MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Synthesis Distortion Estimation in 3D Video Using Frequency and Spatial Analysis

Fang, L.; Cheung, N-M; Tian, D.; Vetro, A.; Sun, H.; Yu, L.

TR2013-087 September 2013

Abstract

We propose an analytical model to estimate the synthesized view quality in 3D video. Specifically, we estimate the depth-error induced distortion using an approach that combines frequency and spatial domain analysis. We also propose to decompose the spatial-variant video signals into gradient-based representations to capture the interaction between image gradients, depth errors and synthesis distortion. Experiment results with video sequences and coding/rendering tools used in MPEG 3DV activities show that our analytical model can accurately estimate the synthesis noise power.

IEEE International Conference on Image Processing (ICIP)

SYNTHESIS DISTORTION ESTIMATION IN 3D VIDEO USING FREQUENCY AND SPATIAL ANALYSIS

Lu Fang *Ngai-Man Cheung* ⋆ *Dong Tian*† *Anthony Vetro*† *Huifang Sun*† *Lu Yu*‡

Univ. of Sci. and Tech. of China ⋆ Singapore Univ. of Tech. and Design † Mitsubishi Electric Research Labs ‡ Zhejiang Univ.

ABSTRACT

We propose an analytical model to estimate the synthesized view quality in 3D video. Specifically, we estimate the depth-error induced distortion using an approach that combines frequency and spatial domain analysis. We also propose to decompose the spatial-variant video signals into gradient-based representations to capture the interaction between image gradients, depth errors and synthesis distortion. Experiment results with video sequences and coding/rendering tools used in MPEG 3DV activities show that our analytical model can accurately estimate the synthesis noise power.

Index Terms— 3D Video, Synthesis Distortion Estimation, Frequency and Spatial Analysis, Depth Error

I. INTRODUCTION

3D video (3DV) has attracted much attention recently [1]–[5]. 3D datasets usually consist of multiple video sequences (texture data) captured by cameras at different positions, along with the associated depth images. The quality of the synthesized view is imperative in 3DV applications [6], [7]. The synthesis quality, however, depends on several factors and complicated interactions between them. In particular, texture and depth images may contain errors due to imperfect sensing or lossy compression [8], [9], and it is not clear how these errors interact and affect the rendering quality. Unlike texture errors, which cause distortion in the luminance/chrominance level, depth errors cause *position errors* in synthesis [10], i.e., pixels are warped to slightly shifted positions during synthesis. The effect of depth errors is very subtle. For instance, the impact of depth errors would vary with the image contents, and images with less textures tend to be more resilient to the depth errors.

An accurate analytical model to estimate the synthesis quality is very valuable for the design of 3DV systems. Nguyen and Do [11] analyzed the rendering quality of image-based rendering (IBR) algorithms and used Taylor series expansion to derive the upper bound of the mean absolute error (MAE) in the synthesis output. Liu et al. [12] approximated errors due to depth map artifacts using a linear model of average magnitude of mean-squared disparity errors over an entire frame and a motion sensitivity factor computed from the energy density. An autoregressive model was proposed by Kim et al. [13] to estimate the synthesis distortion at the block level and was shown to be effective for rate-distortion optimized mode selection. A distortion model as a function of the view location was also proposed by Velisavljevic et al. [14] for bit allocation. Takahashi [15] proposed an optimized view interpolation scheme based on frequency domain analysis of depth map error.

In our previous work [16], we proposed to estimate the synthesis quality using power spectral density (PSD). This assumed that the underlying video frame signals are spatial invariant. However, signals along strong texture edges change much quickly than those in the non-edge regions, i.e., autocovariance function decays much faster in edge pixels. Different from previous approaches, we propose in this work a model that combines frequency analysis and spatial analysis to account for the non-stationary in the signals. Specifically, our contributions are: 1) We propose an analytical model combining frequency and spatial domain analysis to estimate the synthesis view quality, given depth map errors, texture image characteristics (smooth or textural), texture image quality and the camera configuration as the inputs; 2) In particular, we propose to decompose the *spatial variant* signals into gradient-based representations to facilitate analysis. The analysis results show that linear approximations of the spatial variant signals can lead to a computationally-efficient and yet accurate estimation; 3) We verify our model with substantial experiments using video sequences and coding and rendering tools from the MPEG 3DV activities [17], [18]. The rest of the paper is as follows. Sections II discusses our analytical model. Section III presents experiment results and Section IV concludes the paper.

II. ESTIMATE NOISE POWER DUE TO DEPTH CODING

We first discuss our view synthesis model, which consists of frame warping followed by blending. Two reference texture frames captured by the left and right cameras (denoted by $X_l(m,n)$ and $X_r(m,n)$ respectively) along with their associated depth images (denoted by $D_l(m,n)$ and $D_r(m,n)$ respectively) are used to generate the synthesized frame $U(m,n)$ at a certain virtual camera position. First, in frame warping, pixels are copied from X_l to form an intermediate frame U_l, from position (m',n) to (m,n). We assume the cameras are rectified and arranged linearly, and there exists only horizontal disparity given by $m-m'$ determined by the depth images, camera parameters and camera distance [16]. Likewise, pixels are copied from X_r to form the intermediate frame U_r. Then, U_l and U_r are merged (blended) to generate U. We assume merging by linear combination: $U(m,n) = \alpha U_l(m,n) + (1-\alpha)U_r(m,n)$. Here the weight α is determined by the distances between the virtual camera position and the left/right reference camera positions.

In practice, the texture and depth images are lossy encoded. We assume that when the reconstructed texture/depth images $(\hat{X}_l, \hat{X}_r, \hat{D}_l, \hat{D}_r)$ are fed into the synthesis pipeline, we obtain rendering output W. Let $V = U - W$ be the noise in the rendering output due to coding errors in texture/depth images. In [16], we show that under reasonable assumptions the total synthesis noise power ($E[V^2]$) can be estimated by summing two components: one is the synthesis noise power due to *texture image coding* ($E[N^2]$), the other is the synthesis noise power due to *depth image coding* ($E[Z^2]$), i.e.,

$$E[V^2] = E[N^2] + E[Z^2]. \quad (1)$$

We discussed the estimation of $E[N^2]$ in [16] and the focus in this paper is on the estimation of $E[Z^2]$.

In [16], we showed that $E[Z^2]$ can be estimated from $E[Z_l^2]$ and $E[Z_r^2]$, where Z_l and Z_r are the synthesis noise due to depth map coding in the left/right cameras respectively. Previously, we used power spectral density to estimate $E[Z_l^2]$ and $E[Z_r^2]$. This frequency domain analysis assumed that the underlying image signals are spatial invariant (i.e., wide-sense stationary), which we found that in the current application this would cause rather significant estimation discrepancy (In [16], we used a sequence specific constant to compensate this discrepancy). Specifically, across strong texture edges the video contents change much more quickly than the non-edge regions, which does not agree with the spatial invariant assumption. Edge pixels exhibit significantly different correlation statistics compared with those in the non-edge regions (autocovariance function decreases significantly faster in edge pixels). We found that models that fail to account for these non-stationary characteristics would incur considerable estimation discrepancy in rendering quality estimation. In particular, at regions where the video contents change rapidly (strong texture edges as shown in the white part of Fig. 1(b)), pixel shifts would result in substantial rendering errors, and these errors would bias the overall estimate and are not negligible (even though edge regions are only small portions in the video frames).

Consequently, in this work, we propose to partition the video frame signals into *Spatial Invariant (SI)* signals and *Spatial Variant (SV)* signals, and analyze these signals with frequency and spatial techniques respectively. Specifically, we start by analyzing the gradient map of texture image, and partition the video frame into SI and SV regions using a gradient threshold determined automatically using Otsu's algorithm [19], as shown in Fig. 1. We estimate $E[Z_l^2]$ (and likewise $E[Z_r^2]$) for the pixels belonging to SI and SV regions, which we denote as $E[Z_{l,\text{SI}}^2]$ and $E[Z_{l,\text{SV}}^2]$ respectively. Then $E[Z_l^2] = E[Z_{l,\text{SI}}^2] + E[Z_{l,\text{SV}}^2]$. Frequency domain analysis similar to [16] is used for $E[Z_{l,\text{SI}}^2]$. In the following, we will discuss the estimation of $E[Z_{l,\text{SV}}^2]$.

(a) (b)

Fig. 1. (a) Texture image of Kendo sequence (view 3); (b) thresholding result using Otsu's algorithm (black: spatial-invariant regions, white: spatial-variant regions).

Gradient-based Analysis of SV Regions. To estimate the distortion due to depth errors in the spatial-variant (SV) regions $E[Z_{l,\text{SV}}^2]$, we process the frame row-by-row (likewise for $E[Z_{r,\text{SV}}^2]$). For each row, we process one by one each SV region (a SV region consists of consecutive pixels classified as SV). Denote Y_l the frame warping result using $\hat{X}_l$ and D_l, and W_l the frame warping result using $\hat{X}_l$ and $\hat{D}_l$. Let us denote a vector $\vec{S_L}$ as the pixel values of a SV region of extent (width) L in Y_l, and $\vec{S'_L}$ as the one in W_l. Note that W_l ($\vec{S'_L}$) is different from Y_l ($\vec{S_L}$) solely due to the fact that *reconstructed* depth is used in frame warping instead of the original depth. Note also that we consider $\hat{X}_l$ here instead of X_l as we decompose the overall distortion into texture-coding-induced distortion and depth-coding-induced distortion as suggested by (1) and here we focus on depth-coding-induced distortion.

Recall that due to the depth coding artifacts, there exists depth error for depth map, resulting in horizontal disparity error during texture image warping. Specifically, in SV regions, the sharp edge would magnify the effect of horizontal disparity error on the rendering distortion between $\vec{S_L}$ and $\vec{S'_L}$. To model the effect of both gradient value and depth error (horizontal disparity error) on the rendering result, we decompose $\vec{S_L}$ into L *gradient-based component-vectors*, such that

$$\vec{S_L} = \sum_{k=1}^{L} \vec{s_k}, \tag{2}$$

where $k = 1, 2, \cdots, L$ and $\vec{s_k}$ is the k^{th} gradient-based component-vector, given by

$$\vec{s_k} = g_k \vec{1}_k, \tag{3}$$

where g_k is the gradient value at the k^{th} spatial location in $\vec{S_L}$, and $\vec{1}_k$ is a vector with $k-1$ zeros followed by $L-k+1$ ones, i.e., $\vec{1}_k = [\underbrace{0, \cdots, 0}_{k-1}, \underbrace{1, \cdots, 1}_{L-k+1}]$. Fig. 2 depicts an example of the decomposition with $L = 4$.

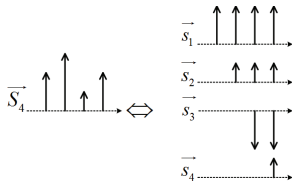

Fig. 2. Example of the decomposition of a SV region into gradient-based component-vectors. The extent of the SV region, L, is 4 in this example. Heights of the entries (arrows) in $\vec{S_4}$ are the *pixel values* in the SV region (figure on the left), whereas height of the non-zero entires in $\vec{s_k}$ is the *gradient value* at the k^{th} location in the SV region (figure on the right).

Given (2) and (3), the squared error between $\vec{S_L}$ and $\vec{S'_L}$ is:

$$\begin{aligned} ||E_S||_2^2 &= ||\vec{S_L} - \vec{S'_L}||_2^2 \\ &= \sum_{k=1}^{L} ||\vec{e_k}||_2^2 + 2\sum_{k=1}^{L-1}\sum_{l=k+1}^{L} \vec{e_k} \cdot \vec{e_l}. \end{aligned} \tag{4}$$

where $\vec{e_k} = \vec{s_k} - \vec{s'_k}$ is the *error vector* for the k^{th} gradient-based component-vector. The first term of (4) is given by

$$\begin{aligned} ||\vec{e_k}||_2^2 &= \sum_{i=1}^{L+d} (\vec{s_k}(i) - \vec{s'_k}(i))^2 \\ &= \begin{cases} 2dg_k^2 & \text{for } k = 1, 2\cdots, L-d; \\ 2(L-k+1)g_k^2 & \text{for } k = L-d+1, \cdots, L, \end{cases} \end{aligned} \tag{5}$$

where d is the average position error for this spatial variant region. The two cases that $k = 1, 2\cdots, L-d$ and $k = L-d+1, \cdots, L$ are illustrated in Fig. 3(a) and 3(b) respectively. With a position error d, the supports of $\vec{s_k}$ and $\vec{s'_k}$ overlap for $k = 1, 2\cdots, L-d$ (Fig. 3(a)). On the other hand, the supports of $\vec{s_k}$ and $\vec{s'_k}$ are disjoint

for $k = L-d+1, \cdots, L$ (Fig. 3(b)). These lead to different ways to calculate the error vector magnitude in (5).

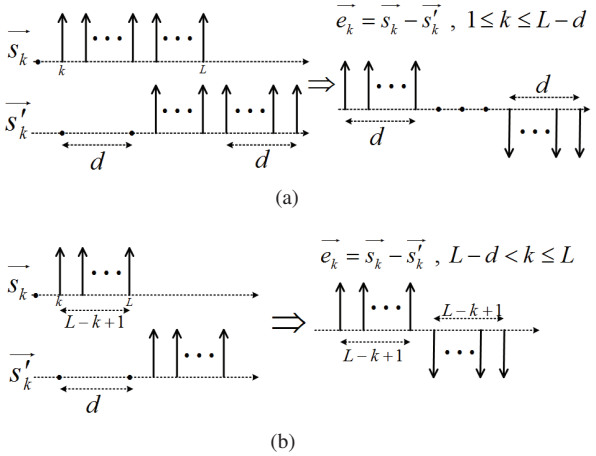

Fig. 3. (a) Rendering error for the k^{th} gradient-based component-vector when $1 \le k \le L-d$; (b) rendering error for the k^{th} gradient-based component-vector when $L-d < k \le L$. In (a), supports of $\vec{s}_k$ and $\vec{s}'_k$ overlap. In (b), supports of $\vec{s}_k$ and $\vec{s}'_k$ are disjoint. Note that the non-zero entries in $\vec{s}_k$ and $\vec{s}'_k$ are the same, since they are the decompositions of $\vec{S}_L$ and $\vec{S}'_L$ respectively, and $\vec{S}'_L$ is the shifted counterpart of $\vec{S}_L$.

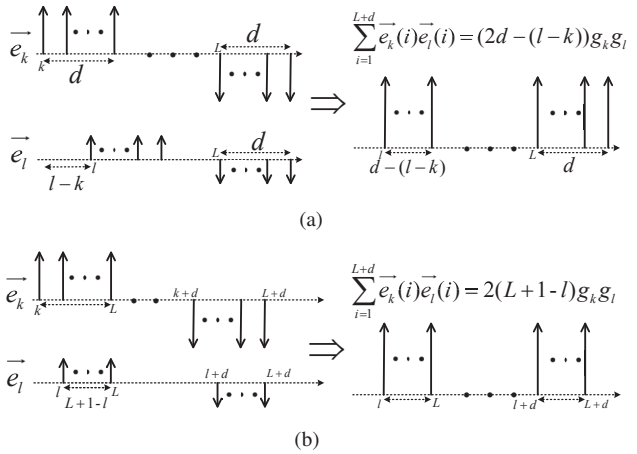

Fig. 4. (a) Rendering error of $\vec{e}_k \cdot \vec{e}_l$ when $1 \le k \le L-d$; (b) rendering error of $\vec{e}_k \cdot \vec{e}_l$ when $L-d < k \le L$.

Similarly, the second term of (4) can be derived from (5) and Fig. 3, as illustrated in (6) and Fig. 4 ,

$$\vec{e}_k \cdot \vec{e}_l = \begin{cases} (2d-(l-k))g_k g_l & \text{for } k = 1, 2 \cdots, L-d; \\ 2(L+1-l)g_k g_l & \text{for } k = L-d+1, \cdots, L. \end{cases} \tag{6}$$

Substituting (5) and (6) into (4), and representing it in matrix form, we have

$$||E_S||_2^2 = 2\sum_{i,j}(\mathbf{D} \circ \mathbf{G})_{ij} = 2\sum_{i,j}(\mathbf{D})_{ij}(\mathbf{G})_{ij} \tag{7}$$

where "$\circ$" represents the *Hadamard product* or the *element-wise multiplication* of two matrices. $\mathbf{D}$ and $\mathbf{G}$ are given by (8) and (9) respectively,

$$\mathbf{G} = [g_1, g_2, \cdots, g_L]^T [g_1, g_2, \cdots, g_L]. \tag{9}$$

The MSE of the rendering distortion in SV regions ($E[Z_{\text{SV}}^2]$) is then computed as

$$E[Z_{l,\text{SV}}^2] = \frac{1}{MN}\sum_{S \in \text{SV}} ||E_S||_2^2, \tag{10}$$

where $M \times N$ is the spatial dimension of a video frame. (10) can be used to estimate the overall distortion caused by depth errors ($E[Z_l^2]$).

Note that here we use the average position error d of a particular SV region instead of per-pixel position errors to estimate the distortion, in order to simplify the computation. This can be justified by: (i) L is usually small. (ii) Variation of position errors is usually small for consecutive pixels. In particular, change in position-error per unit change in depth-map-error is usually small (Table I).

Table I. Change in position-error per unit change in depth-map-error following the MPEG 3DV 2-view test cases [17]

Video Sequence	Change in Position-error
Kendo	0.094118
Balloons	0.094118
PoznanHall2	0.200000
PoznanStreet	0.154902

Low-complexity Estimation. Here we discuss how to simplify (7) to compute $||E_S||_2^2$. A SV region consists of pixels around a sharp edge and the extent (width) of a SV region is usually small (e.g., see Fig. 1(b)). That is, L is very small for a typical $\vec{S}_L$. Thus, it is reasonable to use linear approximation to approximate the L pixel values in a SV region. Specifically, we approximate the gradient values (g_k, $k = 1, 2, \cdots, L$) in the spatial variant regions with the mean of all the g_k:

$$g_0 = \frac{1}{L}\sum_{k=1}^{L} g_k, \tag{11}$$

and the gradient-based component-vectors $\vec{s}_k$ and the texture SV region $\vec{S}_L$ are approximated by $\vec{t}_k = g_0 \vec{1}_k$ and $\vec{T}_L = \sum_{k=1}^{L} \vec{t}_k$ respectively. The corresponding rendering distortion for $\vec{T}_L$, given an average horizontal disparity error d, is simply:

$$\begin{aligned} ||E_T||_2^2 &= 2g_0^2 \sum_{i,j}(\mathbf{D})_{ij} \\ &= \begin{cases} (-\frac{d^3}{3} + L^2 d + Ld + \frac{d}{3})g_0^2 & \text{for } d \le L; \\ L(L+1)g_0^2 & \text{otherwise.} \end{cases} \end{aligned} \tag{12}$$

We use (12) in lieu of (7) to estimate the rendering distortions in SV regions. Clearly, (12) requires negligible computation complexity.

III. EXPERIMENTS

We have performed experiments to verify the accuracy of the proposed models. Following the camera configurations in the MPEG 3DV 2-view test cases [17], two reference views were used to render a virtual view in-between. Both the texture and depth videos were encoded with JMVC Encoder 8.3.1. Each group-of-pictures consisted of an anchor frame and several hierarchically coded B frames. Inter-view prediction was also used in encoding.

$$\mathbf{D} = \begin{bmatrix} d & 2d-1 & 2d-2 & \cdots & d+1 & \cdots & d & d-1 & \cdots & 1 \\ 0 & d & 2d-1 & 2d-2 & \cdots & d+1 & \cdots & d-1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & d & 2d-1 & 2d-3 & 2d-5 & \cdots & 3 & 1 \\ 0 & \cdots & 0 & 0 & d & 2d-2 & 2d-4 & \cdots & 4 & 2 \\ 0 & \cdots & 0 & 0 & 0 & d-1 & 2d-4 & \cdots & 4 & 2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 3 & 4 & 2 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{L \times L} \tag{8}$$

Quantization parameters (qp) were set to be 32, 36, 40 and 44 for both texture and depth image encoding. VSRS 3.5 [18] was used to synthesis the virtual view.

We used PoznanHall2 (1920 × 1088), PoznanStreet (1920 × 1088), Kendo (1024 × 768) and Balloons (1024 × 768) in our experiment. Figure 5 compares the empirical results and the model results for sequence PoznanHall2. The empirical results were measured from the rendering output of VSRS. As shown in the figure, the model can accurately estimate the rendering quality with different encoding conditions and situations. The results also suggest that, with lower quality texture images (e.g., color_qp = 44 in Fig. 5(d)), only small gains in the rendering output can be obtained with improving the quality of the depth images (reducing depth_qp). This is because with lower quality texture images the noise due to texture coding $E[N^2]$ dominates the overall synthesis noise power in (1), and reduction in $E[Z^2]$ has only a small impact. On the other hand, when the texture images have good quality (e.g., color_qp = 32 in Fig. 5(a)), large gains in the rendering quality can be obtained with improving the quality of the depth images (reducing depth_qp). Results for PoznanStreet, Kendo and Balloons are similar [20].

To further illustrate the characteristics of rendering distortion caused by texture error ($E[N^2]$) and depth error ($E[Z^2]$), we plot $E[N^2]$ and $E[Z^2]$ respectively in Fig. 6. As we expect, the empirical and model results of $E[N^2]$ remain unchanged for different depth image quality, as depicted in Fig. 6(a) and (c). Another observation is that a large texture qp (color_qp = 44) causes large texture-error induced distortion $E[N^2]$ (MSE is over 34, see Fig. 6(c)), while depth-error induced distortion $E[Z^2]$ is relatively small (MSE is less than 6, see Fig. 6(d)). Such large $E[N^2]$ dominates the final rendering quality, resulting in a relatively small change in rendering quality at different depth map quality (see Fig. 5(d)). On the other hand, for a smaller color qp (color_qp = 32), the magnitude of the texture-error induced distortion (MSE is around 6) is comparable to the one caused by depth error. Therefore, the total rendering quality would be equally affected by both texture error and depth error, and we can observe noticeable variation in rendering quality at different depth map quality (see Fig. 5(a)).

IV. CONCLUSIONS

We have proposed an analytical model to estimate the synthesized view quality in 3D video. Our model combines frequency and spatial analysis. Frequency analysis provides a concise and compact representation to understand the synthesis distortions, while spatial analysis accounts for non-stationary. We also derived equations to estimate the synthesis distortions in spatial variant regions along strong edges. Experiment results showed that the model can accurately estimate the synthesis noise power.

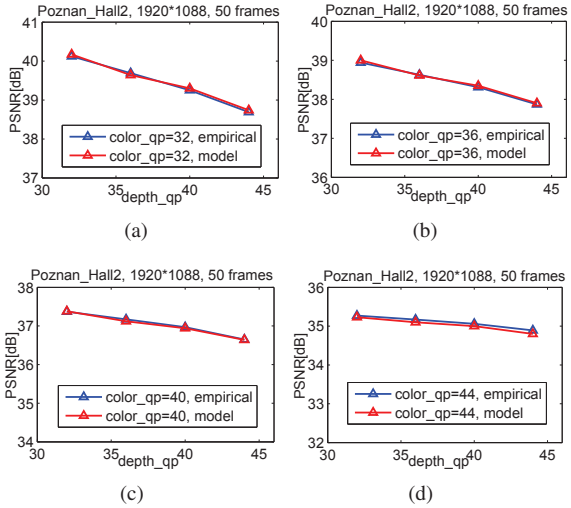

Fig. 5. Modeling result: PoznanHall2.

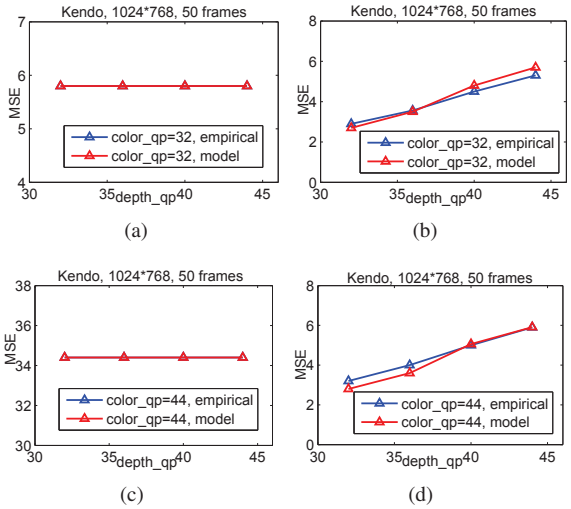

Fig. 6. Modeling result of Kendo: (a) rendering distortion caused by texture error ($E[N^2]$) when color_qp = 32; (b) rendering distortion caused by depth error ($E[Z^2]$) when color_qp = 32; (c) rendering distortion $E[N^2]$ when color_qp = 44; (d) rendering distortion $E[Z^2]$ when color_qp = 44.

V. ACKNOWLEDGMENT

This work was supported in part by SUTD-ZJU Research Collaboration Grant ZJUSP1200104.

VI. REFERENCES

[1] A. Vetro, W. Matusik, H. Pfister, and J. Xin, "Coding approaches for end-to-end 3D TV systems," in *Proc. Picture Coding Symposium (PCS)*, 2004.

[2] T. Maugey, P. Frossard, and G. Cheung, "Consistent view synthesis in interactive multiview imaging," in *Proc. IEEE Int'l Conf. Image Processing (ICIP)*, 2012.

[3] Z. Zhang, R. Wang, C. Zhou, Y. Wang, and W. Gao, "A compact stereoscopic video representation for 3D video generation and coding," in *Proc. IEEE Data Compression Conference (DCC)*, 2012.

[4] D. Min, J. Lu, and M. N. Do, "Depth video enhancement based on weighted mode filtering," *IEEE Trans. Image Processing*, vol. 21, no. 3, pp. 1176–1190, 2012.

[5] D. Min, D. Kim, S. Yun, and K. Sohn, "2D/3D freeview video generation for 3DTV system," *Signal Processing: Image Communication*, vol. 24, no. 1-2, pp. 31–48, 2009.

[6] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. Mueller, P. H. N., and T. Wiegand, "The effects of multiview depth video compression on multiview rendering," *Signal Process.: Image Commun*, vol. 24, pp. 73–88, 2009.

[7] D. Tian, A. Vetro, and M. Brand, "A trellis-based approach for robust view synthesis," in *Proc. IEEE Int'l Conf. Image Processing (ICIP)*, 2011.

[8] I. Daribo, C. Tillier, and B. Pesquet-Popescu, "Adaptive wavelet coding of the depth map for stereoscopic view synthesis," in *Proc. IEEE Int'l Workshop on Multimedia Signal Processing (MMSP)*, 2008.

[9] A. Sanchez, G. Shen, and A. Ortega, "Edge-preserving depth-map coding using tree-based wavelets," in *Proc. Asilomar Conf. Signals, Systems, and Computers*, 2009.

[10] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map distortion analysis for view rendering and depth coding," in *Proc. IEEE Int'l Conf. Image Processing (ICIP)*, 2009.

[11] H. T. Nguyen and M. N. Do, "Error analysis for image-based rendering with depth information," *IEEE Trans. Image Processing*, vol. 18, no. 4, pp. 703–716, 2009.

[12] 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," *Image Commun.*, vol. 24, no. 8, pp. 666–681, Sep. 2009.

[13] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map coding with distortion estimation of rendered view," in *Proc. SPIE Visual Information Processing and Communication (VIPC)*, 2010.

[14] V. Velisavljevic, G. Cheung, and J. Chakareski, "Bit allocation for multiview image compression using cubic synthesized view distortion model," in *Proc. IEEE International Workshop on Hot Topics in 3D*, 2011.

[15] K. Takahashi, "Theoretical analysis of view interpolation with inaccurate depth information," *IEEE Trans. Image Processing*, vol. 21, no. 2, pp. 718–732, 2012.

[16] N.-M. Cheung, D. Tian, A. Vetro, and H. Sun, "On modeling the rendering error in 3D video," in *Proc. IEEE Int'l Conf. Image Processing (ICIP)*, 2012.

[17] MPEG Video and Requirement Group, "Call for proposals on 3D video coding technology," MPEG, Tech. Rep., 2011, MPEG N12036.

[18] MPEG, "View synthesis software manual release 3.5 (VSRS 3.5)," ISO/IEC JTC1/SC29/WG11 MPEG, Tech. Rep., 2009.

[19] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. on System, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[20] L. Fang, N.-M. Cheung, D. Tian, A. Vetro, H. Sun, and O. C. Au, "An analytical model for synthesis distortion estimation in 3D video," *IEEE Trans. Image Processing*, Dec. 2012, submitted December 2012.