Depth-Adaptive Inpainting Algorithm for 3D Video Sequences

Cordula HEITHAUSEN, Fabian JÄGER

Institute of Communications Engineering, RWTH Aachen University, 52056 Aachen, Germany

{heithausen, jaeger}@ient.rwth-aachen.de

Abstract. *Inpainting methods attempt to reconstruct lost or destroyed image information as plausibly and sensibly to the human eye as possible. Information from the neighboring area is extrapolated into unkown image regions. This technique is also employed in view synthesis for 3D videos as well. Warping given scene views to other viewingpoints causes disocclusion artifacts to occur in the synthesized image. Such artifacts need to be filled by means of inpainting. In this paper, a new inpainting approach is presented. Its algorithm specifically adjusts to the application of 3D video sequences by exploiting additional depth information provided by depth maps, one for each corresponding texture frame.*

Keywords

inpainting, inpainting algorithm, 3D video sequences, three-dimensional, depth map, view synthesis.

1. Introduction

3D video coding is not only applied in stereoscopy and autostereoscopic display rendering but also in the development of Free Viewpoint Video (FVV), which allows the user to observe a scene from an arbitrary point of view. Necessary to that end is a view-synthesis algorithm computing various viewer perspectives positioned inbetween those already known. Three-dimensional illusion of pictures and videos requires at the minimum two given views from different perspectives. In addition to that, depth information encoded in depth maps is needed for the synthesis. The process of synthesizing new views involves warping preexisting scene views according to a sought angle of vision. Due to that warp, artifacts emerge within the synthesized image. More precisely, gaps without any texture information will occur, known as dissocclusions.

The most simple attempt to inpaint such gaps is by the method of *Clamping*, which is the transfer of the last known texture values at the edge of the unknown image region to all the sought pixels within that region. If anything, this technique is suitable for very small image gaps as the visual result of constant color reconstruction usually appears unnatural to the human eye. In *Diffusion-based inpainting* the border values of the known image region are used for interpolation. The visual appearance of color diffusion seems more natural than the one achieved by clamping. Two diffusion-based inpainting approaches by Alexandru Telea [3] and Bertalmio [5] served as a foundation for the development of the new inpainting algorithm. However, both clamping and diffusion-based inpainting methods lose information about contiguous structure or texture. This will be illustrated later on by a comparison between these inpainting techniques and the new algorithm introduced hereafter. *Template Matching* [1] continues given structures and textures inwardly as it fills the sought image region with complete texture blocks instead of pixels. The single best matching pixel block within the already known image region, possibly containing some structure or texture, is detected. That reference block is copied to the unknown pixels alongside the edge of the gap. Inpainting procedures such as Template Matching can be optimized by applying a *Segmentation through Structural Analysis* [2] beforehand. By this means a recognition of coherent image segments is employed to limit the search area for reference blocks. The inpainting result benefits from the segmentation due to the fact that the blurring of logically coherent image parts is prevented.

The inpainting algorithm presented in this paper is based on the method of template matching, modified and amended to the application of 3D video sequences with depth maps. Instead of filling the unknown pixel blocks line by line or column by column, pixel-specific priority values subserve the determination of the inpainting-order, thereby avoiding the loss of essential structures. The computation of those structure-preserving prioritiy values descends from [1] as well and is outlined in Section 2.2. Compared to conventional inpainting, the supplement most crucial to applying the algorithm to 3D pictures and videos is the consideration of depth information.

This paper is organized as follows. Section 2 provides a step-by-step overview of the inpainting algorithm. A detailed introduction of the Depth Adaption is given in Section 3, including an optimization of the algorithm by utilization of the known warping direction. The adaptation to videos as the final extension of the algorithm is described in Section

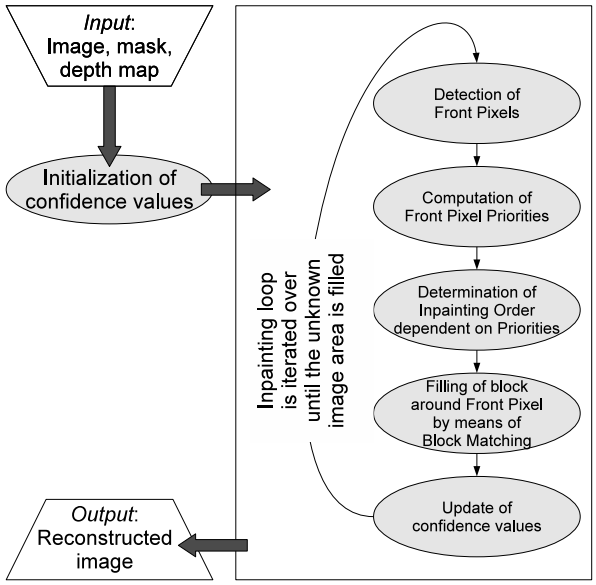

Fig. 1. The inpainting process starts with an initialization of confidence values based on the input mask. The actual inpainting is executed within a loop utilizing the input image and its corresponding depth map to synthesize a reconstructed image.

4. Some exemplary results are presented in Section 5. In Section 6 a conclusion is drawn.

2. The Algorithm

2.1. Conditions

As the algorithm presented in this paper solely treats the inpainting of texture information, the following two assumptions must be made. The inpainting of depth maps is a separate problem and not part of this diquisition. It is assumed that this problem had been solved beforehand, therefore the depth map of each texture frame is presumed entirely given. As a second condition for the algorithm, a disocclusion mask to each texture frame is available clearly defining the unknown parts of the image.

2.2. Procedure

The basic procedure of the presented inpainting algorithm consists of a one-time initialization step executed at the beginning as well as an iterative part. The initial step sets the confidence parameters according to a mask differentiating between the known and the unknown image areas. The iterative part of the algorithm, realized as an inpainting loop, is repeated until all sought pixels have been filled in, as illustrated in Figure 1.

Initialization

To each image pixel an individual value is assigned, henceforth referred to as *confidence*. The initialization step of the presented algorithm implies setting the confidence to an initial value. Pixels belonging to the known image areas receive a confidence $C = 1$ while the masked pixels of the unknown region receive a confidence $C = 0$. Thus the initial confidence distribution is the inverse of the disocclusion mask.

Inpainting Loop

After the initialization, the actual inpainting-process is repeated iteratively until the algorithm terminates. Each iteration starts with the determination of so-called *front pixels*, which are those unknown pixels located on the edge to the known image region. After their identification, these front pixels are sorted by a pixel-specific priority value to obtain a sensible inpainting order. Afterwards, a window of fixed size centered around the front pixel of highest priority is filled by means of a block-matching-approach. Finally, the confidence values of all affected pixels are updated. These three substeps are described more closely hereafter.

- **Front Pixel Priority**
  The information required to fill in a gap can only be gathered from the surrounding image regions known to the algorithm. Hence, inpainting always propagates successively from the border between known and unknown regions inwards. As a result, merely the front pixels located at this border are of interest within an inpainting step. These front pixels can be captured by subtracting the occlusion mask from a dilated occlusion mask and multiplying the image by the resulting difference. After detection of the current pixel front, a priority value for each front pixel is determined as a product of several characteristics that depend on the front pixel neighborhood. The purpose of the front pixel priority is to indicate, which front pixel should be favored in the succession of inpainting in order to conserve important image structures, such as edges leading into the unknown area. The priority value P_j, which could be supplemented with arbitrary factors further contributing to the inpainting order such as the edge angle of incidence, is computed as

$$P_j = E_j \cdot C_j \tag{1}$$

  where E_j is the *edge energy* and C_j is the *confidence* in Pixel j.
  The normalized edge energy E_j sums up the gradient energy of the surrounding area of the front pixel. The more edges are close to the front pixel, the bigger E_j will be. The factor E_j thus ensures that edges are continued into the unknown region prior to smooth regions with less contribution to the image structure.

The confidence C_j describes the ratio of known pixels to unknown pixels within the fixed-sized pixel block around a considered front pixel. The factor C_j hereby makes front pixels surrounded by many known texture pixels more likely to be selected for inpainting.

- **Block Matching**
  After the front pixel of highest priority is determined, a pixel block centered around it as well as a surrounding search area are defined. Both the block size and the size of the search area are fixed and forwarded to the algorithm in advance of the inpainting process. Within the search area all completely known blocks of the same size as the partly unknown block are compared to that sought one for the purpose of finding the best matching block. The criterion for finding that most fitted pixel block is the luminance value difference between the known pixels of the partly unknown block around the front pixel and those of same relative position within the reference block. The reference block yielding the minimum mean squared error is then chosen and referred to as the *best match*.
  Afterwards, each unknown pixel within the sought block, including the central front pixel, receives the color value of the corresponding pixel within the best match.
  In the presented algorithm, the *best match* approach is improved by a *weighted match* method superimposing multiple weighted reference blocks B_i (see equation 2). This is similar to an approach used for denoising images, presented in [6] as the Non-local Means Algorithm, which is also built upon the idea of superimposing weighted references. The individual weighting factors W_i of the weighted matches method are composed as products of three factors which are weighting functions themselves. To these belong an average luminance difference ($W_{\Delta\mathrm{lum}_i}$), the spacial distance between the reference block and the sought one (W_{dist_i}) as well as the average confidence (W_{conf_i}) (see equations 3 to 6). The individual block weights are additionally weighted by a normal distribution function to form the overall reference block. The variance is chosen to be relatively small and the conventional best match method is used as a fall back mode when no block matches with weights above a certain threshold can be found.

$$B_{\mathrm{overall}} = \sum_i W_{\mathrm{block}_i} \cdot B_i \text{ with } i \in \{blocks\} \quad (2)$$

$$W_{\Delta\mathrm{lum}_i} = \sum_j \Delta\mathrm{lum}_j \text{ with } j \in \{pixels\} \quad (3)$$

$$W_{\mathrm{dist}_i} = |\vec{x}_{\mathrm{center}_i} - \vec{x}_{\mathrm{front\ pixel}}| \quad (4)$$

$$W_{\mathrm{conf}_i} = \frac{1}{J} \cdot \sum_j C_j \text{ with } J \mathrel{\hat{=}} \text{pixels per block} \quad (5)$$

$$W_{\mathrm{block}_i} = W_{\Delta\mathrm{lum}_i} \cdot W_{\mathrm{dist}_i} \cdot W_{\mathrm{conf}_i} \quad (6)$$

- **Update of Confidence Values**
  In the end of each iteration, the confidence C of the currently considered front pixel is assigned to all recently filled in texture pixels within the sought pixel block.

3. Depth Adaptation

The involvement of the depth map in the process of inpainting embodies the novelty of this algorithm. In addition to the weighting factors mentioned above, the choice of reference blocks should furthermore depend on the depth difference between the sought pixels and those within the reference block. This enhancement serves the purpose of avoiding inpainting of textures derived from regions of the 3D scene located in the foreground of the unknown pixel block.

3.1. Depth-based Weighting

If depth weighting is assigned to the entire reference block, another trade-off problem will arise as the final reference block may either be too inaccurate or all otherwise contemplable reference blocks may be discarded in case no block with the same depth relations is available. In contrast to a blockwise weighting, the pixelwise weighting utilized in the presented inpainting algorithm allows for integrating a block only partially in the overall sum of reference blocks. After a preliminary weight of an entire reference block has been calculated in the same manner as described in Section 2.2, each pixel within that block is regarded individually. Whether or not a pixel is referenced is decided by its depth relation to the corresponding pixel at the same relative position in the sought pixel block around the front pixel, weighted by a normal distribution function. The depth relation is determined by the subtraction of the depth of the reference from the depth of the unknown pixel. A positive result therefore indicates that the reference pixel lies further back while from a negative result a foreground position of the reference pixel can be concluded. Avoiding the latter is the intention of the depth adaptation of this algorithm, precisely. To discard such foreground pixels from the final reference block, an adjustable *cut-off* is applied to the aforementioned normal distribution function (see Figure 2). The only exception to that rejection is the central reference pixel which is always included in the final block match, to ensure termination of the algorithm.

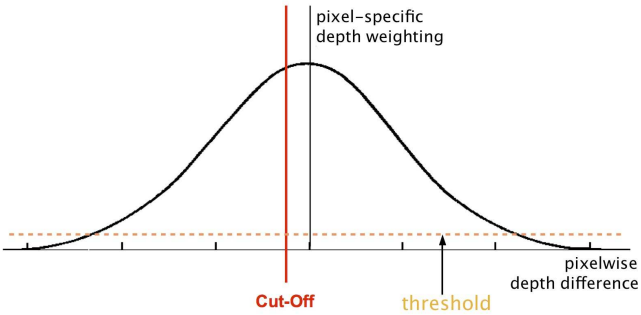

Fig. 2. Normal distributed depth weighting with cut-off.

4. Extensions

4.1. Exploitation of the Warping Direction

The warping direction, known to the algorithm beforehand, can be exploited additionally. The following two approaches make use of the warping direction, which is now assumed to be rightwards, without loss of generality. The two methods are alternative to each other.

Positioning the search area according to the warping direction

The disocclusion artifacts emerging through viewpoint-synthesis occur only to the right of the foreground objects in case of rightwards warping, and to the left in case of leftwards warping, respectively. The unknown area is thus positioned inbetween the foreground object on its left and a given background on its right. This area should only be inpainted with texture of the same or greater depth. Foreground texture being of smaller depth is irrelevant to the reference search. By shifting the initially centrically positioned window rightwards until the front pixel is left-adjusted, many more potential reference blocks, within a search area of the same size as before, are allocated, as illustrated in Figure 3.

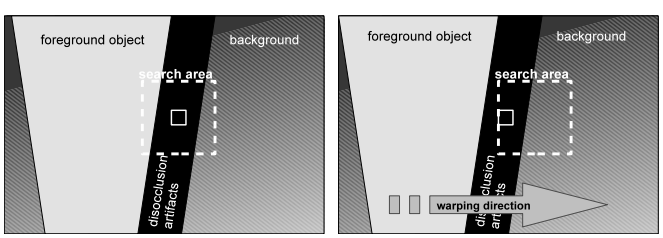

Fig. 3. Warping direction influencing search window.

Computation of the block priority according to the warping direction

Supposing that an area affected by disocclusion is relatively wide and on the right-hand side of a warped foreground object. As a result, a front pixel of highest priority, positioned far left within the disocclusion gap, might have difficulties finding suitable block references, as the search area would mostly or even entirely be spread over the unknown area providing no texture information. By employing a modified front pixel priority only allowing pixels of the right (or left, for leftwards warping) half of the pixel block to account for the confidence C as illustrated in Figure 4, the inpainting process is prevented from starting at the foreground edge.

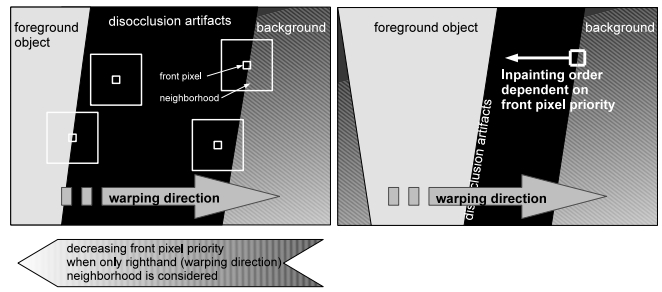

Fig. 4. Warping direction influencing front pixel priority.

4.2. Extension to Video Applications

The inpainting algorithm is finally extended for the use of 3D video sequences. As a video is a series of single images, referred to as *frames*, the described algorithm can easily be applied to each of the video frames separately. Advantageously, great similarities are prominent among most consecutive video frames except those forming a visual scene change. Due to movement in the video content, the current frame usually shows a shifted version of the images shown by its neighbored frames. This coherence is exploited in the algorithm.

The block matching and weighting described in Section 2.2 is now applied to neighbored frames as well, comparing their reference blocks within the equivalently positioned search area to the sought block of the current frame. As a result, additionally to the final reference block of the current frame $B_{\text{intra}} = B_{overall}$, one superimposed block B_{frame_k} per neighbored frame is obtained.

Next the framewise blocks receive individual weights W_{frame_k} composed of two factors, the first one being the superpositioned and normalized sum of all block weightings $W_{\text{B}_{\text{overall}}}$ (Section 2.2) and the second one resulting from the temporal distance between the frames being compared $W_{\text{dist}_{\text{frame}_k}}$ weighted by a normal distribution function (see equation 7). The resulting frame-specific weights are again superimposed and normalized as an overall inter-frame reference block. After this step, two different blocks are available alltogether, the intra-frame reference block considered as well. A combination of these two blocks, weighted and normalized, finally delivers one final inpainting reference. A chart of the weighting concept is depicted in Figure 5.

$$W_{\text{frame}_k} = W_{\text{B}_{\text{overall}}} \cdot W_{\text{dist}_{\text{frame}_k}} \tag{7}$$

$$B_{\text{inter}} = \sum_k \frac{W_{\text{frame}_k} \cdot B_{\text{frame}_k}}{W_{\text{inter}}} \text{ with } k \in \{frames\} \quad (8)$$

$$B_{\text{final}} = \frac{B_{\text{inter}} \cdot W_{\text{inter}} + B_{\text{intra}} \cdot W_{\text{intra}}}{W_{\text{inter}} + W_{\text{intra}}} \quad (9)$$

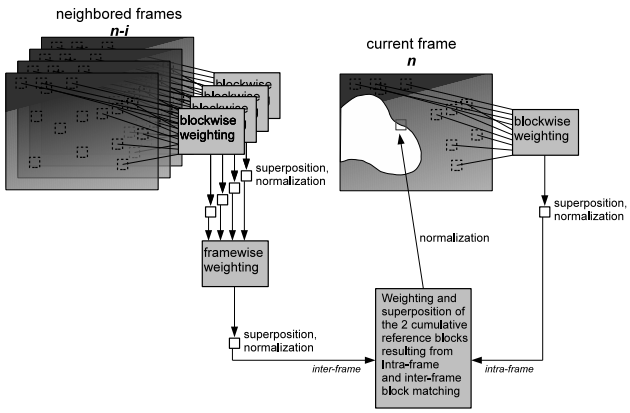

Fig. 5. Concept of video-adapted weighting.

5. Experimental Results

On page 6, some exemplary inpainting results are presented to illustrate the advantages of the new inpainting algorithm. The development of this depth-adaptive algorithm was mainly motivated by the idea of utilizing the depth map of an image for avoidance of filling disocclusion artifacts in the background with foreground texture.
The three image details below clarify the effectiveness of the presented inpating approach regarding its observance of depth-edges. In comparison to the conventional methods of clamping and diffusion, a distinctly more plausible inpainting result is revealed. Less blurriness or ghosting effects emerge as shown in Figure 6 and fine or even periodic structures, such as the edge of the roof in Figure 7 or the fence in Figure 8, are preserved.

6. Conclusion

In this paper a new inpainting algorithm is introduced, specifically designed for the reconstruction of disocclusion artifacts in 3D video sequences. The key to that is depth adaptation realized by consideration of the depth map of each frame in the determination of the inpainting order. By prohibiting block references of foreground regions more plausible inpainting results can be achieved. Moreover, the algorithm is improved by utilization of knowledge about the warping direction and exploitation of neighbored frames for a more efficient reference block search.
But there is further need for research. The algorithm could be taught to determine its parameters autonomously by means of a preceding image analysis, for instance.
There are very few cases in which more primitive inpainting approaches may lead to less visually disturbing results. A hybrid algorithm might help in such exceptions.
Also, the inpainting results are highly dependent on the quality of the depth maps, which, when not artificially synthesized but estimated, are occasionally imprecise. Therefore, depth map improvements lead to an inpainting quality enhancement as well.
Until now the development of the proposed algorithm was purely focused on attaining visually pleasing results. After achieving that goal, complexity reduction is preferable.

References

[1] A. Criminisi,P. Prez, K. Toyama. *Region Filling and Object Removal by Exemplar-Based Image Inpainting*. November 2013.

[2] Dimitar Doshkov, Patrick Ndjiki-Nya, Martin Köppel and Thomas Wiegand. *Automatic structure-aware inpainting for complex image content*. Springer-Verlag Berlin Heidelberg, 2008.

[3] Alexandru Telea. *An image inpainting technique based on the fast marching method*. journal of graphics tools, 2004.

[4] J.A.Sethian. *A fast marching level set method for monotonically advancing fronts*. Proceedings of the National Academy of Science, 1995.

[5] Vicent Caselles Coloma Ballester Marcelo Bertalmio, Guillermo Sapiro. *Image in- painting*.

[6] Jean-Michel Morel Antoni Buades, Bartomeu Coll. *A non-local algorithm for image denoising*. Computer Vision and Pattern Recognition, 2005.

About Authors…

Cordula HEITHAUSEN

was born in Tönisvorst, Germany in 1986. She studied Electrical Engineering and Information Technology at RWTH Aachen University. After her graduation in 2012 she decided to continue researching as a Ph.D. student at the Institut für Nachrichtentechnik at RWTH Aachen University. Cordula's research is focusing on algorithms and applications in image and video coding.

Fabian JÄGER

was born in Berlin, Germany in 1984. He studied Computer Engineering at RWTH Aachen University. After his graduation in 2009 he decided to continue researching as a Ph.D. student at the Institut für Nachrichtentechnik at RWTH Aachen University. Fabian's research is focusing on algorithms and applications in 3D video coding.

Fig.6. Inpainting of background on the right of a moving person, image detail of test sequence Lovebird: a) disocclusion artifacts, marked black; b) Clamping; c) Diffusions-based Inpainting; d) Depth-adaptive, Block-macthing-based Inpainting Algorithm ($\sigma_{\Delta\mathrm{luminance}} = 3$, $\sigma\Delta\mathrm{depth} = 20$, search window size 30, block size 8)

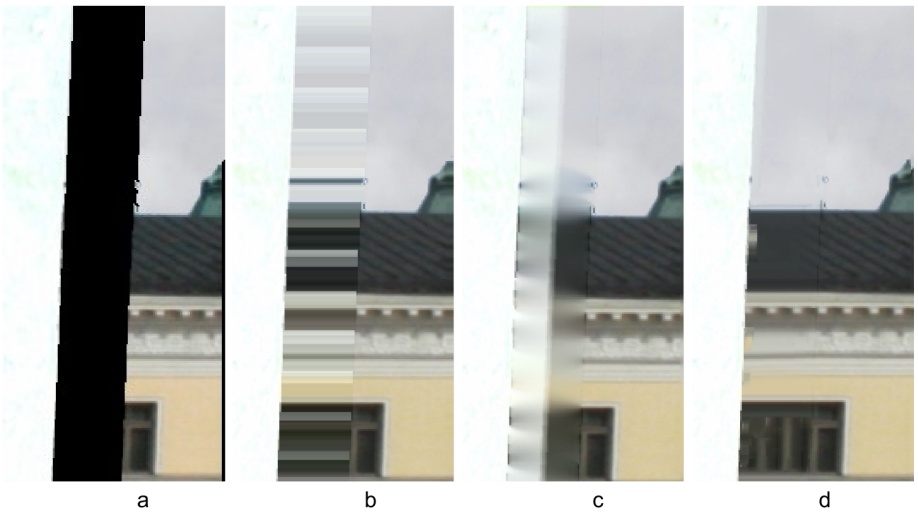

Fig.7. Inpainting of a roof in the background of a pillar, image detail of test sequence Undo Dancer: a) disocclusion artifacts, marked black; b) Clamping; c) Diffusions-based Inpainting; d) Depth-adaptive, Block-macthing-based Inpainting Algorithm

Fig.8. Inpainting of a garden fence behind a car, image detail of test sequence Poznan Street: a) disocclusion artifacts, marked black; b) Clamping; c) Diffusions-based Inpainting; d) Depth-adaptive, Block-macthing-based Inpainting Algorithm