Graph-Based Inpainting of Disocclusion Holes for Zooming in 3D Scenes

Pinar Akyazi
Multimedia Signal Processing Group (MMSPG)
Ecole Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
Email: pinar.akyazi@epfl.ch

Pascal Frossard
Signal Processing Laboratory (LTS4)
Ecole Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
Email: pascal.frossard@epfl.ch

***Abstract*—Color plus depth format allows building 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. The requested view constructed via depth-image-based-rendering (DIBR) on the target image plane has missing information due to the expansion of objects and disoccluded areas. Building on our previous work on expansion hole filling, we propose a novel method that adopts a graph-based representation of the target view in order to inpaint the disocclusion holes under sparsity priors. Experimental results indicate that the reconstructed views have PSNR and SSIM quality values that are comparable to those of the state of the art inpainting methods. Visual results show that we are able to preserve details better without introducing blur and reduce artifacts on boundaries between objects on different layers.**

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

I. Introduction

Multiview systems primarily try to offer the users a smooth navigation experience in a 3D environment through an effective combination of camera images and virtual (i.e., synthesized) views. Such navigation has been extensively studied for user movement equidistant from the 3D scene. However, much less work has been focused on the case where user navigates to a closer point within the scene. In this case, properly representing the details that become available in the virtual views is the main challenge, since the revealed information is not included a priori 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 graph-based method to inpaint disoccluded areas in the synthesized view, that are occluded by foreground objects in the reference view. Given one reference image of a 3D scene and the corresponding depth image, we first construct a target virtual view that is closer to the scene using DIBR algorithm [1] that estimates the virtual image content by employing depth information and geometric projections. 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. When the user comes closer to the scene, expansion holes are created on the requested view due to enlarging objects with respect to the reference view. We had presented a framework for interpolating the expansion holes on the target view in our previous work [2], that approximates the missing values of the graph signal using a sparse representation on a parametric spectral graph dictionary. We now propose a graph-based regularization framework with a sparsity constraint to inpaint the disocclusion holes on the synthesized texture image. We solve our problem using the Orthogonal Matching Pursuit (OMP) algorithm [3] as it often provides an effective trade-off between computational complexity and quality of the sparse reconstruction.

Image inpainting algorithms can be divided into three general classes: statistical, PDE-based and exemplar-based methods. In statistical approaches, an input texture is described by extracting textures through the use of compact parametric statistical models [4]. These methods fail in the presence of additional intensity gradients. PDE-based methods propagate structures from the known part of the image [5]–[7] introducing smoothness priors, however they fail to reconstruct large holes with high frequency textures. PDE-based methods are therefore more well-suited for inpainting small holes and flat structures. Among the three classes, exemplar-based methods are most widely used. The holes in image are filled using exemplars sought throughout local or global search regions, where the most challenging tasks consist in determining the filling order of pixels via a priority term, and choosing the exemplar [8]–[12]. These works only use the texture images and not the depth maps. Depth guided inpainting methods [13]–[19] aim to prioritize the background in choosing the filling order and seek exemplars that are consistent with scene geometry. There are also hybrid methods [20]–[22] that combine multiple classes of approaches for inpainting.

Unlike the mentioned works dealing with horizontal translation warping, it is no longer possible to use the reference texture and depth images as exemplars in the case of zooming in. This is due to expansion of objects in different layers and the details that might be revealed as the user approaches the scene. Since the requested viewpoint is closer to the scene, objects will become larger in the target view compared to the reference view. This difference in dimensions of objects prevents inpainting the disocclusion holes using exemplars

ISBN 978-90-827970-1-5

from the reference view directly. We also do not have a unique direction at which the disocclusion holes will appear. None of the mentioned works propose a solution for inpainting of images closer to the scene. Our method preserves both the texture and the geometry of the scene while jointly inpainting intensity and depth images, even if it cannot benefit from information such as exemplars or a unique direction of holes that are available in translations equidistant to the scene.

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 inpainting problem and describe our new solution based on a sparse representation on a parametric spectral graph dictionary. We validate the performance of our approach in Section IV. Finally, conclusions are presented in Section V.

II. Zooming in 3D scenes and Graph-Based Representation

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, and some image details become more prominent. We build an estimate of the virtual image through DIBR and fill in the expansion holes using the graph-based interpolation method explained in [2]. Graph-based representation 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. The distribution of expansion holes, however, are different than disocclusion holes in the sense that the former appear as cracks and are smaller compared to the latter, as shown in Figure 1. We therefore need to propose a different inpainting strategy to fill larger holes.

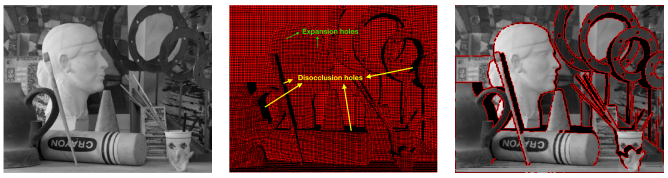


Fig. 1. Target view (left). Map of projected (red) pixels rounded onto grid positions (middle). The black cracks are expansion holes while dense black areas are disocclusion holes. The disocclusion hole boundary $\delta\mathcal{V}_0$ before inpainting is shown in red (right).

In our approach, we denote a graph as $\mathcal{G} = (\mathcal{V}, E)$ where $\mathcal{V}$ are the vertices and E are the edges between vertices. The set of vertices $\mathcal{V}$ is the union of the set of vertices $\mathcal{V}_0$ that correspond to the pixel positions of the disocclusion holes in the target view, and the set of vertices $\mathcal{V}_1 = \mathcal{V} - \mathcal{V}_0$ that correspond to the positions of pixels whose values are known after depth based rendering. The signal lying on $\mathcal{G}$ describes the luminance information and is denoted as y. The objective of our inpainting algorithm is to estimate the values of the signal y on the vertices in $\mathcal{V}_0$. Vertices $i, j \in \mathcal{V}_1$ are connected with an edge if they are on the same depth layer, meaning that their depth values do not differ more than a small threshold, and the edge weight $w(i, j)$ is a measure of similarity between the signal values on vertices i and j. In general, no signal value is available on the vertices in $\mathcal{V}_0$ after depth-based projection and expansion hole interpolation on the reference image, therefore the weight between nodes $i \in \mathcal{V}_0$ and $j \in \mathcal{V}_1$ is a function of geometric distance only.

III. Inpainting

In this section we explain our proposed method to inpaint the disocclusion holes in detail. We first describe how we determine the filling order of the disocclusion hole pixels, starting from the disocclusion hole boundary denoted as $\delta\mathcal{V}_0$. We then describe our inpainting strategy on depth images, and our graph-based inpainting method for texture images.

A. Priority

The performance of inpainting algorithms are vastly influenced by the filling order among the nodes lying on the disocclusion hole boundary. We want to propagate the signal values from $\mathcal{V}_1$ to $\mathcal{V}_0$ starting from the node $p \in \delta\mathcal{V}_0$ such that the $k \times k$ region around p denoted by Ψ_p is (i) the most confident region at the hole boundary $\delta\mathcal{V}_0$, (ii) the region with least depth variation and (iii) favors the propagation of signal values from background onto the holes.

We quantify the effect of the factors stated above by assigning functions and computing the combined impact denoted as the priority function $P(p)$ for each pixel p on $\delta\mathcal{V}_0$. To give higher priority to regions that have greater percentage of nodes with known values, we use the *confidence* term $C(p)$ proposed in [8] preferred by most state-of-the-art inpainting methods. We adopt the *level regularity* term $L(p)$ introduced in [13] which is the inverse depth variance of the corresponding region Ψ_p on the depth image Z, denoted by Z_p. We prioritize the regions with larger overall depth values, using a *depth mean* term $Z_{near} - \bar{Z}_p$ as described in [14], where Z_{near} is the maximum depth value in reference depth image and $\bar{Z}_p$ is the mean depth value around pixel p, computed using available values. We introduce the *background* term $I(p)$, which further influences the filling order to favor pixels p on the background with respect to others. We then combine these terms and formulate out priority function as:

$$P(p) = (C(p) + L(p)) \times (Z_{near} - \bar{Z}_p) \times I(p) \quad (1)$$

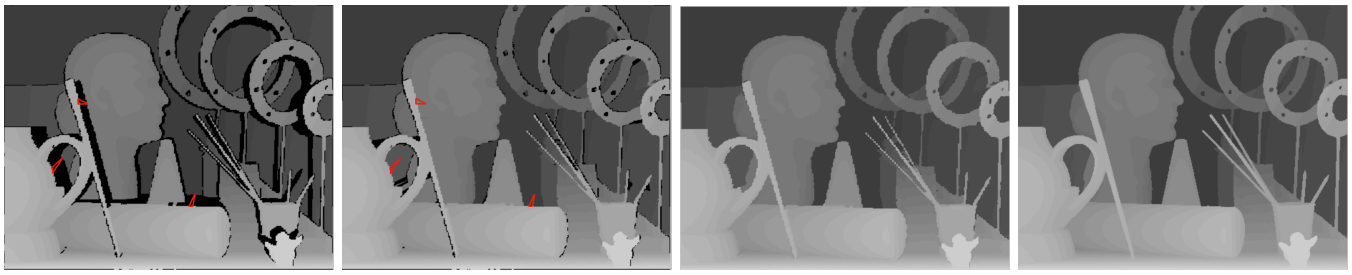
$$\text{with } C(p) = \frac{\sum_{q \in \Psi_p \cap \mathcal{V}_1} C(q)}{|\Psi_p|} \quad (2)$$

$$L(p) = \frac{|Z_p|}{|Z_p| + \sum_{q \in Z_p \cap \mathcal{V}_1} (Z_p(q) - \bar{Z}_p)^2} \quad (3)$$

$$I(p) = \frac{1}{Z_p^2(p)} \quad (4)$$

In our formulation we want the influence of depth mean term and background term to be higher than the confidence

ISBN 978-90-827970-1-5 © EURASIP 2018



(a) (b) (c) (d)

Fig. 2. (a) Art depth image with disocclusion holes and (b) disocclusion holes reduced by projection of the mesh built on reference depth image. Exemplary projected triangles are depicted in red color. (c) Our resulting depth map after proximity background interpolation, (d) ground truth depth map.

and level regularity terms individually, to ensure the propagation of background values into the holes rather than the foreground values. We therefore add the confidence and level regularity terms first, and introduce the sum as a multiplier in (1). The depth mean and background terms dominate the priority measure in this case. We choose the pixel $\hat{p}$ on the disocclusion hole boundary $\delta\mathcal{V}_0$ with the highest priority value, i.e., $\hat{p} = \arg\max_p P(p)$, and move on to the inpainting of the region centered around $\hat{p}$.

B. Depth and Texture Inpainting

After finding the pixel $\hat{p}$ with the highest priority, we find the node set $\mathcal{V}_s$ within the $k \times k$ area centered around $\hat{p}$ such that any node $q \in \mathcal{V}_s$ if $z(q) \leq z(\hat{p}) + t_{\text{depth}}$, where t_{depth} is a depth threshold arbitrarily determined to make sure we inpaint pixels only on the same depth layer.

Our depth map Z has much less disocclusion holes compared to the texture image Y due to the mesh we add on the layer boundaries of the reference depth image prior to performing DIBR. The mesh consists of triangles connecting different layers on the reference image, which are then projected onto the target image plane by DIBR. Examples of projected triangles with corners on different depth layers are depicted using red color in Figure 2(a). Disocclusion holes inside these triangles in the target depth image are filled using the minimum depth value on the corners of each triangle. We infer that this value corresponds to the background depth value to be revealed in the target view. The result of this reduction can be seen on Figure 2(b). If any holes remain on the depth map within the region corresponding to nodes $\mathcal{V}_s$, we fill them by using the smallest known depth value on immediate neighbors of $\hat{p}$. Our final result on the art image is shown on Figure 2(c).

We then move onto texture inpainting. We build a graph $\mathcal{G}_{\mathcal{V}_s}$ on nodes $\mathcal{V}_s$ on the texture image. By working on the graph domain we are able to represent projected nodes lying on an irregular grid together with the grid nodes on the target view, and convey a weighted measure of similarity between known nodes, filled nodes and disocclusion holes. The edge weight connecting nodes i and j is determined as:

$$w(i,j) = \gamma \cdot \exp - \left(\lambda_1 \frac{(y(i)-y(j))^2}{2\sigma_t^2} + \lambda_0 \frac{d_{i,j}^2}{2\sigma_d^2} \right) \tag{5}$$

for which $d_{i,j}$ is the 3D Euclidean distance between nodes i and j, $y(i)$ and $y(j)$ are the texture values on nodes i and j, and σ_d and σ_t characterize the geometric and photometric spreads of the signal on graph, respectively. The tuning parameters γ, λ_0 and λ_1 are determined as:

$$\gamma = \begin{cases} 2 & \text{if } i,j \in \mathcal{V}_{1_{\text{init}}} \\ 1 & \text{otherwise} \end{cases} \tag{6}$$

$$(\lambda_1, \lambda_0) = \begin{cases} (1,0) & \text{if } i,j \in \mathcal{V}_1 \\ (0,1) & \text{otherwise} \end{cases} \tag{7}$$

where $\mathcal{V}_{1_{\text{init}}}$ represents the known nodes on the texture image at the very beginning of the inpainting algorithm, i.e., the initial set $\mathcal{V}_1$. This parametrization, which has been tuned experimentally, ensures stronger connections with known nodes in the initial texture image, and places higher influence on the texture information when it is available. We further introduce a binary diagonal mask $\mathcal{M}$, which takes the value 1 for the entries corresponding to the vertices in $\mathcal{V}_1$ and 0 otherwise. We then discard the disocclusion holes $\mathcal{V}_d \subset \mathcal{V}_s$ with the weakest weights until $\frac{|\mathcal{V}_d|}{|\mathcal{V}_s|} = 0.1$ and obtain the final node set $\hat{\mathcal{V}}_s$. We formulate our inpainting problem as follows:

$$\min_x \|\mathcal{M}(y(\hat{\mathcal{V}}_s) - \mathcal{D}x)\|_2^2 \quad \text{subject to } \|x\|_0 \leq T_0 \tag{8}$$

where y is the target signal, x is a sparse coefficient vector, $\mathcal{D}$ is a dictionary of graph atoms, and T_0 is a sparsity threshold. The minimization function measures the error, while the constraint ensures a sparse reconstruction with T_0 atoms of $\mathcal{D}$, which is an overcomplete spectral graph dictionary. The dictionary is learned on a set of training images [23], [24] and it is able to effectively represent the most relevant features of natural images represented as graph signals [2]. We solve the problem given in (8) using the OMP algorithm [3] as it often provides an effective trade-off between computational complexity and quality of the sparse reconstruction. As we fill the disocclusion holes in $\mathcal{G}_{\hat{\mathcal{V}}_s}$, we update the confidence and mask values:

$$C(p) = C(\hat{p}),\ \mathcal{M}(p) = 1 \quad \forall p \in \hat{\mathcal{V}}_s \cap \mathcal{V}_0 \tag{9}$$

ISBN 978-90-827970-1-5 © EURASIP 2018

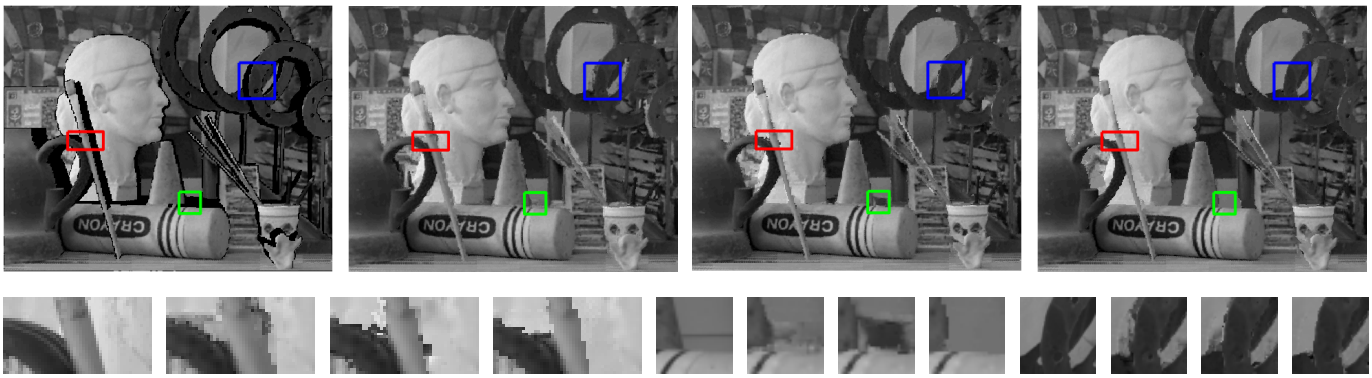


Fig. 3. Top row: Art image with disocclusion holes, results of inpainting methods JTDI [14], DGDI [19] and our method respectively. Bottom row: Enlarged ground truth and inpainting results of JTDI, DGDI, and our methods arranged in 3 groups corresponding to red, green and blue areas in top row, respectively.

With the mask update, we include the nodes $\hat{\mathcal{V}}_s \cap \mathcal{V}_0$ in set $\mathcal{V}_1$ and not in set $\mathcal{V}_0$ as they are no longer holes. We then continue inpainting until $\mathcal{V}_0 = \emptyset$.

The dictionary $\mathcal{D}$ in (8) has to be able to effectively represent the most relevant features of natural images on 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 [23], [24]. 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 [2]. We finally use the polynomial dictionary in order to solve (8) and fill the disocclusion holes using the sparse approximation of unknown signal values.

IV. Experiments

We have evaluated the performance of our method by comparing our results with the exemplar based methods described in Joint Texture-Depth Pixel Inpainting (JTDI) [14] and Depth-Guided Disocclusion Inpainting (DGDI) [19] algorithms. JTDI inpaints texture and depth images simultaneously while DGDI first inpaints the depth image and guides the texture inpainting using the full synthesized depth map. We have therefore provided as input the disoccluded depth map to JTDI, and the complete depth map we have synthesized using our depth map inpainting method to DGDI. We also skip the *split* search and dealing with object aliasing steps in [19] since they require the use of the reference depth and texture images. In our case of zooming in towards the scene we cannot use the reference images as sources for exemplar or object boundary search, as the texture of expanding objects do not have to be identical with the reference texture.

We first learn the spectral graph dictionary as described in [2] and [24]. After moving closer towards the scene, we fill the expansion holes using the algorithm in [2] and move onto inpainting the disocclusion holes. At each iteration of our inpainting method, we first search the disocclusion hole boundary $\delta\mathcal{V}_0$ and find the pixel $\hat{p}$ with the highest priority value according to equation (1). We then choose an initial square area centered at $\hat{p}$ to inpaint. We fix the dimensions of this area by selecting $k = 9$. We also fix the sparsity constraint T_0 to be 10% of the total number of nodes $|\hat{\mathcal{V}}_s|$ for each graph $\mathcal{G}_{\hat{\mathcal{V}}_s}$. We test our algorithm on 12 images from the Middlebury dataset [25] and present example visual results from Art image in Figure 3. PSNR and SSIM [26] measurements are depicted in Table I.

TABLE I
PSNR and SSIM values for the inpainting of target view disocclusion holes using methods JTDI [14], DGDI [19] and our method.

	PSNR			SSIM		
	JTDI	**DGDI**	**Our method**	**JTDI**	**DGDI**	**Our method**
Art	25.34	26.15	26.53	0.8462	0.8498	0.8582
Baby	33.04	32.78	32.68	0.9053	0.9011	0.9033
Books	27.36	26.97	27.27	0.8498	0.8445	0.8499
Bowling	29.09	32.36	32.70	0.9252	0.9356	0.9412
Dolls	27.67	28.10	28.35	0.8738	0.8735	0.8764
Lampshade	32.17	33.95	33.73	0.9314	0.9392	0.9429
Laundry	25.78	26.57	26.12	0.8591	0.8647	0.8599
Middlebury	28.10	27.51	27.74	0.9229	0.9240	0.9276
Moebius	29.24	30.47	30.31	0.8769	0.8853	0.8848
Monopoly	28.79	29.10	28.87	0.9103	0.9113	0.9120
Reindeer	26.94	28.19	27.82	0.8481	0.8649	0.8671
Rocks	29.84	30.08	30.27	0.8598	0.8613	0.8632

The graph representation combined with the spectral graph dictionary in our regularization framework yields a consistent inpainting of the disocclusion holes. Table I indicates our SSIM measures are in general higher than the ones of the competitor algorithms. Graph based representation provides us with extra information compared to patch based methods, since it preserves both grid nodes and projected nodes from the reference image. By embedding the geometry of the scene within the graph we are able to prevent layer blending and bleeding of pixels at layer boundaries. Our learned spectral graph dictionary has the advantage of preserving the common spectral components of natural images, hence image completion is achieved with high quality.

ISBN 978-90-827970-1-5 © EURASIP 2018

We see that JTDI suffers from layer blending, as there are some foreground objects copied to the background in various regions throughout the figures. The priority term still allows foreground patches to be inpainted first, and exemplars are copied regardless of their respective depth values resulting in inconsistencies. DGDI overcomes this problem with a depth aware confidence term, but performs worse than our method on object boundaries, as can be seen on Figure 3. In [19] authors propose to deal with object aliasing at boundaries by using additional constraints on synthesized and reference image geometries, however the arrangement of layers in our framework is different due to the expansion and therefore this approach cannot be employed in our zooming in problem.

Both JTDI and DGDI perform better at regions where there are strong gradients to be propagated, with JTDI having more artifacts due to foreground propagation. The tensor based data term of DGDI prioritizes patches containing contours normal to the disocclusion hole boundary, and exemplars continue linear structures well. Our method fails to convey such strong gradients, regardless of using a data term that prioritizes contours. Future work is to overcome this weakness on areas comprising strong gradients by favoring a combined approach using our framework and exemplar based methods.

V. Conclusion

In this work, given a reference texture and depth image, we have filled the disocclusion holes within a synthesized viewpoint located closer the scene. We proposed a graph-based framework which simultaneously embeds the topology of the scene and signal values. The graph representation allowed us to separate objects in different layers and consistently propagate known signal values onto holes. We have used a parametric dictionary trained on multiple graphs, with atoms carrying the common characteristics of natural images. PSNR and SSIM measures are comparable to state of the art methods. Visual results show that our dictionary-based regularization model under sparsity constraints is able to preserve the details in the target view while preventing layer blending and bleeding of pixels at layer boundaries.

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] P. Akyazi and P. Frossard, "Graph-based interpolation for zooming in 3d scenes," in *Signal Processing Conference (EUSIPCO), 2017 25th European*. IEEE, 2017, pp. 736–767.

[3] 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.

[4] A. Levin, A. Zomet, and Y. Weiss, "Learning how to inpaint from global image statistics." in *ICCV*, vol. 1, 2003, pp. 305–312.

[5] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–I.

[6] D. Tschumperle and R. Deriche, "Vector-valued image regularization with pdes: A common framework for different applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 506–517, 2005.

[7] M. Ghoniem, Y. Chahir, and A. Elmoataz, "Geometric and texture inpainting based on discrete regularization on graphs," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*. IEEE, 2009, pp. 1349–1352.

[8] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on image processing*, vol. 13, no. 9, pp. 1200–1212, 2004.

[9] O. Le Meur, J. Gautier, and C. Guillemot, "Examplar-based inpainting based on local geometry," in *2011 18th IEEE International Conference on Image Processing*. IEEE, 2011, pp. 3401–3404.

[10] N. Komodakis and G. Tziritas, "Image completion using efficient belief propagation via priority scheduling and dynamic pruning," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2649–2661, 2007.

[11] K. He and J. Sun, "Statistics of patch offsets for image completion," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 16–29.

[12] P. Buyssens, M. Daisy, D. Tschumperlé, and O. Lézoray, "Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions," *IEEE transactions on image processing*, vol. 24, no. 6, pp. 1809–1824, 2015.

[13] I. Daribo and B. Pesquet-Popescu, "Depth-aided image inpainting for novel view synthesis," in *Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on*. IEEE, 2010, pp. 167–170.

[14] S. Reel, G. Cheung, P. Wong, and L. S. Dooley, "Joint texture-depth pixel inpainting of disocclusion holes in virtual view synthesis," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*. IEEE, 2013, pp. 1–7.

[15] S. S. Yoon, H. Sohn, Y. J. Jung, and Y. M. Ro, "Inter-view consistent hole filling in view extrapolation for multi-view image generation," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2883–2887.

[16] L. Ma, L. Do, and P. H. de With, "Depth-guided inpainting algorithm for free-viewpoint video," in *Image Processing (ICIP), 2012 19th IEEE International Conference on*. IEEE, 2012, pp. 1721–1724.

[17] J. Gautier, O. Le Meur, and C. Guillemot, "Depth-based image completion for view synthesis," in *3DTV Conference: The True Vision-capture, Transmission and Display of 3D Video (3DTV-CON), 2011*. IEEE, 2011, pp. 1–4.

[18] X. Xu, L.-M. Po, C.-H. Cheung, L. Feng, K.-H. Ng, and K.-W. Cheung, "Depth-aided exemplar-based hole filling for dibr view synthesis," in *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*. IEEE, 2013, pp. 2840–2843.

[19] P. Buyssens, O. Le Meur, M. Daisy, D. Tschumperlé, and O. Lézoray, "Depth-guided disocclusion inpainting of synthesized rgb-d images," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 525–538, 2017.

[20] E. d'Angelo and P. Vandergheynst, "Towards unifying diffusion and exemplar-based inpainting," in *2010 IEEE International Conference on Image Processing*. IEEE, 2010, pp. 417–420.

[21] V. Sairam, R. R. Sarma, S. Balasubramanian, and A. S. Hareesh, "A unified framework for geometry and exemplar based image inpainting," in *Image Information Processing (ICIIP), 2013 IEEE Second International Conference on*. IEEE, 2013, pp. 511–515.

[22] W. Liu, L. Ma, B. Qiu, and M. Cui, "Stereoscopic view synthesis based on region-wise rendering and sparse representation," *Signal Processing: Image Communication*, vol. 47, pp. 1–15, 2016.

[23] 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.

[24] 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.

[25] 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.

[26] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.