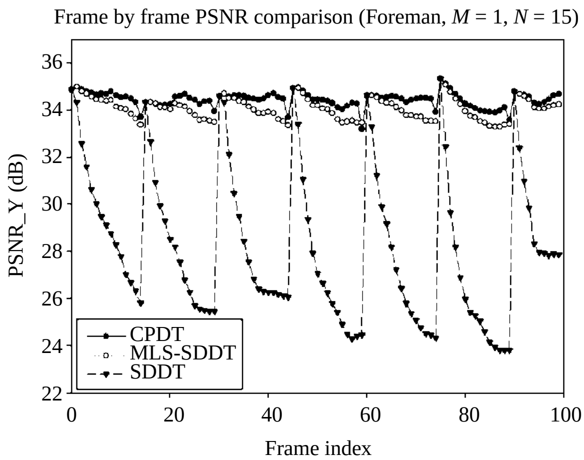

Fig. 13 Frame by frame comparison for FGS-to-MPEG-2 transcoding with GOP of M = 1, N = 15 on Foreman sequence

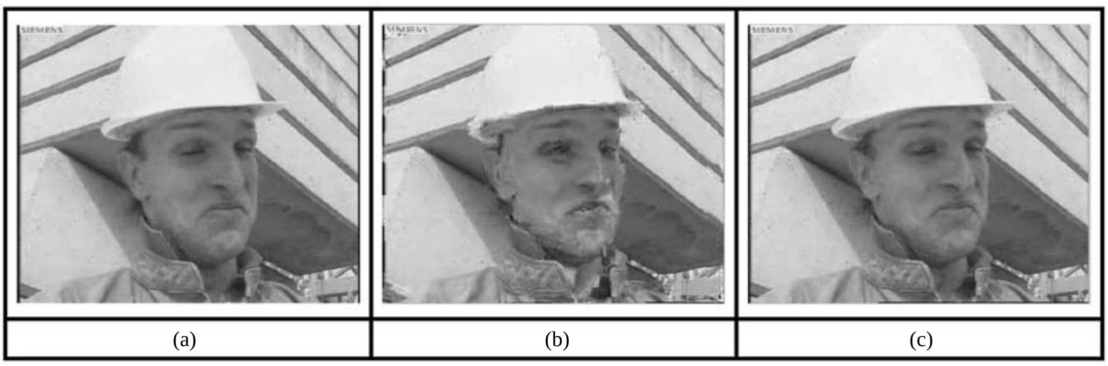

Fig. 14 Visual quality comparison for FGS-to-MPEG-2 transcoding at frame index of 73. (a) CPDT (b) SDDT (c) MLS-SDDT

has a similar single MC architecture in spatial domain for FGS to MPEG-4 single layer transcoding. From this Table, the MLS-SDDT can provide minor PSNR loss (0.0 ~ 0.4 dB) as compared to the other two transcoding architectures, but with the lowest computational complexity.

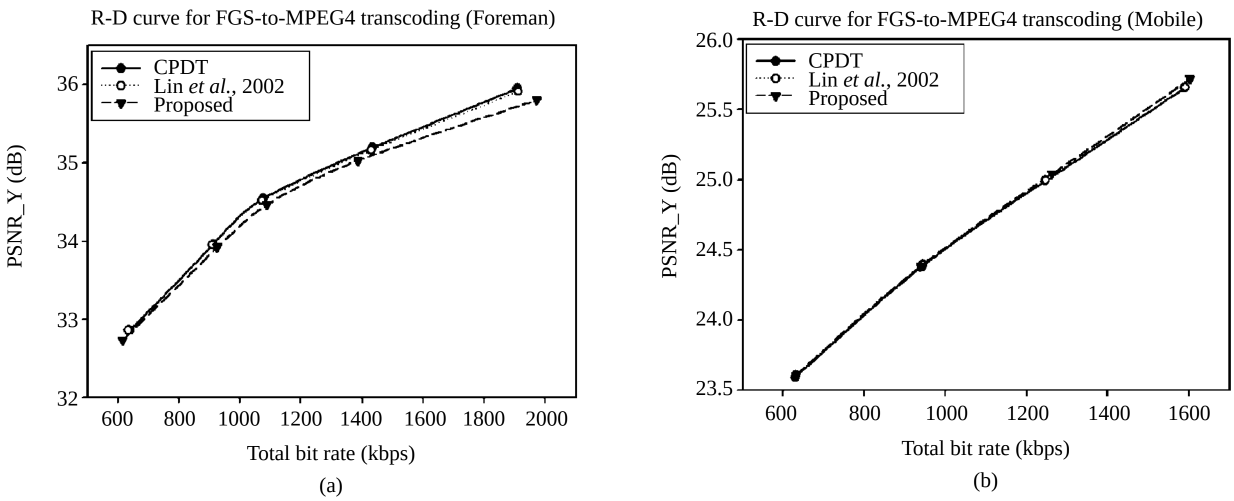

Fig. 15 R-D performance comparison for FGS-to-MPEG-4 Simple Profile transcoding with R_{BL} = 512 kbps. (a) Foreman (b) Mobile

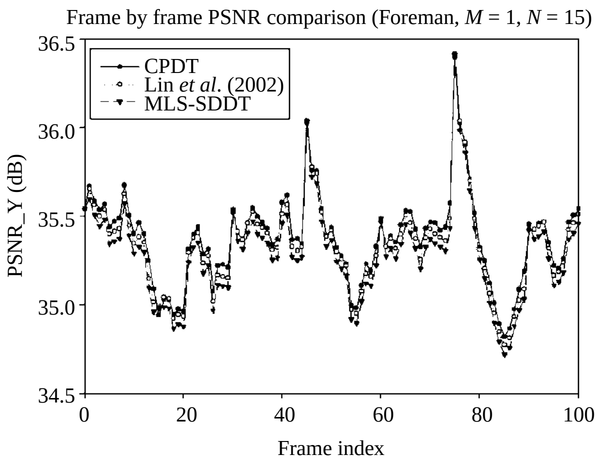

Fig. 16 Frame by frame comparison for FGS-to-MPEG-4 Simple Profile transcoding with GOP of M = 1, N = 15 on Foreman sequence

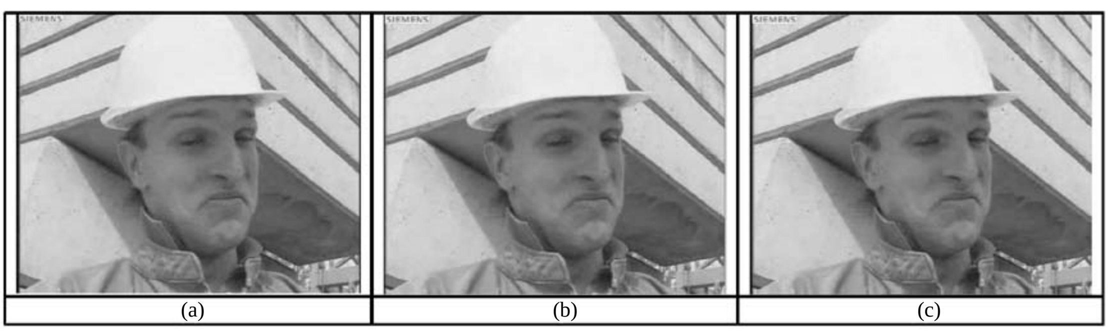

Fig. 17 Visual quality comparison for FGS-to-MPEG-4 Simple Profile transcoding at frame index of 73. (a) CPDT (b) Lin *et al*.'s work (2002) (c) MLS-SDDT

Figure 15 shows the R-D curves for the Foreman sequence with R_{BL} = 512 kbps. From this Figure, we can see minor R-D performance loss compared to the CPDT and Lin *et al*.'s work (2002). Fig. 16 shows the frame by frame PSNR comparison, and Fig. 17 shows the visual quality comparison. From Fig. 16 and Fig. 17, very similar visual quality between the MLS-SDDT and the other two architectures can be observed.

V. CONCLUSIONS

In this paper, we proposed a Multi Layer Streaming-Simplified DCT Domain Transcoder (MLS-SDDT) to improve transcoding complexity and video quality. The MLS-SDDT contains 2 major contributions: (1) the development of an efficient FGS to MPEG-1/2/4 single layer transcoder based on low complexity SDDT architecture, and (2) an R-D optimized multi-layer streaming technique to resolve serious error-drifting problems. By applying the MLS-SDDT to FGS-to-MPEG-1/2/4 single layer transcoding, our experiments show 1.4~7.0 dB PSNR improvement for MPEG-1 and 1.9~8.6 dB improvement for MPEG-2 compared to SDDT architecture. The visual quality is also significantly improved. For the FGS to MPEG-4 single layer transcoding, the MLS-SDDT can achieve minor PSNR loss (0.0 ~ 0.4 dB) with lower computational complexity as compared to the SDDT.

For future work, H.264/AVC has become one of the most popular video compression standards, and we are considering adding H.264/AVC to the the proposed architecture, for better multi-format transcoding. Due to many new complex coding techniques such as 6-tap motion compensation, and intra prediction in H.264/AVC, it is a challenge is to integrate H.264/AVC efficiently into single transcoding architecture with minor quality loss.

REFERENCES

Assuncao, P. A. A., and Ghanbari, M., 1998, "A Frequency-Domain Video Transcoder for Dynamic Bit-Rate Reduction of MPEG-2 Bit Streams," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 8, No. 8, pp. 953-967.

Bjork, N., and Christopoulos, C., 1998, "Transcoder Architecture for Video Coding," *IEEE Transactions on Consumer Electronics*, Vol. 44, No. 1, pp. 88-98.

Chang, S.-F., and Messerschmitt, D. G., 1995, "Manipulation and Compositing of MC-DCT Compressed Video," *IEEE Journal on Selected Areas in Communications*, Vol. 13, No. 1, pp. 1-11.

Keesman, G., Hellinghuizen, R., Hoeksema, F., and Heideman, G., 1996, "Transcoding of MPEG Bitstreams," *Signal Processing: Image Communication*, Vol. 8, No. 6, pp. 481-500.

Lin, Y.-C., Wang, C. N., Chiang, T., Vetro, A., and Sun, H., 2002, "Efficient FGS-to-Single Layer Transcoding," *Proceedings of IEEE International Conference on Consumer Electronics*, pp. 134-135.

Merhav, N., and Bhaskaran, V., 1997, "Fast Algorithms for DCT-Domain Image Downsampling and for Inverse Motion Compensation," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 7, No. 3, pp. 468-476.

Mohan, R., Smith, J. R., and Li, C.-S., 1999, "Adapting Multimedia Internet Content for Universal Access," *IEEE Transactions on Multimedia*, Vol. 1, No. 1, pp. 104-114.

Morrison, D. G., Nilson, M. E., and Ghanbari, M., 1994, "Reduction of the Bit-Rate of Compressed Video While in Its Coded Form," *Proceedings of 6th International Workshop Packet Video*, pp. D17.1-D17.4.

Shanableh, T., and Ghanbari, M., 2000, "Heterogeneous Video Transcoding to Lower Spatial-Temporal Resolutions and Different Encoding Formats," *IEEE Transactions on Multimedia*, Vol. 2, No. 2, pp. 101-110.

Shen, B., Ishwar, I. K., and Bhaskaran, V., 1999, "Adaptive Motion-Vector Resampling for Compressed Video Downscaling," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 9, No. 6, pp. 929-936.

Song, J., and Yeo, B.-L., 2000, "A Fast Algorithm for DCT-Domain Inverse Motion Compensation Based on Shared Information in A Macroblock," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 10, No. 5, pp. 767-775.

Sun, H., Kwok, W., and Zdepski, J. W., 1996, "Architectures for MPEG Compressed Bitstream Scaling," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 6, No. 2, pp. 191-199.

Xin, J., Lin, C.-W., Sun, M.-T., 2005, "Digital Video Coding," *Proceedings of IEEE*, Vol. 93, No. 1. pp. 84-97.

Yin, P., Vetro, A., Liu, B., and Sun, H., 2002, "Drift Compensation for Reduced Spatial Resolution Transcoding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 12, No. 11, pp. 1009-1020.

Youn, J., Sun, M.-T., and Lin, C.-W., 1999, "Motion Vector Refinement for High-Performance Transcoding," *IEEE Transactions on Multimedia*, Vol. 1, No. 1, pp. 30-40.

Manuscript Received: Mar. 06, 2007
Revision Received: May 17, 2007
and Accepted: Jun. 23, 2007