Video Conferencing Oriented Low-Complexity Coarse-Level Mode-Mapping Based H.264/AVC to H.264/SVC Spatial Transcoding

Lei Sun*, Jie Leng*, Jia Su*, Yiqing Huang†, Hiroomi Motohashi† and Takeshi Ikenaga*
* Waseda University, Kitakyusyu-shi, Japan
E-mail: sunlei@ruri.waseda.jp Tel: +81-080-39968453
† Ricoh R&D Center, Yokohama-shi, Japan
E-mail: isei.ko@nts.ricoh.co.jp Tel: +81-045-5933411

***Abstract*—As an extension of H.264/AVC, Scalable Video Coding (SVC) provides flexible adaptation to heterogeneous networks and end-users, which provides great scalability for applications such as video conferencing. However, due to the existence of legacy H.264/AVC-based systems, transcoding between AVC and SVC becomes necessary. Currently there are few works done on AVC-to-SVC spatial transcoding, and re-encoding method involves high computational cost. This paper proposes a low-complexity coarse-level mode-mapping based AVC-to-SVC spatial transcoder for video conferencing applications. First, to omit unnecessary motion estimations (ME) for layers with reduced resolution, an ME skipping scheme based on AVC mode distribution is proposed with an adaptive search range. After that an adaptive coarse-level mode-mapping method is presented for fast mode decision. Finally, motion vector (MV) refinement is introduced for further lower-layer time reduction. As for the top layer, direct encapsulation is proposed to preserve better quality and another scheme involving inter-layer predictions is also provided for bandwidth-crucial applications. Simulation results show that proposed transcoder achieves up to 90.6% time reduction without significant coding efficiency loss compared to re-encoding method.**

I. Introduction

The SVC extension of H.264/AVC standard provides the ability to adapt to diverse user-end capabilities and requirements and enables transmission of one bitstream consisting of multiple subset bitstreams [1]. The subset streams can be extracted adaptively according to user requirements and are organized in layered structure. 3 kinds of scalability are provided: spatial, temporal and quality scalability. It is expected to be a good solution for multipoint video conferencing which involve multiple terminals with different characteristics. Performance evaluations of SVC and its key technologies can be found in [2], [3], [4].

Though high coding efficiency is achieved benefitting from the inter-layer predictions [4], it is impossible for every system to support SVC codec. There are a lot of legacy systems or terminals which do not support SVC standard. These systems or terminals are potential participants in a future video conferencing application. To communicate with such kind of user ends, transcoding is necessary for SVC-based systems. Let's assume a multiparty video conferencing scenario, as shown in Fig.1. Part A is a new video conferencing system based on SVC standard, and part B is a legacy multipoint control unit (MCU) [5] based system which supports only H.264/AVC standard. Assume that a desktop PC in B sends a frame to a mobile in A, the receiver may be unable to decode or even receive because of limited processing ability and bandwidth. Thus, transcoding between AVC and SVC is needed. As a solution to provide backward compatibility, the gateway for system A should integrate the functionality of AVC/SVC transcoding.

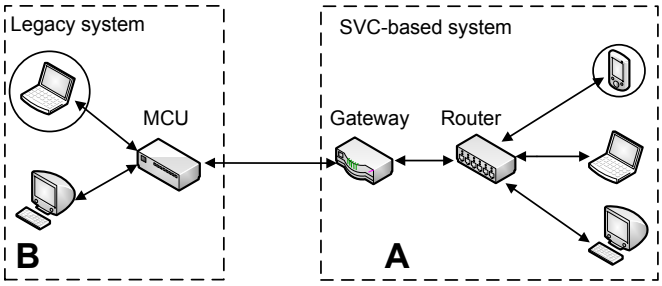


Fig. 1. A hybrid video conferencing scenario.

A straightforward solution in transcoding field is the cascaded re-encoding architecture [6], which fully decodes the input bitstream and then re-encodes. It usually requires high computational cost. In earlier works on transcoding, the majority of interest focused on 2 directions: homogeneous transcoding (same coding standard for input & output bitstreams) [6] and heterogeneous transcoding (different coding standards for input & output bitstreams) [7]. The AVC/SVC transcoding is more of a homogeneous transcoding due to SVC's AVC-compatible single layer encoding, though SVC has a layered structure. Most conventional works focus on single layer transcoding for bit-rate reduction [8], [9], spatial/temporal resolution reduction [10], [12], [13], CBR (constant bit-rate) and VBR (variable bit-rate) conversion, error-resilience transcoding and so on. Newly developed works include AVC/SVC temporal transcoding [15], quality transcoding [16], and SVC-to-AVC spatial transcoding [17]. AVC-to-SVC spatial transcoding

TABLE I
COMPUTATIONAL COMPLEXITY DISTRIBUTION (QP = 20).

Sequence	AVC decoding	Downsampling	SVC encoding
akiyo	2.29%	1.89%	95.82%
panzoom2	2.45%	1.63%	95.92%
vidyo1	2.16%	1.71%	96.13%
vidyo3	2.17%	1.63%	96.20%
bus	2.42%	1.16%	96.42%
football	2.38%	1.19%	96.43%
flower_garden	2.49%	1.13%	96.38%
cheer_leaders	2.26%	0.92%	96.82%

has not been fully investigated in existing literatures except for [18] which achieves about 2/3 time reduction. However, the PSNR (Peak Signal-to-Noise Ratio) often drops about 1 dB at the same bit-rate compared with re-encoding method, due to the introduced inter-layer prediction, non-optimal mode decision and proportional MV scaling.

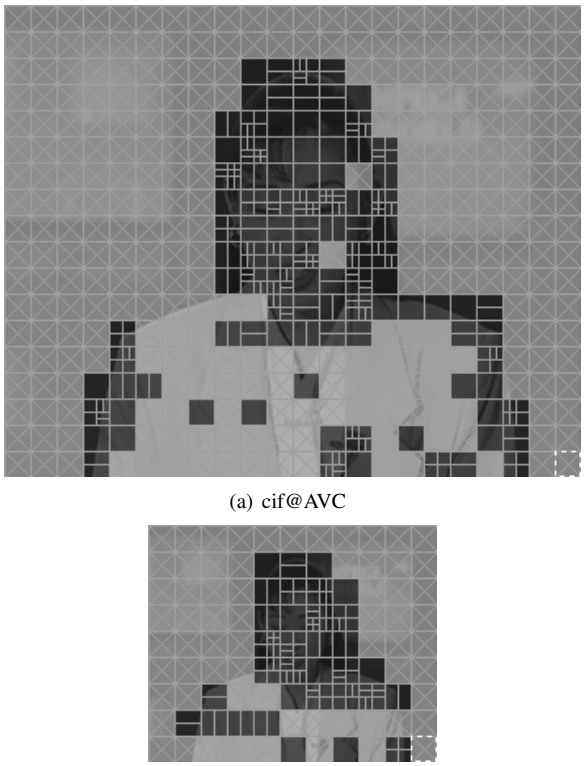
For reduced resolution transcoding, generally DCT(Discrete Cosine Transform)-domain approaches [12], [13], [14] achieve larger time reduction compared with pixel-domain approaches. However, drift problem occurs and should be compensated, which needs additional calculation effort and decreases the overall gain. The resultant PSNR drops a lot, averagely 0.3-0.4 dB for [12], 0.7-1.6 dB for [13] and 0.5-1.5 dB for [14]. Since in video conferencing systems, the quality is usually expected to remain as much as possible, these DCT-domain approaches are not suitable. References [10] and [19] are pixel-domain transcoding methods. In [10] the authors utilize 4-to-1 MV mapping with refinement which involves no sub-macroblocks (H.263), and the complexity reduction is claimed to be 23% averagely with approximately 0.7 dB PSNR loss. Reference [19] presents a mode-mapping based downscaling transcoding method and the PSNR drops about 1-4 dB due to the underlying proportional mapping.

This paper proposes a low-complexity coarse-level mode-mapping based transcoding architecture for video conferencing systems. For SVC lower-layer encoding, an ME skipping scheme based on the mode distribution of input AVC stream is adopted for saving unnecessary ME calculations and the associated search range is determined through an adaptive way. For non-skippable MBs, a following adaptive coarse-level mode-mapping method is applied for fast INTER mode decision. Finally, MV refinement is introduced for some special cases to further reduce the time. For SVC top-layer encoding, 2 schemes with different focus on quality or bit-rate are discussed.

This paper is organized as follows. Section II gives analysis of the reference re-encoding model. Section III & IV explain the proposed methods in detail for lower layers and top layer respectively, followed by overall architecture description in Section V. Experimental results are given in Section VI, and conclusions are made in Section VII.

II. REFERENCE RE-ENCODING METHOD

The cascaded pixel-domain re-encoding architecture [18] is selected as the start point of proposed transcoder and serves as a reference model. There are 3 procedures involved in AVC to SVC spatial transcoding - AVC decoding, downsampling and SVC encoding. Table I shows the time cost in re-encoding model for 8 test sequences which will be specified in Section VI. The most time-consuming part is the SVC encoding part, which involves motion estimations. AVC decoding and downsampling procedures are trivial in computational cost compared with SVC encoding. Therefore, complexity reduction in SVC encoding is necessary on the road towards low-delay transcoding. Reduction in top layer is simple since R-D (Rate-Distortion) optimized information from AVC bitstream is available. The following paragraph discusses the possible solutions for time reduction in reduced-resolution transcoding.



(a) cif@AVC

(b) qcif@SVC

Fig. 2. Intuitive mode & partition comparison for akiyo sequence, frame 37. (light grey: SKIP, dark grey: INTER (non-SKIP))

Though more general and statistical analysis is preferred, we only give a representative example to show the trend of motion data correlation between original frame and reduced-resolution frame. Fig.2 shows an intuitive mode/partition comparison for AVC frame and SVC downsized frame at a randomly selected frame in dyadic transcoding. Here the INTER refers to non-SKIP inter modes, and it holds for the rest of this paper. It can be inferred from the figure that AVC-coded frame and corresponding downsized SVC-coded frame have similar mode distribution. Therefore, mode information reuse is possible. Based on this observation, the mode skipping schemes in Sec.III.A is proposed. Besides, although the mode distributions are similar, the MB partitions are usually loosely coupled as

shown in the figure. Therefore, conventional methods based on proportional partition mapping is not suitable. To address this problem, the mode mapping and MV refinement schemes are proposed in Sec.III.B and Sec.III.C.

III. Proposed lower-layer transcoding schemes

A. ME skipping scheme

Though downsampling methods cannot completely construct the relation between high-resolution and low-resolution frames, we found that there is a rough rule between them, that is the similar mode distribution. If the lower-layer MB is with mode X (INTER, INTRA or SKIP - no concern about sub-partitions), then the top-layer co-located region (consisting of s^2 MBs and s is the scaling factor) is probably also with mode X. This is usually the case except for some irregular MBs caused by downsampling losses. An ME skipping scheme is proposed based on this rule.

Firstly, an adaptive search range in high-resolution frame is defined, and the lower bound for the search range is $s \times s$ (co-located MBs). And assume that the top-layer resolution is $M \times N$, then the upper bound is set to $U \times U$ according to (1) & (2).

$$SR\_UB\_Width = round\left(\sqrt{\frac{M}{16} \times \frac{N}{16} \times r}\right) \quad (1)$$

$$U = s \times round\left(\frac{SR\_UB\_Width}{s}\right) \quad (2)$$

Here the round(.) operator calculates the nearest integer for the parameter, and r is the percentage of MBs in search range over entire frame. Equation (1) calculates the search range width for upper bound in terms of MBs, and (2) maps the value to multiples of scaling factor in order to make the search range symmetrical to co-located region. Too large r decreases overall time reduction and too small value will lead to non-statistical result. Through vast experiments over different sequences, we found that generally 0.04 gave good performance. In dyadic transcoding (s = 2), the upper bound is 4×4 for CIF size, 6×6 for VGA size and 12×12 for 720p size when r is set to 0.04. Fig.3 shows an example for VGA sequence. The grey region shows the co-located MBs and the numbers mean the search order (from 1 to 36 with increasing distance to center and decreasing relevance to lower-layer MB).

In proposed transcoder, the search range is adapted MB by MB according to the homogeneity of previous MB between lower bound and upper bound. If the search range of previous MB contains only one type of mode, it is considered smoothed and the current MB will decrease the search range by [-2, -2] (e.g. 6×6 -> 4×4). Otherwise, it is considered detailed and the search range will be enlarged by [+2, +2]. Of course, the search range will not across the boundaries.

Then check the modes of these MBs in the search range in predefined order. If SKIP, INTER or INTRA exists, estimate this mode respectively. On the contrary, if some mode does not exist in the search range, then skip the estimation for this mode. This scheme works efficiently when some mode is concentrated in limited areas, which means the other modes may be skipped by proposed scheme.

26	25	24	23	22	21
27	10	9	8	7	20
28	11	1	2	6	19
29	12	3	4	5	18
30	13	14	15	16	17
31	32	33	34	35	36

Fig. 3. Search range and search order for VGA sequence.

B. Coarse-level mode-mapping methods

As mentioned in Sec.II, although the mode distributions are similar between high-resolution and low-resolution frames, this is not the case for partitions and MVs of INTER MBs. In fact they have rarely proportional relations. Therefore, lower layer motion data should not be mapped by scaling partition and MV directly [19]. Instead, we find another coarse-level rule that if lower-layer MB has few details, AVC co-located MBs usually also have few details. On the contrary, if lower-layer MB has many details, AVC co-located MBs probably also have many details (but not have to be proportional). Based on this rule, 3 mode mapping sub-schemes are explained in the following paragraphs for INTER estimation, with different tradeoff between coding efficiency and complexity. They are combined adaptively in proposed transcoder in order to achieve an optimal result.

The first method is the direct-mapping method, which is a 4-to-1 mapping as shown in Fig.4. This method first checks the co-located MBs to see if 8×8 mode (no concern about sub-partitions) exists. If it exists, stop the procedure and estimate 8×8 (incl. sub-partitions) only. Otherwise, continue to check 8×16, 16×8 and 16×16 similarly. If no mode is selected at last, all INTER modes will be estimated.

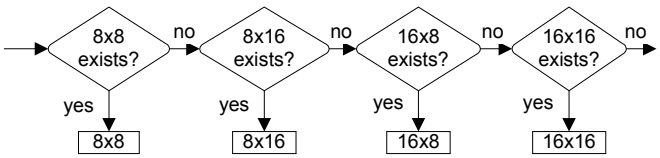


Fig. 4. Direct mapping method.

The candidate-mapping approach is the second method which performs ME for candidate modes only. This method selects co-located MBs' modes as candidates for lower-layer encoding, as illustrated in Fig.5. It checks 8×8, 8×16, 16×8 and 16×16 sequentially in co-located MBs. If some mode exists, it will be added to estimation list. Otherwise, it will not be added. When the procedure finishes, only the modes

TABLE II
PERFORMANCE COMPARISON OF PROPOSED COARSE-LEVEL MODE-MAPPING METHODS (LOWER-LAYER).

Sequence	direct			candidate			priority			adaptive		
	C1	C2	C3	C1	C2	C3	C1	C2	C3	C1	C2	C3
akiyo	+6.35	-0.32	-69.3	+5.02	-0.25	-68.2	+2.16	-0.11	-63.3	+3.31	-0.17	-68.0
panzoom2	+8.43	-0.42	-64.4	+6.61	-0.31	-63.5	+3.04	-0.15	-60.5	+3.37	-0.19	-63.7
vidyo1	+11.10	-0.44	-71.6	+6.93	-0.27	-69.2	+4.16	-0.14	-68.2	+5.01	-0.21	-68.7
vidyo3	+8.03	-0.38	-72.2	+4.27	-0.21	-70.1	+2.70	-0.13	-68.3	+3.04	-0.17	-71.0
bus	+6.34	-0.41	-39.5	+3.12	-0.18	-33.1	+2.25	-0.13	-28.3	+2.41	-0.16	-33.7
football	+5.52	-0.32	-34.7	+4.43	-0.22	-31.4	+3.49	-0.20	-23.1	+3.57	-0.22	-30.3
flower_garden	+5.61	-0.34	-50.3	+3.96	-0.20	-44.7	+3.35	-0.21	-43.2	+3.45	-0.22	-47.5
cheer_leaders	+4.33	-0.35	-32.2	+3.22	-0.22	-27.6	+2.44	-0.17	-20.2	+2.55	-0.18	-24.6

Criterions: C1: BDBR(%), C2: BDPSNR(dB), C3: $\Delta time(\%)$

in estimation list will be estimated. If no mode exists in the estimation list at last, all INTER modes will be estimated.

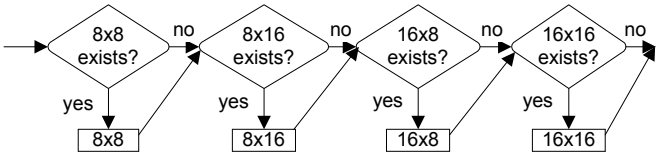


Fig. 5. Candidate mapping method.

The third method is the priority-mapping method, which performs ME based on priority. The priory is defined as: 8×8 $>$ {8×16, 16×8} $>$ 16×16. This is because the complexity and uncertainty of detailed MB is larger than smooth one. This method checks from 8×8 to 16×16 as shown in Fig.6. If some mode exists, estimate all modes with larger (or equal) priority. Similarly, if no INTER mode exists at last, all modes will be estimated.

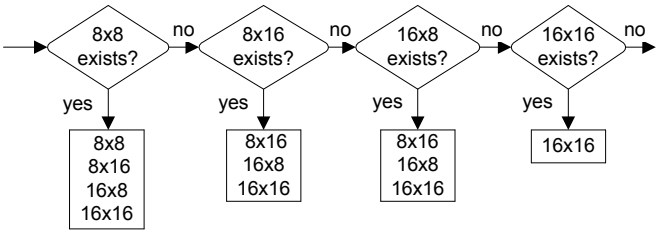


Fig. 6. Priority mapping method.

These methods involve no sub-partitions due to the irregularity of sub-partitions and reuse the AVC stream information at a coarser level. Table II shows that the direct mapping method has the largest complexity reduction while priority mapping method achieves the best coding efficiency, and candidate mapping method performs moderately.

In proposed transcoder, they are combined adaptively according to the homogeneity of current search range. Three levels are defined: level 1 with less than 1/3 SKIP or INTER_16×16 modes in current search range, level 2 with less than 2/3 but more than 1/3 SKIP or INTER_16×16 modes, level 3 with more than 2/3 SKIP or INTER_16×16 modes. Level 1 is the most detailed, so the most accurate priority-mapping method is adopted. Level 2 is moderately detailed and candidate-mapping method is selected. Direct-mapping method is used in level 3 which is the least detailed. Table II shows the lower layer gains for the 3 methods as well as the adaptive usage, compared with re-encoding model. It can be seen that adaptive method achieves almost the same time reduction as candidate method, while obtaining comparable coding efficiency to priority method. It should be noticed that proposed adaptive method achieves 67.9% time reduction averagely for the top 4 sequences, which are video-conferencing similar scenes. For the following 4 sequences, the time reduction is only 34.0% averagely.

C. MV refinement scheme

Most conventional works on MV refinement were based on nearly whole-frame MV mapping [11], [20], which turned out to be inaccurate and caused efficiency loss. MV refinement is expected to be more efficient in homogeneous area compared with detailed area since less MVs are involved. Detailed area which leads to more MVs will increase the complexity and uncertainty for MV mapping. In proposed transcoder, MV refinement is only applied for MBs which satisfy 2 conditions. First, the co-located MBs are all SKIP or INTER_16×16. Second, another check is executed - the MV diversity of co-located MBs. Equation (3) calculates the arithmetic average MV among co-located MBs. s is the scaling factor and MV_i_x & MV_i_y represent the horizontal and vertical components for i_{th} MB respectively. Equation (4) calculates the diversities of horizontal and vertical MV components by summing the absolute difference (SAD) between the MVs of co-located MBs and the average MV.

$$\begin{cases} \overline{MV\_x} = \frac{1}{s^2} \sum_{i=0}^{s^2-1} MV_i\_x \\ \overline{MV\_y} = \frac{1}{s^2} \sum_{i=0}^{s^2-1} MV_i\_y \end{cases} \tag{3}$$

$$\begin{cases} SAD_x = \sum_{i=0}^{s^2-1} |MV_i\_x - \overline{MV\_x}| \\ SAD_y = \sum_{i=0}^{s^2-1} |MV_i\_y - \overline{MV\_y}| \end{cases} \tag{4}$$

Then the SAD values are compared with pre-defined thresholds, as shown in (5). If (5) holds, the MV refinement is applicable. Otherwise, it will not be performed. Smaller thresholds will constrict the applicable rate while larger thresholds will result in worse coding efficiency. In our experiments,

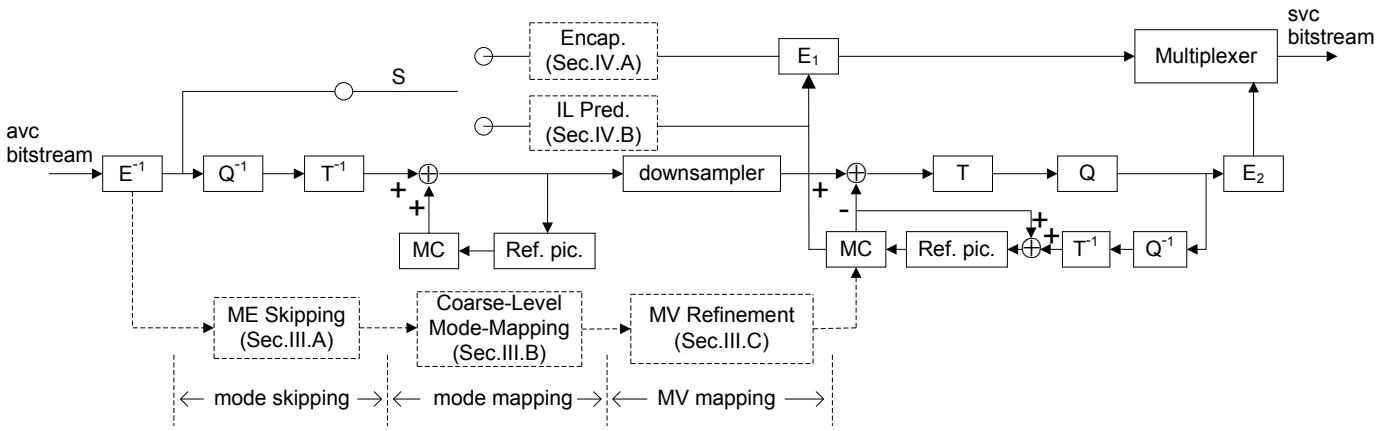


Fig. 7. Proposed transcoder. (Encap.: Encapsulation, IL Pred.: Inter-Layer Prediction)

the thresholds Th_x and Th_y are both set to 4 since small thresholds give no harm anyhow.

$$\begin{cases} SAD_x \leq Th_x \\ SAD_y \leq Th_y \end{cases} \tag{5}$$

$$\begin{cases} MV\_scaled\_x = \frac{1}{s}\overline{MV\_x} \\ MV\_scaled\_y = \frac{1}{s}\overline{MV\_y} \end{cases} \tag{6}$$

If the MV refinement is feasible, INTER_16×16 is chosen as the lower-layer MB mode. The scaled average MV calculated by (6) will be used for lower-layer ME, and it is used as the starting point for motion search with a much smaller search range compared to original search window. In our experiments, the refinement search range is selected as [-2, +2] for both horizontal and vertical components.

IV. Proposed top-layer schemes

A. Direct encapsulation

A first approach for top-layer transcoding is to extract the AVC stream information directly along with necessary SVC header encapsulation. It is simple and preserves the quality. Reference [4] points out that the SVC coding tools are less efficient for spatial scalability especially for simple and slow-motion scenes, which is often the case in video conferencing applications. Therefore, direct encapsulation is recommended for video conferencing if the bandwidth is sufficient.

B. Inter-layer prediction utilization

The inter-layer predictions can be utilized when the bandwidth is crucial, as another choice. In direct encapsulation method, R-D costs for different modes which the inter-layer predictions need can not be obtained since there is no ME performed in top layer. We only re-calculate the R-D cost according to the mode and MV got from the AVC bitstream, which have been R-D optimized already. Since the source sequence can not be obtained on the transcoder side, we use the decoded sequence instead as the input for R-D cost calculation in SVC encoder, just like the re-encoding method. The inter-layer motion and intra predictions act the same as original SVC encoder, and residual prediction is modified to be performed only for the corresponding mode and MV got from AVC frame. As a result the processing speed decreases a little, but the overall complexity is still kept very low since ME is not performed. The drawback for this scheme is the degraded quality due to a second-time encoding loss. It is recommended for bandwidth-crucial applications.

V. Overall transcoding architecture

Fig.7 shows the overall architecture for proposed transcoder. For simplicity and clarity we only show the 2-layer structure. Both motion compensation and downsampling are processed in pixel domain. The marks E, Q, T, E_1, E_2 stand for entropy coding, quantization, transformation, top-layer entropy coding and lower-layer entropy coding respectively.

In the lower-layer ME, three schemes are proposed - one mode skipping schemes, one mode mapping scheme(incl. 3 sub-schemes) and one MV mapping scheme. First, the ME skipping scheme is applied with the intention to skip unlikely mode types which is described in Sec.III.A. Then, a coarse-level mode-mapping scheme is applied for INTER MBs which are not skipped by previous step and Sec.III.B explains the details. At last, MV refinement is applied for further time reduction as explained in Sec.III.C.

The switch S in the figure changes the top-layer strategy according to the network condition. If the bandwidth is enough, the upper routine described in Sec.IV.A with well-preserved top-layer quality will be selected. Otherwise, the lower routine described in Sec.IV.B will be adopted for lower bit-rate with degraded top-layer quality.

VI. Experimental results

Proposed methods are applied to some sequences in this section and the results are shown here. All experiments are performed on an Intel Core 2 (2.67GHz) computer with 2.0GB RAM and software implementation is based on JM (Joint Model) 17.2 and JSVM (Joint Scalable Video Model) 9.18. JSVM's AVC compatible decoder and down-converter are used

TABLE III
OVERALL PERFORMANCE COMPARISON OF PROPOSED TRANSCODER(TOP-LAYER PSNR, TOTAL BIT-RATE & TIME).

Sequence	Direct Encapsulation			Inter-Layer Prediction		
	Δ bit-rate (%)	Δ Y-PSNR (dB)	Δ time(%)	Δ bit-rate (%)	Δ Y-PSNR (dB)	Δ time(%)
akiyo	+6.68	+1.508	-89.4	+1.57	-0.077	-81.7
panzoom2	+9.94	+1.482	-89.1	+1.45	-0.059	-77.6
vidyo1	+8.21	+1.361	-91.2	+1.44	-0.151	-82.5
vidyo3	+4.97	+1.420	-90.6	+1.82	-0.131	-83.3
bus	+6.89	+1.572	-88.7	+3.37	-0.082	-71.6
football	+4.03	+1.551	-89.2	+1.29	-0.113	-73.2
flower_garden	+7.14	+1.643	-89.0	+1.63	-0.082	-71.4
cheer_leaders	+4.02	+1.457	-88.1	+0.85	-0.171	-72.0

for AVC decoding and downsampling processes respectively. 8 sequences are examined with 2-layer dyadic spatial scalability. Akiyo, panzoom2, football and bus are CIF to QCIF transcoding at 30 fps; flower_garden and cheer_leaders are VGA to QVGA transcoding at 30 fps; vidyo1 and vidyo3 are 720p to 360p (640×360) transcoding at 60 fps. Akiyo, panzoom2, vidyo1 and vidyo3 are sequences similar to video-conferencing scenes with still background, slow motions or camera motions, while the other 4 are complex and detailed ones. For each sequence 150 frames are tested with the GOP structure of IPPP. The search range is 16 for CIF-to-QCIF transcoding and 32 for the rest. In experiments the QPs (Quantization Parameter) for AVC encoder and transcoder are set to same values, and QPs are selected as 20, 24, 28 and 32. Other parameters are carefully selected to insure the comparability between proposed transcoder and the reference model.

In order to fairly compare the top layer quality, the Y-PSNR between original sequence (user side, encoded by AVC and sent to transcoder) and the reconstructed sequence after transcoding are calculated, since it's meaningless to calculate the Y-PSNR between decoded sequence (transcoder side, decoded and used as SVC encoder input) and reconstructed sequence which is identical to the original one in case of direct encapsulation. Besides, Bjøntegaard Delta only reflects the average gain at same bit-rate or same PSNR, which does not show the performance at same QP. In order to show the advantage of inter-layer prediction (ILP) approach over direct encapsulation (lower bit-rate at same QP), the R-D curves are shown in Fig.9 and the average deltas are calculated for 4 QPs, as shown in Table 5. Table 5 shows the overall performance of proposed transcoder compared with re-encoding model and the 2 top-layer schemes are both examined. We can see that proposed transcoder achieves up to 91.2% time reduction compared with re-encoding method.

The direct encapsulation method gains averagely 89.9% time reduction for tested sequences with 6.49% bit-rate increase and 1.50 dB quality increment. The time reduction for top 4 sequences is 1.3% larger than the lower 4 sequences. The merit for this method is the significant time reduction and the well-preserved top-layer quality since there is no second-time encoding.

By contrast, the ILP approach keeps the bit-rate low while still obtaining 76.7% time reduction averagely. It decreases the bit-rate increment to 1.68%. The time reduction for lower 4 sequences decreases by 9.2% compared with top 4 sequences. It is suitable for applications with limited network bandwidth. The main drawback is the degraded top-layer quality due to re-quantization loss which is a little worse than re-encoding method.

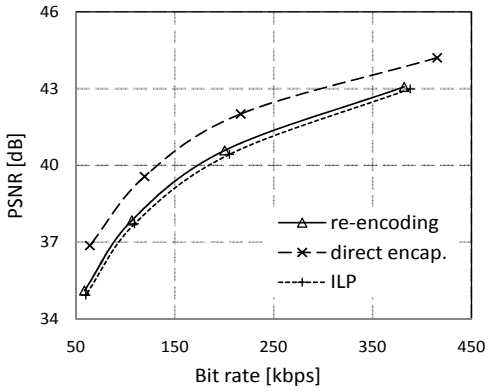
Fig.9 shows the R-D curves of top layer for re-encoding method and proposed transcoder with 2 different top-layer methods. It is shown that the direct encapsulation method achieves best coding efficiency while ILP method is slightly worse than re-encoding method.

VII. CONCLUSIONS

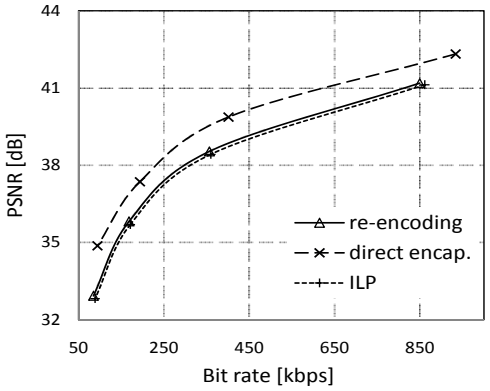
This paper proposes a low-complexity AVC to SVC spatial transcoder based on coarse-level mode mapping for video conferencing applications. For lower layer (with reduced picture size) transcoding, a mode skipping methods is applied first and then a coarse level mode-mapping method is applied which adaptively selects different sub-schemes, followed by an MV refinement scheme for further time reduction. As for the top layer, 2 schemes are presented corresponding to the network condition. Sec.IV.A depicts the direct encapsulation method which is suitable when the bandwidth is sufficient, and Sec.IV.B shows another approach which utilizes the inter-layer predictions of SVC for reducing the bit-rate. Simulation results show that direct encapsulation method achieves significant time reduction with much higher coding efficiency than re-encoding method, since no second-time quantization is involved. The ILP method achieves lower bit-rate than direction encapsulation when the QP is the same, while the time reduction reduces by 13.2% averagely. The coding performance of ILP method is slightly worse than re-encoding method.

VIII. ACKNOWLEDGMENTS

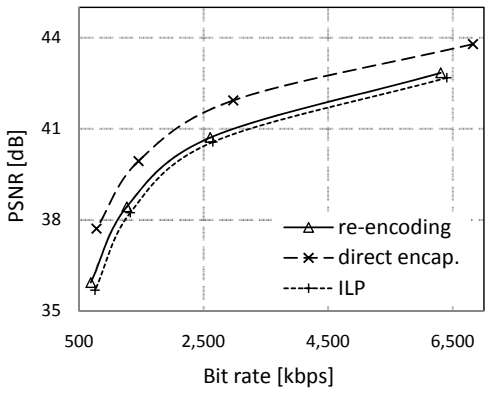
This work was supported by Waseda University Global COE Program "International Research and Education Center for Ambient SoC" sponsored by MEXT, Japan. This work was also supported by KAKENHI (23300018), Waseda University, Japan.



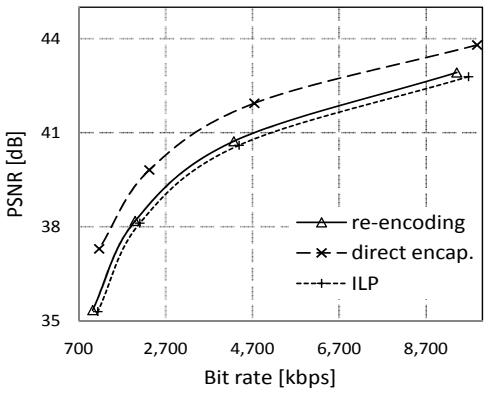
(a) akiyo



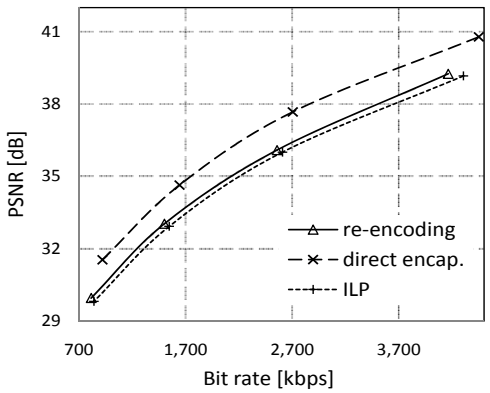
(b) panzoom2



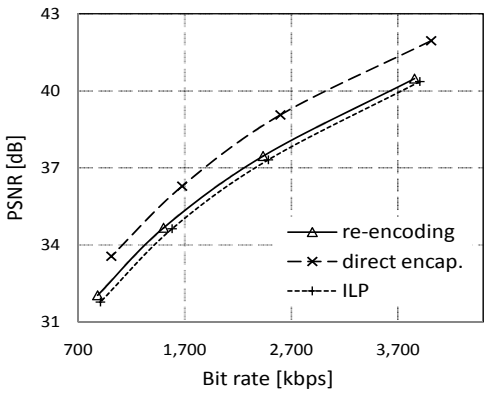
(c) vidyo1



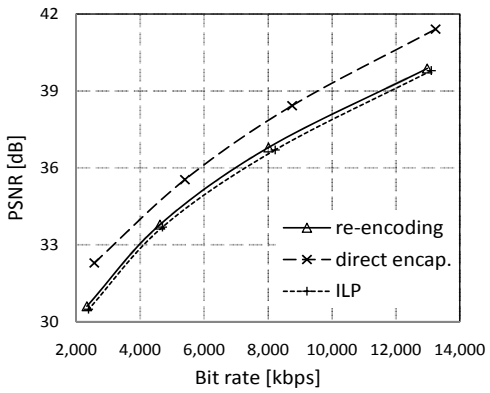
(d) vidyo3



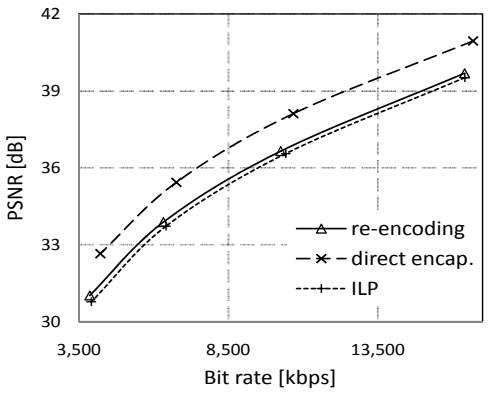
(e) bus



(f) football



(g) flower_garden



(h) cheer_leaders

Fig. 8. R-D curves comparison for top layer.

REFERENCES

[1] H. Schwarz, and D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard", IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 9, pp. 1103-1120, 2007.

[2] H. Choi, K. Lee, S.J. Bae, J.W. Kang and J.J. Yoo, "Performance Evaluation of the Emerging Scalable Video Coding", IEEE International Conference on Consumer Electronics (ICCE), , pp. 1-2, Las Vegas, NV, 2008.

[3] T. Oelbaum, H. Schwarz, M. Wien and T. Wiegand, "Subjective Performance Evaluation of the SVC Extension of H.264/AVC", 15th IEEE International Conference on Image Processing (ICIP), pp. 2772-2775, San Diego, CA, 2008.

[4] X. Li, P. Amon, A. Hutter, and A. Kaup, "Performance Analysis of Inter-Layer Prediction in Scalable Video Coding Extension of H.264/AVC", IEEE Transactions on Broadcasting, vol. 57, no. 1, pp. 66-74, 2011.

[5] M.H. Willebeek-LaMair, D.D. Kandlur, and Z.Y. Shae, "On Multipoint Control Units for Videoconferencing", 19th Conference on Local Computer Networks (LCN), pp. 356-364, Minneapolis, MN, 1994.

[6] A. Vetro, C. Christopoulos, and H. Sun, "Video Transcoding Architectures and Techniques: An Overview", IEEE Signal Processing Magazine, vol. 20, no. 2, pp. 18-29, 2003.

[7] T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding to Lower Spatio-Temporal Resolutions and Different Encoding Formats", IEEE Transactions on Multimedia, vol. 2, no. 2, pp. 101-110, 2000.

[8] H. Sun, W. Kwok, and J.W. Zdepski, "Architectures for MPEG Compressed Bitstream Scaling", IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, no. 2, pp. 191-199, 1996.

[9] P.A.A. Assuncao and M. Ghanbari, "Post-processing of MPEG2 Coded Video for Transmission at Lower Bit Rates", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 4, pp. 1998-2001, Atlanta, GA, 1996.

[10] N. Bjork and C. Christopoulos, "Transcoder Architectures for Video Coding", IEEE Transactions on Consumer Electronics, vol. 44, no. 1, pp. 88-98, 1998.

[11] B. Shen, I.K. Sethi, and B. Vasudev, "Adaptive Motion-Vector Resampling for Compressed Video Downscaling", IEEE Transactions on Circuits and Systems for Video Technology, vol. 9, no. 6, pp. 929-936, 1999.

[12] W. Zhu, K.H. Yang, and M.J. Beacken, "CIF-to-QCIF Video Bitstream Down-Conversion in the DCT Domain", Bell Labs Technical Journal, vol. 3, no. 3, pp. 21-29, 1998.

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

[14] J. De Cock, S. Notebaert, K. Vermeirsch, P. Lambert, and R. Van de Walle, "Efficient Spatial Resolution Reduction Transcoding for H.264/AVC", IEEE International Conference on Image Processing (ICIP), pp. 1208-1211, San Diego, CA, 2008.

[15] R. Garrido-Cantos, J. De Cock, J. Luis Martinez, S. Van Leuven, and P. Cuenca, "Motion-based Temporal Transcoding from H.264/AVC-to-SVC in Baseline Profile", IEEE Transactions on Consumer Electronics, vol. 57, no. 1, pp. 239-246, 2011.

[16] J. De Cock, S. Notebaert, P. Lambert, and R. Van de Walle, "Architectures for Fast Transcoding of H.264/AVC to Quality-Scalable SVC Streams", IEEE Transactions on Multimedia, vol. 11, no. 7, pp. 1209-1224, 2009.

[17] H. Liu, Y. Wang, Y. Chen, and H. Li, "Spatial transcoding from Scalable Video Coding to H.264/AVC", IEEE International Conference on Multimedia and Expo (ICME), pp. 29-32, New York, NY, 2009.

[18] R. Sachdeva, S. Johar, and E. Piccinelli, "Adding SVC Spatial Scalability to Existing H.264/AVC Video", IEEE/ACIS International Conference on Computer and Information Science (ICIS), pp. 1090-1095, Shanghai, 2009.

[19] P. Zhang, Y. Liu, Q. Huang, and W. Gao, "Mode Mapping Method for H.264/AVC Spatial Downscaling Transcoding", IEEE International Conference on Image Processing (ICIP), vol. 4, pp. 2781-2784, Singapore, 2004.

[20] J. Youn, M. Sun, and C. Lin "Motion Vector Refinement for High-Performance Transcoding", IEEE Transactions on Multimedia, vol. 1, no. 1, pp. 30-40, 1999.