Graph-Based Interpolation for Zooming in 3D Scenes

Pinar Akyazi and Pascal Frossard
Signal Processing Laboratory (LTS4)
Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
Email: {pinar.akyazi, pascal.frossard}@epfl.ch

***Abstract*—In multiview systems, color plus depth format builds 3D representations of scenes within which the users can freely navigate by changing their viewpoints. In this paper we present a framework for view synthesis when the user requests an arbitrary viewpoint that is closer to the 3D scene than the reference image. On the target image plane, the requested view obtained via depth-image-based-rendering (DIBR) is irregularly structured and has missing information due to the expansion of objects. We propose a novel framework that adopts a graph-based representation of the target view in order to interpolate the missing image pixels under sparsity priors. More specifically, we impose that the target image is reconstructed with a few atoms of a graph-based dictionary. Experimental results show that the reconstructed views have better PSNR and MSSIM quality than the ones generated within the same framework with analytical dictionaries, and are comparable to the ones reconstructed with TV regularization and linear interpolation on graphs. Visual results, however, show that our method better preserves the details and results in fewer disturbing artifacts than the other interpolation methods.**

***Index Terms*—Graph signal processing (GSP), depth-image-based-rendering (DIBR), free viewpoint navigation, interpolation**

I. Introduction

Multiview image processing has been receiving increased attention lately with the advent of interactive navigation applications and immersive communication. One of the main challenges in multiview systems is to offer a smooth navigation in a 3D environment through an effective combination of camera images and virtual views. While such navigation has been quite intensively studied for settings when a user virtually moves around the 3D scene, less work has been done when navigation brings the user towards the scene. In this case, the main challenge consists in properly interpolating details that might become apparent in the virtual views, even if they are not fully available in the reference views.

In this paper, we focus on the specific problem of zooming in a 3D scene from a reference camera image. We propose a novel method for interpolating the 3D visual information on a regular image grid. This poses significant image reconstruction challenges due to the lack of sufficient details in the reference image, and the different expansion rates for objects in different depth layers.

Specifically, given one reference image of a 3D scene and the corresponding depth image, we want to reconstruct a target virtual view that is closer to the scene via DIBR algorithm [1] that uses depth information to estimate the image content by geometric projections. Because DIBR is a pixel-based projection method, pixels on the target view lie on an irregular structure which necessitates a proper interpolation strategy to reconstruct the virtual view. We represent the projected pixels on the target view as a signal on a graph, which provides us with the benefit of embedding the scene geometry within the graph topology. We then propose a regularization framework with a sparsity constraint and solve the interpolation problem using the Orthogonal Matching Pursuit (OMP) algorithm [2].

Image interpolation has been addressed in the literature mainly using deterministic and statistical methods [3], [4]. Deterministic methods assume a functional relationship between samples while statistical methods aim to minimize an estimation error. More recently, graph-based methods have been proposed for regularization [5], [6], [7]. Interpolation of graph signals is handled in [8] from a sampling perspective, while sparsity based interpolation methods using spectral graph theory are presented in [9], [10], [11]. The work in [12] formulates a patch-based maximum aposteriori problem to fill the expansion holes using a smoothness prior on the graph signal. None of the mentioned works have an irregular domain representation and the ability to preserve details without restrictive priors on band-limitedness. In particular, the method proposed in this paper permits to represent signals on irregular graphs, taking into account both the geometry of the scene and the signal values themselves, hence yielding to a more detailed and consistent representation desired in a free viewpoint navigation system.

The outline of the paper is as follows. We introduce our view synthesis framework and the graph-based view representation formalism in Section II. In Section III we formulate our graph-based interpolation problem and describe our new solution based on a parametric spectral graph dictionary. We validate the performance of our approach in Section IV. Finally, conclusions are presented in Section V.

II. Depth-image-based interpolation framework

We consider a framework where we reconstruct a virtual image based on a reference camera image and a depth map captured further away from a 3D scene. With such forward displacement, all objects of the scene are expanding, with a faster rate for foreground objects than for the background, so

ISBN 978-0-9928626-7-1

that some image details become more prominent. We build an estimate of the virtual image through DIBR. The projected pixels, however, do not necessarily fall on the integer pixel positions in the virtual view, as illustrated in Fig. 1. We therefore need to interpolate the projected pixels, and possibly fill in the expansion holes, in order to reconstruct a proper estimate of the virtual view.

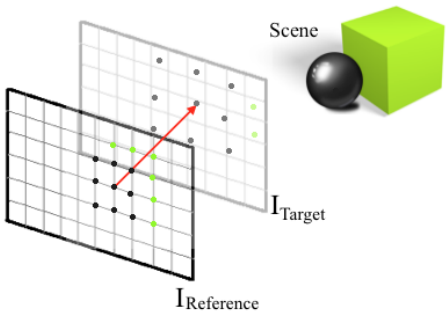


Fig. 1. Illustration of DIBR where the target image plane is closer to the scene with respect to the reference image.

We propose to adopt a graph-based representation of the target view, as it permits to describe data that lie on irregular structures, such as the one created by depth-based projection of the pixels in the reference image. A graph is denoted as $\mathcal{G} = (\mathcal{V}, E)$ where $\mathcal{V}$ are the vertices and E are the edges in between. The set of vertices $\mathcal{V}$ is the union of the set of vertices $\mathcal{V}_0$ that correspond to the pixel positions in the target view, and the set of vertices $\mathcal{V}_1$ that correspond to the positions of the projected pixels. The signal lying on $\mathcal{G}$ describes the luminance information and is denoted as y. In general, no signal value is available on the vertices in $\mathcal{V}_0$ after depth-based projection of the reference image, which rather adds values on the vertices in $\mathcal{V}_1$. The objective of our interpolation algorithm is exactly to estimate the values of the signal y on the vertices in $\mathcal{V}_0$ for target view rendering. In more details, we build the graph by connecting vertices i and j with an edge of weight $w(i,j)$ if their 3D Euclidean distance is smaller than a threshold, and if both vertices correspond to objects on the same depth layer. We choose to set the edge weights to be a combination of spatial distance and luminance difference between connected nodes, and we compute them as

$$w(i,j) = \exp - \left(\frac{d_{i,j}^2}{2\sigma_{dist}^2} + \frac{(y_i - y_j)^2}{2\sigma_{val}^2} \right) \qquad (1)$$

for which $d_{i,j}$ is the 3D Euclidean distance between nodes i and j, y_i and y_j are the signal values on nodes i and j, and σ_{dist} and σ_{val} characterize the geometric and photometric spreads of the signal on graph, respectively. It can be shown that this type of weights is similar to the weights used in bilateral filtering, which has been very successfully used in different image processing tasks [13].

In order to compute the edge weights, we estimate the values of the signal on vertices $\mathcal{V}_0$ by linear interpolation of the signal values on neighbour vertices in $\mathcal{V}_1$. Specifically, we first build a triangular mesh on the reference depth image that connects pixels belonging to the same layer. We do not draw connections between pixels when the difference between their depths exceed the predefined threshold. With the camera displacement towards the 3D scene, the mesh triangles are projected onto the target view. When a background pixel falls inside a triangle corresponding to a foreground object, this projection is discarded from the target view, which further prevents layer blending. The neighbour nodes that are finally chosen for linear interpolation of signal on the vertices in $\mathcal{V}_1$ correspond to the corners of each projected triangle.

III. Graph-based Interpolation Problem

Equipped with the above graph-based representation, we describe now our image interpolation problem. The objective is to use the information obtained by depth-based projection of the vertices $\mathcal{V}_1$ to interpolate the image values on the grid given by $\mathcal{V}_0$. This objective may lead to an ill-posed problem, especially when there is a relatively large forward displacement between the reference and synthetic view. We therefore add a sparsity prior on the reconstructed image, such that the details and texture have higher chances to be preserved in the synthetic view. Altogether, we formulate our image interpolation problem as follows:

$$\min_x \|\mathcal{M}(y - \mathcal{D}x)\|_2^2 \quad \text{subject to } \|x\|_0 \leq T_0 \qquad (2)$$

where y is the target image, x is a sparse coefficient vector, $\mathcal{D}$ is a dictionary of graph atoms, and T_0 is a sparsity threshold. We further introduce a binary mask $\mathcal{M}$ which takes the value 1 for the entries corresponding to the vertices in $\mathcal{V}_1$ and 0 otherwise. The minimization function measures the error, while the constraint ensures a sparse reconstruction with T_0 atoms. We solve the problem given in (2) using the OMP algorithm [2] as it often provides an effective tradeoff between computational complexity and quality of the sparse reconstruction. We however note that other sparse reconstruction methods like ℓ_1 minimization algorithms for example could be used to solve (2) or similar objective functions.

Obviously, the choice of the dictionary $\mathcal{D}$ in (2) has a large influence on the quality of the reconstruction result. The dictionary has to be able to effectively represent the most relevant features of natural images on irregular graphs. We therefore propose to use here a spectral graph dictionary learned on a set of training images. We form the dictionary as a concatenation of subdictionaries that are polynomials of the Laplacian $\mathcal{L}$ of the graph $\mathcal{G}$, as defined in [14], [15]. As the atoms are constructed on a polynomial kernel, they are well localized on the graph, which permits to effectively represent the local characteristics of the target images. As an additional atom to our learned dictionary we add the eigenvector of $\mathcal{L}$ that corresponds to the smallest eigenvalue of $\mathcal{L}$, which is a constant valued vector analogous to the DC component of signals. We finally use the polynomial dictionary in order to solve (2) and obtain the sparse approximation of the projected signal on any target graph.

ISBN 978-0-9928626-7-1 © EURASIP 2017

IV. Experiments

A. Experimental Settings

We now evaluate the performance of our interpolation algorithm in different datasets and for different zoom in factors. We compare our solution with similar algorithms to the one described in Section III, but with alternative dictionaries. In particular, we compare our parametric dictionary with two transform dictionaries: the Graph Fourier Transform (GFT) dictionary [16], which is the orthonormal dictionary that is composed of the eigenvectors of the graph laplacian $\mathcal{L}$, and the Spectral Graph Wavelet Transform (SGWT) dictionary [17] that has $J = 3$ scales and a lowpass subdictionary. We also compare our method with TV regularization on weighted graphs [6], [7] and a classical interpolation method that simply moves the projected pixels onto the closest grid points on the target image and eventually performs linear interpolation to estimate the unknown values of the signal.

In order to learn our graph spectral dictionary $\mathcal{D}$, we use depth and texture patches of size $n \times n$ from the Microsoft MSR 3D video datasets [18], which are used to simulate small forward displacements. The resulting graph signals, on grid pixels and projected pixels, are used to train our dictionary. We chose $n = 10$ for patch size, $K = 5$ for the degree of polynomial functions and $S = 4$ for the number of subdictionaries. We fix the sparsity constraint to 10% of the signal dimension in the dictionary learning algorithm [15] and we use a total of 2880 training signals on different graphs.

We then implement the image interpolation algorithm on patches of the complete images, namely on overlapping graphs that are in $b \times b$ pixel range in the target image plane, with $b = 30$. We test our interpolation algorithm on 6 images from the 2005 Middlebury Stereo Dataset [19]. We have performed our experiments using two different user displacements. The first case is a smaller user displacement that yields to approximately 59% and 34% expansion holes in the target view for graph based methods and the linear interpolation method, respectively. For the larger user displacement, the respective ratios increase to 67% and 51%. We present the visual results in Fig. 2-5. PSNR and MSSIM [20] values are presented in Tables I-IV.

TABLE I
PSNR values for the interpolation of target views with 59% and 34% expansion holes for graph based and linear interpolation methods, respectively.

	GFT	**SGWT**	**Linear**	**TV0**	**TV0.5**	**TV1**	**Our method**
Art	32.49	30.17	34.41	29.92	33.84	34.44	33.90
Moebius	30.33	28.75	34.50	30.66	34.41	35.10	34.38
Books	28.06	24.25	31.43	28.11	30.89	31.18	29.53
Laundry	27.62	25.38	31.27	28.79	31.43	31.72	30.54
Reindeer	30.27	29.87	34.55	30.40	33.73	34.13	33.08
Dolls	27.86	26.96	32.14	27.75	32.02	32.49	31.25

TABLE II
PSNR values for the interpolation of target views with 67% and 51% expansion holes for graph based and linear interpolation methods, respectively.

	GFT	**SGWT**	**Linear**	**TV0**	**TV0.5**	**TV1**	**Our method**
Art	29.80	27.24	31.58	29.09	32.57	32.92	32.31
Moebius	28.35	25.8	31.57	29.94	33.22	33.69	33.10
Books	26.27	22.45	28.56	27.04	29.38	29.57	28.30
Laundry	25.12	23.30	28.08	26.89	29.10	29.26	28.44
Reindeer	28.67	28.01	32.08	29.90	32.69	32.94	32.24
Dolls	25.58	23.85	29.27	26.80	30.32	30.62	29.59

TABLE III
MSSIM values for the interpolation of target views with 59% and 34% expansion holes for graph based and linear interpolation methods, respectively.

	GFT	**SGWT**	**Linear**	**TV0**	**TV0.5**	**TV1**	**PolDict**
Art	0.9917	0.9863	0.9955	0.9692	0.9911	0.9933	0.9935
Moebius	0.9815	0.9764	0.9942	0.9556	0.9843	0.9889	0.9899
Books	0.9716	0.9574	0.9898	0.9467	0.9776	0.9820	0.9796
Laundry	0.9815	0.9762	0.9913	0.9567	0.9851	0.9882	0.9878
Reindeer	0.9817	0.9801	0.9938	0.9655	0.9875	0.9901	0.9895
Dolls	0.9778	0.9766	0.9904	0.9488	0.9866	0.9896	0.9881

TABLE IV
MSSIM values for the interpolation of target views with 67% and 51% expansion holes for graph based and linear interpolation methods, respectively.

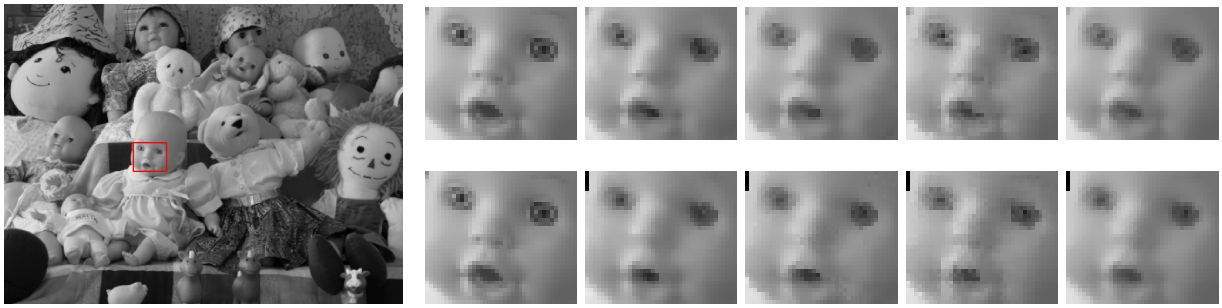
	GFT	**SGWT**	**Linear**	**TV0**	**TV0.5**	**TV1**	**PolDict**
Art	0.9856	0.9771	0.9913	0.9672	0.9897	0.9919	0.9919
Moebius	0.9713	0.9638	0.9880	0.9508	0.9818	0.9864	0.9875
Books	0.9602	0.9419	0.9796	0.9395	0.9730	0.9776	0.9763
Laundry	0.9687	0.9619	0.9806	0.9470	0.9790	0.9820	0.9810
Reindeer	0.9728	0.9645	0.9888	0.9631	0.9857	0.9881	0.9873
Dolls	0.9650	0.9622	0.9843	0.9396	0.9818	0.9851	0.9828

B. Interpolation Performance

We first see in Tables I to IV that our interpolation method with a learned dictionary outperforms the alternative dictionaries for all test images in expansion hole filling, in terms of both PSNR and MSSIM metrics. The atoms of the GFT dictionary are not well localized in the vertex domain, while SGWT atoms are more localized both in the vertex and spectral domains but not specifically adapted to the statistics of the graph signals under consideration. Our learned dictionary has the advantage of preserving the common spectral components of the class of natural images, hence provides better reconstruction quality.

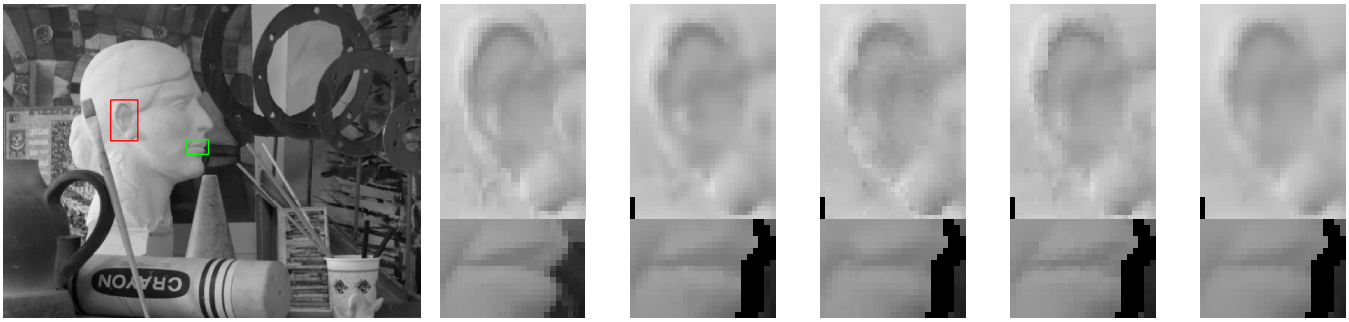
The other graph based interpolation method that we have used for comparison is the TV regularization. We have performed TV regularization with three different fidelity parameters λ for the fitting term and used the linearly interpolated signal values as initial data. In this case, the quantitative metrics like PSNR and MSSIM take higher values for the TV regularized target views with $\lambda = 1$ compared to our method, especially when the user displacement is smaller. Likewise, the PSNR and MSSIM measures for linear interpolation are better for smaller user displacement since less holes are present on the target view. Visual artifacts for TV regularization methods and linear interpolation are however clearly visible on the interpolated images in Fig. 2-5(e) and (f). Significant rounding artifacts around objects are present as a result of the linear interpolation while the blurring effect of TV regularization is observable throughout the figures. TV regularization also brings a higher computational cost as it requires iterations until convergence. Although linear interpolation has the smallest

ISBN 978-0-9928626-7-1 © EURASIP 2017



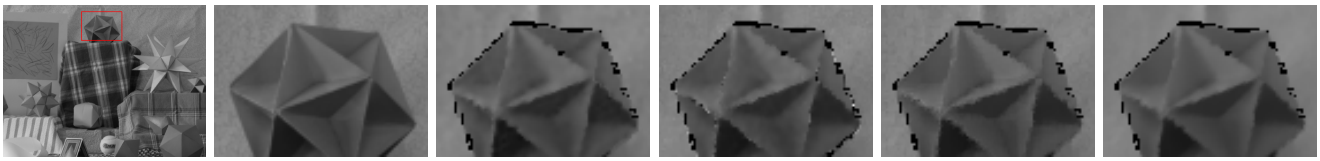
(a) (b) (c) (d) (e) (f)

Fig. 2. (a) Dolls image. (b) Detail view of the ground truth within red rectangle. (c)-(f) Interpolation results within red rectangle: (c) our method, (d) GFT, (e) Linear, (f) $TV_{\lambda=1}$ for 59% and %34 expansion holes (top row), 67% and 51% expansion holes (bottom row) for graph based and linear interpolation methods, respectively.



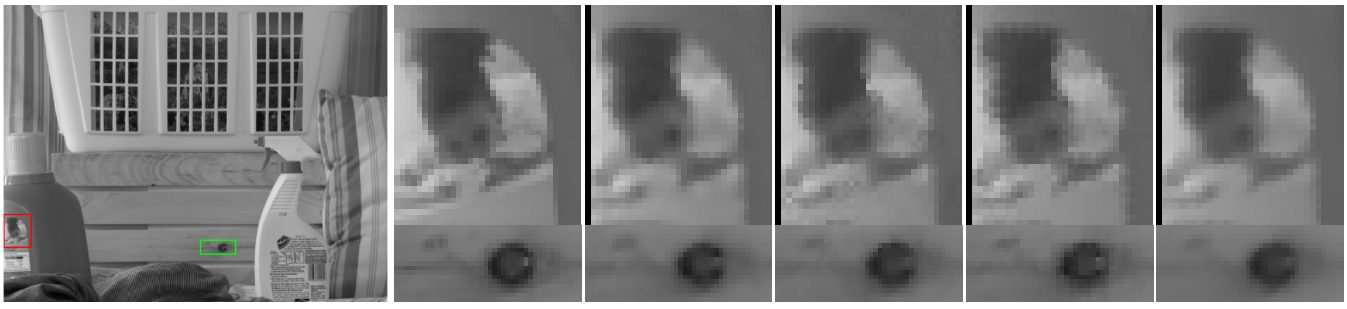
(a) (b) (c) (d) (e) (f)

Fig. 3. (a) Art image. (b)-(f) Detail view of the ground truth and interpolation results within red (top row) and green (bottom row) rectangles: (b) ground truth, (c) our method, (d) GFT, (e) Linear, (f) $TV_{\lambda=1}$ for 67% and 51% expansion holes for graph based and linear interpolation methods, respectively.



(a) (b) (c) (d) (e) (f)

Fig. 4. (a) Moebius image. (b)-(f) Detail view of the ground truth and interpolation results within red rectangle: (b) ground truth, (c) our method, (d) GFT, (e) Linear, (f) $TV_{\lambda=1}$ for 67% and 51% expansion holes for graph based and linear interpolation methods, respectively.



(a) (b) (c) (d) (e) (f)

Fig. 5. (a) Laundry image. (b)-(f) Detail view of the ground truth and interpolation results within red (top row) and green (bottom row) rectangles: (b) ground truth, (c) our method, (d) GFT, (e) Linear, (f) $TV_{\lambda=1}$ for 67% and 51% expansion holes for graph based and linear interpolation methods, respectively.

computational complexity among all other methods, the results become less reliable with increasing zoom factors while rounding errors are highlighted even more. Our method does not present these artifacts, as expected from the benefits of graph based representation, and performs better than the linear interpolation method on object boundaries and continuous structures, as well as textured areas. Fig. 4 shows that our method preserves the background texture better than TV, while

ISBN 978-0-9928626-7-1 © EURASIP 2017

there are less artifacts on the foreground compared to linear interpolation and GFT. Similarly, we see that our method has less blur both in the foreground and background in Fig. 5 compared to other methods. The top row shows that our method is able to achieve a more detailed reconstruction of foreground textures with much fewer artifacts. Despite a higher computational complexity for solving the optimization problem in equation (2) compared to basic linear interpolation, we obtain higher quality target views using our polynomial dictionary.

V. CONCLUSION

In this work, given a reference texture and depth image, we have synthesized a target image located at a closer viewpoint to the scene. We have represented the target view as a weighted graph that separates objects in different layers and contains information on both the topology of the scene and signal values. We have learned a parametric dictionary on multiple graphs whose atoms carry the common characteristics of natural images in our framework, and used a dictionary-based regularization method under sparsity constraints. Visual results show that our method preserves the details in the target view better and avoids rounding errors and smoothing effects that are present in competitor solutions, while the quantitative performance is comparable to other interpolation methods.

ACKNOWLEDGMENT

This work has been partly funded by the Swiss National Science Foundation under Grant 200021-149800.

REFERENCES

[1] C. Fehn, "Depth-image-based rendering (dibr), compression, and transmission for a new approach on 3d-tv," in *Electronic Imaging 2004*. International Society for Optics and Photonics, 2004, pp. 93–104.

[2] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[3] P. Thévenaz, T. Blu, and M. Unser, "Image interpolation and resampling," *Handbook of medical imaging, processing and analysis*, pp. 393–420, 2000.

[4] T. F. Chan and J. Shen, *Image processing and analysis: variational, PDE, wavelet, and stochastic methods*. SIAM, 2005.

[5] S. K. Narang, A. Gadde, E. Sanou, and A. Ortega, "Localized iterative methods for interpolation in graph structured data," in *Proc. of the IEEE Global Conference on Signal and Information Processing*, 2013, pp. 491–494.

[6] M. Ghoniem, Y. Chahir, and A. Elmoataz, "Geometric and texture inpainting based on discrete regularization on graphs," in *Proc. of the IEEE International Conference on Image Processing*, 2009, pp. 1349–1352.

[7] A. Elmoataz, O. Lezoray, and S. Bougleux, "Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing," in *IEEE Transactions on Image Processing*, vol. 17, no. 7, 2008, pp. 1047–1060.

[8] S. K. Narang, A. Gadde, and A. Ortega, "Signal processing techniques for interpolation in graph structured data," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 5445–5449.

[9] Y. Mao, G. Cheung, and Y. Ji, "Image interpolation for dibr view synthesis using graph fourier transform," in *Proc. of the 3DTV-Conference:*, 2014, pp. 1–4.

[10] Y. Mao, G. Cheung, A. Ortega, and Y. Ji, "Expansion hole filling in depth-image-based rendering using graph-based interpolation." in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 1859–1863.

[11] Y. Mao, G. Cheung, and Y. Ji, "Graph-based interpolation for dibr-synthesized images with nonlocal means," in *Proc. of the IEEE Global Conference on Signal and Information Processing*, 2013, pp. 451–454.

[12] ——, "On constructing z-dimensional dibr-synthesized images," *IEEE Transactions on Multimedia*, vol. 18, no. 8, pp. 1453–1468, 2016.

[13] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. of the IEEE Conference on Computer Vision*, 1998, pp. 839–846.

[14] D. Thanou, D. I. Shuman, and P. Frossard, "Learning parametric dictionaries for signals on graphs," *IEEE Transactions on Signal Processing*, vol. 62, no. 15, pp. 3849–3862, 2014.

[15] D. Thanou and P. Frossard, "Multi-graph learning of spectral graph dictionaries," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 3397–3401.

[16] D. Shuman, S. K. Narang, P. Frossard, A. Ortega, P. Vandergheynst *et al.*, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," in *IEEE Signal Processing Magazine*, vol. 30, no. 3, 2013, pp. 83–98.

[17] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2011.

[18] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *ACM Transactions on Graphics*, vol. 23, no. 3, 2004, pp. 600–608.

[19] H. Hirschmuller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[20] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on*, vol. 2. IEEE, 2003, pp. 1398–1402.

ISBN 978-0-9928626-7-1