28th Picture Coding Symposium, PCS2010, December 8-10, 2010, Nagoya, Japan

OPTIMAL RATE ALLOCATION FOR VIEW SYNTHESIS ALONG A CONTINUOUS VIEWPOINT LOCATION IN MULTIVIEW IMAGING

Vladan Velisavljević[1], *Gene Cheung*[2], *Jacob Chakareski*[3]

[1]Deutsche Telekom Laboratories, Berlin, Germany,
[2]National Institute of Informatics, Tokyo, Japan,
[3]Ecole Polytechnique Fédérale de Lausanne, Switzerland

ABSTRACT

We consider the scenario of view synthesis via depth-image based rendering in multi-view imaging. We formulate a resource allocation problem of jointly assigning an optimal number of bits to compressed texture and depth images such that the maximum distortion of a synthesized view over a continuum of viewpoints between two encoded reference views is minimized, for a given bit budget. We construct simple yet accurate image models that characterize the pixel values at similar depths as first-order Gaussian auto-regressive processes. Based on our models, we derive an optimization procedure that numerically solves the formulated min-max problem using Lagrange relaxation. Through simulations we show that, for two captured views scenario, our optimization provides a significant gain (up to 2dB) in quality of the synthesized views for the same overall bit rate over a heuristic quantization that selects only two quantizers - one for the encoded texture images and the other for the depth images.

Index Terms— Multi-view imaging, Rate allocation

1. INTRODUCTION

Multi-view systems capture images of the same scene using time-synchronized and spatially correlated cameras located at a number of different viewpoints. For depth-image-based rendering (DIBR) [1], the captured texture images are encoded and transmitted to a decoder together with depth information estimated or measured at the same or different viewpoints. The reconstructed texture and depth images allow the decoder to synthesize views at intermediate viewpoints using techniques like 3D warping [2]. As the fidelity of the synthesized views depends on the quality of both texture and depth images, it is important to jointly optimize their compression performance for a given available bit-rate.

Compression tools for multiview texture and depth images have been studied recently in [3, 4] for the goal of achieving higher coding gain by exploiting the inter-view and spatial correlations inherent in the images. However, the authors do not formally optimize the bit allocation among the compressed texture and depth images within a rate-distortion formulation that would account for the reconstruction error of both encoded views and synthesized views. Part of the difficulty lies in the fact that unlike encoded views, viewpoints in which synthesized views will be reconstructed at the decoder are not known at encoding time. In our previous work [5], we presented an operational bit allocation strategy for multiview texture and depth map compression that employs, as an evaluation metric, a discrete set of synthesized viewpoints known at encoding time. In the present paper, we generalize this metric to the one where *the entire continuum of feasible synthesized viewpoints* is considered during bit allocation.

In particular, we solve a generalized RD-optimized bit allocation for multiview image compression assuming that: (1) a differential coder with disparity compensation based on depth layer segmentation (see Section 2.1) for multiview texture and depth images is used; (2) a DIBR method for view synthesis from neighboring encoded texture and depth images is employed at the decoder; and (3) a view can be synthesized at *any* viewpoint between two encoded (reference) views. Note that, even though the assumption (1) excludes some types of coders, there is still a variety of practical coders that are analyzed in our work (e.g., the coders used in [4, 6, 7]).

The proposed optimization method optimally distributes bits between the encoded texture and depth images given an overall bit budget so that the synthesized views in a known interval of viewpoint locations are guaranteed not to exceed a bounded fidelity measure (e.g., bounded distortion in terms of mean-square error). We show in Section 4 that, for a two cameras (encoded viewpoints) scenario, the optimal allocation results into an improved quality of the synthesized views (even up to 2dB) relative to a heuristic approach that selects only two quantization values, one for both texture images and the other for both depth images at the reference (encoded) views. However, our analysis can be easily generalized to a larger set of cameras and also to multiview video sequences.

To the best of our knowledge, there are no prior works that directly addressed the bit allocation problem for DIBR for a continuum of synthesized viewpoints between two reference views. For instance, the authors in [8] constructed a detailed distortion model for synthetic views but optimized bit allocation only for a single discrete viewpoint. Similarly, the work in [9] modeled synthesized view distortion as a simple exponential function of texture and depth coding rates, oversimplifying the fact that distortion values at different synthesized viewpoints are inherently different. As multi-view systems are increasingly becoming popular, the substantial gains provided by our optimization can lead to more efficient operation in a number of applications, e.g., immersive telepresence and telecollaboration, remote monitoring, free-view point and 3D TV, virtual worlds and online gaming, etc.

The rest of the paper is organized as follows. The formulation of the problem of joint compression of texture and depth images is given in Section 2, including the proposed bit rate and distortion modeling that are based on a set of parameters measured from the captured signals. Then, in Section 3 we describe the optimization procedure that we derive to solve the bit allocation problem under consideration. Next, in Section 4 we evaluate the compression performance of the proposed optimization. Finally, concluding remarks

The work of J. Chakareski was supported by the Swiss National Science Foundation under Ambizione grant PZ00P2-126416.

978-1-4244-7133-1/10/$26.00 ©2010 IEEE

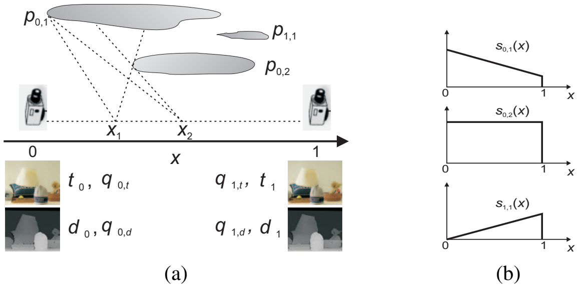


Fig. 1. (a) The coding problem setup. The shown examples of the captured texture and depth images are taken from the data set `Midd2` [10]. (b) The sizes of the visible portions of 3 patches are approximated by linear functions of the synthesis viewpoint location x.

are provided in Section 5.

2. PROBLEM FORMULATION

We first explain in words the bit allocation problem under consideration. Quantization levels of the texture and depth images at two viewpoints, $n = 0, 1$, need to be selected such that, for a given bitrate budget, a chosen metric of fidelity of a texture image rendered at the decoder along the viewpoint trajectory is optimized. The specific optimization criterion that we selected is min-max, i.e., we are interested in minimizing the maximum distortion achieved by a view synthesized anywhere between reference views 0 and 1. It should be mentioned that there are other optimization criteria that can be employed in this context, e.g., minimizing the average distortion of synthesized views along the view trajectory. However, our specific choice of min-max allows us to provide a minimum quality guarantee for view synthesis anywhere in the viewpoint interval. The setting of the problem that we study is shown in Fig. 1(a).

In more detail, texture and depth images t_0 and d_0 at viewpoint 0 are encoded using a selected image codec with quantizers $q_{0,t}$ and $q_{0,d}$, respectively. Note that the only imposed constraint in the codec is an independent choice of quantizers $q_{0,t}$ and $q_{0,d}$, whereas the encoding of texture and depth images can be done either jointly or separately. The images t_1 and d_1 at viewpoint 1 are encoded using inter-view disparity compensated differential coding [11], so that only the areas that are occluded at viewpoint 0 but visible at viewpoint 1 are encoded with the quantizers $q_{1,t}$ and $q_{1,d}$, respectively. Similarly to $q_{0,t}$ and $q_{0,d}$, the quantizers $q_{1,t}$ and $q_{1,d}$ are chosen independently.

The texture images $t(x)$'s at location $x \in [0, 1]$ are synthesized using the decoded texture and depth images at viewpoints 0 and 1. Note that the synthesis viewpoint location is continuous and unknown at encoding time (only the interval is known), unlike our previous work [5], where we considered a discrete set of location points that were specified ahead of time.

2.1. Partition into patches

To analyze the rate-distortion performance of the encoded texture and depth images, we partition t_0 and d_0 jointly into P_0 patches of pixels with approximately stationary content. The partitioning can be implemented in several ways, but, for simplicity, we define a patch as a group of pixels with depths between two pre-selected bounding values (such patches are also called *layers* in [6]). We assume further that the shape of each patch is common to both texture and depth images, such that a partitioning means shapes only need to be encoded once for both texture and depth maps, as discussed in [4]. Patches allow for an efficient modeling of the pixel values using stationary processes, to be discusssed later.

Note that the bitrate required for shape encoding is typically small (on the order of 10^{-2}bpp). In the present paper, we assume shape coding has already been performed, and we focus on the orthogonal problem of optimal bit allocation among texture and depth maps using the remaining bits.

Each patch i of view 0 is denoted as $p_{0,i}$, where $i = 1, \ldots, P_0$, and is characterized by the variance of the texture and depth pixels, $\sigma^2_{0,t,i}$ and $\sigma^2_{0,d,i}$, respectively, measured directly on the captured images. The size of each patch $p_{0,i}$ is measured and expressed in terms of the number of pixels as $s_{0,i}(x = 0)$ at viewpoint $x = 0$, and as $s_{0,i}(x = 1)$ at viewpoint $x = 1$. Note that, due to possible occlusions, $s_{0,i}(1) \le s_{0,i}(0)$.

Similarly, the differential images t_1 and d_1 at view 1 are partitioned into P_1 patches $p_{1,i}$, $i = 1, \ldots, P_1$, each characterized by the variances $\sigma^2_{1,t,i}$ and $\sigma^2_{1,d,i}$, respectively. The corresponding size of the patches is expressed as $s_{1,i}(x = 1)$ at viewpoint 1, whereas, $s_{1,i}(x = 0) = 0$ since the patches $p_{1,i}$ are invisible at viewpoint 0 by the definition of the differential encoding.

The size of the non-occluded portions of each patch visible from the synthesis viewpoint location $0 \le x \le 1$ depends on the shape, depth and relative position of the patches. For simplicity, this size is modeled here by a linear relation as a first-order approximation:

$$\begin{aligned} s_{0,i}(x) &= [s_{0,i}(1) - s_{0,i}(0)] \cdot x + s_{0,i}(0) \\ s_{1,i}(x) &= [s_{1,i}(1) - s_{1,i}(0)] \cdot x + s_{1,i}(0) = s_{1,i}(1) \cdot x. \end{aligned} \tag{1}$$

Here $S_0(0) = \sum_{i=1}^{P_0} s_{0,i}(0)$ is equal to the total number of pixels captured at viewpoint 0 (camera resolution), whereas $S_1(1) = \sum_{i=1}^{P_1} s_{1,i}(1)$ is the number of pixels that are additionally captured at viewpoint 1 and invisible at 0. Even though such a linear relation does not generally capture the complex three-dimensional geometry of the scene, it models well how the size of the patches in the background changes with the viewpoint location x due to occlusion, as illustrated in Fig. 1 (b). It should be mentioned that the model can be further improved by considering piecewise linear functions instead, however at the price of higher complexity for the related analysis and optimization.

2.2. Encoding rate

Since each patch contains pixels located at approximately the same depth, thereby avoiding edges, these pixels in both the texture and depth images can be efficiently modeled as stationary first-order Gaussian auto-regressive processes. Specifically, given the quantizer q and the variance σ^2, the Kolmogorov encoding bitrate-per-pixel for such processes is given by [12]:

$$R(q) = \max\left\{0, \frac{1}{2} \cdot \ln\left(\frac{\sigma^2}{q^2}\right)\right\}. \tag{2}$$

Relaxing (2) for practical coders, the encoding bitrate-per-pixel for each patch $p_{n,i}$, where $n = 0, 1$ and $i = 1, \ldots, P_n$, is given by:

$$R_{n,k,i}(q_{n,k}) = \max\left\{0, \alpha_k \cdot \ln\left(\frac{\sigma^2_{n,k,i}}{q^2_{n,k}}\right)\right\}, \tag{3}$$

Here, $k \in \{t, d\}$ stands for either texture or depth image, whereas parameters α_t and α_d depend on the chosen image coder.

Given the set of quantizers $\mathcal{Q} = \{q_{0,t}, q_{0,d}, q_{1,t}, q_{1,d}\}$, the total bitrate $R(\mathcal{Q})$ is given by:

$$\begin{aligned} R(\mathcal{Q}) = &\sum_{i=1}^{P_0} s_{0,i}(0) \cdot [R_{0,t,i}(q_{0,t}) + R_{0,d,i}(q_{0,d})] + \\ &\sum_{i=1}^{P_1} s_{1,i}(1) \cdot [R_{1,t,i}(q_{1,t}) + R_{1,d,i}(q_{1,d})]. \end{aligned}$$

Using (3) and assuming $q_{n,k} < \sigma_{n,k,i}$, $R(\mathcal{Q})$ becomes:

$$R(\mathcal{Q}) = S_0(0) \cdot \left[\alpha_t \ln\left(\frac{\sigma_{0,t}^2}{q_{0,t}^2}\right) + \alpha_d \ln\left(\frac{\sigma_{0,d}^2}{q_{0,d}^2}\right)\right] + S_1(1) \cdot \left[\alpha_t \ln\left(\frac{\sigma_{1,t}^2}{q_{1,t}^2}\right) + \alpha_d \ln\left(\frac{\sigma_{1,d}^2}{q_{1,d}^2}\right)\right], \quad (4)$$

where $S_n(x) = \sum_{i=1}^{P_n} s_{n,i}(x)$ and the equivalent variances equal the weighted geometric mean $\sigma_{n,k}^2 = \prod_{i=1}^{P_n} (\sigma_{n,k,i}^2)^{s_{n,i}(n)/S_n(n)}$.

As explained earlier, the bitrate required for encoding the shape of the patches $p_{n,i}$ is neglected in our analysis since in typical coders this quantity is much smaller than the bitrate expressed by (4). Furthermore, it should be pointed out, that although the sum $S_0(x) + S_1(x)$ should always equal the image resolution of a view, due to the approximative linear modeling in (1), one can construct an unlikely scenario where the sum will be strictly smaller than the overall number of pixels in a view.

2.3. View synthesis distortion

The synthesized texture images $t(x)$, $0 < x < 1$, consist of the unoccluded parts of the patches $p_{0,i}$ and $p_{1,i}$ decoded from the quantized versions of the texture images t_0 and t_1 and shifted by the disparity computed from the quantized versions of the depth images d_0 and d_1. The distortion of the synthesized views $t(x)$, expressed as mean-square error, is caused by both texture and depth image quantization. We analyze the impact of these two factors next.

Assume pixel $y_{n,i}$ belongs to the patch $p_{n,i}$, $n = 0, 1$. The quantized version of $y_{n,i}$ using the quantizer $q_{n,t}$ is denoted as $\hat{y}_{n,i}$, whereas the corresponding rendered pixel at the synthesis viewpoint is denoted as $\bar{y}_{n,i}$. Note that $\bar{y}_{n,i}$ is not necessarily equal to $\hat{y}_{n,i}$ due to a possible misplacement of the rendered pixels caused by the depth image quantization error, as also explained in [13].

The expected distortion of pixel $y_{n,i}$ is given by:

$$\begin{aligned} D_{n,i}(q_{n,t}, q_{n,d}) &= E[(y_{n,i} - \bar{y}_{n,i})^2] \\ &= E[y_{n,i}^2] + E[\bar{y}_{n,i}^2] - 2E[y_{n,i} \cdot \bar{y}_{n,i}]. \end{aligned} \quad (5)$$

To compute each of the 3 terms in (5), first, assume the pixel values are zero-mean. Thus, we have:

$$E[y_{n,i}^2] = \sigma_{n,t,i}^2. \quad (6)$$

Then, since $\bar{y}_{n,i}$'s are obtained by resampling pixels $\hat{y}_{n,i}$'s, the corresponding variances remain the same, that is, $E[\bar{y}_{n,i}^2] = E[\hat{y}_{n,i}^2]$. The quantized pixels are expressed as $\hat{y}_{n,i} = y_{n,i} + \epsilon_q$, where the quantization error ϵ_q is assumed to be zero-mean and independent from $\hat{y}_{n,i}$. Following the results in [12] and related to (3), the variance of the quantization error is modeled by:

$$D(q) = E[\epsilon_q^2] = \beta_t \cdot \min\left\{q^2, \sigma^2\right\}, \quad (7)$$

where parameter β_t depends on the chosen image coder. Assuming $q_{n,t} < \sigma_{n,t,i}$, it follows that:

$$E[\bar{y}_{n,i}^2] = E[\hat{y}_{n,i}^2] = \sigma_{n,t,i}^2 - \beta_t q_{n,t}^2. \quad (8)$$

Finally, $E[y_{n,i} \cdot \bar{y}_{n,i}] = E[(\hat{y}_{n,i} - \epsilon_q) \cdot \bar{y}_{n,i}] = E[\hat{y}_{n,i} \cdot \bar{y}_{n,i}]$. This term can be expressed using the correlation $\rho_{\hat{y},\bar{y}}$ between $\hat{y}_{n,i}$ and $\bar{y}_{n,i}$, as $\rho_{\hat{y},\bar{y}} \cdot (E[\hat{y}_{n,i}^2] \cdot E[\bar{y}_{n,i}^2])^{1/2}$. Using (8), it follows that $E[\hat{y}_{n,i} \cdot \bar{y}_{n,i}] = \rho_{\hat{y},\bar{y}} \cdot (\sigma_{n,t,i}^2 - \beta_t q_{n,t}^2)$. Note that whenever convenient we omit the index (n, i) of the subscript variables of the correlation, for simplicity of notation.

Now, similarly to [13], the correlation $\rho_{\hat{y},\bar{y}}$ of the Gaussian autoregressive process is approximated by $\rho_{\hat{y},\bar{y}} = \rho_0^{\sqrt{\beta_d} \cdot q_{n,d} \cdot \delta_n}$, where ρ_0 is the correlation between neighboring pixels measured directly in the captured patches and $\sqrt{\beta_d} \cdot q_{n,d} \cdot \delta_n$ is an average shift due to the quantization error associated with $q_{n,d}$. Here, β_d depends on the depth image coder and the distances to the reference views: $\delta_0 = x$ and $\delta_1 = 1 - x$. Thus, we have:

$$E[y_{n,i} \cdot \bar{y}_{n,i}] = (\sigma_{n,t,i}^2 - \beta_t q_{n,t}^2) \cdot \rho_0^{\sqrt{\beta_d} \cdot q_{n,d} \cdot \delta_n}. \quad (9)$$

Using (6), (8), and (9), the distortion $D_{n,i}(q_{n,t}, q_{n,d}, x)$ becomes

$$\begin{aligned} D_{0,i} &= 2(\sigma_{0,t,i}^2 - \beta_t q_{0,t}^2)(1 - \rho_0^{\sqrt{\beta_d} \cdot q_{0,d} \cdot x}) + \beta_t q_{0,t}^2, \\ D_{1,i} &= 2(\sigma_{1,t,i}^2 - \beta_t q_{1,t}^2)(1 - \rho_0^{\sqrt{\beta_d} \cdot q_{1,d} \cdot (1-x)}) + \beta_t q_{1,t}^2. \end{aligned} \quad (10)$$

Hence, the total view synthesis distortion at viewpoint x, $0 \leq x \leq 1$, is given as a linear combination of the distortions $D_{0,i}(q_{0,t}, q_{0,d}, x)$ and $D_{1,i}(q_{1,t}, q_{1,d}, x)$ with the sizes of the unoccluded portions of the patches as weight factors, that is,

$$D(\mathcal{Q}, x) = \sum_{i=1}^{P_0} s_{0,i}(x) D_{0,i} + \sum_{i=1}^{P_1} s_{1,i}(x) D_{1,i}. \quad (11)$$

It should be mentioned that the scene can contain areas occluded in both reference views but visible in a synthesized view along the viewpoint trajectory. However, such areas are irrelevant for the analysis presented here because, in such a case, an encoder cannot control the associated distortion by cleverly reallocating bits between texture and depth images of views 0 and 1.

3. RATE-DISTORTION OPTIMIZATION

The goal of the optimization is to search for the quantization levels $\mathcal{Q}^*$ that result into the smallest upper bound of the distortion $D(\mathcal{Q}, x)$ in (11), for $0 \leq x \leq 1$, given that the total bitrate $R(\mathcal{Q})$ expressed in (4) does not exceed a bitrate constraint R_{max}. In mathematical terms, we write:

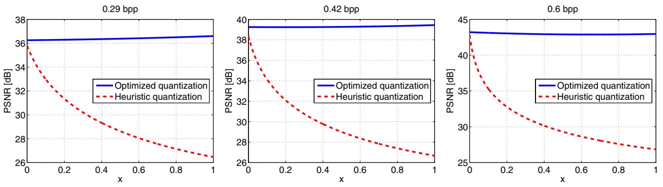
$$\mathcal{Q}^* = \arg\min_{\mathcal{Q}} \left[\max_{0 \leq x \leq 1} D(\mathcal{Q}, x)\right], \text{ s.t. } R(\mathcal{Q}) \leq R_{max}. \quad (12)$$

Below, we briefly describe our approach for computing the solution of (12). First, we unravel the inner maximization sub-problem in (12). In particular, we take the partial derivative $D'(\mathcal{Q}, x)$ of $D(\mathcal{Q}, x)$ with respect to x and find its root numerically, i.e., the value x^* at which $D'(\mathcal{Q}, x^*) = 0$. This is done using the `fzero` function in `MATLAB`. One can show that the second derivative $D''(\mathcal{Q}, x)$ is strictly negative in the range $0 \leq x \leq 1$ for realistic values of our model parameters, which means that the root x^* of $D'(\mathcal{Q}, x)$, if $0 \leq x^* \leq 1$, must be a unique local maximum in the range. We can then conclude that the solution $\arg\max_{0 \leq x \leq 1} D(\mathcal{Q}, x)$ is either equal to x^*, if $0 \leq x^* \leq 1$, or otherwise is in the binary set $\{0, 1\}$. In other words, we need to check at most two x values to determine maximum $D(\mathcal{Q}, x)$. Let $f(\mathcal{Q})$ be the resulting maximum value.

Given that $f(\mathcal{Q})$ can now be solved numerically in a straightforward manner, we construct the Lagrangian unconstrained version of (12) for a given Lagrange multiplier $\lambda > 0$ as follows:

$$\mathcal{Q}^o = \arg\min_{\mathcal{Q}} f(\mathcal{Q}) + \lambda D(\mathcal{Q}) \quad (13)$$

It can be easily shown that an optimal solution $\mathcal{Q}^o$ to (13) is also an optimal solution to an instant of (12) where $R_{max} = R(\mathcal{Q}^o)$.



(a) (b) (c)

Fig. 2. Distortion $D(\mathcal{Q}, x)$ for $x \in [0, 1]$ using the heuristic quantizer Q_h (red) and optimal quantizer Q^* (blue). The graphs are shown for the total bit rates (a) 0.29 bpp , (b) 0.42 bpp and (c) 0.60 bpp. The maximal distortion is significantly reduced in all cases when the optimized quantization is applied.

To solve (13), we start from an initial guess of $\mathcal{Q}^i$ and then perform simple local searches that iteratively refine $\mathcal{Q}^i$ in small increments, while decreasing the objective function in (13), until convergence.

4. RESULTS

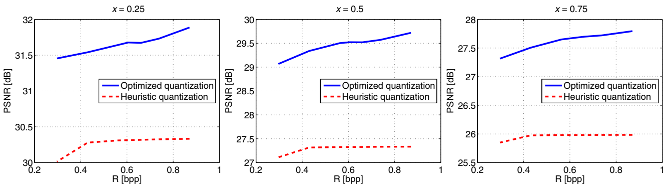
To test our optimal bit allocation, we jointly encode texture and depth images using the shape-adaptive wavelet-based coder from [4]. We use the data set `Midd2` from [10][1] that contains seven 1110×1366 texture images at the viewpoints enumerated $0, \ldots, 6$ and two depth images at the viewpoints 1 and 5. The pair t_0 and d_0 is associated to the texture and depth images at the viewpoint 1, whereas t_1 and d_1 represent the images at the viewpoint 5. The other available texture images captured at the intermediate viewpoints 2, 3 and 4 are used for evaluating the fidelity of our synthesized views at $x = 0.25$, 0.5 and 0.75, respectively.

The simple view synthesis method used here resamples the pixels from either t_0 or t_1 and shifts them according to the associated depth information in d_0 or d_1. In case of overlapping of several pixels at the same coordinate of the synthesized view, the nearest pixel with the smallest depth is retained.

It should be mentioned that due to a restricted availability of the data sets and viewpoint locations of the captured views, we cannot evaluate the view synthesis quality using our method at the entire continuum of the viewpoint locations. Instead, we provide the comparison only at the three available intermediate viewpoints. However, since the captured viewpoint locations are close enough and the scene is smooth, the generality of our comparison is not affected by this limitation of the data.

First, in Fig. 2, we compare the distortion $D(\mathcal{Q}, x)$ from (11) for $x \in [0, 1]$ and for different bit rates in two cases: (1) with the optimal quantizer Q^*, as computed in Section 3, and (2) with a heuristic quantization Q_h that consists of only two quantizers, one for t_0 and t_1 and the other for d_0 and d_1. In both cases, the total bit rate and the ratio between the bit rates allocated to the texture and depth maps are kept constant. Note that the maximal distortion is significantly reduced (for even more than 10dB at higher bit rates) using the optimal quantizer Q^*.

Further, in Fig. 3, we show the actual RD performance of view synthesis at the three viewpoints $x = \{0.25, 0.5, 0.75\}$ using the encoded images with the optimal and heuristic quantizers. The quality of the synthesized views using the optimized quantizers is constantly better (1-2dB) than that for the heuristic quantizers over a wide range of bit rates. Note that the resulting RD performance in this case is not equal to the expected one shown in Fig. 2. This difference is mainly due to the simplified sub-optimal view synthesis method that we implemented and the lack of exact depth information.



(a) (b) (c)

Fig. 3. The RD performance of view synthesis at 3 viewpoints (a) $x = 0.25$, (b) $x = 0.5$ and (c) $x = 0.75$. The optimized quantization results in an improved quality of the synthesized views than the heuristic quantization.

5. CONCLUSIONS

We considered RD optimal bit allocation for coding of multiview images and depth-image based view synthesis. Our optimization addresses the joint quantization assignment to encoded texture and depth images captured at a set of discrete viewpoints so that the maximal distortion of the synthesized views anywhere in between the captured viewpoints is minimized for a given bit budget. The optimization leverages models that characterize the texture and depth image pixels as first-order Gaussian auto-regressive processes. We compare the optimized quantization to a conventional heuristic solution, where only two quantizers are applied, one for texture and the other for depth maps, and we show a significant gain in the quality of the synthesized views.

Acknowledgement

We want to thank Matthieu Maitre and Minh N. Do for providing us with their texture/depth image compression software used in [4].

6. REFERENCES

[1] P. Merkle, A. Smolic, K. Mueller, and T. Wiegand, “Multi-view video plus depth representation and coding,” in *IEEE ICIP*, 2007.

[2] Y. Morvan, D. Farin, and P. H. N. de With, “Multiview depth-image compression using an extended H.264 encoder,” *Advanced Concepts for Intelligent Vision Systems*, vol. 4678, pp. 675–686, 2007.

[3] Y. Morvan, D. Farin, and P. H. N. de With, “Depth-image compression based on an R-D optimized quadtree decomposition for the transmission of multiview images,” in *IEEE ICIP*, 2007.

[4] M. Maitre and M. N. Do, “Joint encoding of the depth image based representation using shape-adaptive wavelets,” in *IEEE ICIP*, 2008.

[5] G. Cheung and V. Velisavljević, “Efficient bit allocation for multiview image coding & view synthesis,” in *IEEE ICIP*, 2010.

[6] Gelman A., Dragotti P. L., and Velisavljević V., “Multiview image compression using a layer-based representation,” in *IEEE ICIP*, 2010.

[7] Sánchez A., Shen G., and Ortega A., “Edge-preserving depth-map coding using graph-based wavelets,” in *Proc. of the Asilomar*, 2009.

[8] Y. Liu, Q. Huang, S. Ma, D. Zhao, and W. Gao, “Joint video/depth rate allocation for 3D video coding based on view synthesis distortion model,” in *Elsevier, Signal Processing: Image Communication*, September 2009, vol. 24, no.8, pp. 666–681.

[9] I. Daribo, C. Tillier, and B. Pesquet-Popescu, “Motion vector sharing and bitrate allocation for 3D video-plus-depth coding,” in *EURASIP Journal on Advances in Signal Processing*, January 2009, vol. 2009.

[10] “Middlebury 2006 stereo datasets,” http://vision.middlebury.edu/stereo/data/scenes2006/.

[11] J.H. Kim, P.L. Lai, J. Lopez, A. Ortega, Y. Su, P. Yin, and C. Gomila, “New coding tools for illumination and focus mismatch compensation in multiview video coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1519–1535, 2007.

[12] Gray R. M. and Hashimoto T., “Rate-distortion functions for nonstationary Gaussian autoregressive processes,” in *IEEE Data Compression Conf.*, Snowbird, UT, March 2008.

[13] Kim W.-S., Ortega A., Lai P., Tian D., and Gomila C., “Depth map coding with distortion estimation of rendered view,” in *Proc. of the SPIE Visual Information Proc. and Communication*, January 2010.

[1] Note that the Middlebury data sets [10] have been generated mainly for stereo matching processing. However, since we focus here on multiview *still* imaging, these data sets perfectly fit our experimentation.