

Research Article

Optimal Multilayer Adaptation of SVC Video over Heterogeneous Environments

Truong Cong Thang,^{1,2} Jung Won Kang,² Jeong-Ju Yoo,² and Yong Man Ro¹

¹ Multimedia Group, Information and Communications University, Daejeon 305-732, South Korea

² Broadcasting Media Research Group, Electronics and Telecommunications Research Institute, Daejeon 305-700, South Korea

Correspondence should be addressed to Truong Cong Thang, tcthang@etri.re.kr

Received 21 August 2007; Accepted 22 November 2007

Recommended by Jianwei Huang

Scalable video coding (SVC) is a new video coding format which provides scalability in three-dimensional (spatio-temporal-SNR) space. In this paper, we focus on the adaptation in SNR dimension. Usually, an SVC bitstream may contain multiple spatial layers, and each spatial layer may be enhanced by several FGS layers. To meet a bitrate constraint, the fine-grained scalability (FGS) data of different spatial layers can be truncated in various manners. However, the contributions of FGS layers to the overall/collective video quality are different. In this work, we propose an optimized framework to control the SNR scalability across multiple spatial layers. Our proposed framework has the flexibility in allocating the resource (i.e., bitrate) among spatial layers, where the overall quality is defined as a function of all spatial layers' qualities and can be modified on the fly.

Copyright © 2008 Truong Cong Thang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

In the context of Universal Multimedia Access (UMA), multimedia contents should be adapted to meet various constraints of heterogeneous environments [1]. Among existing media types, video content imposes many challenges to the development of a transparent delivery chain [2]. Currently, there are two main technologies for video adaptation, namely, transcoding and scalable coding. Due to the high complexity of transcoding, many efforts have been focused on the development of scalable coding [3, 4].

Scalable video coding (SVC) [5] is a promising video format for applications of multimedia communication. SVC format, which is extended from the latest advanced video coding (AVC) [6], is appropriate to create a wide variety of bitrates with high-compression efficiency. An original SVC bitstream can be easily truncated in different manners to meet various characteristics and variations of devices and connections. The scalability is possible in 3 dimensions: spatial, temporal, and SNR. The spatial scalability of SVC intelligently combines multiple spatial layers into a single bitstream, which has much better coding efficiency than simulating multiple streams of different spatial sizes. The tempo-

ral scalability is supported by hierarchical B pictures which enable both the ease of truncation and high-coding efficiency. Besides, fine-grained scalability (FGS) data of SNR scalability can be truncated arbitrarily to meet the bitrate constraint of connection. Usually, FGS data is truncated in a top-down manner [7], that is, starting from the highest spatial layer to the lowest spatial layer.

Though scalable coding formats in general and SVC in particular provide flexibility in truncating the coded bitstream, there is a strong demand for the optimal adaptation strategies and solutions in various contexts [8]. In recent years, much research has been focused on the adaptation of MPEG-4 FGS video (e.g., [9, 10]), where the bitstream contains only one spatial layer. In our previous works [11, 12], we have developed an MPEG-21-enabled adaptation system, where the SVC bitstream is adapted in the full spatio-temporal-SNR space. However, the goal is still to optimize the quality of only one resolution.

In this work, we focus on FGS data truncation of multispatial layer (or multilayer for short) SVC bitstream, so as to maximize the overall/collective quality of the spatial layers provided by the adapted bitstream. For example, let us consider the following scenario (Figure 1). Suppose that

a surveillance video is encoded by SVC format with two spatial layers, each of which is enhanced by FGS data. That video is streamed to a remote building where two users will consume the content. The first user has a PC which will decode the highest spatial layer and the second user has a PDA which decodes the lowest spatial layer. To meet the connection bitrate of that building, the FGS data will be truncated. Note that the FGS data may account for a significant portion (e.g., two thirds) of the total bitrate.

Currently, the FGS data of the above bitstream can be truncated with a few approaches. With the conventional approach of top-down truncation [7], the lowest spatial layer always gets the best possible quality while the highest spatial layer may be much degraded. On the contrary, with the approach of [13], some FGS data in the lower spatial layer can be removed so as the highest spatial layer always has the best possible quality. We call this approach as highest-max, implying the maximization of the highest spatial layer's quality. It should be noted that the highest-max truncation is not "bottom-up" truncation, in which truncation simply starts from the lowest spatial layer to the highest spatial layer. As discussed later, the bottom-up truncation is actually not useful.

Additionally, in practice the requirements from users may be complex and variant in time. For example, the above two users request a "weighted balance" of qualities between them (or between the two spatial layers); or when a key (primary) user moves between end-devices, the quality should be reallocated accordingly. We consider this fact as a kind of user collaboration [14], which should be exploited to improve the overall/collective quality across multiple users.

In this paper, we propose a general framework to adapt SVC bitstream having multiple spatial layers. Our proposed framework has the flexibility in allocating the resource (i.e., bitrate) among spatial layers, where the overall quality is defined as a function of all spatial layers' qualities and can be modified on the fly. The adaptation process is first formulated as a constrained optimization problem. Then we propose a solution based on the Viterbi algorithm to find the optimal bitrate allocation between spatial layers. We will also show that the approaches of [7, 13] are just two extreme cases of our general framework.

This paper is organized as follows. In Section 2, we present the problem formulation. The solution to this problem, which is based on Viterbi algorithm, is proposed in Section 3. Section 4 presents the experiments to show the effectiveness and performance of our framework. Finally, conclusion is provided in Section 5.

2. PROBLEM FORMULATION

The FGS truncation process in SVC can be conceptually illustrated in Figure 2. Suppose that we have an SVC bitstream which consists of 2 spatial layers. Each spatial layer is composed of a base quality layer and FGS data which progressively enhance the SNR quality of that spatial layer. FGS data of a lower spatial layer can be used for interlayer prediction of a higher spatial layer. However, the FGS data can be truncated arbitrarily, regardless of the location. Anyway, the FGS

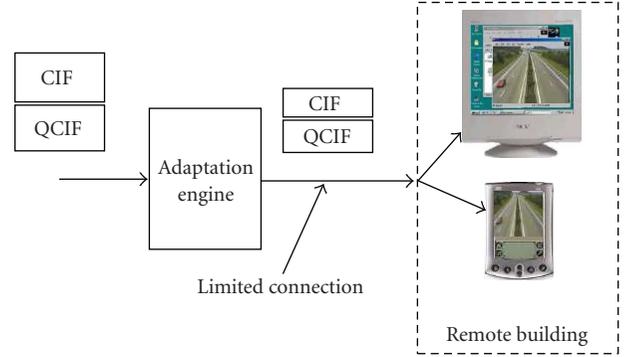


FIGURE 1: A scenario of two users with one SVC bitstream.

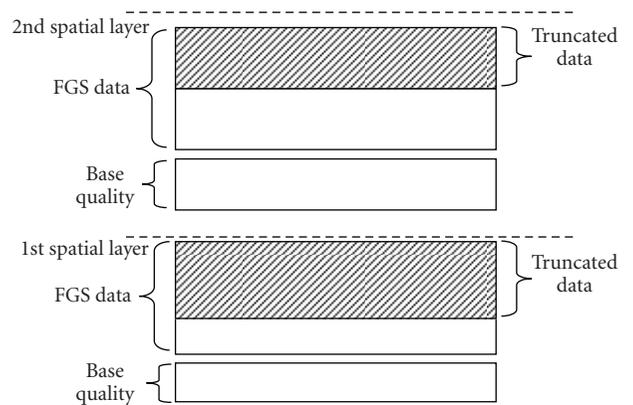


FIGURE 2: FGS data truncation of an SVC bitstream with multiple spatial layers.

data of a given spatial layer should be truncated "top-down", that is, from the highest quality to the base quality.

Note that, the base quality layer represents the minimum quality of a spatial layer. Nonetheless, in practice, users could request quality thresholds of their own, which may be higher than those of base quality layers.

Denote OQ as the "overall quality" (or collective quality) of the truncated bitstream, N the number of spatial layers, R_i and Q_i the "FGS bitrate" and corresponding quality of spatial layer i , and Q_i^{\min} the requested minimum quality of spatial layer i . Also let R^c denote the bitrate constraint of all FGS data, which is the difference of the overall bitrate constraint and the base quality bitrate. The adaptation framework can be formulated as follows:

maximize OQ subject to

$$\sum_{i=1}^N R_i \leq R^c, \quad Q_i \geq Q_i^{\min} \quad \text{with } i = 1, \dots, N. \quad (1)$$

OQ is generally defined as a function of spatial layers' qualities:

$$OQ = f(Q_1, Q_2, \dots, Q_N). \quad (2)$$

Currently, we compute the overall quality using the weighted sum as follows:

$$OQ = \sum_{i=1}^N w_i \cdot Q_i, \quad (3)$$

where w_i is the weight of layer i , $0 \leq w_i \leq 1$.

With (3), the quality harmonization between different spatial layers can be adjusted by changing the values of w_i 's. For example, given the scenario described in Section 1, if $w_1 = 1$ and $w_2 = 0$, the truncation will be top-down so as the first spatial layer always has the best possible quality.

It should be noted that, due to interlayer prediction in SVC, the quality of a higher spatial layer depends on the qualities, or more exactly on the bitrates, of lower spatial layers. That is,

$$Q_i = g_i(Q_{i-1}). \quad (4)$$

So truncating all FGS data of lower spatial layers to “make place” for FGS data of the highest spatial layer may not always give the best possible quality for the highest spatial layer. This will be discussed in more detail in the experiments.

As this framework is essentially a resource allocation problem, it can be extended to cover temporal scalability as long as we employ a quality metric that support multidimensional adaptation (e.g., [15]). In the following section, we will present a method based on Viterbi algorithm to solve optimization problem (1).

3. SOLUTION BY THE VITERBI ALGORITHM

Although the FGS data can be truncated finely, the truncation in practice is done in discrete steps (e.g., with a unit of 1 Kbps). So the bitrates R_i 's in the above problem formulation can take discretized values with some step size. Further, as described above, the dependency between spatial layers should be considered in optimization problem (1). So this problem can be solved optimally by the Viterbi algorithm of dynamic programming [16–18]. In the following, we call a *selection* as a discretized truncation operation at a given spatial layer.

The principle of the Viterbi algorithm lies in building a trellis to represent all viable allocations at each instant, given all the predefined constraints. The basic terms used in the algorithm are defined as follows (Figure 3).

- (i) *Trellis*: A trellis is made of all surviving paths that link the initial node to the nodes in the final stage.
- (ii) *Stage*: Each stage corresponds to a spatial layer to be truncated.
- (iii) *Node*: In our problem, each node is represented by a pair (i, a_i) , where i is the stage number, and a_i is the accumulated bitrate of all FGS data until this stage.
- (iv) *Branch*: Given selection k_i at stage i which has the bitrate R_{ik_i} , a node $(i-1, a_{i-1})$ in the previous stage ($i-1$) will be linked by a branch of value $Q_i(k_i, a_{i-1})$ to node (i, a_i) with

$$a_i = a_{i-1} + R_{ik_i}, \quad (5)$$

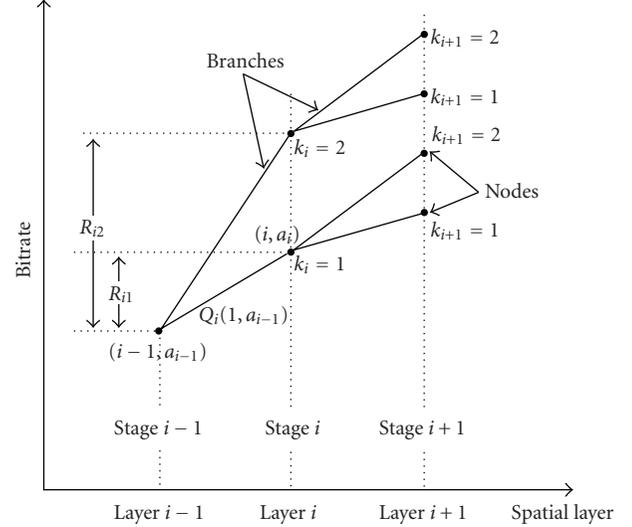


FIGURE 3: Trellis diagram grown by the Viterbi algorithm. Each stage corresponds to a spatial layer, and each branch corresponds to a *selection* for a given spatial layer.

satisfying

$$a_i \leq R^c. \quad (6)$$

- (v) *Path*: A path is a concatenation of branches. A path from the first stage to the final stage corresponds to a set of possible selections for all spatial layers.

In SVC, the higher spatial layers are dependent on the lower spatial layers (but not vice versa). So when the trellis is growing, the stages are arranged in the increasing order of spatial layers (i.e., from the lowest spatial layer to the highest spatial layer). Note that, the first stage (stage 0) is just an initial point, which does not correspond to any spatial layers. Similarly, the quality $Q_i(k_i, a_{i-1})$ depends on not only selection k_i of layer i but also the selections corresponding to previous nodes in the path. Moreover, thanks to the pruning described below, each node (i, a_i) will correspond to only one selection k_i . So we can rewrite $Q_i(k_i, a_{i-1}) = Q_i(k_i, k_{i-1}, \dots, k_1)$.

From the above, we can see that the optimal path, corresponding to the optimal set of selections, is the one having the highest weighted sum $\sum_{i=1}^N w_i \cdot Q_i$. We now apply the Viterbi algorithm to generate the trellis and to find the optimal path as shown in Algorithm 1 [17, 18].

Let K_i denote the number of selections for spatial layer i . With the above algorithm, from the initial node $(0, 0)$, there will be at most K_1 branches growing to K_1 nodes of stage 1. The number of branches will be K_1 if all values of a_1 are not greater than R^c . Similarly, there will be at most K_2 branches grown from each node of stage 1. Due to this growing, there may be more than one branch reaching to the same accumulated bitrate (or arriving to the same node). However, thanks to step 2, there remains only one branch (i.e., the best one) that arrives to a node.

We see that the complexity of this solution depends on the number of layers and the number of selections which is determined by the truncation step size. Officially, the number

Step 0: $i = 0$. Start from the initial node $(0, 0)$.

Step 1: At each stage i , add possible branches to the end nodes of the surviving paths. At each node, a branch is grown for each of the available selections; the branch must satisfy condition (6).

Step 2: Among all paths arriving at a node in stage $i + 1$, the one having the highest accumulated sum of $\sum_{t=1}^{i+1} w_t \cdot Q_t$ is kept, and the rest are pruned.

Step 3: $i = i + 1$. If $i \leq N$, go back to step 1, otherwise go to Step 4.

Step 4: At the final stage, compare all surviving paths then select the path having the highest value of $\sum_{i=1}^N w_i \cdot Q_i$. That path corresponds to the optimal set of selections for all spatial layers.

ALGORITHM 1

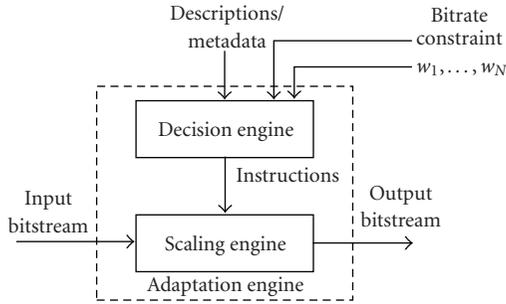


FIGURE 4: Architecture of an SVC adaptation system.

of spatial layers in SVC can be up to 8. However, to maintain a good coding efficiency, an SVC bitstream contains at most three spatial layers (with different resolutions) [7]. As shown later in the next section, with practical conditions, the optimal solution based on the Viterbi algorithm can be found in real time.

It should be noted that the solution provided by the above algorithm is optimal for the “discretized” problem. However, as mentioned earlier, the practical truncation is often based on a specific step size. From our experience, a truncation equal to 1% of the total FGS bitrate would not result in any perceptual difference. So, practitioners would look for a solution of the discretized problem, rather than the *continuous-valued* problem.

Currently, the R-D information (i.e., R_i , Q_i) in our framework is operational. Although the operational R-D data is not easy to obtain in real time, they can be computed in advance and used as metadata to adapt the bitstream on the fly as in previous work of video coding [16, 19]. Moreover, some analytical models can be used to represent the R-D information in a compact manner [9, 19].

4. EXPERIMENTS

In this section, some experiments are presented to show the flexibility and usefulness of our proposed framework. We developed an SVC adaptation engine which consists of a de-

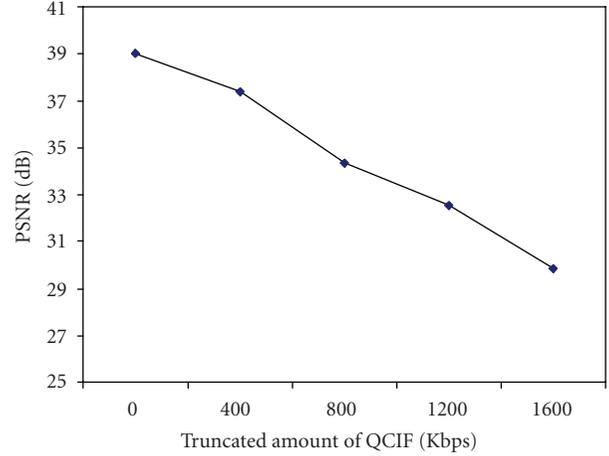


FIGURE 5: R-D information of QCIF layer. The FGS truncation is applied to QCIF layer only.

cision engine and a scaling engine (Figure 4). The decision engine employs metadata about the operational R-D information of input bitstream, and other metadata including bitrate constraint, the weights w_i 's of spatial layers, and then provides as output the adaptation instructions. The instructions here are the amount of FGS bitrate which should be truncated in each spatial layer. The scaling engine takes the instructions and adapts the input bitstream accordingly.

4.1. Allocation results

Test videos are encoded by the recent software JSVM7.12. The results presented below are for the football video, encoded with 2 spatial layers, QCIF and CIF both having frame rate of 30 fps and GOP size of 16. Correspondingly, two users will consume this content as in the scenario of Section 1. The base quality QP values of both spatial layers are 38. QCIF spatial layer is enhanced by 3 FGS layers and CIF spatial layer by 2 FGS layers. The FGS bitrates of CIF and QCIF layers are, respectively, 1924 (Kbps) and 1877 (Kbps). We assume that users have no special requests on the quality threshold (i.e., Q_i^{\min}). Quality metric used in optimization problem (1) is PSNR value averaged over all video frames. The overall quality is given by

$$OQ = w_1 \cdot Q_1 + w_2 \cdot Q_2. \quad (7)$$

For ease of presentation and discussion, the step size for FGS truncation is set to be 400 (Kbps) and the quality is shown according to the amount of truncated bitrate. Each spatial layer will be truncated at four points, namely, 400, 800, 1200, and 1600. Figures 5 and 6 show the operational R-D information of QCIF layer and CIF layer according to the amount of truncated data.

Now suppose that $w_1 = 0.33$ and $w_2 = 0.67$. These weight values would give some balance between the two spatial layers as the PSNR value of QCIF layer is often higher than that of CIF layer. The objective of truncation will be to optimize the overall quality $OQ = 0.33 \cdot Q_1 + 0.67 \cdot Q_2$. The optimal selections are represented by the solid path (denoted by

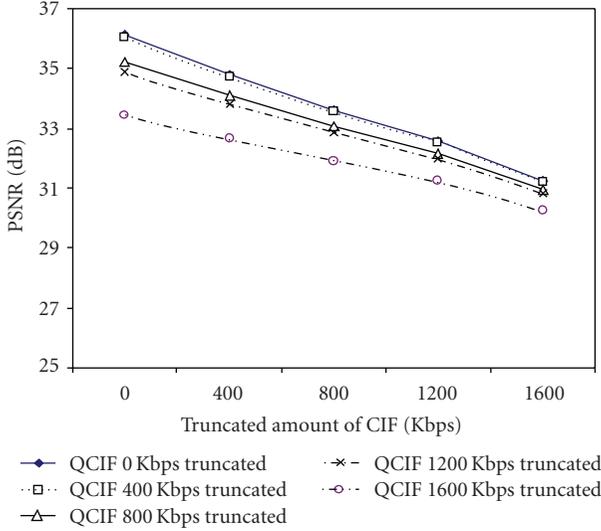


FIGURE 6: R-D information of CIF layer. The FGS truncation is applied to both QCIF and CIF layers.

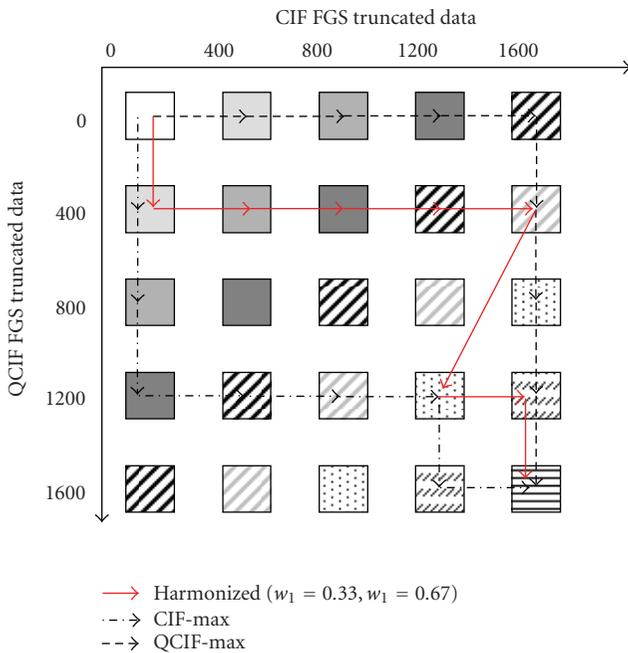


FIGURE 7: Illustration of different FGS truncation methods. Here FGS data in CIF and QCIF layers are truncated flexibly.

harmonized path) in Figure 7. We can see that when the total truncated amount is increased (from 0 Kbps to 3200 Kbps, with step size of 400 Kbps), the selections of multilayer truncation correspond to the boxes (400, 0), (400, 400), (400, 800), (400, 1200), (400, 1600), (1200, 1200), (1200, 1600), (1600, 1600), where (a, b) indicates that truncated amounts of QCIF and CIF layers are, respectively, a Kbps and b Kbps. Note that, in Figure 7, the boxes of the same pattern and gray level have the same total amount of truncated data (in both CIF and QCIF layers).

If $w_1 = 1$ and $w_2 = 0$, this implies a top-down truncation used always to maximize QCIF layer’s quality. Obviously, the selections in this case are represented by the dashed path (denoted as *QCIF-max* path), where FGS data of CIF layer are truncated first.

If $w_1 = 0$ and $w_2 = 1$, this implies a truncation that aims to maximize CIF layer’s quality. The selections in this case are represented by the dashed-dotted path (denoted as *CIF-max* path). As shown by this path, FGS data of QCIF layer are first truncated until the amount of 1200 (Kbps), then FGS data of CIF layer are truncated. Here, the selections of (1600, 400) and (1600, 800) are not used because a truncated amount of 1600 (Kbps) in QCIF layer would result in a significant degradation in CIF layer due to interlayer prediction. So, FGS data of QCIF layer will not be completely truncated before truncating CIF FGS data. That is, a bottom-up truncation would not be a good choice for most practical conditions.

Figure 8 shows the advantage of the harmonized truncation in detail. The weight values are as above, $w_1 = 0.33$ and $w_2 = 0.67$. In these figures, the horizontal axis represents the total amount of truncated FGS data (in both CIF and QCIF layers), and the vertical axis represents the PSNR values of each spatial layer (QCIF in Figure 8(a) and CIF in Figure 8(b)). We can see that, with *CIF-max* truncation, the quality of the CIF layer is always maximized (Figure 8(b)), but the quality of QCIF layer decreases very quickly (Figure 8(a)). With *QCIF-max* truncation, the phenomenon is inverted. Meanwhile, the curve of harmonized truncation shows an intermediate solution between these two extreme cases. For example, when the total amount of truncated data is 1600 Kbps, the quality of QCIF layer is 37.4 dB, that is, 4.9 dB higher than that of *CIF-max* truncation; and the quality of CIF layer is 32.54 dB, that is, 1.3 dB higher than that of *QCIF-max* truncation.

Now let $w_1 = 0.15$ and $w_2 = 0.85$, which implies an emphasis on the CIF layer. The solution provided by the above algorithm corresponds to the path of (400, 0), (400, 400), (1200, 0), (1200, 400), (1200, 800), (1200, 1200), (1200, 1600), and (1600, 1600). Figure 9 shows the corresponding quality comparison. We can see that the harmonized curve now gets close to the *CIF-max* curve. However, at some points, the gain in QCIF layer is still several dBs compared to *QCIF-max* method (Figure 9(a)). So, by adjusting the weight values, we can flexibly control the tradeoff between the two layers. We found that the shapes of curves having finer steps are very similar to those of the current curves. This means that the current curves (with step size of 400 kbps) represent sufficiently the adaptation behavior.

When the weight values are equal ($w_1 = 0.5$ and $w_2 = 0.5$), the harmonized truncation of this given bitstream turns out to be the same as *QCIF-max* truncation. This is due to the fact that the PSNR value of QCIF layer is often higher than that of CIF layer (as mentioned above), so the QCIF layer is always “emphasized” in truncation process. This means that the *intuitive* nonweighted sum of PSNR values of CIF and QCIF layers would not give any tradeoff for the two layers.

Figures 10 and 11 show the optimality of the harmonized path compared to the *CIF-max* and *QCIF-max* paths for two case, ($w_1 = 0.33$, $w_2 = 0.67$) and ($w_1 = 0.15$, $w_2 = 0.85$). The

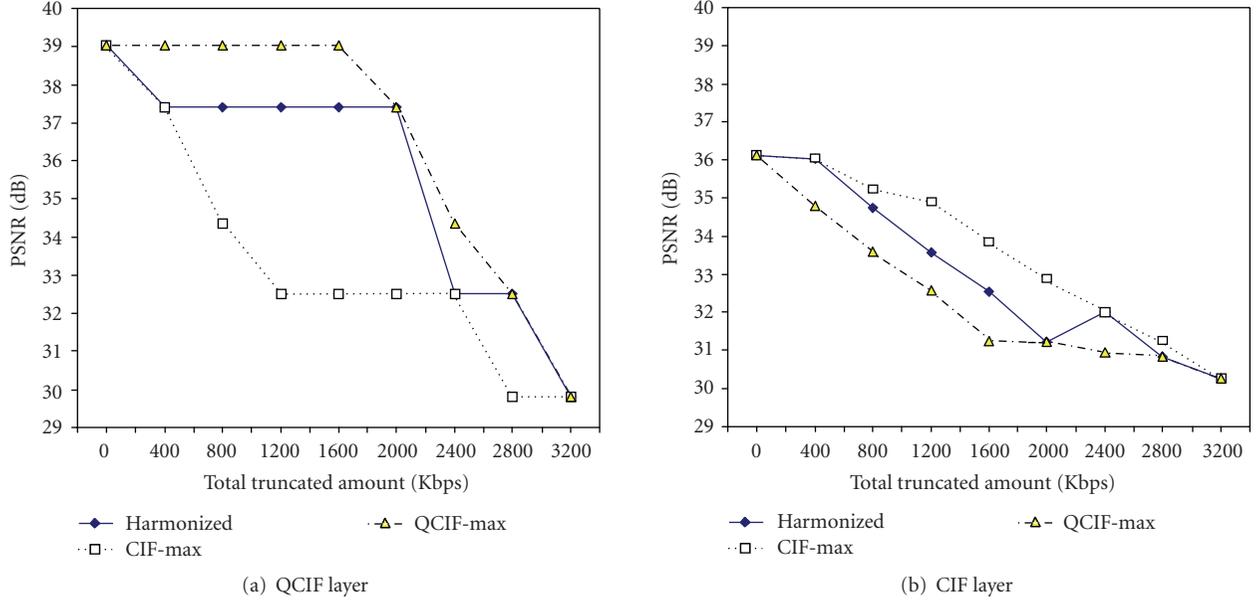


FIGURE 8: Comparison of three truncation methods: harmonized (with $w_1 = 0.33$, $w_2 = 0.67$), CIF-max, and QCIF-max.

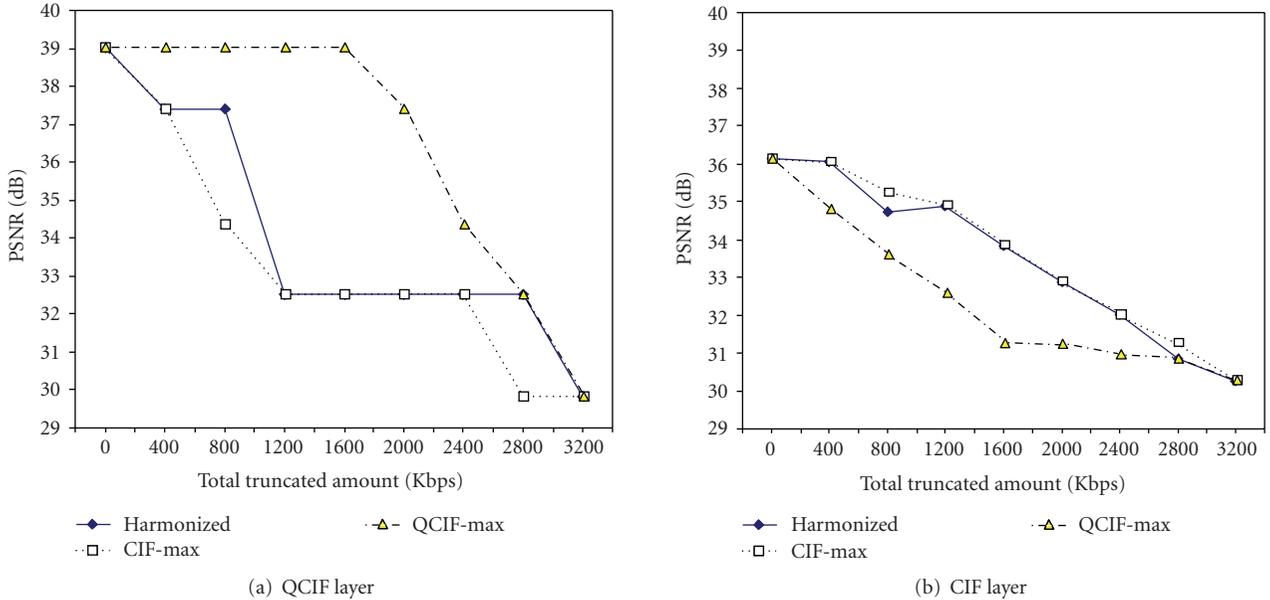


FIGURE 9: Comparison of three truncation methods: harmonized (with $w_1 = 0.15$, $w_2 = 0.85$), CIF-max, and QCIF-max.

horizontal axis represents the total amount of truncated FGS data, and the vertical axis represents the overall quality computed by (7). We can see that the overall quality of the harmonized path is always higher than or equal to those of the other two paths. This means that the truncations based on CIF-max and QCIF-max paths cannot provide the optimal results.

It should be noted that the PSNR value in Figures 10 and 11 just represents the collective quality, which is used to guarantee the optimal tradeoff between layers. In order to see the advantage of our proposed method in improving users' quality, one should also consider the R-D curves of specific spa-

tial layers (i.e., Figures 8 and 9). For example, though the gaps between the curves of Figure 10 are sometimes small, the actual improvement for specific users may be up to several dBs as seen in Figures 8(a) and 8(b). We have found similar observations with other sequences. In fact, as long as there exists a gap between the two extreme truncations, a tradeoff between them can always be achieved.

4.2. Algorithm complexity

To check the complexity of the algorithm, we measure the processing time of the algorithm with different step sizes,

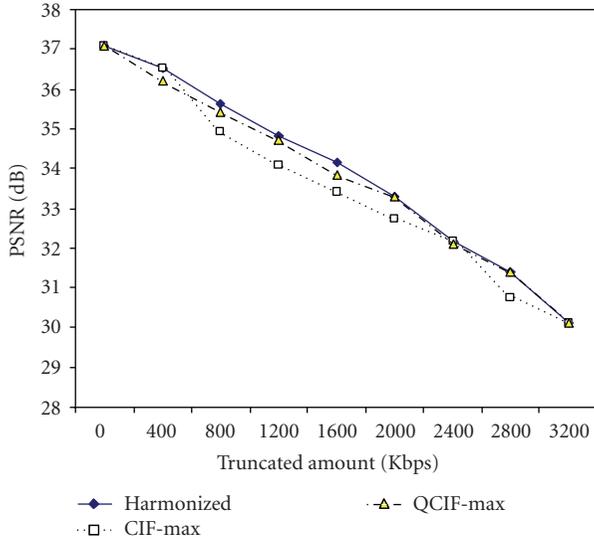


FIGURE 10: Overall quality of different truncation solutions ($w_1 = 0.33$ and $w_2 = 0.67$).

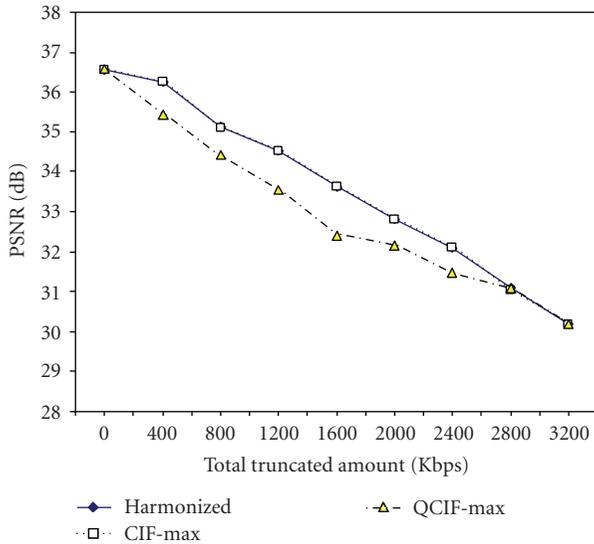


FIGURE 11: Overall quality of different truncation solutions ($w_1 = 0.15$ and $w_2 = 0.85$).

namely, 1 Kbps, 2 Kbps, 5 Kbps, and 10 Kbps. The quality values of new truncation selections are linearly interpolated from the previous sample points obtained with the step size of 400 Kbps (which is similar to [20]). The complexity is represented by processing time which is measured by the number of system clock ticks (1000 ticks per second). The proposed algorithm is run on a notebook having Pentium M 1.86 GHz processor and 1 G RAM. Figure 12 shows the processing time with respect to the total amount of truncated bitrate. We can see that when the step size is 1 Kbps, the processing time can be up to 80 milliseconds; however, with the other step sizes, the processing time is just around 20 milliseconds. Especially, when step size is 10 Kbps, the complexity become so small that the processing time is mostly zero (more exactly, less than 1 tick).

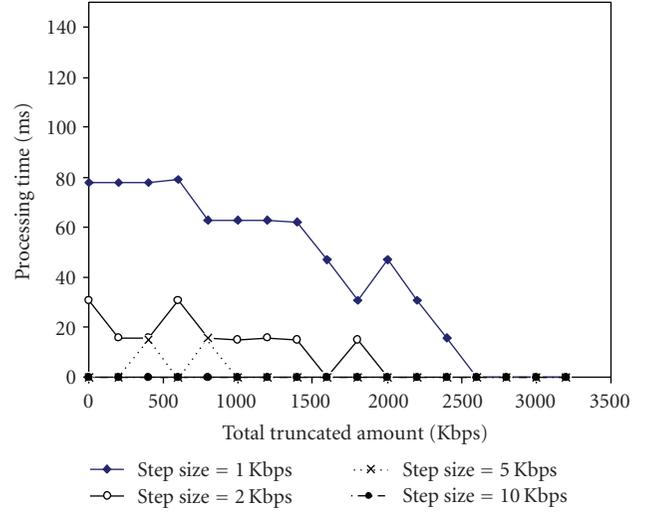


FIGURE 12: Processing time with different step sizes (2-layer bit-stream).

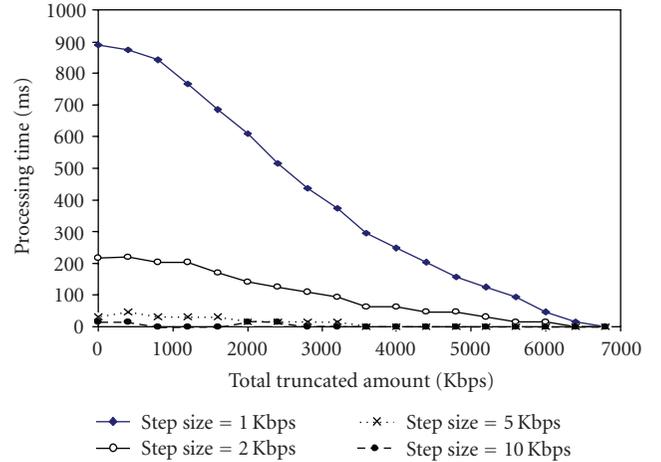


FIGURE 13: Processing time with different step sizes (3-layer bit-stream).

As the number of spatial layers of an SVC bitstream is at most 3 in practice [7], we add to the bitstream one more spatial layer (4CIF), of which the amount of FGS data is 3500 Kbps. The algorithm is run again with step sizes of 1 Kbps, 2 Kbps, 5 Kbps, 10 Kbps and the corresponding results are shown in Figure 13. Now we see that the processing time with step size of 1 Kbps increases significantly which is up to 900 milliseconds. However, when step size is 10 Kbps, still the processing time is usually less than 1 millisecond, sometimes reaching to 15 milliseconds. Note that, with this bitstream, even the step size of 10 Kbps is less than 0.2% of the total FGS bitrate.

Meanwhile, it should be noted that in practical video communication, the acceptable processing delay can be up to 400 milliseconds for two-way application and 10 seconds for one-way application [21].

Obviously, with a bitstream of higher bitrate, the step size should be increased proportionally. Whereas, from the above

example we can see that even if the step size is just 0.5% or 1% of the total bitrate, the processing time of the Viterbi algorithm would become negligible. Moreover, from our previous experience with subjective tests on video quality [22], with quality scale of just 9 or 10 levels, it is still very difficult for end-users to differentiate the adjacent quality levels. This means that the step size may not need to be as small as 1% of the total bitrate. The exact step size which results in the just noticeable difference (JND) in user perception is an interesting issue in our future work.

From the above, we can see that when there is any change in user requests or in bitrate constraint, the optimization problem can be recomputed on the fly and the adaptation will be seamless to the users. This means that our proposed framework can provide the truncation flexibility with optimal result for any conditions of bitrate constraint and quality tradeoff between layers.

5. CONCLUSIONS

In this paper, we proposed a general framework to adapt SVC bitstream through FGS truncation across multiple spatial layers. Our proposed framework has the flexibility in allocating the resource (i.e., bitrate) among spatial layers, where the overall quality is defined as a function of all spatial layers' qualities and can be modified on the fly. The adaptation process of the proposed framework was formulated as a constrained optimization problem and then optimally solved by the Viterbi algorithm. Through experiments, we also showed that the current approaches of FGS truncation were special cases of our general framework. For future work, we will consider some perceptual quality metrics in our adaptation system and employ analytical models for R-D representation. Also, the framework will be extended to cover other constraints of heterogeneous environments, such as terminal capability and packet loss.

ACKNOWLEDGMENTS

The authors would like to thank Dong Su Lee of ICU for his help in this work. This work was supported by the IT R&D program of MIC/IITA [2005-S-103-03, Development of Ubiquitous Content Access Technology for Convergence of Broadcasting and Communications] and by 2nd Phase of Brain Korea 21 project sponsored by Ministry of Education and Human Resources Development (Seoul, South Korea).

REFERENCES

- [1] A. Vetro, "MPEG-21 digital item adaptation: enabling universal multimedia access," *IEEE Multimedia*, vol. 11, no. 1, pp. 84–87, 2004.
- [2] S.-F. Chang and A. Vetro, "Video adaptation: concepts, technologies, and open issues," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 148–158, 2005.
- [3] A. Vetro, "Transcoding, scalable coding and standardized metadata," in *Proceeding of the 8th International Workshop on Visual Content Processing and Representation (VLBV '03)*, vol. 12849, pp. 15–16, Madrid, Spain, September 2003.
- [4] 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.
- [5] H. Schwarz, D. Marpe, and T. Wiegand, "SNR-scalable extension of H.264/AVC," in *Proceedings of IEEE International Conference on Image Processing (ICIP '04)*, vol. 5, pp. 3113–3116, Singapore, October 2004.
- [6] ITU-T ISO/IEC JTC1, "Video coding for generic audiovisual services," ITU-T Recommendation H.264 ISO/IEC 14496-10 AVC, 2003.
- [7] Joint Scalable Video Model (JSVM)9.0. ITU-T VCEQ JVT-V202, Marrakech, January 2007.
- [8] D. Mukherjee, E. Delfosse, J.-G. Kim, and Y. Wang, "Optimal adaptation decision-taking for terminal and network quality-of-service," *IEEE Transactions on Multimedia*, vol. 7, no. 3, pp. 454–462, 2005.
- [9] C. Hsu and M. Hefeeda, "Rate-distortion models for FGS-encoded video sequences," in *Proceeding of the 17th International Conference on Computer Theory and Applications (ICCTA '06)*, pp. 334–337, Alexandria, Egypt, September 2006.
- [10] T. Kim and M. H. Ammar, "Optimal quality adaptation for MPEG-4 fine-grained scalable video," in *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '03)*, vol. 1, pp. 641–651, San Francisco, Calif, USA, March-April 2003.
- [11] J. W. Kang, S.-H. Jung, J.-G. Kim, and J.-W. Hong, "Development of QoS-aware ubiquitous content access testbed," in *Proceedings of the International Conference on Consumer Electronics (ICCE '07)*, pp. 1–2, Las Vegas, Nev, USA, January 2007.
- [12] T. C. Thang, Y. S. Kim, Y. M. Ro, J. W. Kang, and J.-G. Kim, "SVC bistream adaptation in MPEG-21 multimedia framework," in *Proceeding of the International Packet Video Workshop (PV '06)*, Hangzhou, China, April 2006.
- [13] M. Mathew, K. Lee, and W.-J. Han, "Multi layer quality layers," ITU-T VCEQ JVT-S043, Geneva, April 2006.
- [14] Z. Li, J. Huang, and A. K. Katsaggelos, "Pricing based collaborative multi-user video streaming over power constrained wireless down link," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '06)*, vol. 5, Toulouse, France, May 2006.
- [15] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312–322, 2004.
- [16] A. Ortega, K. Ramchandran, and M. Vetterli, "Optimal trellis-based buffered compression and fast approximations," *IEEE Transactions on Image Processing*, vol. 3, no. 1, pp. 26–39, 1994.
- [17] G. D. Forney, "The Viterbi algorithm," *Proceedings of IEEE*, vol. 61, pp. 268–278, 1973.
- [18] T. C. Thang, Y. J. Jung, and Y. M. Ro, "Effective adaptation of multimedia documents with modality conversion," *Signal Processing: Image Communication*, vol. 20, no. 5, pp. 413–434, 2005.
- [19] X. M. Zhang, A. Vetro, Y. Q. Shi, and H. Sun, "Constant quality constrained rate allocation for FGS-coded video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 2, pp. 121–130, 2003.
- [20] L.-J. Lin and A. Ortega, "Bit-rate control using piecewise approximated rate-distortion characteristics," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 4, pp. 446–459, 1998.
- [21] ITU-T, "End-user multimedia QoS categories," Recommendation G.1010, 2001.
- [22] T. C. Thang, Y. J. Jung, and Y. M. Ro, "Modality conversion for QoS management in universal multimedia access," *IEE Proceedings: Vision, Image and Signal Processing*, vol. 152, no. 3, pp. 374–384, 2005.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

