EFFICIENT MRF-BASED DISOCCLUSION INPAINTING IN MULTIVIEW VIDEO

Beerend Ceulemans[1,2], Shao-Ping Lu[1,2], Gauthier Lafruit[3], Peter Schelkens[1,2], Adrian Munteanu[1,2]

[1]iMinds VZW, Ghent, Belgium,
[2]ETRO (Department of Electronics and Informatics), Vrije Universiteit Brussel,
[3]LISA (Laboratories of Image, Signal processing and Acoustics), Université Libre de Bruxelles

ABSTRACT

View synthesis using depth image-based rendering generates virtual viewpoints of a 3D scene based on texture and depth information from a set of available cameras. One of the core components in view synthesis is image inpainting which performs the reconstruction of areas that were occluded in the available cameras but are visible from the virtual viewpoint. Inpainting methods based on Markov random fields (MRFs) have been shown to be very effective in inpainting large areas in images. In this paper, we propose a novel MRF-based inpainting method for multiview video. The proposed method steers the MRF optimization towards completion from background to foreground and exploits the available depth information in order to avoid bleeding artifacts. The proposed approach allows for efficiently filling-in large disocclusion areas and greatly accelerates execution compared to traditional MRF-based inpainting techniques. The experimental results show that view synthesis based on the proposed inpainting method systematically improves performance over the state-of-the-art in multiview view synthesis. Average PSNR gains up to 1.88 dB compared to the MPEG View Synthesis Reference software were observed.

Index Terms— Multiview video, view synthesis, disocclusion inpainting, Markov Random Field

1. INTRODUCTION

The advent of autostereoscopic displays providing horizontal parallax facilitates the visualization and interpretation of complex data, and opens new opportunities in numerous domains, such as 3D media creation, management and distribution, digital signage, medical visualization, augmented reality, gaming, to name a few. These applications require the content to be acquired from many viewpoints, raising many questions related on how to actually record, process and transmit such data. A major role in this context is played by view synthesis methods which are being used in order to generate novel camera viewpoints based on a limited set of video inputs. View synthesis can be applied for obvious tasks such as the creation of super-multiview content starting from a small number of original cameras needed in order to feed autostereoscopic displays [1–3]. View synthesis has also the potential to adjust the baseline of stereoscopic video to serve diverse display devices ranging from mobile devices to large cinema screens [4]. Furthermore, view synthesis has already been successfully used in order to create better prediction signals in 3D video coding systems and thereby improve their compression performance [5–7].

MPEG-FTV, an ad-hoc group within the MPEG community, recently issued a call for evidence on super-multiview and free-navigation technologies [8]. The call targets the design of (i) better compression methods for super-multiview content in dense but not necessarily linear camera setups, and (ii) new view synthesis techniques that can handle large and non-linear camera arrangements. The group also maintains a Depth Estimation Reference Software (DERS) [9] and View Synthesis Reference Software (VSRS) [10] representing the state-of-the-art in the field.

When depth information is available, arbitrary virtual viewpoints can be generated using depth image-based rendering (DIBR) techniques. Using the depth map and the camera calibration matrices, pixels from known reference cameras can be projected onto the imaging plane of a desired virtual camera. However, rendering a virtual viewpoint usually uncovers a part of the scene that was occluded for some or all of the reference cameras. The rendered image will therefore contain holes that need to be concealed in order to provide a pleasant user experience. In image inpainting, many state-of-the-art methods exist that are designed to remove unwanted parts of an image by seamlessly copying existing image structures in their place [11–14]. However, directly applying these methods to conceal disoccluded areas usually does not yield acceptable results. Various extensions of these algorithms have been investigated, e.g. the classical PatchMatch algorithm [14] has been applied to view synthesis in [15] and the Markov random field (MRF) -based inpainting method of [13] was adapted for disocclusion filling in [16, 17]. Other works focus on graph-based reconstruction techniques [18, 19] or superpixel segmentation [20].

In this paper, we propose an MRF-based disocclusion filling method that builds on [13, 16, 17]. The original method of [13] is designed for image inpainting or texture synthesis. However, even with speed-up enhancements such as fre-

quency domain computations and multi-scale processing, the method is relatively slow when directly applied for disocclusion inpainting. Moreover, it generates artifacts by bleeding pixels from foreground objects into the background. In [16], the method is essentially extended by disabling the edge of the MRF that lies on the boundary of a foreground object and by incorporating depth information in the cost function. While avoiding the bleeding artifact, the method is not stated to have gained a significant speedup. In [17], an additional extension is proposed that limits the number of patches that need to be evaluated per node in the MRF, which greatly reduces computation time. We further build on these state-of-the-art results. In this paper we propose to steer the search more in the direction of the camera movement by constraining the selection of candidate labels prior to the optimization. Additionally, we introduce a new easy-to-compute and intuitive priority function to favor MRF nodes that connect to known background regions.

2. PROPOSED SYNTHESIS METHOD

2.1. 3D warping

In order to warp known pixels from a real camera viewpoint, our approach follows the technique that is used in VSRS [10]. First, the depth map of the original view is warped to the virtual camera. In this initial warping step, small cracks are removed using median filtering. Next, each virtual pixel that has a valid depth value is projected to the reference camera and assigned to the corresponding pixel value. In this step, unlike [10], we avoid sampling colors from the edges of foreground objects in order to avoid ghosting artifacts. This is achieved by performing Canny edge detection on the reference depth map. If a virtual pixel is warped to an edge pixel, it is not assigned a color. The reason for this is that depth values at object boundaries are often ill-defined. Even in perfect depth maps that are rendered from computer graphics content, these edge-pixels are often a mix of foreground and background colors. In [10], some morphological operators are applied to the disocclusion holemap that will be used for inpainting; however, as shown in the example of Fig. 1, this is not sufficient to suppress edge pixels that are warped in the background. Additionally, we also remove spurious pixels or blobs of pixels that are warped into a large hole. Erosion and dilation operators are applied to the disocclusion holemap and pixels that are removed by these operations are marked as unreliable, so their value will be estimated in the inpainting stage.

2.2. Disocclusion inpainting

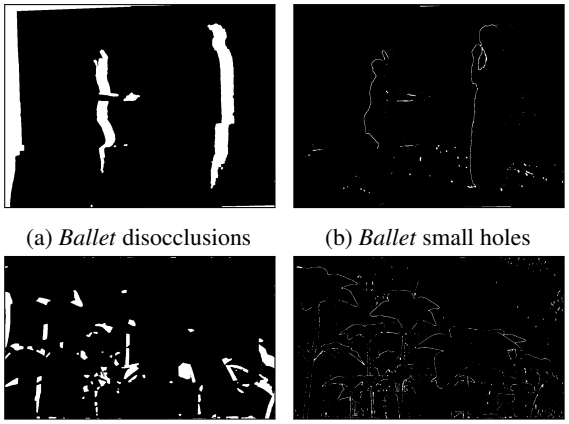
After warping all pixels from one or more reference cameras, regions with unknown pixel values in the virtual image will still remain. We will refer to these areas as holes. The holes can be classified in two categories: the first class consists of very small groups or thin lines of missing pixels while the holes in the second class are considerably larger. The small holes originate from the discrete nature of the data and can be filled-in fairly easily using conventional techniques. However, inpainting the large holes is challenging. In these regions, a part of the background that is occluded for the reference camera now becomes visible (*disoccluded*) from the virtual viewpoint. In VSRS [10], both classes are dealt with by inpainting using Telea's method [21]. However, this method does not work well for large holes, calling for the design of improved techniques for disocclusion inpainting in multiview video. In our approach, we roughly separate the small and large holes by applying a top-hat transform using a circular structuring element of radius 7 to the binary holemap. After this transform, only the small holes remain, and when subtracted from the full holemap, we obtain the disocclusion holes.

Fig. 1: Warping edge pixels may produce ghost artifacts (left [10]), the proposed method (right) avoids this.

The separated hole maps of "BigBuckBunny_Flowers" and "Ballet" are shown in Fig. 2. Small holes are inpainted using Telea's method [21], as implemented in OpenCV, while for the larger holes we propose a patch-based method, as described in the following section.

2.2.1. MRF-based disocclusion inpainting

In order to faithfully reconstruct disoccluded background regions, we propose a patch-based method. A grid of overlapping patches of size $w \times w$ is defined over the image. For each patch that is not fully known, we aim to fill-in the missing pixel values in order to obtain a visually pleasing result. Following the notation of [13], we define an MRF as a set of nodes $\mathcal{V}$ and a set of edges $\mathcal{E}$ that makes up a 4-neighborhood system. Each node p_i can be assigned to be filled in with a particular patch x_i that does not contain any missing pixels. $\mathbf{x}$ represents a vector containing a labeling x_i for all $p_i \in \mathcal{V}$.



(a) *Ballet* disocclusions (b) *Ballet* small holes

(c) *BigBuckBunny* disocclusions (d) *BigBuckBunny* small holes

Fig. 2: Holemaps of frame 1 of *Ballet* (top) and *BigBuckBunny_Flowers* (bottom)

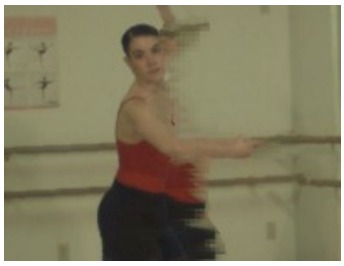


Fig. 3: A foreground object bleeds into the background using traditional image inpainting. (Figure taken from [16].)

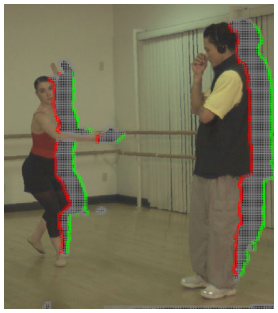


Fig. 4: MRF nodes on the edge of a foreground object are not allowed to compute their local evidence term $V_i(\cdot)$ (red), while the nodes on the opposite side of the grid are initialized with a Z-bonus of 1 (green) to encourage filling from the background and avoid the bleeding artifact (Fig. 3).

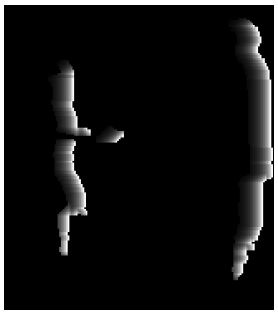


Fig. 5: The Z-bonus values for the nodes in the same region of interest as Fig. 4 after the first forward pass of the p-BP algorithm. Brighter pixels indicate a higher values.

The energy of the MRF is defined as:

$$E(\mathbf{x}) = \sum_{p_i \in \mathcal{V}} V_i(x_i) + \sum_{(i,j) \in \mathcal{E}} V_{ij}(x_i, x_j) \qquad (1)$$

where $V_i(x_i)$ denotes the sum of squared differences (SSD) between any known pixels in the existing patch at node p_i and the corresponding pixel in the patch that is assigned to fill the missing pixels. Note that this means that for nodes that have no known pixels, this term is always 0. Similarly, $V_{ij}(x_i, x_j)$ denotes the SSD within the overlap when patches x_i and x_j are assigned to nodes p_i and p_j, respectively.

The solution to the inpainting problem, $\mathbf{x}^*$, is then defined to be the labeling that minimizes this energy:

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}}\, E(\mathbf{x}) \qquad (2)$$

In traditional image inpainting using MRFs, any fully known $w \times w$ size patch that can be extracted from the image is a valid candidate for assignment to any MRF node. This usually means that all MRF nodes have a number of potential labels that is proportional to the number of pixels in the image and this makes the evaluation of all possible combinations infeasible. However, due to the nature of the large disocclusion holes, we know that the missing pixels should belong to the background and, if the camera warp corresponds to a horizontal movement, candidate patches should be searched mainly in the horizontal direction. This greatly reduces the number of candidate labels per node. Similar to [16], the nodes around a foreground object are no longer allowed to compute their local evidence $V_i(x_i)$ in order to avoid foreground bleeding - see Fig. 3. Unlike [16] we however do not include an additional penalty on selecting non-adjacent labels for adjacent nodes. A similar effect is achieved if we follow the approach of [17] and limit the allowed labels for each node to only a local window. However, [17] considers a square 100×100 window for this candidate selection while we want to limit the vertical search so we typically work with rectangular windows. This ensures that the filling process is constrained to follow the camera motion and it additionally limits the amount of labels that can be selected, thus speeding up the optimization.

In classical priority belief propagation (p-BP) [13], the MRF nodes send their messages according to a priority function which is updated dynamically. The priority at a particular iteration is computed from the belief values and basically

encodes how close the node is to make a decision. More confident nodes send their messages first and in this way the convergence of the algorithm can be accelerated. We follow the priority definition of [13] but we offset the priority with a *Z-bonus*. The Z-bonus is initially set to 0 for all nodes. Then, while we scan the MRF to find out for which nodes we want to disable the $V_i(\cdot)$ term, we give a Z-bonus of 1 to the nodes on the other side of the MRF-grid. The process is depicted in Fig. 4. Note that holes that lie within the same depth plane do not get Z-bonusses as both sides of the hole are equally reliable in these situations. Our priority function is defined as:

$$P(p_i) = Z\text{-}bonus(p_i) + \frac{1}{|\mathbb{CS}(p_i)|} \tag{3}$$

The second term is the priority definition of [13] and it is the inverse of the cardinality of the so-called confusion set of a node. The confusion set is the set of labels for which the belief is larger than a threshold plus the maximum belief value of that node.

Using the priority as defined in eq. 3 as an ordering function, we apply the p-BP algorithm to send messages between the MRF nodes. Each iteration, the algorithm makes a forward and backwards pass through the set of nodes, having each visited node send its messages to its neighbors if they have not been visited before in the same pass. In the forward pass, we also include the propagation of our Z-bonus. A node that sends messages to its neighbors during the forward pass also propagates 80% of its Z-bonus value. This way, we ensure that the scheduling function keeps prioritizing nodes on the edge of the MRF that intersects with the background of the scene. The Z-bonus is a very simple and intuitive addition to the priority function and its computation creates no overhead. Z-bonus values after a first forward pass of p-BP are shown in Fig. 5.

3. RESULTS

We evaluated the proposed method on the well-known *Ballet* video sequence provided by Microsoft [22]. This sequence has a resolution of 1024×768 and is accompanied by depth maps estimated from stereo. We also performed experiments on the *BigBuckBunny_Flowers* and *BigBuckBunny_Butterfly* sequences, provided by Holografika in the context of the MPEG-FTV CfE [8]. These sequences contains computer generated 3D content at a resolution of 1280×768 with ground truth depth. 79 viewpoints of 121 frames are provided and the cameras are numbered $6 \rightarrow 84$.

We extrapolated camera 5 in the Ballet sequence to camera 4 and visually compare against the ground truth as well as the state-of-the-art methods of [16] and [17]. This visual comparison is shown in Fig. 8. For the BigBuckBunny sequences, we rendered all 121 frames for the 72 camera views that are requested by the MPEG-FTV CfE [8] by interpolating them from the two closest reference cameras. Cameras 6, 19, 32, 45, 58, 71 and 84 are assumed to be known and the intermediate cameras are synthesized. Since ground truth information is available, we can objectively compare the quality of the synthesized views using the peak signal-to-noise ratio (PSNR).

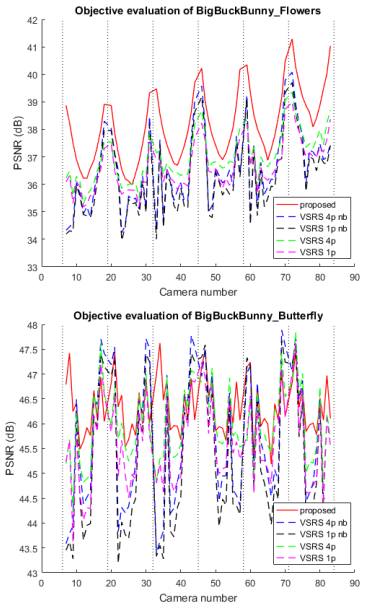


Fig. 6: Objective comparison against VSRS [10] on the Big Buck Bunny sequences. PSNR gains are averaged out over all the frames in the sequence. (top) Flowers, (bottom) Butterfly

Figure 6 shows for all virtual cameras the PSNR values of the proposed method and four configurations of VSRS [10], averaged out over all frames. In the VSRS settings, we evaluate single versus quarter *pixel precision* in the warping and turning the *view blending* option on or off. The PSNR analysis shows that our method consistently improves on VSRS when synthesizing views that are farther away from the reference cameras. Within the VSRS results, pixel precision does not appear to have a large impact for this particular sequence. The view blending option however does prove to be advantageous. Overall, our method yields an average PSNR gains of 1.15 to 1.89 dB with respect to the reference VSRS settings on the Flowers sequence and 0.37 to 1.08 dB on the Butterfly sequence. A visual result is depicted in Fig. 7 where camera 12 is interpolated from cameras 6 and 19 using both the proposed method and the best performing VSRS setting.

During our experiments, we found that the proposed method is substantially faster than the classical p-BP algorithm as presented in [13]. Our reference implementation of [13] needs around 20 minutes to process a single frame while the proposed method reduces this time to the order of tens of seconds. This speedup is mainly caused by limiting the candidate labels for an MRF node to its local background region.

(a) ground truth

(b) result from [10]

(c) proposed method

Fig. 7: Visual comparison of our method against ground truth and the View Synthesis Reference Software [10].

4. CONCLUSION

This paper proposes a method to render novel viewpoints from sparse and non-linear multiview-plus-depth content in order to address the needs of advanced multimedia systems that deliver free-viewpoint or super-multiview content. A new MRF-based inpainting method is proposed that exploits available depth information in order to avoid bleeding artifacts. Using a simple but intuitive novel priority-function, the priority belief propagation algorithm is accelerated. Visual results are competitive with the state-of-the-art and improvements in PSNR up to 1.88 dB on average have been shown with respect to the MPEG View Synthesis Reference Software.

(a) ground truth

(b) result from [16]

(c) result from [17]

(d) proposed method

Fig. 8: Visual comparison of our method against ground truth and the MRF-based methods of [16] and [17]. (Figures were taken from the papers.)

References

[1] Byoungho Lee, "Three-dimensional displays, past and present," *Physics today*, vol. 66, no. 4, pp. 36–41, 2013.

[2] Hakan Urey, Kishore V Chellappan, Erdem Erden, and Phil Surman, "State of the art in stereoscopic and autostereoscopic displays," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 540–555, 2011.

[3] Tibor Balogh, "The holovizio system," in *Electronic Imaging*. International Society for Optics and Photonics, 2006, pp. 60550U–60550U.

[4] Robert T Held and Martin S Banks, "Misperceptions in stereoscopic displays: a vision science perspective," in *Proceedings of the 5th symposium on Applied perception in graphics and visualization*. ACM, 2008, pp. 23–32.

[5] Marek Domanski, Olgierd Stankiewicz, Krzysztof Wegner, Maciej Kurc, Jacek Konieczny, Jakub Siast, Jakub Stankowski, Robert Ratajczak, and Tomasz Grajek, "High efficiency 3d video coding using new tools based on view synthesis," *IEEE Transactions on Image Processing*, vol. 22, no. 9, pp. 3517–3527, 2013.

[6] Anthony Vetro and Dong Tian, "Analysis of 3d and multiview extensions of the emerging hevc standard," in *SPIE Optical Engineering + Applications*. International Society for Optics and Photonics, 2012, pp. 84990Y–84990Y.

[7] Feng Zou, Dong Tian, Anthony Vetro, Huifang Sun, Oscar C Au, and Shogo Shimizu, "View synthesis prediction in the 3-d video coding extensions of avc and hevc," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 10, pp. 1696–1708, 2014.

[8] N15733, "Call for evidence on free-viewpoint television: Super-multiview and free navigation," October 2015.

[9] Krzysztof Wegner, Olgierd Stankiewicz, Masayuki Tanimoto, and Marek Domanski, "Enhanced depth estimation reference software (DERS) for free-viewpoint television," October 2013, M31518.

[10] Krzysztof Wegner, Olgierd Stankiewicz, Masayuki Tanimoto, and Marek Domanski, "Enhanced view synthesis reference software (VSRS) for free-viewpoint television," October 2013, M31520.

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

[12] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf, "Image completion using planar structure guidance," *ACM Transactions on Graphics*, vol. 33, no. 4, pp. 129, 2014.

[13] Nikos Komodakis and Georgios 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.

[14] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Transactions on Graphics*, vol. 28, no. 3, pp. 24, 2009.

[15] Shaoping Lu, Jan Hanca, Adrian Munteanu, and Peter Schelkens, "Depth-based view synthesis using pixel-level image inpainting," in *International Conference on Digital Signal Processing*. IEEE, 2013, pp. 1–6.

[16] Julian Habigt and Klaus Diepold, "Image completion for view synthesis using markov random fields and efficient belief propagation," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2013, pp. 2131–2134.

[17] Tijana Ruzic, Ljubomir Jovanov, Hiep Quang Luong, Aleksandra Pizurica, and Wilfried Philips, "Depth-guided patch-based disocclusion filling for view synthesis via markov random field modelling," in *International Conference on Signal Processing and Communication Systems (ICSPCS)*. IEEE, 2014, pp. 1–9.

[18] Yu Mao, Gene Cheung, Antonio Ortega, and Yusheng Ji, "Expansion hole filling in depth-image-based rendering using graph-based interpolation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 1859–1863.

[19] Yu Mao, Gene Cheung, and Yusheng Ji, "Image interpolation for dibr viewsynthesis using graph fourier transform," in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2014*, July 2014, pp. 1–4.

[20] Tomoyuki Tezuka, Mehrdad P. Tehrani, Kazuyoshi Suzuki, Keita Takahashi, and Toshiaki Fujii, "View synthesis using superpixel based inpainting capable of occlusion handling and hole filling," in *Picture Coding Symposium (PCS)*, May 2015, pp. 124–128.

[21] Alexandru Telea, "An image inpainting technique based on the fast marching method.," *Journal of graphics tools*, vol. 9, pp. 23–31, 2004.

[22] Lawrence C Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski, "High-quality video view interpolation using a layered representation," in *ACM Transactions on Graphics (TOG)*. ACM, 2004, vol. 23, pp. 600–608.