CONVEXITY CHARACTERIZATION OF VIRTUAL VIEW RECONSTRUCTION ERROR IN MULTI-VIEW IMAGING

Vladan Velisavljevic[†], *Camilo Dorea*[◇], *Jacob Chakareski*[⋆], *Ricardo de Queiroz*[◇]
[†]School CST, University of Bedfordshire, Luton, UK
[◇]Dept. of Computer Science, University of Brasilia, Brasilia, Brazil
[⋆]Dept. of Electrical and Computer Engineering, University of Alabama, Tuscaloosa, AL, USA

ABSTRACT

Virtual view synthesis is a key component of multi-view imaging systems that enable visual immersion environments for emerging applications, e.g., virtual reality and 360-degree video. Using a small collection of captured reference viewpoints, this technique reconstructs any view of a remote scene of interest navigated by a user, to enhance the perceived immersion experience. We carry out a convexity characterization analysis of the virtual view reconstruction error that is caused by compression of the captured multi-view content. This error is expressed as a function of the virtual viewpoint coordinate relative to the captured reference viewpoints. We derive fundamental insights about the nature of this dependency and formulate a prediction framework that is able to accurately predict the specific dependency shape, convex or concave, for given reference views, multi-view content and compression settings. We are able to integrate our analysis into a proof-of-concept coding framework and demonstrate considerable benefits over a baseline approach.

Index Terms— Multi-view Imaging, Virtual View Synthesis, Depth Image Based Reconstruction

1. INTRODUCTION

We are entering an era of transformational changes in digital content consumption and experience, spurred by advances in imaging and cyber-physical/human systems. Emerging technologies such as virtual/augmented reality [1], 360-degree video [2], plenoptic cameras [3], and multi-view imaging [4] are enabling the design of novel applications that immerse us into volumetric visual representations that we can actively explore, navigate, and interact with. Thus, the (flat/2D and passive/remote) digital media experience, as we know it (for some time now), will never be the same. Simultaneously, these transformative changes are driving innovation across our society, by helping introduce diverse applications with impact on education and training, health-care, telecommuting, etc. Many further advances are expected on the road to immersive communication [5].

The work of J. Chakareski was supported by NSF award CCF-1528030.
978-1-5090-3649-3/17/$31.00 ©2017 IEEE

Emerging applications for reconstructing remote environments for volumetric visual immersion such as virtual reality and 360-degree video rely on multi-view imaging and virtual view synthesis [6] to enable such experiences. In particular, using a small collection of captured viewpoints, virtual view synthesis reconstructs any viewpoint of a remote scene of interest navigated by a user, to enhance his sensation of remote immersion. It is critical to have an understanding of the reconstruction error (or fidelity) of viewpoints synthesized thereby, to assess the quality of experience delivered to the user. Such knowledge can enable development of efficient resource allocation strategies. Bit allocation among captured viewpoints may, for instance, be used to guarantee a minimal quality of service or to optimize a global quality metric among multiple users. The problem is challenging due to the complex interdependencies that arise in this context between the fidelity of the captured data and the relative position of the virtual views in the aggregate view space navigated by the user.

View synthesis reconstruction error within multi-view plus depth coding systems has been reported [6] as a *concave* curve in which the mean-square error (MSE) as a function of virtual viewpoint position reaches a maximum at the virtual viewpoint farthest from the reference captured viewpoints used in its synthesis. Nevertheless, coding conditions imposed upon reference images can significantly alter MSE behavior, to the point where the expected concave shape becomes *convex* as shown in Fig. 1.

Understanding and modeling such synthesized view distortion is essential to numerous multi-view video compression and streaming applications. For instance, [8] considers multi-view multicast, where the captured video and depth signals are encoded using the scalable video coding standard H.264/SVC [9], and each client is served two reference video and depth signals. A linear synthesis view distortion model is borrowed from [10] and its coefficients are estimated from data. Furthermore, in [11], joint source-channel coding for multi-view video multi-cast has been studied, while [12] investigates user-action driven view and rate scalable multi-view coding. Both of these studies leverage an earlier cubic synthesized view distortion model derived in [13]. Similarly, a related distortion model has been pursued in [14]. Char-

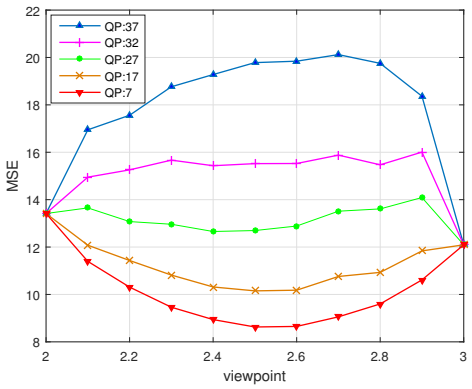


Fig. 1. MSE of virtual views at various viewpoint positions synthesized from coded reference images 2 and 3 of Ballet [7]. *Concave* curvature, when depth images are compressed at QP:37, becomes *convex* as depth compression transitions to QP:7. All texture images were compressed at QP:37 and MSE is relative to synthesis from uncompressed reference images.

acterizing the rate-distortion trade-offs of view synthesis has been also studied in [15]. Another linear distortion model has been examined in [16], comprising three terms: video coding-induced distortion, depth quantization-induced distortion, and inherent geometry distortion. Its practicality is, however, restricted by its high complexity. The impact of depth-map coding on virtual view synthesis is studied in [17], but the impact of texture coding has not been integrated into the analysis. Recently, in [18] a linear distortion model has also been adopted to optimally allocate bit-rate in response to viewer attention.

Linear modeling constitutes a simple, yet coarse approximation to reconstruction errors such as those of Fig. 1. More complex cubic models, developed within aforementioned related work, have also failed to adequately capture the degree of variation present in synthesized view distortions. Towards this goal, we carry out an analysis of the reconstruction error of a virtual viewpoint as a function of its relative position with respect to the reference captured viewpoints used to synthesize it. We derive fundamental insights about the nature of this dependency and formulate a prediction framework that is able to accurately predict the specific dependency shape, convex or concave, for given viewpoints, multi-view content and compression settings. In bit allocation strategies aimed at optimizing delivered quality in real time, convexity characterization can serve to determine maximum MSE, in the convex case, or estimate viewpoint position of such maximum in a computationally-efficient manner, without explicit virtual view synthesis. In this sense, it offers improved precision and speed over prior modeling approaches. Furthermore, once we establish the convexity mode, quick analytical characterization of the distortion dependency can be established, for more advanced bit allocation analysis, inclusive of having diverse objective functions. On the other hand, without such a facility, optimal bit allocation would need to explore a complex discrete problem that would be computationally expensive and prohibit real-time operation. We are able to integrate our analysis into a proof-of-concept coding framework and demonstrate considerable benefits over a baseline approach. Our preliminary results are very promising and motivate further investigation.

2. SYNTHESIZED VIEW DISTORTION MODEL

Denote the captured texture and depth images of the left reference view 0 as t_0 and d_0. Similarly, denote the same for the right reference view 1 as t_1 and d_1. A virtual view t_v is obtained by warping the reference views 0 and 1 to the viewpoint $0 \leq v \leq 1$ using DIBR [19] techniques to generate two projections t'_0 and t'_1. Warping ensures that the pixel n of the virtual view t_v, denoted as $t_v(n)$, is obtained by enforcing a disparity shift exploiting the information captured in the depth images so that $t'_0(n) = t_0(n_0 - vd_0(n_0))$ and $t'_1(n) = t_1(n_1 + (1-v)d_1(n_1))$, where n_0 and n_1 are the corresponding pixel coordinates in the left and right reference views, respectively. However, due to various constraints in realistic circumstances such as occlusion and rounding to the integer pixel coordinates, these two projections are not perfectly identical. For that reason, the pixels in the virtual view are blended using the following relation [20]:

$$t_v(n) = \begin{cases} (1-v)t'_0(n) + vt'_1(n) & t'_0(n), t'_1(n) \neq 0, \\ t'_0(n) & t'_1(n) = 0, \\ t'_1(n) & t'_0(n) = 0, \\ 0 & t'_0(n) = t'_1(n) = 0 \end{cases}, \quad (1)$$

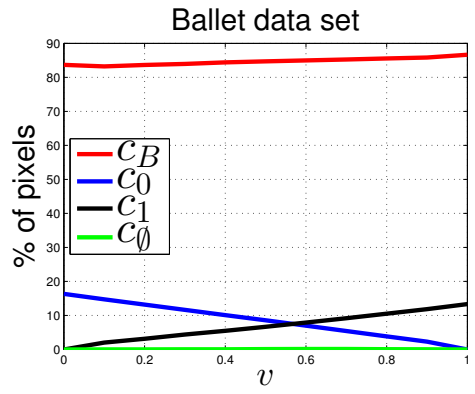
where $t'_0(n) = 0$ and $t'_1(n) = 0$ represent unavailability of the respective pixel from the left or right reference.

Encoding the texture and depth reference images impacts the quality of the virtual view obtained by (1). To estimate the MSE of the encoded virtual view, denote the encoded versions of t_0, d_0, t_1 and d_1 as $\tilde{t}_0$, $\tilde{d}_0$, $\tilde{t}_1$ and $\tilde{d}_1$, respectively. Furthermore, denote the encoded versions of the warped reference textures as $\tilde{t}'_0(\tilde{n}) = \tilde{t}_0(n_0 - v\tilde{d}_0(n_0))$ and $\tilde{t}'_1(\tilde{n}) = \tilde{t}_1(n_1 + (1-v)\tilde{d}_1(n_1))$. Consequently, denote the virtual view (1) obtained using encoded components as $\tilde{t}_v(\tilde{n})$.

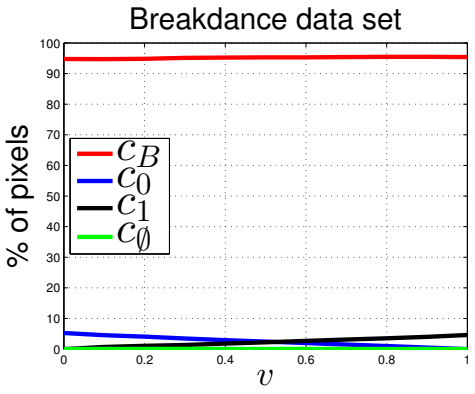
Now, the MSE of the virtual view can be expressed as

$$D_v = E[(\tilde{t}_v(\tilde{n}) - t_v(n))^2]. \quad (2)$$

To estimate the contribution of the 4 cases in (1) to the MSE in (2), denote the respective proportions of pixels synthesized using each case as $c_B(v)$, $c_0(v)$, $c_1(v)$ and $c_\emptyset(v)$, where the sum $c_B(v) + c_0(v) + c_1(v) + c_\emptyset(v) = 1$ is constant across the viewpoint coordinate v. Note that $c_0(v=1) = c_1(v=0) = 0$ because all pixels from the reference views are available at the particular reference viewpoints. Assuming the reference viewpoints are close enough so that occlusion or holes in 3D



(a)



(b)

Fig. 2. Captured numbers of pixels that contribute to virtual views across the viewpoint coordinate $0 \leq v \leq 1$ for the 4 cases in (1): (a) Ballet ($C_0 = 16.34\%$, $C_1 = 13.35\%$), (b) Breakdancers ($C_0 = 5.24\%$, $C_1 = 4.62\%$).

objects do not affect significantly these numbers, we adopt linear models for $c_0(v)$, $c_1(v)$ and we neglect $c_\emptyset(v)$, i.e.,

$$\begin{aligned} c_0(v) &= C_0 \cdot (1 - v), \\ c_1(v) &= C_1 \cdot v, \\ c_\emptyset(v) &= 0, \end{aligned} \tag{3}$$

where the constants C_0 and C_1 can be easily estimated as $C_0 = c_0(v = 0)$ and $C_1 = c_1(v = 1)$. Note that the number $c_B(v)$ is also linear in v, that is, $c_B(v) = 1 - C_0 + (C_0 - C_1)v$. This model is verified in our experimentation for several data sets and various encoding settings. An example is demonstrated in Fig. 2, where the number of pixels is plotted against the viewpoint coordinate v for two data sets. Furthermore, the model remains accurate for a range of encoding rates with the parameters C_0 and C_1 unchanged.

To simplify estimation of the MSE from (2), the error can be split into 3 terms as

$$\begin{aligned} D_v &= E[(\tilde{t}_v(\tilde{n}) - t_v(\tilde{n}) + t_v(\tilde{n}) - t_v(n))^2] = \\ &= E[(\tilde{t}_v(\tilde{n}) - t_v(\tilde{n}))^2] + E[(t_v(\tilde{n}) - t_v(n))^2] + \\ &+ 2E[(\tilde{t}_v(\tilde{n}) - t_v(\tilde{n}))(t_v(\tilde{n}) - t_v(n))]. \end{aligned} \tag{4}$$

The first term of (4), $E[(\tilde{t}_v(\tilde{n}) - t_v(\tilde{n}))^2]$, represents the error caused by encoding only the texture pixel intensities, while the pixel coordinate $\tilde{n}$ remains the same. Hence, the contribution of this term to D_v comes from the three cases in (1) that are weighted by $c_B(v)$, $c_0(v)$ and $c_1(v)$,[1] respectively. As a result, this term consists of three components factored by the MSE of the left and right reference texture images, D_{t_0} and D_{t_1}, respectively, and by the cross-correlation of the encoding errors $E_{t_0,t_1} = E[(\tilde{t}'_0(\tilde{n}) - t'_0(\tilde{n}))(\tilde{t}'_1(\tilde{n}) - t'_1(\tilde{n}))]$. Note that, in the modeling, we assume that neither the pixel coordinate error caused by encoding the reference depth images nor warping the pixels influences the encoding MSE of the reference texture images and, thus, $E[(\tilde{t}'_0(\tilde{n}) - t'_0(\tilde{n}))^2] = E[(\tilde{t}_0(n) - t_0(n))^2] = D_{t_0}$ and similarly for D_{t_1}.

The second term of (4), $E[(t_v(\tilde{n}) - t_v(n))^2]$, corresponds to the error introduced by displacement of the synthesized pixels in the virtual view owing to the encoding error of the reference depth images. Similarly to [14], we estimate this error by assuming the Gauss-Markov model for texture pixels, that is, $t(n + 1) = \rho t(n) + \omega(n)$, where ρ is the correlation across neighbor pixels, $\omega(n)$ the Gaussian noise with the variance $\sigma^2 = (1 - \rho^2)\sigma_t^2$ and $\sigma_t^2 = E[t^2(n)]$ is the mean energy of the texture image. It follows from this model that the absolute difference $|t(\tilde{n}) - t(n)|$ is given as

$$|t(\tilde{n}) - t(n)| = (\rho^{|\tilde{n}-n|} - 1)t(n) + \sum_{m=0}^{|\tilde{n}-n|-1} \rho^m \omega(\tilde{n} - m). \tag{5}$$

Since, in natural images, the correlation $0 \leq \rho \leq 1$ is nearly 1, we use the approximation $\rho^n \approx 1 - (1 - \rho)n$. Hence, applying this to (5) and accumulating the influence of the three cases weighted by $c_B(v)$, $c_0(v)$ and $c_1(v)$, the second term of (4) is derived as the sum of three components factored by the mean-absolute error of the left and right reference depth images, A_{d_0} and A_{d_1}, respectively, and the error cross-correlation $E_{d_0,d_1} = E[(\tilde{d}_0(n_0) - d_0(n_0))(\tilde{d}_1(n_1) - d_1(n_1))]$.

Finally, the third term of (4) consists of the mean product of the two error components. Owing to the fact that both components are zero mean and because of the nonzero shift between $t_v(\tilde{n})$ and $t_v(n)$, this term is neglected in the sequel. Note that, for similar reasons, the components E_{t_0,t_1} and E_{d_0,d_1} from the first and second terms are also neglected.

3. CONVEXITY PREDICTION

To estimate convexity of the virtual view distortion curve D_v, we calculate the second derivative $D''_v = \partial^2 D / \partial v^2$ of the

[1] Recall that the fourth case is neglected assuming $c_\emptyset(v) = 0$.

model proposed in Section 2 as

$$\begin{aligned} D''_v = {} & D_{t_0}[6(C_1 - C_0)v + 2(1 + C_0 - 2C_1)] + \\ & D_{t_1}[6(C_1 - C_0)v + 2(1 - C_0)] + \\ & S_d A_{d_0}[6(C_1 - C_0)v^2 - 3(2C_1 - C_0 - 1)v + C_1 - 2] + \\ & S_d A_{d_1}[-6(C_1 - C_0)v^2 - 3(1 - C_1)v + 1 - C_1 - C_0], \end{aligned} \tag{6}$$

where S_d is the scaling factor estimated from the data that embraces the impact of the correlation ρ and variance σ_t^2 on the MSE caused by encoding depth reference images.

Considering that the sign of D''_v from (6) determines local convexity of the distortion curve, we estimate convexity of the whole curve by calculating the sign of (6) at N equidistant viewpoint coordinates $0 < v < 1$. If the sign is positive or negative in more than $N/2$ coordinates, the whole distortion curve is classified as convex or concave, respectively. Note that by exploiting only the sign of the second derivative $D''(v)$, we suppress the effect of noise that is enhanced by the double derivation operator.

The model parameters C_0 and C_1 are estimated from the data at a small additional computational cost requiring one step of view warping, whereas the encoding error parameters D_{t_0}, D_{t_1}, A_{d_0} and A_{d_1} are calculated from the encoded versions of the texture and depth reference images (thus without a need for view warping calculations). The practical values for these parameters obtained in our experiments infer that the factors associated to the texture distortions contribute to (6) with a positive value folding the distortion curve to the convex shape, whereas the same associated to the depth distortions contributes with a negative amount and, hence, results in the concave shape. This phenomenon is evidenced in our experimentations presented in Section 4.

4. EXPERIMENTAL RESULTS

We have verified our prediction model on three publicly available data sets: Ballet, Breakdance [7] and Poznan Street [21]. For the former two sets, left and right references are chosen as views 2 and 3, while the latter employs views 3 and 5. Intermediate virtual views are formed through DIBR with the MPEG View Synthesis Reference Software v3.5 [22] at equidistant viewpoints each at $1/10$ of baseline distance between reference views resulting in 9 virtual viewpoints. Both texture and depth reference images are compressed with H.264/AVC JM Reference Software v17.2 [23] under various combinations of quantization parameters (QP) from within the range $\{7, 12, ...42\}$. All combinations of left and right texture as well as left and right depth compression are tested for coarse quantization, that is, for QP values of 22 and higher. For finer quantization at QP$<$ 22, left and right references, whether texture or depth, are assumed to be symmetrically compressed (same QP). In all, 680 different QP combinations were tested for each data set. Reconstruction error is measured in terms of the MSE between the virtual view synthesized from compressed references and the one synthesized from original reference images.

Table 1. Classification results for each dataset. **X** and **C** represent 'convex' and 'concave' cases, whereas **GTH** and **CR** stand for groundtruth and correct classification rate. The correct classification rate is shown in the last line for each data set. Note that our proposed scheme is capable of accurately distinguishing concave from convex profiles, thus improving upon a generally assumed common approach based on linear interpolation distortion.

Novel	**Ballet**		**Breakdance**		**Poznan**	
GTH	**X**	**C**	**X**	**C**	**X**	**C**
X	66	22	550	46	309	0
C	1	591	5	79	82	289
CR	96.6%		92.5%		87.9%	

Under the various reference image compression combinations, MSE as a function of virtual viewpoint positions can vary significantly ranging from a concave to convex shape as illustrated in Fig. 1. To characterize the curvature of the measured distortions, we classify all MSE curves as either convex or concave. Classification is based on comparison of actual MSE to the linearly interpolated values at the 9 synthesis viewpoints. If the actual MSE is larger than the linear interpolation at more than a half of the viewpoints (i.e. ≥ 5), such an MSE curve is classified as concave, otherwise, convex. This binary classification is used to generate the groundtruth data for evaluation of our prediction model.

The classification used in our novel method is also binary and based on the sign of the second derivative D''_v from (6) measured at 9 synthesis viewpoints. If this sign is positive for more than half of the viewpoints (i.e., $\geq$ 5), the modeled distortion curve is classified as convex, otherwise, concave. The parameters C_0 and C_1 are calculated as explained in Section 3, whereas S_d is estimated using the least-square error linear estimator. Fig. 3 shows the groundtruth and model classification for several examples. Note that these examples conform with the phenomenon identified in Section 3 that encoding texture at higher QPs contributes to the convex distortion curve, whereas the same for the depth images results in the concave curvature.

To measure the performance of the proposed binary classification, we show in Table 1 the number of correct and incorrect concave and convex classification results across all 680 tested reference image compression combinations as well as the correct classification rate for the chosen data sets. Our model accurately captures both the concave and convex behavior of view synthesis distortions. The best classification performance is attained for the Ballet data set in which a predominance of concave cases is present. The same compression range leads to a predominance of convex cases for Breakdancers which is also well predicted at rate above 90%. The Poznan data set provides a more equitable division among

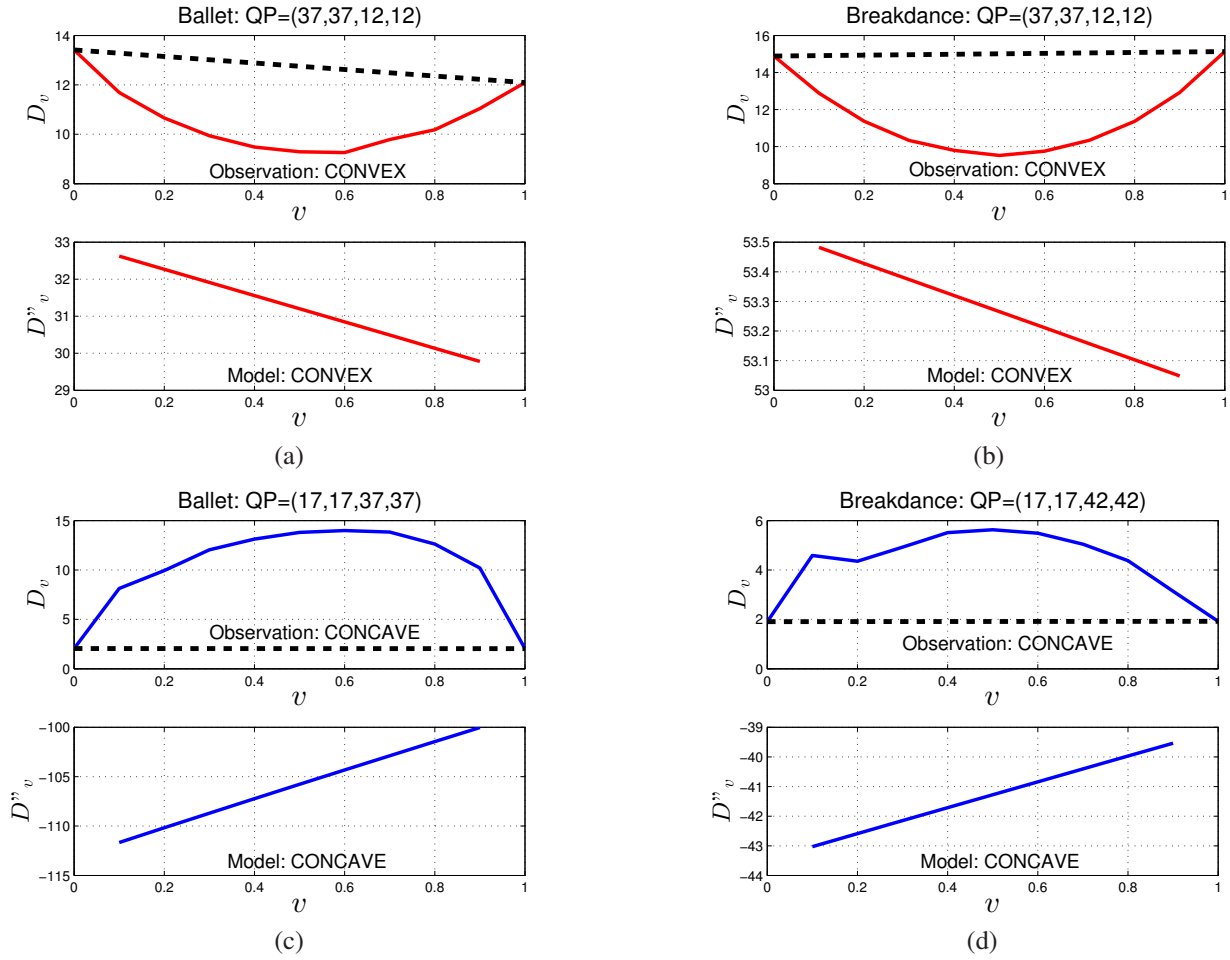


Fig. 3. View distortion: Observed and linearly interpolated values, together with modeled D''_v for (a) Ballet: convex case, (b) Breakdancers: convex case, (c) Ballet: concave case, (d) Breakdancers: concave case. Left texture, right texture, left depth and right depth reference images are respectively compressed at specified QP values.

concave and convex cases, several of which are approximately linear, leading to slightly lower classification results.

To the best of our knowledge, there are no prior methods for distortion shape characterization that we could use as reference. Thus, we consider as a common baseline a linear interpolation distortion model (e.g. [16] or [18]). Note that our proposed prediction framework can accurately and efficiently distinguish concave from convex distortion profiles, thus improving upon the baseline approach.

5. CONCLUSION

We propose a novel binary classification method to determine the shape of virtual view distortion curves in multi-view image compression, convex or concave. We demonstrate that our method is capable of achieving high classification accuracy for several data sets, while retaining reduced computational complexity. The presented preliminary results are promising, while the envisaged future work will attempt to further reduce the need for estimating the model parameters.

6. REFERENCES

[1] O. Bimber and R. Raskar, *Spatial Augmented Reality: Merging Real and Virtual Worlds*, CRC Press, Boca Raton, FL, USA, 2005.

[2] M. Budagavi, J. Furton, G. Jin, A. Saxena, J. Wilkinson, and A. Dickerson, "360 degrees video ccoding using region adaptive smoothing," in *IEEE International Conference on Image Processing*, Québec City, Canada, September 2015.

[3] T. Georgiev and A. Lumsdaine, "The focused plenoptic camera," in *IEEE International Conference on Computational Photography*, San Francisco, CA, USA, April 2009.

[4] K. Muller, P. Merkle, and T. Wiegand, "3-d video representation using depth maps," *Proceedings of the IEEE*, vol. 99, no. 4, 2011.

[5] J. G. Apostolopoulos, P. A. Chou, Bruce Culbertson, T. Kalker, M. D. Trott, and S. Wee, "The road to immersive communication," *Proceedings of the IEEE*, vol. 100, no. 4, pp. 974–990, Apr. 2012.

[6] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Multi-view video plus depth representation and coding," in *IEEE International Conference on Image Processing*, San Antonio, TX, USA, September 2007.

[7] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *ACM SIGGRAPH'04*, 2004.

[8] A. Hamza and M. Hefeeda, "Energy-efficient multicasting of multiview 3D videos to mobile devices," *ACM Trans. Multimedia Computing, Communications and Applications*, vol. 8, no. 3s, pp. 45:1–25, Sept. 2012.

[9] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, Sept. 2007.

[10] H. Yuan, Y. Chang, J. Huo, F. Yang, and Z. Lu, "Model-based joint bit allocation between texture videos and depth maps for 3-D video coding," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 21, no. 4, pp. 485–497, Apr. 2011.

[11] J. Chakareski, V. Velisavljević, and V. Stanković, "View-popularity-driven joint source and channel coding of view and rate scalable multi-view video," *IEEE J. Selected Topics in Signal Processing*, vol. 9, no. 3, pp. 474–486, Apr. 2015, special issue on Interactive Media Processing for Immersive Communication.

[12] J. Chakareski, V. Velisavljević, and V. Stanković, "User-action-driven view and rate scalable multiview video coding," *IEEE Trans. Image Processing*, vol. 22, no. 9, pp. 3473–3484, Sept. 2013, special issue on 3D Video Representation, Compression, and Rendering.

[13] V. Velisavljević, G. Cheung, and J. Chakareski, "Bit allocation for multiview image compression using cubic synthesized view distortion model," in *Proc. 2nd Int'l Workshop on Hot Topics in 3D (Hot3D)*, Barcelona, Spain, July 2011, IEEE.

[14] G. Cheung, V. Velisavljevic, and A. Ortega, "On dependent bit allocation for multiview image coding with depth-image-based rendering," *IEEE Trans. Image Proc.*, vol. 20, no. 11, 2011.

[15] V. Velisavljević, G. Cheung, and J. Chakareski, "Optimal rate allocation for view synthesis along a continuous viewpoint location in multiview imaging," in *Proc. Picture Coding Symposium*, Nagoya, Japan, Dec. 2010, pp. 482–485.

[16] 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," *Elseview Signal Processing: Image Communication*, vol. 24, no. 8, 2009.

[17] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map distortion analysis for view rendering and depth coding," in *IEEE International Conference on Image Processing*, Cairo, Egypt, November 2009.

[18] C. Dorea and R. L. de Queiroz, "Attention-weighted texture and depth bit-allocation in general-geometry free-viewpoint television," *to appear in IEEE Trans. Circuits and Systems for Video Technology*, 2017.

[19] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new apprioach on 3D-TV," *Proc. SPIE 5291, Stereoscopic Displays and Virtual Reality Systems XI*, vol. 93, May 2004.

[20] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. M uller, P.H.N. de With, and T. Wiegand, "The effects of multiview depth video compression on multiview rendering," *Signal Proc.: Image Comm.*, vol. 24, no. 1-2, 2009.

[21] M. Domanski, T. Grajek, K. Klimaszewski, M. Kurc, O. Stankiewicz, J. Stankowski, and K. Wegner, "Poznan multiview video test sequences and camera parameters," in *ISO/IEC JTC1/SC29/WG11 MPEG 2009/M17050*, Xian, China, October 2009.

[22] M. Tanimoto, M. Fujii, T. Suzuki, K. Fukushima, and N. Mori, "Reference softwares for depth estimation and view synthesis," in *ISO/IEC JTC1/SC29/WG11 MPEG 2008/M15377*, Archamps, France, April 2008.

[23] *ITU-T Recommendation and International Standard of Joint Video Specification*, ITU-T Rec H.264/ISO/IEC 14496-10 AVC, March 2005.