# FDQM: Fast Quality Metric for Depth Maps Without View Synthesis

Won-Dong Jang, *Student Member, IEEE*, Tae-Young Chung, *Member, IEEE*, Jae-Young Sim, *Member, IEEE*, and Chang-Su Kim, *Senior Member, IEEE*

***Abstract*— We propose a fast quality metric for depth maps, called fast depth quality metric (FDQM), which efficiently evaluates the impacts of depth map errors on the qualities of synthesized intermediate views in multiview video plus depth applications. In other words, the proposed FDQM assesses view synthesis distortions in the depth map domain, without performing the actual view synthesis. First, we estimate the distortions at pixel positions, which are specified by reference disparities and distorted disparities, respectively. Then, we integrate those pixel-wise distortions into an FDQM score by employing a spatial pooling scheme, which considers occlusion effects and the characteristics of human visual attention. As a benchmark of depth map quality assessment, we perform a subjective evaluation test for intermediate views, which are synthesized from compressed depth maps at various bitrates. We compare the subjective results with objective metric scores. Experimental results demonstrate that the proposed FDQM yields highly correlated scores to the subjective ones. Moreover, FDQM requires at least 10 times less computations than conventional quality metrics, since it does not perform the actual view synthesis.**

***Index Terms*— 3-D video, depth map quality assessment, image quality assessment, multiview video plus depth (MVD), spatial pooling, virtual view synthesis.**

## I. Introduction

RECENTLY, 3-D video technologies have been researched intensively, and their various applications have been developed, including 3-D television and free-view television. To represent 3-D scenes in these applications, multiview sequences can be used, which are taken from different viewpoints. A multiview sequence, however, requires a larger amount of data than a single-view sequence. Moreover, high-definition or ultrahigh definition television has become popular, which further increases the data requirement of multiview sequences. It is necessary to compress multiview video sequences compactly [1].

The data size of a multiview video sequence is proportional to the number of views. Thus, the multiview video plus depth (MVD) format [2] was proposed to reduce the number of views. In the MVD format, typically two or three views of color and depth videos are encoded, and an intermediate view from an arbitrary viewpoint is synthesized from the encoded views using the depth-image-based rendering (DIBR) techniques [3]. In DIBR, 3-D points are reconstructed from pixels in the encoded views and then projected onto the intermediate view. Various attempts have been made to compress MVD data [4]–[7]. The performance of an MVD compression algorithm is typically evaluated by measuring the qualities of 2-D synthesized intermediate views. Specifically, a reference intermediate view is synthesized using original color and depth maps, and its distorted version is synthesized from compressed color and depth maps. Then, the difference between the reference (or error-free) view and the distorted view is computed using a 2-D image quality metric. However, this process should perform the computationally heavy view synthesis twice. It is desirable to develop an efficient quality metric for synthesized views without the actual view synthesis, although there are various conventional metrics [8]–[10].

In this paper, we investigate the effects of depth map compression errors on the qualities of 2-D synthesized views in MVD applications. Then, we propose a fast depth quality metric (FDQM) for erroneous depth maps. Based on the assumption of local disparity constancy, we estimate the distortions of synthesized views, caused by depth errors, without the actual view synthesis. We compute pixel-wise distortions and integrate them into an FDQM score using a spatial pooling scheme, which reflects occlusion effects and human visual system (HVS) characteristics. For the performance evaluation, we compress depth maps with various quantization parameters (QPs) and synthesize intermediate views from the compressed maps. We perform a subjective evaluation test to assess the qualities of the distorted intermediate views in comparison with the error-free ones. We measure the correlation of the subjective scores to the objective scores of the proposed FDQM and several conventional quality metrics, respectively. The test results show that FDQM is highly correlated to the subjective assessment and yields comparable or better performance than the conventional metrics, while demanding significantly less computations.

Manuscript received May 20, 2014; revised September 9, 2014; accepted November 8, 2014. Date of publication November 20, 2014; date of current version June 30, 2015. This work was supported in part by the National Research Foundation of Korea through the Ministry of Science, ICT and Future Planning (MSIP) under Grant 2009-0083495 and in part by the National Research Foundation of Korea through the Korean Government within the MSIP under Grant 2012-011031. This paper was recommended by Associate Editor W. Zeng.

W.-D. Jang and C.-S. Kim are with the School of Electrical Engineering, Korea University, Seoul 136-701, Korea (e-mail: wdjang@mcl.korea.ac.kr; changsukim@korea.ac.kr).

T.-Y. Chung is with the Software Center, Samsung Electronics Company, Ltd., Suwon 440-746, Korea (e-mail: ty83.chung@samsung.com).

J.-Y. Sim is with the School of Electrical and Computer Engineering, Ulsan National Institute of Science and Technology, Ulsan 689-798, Korea (e-mail: jysim@unist.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2014.2372343

The remainder of this paper is organized as follows. Section II reviews conventional image quality metrics. Section III describes how to estimate view synthesis distortions without the actual view synthesis. Section IV presents the proposed FDQM for depth maps. Section V provides experimental results. Finally, Section VI concludes this paper and discusses future work.

## II. Related Work

The peak signal-to-noise ratio (PSNR) is a widely used image quality metric. However, since PSNR poorly correlates with human perception characteristics in many cases [11], many alternative quality metrics have been developed. This section briefly surveys conventional metrics for the quality assessment of ordinary 2-D images or stereoscopic 3-D images. More detailed survey and evaluation of image quality metrics can be found in [8]–[10].

### A. 2-D Image Quality Assessment

Numerous 2-D image quality metrics have been proposed. For example, Teo and Heeger [12] introduced a perceptual distortion measure by modeling the response properties of neurons in the primary visual cortex and the psychophysics of spatial pattern detection. Lai and Kuo [13] proposed a quality measure, which uses the Haar wavelet to model the space-frequency localization of HVS. Damera-Venkata *et al*. [14] developed the noise quality measure (NQM). It first performs the image restoration on a reference image, as well as on a degraded image, and then measures the contrast difference between the restored images at various scales. Wang and Bovik [15] proposed the universal quality index (UQI), which considers structural distortions as well as luminance distortions. Wang *et al*. [16] generalized UQI and developed the structural similarity (SSIM). Wang *et al*. [17] also proposed the multiscale SSIM (MS-SSIM) to compute SSIM at different image scales. Wang and Li [18] introduced the information content weighted PSNR (IW-PSNR) and the information content weighted SSIM (IW-SSIM) by applying a statistical weighting scheme to the conventional metrics of PSNR and SSIM. Sheikh *et al*. [19] presented the information fidelity criterion (IFC), which measures the mutual information between reference and distorted signals. Sheikh and Bovik [20] developed the visual information fidelity (VIF), which computes the mutual information between a reference signal and a perceived signal based on HVS modeling.

In addition, Shnayderman *et al*. [21] introduced the singular value decomposition based quality metric (M-SVD), which measures the squared differences between the singular values of reference and distorted image blocks. Ponomarenko *et al*. [22] proposed the HVS-based PSNR (PSNR-HVS-M), which exploits the contrast sensitivity masking property in the discrete cosine transform domain. Chandler and Hemami [23] proposed a wavelet-based metric, called visual signal-to-noise ratio (VSNR), which considers HVS properties, such as distortion contrast and global precedence. Larson and Chandler [24] employed two separate measures, one for low-quality images and the other for high-quality images, and combined them into an overall metric, called most apparent distortion (MAD). Zhang *et al*. [25] proposed the feature similarity (FSIM), which uses the features of phase congruency and gradient magnitude to measure the similarity between reference and distorted images. Liu *et al*. [26] developed a quality metric based on gradient similarity (GSM), which combines luminance similarity and GSM together. Wu *et al*. [27] introduced a perceptual quality metric based on the internal generative mechanism (IGM), which models visual information degradation and uncomfortable sensation. Xue *et al*. [28] measured a gradient magnitude similarity to design an image quality metric, called gradient magnitude similarity deviation (GMSD).

These general 2-D image quality metrics can be used for the assessment of depth maps in MVD applications. On the one hand, we may measure the difference between a reference depth map and its distorted version directly, but this cannot reflect the qualities of synthesized intermediate views accurately. On the other hand, we may use the metrics to measure the difference between reference and distorted intermediate views, which are synthesized from error-free and erroneous depth maps, respectively. In this case, however, the view synthesis should be carried out, requiring high-computational complexity.

Recently, Conze *et al*. [29] proposed the VSQA metric to assess synthesized intermediate views in multiview video applications. It measures the quality of a synthesized view using a weighted SSIM score, where the weights are determined based on color intensity, orientation diversity, and contrast. However, VSQA mainly assesses the rendering capabilities of various DIBR techniques, rather than the effects of compression errors in multiview video sequences.

### B. 3-D Image Quality Assessment

There have been several studies on the 3-D quality assessment of stereoscopic images, which can be regarded as image frames of double-view video sequences. Benoit *et al*. [30] measured the distortion of a depth map, as well as those of left and right images. De Silva *et al*. [31] presented a stereoscopic video quality metric, called stereoscopic structural distortion, which considers structural distortions, blur distortions, and depth distortions jointly. Hewage *et al*. [32] analyzed the qualities of stereoscopic images, which were synthesized from erroneous color and depth images. Based on this paper, Joveluro *et al*. [33] proposed a quality metric considering luminance and contrast differences.

Notice that all these metrics for stereoscopic images assess the qualities of 3-D images, which are rendered on 3-D monitors and watched with special glasses. However, as mentioned previously, in MVD applications, the performance of a compression algorithm is typically evaluated by measuring the qualities of 2-D synthesized views. In such a case, we need a proper 2-D quality metric for synthesized views.

### C. Summary

Whereas there are many quality metrics for 2-D images and several metrics for stereoscopic 3-D images, little work has

been done to develop a specific quality metric for depth maps in MVD applications. Depth errors have different impacts on synthesized intermediate views than color errors; a small error in a depth map may lead to severe degradation in a synthesized view. These properties of depth data have not been considered systematically in the conventional metrics. In this paper, we propose a fast quality metric for depth maps, FDQM, which measures the impacts of depth map errors on the qualities of 2-D synthesized views efficiently.

## III. Estimation of View Synthesis Distortions

### A. View Synthesis and Its Complexity

Let us first describe a simple view synthesis procedure. An intermediate view is synthesized by warping left and right views using disparities. Let $\mathbf{p}$ be a pixel in the right view. In addition, let $\mathbf{d}(\mathbf{p})$ be the disparity at $\mathbf{p}$, which is determined from the depth $z(\mathbf{p})$ via

$$\mathbf{d}(\mathbf{p}) = \left[ lb\left( \frac{z(\mathbf{p})}{255}\left( \frac{1}{z_n} - \frac{1}{z_f} \right) + \frac{1}{z_f} \right), 0 \right]^T \tag{1}$$

where $l$ is the focal length, $b$ is the baseline distance between the right view and the intermediate view, and $z_n$ and $z_f$ denote the nearest and the farthest depths, respectively [6]. Then, the color $J_{\text{right}}$ of the intermediate view is synthesized from the right view $I_{\text{right}}$ via

$$J_{\text{right}}(\mathbf{p} + \mathbf{d}(\mathbf{p})) = I_{\text{right}}(\mathbf{p}). \tag{2}$$

In addition, the synthesized view $J_{\text{left}}$ is symmetrically obtained from the left view $I_{\text{left}}$. Then, $J_{\text{right}}$ and $J_{\text{left}}$ are blended together to yield the final intermediate view

$$J(\mathbf{p}) = \lambda J_{\text{right}}(\mathbf{p}) + (1-\lambda) J_{\text{left}}(\mathbf{p}) \tag{3}$$

where $\lambda$ specifies the relative position of the intermediate view between the left and right views.

We analyze the computational complexity of this simple view synthesis. Let $C_a$ and $C_m$ denote the complexity of a single addition and a single multiplication, respectively. First, the depth-to-disparity conversion in (1) is required for all pixels in both left and right views, resulting in the complexity of $2N(C_a + C_m)$, where $N$ is the number of pixels in a view. Second, the warping in (2) is done with one addition per pixel, requiring the complexity of $2NC_a$ for both views. Note that a warped image may contain holes, when matching pixels are unavailable in the original view. In addition, occlusion may occur, when two or more color values are warped to the same pixel position. Therefore, hole filling and occlusion handling are necessary to generate a complete warped image. However, assuming they are not performed in the simple view synthesis, we exclude their computational complexities in this analysis. Third, the blending in (3) demands two multiplications and one addition. Thus, the complexity of $N(C_a + 2C_m)$ is required for combining the two warped images. Consequently, the overall complexity

$$C_{\text{syn}} = 5NC_a + 4NC_m \tag{4}$$

is required to render an intermediate frame. Notice that $C_{\text{syn}}$ in (4) should be regarded as a low bound for the view synthesis complexity. More sophisticated view synthesizers demand much more computations. For example, the view synthesis reference software (VSRS) [34], commonly used in 3-D video communications, demands a significantly higher complexity due to additional operations, such as homography computation, hole filling, and boundary noise removal.

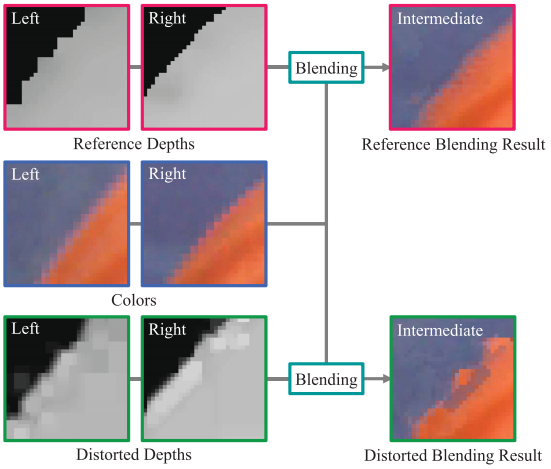



Fig. 1. Effects of depth map errors on view synthesis results. Color images in the left and right views are warped using depth maps and then blended to synthesize an intermediate view. The reference blending result (top) is obtained from the error-free depth maps, and the distorted blending result (bottom) is obtained from the erroneous depth maps distorted by compression errors. The distorted blending result is degraded severely near the object boundary.

### B. View Synthesis Distortion Model

In MVD data communications, the encoder compresses and transmits color and depth videos in the left and right views, and the decoder synthesizes an intermediate view from the received color and depth videos. Compression errors in depth maps affect the qualities of synthesized views. As shown in Fig. 1, distorted depth maps cause severe artifacts in synthesized views, especially along object boundaries. To measure the distortions in a synthesized view, we may perform the view synthesis procedure in Section III-A directly. However, even the simple synthesis requires the complexity in (4), which may be burdensome in applications. In addition, as mentioned above, more sophisticated synthesizers demand even higher complexities, and using these synthesizers is not feasible. It is hence desirable to estimate the effects of depth errors on synthesized views without the actual view synthesis. For example, suppose that we should determine a QP for a depth map during the encoding of an MVD sequence. For the rate-distortion optimization, we should estimate the quality of a synthesized view for each candidate QP. If the estimation is possible with less computations than the actual view synthesis, the computational burden of the rate-distortion optimization can be reduced significantly.

Our view synthesis distortion model estimates the qualities of synthesized views faithfully from the distortions of depth maps without the actual view synthesis, but requires an even lower complexity than the simple synthesis in Section III-A. For the sake of simplicity, we describe the view synthesis from the right view only, omit the subscript right from notations, and

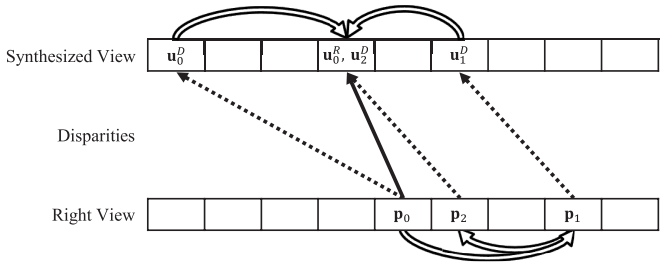



Fig. 2. Distortion estimation at a reference pixel position $\mathbf{u}_0^R$ in a synthesized view. Pixel $\mathbf{u}_0^R$ corresponds to pixel $\mathbf{p}_0$ in a right view via a reference disparity, indicated by a solid arrow. Dotted arrows depict distorted disparities.

use $I$ and $J$ instead of $I_{\text{right}}$ and $J_{\text{right}}$. Our distortion model considers only the pixel positions in the right view that match valid pixel positions in the synthesized view. In other words, if a pixel in the right view corresponds to a pixel position outside the synthesized view, its distortion is not considered. Let $\mathbf{P}$ and $\mathbf{Q}$ denote the sets of pixels in the right view, which are mapped to valid pixel positions in the synthesized view by reference (or error-free) disparities and distorted disparities, respectively. We refer to the valid pixel positions in the synthesized view, which correspond to $\mathbf{P}$ and $\mathbf{Q}$, as *reference pixel positions* and *distorted pixel positions*, respectively. Let us estimate the view synthesis distortions at the reference pixel positions and at the distorted pixel positions subsequently.

*1) Distortions at Reference Pixel Positions:* Given a pixel position $\mathbf{p}_0 \in \mathbf{P}$ in the right view $I$, the reference disparity $\mathbf{d}^R(\mathbf{p}_0)$, which is depicted by a solid arrow in Fig. 2, determines the reference pixel position $\mathbf{u}_0^R$ at the synthesized view $J$

$$\mathbf{u}_0^R = \mathbf{p}_0 + \mathbf{d}^R(\mathbf{p}_0). \tag{5}$$

The true pixel value $J(\mathbf{u}_0^R)$ at $\mathbf{u}_0^R$ should be identical with $I(\mathbf{p}_0)$. However, errors in the depth map may distort the reference disparity, yielding a different reconstruction value $\tilde{J}(\mathbf{u}_0^R)$ at $\mathbf{u}_0^R$. We attempt to estimate the distortion $|J(\mathbf{u}_0^R) - \tilde{J}(\mathbf{u}_0^R)|$ efficiently without the actual view synthesis.

Suppose that the reference disparity $\mathbf{d}^R(\mathbf{p}_0)$ is reconstructed to a distorted disparity $\mathbf{d}^D(\mathbf{p}_0)$ in the decoder, which is depicted by a dotted arrow in Fig. 2. It matches a wrong pixel $\mathbf{u}_0^D$ to $\mathbf{p}_0$, given by

$$\mathbf{u}_0^D = \mathbf{p}_0 + \mathbf{d}^D(\mathbf{p}_0). \tag{6}$$

To estimate the reconstruction value $\tilde{J}(\mathbf{u}_0^R)$, we assume that the distorted disparities of neighboring pixels of $\mathbf{p}_0$ are identical with $\mathbf{d}^D(\mathbf{p}_0)$. Under this assumption of local disparity constancy, $\mathbf{p}_1$ in the right view, which is at the distance of the disparity difference $(\mathbf{u}_0^R - \mathbf{u}_0^D)$ from $\mathbf{p}_0$, would match $\mathbf{u}_0^R$ in Fig. 2. In other words, we have

$$\mathbf{p}_1 = \mathbf{p}_0 + \left(\mathbf{u}_0^R - \mathbf{u}_0^D\right) \tag{7}$$

$$= \mathbf{p}_0 + \mathbf{d}^R(\mathbf{p}_0) - \mathbf{d}^D(\mathbf{p}_0) \tag{8}$$

and $I(\mathbf{p}_1)$ is an estimate of the reconstruction value $\tilde{J}(\mathbf{u}_0^R)$ at the reference pixel position $\mathbf{u}_0^R$.

However, a single candidate $I(\mathbf{p}_1)$ may not be a reliable estimator of $\tilde{J}(\mathbf{u}_0^R)$. Therefore, we obtain more candidates. As shown in Fig. 2, $\mathbf{p}_1$ is matched to

$$\mathbf{u}_1^D = \mathbf{p}_1 + \mathbf{d}^D(\mathbf{p}_1) \tag{9}$$

by the distorted disparity $\mathbf{d}^D(\mathbf{p}_1)$. We again assume that the distorted disparities of neighboring pixels of $\mathbf{p}_1$ are equal to $\mathbf{d}^D(\mathbf{p}_1)$. Then, as in (8)

$$\mathbf{p}_2 = \mathbf{p}_1 + \left(\mathbf{u}_0^R - \mathbf{u}_1^D\right) \tag{10}$$

$$= \mathbf{p}_0 + \mathbf{d}^R(\mathbf{p}_0) - \mathbf{d}^D(\mathbf{p}_1) \tag{11}$$

would match $\mathbf{u}_0^R$, and $I(\mathbf{p}_2)$ becomes another estimator of $\tilde{J}(\mathbf{u}_0^R)$. In this way, $I(\mathbf{p}_1)$ is the first-order estimator of $\tilde{J}(\mathbf{u}_0^R)$, and $I(\mathbf{p}_2)$ is the second-order estimator of $\tilde{J}(\mathbf{u}_0^R)$. Then, $\mathbf{p}_2$ is matched to

$$\mathbf{u}_2^D = \mathbf{p}_2 + \mathbf{d}^D(\mathbf{p}_2) \tag{12}$$

by the distorted disparity $\mathbf{d}^D(\mathbf{p}_2)$. In the example of Fig. 2, $\mathbf{u}_2^D$ equals $\mathbf{u}_0^R$, and thus the second-order estimator $I(\mathbf{p}_2)$ exactly equals $\tilde{J}(\mathbf{u}_0^R)$.

We generalize this estimation scheme in (8) and (11) to obtain higher order estimators recursively

$$\mathbf{p}_i = \mathbf{p}_{i-1} + \left(\mathbf{u}_0^R - \mathbf{u}_{i-1}^D\right) \tag{13}$$

$$= \mathbf{p}_0 + \mathbf{d}^R(\mathbf{p}_0) - \mathbf{d}^D(\mathbf{p}_{i-1}), \quad i = 1, 2, 3, \ldots \tag{14}$$

and $I(\mathbf{p}_i)$ becomes the $i$th-order estimate of $J(\mathbf{u}_0^R)$. Then, $\mathbf{p}_i$ is matched to

$$\mathbf{u}_i^D = \mathbf{p}_i + \mathbf{d}^D(\mathbf{p}_i), \quad i = 1, 2, 3, \ldots \tag{15}$$

We obtain the overall estimator of the reconstruction value $\tilde{J}(\mathbf{u}_0^R)$ by combining these candidates $I(\mathbf{p}_i)$'s linearly. As the distance $\|\mathbf{u}_0^R - \mathbf{u}_i^D\|$ gets shorter, $I(\mathbf{p}_i)$ becomes a more reliable estimator for $\tilde{J}(\mathbf{u}_0^R)$. Thus, we employ an exponential weight function

$$\varphi(\mathbf{p}_0, \mathbf{p}_i) = e^{-\|\mathbf{u}_0^R - \mathbf{u}_i^D\|} \tag{16}$$

and determine the overall estimator $\hat{J}(\mathbf{u}_0^R)$ of $\tilde{J}(\mathbf{u}_0^R)$ as

$$\hat{J}(\mathbf{u}_0^R) = \frac{\sum_{i=1}^{m(\mathbf{p}_0)} \varphi(\mathbf{p}_0, \mathbf{p}_i) I(\mathbf{p}_i)}{\sum_{i=1}^{m(\mathbf{p}_0)} \varphi(\mathbf{p}_0, \mathbf{p}_i)} \tag{17}$$

where $m(\mathbf{p}_0)$ denotes the number of employed candidates $I(\mathbf{p}_i)$'s. Consequently, we estimate the distortion at the reference pixel position $\mathbf{u}_0^R$ as the squared difference $(J(\mathbf{u}_0^R) - \hat{J}(\mathbf{u}_0^R))^2$ between the true value $J(\mathbf{u}_0^R) = I(\mathbf{p}_0)$ and the estimated reconstruction value $\hat{J}(\mathbf{u}_0^R)$.

It is worth pointing out that all terms in (14)–(17) are computed without requiring the view synthesis. The proposed model, hence, does not perform the actual view synthesis to estimate the distortions in synthesized pixel values. Furthermore, the proposed model has the desirable *locking property* that, if the $i^*$th order estimator is accurate, all higher order estimators are also accurate. Specifically, suppose that the $i^*$th order estimator $I(\mathbf{p}_{i^*})$ equals $\tilde{J}(\mathbf{u}_0^R)$, i.e., $\mathbf{u}_0^R = \mathbf{u}_{i^*}^D$. Then, $\mathbf{p}_{i^*+1} = \mathbf{p}_{i^*}$ from (13), and all estimators with higher orders than $i^*$ become identical to the true reconstruction value $\tilde{J}(\mathbf{u}_0^R)$. This indicates that our recursive estimation tends to converge to the true value.

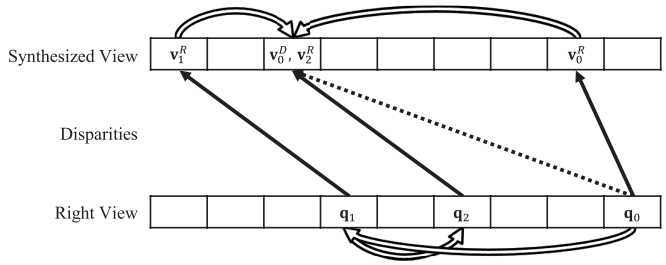



Fig. 3. Distortion estimation at a distorted pixel position $\mathbf{v}_0^D$ in a synthesized view. Pixel $\mathbf{v}_0^D$ corresponds to pixel $\mathbf{q}_0$ in a right view via a distorted disparity, indicated by a dotted arrow. Solid arrows depict reference disparities.

*2) Distortions at Distorted Pixel Positions:* Next, we estimate the view synthesis distortion at each distorted pixel position. To this end, in contrast to the model in Fig. 2, we exchange the roles of reference disparities and distorted disparities. As shown in Fig. 3, let $\mathbf{q}_0 \in \mathbf{Q}$ be a pixel position in the right view. It is mapped to a distorted pixel position

$$\mathbf{v}_0^D = \mathbf{q}_0 + \mathbf{d}^D(\mathbf{q}_0) \tag{18}$$

in the synthesized view by the distorted disparity $\mathbf{d}^D(\mathbf{q}_0)$. The corresponding reference pixel position

$$\mathbf{v}_0^R = \mathbf{q}_0 + \mathbf{d}^R(\mathbf{q}_0) \tag{19}$$

is determined by the reference disparity $\mathbf{d}^R(\mathbf{q}_0)$. From (18), the reconstruction value $\tilde{J}(\mathbf{v}_0^D)$ is equal to $I(\mathbf{q}_0)$.

To compute the distortion $|J(\mathbf{v}_0^D) - \tilde{J}(\mathbf{v}_0^D)|$ at the distorted pixel position $\mathbf{v}_0^D$, we should estimate the true value $J(\mathbf{v}_0^D)$, which is obtained using a reference disparity. We assume that the reference disparities of neighboring pixels of $\mathbf{q}_0$ are identical with $\mathbf{d}^R(\mathbf{q}_0)$. Then

$$\mathbf{q}_1 = \mathbf{q}_0 + \left(\mathbf{v}_0^D - \mathbf{v}_0^R\right) \tag{20}$$

$$= \mathbf{q}_0 + \mathbf{d}^D(\mathbf{q}_0) - \mathbf{d}^R(\mathbf{q}_0) \tag{21}$$

would match $\mathbf{v}_0^D$. Thus, $I(\mathbf{q}_1)$ becomes the first-order estimator of $J(\mathbf{v}_0^D)$. In general, as in (14), we have the recursion

$$\mathbf{q}_i = \mathbf{q}_0 + \mathbf{d}^D(\mathbf{q}_0) - \mathbf{d}^R(\mathbf{q}_{i-1}), \quad i = 1, 2, 3, \ldots \tag{22}$$

and $I(\mathbf{q}_i)$ becomes the $i$th-order estimator of $J(\mathbf{v}_0^D)$. The corresponding pixel $\mathbf{v}_i^R$ in the synthesized view is given by

$$\mathbf{v}_i^R = \mathbf{q}_i + \mathbf{d}^R(\mathbf{q}_i), \quad i = 1, 2, 3, \ldots \tag{23}$$

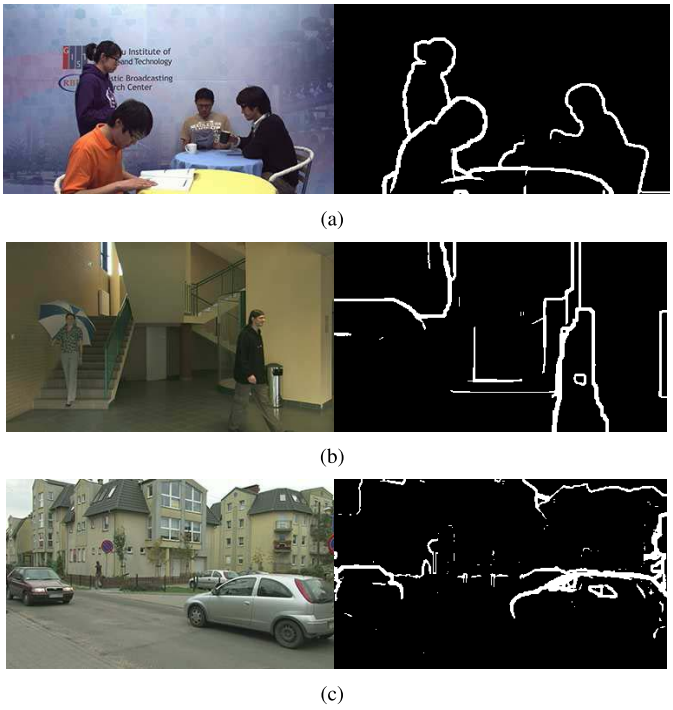
In the example of Fig. 3, the second-order estimator $I(\mathbf{q}_2)$ becomes identical to the true value $J(\mathbf{v}_0^D)$. Then, because of the locking property of the recursion in (20), all estimators with higher orders than 2 also become identical to $J(\mathbf{v}_0^D)$.

Next, we obtain the overall estimator $\check{J}(\mathbf{v}_0^D)$ of the true value $J(\mathbf{v}_0^D)$ by combining the candidates $I(\mathbf{q}_i)$'s, which is given by

$$\check{J}(\mathbf{v}_0^D) = \frac{\sum_{i=1}^{n(\mathbf{q}_0)} \psi(\mathbf{q}_0, \mathbf{q}_i) I(\mathbf{q}_i)}{\sum_{i=1}^{n(\mathbf{q}_0)} \psi(\mathbf{q}_0, \mathbf{q}_i)} \tag{24}$$

where

$$\psi(\mathbf{q}_0, \mathbf{q}_i) = e^{-\|\mathbf{v}_0^D - \mathbf{v}_i^R\|} \tag{25}$$



(a) (b) (c)

Fig. 4. Boundary region detection for the adaptive distortion estimation. White pixels depict boundary pixels, whereas black pixels represent smooth regions. (a) *Cafe*. (b) *Hall2*. (c) *Street*.

and $n(\mathbf{q}_0)$ denotes the number of employed candidates $I(\mathbf{q}_i)$'s. Finally, we estimate the view synthesis distortion at the distorted pixel position $\mathbf{v}_0^D$ as the squared difference $(\check{J}(\mathbf{v}_0^D) - \tilde{J}(\mathbf{v}_0^D))^2$ between the estimated true value $\check{J}(\mathbf{v}_0^D)$ and the reconstruction value $\tilde{J}(\mathbf{v}_0^D) = I(\mathbf{q}_0)$. Again notice that all terms in (22)–(25) are computed without the view synthesis.

### C. Computational Complexity of Distortion Estimation

We analyze the computational complexity of the proposed view synthesis distortion model. First, the depth-to-disparity conversion in (1) requires the complexity of $2N(C_a + C_m)$ for the distorted depth maps of the left and right views. Next, we should compute $\hat{J}(\mathbf{u}_0^R)$ in (17). When $\mathbf{d}^R(\mathbf{p}_0)$ and $\mathbf{d}^D(\mathbf{p}_0)$ are identical, there is no view synthesis distortion and we skip computing $\hat{J}(\mathbf{u}_0^R)$. Thus, we estimate $\hat{J}(\mathbf{u}_0^R)$ selectively for only $2\alpha N$ pixels in the left and right views. Here, $\alpha$ is the ratio of the pixels, whose reference disparities are different from distorted ones, to all pixels in a view.

The complexity for computing $\hat{J}(\mathbf{u}_0^R)$ is proportional to the number $m(\mathbf{p}_0)$ of candidate pixel values in (17). We control $m(\mathbf{p}_0)$ adaptively. In smooth regions, where the disparities of neighboring pixels are similar, a small $m(\mathbf{p}_0)$ is sufficient to estimate the reconstruction pixel values reliably. In contrast, near object boundaries, disparities tend to be irregular and a large $m(\mathbf{p}_0)$ is required. Thus, we first detect boundary regions that exhibit large gradient magnitudes of reference disparities. Then, we set $m(\mathbf{p}_0) = 3$ for the boundary pixels, and $m(\mathbf{p}_0) = 1$ otherwise.

To detect boundary regions, we first subsample a reference depth map with a sampling ratio of 1/8 in each

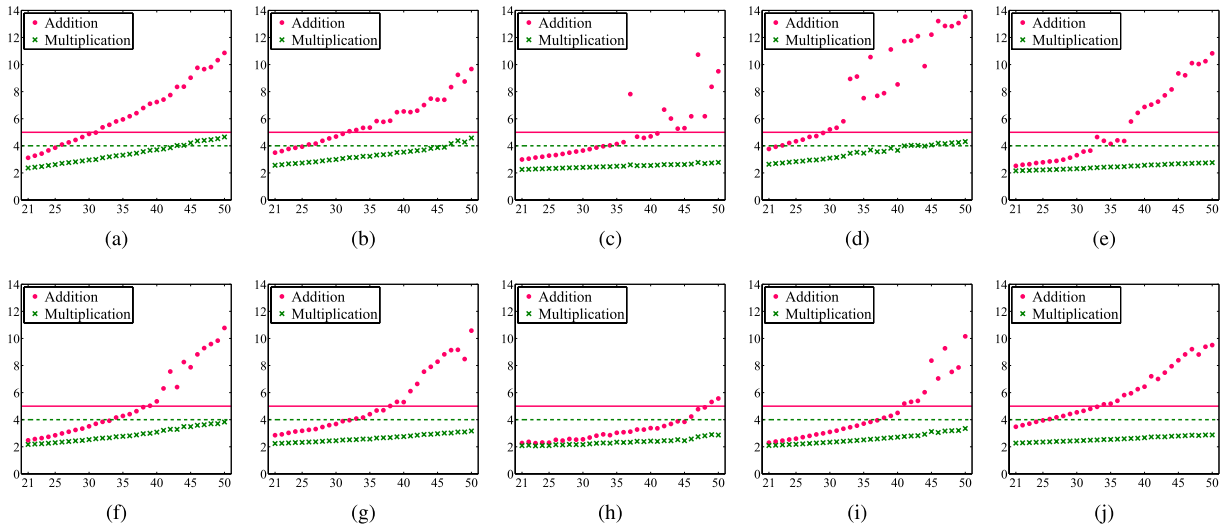



Fig. 5. Comparison of the computational complexities of the proposed algorithm and the simple view synthesis in terms of QP. The $x$-axis represents QPs and the $y$-axis denotes coefficient values. Dot and x marks are the coefficients $(2+8\alpha+32\alpha\beta)$ and $(2+16\alpha\beta)$ for the complexity $C_{\text{prop}}$ of the proposed algorithm, whereas solid and dashed lines depict the fixed coefficients 5 and 4 for the complexity $C_{\text{syn}}$ of the simple view synthesis scheme in Section III-A. (a) *Akko & Kayo*. (b) *Book Arrival* (c) *Cafe*. (d) *Champagne*. (e) *Hall2*. (f) *Kendo*. (g) *Lovebird1*. (h) *Mobile*. (i) *Panto*. (j) *Street*.

$x$- or $y$-direction. Then, we compute gradient magnitudes in the subsampled depth map using the Sobel operators. Then, we interpolate the gradient magnitude map to the original size and detect the boundary regions by thresholding. The subsampling/interpolation process is performed to detect pixels in the vicinity of sharp edges as boundary ones.

Fig. 4 shows detected boundary regions. Note that the boundary regions are small as compared with the whole image size. For each boundary pixel, we perform $6C_a$ to determine three candidate pixels recursively via (14) and use $4C_a$ and $4C_m$ for the weighted averaging in (17). For each nonboundary pixel, we need $2C_a$ only to find a single pixel value, which is directly utilized as $\hat{J}(\mathbf{u}_0^R)$. Consequently, the computational complexities to compute $\hat{J}(\mathbf{u}_0^R)$ for boundary pixels and nonboundary pixels in both views are $2\alpha\beta N(10C_a + 4C_m)$ and $2\alpha(1-\beta)N \cdot 2C_a$, respectively, where $\beta$ denotes the ratio of boundary pixels to the pixels, whose reference disparities are different from distorted ones.

We apply the same technique to set $n(\mathbf{q}_0)$ in (24). Hence, computing $\check{J}(\mathbf{v}_0^D)$ requires the same complexity as computing $\hat{J}(\mathbf{u}_0^R)$. To summarize, the proposed algorithm requires the overall complexity of

$$\begin{aligned} C_{\text{prop}} &= 2N(C_a + C_m) + 2\{2\alpha\beta N(10C_a + 4C_m) \\ &\qquad + 2\alpha(1-\beta)N \cdot 2C_a\} \\ &= (2 + 8\alpha + 32\alpha\beta)NC_a + (2 + 16\alpha\beta)NC_m \end{aligned} \tag{26}$$

to estimate the distortions at reference pixel positions and distorted pixel positions in a synthesized view.

The main objective of the proposed algorithm is to estimate the qualities of synthesized views, while encoding depth maps. The encoder typically compresses each depth map many times with various modes and QPs and chooses the best mode and QP to provide the optimal rate-distortion performance. It is essential to reduce the complexity of the mode decision or the rate-distortion optimization. In such a case, the proposed algorithm is useful since it can estimate view synthesis distortions efficiently without the actual view synthesis. In this case, the boundary region detection can be performed only once as a preprocessing step using reference disparities. Moreover, we empirically observe that the boundary detection consumes only 4%–8% of the total computational time of FDQM. Therefore, we ignore its complexity in (26). For the same reason, we also ignore the complexity of the reference depth-to-disparity conversion of the proposed algorithm in (26).

Fig. 5 compares the complexity $C_{\text{prop}}$ of the proposed algorithm in (26) with the complexity $C_{\text{syn}}$ of the simple view synthesis scheme in (4). Specifically, it plots the coefficients $(2 + 8\alpha + 32\alpha\beta)$ and $(2+16\alpha\beta)$ in (26), where $\alpha$ and $\beta$ are measured from each sequence. The coefficients $(2+8\alpha+32\alpha\beta)$ and $(2 + 16\alpha\beta)$ are $<5$ and 4, respectively, yielding $C_{\text{prop}} < C_{\text{syn}}$ for most QPs. The exceptions are high QPs, which are rarely used for depth compression. This indicates that the proposed algorithm can estimate view synthesis distortions with an even lower computational complexity than the simple view synthesis scheme in Section III-A. Moreover, the commonly used view synthesis method, VSRS [34], requires a significantly higher computational complexity than $C_{\text{syn}}$. By avoiding the sophisticated view synthesis, the proposed algorithm can reduce the huge computational burden. This is possible because we estimate view synthesis distortions adaptively at selected pixels only, assuming that disparities are locally consistent among neighboring pixels.

## IV. Quality Assessment for Depth Maps

We design an efficient quality metric, FDQM, for erroneous depth maps. Fig. 6 is the block diagram for the FDQM computation. First, we convert reference and distorted depth values into reference and distorted disparities. Second, we measure

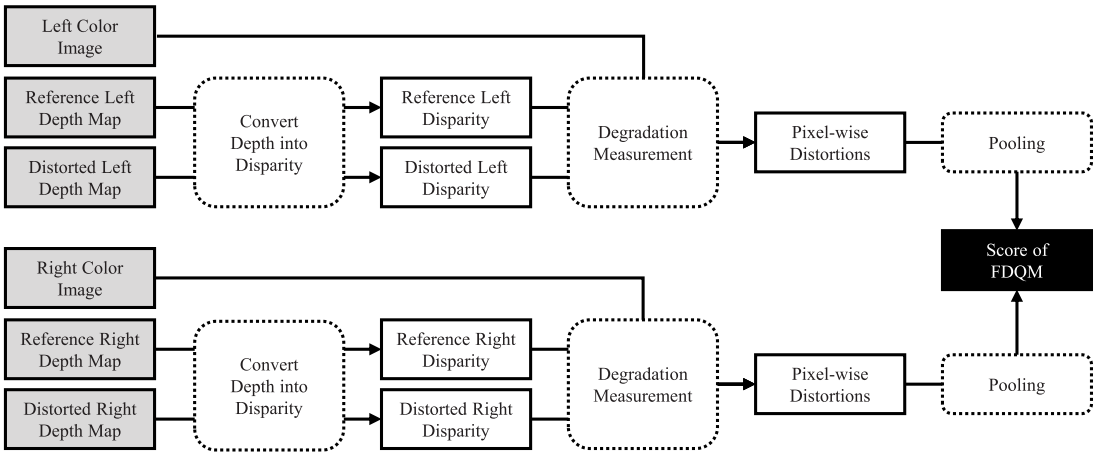



Fig. 6. Block diagram for the proposed FDQM computation.

the distortions of synthesized pixels, caused by disparity errors. Finally, we employ a spatial pooling scheme to integrate the pixel-wise distortions to yield an FDQM index.

### A. Pixel-Wise Distortions

For each $\mathbf{p}_0 \in \mathbf{P}$, we measure the distortion $\Phi^R(\mathbf{p}_0)$ at the reference pixel position $\mathbf{u}_0^R$ in the synthesized view as the normalized squared difference between the true pixel value $J(\mathbf{u}_0^R) = I(\mathbf{p}_0)$ and the estimated reconstruction value $\hat{J}(\mathbf{u}_0^R)$ in (17)

$$\Phi^R(\mathbf{p}_0) = \left(\frac{I(\mathbf{p}_0) - \hat{J}(\mathbf{u}_0^R)}{255}\right)^2. \tag{27}$$

Similarly, for each $\mathbf{q}_0 \in \mathbf{Q}$, we measure the distortion $\Phi^D(\mathbf{q}_0)$ at the distorted pixel position $\mathbf{v}_0^D$ as the difference between the reconstructed pixel value $\tilde{J}(\mathbf{v}_0^D) = I(\mathbf{q}_0)$ and the estimated true value $\check{J}(\mathbf{v}_0^D)$ in (24)

$$\Phi^D(\mathbf{q}_0) = \left(\frac{I(\mathbf{q}_0) - \check{J}(\mathbf{v}_0^D)}{255}\right)^2. \tag{28}$$

These pixel-wise distortions are measured from the left view and the right view, respectively, as shown in Fig. 6. Note that $0 \le \Phi^R(\mathbf{p}_0), \Phi^D(\mathbf{q}_0) \le 1$.

### B. Spatial Pooling

While the conventional metrics in [12]–[14] and [22]–[24] model HVS to consider perceptual attributes, FDQM is basically a pixel-wise distortion assessment scheme. The pixel-wise assessment may poorly correlate with HVS. To alleviate this problem, we attempt to reflect perceptual attributes of HVS by employing a simple spatial pooling technique.

Spatial pooling techniques [18], [35], [36] attempt to estimate visual qualities more faithfully by exploiting the property that HVS perceives pixel distortions in a spatially varying manner. However, the conventional pooling techniques can measure the qualities of color images only. We propose a spatial pooling scheme for the quality assessment of erroneous depth maps. In other words, we obtain a weighted sum of the pixel-wise distortions in (27) and (28). To determine spatially varying weights, we exploit the tendency that HVS is more attracted to object boundaries than to smooth regions. In general, object boundaries exhibit large gradient magnitudes in both color and disparity images. Thus, we use the disparity gradient map, computed in Section III-C. In addition, we adopt the same subsampling/interpolation process to obtain a color gradient map. Then, we increase pooling weights for pixels, which have bigger gradient magnitudes in color or disparity.

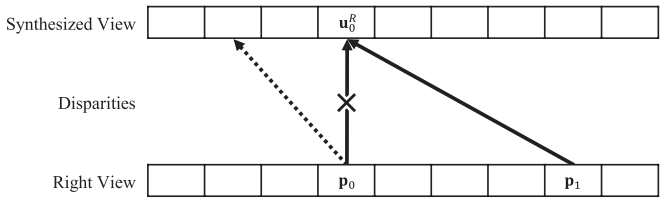
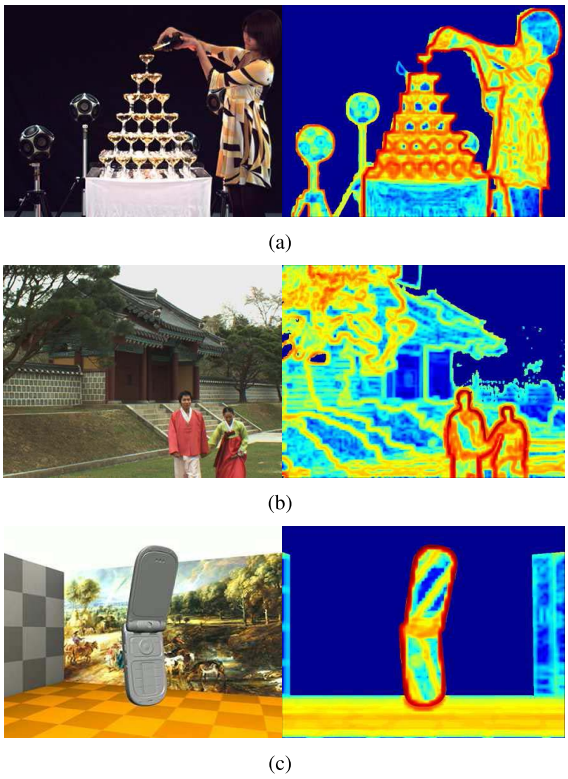



Fig. 7. Occlusion at pixel $\mathbf{u}_0^R$ in the synthesized view. Two pixels $\mathbf{p}_0$ and $\mathbf{p}_1$ in the right view are matched to the same pixel $\mathbf{u}_0^R$ by the disparities $\mathbf{d}^R(\mathbf{p}_0)$ and $\mathbf{d}^R(\mathbf{p}_1)$.

In addition, we consider occlusion effects in determining the pooling weights. Suppose that two pixels $\mathbf{p}_0$ and $\mathbf{p}_1$ in the right view are matched to the same pixel $\mathbf{u}_0^R$ in the synthesized view via reference disparities, as shown in Fig. 7. Let $d_x^R(\mathbf{p})$ be the horizontal component of a reference disparity $\mathbf{d}^R(\mathbf{p})$. In this example, since $|d_x^R(\mathbf{p}_0)| < |d_x^R(\mathbf{p}_1)|$, the synthesized color $\hat{J}(\mathbf{u}_0^R)$ should be determined by $I(\mathbf{p}_1)$ instead of $I(\mathbf{p}_0)$ due to the occlusion. This indicates that a bigger disparity contributes more to a synthesized view than a smaller disparity does in general. Therefore, we define a pooling weight $w^R(\mathbf{p})$ for the pixel-wise distortion $\Phi^R(\mathbf{p})$ at a reference pixel position $\mathbf{p} \in \mathbf{P}$ as

$$w^R(\mathbf{p}) = \left|d_x^R(\mathbf{p})\right|(\rho f(\mathbf{p}) + (1-\rho)g(\mathbf{p})) \tag{29}$$

where $f(\mathbf{p})$ and $g(\mathbf{p})$ denote the gradient magnitudes of the color and the reference disparity, respectively. In addition, $\rho$ controls the importance between $f(\mathbf{p})$ and $g(\mathbf{p})$. In this paper, $\rho$ is experimentally fixed to 0.1. Note that, for scaling purpose, $d_x^R(\mathbf{p})$, $f(\mathbf{p})$, and $g(\mathbf{p})$ are prenormalized into the



(a) (b) (c)

Fig. 8. Pooling weight maps, in which red and blue pixels represent large and small weights, respectively. (a) *Champagne*. (b) *Lovebird1*. (c) *Mobile*.

range of [0, 1], respectively. To summarize, in (29), a pixel is assigned a bigger weight, when it has a larger disparity and larger gradient magnitudes in the color and disparity images. Fig. 8 shows examples of pooling weights. We see that object boundaries at near distances are associated with large pooling weights.

Similarly, we obtain a weighted sum of the pixel-wise distortions at distorted pixel positions in (28). As in (29), we define a pooling weight $w^D(\mathbf{q})$ for $\mathbf{q} \in \mathbf{Q}$ as

$$w^D(\mathbf{q}) = \left|d_x^D(\mathbf{q})\right|(\rho f(\mathbf{q}) + (1-\rho)g(\mathbf{q})) \tag{30}$$

where $d_x^D(\mathbf{q})$ is the horizontal component of the distorted disparity $\mathbf{d}^D(\mathbf{q})$.

Finally, we integrate the pixel-wise distortions with the pooling weights into an overall distortion measure

$$\Omega(\mathbf{P},\mathbf{Q}) = \frac{\sum_{\mathbf{p}\in\mathbf{P}} w^R(\mathbf{p})\Phi^R(\mathbf{p}) + \sum\limits_{\mathbf{q}\in\mathbf{Q}} w^D(\mathbf{q})\Phi^D(\mathbf{q})}{\sum_{\mathbf{p}\in\mathbf{P}} w^R(\mathbf{p}) + \sum\limits_{\mathbf{q}\in\mathbf{Q}} w^D(\mathbf{q})} \tag{31}$$

where $0 \le \Omega(\mathbf{P},\mathbf{Q}) \le 1$.

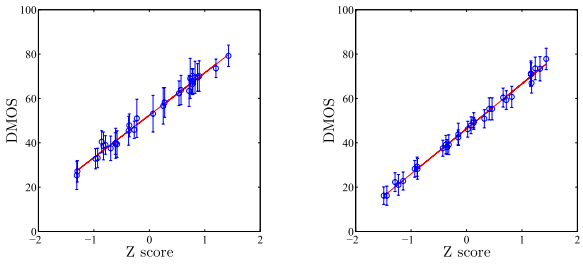
### C. FDQM

Let $\mathbf{P}_{\text{left}}$ and $\mathbf{P}_{\text{right}}$ denote the sets of $\mathbf{P}$ in the left and right views, and $\mathbf{Q}_{\text{left}}$ and $\mathbf{Q}_{\text{right}}$ denote the sets of $\mathbf{Q}$ in the left and right views, respectively. We measure the view synthesis distortions $\Omega(\mathbf{P}_{\text{left}}, \mathbf{Q}_{\text{left}})$ and $\Omega(\mathbf{P}_{\text{right}}, \mathbf{Q}_{\text{right}})$ from the erroneous depth maps for the left and right views, respectively. By representing the result in the decibel unit, we obtain the

TABLE I

PROPERTIES OF TEST MULTIVIEW SEQUENCES

| Sequence | Frame size | Employed frame index | Employed view indices | |
|---|---|---|---|---|
| | | | Left | Right |
| Akko & Kayo | 640 × 480 | 64 | 27 | 29 |
| Book Arrival | 1024 × 768 | 39 | 8 | 10 |
| Cafe | 1920 × 1080 | 1 | 1 | 5 |
| Champagne | 1280 × 960 | 1 | 37 | 41 |
| Hall2 | 1920 × 1088 | 198 | 5 | 7 |
| Kendo | 1024 × 768 | 196 | 1 | 5 |
| Lovebird1 | 1024 × 768 | 1 | 4 | 8 |
| Mobile | 720 × 540 | 1 | 3 | 7 |
| Pantomime | 1280 × 960 | 123 | 37 | 41 |
| Street | 1920 × 1088 | 1 | 5 | 3 |




(a) (b)

Fig. 9. Relation between $Z$ scores and DMOSs on (a) *Akko & Kayo* and (b) *Cafe*. For each QP, the averaged initial DMOS of all subjects is represented by a blue circle with its 95% confidence interval. The $x$-coordinate of a blue circle denotes the averaged $Z$ score. The red line represents the optimal fitting line between the $Z$ scores and the DMOSs.

FDQM score

$$\text{FDQM} = 10\log_{10}\left\{\frac{1}{\lambda\cdot\Omega(\mathbf{P}_{\text{right}},\mathbf{Q}_{\text{right}}) + (1-\lambda)\cdot\Omega(\mathbf{P}_{\text{left}},\mathbf{Q}_{\text{left}})}\right\} \tag{32}$$

where $\lambda$ specifies the location of the intermediate synthesized view between the left and right views. Notice that low distortions correspond to a high FDQM score.

## V. EXPERIMENTAL RESULTS

We evaluate the performance of the proposed FDQM using 10 multiview sequences: *Akko & Kayo*, *Book Arrival*, *Cafe*, *Champagne*, *Hall2*, *Kendo*, *Lovebird1*, *Mobile*, *Panto*, and *Street*. We use the provided depth maps of *Akko & Kayo*, *Book Arrival*, and *Kendo*, and generate depth maps for the other sequences using the depth estimation reference software [34] with the default configuration. Table I summarizes properties of the test sequences. The order of views in the *Street* sequence is reversed as compared with the other sequences. We assess the qualities of intermediate views at the center position between left and right views, i.e., $\lambda = 0.5$. We assume that color images are uncompressed and error-free, whereas depth maps are encoded using the 3-D video test model based on advanced video coding (3DV-ATM) reference software [37] in the intra mode with 30 different levels of QP from 21 to 50. Then, FDQM estimates the quality of each distorted intermediate view, synthesized from true color images and compressed depth maps, in comparison with the reference

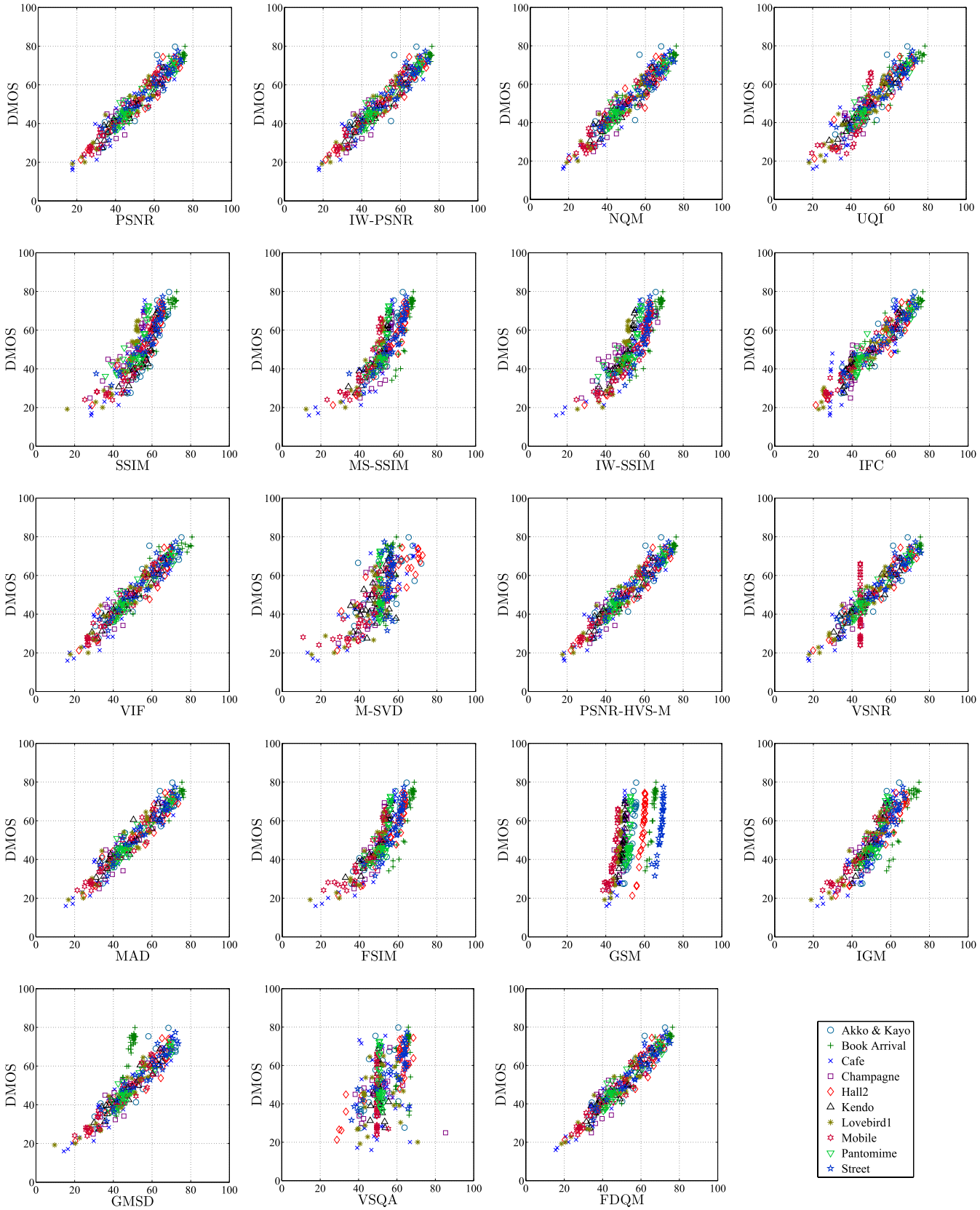



Fig. 10. Relation between the subjective scores (DMOSs) and the objective scores after regression.

intermediate view, synthesized from true color images and true depth maps. However, FDQM does not perform the actual view synthesis. On the contrary, the conventional quality metrics [16]–[20], [23], [25], [26] should synthesize intermediate views to assess their qualities. For the view synthesis, VSRS [34] is employed.

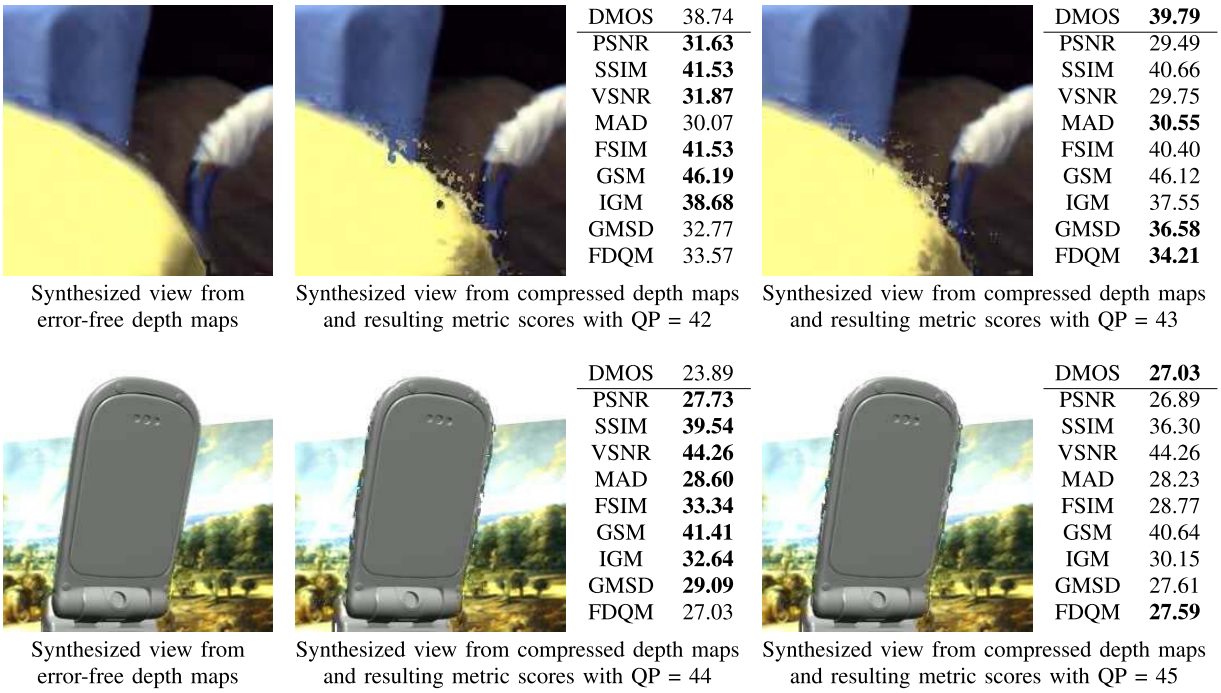



Fig. 11. Synthesized views with different QPs and their quality assessment results on *Cafe* (top) and *Mobile* (bottom). For each metric, the higher score between the two different QPs is typed in boldface.

To evaluate the performance of each quality metric, we compute the correlation between objective scores of the metric and subjective scores from human assessment. All experiments are performed on a standard PC with a 2.6-GHz quad core processor and 8-GB memory.

### A. Subjective Assessment

We use the simultaneous double stimulus for continuous evaluation method, recommended by the International Telecommunication Union - Radiocommunication (ITU-R) [38], which shows a reference image and a distorted image on a single display at the same time for subjective comparison. In other words, we show a reference synthesized view and a distorted synthesized view simultaneously, and human subjects judge the quality of the distorted view by selecting a score within the range $[0, 100]$. For unbiased assessment, we shuffle the orders of distorted views randomly. We divide the 10 test sequences into two sets, {*Akko & Kayo*, *Book Arrival*, *Hall2*, *Kendo*, *Pantomime*} and {*Cafe*, *Champagne*, *Lovebird1*, *Mobile*, *Street*}, which are tested, respectively, by 18 and 29 undergraduate or graduate students who are inexperienced with image quality assessment. Each subject assesses the qualities of 150 distorted views from the five test sequences with the 30 different QP levels. We limit the whole experimental time for each human subject to 30 min as recommended in [38]. The source code of FDQM, the test dataset, and the subjective scores are available at our project website.[1]

For each test with a given sequence and a given QP, we first remove outlier scores. A score is declared as an outlier when it is not within 1.15 standard deviations from the average score of all subjects. The scores after the outlier removal serve as initial difference mean opinion scores (DMOSs). In Fig. 9, the averaged initial DMOSs and 95% confidence intervals are shown as blue circles and bars, respectively. Thus, for each sequence, there are 30 blue circles corresponding to the 30 different QPs. However, note that a higher DMOS does not always correspond to a lower QP (a higher bit-rate). Next, to alleviate the bias of the subjects, we compute the standardized $Z$ score of an initial DMOS using the method in [39]. Specifically, for each subject, we compute the mean and standard deviation of the initial DMOSs across all QPs. We subtract the mean from an initial DMOS, and then divide it by the standard deviation. The $Z$ scores of all subjects for each QP are then averaged together to yield the averaged $Z$ score, which is represented by the $x$-coordinate of each blue circle in Fig. 9. We use the least squares method to find the best fitting line between the averaged initial DMOSs and the averaged $Z$ scores, which is depicted by the red line in Fig. 9. Last, for each averaged $Z$ score, we compute the fitted DMOS from the line equation, which is employed as the final standardized DMOS.

### B. Performance Comparison

We compare the performance of the proposed FDQM with 18 conventional quality metrics: PSNR, IW-PSNR [18], NQM [14], UQI [15], SSIM [16], MS-SSIM [17], IW-SSIM [18], IFC [19], VIF [20], M-SVD [21], PSNR-HVS-M [22], VSNR [23], MAD [24], FSIM [25], GSM [26], IGM [27], GMSD [28], and VSQA [29]. These conventional metrics are applied to intermediate views, synthesized by VSRS [34]. In contrast, our FDQM assesses the qualities of synthesized views without the actual view

[1] Available at: http://mcl.korea.ac.kr/projects/FDQM/.

TABLE II

COMPARISON OF SROCCS, PLCCS, AND RMSES BETWEEN THE SUBJECTIVE DMOSS AND THE OBJECTIVE METRIC SCORES. THE NUMBERS IN BRACKETS ARE THE RANKS OF THE METRICS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDFACE

| SROCC | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sequence | PSNR | IW-PSNR | NQM | UQI | SSIM | MS-SSIM | IW-SSIM | IFC | VIF | PSNR-HVS-M | VSNR | MAD | FSIM | GSM | FDQM |
| Akko & Kayo | 0.9359 (8) | 0.9119 (15) | 0.9168 (14) | 0.9395 (2) | 0.9382 (7) | **0.9430 (1)** | 0.9350 (9) | 0.9328 (10) | 0.9395 (2) | 0.9239 (12) | 0.9239 (12) | 0.9319 (11) | 0.9386 (5) | 0.9386 (5) | 0.9395 (2) |
| Book Arrival | 0.8834 (4) | **0.8874 (1)** | 0.8776 (12) | 0.8825 (7) | 0.8803 (10) | 0.8816 (9) | 0.8830 (6) | 0.8723 (13) | 0.8821 (8) | **0.8874 (1)** | 0.8647 (14) | 0.8625 (15) | 0.8870 (3) | 0.8790 (11) | 0.8834 (4) |
| Cafe | 0.9568 (7) | 0.9613 (3) | 0.9546 (10) | 0.9310 (14) | 0.9471 (12) | 0.9564 (8) | 0.9604 (4) | 0.9226 (15) | 0.9426 (13) | 0.9528 (11) | 0.9564 (8) | 0.9640 (2) | 0.9577 (6) | 0.9604 (4) | **0.9733 (1)** |
| Champagne | 0.8598 (13) | 0.8594 (14) | 0.8647 (7) | **0.8768 (1)** | 0.8643 (8) | 0.8607 (12) | 0.8656 (5) | 0.8572 (15) | 0.8661 (4) | 0.8638 (9) | 0.8705 (3) | 0.8759 (2) | 0.8630 (10) | 0.8656 (5) | 0.8630 (10) |
| Hall2 | 0.9315 (2) | 0.9275 (4) | 0.9061 (14) | 0.9150 (13) | 0.9235 (10) | 0.9221 (12) | 0.9239 (8) | 0.8812 (15) | 0.9226 (11) | 0.9279 (3) | 0.9270 (5) | 0.9270 (5) | 0.9244 (7) | **0.9328 (1)** | 0.9239 (8) |
| Kendo | 0.9773 (2) | 0.9702 (11) | 0.9617 (14) | 0.9769 (3) | 0.9746 (7) | 0.9702 (11) | 0.9733 (9) | 0.9519 (15) | 0.9755 (5) | 0.9769 (3) | **0.9782 (1)** | 0.9635 (13) | 0.9755 (5) | 0.9729 (10) | 0.9746 (7) |
| Lovebird1 | 0.9315 (12) | 0.9346 (5) | **0.9413 (1)** | 0.9341 (7) | 0.9359 (4) | 0.9390 (2) | 0.9333 (9) | 0.9390 (2) | 0.9341 (7) | 0.9297 (15) | 0.9315 (12) | 0.9333 (9) | 0.9310 (14) | 0.9346 (5) | 0.9319 (11) |
| Mobile | 0.9582 (5) | 0.9408 (12) | 0.9315 (14) | 0.9568 (6) | 0.9542 (8) | 0.9546 (7) | 0.9604 (3) | 0.9462 (10) | 0.9453 (11) | **0.9608 (1)** | -0.6774 (15) | 0.9600 (4) | 0.9524 (9) | 0.9333 (13) | **0.9608 (1)** |
| Pantomime | 0.8438 (11) | 0.8456 (10) | **0.8670 (1)** | 0.8194 (14) | 0.8505 (6) | 0.8523 (3) | 0.8523 (3) | 0.8145 (15) | 0.8474 (7) | 0.8474 (7) | 0.8429 (13) | 0.8509 (5) | 0.8541 (2) | 0.8438 (11) | 0.8474 (7) |
| Street | 0.9444 (7) | 0.9475 (2) | **0.9524 (1)** | 0.9426 (14) | 0.9444 (7) | 0.9453 (4) | 0.9444 (7) | 0.9395 (15) | 0.9453 (4) | 0.9444 (7) | 0.9444 (7) | 0.9448 (6) | 0.9444 (7) | 0.9444 (7) | 0.9466 (3) |
| Average | 0.9223 (5) | 0.9186 (11) | 0.9174 (13) | 0.9175 (12) | 0.9213 (8) | 0.9225 (4) | 0.9232 (2) | 0.9057 (14) | 0.9200 (10) | 0.9215 (6) | 0.7562 (15) | 0.9214 (7) | 0.9228 (3) | 0.9205 (9) | **0.9244 (1)** |
| PLCC | | | | | | | | | | | | | | | |
| Sequence | PSNR | IW-PSNR | NQM | UQI | SSIM | MS-SSIM | IW-SSIM | IFC | VIF | PSNR-HVS-M | VSNR | MAD | FSIM | GSM | FDQM |
| Akko & Kayo | 0.9455 (3) | 0.9221 (15) | 0.9238 (14) | 0.9326 (9) | 0.9395 (6) | 0.9267 (11) | 0.9320 (10) | 0.9414 (5) | 0.9418 (4) | 0.9379 (8) | 0.9393 (7) | **0.9535 (1)** | 0.9261 (12) | 0.9240 (13) | 0.9486 (2) |
| Book Arrival | 0.9604 (5) | 0.9626 (3) | 0.9541 (7) | 0.9360 (13) | 0.9406 (12) | 0.9486 (11) | 0.9523 (8) | 0.9272 (15) | 0.9287 (14) | 0.9626 (4) | **0.9690 (1)** | 0.9639 (2) | 0.9489 (10) | 0.9497 (9) | 0.9565 (6) |
| Cafe | 0.9592 (5) | 0.9618 (4) | 0.9671 (2) | 0.9385 (9) | 0.8694 (11) | 0.8250 (15) | 0.8534 (14) | 0.8586 (12) | 0.9495 (8) | 0.9571 (7) | 0.9590 (6) | 0.9656 (3) | 0.8787 (10) | 0.8557 (13) | **0.9729 (1)** |
| Champagne | 0.9037 (3) | 0.9010 (7) | 0.9011 (6) | 0.8912 (10) | 0.8101 (12) | 0.7922 (15) | **0.9073 (1)** | 0.8525 (11) | 0.8967 (8) | 0.9017 (5) | 0.9062 (2) | 0.9020 (4) | 0.7984 (13) | 0.7935 (14) | 0.8924 (9) |
| Hall2 | 0.9651 (3) | **0.9667 (1)** | 0.9475 (11) | 0.9485 (10) | 0.9514 (7) | 0.9361 (13) | 0.9304 (15) | 0.9510 (8) | 0.9488 (9) | 0.9664 (2) | 0.9651 (4) | 0.9550 (6) | 0.9355 (14) | 0.9428 (12) | 0.9599 (5) |
| Kendo | 0.9743 (2) | 0.9651 (7) | 0.9596 (8) | **0.9766 (1)** | 0.9206 (11) | 0.8875 (14) | 0.9029 (12) | 0.9444 (10) | 0.9727 (3) | 0.9679 (5) | 0.9686 (4) | 0.9565 (9) | 0.8966 (13) | 0.8855 (15) | 0.9659 (6) |
| Lovebird1 | 0.9542 (5) | 0.9559 (2) | 0.9518 (6) | 0.9473 (9) | 0.8794 (13) | 0.8545 (15) | 0.8642 (14) | **0.9642 (1)** | 0.9484 (7) | 0.9550 (3) | 0.9545 (4) | 0.9467 (10) | 0.8898 (12) | 0.8901 (11) | 0.9476 (8) |
| Mobile | 0.9628 (6) | 0.9595 (7) | 0.9511 (8) | 0.7878 (14) | 0.8863 (12) | 0.8859 (13) | 0.9179 (9) | **0.9693 (1)** | 0.9632 (5) | 0.9647 (4) | -0.8061 (15) | 0.9656 (2) | 0.9069 (10) | 0.8867 (11) | 0.9652 (3) |
| Pantomime | 0.9593 (2) | 0.9585 (4) | **0.9612 (1)** | 0.9438 (9) | 0.7589 (12) | 0.7506 (13) | 0.7657 (11) | 0.9366 (10) | 0.9558 (7) | 0.9591 (3) | 0.9580 (5) | 0.9572 (6) | 0.7476 (14) | 0.7302 (15) | 0.9550 (8) |
| Street | 0.9540 (5) | 0.9536 (6) | 0.9588 (2) | 0.9512 (9) | 0.9021 (13) | 0.8879 (14) | 0.8818 (15) | 0.9421 (10) | 0.9521 (8) | 0.9535 (7) | 0.9567 (3) | 0.9547 (4) | 0.9041 (12) | 0.9061 (11) | **0.9603 (1)** |
| Average | **0.9539 (1)** | 0.9507 (5) | 0.9476 (6) | 0.9253 (9) | 0.8858 (11) | 0.8695 (14) | 0.8908 (10) | 0.9287 (8) | 0.9458 (7) | 0.9526 (2) | 0.7770 (15) | 0.9521 (4) | 0.8833 (12) | 0.8764 (13) | 0.9524 (3) |
| RMSE | | | | | | | | | | | | | | | |
| Sequence | PSNR | IW-PSNR | NQM | UQI | SSIM | MS-SSIM | IW-SSIM | IFC | VIF | PSNR-HVS-M | VSNR | MAD | FSIM | GSM | FDQM |
| Akko & Kayo | 4.9014 (3) | 5.8254 (9) | 5.7628 (8) | 6.1253 (10) | 11.4761 (14) | 8.9757 (13) | 6.9470 (11) | 5.0778 (5) | 5.0616 (4) | 5.2225 (7) | 5.1625 (6) | **4.5364 (1)** | 7.5954 (12) | 13.1291 (15) | 4.7617 (2) |
| Book Arrival | 3.8336 (5) | 3.7285 (3) | 4.1217 (7) | 4.8445 (8) | 7.5056 (11) | 10.8969 (14) | 9.3811 (12) | 5.1555 (10) | 5.1065 (9) | 3.7314 (4) | **3.4027 (1)** | 3.6673 (2) | 10.3150 (13) | 12.2601 (15) | 4.0174 (6) |
| Cafe | 5.0717 (5) | 4.9082 (4) | 4.5606 (2) | 6.1941 (9) | 11.4366 (14) | 10.5511 (13) | 9.9062 (12) | 9.1936 (10) | 5.6260 (8) | 5.1947 (7) | 5.0816 (6) | 4.6653 (3) | 9.6352 (11) | 15.8528 (15) | **4.1453 (1)** |
| Champagne | 4.6545 (3) | 4.7163 (7) | 4.7139 (6) | 4.9309 (10) | 6.6002 (12) | 9.3388 (15) | **4.5710 (1)** | 5.6837 (11) | 4.8122 (8) | 4.7000 (5) | 4.5968 (2) | 4.6935 (4) | 8.0828 (13) | 9.0134 (14) | 4.9075 (9) |
| Hall2 | 3.7947 (3) | **3.7083 (1)** | 4.6323 (10) | 4.5902 (9) | 6.0549 (11) | 6.6569 (12) | 7.5317 (14) | 4.4788 (7) | 4.5746 (8) | 3.7225 (2) | 3.7953 (4) | 4.2990 (6) | 7.4460 (13) | 13.2576 (15) | 4.0636 (5) |
| Kendo | 2.5729 (2) | 2.9935 (7) | 3.2142 (8) | **2.4599 (1)** | 10.3986 (15) | 7.0219 (12) | 7.3496 (13) | 3.7582 (10) | 2.6505 (3) | 2.8729 (5) | 2.8419 (4) | 3.3357 (9) | 6.5153 (11) | 10.2766 (14) | 2.9612 (6) |
| Lovebird1 | 3.9919 (5) | 3.9170 (2) | 4.0914 (6) | 4.2749 (9) | 7.2753 (12) | 7.4588 (13) | 9.0680 (14) | **3.5362 (1)** | 4.2316 (7) | 3.9572 (3) | 3.9794 (4) | 4.2963 (10) | 6.6958 (11) | 11.6938 (15) | 4.2604 (8) |
| Mobile | 3.6524 (6) | 3.8102 (7) | 4.1745 (8) | 8.5832 (11) | 9.7808 (13) | 7.6611 (10) | 9.1042 (12) | **3.3261 (1)** | 3.6358 (5) | 3.5623 (4) | 13.7507 (15) | 3.5156 (2) | 6.2746 (9) | 11.7106 (14) | 3.5366 (3) |
| Pantomime | 3.2409 (2) | 3.2699 (4) | **3.1646 (1)** | 3.7935 (9) | 7.9274 (12) | 9.2854 (14) | 7.8702 (11) | 4.0200 (10) | 3.3732 (7) | 3.2474 (3) | 3.2895 (5) | 3.3202 (6) | 8.8203 (13) | 10.8107 (15) | 3.4019 (8) |
| Street | 3.6222 (5) | 3.6379 (6) | 3.4372 (2) | 3.7314 (9) | 5.6598 (11) | 6.6360 (12) | 8.6869 (14) | 4.0535 (10) | 3.6977 (8) | 3.6432 (7) | 3.5196 (3) | 3.5954 (4) | 7.8775 (13) | 15.3156 (15) | **3.3720 (1)** |
| Average | **3.9336 (1)** | 4.0515 (5) | 4.1873 (6) | 4.9528 (10) | 8.4115 (13) | 8.4483 (14) | 8.0416 (12) | 4.8284 (8) | 4.2770 (7) | 3.9854 (3) | 4.9420 (9) | 3.9925 (4) | 7.9258 (11) | 12.3320 (15) | 3.9428 (2) |

TABLE III

STATISTICAL LEFT-TAILED $F$-TEST RESULTS. THE 10 BITS IN EACH CELL INDICATE THE $F$-TEST RESULTS FOR THE TEN SEQUENCES *Akko & Kayo*, *Book Arrival*, *Cafe*, *Champagne*, *Hall2*, *Kendo*, *Lovebird1*, *Mobile*, *Pantomime*, AND *Street* IN ORDER. BIT 1 INDICATES THAT THE METRIC IN THE ASSOCIATED ROW IS SIGNIFICANTLY BETTER THAN THE METRIC IN THE ASSOCIATED COLUMN, WHEREAS BIT 0 MEANS NO SIGNIFICANT DIFFERENCE. $H$ COUNTS THE NUMBER OF 1S IN EACH ROW. THE HIGHEST $H$ COUNTS ARE HIGHLIGHTED IN BOLDFACE

| | PSNR | IW-PSNR | NQM | UQI | SSIM | MS-SSIM | IW-SSIM | IFC | VIF | PSNR-HVS-M | VSNR | MAD | FSIM | GSM | FDQM | $H$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PSNR | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 1111111111 | 1111111111 | 1110111111 | 0010010000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 1111111111 | 1111111111 | 0000000000 | 53(2) |
| IW-PSNR | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 1111111111 | 1111111111 | 0110111111 | 0110000000 | 0100000000 | 0000000000 | 0000000100 | 0000000000 | 0111111111 | 1111111111 | 0000000000 | 52(4) |
| NQM | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 1111011111 | 1111111111 | 0110111111 | 0010000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 0111111111 | 1111111111 | 0000000000 | 49(7) |
| UQI | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 1110011011 | 1111111011 | 0110111011 | 0010010000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 0111111011 | 1111111011 | 0000000000 | 43(10) |
| SSIM | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0100000000 | 0000000001 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 0100000001 | 1111111101 | 0000000000 | 14(11) |
| MS-SSIM | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 0000000000 | 1010111101 | 0000000000 | 8(14) |
| IW-SSIM | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 0001000000 | 0001000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 0001000000 | 1011110110 | 0000000000 | 12(12) |
| IFC | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 1100011111 | 1101111111 | 0100111111 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 1101111111 | 1111111111 | 0000000000 | 44(9) |
| VIF | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 1111011111 | 1111111111 | 0110111111 | 0010010000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 1111111111 | 1111111111 | 0000000000 | 51(6) |
| PSNR-HVS-M | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 1111111111 | 1111111111 | 0110111111 | 0110000000 | 0100000000 | 0000000000 | 0000000100 | 0000000000 | 1111111111 | 1111111111 | 0000000000 | 53(2) |
| VSNR | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0100000000 | 1111111011 | 1111111011 | 0110111011 | 0110000000 | 0100000000 | 0000000000 | 0000000000 | 0000000000 | 1111111011 | 1111111011 | 0000000000 | 47(8) |
| MAD | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 1111111111 | 1111111111 | 1110111111 | 0110000000 | 0100000000 | 0000000000 | 0000000100 | 0000000000 | 1111111111 | 1111111111 | 0000000000 | **54(1)** |
| FSIM | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 0000000000 | 1010111101 | 0000000000 | 9(13) |
| GSM | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0000000000 | 0(15) |
| FDQM | 0000000000 | 0000000000 | 0000000000 | 0010000100 | 1110111111 | 1111111111 | 1110111111 | 0010000000 | 0000000000 | 0000000000 | 0000000100 | 0000000000 | 1111111111 | 1111111111 | 0000000000 | 52(4) |

synthesis, requiring significantly less computations than the conventional metrics.

To evaluate the performance of each metric, we compute the correlation between the objective metric scores and the subjective standardized DMOSs, obtained in Section V-A, using three measures: Spearman rank order correlation coefficient (SROCC), Pearson linear correlation coefficient (PLCC), and root mean squared error (RMSE). SROCC computes the correlation between the DMOSs and the ranks of the objective scores. PLCC measures the correlation between the DMOSs and the objective scores that are transformed by a nonlinear regression technique [38], which makes the range of the scores equal to that of the DMOSs. For the transform, we use the five-parameter regression function in [40]. RMSE estimates the RMSE between the DMOSs and the transformed objective scores. Whereas higher correlation coefficients of SROCC and PLCC indicate better performance of an objective metric, a smaller RMSE corresponds to better performance.

TABLE IV

COMPUTATIONAL TIMES OF THE OBJECTIVE METRIC EVALUATION. THE NUMBERS IN BRACKETS DENOTE THE COMPUTATIONAL TIMES EXCLUDING THE VIEW SYNTHESIS. THE FASTEST COMPUTATIONAL TIMES ARE HIGHLIGHTED IN BOLDFACE. THE UNIT OF TIME IS SECOND

| Sequence | PSNR | IW-PSNR | NQM | UQI | SSIM | MS-SSIM | IW-SSIM | IFC | VIF | M-SVD | PSNR-HVS-M | VSNR | MAD | FSIM | GSM | IGM | GMSD | VSQA | FDQM |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Akko & Kayo | 1.3512 (0.0271) | 1.7035 (0.3793) | 1.6560 (0.3319) | 1.4053 (0.0812) | 1.4238 (0.0996) | 1.4645 (0.1403) | 1.7965 (0.4723) | 2.5318 (1.2076) | 2.5494 (1.2252) | 1.8867 (0.5625) | 4.9124 (3.5882) | 1.3931 (0.0690) | 3.0756 (1.7514) | 1.6770 (0.3528) | 1.3691 (0.0450) | 17.6712 (16.3470) | 1.3588 (0.0346) | 18.9182 (17.5940) | **0.1166** |
| Book Arrival | 3.2825 (0.0581) | 4.2606 (1.0362) | 4.0625 (0.8380) | 3.4255 (0.2011) | 3.4684 (0.2440) | 3.5635 (0.3391) | 4.4933 (1.2689) | 6.4874 (3.2630) | 6.5272 (3.3028) | 4.7317 (1.5073) | 12.3856 (9.1612) | 3.4049 (0.1805) | 7.6028 (4.3784) | 3.6910 (0.4665) | 3.3081 (0.0836) | 47.5014 (44.2770) | 3.3016 (0.0772) | 48.7494 (45.5250) | **0.2930** |
| Cafe | 15.8456 (0.1393) | 18.4897 (2.7835) | 18.0935 (2.3873) | 16.2284 (0.5222) | 16.3330 (0.6267) | 16.5721 (0.8659) | 19.1073 (3.4011) | 24.3125 (8.6063) | 24.4563 (8.7501) | 19.7538 (4.0476) | 39.7642 (24.0580) | 16.2257 (0.5195) | 27.3802 (11.6740) | 16.4776 (0.7714) | 15.9022 (0.1960) | 131.4362 (115.7300) | 15.8994 (0.1932) | 137.8962 (122.1900) | **0.7775** |
| Champagne | 5.5930 (0.0815) | 7.1270 (1.6155) | 6.8557 (1.3442) | 5.8110 (0.2995) | 5.8821 (0.3706) | 6.0262 (0.5147) | 7.4978 (1.9863) | 10.5821 (5.0706) | 10.6542 (5.1427) | 7.8579 (2.3464) | 19.8445 (14.3330) | 5.7911 (0.2796) | 12.4010 (6.8895) | 5.9625 (0.4510) | 5.6250 (0.1136) | 75.8905 (70.3790) | 5.6244 (0.1129) | 76.8045 (71.2930) | **0.5228** |
| Hall2 | 8.5823 (0.1365) | 11.2636 (2.8179) | 10.8731 (2.4274) | 8.9543 (0.5085) | 9.0718 (0.6261) | 9.3131 (0.8673) | 11.8850 (3.4393) | 17.1025 (8.6568) | 17.2259 (8.7802) | 12.5525 (4.1068) | 32.8567 (24.4110) | 8.9351 (0.4894) | 20.5087 (12.0630) | 9.3370 (0.8913) | 8.6365 (0.1908) | 124.2357 (115.7900) | 8.6324 (0.1866) | 129.7357 (121.2900) | **0.8201** |
| Kendo | 3.3063 (0.0579) | 4.2805 (1.0321) | 4.0899 (0.8415) | 3.4479 (0.1995) | 3.4924 (0.2440) | 3.5872 (0.3388) | 4.5163 (1.2679) | 6.5094 (3.2610) | 6.5506 (3.3022) | 4.7140 (1.4656) | 11.8599 (8.6115) | 3.4289 (0.1806) | 7.7570 (4.5086) | 3.7141 (0.4658) | 3.3320 (0.0837) | 46.8394 (43.5910) | 3.3255 (0.0771) | 48.7764 (45.5280) | **0.2952** |
| Lovebird1 | 3.5313 (0.0581) | 4.5050 (1.0318) | 4.3168 (0.8436) | 3.6707 (0.1974) | 3.7172 (0.2439) | 3.8120 (0.3388) | 4.7406 (1.2674) | 6.7340 (3.2608) | 6.7695 (3.2963) | 4.9368 (1.4636) | 12.2639 (8.7907) | 3.6536 (0.1803) | 7.8314 (4.3582) | 3.9411 (0.4678) | 3.5569 (0.0837) | 47.8372 (44.3640) | 3.5506 (0.0773) | 48.9982 (45.5250) | **0.3016** |
| Mobile | 1.7467 (0.0341) | 2.1948 (0.4822) | 2.1365 (0.4239) | 1.8155 (0.1029) | 1.8387 (0.1260) | 1.8877 (0.1751) | 2.3114 (0.5988) | 3.2416 (1.5290) | 3.2621 (1.5495) | 2.3979 (0.6853) | 5.8717 (4.1591) | 1.8041 (0.0915) | 3.9247 (2.2121) | 2.1663 (0.4537) | 1.7696 (0.0570) | 22.9606 (21.2480) | 1.7562 (0.0436) | 24.0446 (22.3320) | **0.1398** |
| Pantomime | 5.0845 (0.0819) | 6.6230 (1.6204) | 6.3249 (1.3223) | 5.3040 (0.3014) | 5.3725 (0.3699) | 5.5165 (0.5139) | 6.9924 (1.9898) | 10.0728 (5.0702) | 10.1493 (5.1467) | 7.3722 (2.3696) | 19.2336 (14.2310) | 5.2820 (0.2794) | 11.9241 (6.9215) | 5.4631 (0.4605) | 5.1141 (0.1115) | 75.3396 (70.3370) | 5.1136 (0.1110) | 76.2206 (71.2180) | **0.4563** |
| Street | 9.5767 (0.1372) | 12.2557 (2.8162) | 11.8673 (2.4278) | 9.9457 (0.5062) | 10.0653 (0.6258) | 10.3063 (0.8668) | 12.8781 (3.4386) | 18.0993 (8.6598) | 18.2163 (8.7768) | 13.4990 (4.0595) | 33.6035 (24.1640) | 9.9295 (0.4900) | 21.4455 (12.0060) | 10.3233 (0.8838) | 9.6301 (0.1906) | 127.6795 (118.2400) | 9.6265 (0.1870) | 130.6995 (121.2600) | **0.8435** |
| Average | 5.7900 (0.0812) | 7.2703 (1.5615) | 7.0276 (1.3188) | 6.0008 (0.2920) | 6.0665 (0.3577) | 6.2049 (0.4961) | 7.6219 (1.9130) | 10.5673 (4.8585) | 10.6361 (4.9273) | 7.9703 (2.2614) | 19.2596 (13.5508) | 5.9848 (0.2760) | 12.3851 (6.6763) | 6.2753 (0.5665) | 5.8244 (0.1155) | 71.7391 (66.0303) | 5.8189 (0.1101) | 74.0843 (68.3755) | **0.4566** |

Fig. 10 shows the regression results of the objective metric scores, and thus the amounts of the correlation between the objective scores and the subjective scores. FDQM exhibits strong correlation with the subjective DMOSs on all test sequences. In contrast, UQI, SSIM, MS-SSIM, IW-SSIM, IFC, M-SVD, VSNR, FSIM, GSM, IGM, GMSD, and VSQA yield less correlated scores with the subjective ones. For instance, SSIM, MS-SSIM, IW-SSIM, and FSIM yield curved, instead of linear, distributions. In addition, UQI and VSNR on the *Mobile* sequence, IGM and GMSD on the *Book Arrival* sequence, IFC and M-SVD on many sequences, and GSM and VSQA on all sequences provide almost vertical distributions, which indicate irrelevances of the objective scores to the subjective DMOSs. However, PSNR, IW-PSNR, NQM, VIF, PSNR-HVS-M, and MAD yield relatively good performance.

Table II compares the SROCC, PLCC, and RMSE results. Again, the proposed FDQM yields high correlation results, in terms of SROCC and PLCC, and small RMSEs, especially on *Akko & Kayo*, *Cafe*, *Mobile*, and *Street*. The performance of FDQM is relatively degraded on *Champagne*, *Lovebird1*, and *Pantomime*, which have complicated depth maps and exhibit high dynamic ranges of depth values. For these sequences, our assumption that neighboring pixels have similar disparities becomes less accurate. However, notice that FDQM is comparable with or better than the generally used quality metrics, such as PSNR and SSIM, on all sequences.

We also conduct the statistical left-tailed $F$-test [41] to the nonlinear regression residuals to compare two metrics. The test produces a binary value 1 when the first metric is significantly better than the second metric, and 0 otherwise. We set the significance level to 0.05 with 95% confidence. Table III shows the significance test results, where the first metrics and the second metrics are listed in the leftmost column and the topmost row, respectively. We observe that PSNR, IW-PSNR, NQM, IFC, VIF, PSNR-HVS-M, VSNR, MAD, and FDQM yield good performances and have little significant difference from one another. Note that these metrics outperform SSIM, MS-SSIM, IW-SSIM, FSIM, and GSM significantly.

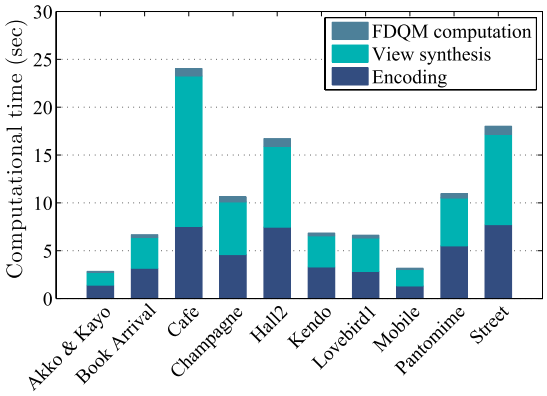



Fig. 12. Analysis of the computational times of FDQM computation, view synthesis and video encoding. For each sequence, the computational times are averaged over the 30 QPs.

Fig. 11 demonstrates the reliability of FDQM. These examples show that HVS may yield higher DMOSs on synthesized views with higher QPs (lower bitrates) than those with lower QPs (higher bitrates). In these examples, being consistent with the DMOSs, the proposed FDQM provides higher scores for the higher QPs. However, MAD and GSMD on *Mobile* and PSNR, SSIM, VSNR, FSIM, GSM, and IGM on both *Cafe* and *Mobile* provide lower scores for the higher QPs, which do not reflect the characteristics of HVS faithfully.

### C. Computational Complexity Comparison

Table IV compares the computational times for the quality assessment with the 30 different levels of QP. Again, the conventional metrics should synthesize each intermediate view, whereas FDQM estimates its quality without the view synthesis. Hence, we list the times including and excluding the view synthesis, respectively, for the conventional metrics. The conventional metrics are implemented in MATLAB, and the view synthesis software VSRS [34] is implemented in C++. The proposed FDQM is implemented in C++. Notice that FDQM is at least ten times faster than the conventional metrics, if we include their view synthesis times. These

results indicate that FDQM assesses the qualities of depth maps faithfully, while reducing the computational complexity significantly.

Fig. 12 shows the efficacy of FDQM by measuring the run-times of FDQM computation, view synthesis, and video encoding. In this test, the left and right depth maps are encoded at the 30 different QPs, and the computational times are averaged over all the QPs. Notice that the computational times of FDQM are much faster than those of both the encoding and view synthesis. Therefore, we can find an optimal QP using FDQM during video encoding, without too much computational overhead. On the other hand, suppose that we use conventional metrics, requiring the view synthesis, to find optimal QPs. Then, the view synthesis alone takes longer than the actual video encoding. This indicates that FDQM can be efficiently used in practical applications, especially for the rate-distortion optimized compression of depth maps.

## VI. Conclusion

In this paper, we proposed an FDQM for the quality assessment of depth maps in MVD applications. FDQM assesses how severely depth map errors degrade the qualities of synthesized intermediate views. However, for faster computation, FDQM avoids the actual view synthesis based on the local constancy assumption of disparities. First, FDQM estimates pixel-wise view synthesis distortions. Then, it integrates those pixel-wise distortions into an overall score with a spatial pooling scheme, which considers occlusion effects as well as HVS characteristics. Whereas the conventional metrics measure the distortion of an intermediate view after its synthesis, the proposed FDQM estimates the view synthesis distortion on the original view domain to save the complexity. Experimental results demonstrated that FDQM yields faithful assessment results, which are highly correlated to subjective scores, and also requires significantly less computations that the conventional metrics.

Future research issues include the fast quality assessment of temporally adjacent depth maps in video sequences and the view synthesis distortion modeling in a frequency domain.

## References


[1] M. Flierl and B. Girod, "Multiview video compression," *IEEE Signal Process. Mag.*, vol. 24, no. 6, pp. 66–76, Nov. 2007.

[2] *Multi-View Video Plus Depth (MVD) Format for Advanced 3D Video Systems*, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG document JVT-W100, Apr. 2007.

[3] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," *Proc. SPIE*, vol. 5291, pp. 93–104, May 2004.

[4] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map coding with distortion estimation of rendered view," *Proc. SPIE*, vol. 7543, pp. 75430B-1–75430B-10, Jan. 2010.

[5] Q. Zhang, P. An, Y. Zhang, and Z. Zhang, "Efficient rendering distortion estimation for depth map compression," in *Proc. 18th IEEE ICIP*, Sep. 2011, pp. 1105–1108.

[6] T.-Y. Chung, W.-D. Jang, and C.-S. Kim, "Efficient depth video coding based on view synthesis distortion estimation," in *Proc. IEEE VCIP*, Nov. 2012, pp. 1–4.

[7] T.-Y. Chung, J.-Y. Sim, and C.-S. Kim, "Bit allocation algorithm with novel view synthesis distortion model for multiview video plus depth coding," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3254–3267, Aug. 2014.

[8] Z. Wang and A. C. Bovik, *Modern Image Quality Assessment*. San Rafael, CA, USA: Morgan & Claypool, 2006.

[9] W. Lin and C.-C. J. Kuo, "Perceptual visual quality metrics: A survey," *J. Vis. Commun. Image Rerpresent.*, vol. 22, no. 4, pp. 297–312, May 2011.

[10] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "A comprehensive evaluation of full reference image quality assessment algorithms," in *Proc. 19th IEEE ICIP*, Sep./Oct. 2012, pp. 1477–1480.

[11] B. Girod, "What's wrong with mean-squared error?" in *Digital Images and Human Vision*, A. B. Watson, Ed. Cambridge, MA, USA: MIT Press, 1993, pp. 207–220.

[12] P. C. Teo and D. J. Heeger, "Perceptual image distortion," *Proc. SPIE*, vol. 2179, pp. 127–141, May 1994.

[13] Y.-K. Lai and C.-C. J. Kuo, "Image quality measurement using the Haar wavelet," *Proc. SPIE*, vol. 3169, pp. 127–138, Oct. 1997.

[14] N. Damera-Venkata, T. D. Kite, W. S. Geisler, B. L. Evans, and A. C. Bovik, "Image quality assessment based on a degradation model," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 636–650, Apr. 2000.

[15] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Process. Lett.*, vol. 9, no. 3, pp. 81–84, Mar. 2002.

[16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[17] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. Conf. Rec. 37th IEEE Asilomar Conf. Signals, Syst., Comput.*, vol. 2. Nov. 2003, pp. 1398–1402.

[18] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1185–1198, May 2011.

[19] H. R. Sheikh, A. C. Bovik, and G. de Veciana, "An information fidelity criterion for image quality assessment using natural scene statistics," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2117–2128, Dec. 2005.

[20] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[21] A. Shnayderman, A. Gusev, and A. M. Eskicioglu, "An SVD-based grayscale image quality measure for local and global assessment," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 422–429, Feb. 2006.

[22] N. Ponomarenko, F. Silvestri, K. Egiazarian, M. Carli, J. Astola, and V. Lukin, "On between-coefficient contrast masking of DCT basis functions," in *Proc. 3rd Int. Workshop Video Process. Quality Metrics Consum. Electron.*, 2007, pp. 1–4.

[23] D. M. Chandler and S. S. Hemami, "VSNR: A wavelet-based visual signal-to-noise ratio for natural images," *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2284–2298, Sep. 2007.

[24] E. C. Larson and D. M. Chandler, "Most apparent distortion: Full-reference image quality assessment and the role of strategy," *J. Electron. Imag.*, vol. 19, no. 1, pp. 011006-1–011006-21, Jan. 2010.

[25] L. Zhang, D. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.

[26] A. Liu, W. Lin, and M. Narwaria, "Image quality assessment based on gradient similarity," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1500–1512, Apr. 2012.

[27] J. Wu, W. Lin, G. Shi, and A. Liu, "Perceptual quality metric with internal generative mechanism," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 43–54, Jan. 2013.

[28] W. Xue, L. Zhang, X. Mou, and A. C. Bovik, "Gradient magnitude similarity deviation: A highly efficient perceptual image quality index," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 684–695, Feb. 2014.

[29] P.-H. Conze, P. Robert, and L. Morin, "Objective view synthesis quality assessment," *Proc. SPIE*, vol. 8288, pp. 82881M-1–82881M-14, Feb. 2012.

[30] A. Benoit, P. Le Callet, P. Campisi, and R. Cousseau, "Using disparity for quality assessment of stereoscopic images," in *Proc. 15th IEEE ICIP*, Oct. 2008, pp. 389–392.

[31] V. De Silva, H. K. Arachchi, E. Ekmekcioglu, and A. Kondoz, "Toward an impairment metric for stereoscopic video: A full-reference video quality metric to assess compressed stereoscopic video," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3392–3404, Sep. 2013.

[32] C. T. E. R. Hewage, S. T. Worrall, S. Dogan, S. Villette, and A. M. Kondoz, "Quality evaluation of color plus depth map-based stereoscopic video," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 2, pp. 304–318, Apr. 2009.

[33] P. Joveluro, H. Malekmohamadi, W. A. C. Fernando, and A. M. Kondoz, "Perceptual video quality metric for 3D video quality assessment," in *Proc. 3DTV-Conf., True Vis.-Capture, Transmiss., Display 3D Video*, Jun. 2010, pp. 1–4.

[34] *Reference Softwares for Depth Estimation and View Synthesis*, ISO/IEC JTC1/SC29/WG11 document M15377, Apr. 2008.
[35] Z. Wang and X. Shang, "Spatial pooling strategies for perceptual image quality assessment," in *Proc. IEEE ICIP*, Oct. 2006, pp. 2945–2948.
[36] A. K. Moorthy and A. C. Bovik, "Visual importance pooling for image quality assessment," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 2, pp. 193–201, Apr. 2009.
[37] *Reference Software for Multiview Video Coding*, ISO/IEC JTC1/SC29/WG11 document N10704, Jul. 2009.
[38] *Methodology for the Subjective Assessment of the Quality of Television Pictures*, ITU-R Rec. document BT.500-13, Jan. 2012.
[39] A. M. van Dijk, J.-B. Martens, and A. B. Watson, "Quality asessment of coded images using numerical category scaling," *Proc. SPIE*, vol. 2451, pp. 90–101, Feb. 1995.
[40] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.
[41] R. A. Fisher, *Statistical Methods for Research Workers*. London, U.K.: Oliver & Boyd, 1925.

**Won-Dong Jang** (S'13) received thev B.S. degree in electrical engineering from Korea University, Seoul, Korea, in 2011, where he is currently working toward the Ph.D. degree in electrical engineering.

His research interests include image quality assessment, image segmentation, and video understanding.

**Tae-Young Chung** (S'08–M'14) received the B.S. and Ph.D. degrees from the School of Electrical Engineering, Korea University, Seoul, Korea, in 2006 and 2013, respectively.

He is with the Software Research and Development Center, Samsung Electronics Company, Ltd., Suwon, Korea. His research interests include error resilient coding, multiview video coding, stereo matching, and computer vision.

**Jae-Young Sim** (S'02–M'06) received the B.S. degree in electrical engineering and the M.S. and Ph.D. degrees in electrical engineering and computer science from Seoul National University, Seoul, Korea, in 1999, 2001, and 2005, respectively.

He was a Research Staff Member with the Samsung Advanced Institute of Technology, Samsung Electronics Company, Ltd., Yongin, Korea, from 2005 to 2009. In 2009, he joined the School of Electrical and Computer Engineering, Ulsan National Institute of Science and Technology, Ulsan, Korea, where he is currently an Associate Professor. His research interests include image and 3-D visual signal processing, multimedia data compression, and computer vision.

**Chang-Su Kim** (S'95–M'01–SM'05) received the Ph.D. degree in electrical engineering from Seoul National University (SNU), Seoul, Korea.

He was a Visiting Scholar with the Signal and Image Processing Institute, University of Southern California, Los Angeles, CA, USA, from 2000 to 2001. From 2001 to 2003, he coordinated the 3-D Data Compression Group, National Research Laboratory for 3-D Visual Information Processing, SNU. From 2003 and 2005, he was an Assistant Professor with the Department of Information Engineering, Chinese University of Hong Kong, Hong Kong. In 2005, he joined the School of Electrical Engineering, Korea University, Seoul, where he is currently a Professor. He has authored over 210 technical papers in international journals and conferences. His research interests include image processing and computer vision.

Dr. Kim received the Distinguished Dissertation Award from SNU in 2000, the IEEK/IEEE Joint Award for Young IT Engineer of the Year in 2009, and the Best Paper Award from *Journal of Visual Communication and Image Representation* in 2014. He is an Editorial Board Member of *Journal of Visual Communication and Image Representation* and an Associate Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING.