Popularity-Aware Rate Allocation in Multi-View Video

Attilio Fiandrotti[a], Jacob Chakareski[b], Pascal Frossard[b]

[a]Computer and Control Engineering Department, Politecnico di Torino, Turin, Italy
[b]Signal Processing Laboratory (LTS4), Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

ABSTRACT

We propose a framework for popularity-driven rate allocation in H.264/MVC-based multi-view video communications when the overall rate and the rate necessary for decoding each view are constrained in the delivery architecture. We formulate a rate allocation optimization problem that takes into account the popularity of each view among the client population and the rate-distortion characteristics of the multi-view sequence so that the performance of the system is maximized in terms of popularity-weighted average quality. We consider the cases where the global bit budget or the decoding rate of each view is constrained. We devise a simple rate-video-quality model that accounts for the characteristics of interview prediction schemes typical of multi-view video. The video quality model is used for solving the rate allocation problem with the help of an interior point optimization method. We then show through experiments that the proposed rate allocation scheme clearly outperforms baseline solutions in terms of popularity-weighted video quality. In particular, we demonstrate that the joint knowledge of the rate-distortion characteristics of the video content, its coding dependencies, and the popularity factor of each view is key in achieving good coding performance in multi-view video systems.

Keywords: multi-view video, rate allocation, popularity-driven, rate-video-quality modeling, Lagrange optimization, 3DTV

1. INTRODUCTION

Video applications have recently experienced important changes due to both the need for enriched and interactive services and the development of new vision sensors. In particular, multi-view video has been receiving a lot of attention lately, as it offers the possibility to encode and deliver simultaneously several views that represent the same scene from different perspectives. Multi-view video opens the door to many novel and exciting applications such as three-dimensional television (3DTV) or immersive communications, for example. Furthermore, the availability of multiple views offers the possibility for the users to choose the content to be displayed in television or gaming services; it certainly represents an interesting solution for interactive multimedia systems.

The definition of multiple views however clearly increases the storage and bandwidth requirements in interactive television services. At the same time, the multiple views certainly convey highly redundant information due to both temporal and spatial correlation in the set of image sequences. This redundancy can however be drastically reduced by spatio-temporal prediction during the encoding process. Typically, a joint encoder in multi-view video can predict an image from neighbor images in the same view or in adjacent views. Recent standardization efforts in the H.264/Multi-view Video Codec (MVC)[1] have shown that joint multi-view encoding frequently achieves better overall compression efficiency than H.264/AVC-based simulcast,[2] which simply consists of independent encoding and transmission of the different views. However, motion and disparity compensation in joint encoding introduces a lot of dependencies between the images. These dependencies have to be considered carefully in the coding strategy and particularly in the bit allocation strategy when the coding rate is constrained.

In this paper, we address the problem of rate allocation in multi-view video coding for interactive television systems. We consider that the different views have different popularity as they get different number of subscribers, so that the performance of the system is measured as a popularity-weighted average video quality. Then we address two main allocation problems. In one case, the total bit budget for all the views has to be minimized in order to control the resources required by the system. In the other case, the bit rate necessary to decode any of the views in the interactive system is also constrained when users have limited access bandwidth. This
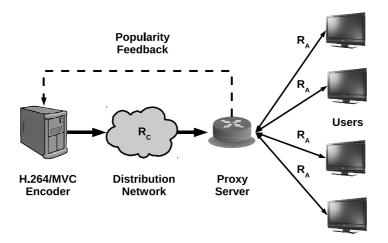
Figure 1: An MVC streaming scenario with global rate constraint R_C and access bandwidth constraints R_A.

decoding bit rate includes the rate of the view of interest as well as the coding rate of the reference views. The rate constraints are illustrated in a typical MVC streaming framework shown in Figure 1.

We first propose a simple rate-distortion model for multi-view video, where the quality of each view follows a increasing logarithmic function of the view encoding rate. In addition, this quality is driven by the quality or the encoding rate of its direct predictor view. We then formulate a Lagrangian optimization problem that targets an efficient bit allocation among the different views, such that the popularity-weighted video quality is maximized while a minimal quality is guaranteed for each of the views and, at the same time, constraints are imposed on the overall coding rate or on the transmission rate of any view. This optimization problem is solved by an interior point method.[3] We then validate our rate-distortion model by coding experiments with common multi-view sequences. We show that our rate allocation strategy performs better than baseline solutions in terms of popularity-weighted average quality in the cases where the total rate or the decoding rate of any view is constrained. In particular, we show that the distribution of the quality in the different views follows closely the view popularity distribution and that the gain in average quality can exceed 1 dB. These performance improvements are due to the fact that our rate allocation strategy considers jointly view popularity, prediction dependencies, and rate-distortion characteristics when computing its coding decisions.

The resource allocation problem has been widely studied in the video communication community, but the case of multi-view video coding has surprisingly been largely overlooked. For example, Chakareski et al. have addressed the resource allocation problem in the scenario where independent video sequences are transmitted over a shared medium.[4] They propose an optimization framework that achieves optimal performance through an accurate modeling of the rate-distortion characteristics of the contents. While a similar optimization framework based on accurate rate-distortion modeling could be extended to multi-view video, the increased level of dependencies renders the problem quite complex in this case. A few works have studied the effects of interview prediction in multi-view coding[5] or the modeling of stereoscopic video in the context of communications over lossy channels.[6] The latter introduces a rate-distortion model that takes into account the interview prediction between left and right views and uses it to optimally allocate the resources in the network. However, the extension of such a framework to a high number of views is not trivial. To the best of our knowledge, the joint consideration of view popularity, coding dependencies and rate-distortion characteristics for multi-view video communication under bandwidth constraints has not been addressed before.

The structure of this work is organized as follows. In Section 2, we formulate the rate allocation problem that targets the maximization of the popularity-weighted average quality-of-service. Section 3 then proposes experiments that validate our simple rate-distortion model and examine the performance of our rate allocation strategy. Finally, conclusions are drawn in Section 4.

2. RATE-VIDEO-QUALITY OPTIMIZATION

Let there be N views of a video scene. The content is experienced by an audience comprising U users. Each user is characterized with an access link of capacity R_A. Let u_i denote the number of users interested in view $i = 1, \ldots, N$. Then, the popularity factor of view i is defined as $w_i = u_i/U$. We are interested in assigning encoding rates R_i, for $i = 1, \ldots, N$, to the various views such that their overall popularity-weighted video quality is maximized. The optimal allocation needs to satisfy several rate and video quality constraints. In particular, (i) the overall rate $\sum_i R_i$ should not exceed a total bit-rate budget R_C, (ii) the video quality of each view should not drop below a view-specific threshold, and (iii) the capacity of the access link of a user should not be exceeded. The above optimization can be formally written as

$$\max_{\mathbf{R}} \sum_{i=1}^{N} w_i Q_i(\mathbf{R}) \tag{1}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} R_i \leq R_C \,,$$

$$Q_i(\mathbf{R}) \geq Q_C^{(i)}, \quad \text{for } i = 1, \ldots, N \,,$$

$$\sum_{j \preceq i} R_j \leq R_A, \quad \text{for } i = 1, \ldots, N \,,$$

where $\mathbf{R} = (R_1, \ldots, R_N)$ denotes the vector of allocated rates and $Q_i(\mathbf{R})$ denotes the video quality of a view as a function of the rate allocation. Furthermore, $Q_C^{(i)}$ denotes the minimum video quality threshold for view i, while the last line of constraints in (1) captures the fact that for decoding view i all its ancestor views ($j \preceq i$) in the multi-view compression hierarchy need to be received as well.

To reduce the complexity of the optimization problem in (1) we model the functions $Q_i(\mathbf{R})$ as follows. If view i is independently encoded, i.e., with no reference to any other view, then $Q_i(\mathbf{R})$ becomes $Q_i(R_i)$ which we formulate as

$$Q_i(R_i) = a_i + b_i * log(R_i) \tag{2}$$

where the parameters a_i and b_i are estimated empirically from actual compressed multi-view content. Logarithmic models, similar to (2), have been commonly used in studies involving compressed single-view (monoscopic) video content.[7,8] On the other hand, for all predictively encoded views i we simplify $Q_i(\mathbf{R})$ to be a function only of the rates allocated to its reference view(s), in addition to R_i. Specifically, let view i be bi-directionally predicted from views j and l. Then, we write

$$
\begin{aligned}
Q_i(\mathbf{R}) &= Q_i(R_i, R_j + R_l) \\
&= \frac{R_j + R_l - R_{min}^{(j+l)}}{R_{max}^{(j+l)} - R_{min}^{(j+l)}} Q_i(R_i | R_j + R_l = R_{max}^{(j+l)}) + \frac{R_{max}^{(j+l)} - R_j - R_l}{R_{max}^{(j+l)} - R_{min}^{(j+l)}} Q_i(R_i | R_j + R_l = R_{min}^{(j+l)}),
\end{aligned}
\tag{3}
$$

where $R_{max}^{(j+l)}$ and $R_{min}^{(j+l)}$ are parameters that represent the maximum and minimum rate values that the sum $R_j + R_l$ can achieve, while $Q_i(R_i | R_j + R_l = R_{min}^{(j+l)})$ and $Q_i(R_i | R_j + R_l = R_{max}^{(j+l)})$ correspond to the model in (2) describing the quality-rate characteristics of view i when its reference views are encoded at the sum rates $R_{min}^{(j+l)}$ and $R_{max}^{(j+l)}$, respectively. Again, these two characteristics are obtained empirically from the actual compressed multi-view sequence. Note that in (3) to further reduce complexity we modeled $Q_i(\mathbf{R})$ only as a function of $R_l + R_j$ rather than of their individual values. Finally, in the case of views i encoded predictively from a single reference view j, the expression (3) is still employed to obtain $Q_i(\mathbf{R}) = Q_i(R_i, R_j)$ where instead of the sum $R_l + R_j$ we simply use now R_j only. Correspondingly, the minimum and maximum rate parameters then become $R_{min}^{(j)}$ and $R_{max}^{(j)}$, respectively.
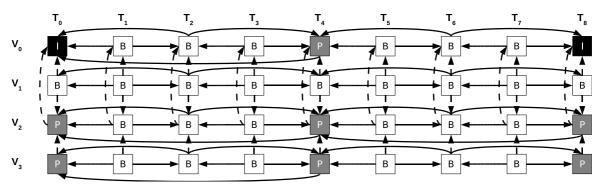
Figure 2: Simplified GOP structure of the encoding scheme used in this work.

The optimization in (1) represents a convex programming problem. By employing our models in (2) and (3) we solve our constrained non linear optimization problem using the interior point method implementation provided by Matlab optimization toolbox.[9] In Section 3, we examine the performance of the proposed optimization and quality-rate models on different multi-view sequences.

3. EXPERIMENTAL RESULTS

3.1 Setup

We briefly describe the setup used in our experiments. First, we choose to use the coding structure illustrated in Figure 2, based on a pyramidal temporal prediction scheme where the GOP size is equal to four pictures. Only for view zero, that is the main view, one picture on every eight is Intra-coded for improved random temporal accessibility. Even-numbered views are predicted by the lower-id, even-numbered view (e.g.: view two is predicted from the main view.), while odd-numbered views are bipredicted from the two adjacent neighbor views (e.g.: view one is predicted from view two and the main view.) This coding structure has been found to be a good solution for our streaming framework among the coding schemes proposed in.[2] In particular, the use of bipredictive frames permits to reduce the number of reference frames one has to decode in order to display one specific view, since the decoding path becomes shorter on the average when bipredicted pictures are used instead of predicted pictures exclusively. This means that the bandwidth requirements are generally reduced in our streaming scenario, or equivalently that the encoding quality is higher for the given bandwidth constraints. While the coding structure of Figure 2 represents a good compromise between coding efficiency and flexibility in an interactive streaming scenario, the algorithms proposed in this paper apply to any multiview coding structure.

We have then used two multiview video sequence, *Breakdancer*[10] and *Race*.[11] These sequences have eight views each, 100 frames per view and a CIF resolution. We have encoded these sequences at multiple rates in order to build the video-quality-rate model proposed in the previous section. Since the MVC reference encoder JMVC[12] lacks rate control capabilities, we have implemented the quadratic rate control algorithm described in[13] for the construction of the quality-rate model. This algorithm is used in the H.264/SVC JSVM reference encoder,[14] in the H.264/AVC JM reference encoder[15] and is at the basis of a proposal for rate control in MVC.[16] We focus on a target encoding quality in the range of $30 \sim 40$ dB, to ensure an acceptable viewing quality. This respectively represents bitrates in the range of $50 \sim 250$ Kbps and $100 \sim 350$ Kbps for the *Race* and *Breakdancer*. The resulting rate-quality values are used to compute the parameters of the quality-rate model in Eqs. (2) and (3).

Finally, we consider three different popularity distribution functions in order to model the relative number of users that request the different views in the multiview streaming system. In particular, we consider the *Flat* distribution, where all views have all the same popularity, and *Gaussian* and *Exponential* distributions, where the main view has the highest popularity and the other views have a popularity that follow a Gaussian or an exponential function, respectively. We further set the minimal quality of any view to be $Q_m^{(i)}$ for 30 dB, irrespectively of the popularity distribution.

3.2 MVC Video-Quality-Rate Model
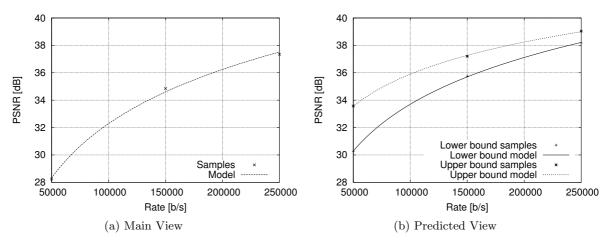

(a) Main View

(b) Predicted View

Figure 3: QR characteristics of the Breakdancer sequence.

We illustrate the accuracy of our Quality-Rate (QR) model by comparing sets of collected samples with the corresponding logarithmic models. Figure 3a shows samples of the main view of the Breakdancer sequence collected at 50, 150 and 250 Kb/s encoding rates. The figure also shows the corresponding interpolated logarithmic curve as described in Eq. (2), where parameters a and b are set, respectively, to -33.46 and 5.71. The figure shows that the logarithmic curve interpolates accurately the collected samples. Similarly, Figure 3b shows two sets of samples for the second view of the Breakdancer sequence. The figure also shows the corresponding logarithmic curves described in Eq. (3). The close match between samples and curves shows that our model can accurately estimate predicted views as well. Similar results were obtained for the Race sequence. Then, we compare expected and actual quality for a set of test encodings and calculate the prediction error as shown in Table 1 (every view is encoded at 150 Kb/s.) On the average, the error between predicted and actual encoding PSNR is lower than two percents, which demonstrates the validity of the model.

View	Type	Expected PSNR [dB]		Actual PSNR [dB]		Error [%]	
		Breakdancer	Race	Breakdancer	Race	Breakdancer	Race
Main	AVC	37.52	35.87	37.34	36.05	0.46	0.51
One	MVC-B	37.68	38.33	37.63	38.43	0.13	0.28
Two	MVC-P	39.00	37.64	39.04	37.56	0.10	0.21
Three	MVC-B	37.78	38.36	37.68	38.45	0.24	0.25
Four	MVC-P	40.03	38.51	40.02	38.55	0.01	0.10
Five	MVC-B	38.16	38.71	38.30	38.84	0.38	0.34
Six	MVC-P	39.36	37.86	39.37	37.92	0.02	0.17
Seven	MVC-P	37.95	38.26	37.85	38.37	0.26	0.29

Table 1: Expected and actual encoding PSNR for Breakdancer and Race sequences.

3.3 Network Constrained Streaming

We explore in this section the case where the overall encoding rate is bounded uniquely by the constraint R_C (i.e., $R_A = \infty$ in Eq. (1).)

We introduce two rate allocation baseline strategies for performance evaluation. Both strategies allocate a given bit budget R_C without any knowledge of the QR characteristics of the video content. The first baseline strategy (*Baseline-A*) is popularity unaware and simply allocates the available bandwidth R_C in equal shares for each view. The second baseline strategy (*Baseline-B*) is aware of the popularity factor and allocates the bit budget
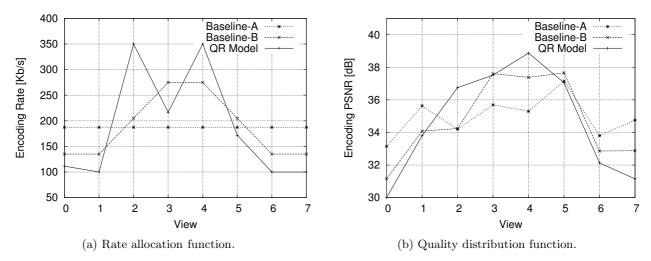
(a) Rate allocation function.

(b) Quality distribution function.

Figure 4: Encoding rate and quality for the different views, *Race* sequence, *Gaussian* user distribution, $R_C = 1.5$ Mb/s, $R_A = \infty$.

proportionally to the popularity of the views. In detail, it first allocates every view a minimum bandwidth, while the remaining bit budget is allocated among the views according to their popularity. Since both baseline strategies are totally unaware of the QR characteristics of the video content, they cannot however guarantee any minimum quality. Finally, note that when the view popularity is even (Flat distribution) both baseline strategies are equivalent.

Sequence	Flat Distribution		Gaussian Distribution			Exponential Distribution		
	Proposed	Gain vs Base-A/B	Proposed	Gain versus		Proposed	Gain versus	
				Base-A	Base-B		Base-A	Base-B
Breakdancer	38.40	0.38	39.00	0.98	0.32	38.52	0.50	0.21
Race	35.77	0.54	36.65	1.40	0.48	36.22	0.97	0.27

Table 2: Weighted encoding PSNR for different bit allocation strategies ($R_C = 1.5$ Mb/s, $R_A = \infty$.)

Table 2 compares our proposed rate allocation scheme with the two baseline strategies. For every distribution of users we report the weighted quality achieved by our scheme and the gain with respect to the baseline schemes (higher numbers correspond to better performance of our framework). When the popularity stays even for all the views (i.e., *Flat* popularity distribution), our rate allocation scheme performs better than both baseline solutions. In this case, the knowledge of the QR characteristics of the video content is the unique key to better quality. When the user population becomes non uniform, *Baseline-B* performs better than *Baseline-A* due to its awareness of the popularity factor. However, our proposed strategy outperforms both baseline schemes because it is aware of both the popularity factor and the characteristics of the video content.

An detailed look at how the various strategies allocate the bit budget helps to understand why our proposed scheme outperforms baseline schemes. Figure 4a shows how our proposed strategy and the two baseline schemes allocate the rate for a *Gaussian* popularity distribution. The corresponding quality curves are shown in Figure 4b. *Baseline-A* allocates the rate evenly among the views: clearly this is the worst possible option since it neglects both the popularity factor and the QR-characteristics. In fact, not only *Baseline-A* achieves the worst quality as shown in Table 2 (loss of 1.40 dB with respect to *Proposed*), but the quality curve does not match the user distribution function at all. *Baseline-B* allocates the rate so that the rate allocation function matches the user distribution, achieving better weighted quality than *Baseline-A* and showing a quality distribution function that resembles more closely the user distribution function. Finally, we see that the rate allocation function of the proposed strategy accounts both for the user distribution function and the characteristics of the encoded content.

As a result, it achieves higher weighted quality while its quality distribution function closely matches the user distribution function. In particular, we see that the proposed strategy allocates different rates to views three and four, as well as to views zero and one, despite equal popularity. Indeed, the proposed strategy is aware of the coding dependencies between views and allocates the bandwidth so that views used as predictors are assigned more bandwidth than the others under equal popularity.

Sequence	Flat Distribution		Gaussian Distribution			Exponential Distribution		
	Proposed	Gain vs Base-A/B	Proposed	Gain versus		Proposed	Gain versus	
				Base-A	Base-B		Base-A	Base-B
Breakdancer	36.54	0.53	37.50	1.49	0.63	37.20	1.19	0.51
Race	33.27	0.45	33.81	1.09	0.40	34.04	1.22	0.98

Table 3: Weighted encoding PSNR for different strategies, $R_C = 1.0$ Mb/s, $R_A = \infty$.

The experiments are then repeated by reducing R_C to 1.0 Mb/s and the results are shown in Table 3. We see that the reduced bit budget produces lower PSNR figures, while the gap between the proposed rate allocation scheme and baseline schemes increases. As the bit budget decreases, the views operate in fact in the steep low-quality area of their QR curves, where even small differences in bit allocation result in big quality changes. In this situation the ignorance of the QR model of the baseline strategy is an even more severe handicap and leads our model-based rate allocation scheme to comparatively better results.

3.4 User-Side Rate Constrained Streaming

We now investigate the case where the capacity R_A of the communication lines between the users and the proxy server shown in the right part of Figure 1 is finite. We present two new different three-stages baseline strategies that we call *Baseline-C* and *Baseline-D*. The strategy *Baseline-C*, is popularity agnostic. During the first stage, it allocates the R_A bit-budget evenly among view i and its predictors for each view i among the N views of the system. For example, in the specific case of the second view in Figure 2, the bit-budget R_A would be equally allocated between the main view and view two. Then, during the second stage, for every view i, the lowest non-zero rate value among the N independently computed values for that view is selected. Step two produces a single solution that jointly satisfies all the constraints on the links between proxy and the users. However, such a solution may not satisfy the R_C constraint on the distribution network capacity, thus a third step is required. As a third and final step, if the total allocated bandwidth exceeds the constraint on R_C, *Baseline-C* computes the number of bits in excess and the rate of each view is reduced by the number of bits in excess over N

We now describe the strategy *Baseline-D*, which is popularity-aware. During the first stage, for each view i among the N views of the system, it allocates the available bit-budget R_A among view i and its predictors proportionally to their popularity. The second stage is identical to the second stage of *Baseline-C*. During the third stage, if the constraint R_C on the distribution network capacity is not satisfied, the exceeding bandwidth is removed. In particular, due to the fact that *Baseline-D* is popularity-aware, the rate of each view is reduced by a number of bits that is inversely proportional to its popularity.

Sequence	Flat Distribution		Gaussian Distribution			Exponential Distribution		
	Proposed	Gain vs Base-C/D	Proposed	Gain versus		Proposed	Gain versus	
				Base-C	Base-D		Base-C	Base-D
Breakdancer	38.12	0.94	38.91	0.81	0.39	38.47	0.35	0.09
Race	35.14	0.53	36.31	1.17	0.14	35.77	0.63	0.29

Table 4: Weighted encoding PSNR for different strategies, $R_C = 1.5$ Mb/s, $R_A = 1.0$ Mb/s.

Table 4 shows how our rate allocation framework performs when R_C is equal to 1.5 Mb/s and R_A is 1.0 Mb/s. A comparison with the results relative to the case where $R_A = \infty$ in Table 2 shows that introducing the constraint on R_A produces a general quality reduction. Such a quality drop is expected, since each additional constraint added to our optimization problem narrows down the search space for the optimal solution. Table 4 shows that

our proposed strategy consistently outperforms the reference schemes in every test scenario thanks to the joint knowledge of the popularity distribution and the video content characteristics.

4. CONCLUSIONS

We have proposed an optimization framework for rate allocation in multi-view that jointly considers the popularity of each view, the prediction dependencies between the views, and their rate-video-quality characteristics. In conjunction with the framework, we have designed simple models characterizing the video quality versus encoding rate trade-offs for both independently encoded and predictively encoded views. Using the models, we effectively solve the optimization problem under consideration using an interior point method in the case of constrained overall data rate for the multi-view content and constrained decoding rate of each view. Our experimental results show that the proposed optimization due to its design provides performance advantages over baseline schemes that do not consider the rate-video-quality characteristics and the view popularity in their allocation. Furthermore, our rate-video-quality models show a considerable degree of accuracy when applied on different multi-view sequences.

REFERENCES

[1] Joint Video Team of MPEG and ITU-T, "Joint draft 8.0 on multiview video coding (JVT-AB204)," Hannover, Germany, 20-25 July, 2008.

[2] Merkle, P., Smolic, A., Müller, K., and Wiegand, T., "Efficient prediction structures for multiview video coding," *IEEE Transactions on circuits and systems for video technology* **17**(11), 1461–1473 (2007).

[3] Boyd, S. and Vandenberghe, L., [*Convex optimization*] (2004).

[4] Chakareski, J. and Frossard, P., "Rate-distortion optimized distributed packet scheduling of multiple video streams over shared communication resources," *IEEE Transactions on Multimedia* **8**(2), 207–218 (2006).

[5] Kim, J., Garcia, J., and Ortega, A., "Dependent bit allocation in multiview video coding," *IEEE International Conference on Image Processing, 2005* **2** (2005).

[6] Tan, A., Aksay, A., Akar, G., and Arikan, E., "Rate-distortion optimization for stereoscopic video streaming with unequal error protection," *EURASIP Journal on Applied Signal Processing* (2009).

[7] Zhuo, L., Gao, X., Wang, Z., Feng, D., and Shen, L., "A Novel Rate-Quality Model based H.264/AVC Frame Layer Rate Control Method," *Proc. IEEE Int'l Conf. Information, Communications, and Signal Processing* (2007).

[8] Ponec, M., Sengupta, S., Chen, M., Li, J., and Chou, P., "Multi-rate peer-to-peer video conferencing: A distributed approach using scalable coding," *IEEE International Conference on Multimedia & Expo* (2009).

[9] Zhang, Y., "Solving large-scale linear programs by interior-point methods under the MATLAB environment," *Optimization Methods and Software* **10**(1), 1–31 (1998).

[10] "Breakdancer sequence," *Available at http://research.microsoft.com/ vision/ InteractiveVisualMediaGroup/ 3DVideoDownload/ .*

[11] "Race sequence," *Available at $ftp://ftp.ne.jp/KDDI/multiview$.*

[12] "H.264/MVC reference software JMVC 5.1.1," *Downloadable from CVS repository with: $cvs-d:pserver: jvtuser@garcon.ient.rwth-aachen.de:/cvs/jvtco-rJMVC\_5\_1\_1jmvc$.*

[13] Leontaris, A. and Tourapis, A., "Rate control for the Joint Scalable Video Model (JSVM)," *Video Team of ISO/IEC MPEG and ITU-T VCEG, JVT-W043, San Jose, California* (2007).

[14] "H.264/SVC reference software JSVM 9.8," *Available at CVS repository with: $cvs-d:pserver: jvtuser@garcon.ient.rwth-aachen.de:/cvs/jvtco-rJSVM\_9\_8jsvm$.*

[15] "H.264/AVC reference software JM 16.0," *Available at http:// iphome.hhi.de/ suehring/ tml/ download/ old_jm/ jm16.0.zip .*

[16] Yan, T., Shen, L., An, P., Wang, H., and Zhang, Z., "Frame-layer rate control algorithm for multi-view video coding," *Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation* , 1025–1028 (2009).