Depth-Aware Patch-based Image Disocclusion for Virtual View Synthesis
Pierre Buyssens, Maxime Daisy, David Tschumperlé, Olivier Lézoray

To cite this version:
Pierre Buyssens, Maxime Daisy, David Tschumperlé, Olivier Lézoray. Depth-Aware Patch-based Image Disocclusion for Virtual View Synthesis. SIGGRAPH Asia, Nov 2015, Kobe, Japan. hal-01206537

HAL Id: hal-01206537
https://hal.archives-ouvertes.fr/hal-01206537
Submitted on 30 Sep 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Depth-Aware Patch-based Image Disocclusion for Virtual View Synthesis

Pierre Buyssens, Maxime Daisy, David Tschumperlé, Olivier Lézoray*
Normandie Université, UNICAEN, ENSICAEN, GREYC UMR CNRS 6072, Caen, France

Figure 1: Illustration of our inpainting algorithm on the Adirondack image [Scharstein et al. 2014]. From left to right: Synthesized view with disoccluded regions (in black, i.e., 526389 pixels to inpaint), our reconstruction result, cropped and zoomed parts of the inpainted result.

Abstract

In this paper we propose a depth-aided patch based inpainting method to perform the disocclusion of holes that appear when synthesizing virtual views from RGB-D scenes. Depth information is added to each key step of the classical patch-based algorithm from [Criminisi et al. 2004] to guide the synthesis of missing structures and textures. These contributions result in a new inpainting method which is efficient compared to state-of-the-art approaches (both in visual quality and computational burden), while requiring only a single easy-to-adjust additional parameter.


Keywords: Depth-Aided Disocclusion, Patch-based Inpainting, RGB-D Virtual View Synthesis.

1 Introduction and context

3DTV and Free-Viewpoint Rendering (FVR) have become key technologies that could stimulate the emergence of multimedia experiences such as 3D cinema, display, broadcasting ... Depth Image Based Rendering (DIBR) has then become an important feature for the synthesis of new virtual views, and consists in rendering a depth map in addition to the classical color image. Warping these images from a new point of view led to virtual synthesized views [Fehn 2004].

The major and recent problem of the so-called occluded areas arises when these images are warped: background (BG) areas that were hidden by some foreground (FG) objects in the original view have to be rendered in the synthesized view (Fig. 1, left). In this case, both color image and depth map contain holes that have to be filled.

Filling these holes is known as disocclusion and is a particular case of the more general inpainting problem.

The disocclusion methods proposed in the literature can roughly be divided into two groups: the first ones proceed to the disocclusion of both color image and depth map at the same time [Reel et al. 2013], while the second ones first inpaint the depth map, then use this reconstructed depth map to guide the disocclusion of the color image. Particularly, the methods proposed in [Gautier et al. 2011], [Daribo and Pesquet-Popescu 2010], [Yoon et al. 2014] deal with color images, considering the already inpainted depth map.

As pointed in [Yoon et al. 2014], inpainting the depth map is not a so-easy task, but is possible with dedicated algorithms: inpainting first the depth map and use it to guide the inpainting of the color image leads to more flexible approaches.

Contributions: In this paper, we also consider that the depth map has been previously inpainted. Based on the patch-based inpainting algorithm of [Criminisi et al. 2004], the proposed method revisits each of its key steps by incorporating the depth information. Each modification is argued and the proposed approach involves only one global extra parameter $\lambda$ compared to the seminal approach of [Criminisi et al. 2004]. This threshold parameter $\lambda$ discriminates adjacent pixels (according to their respective depths) into foreground and background. Particularly, two adjacent pixels $p$ and $q$ belong to the same object (foreground or background) if $|\text{depth}(p) - \text{depth}(q)| < \lambda$.

2 Revisiting Patch-based inpainting

Before detailing our proposed method, we first introduce the notations that are used throughout this Section, and draw the sketch of the patch-based inpainting algorithm of [Criminisi et al. 2004].

2.1 Notations

The original color image $I_o$ and depth map $J_o$ are warped according to an offset map to synthesize a new scene view. The resulting synthesized color image $I_s$ and depth map $J_s$ contain occlusions, i.e., parts of the image that were hidden by some foreground objects in $I_o$ and $J_o$ and that have been uncovered (thus, to be inpainted). In this paper, we consider that the depth image $J_s$ is already inpainted with a dedicated method such as the one in [Yoon et al. 2014].
\(I_s\) is considered as a function \(I_s : \mathcal{I} \rightarrow \mathbb{R}^3\) (color image) where \(\mathcal{I}\) defines the image domain, \(\Omega\) is the masked part of the image \(I_s\) (i.e., the unknown pixels to resynthesize), and \(\partial \Omega\) is the interior boundary of the mask. In the following, a patch \(\Psi_p\) centered on the pixel \(p\) is considered as a function \(\Psi_p : \mathcal{N}_p \rightarrow \mathbb{R}^3\) where \(\mathcal{N}_p \subset \mathcal{I}\) is the square support of \(\Psi_p\). Note that this patch can itself be incomplete (i.e., some of its pixels are unknown).

### 2.2 Sketch of Patch-based inpainting algorithm

The seminal greedy patch-based algorithm of [Criminisi et al. 2004] consists in several iterations of the 4 following steps:

1. A priority term is assigned to each pixel \(p \in \partial \Omega\), and is computed as \(P(p) = D(p) \times C(p)\), where \(D(p)\) is the so-called data term, and \(C(p)\) the confidence term. The first term is based on the local gradient in \(\mathcal{N}_p\) and reflects the structures that entered the mask \(\Omega\), while the second term reflects the number of valid (known) pixels in \(\mathcal{N}_p\). The pixel \(t\) with the maximum priority is chosen as the target pixel.

2. Given the target patch \(\Psi_t\) centered on \(t\), the second step consists of searching in \(\Omega\) the patch \(\Psi_t\) that minimizes the Sum of Squared Differences (SSD) among the known part of \(\Psi_t\):

\[
\Psi_t = \left\{ \Psi_p \mid \arg\min_{p \in \mathcal{N}_p \cap (\mathcal{I} - \Omega)} d_{SSD}(\Psi_t, \Psi_p) \right\}
\]

3. Paste values from \(\Psi_t\) around \(t\) in \(\Omega\):

\[
\Psi_t(q) = \Psi_t(p) \mid q \in \mathcal{N}_t \cap \Omega
\]

4. Update data and confidence terms.

The main advantages of this inpainting algorithm are its ability to first reconstruct main structures that enter the mask thanks to the priority term, as well as synthesize large portions of texture thanks to the use of patches.

We now revisit each step of this algorithm to make them use the information from the depth map.

### 2.3 Depth-aware Priority term

Since holes are mainly due to occlusions of the background by a foreground object, holes have often to be filled with background components. To achieve this goal, the most natural filling order is to start filling the background. To this end, [Yoon et al. 2014] proposed to compute the priority term as:

\[
P(p) = D(p) \alpha \times C(p) \beta \times E(p) \gamma
\]

with \(E(p)\) being the inverse of the depth map, and \(\alpha, \beta,\) and \(\gamma\) hyperparameters empirically fixed by the authors. [Reel et al. 2013] proposed the following priority term:

\[
P(p) = (C(p) + D(p) + L(p)) \times (Z_{out} - \bar{Z}_p)
\]

where \(L(p)\) is the depth variance among the target patch centered on \(p\), \(Z(p)\) its mean, and \(Z_{out} = 255\). Finally [Gautier et al. 2011] proposed to set to zero the priorities of all pixels that lie on the right side of the mask, if the warping is performed from right to left. This ad-hoc solution then forces the inpainting process to start on a given side of the mask.

What we propose in this paper avoids adding a depth-dependent extra term to the initial priority term \(P(p) = D(p) \times C(p)\). The data term \(D(p)\) is kept unchanged, and only the confidence term \(C(p)\) is modified. The rationale behind the initial confidence term is to count the number of reliable data around a pixel \(p\). We then propose to define the reliable data around a pixel \(p\) as the number of pixels that are at the same depth as \(p\) (w.r.t \(\lambda\)):

\[
C(p) = \frac{1}{|\mathcal{N}_p|} \sum_{q \in \mathcal{N}_p \cap (\mathcal{I} - \Omega)} \mathbb{1}_{|J_x(p) - J_x(q)| < \lambda}
\]

where \(|\mathcal{N}_p|\) is the size of \(\mathcal{N}_p\) (i.e., the number of pixels), and with \(C(p) = 1, \forall p \in \Omega\). Figure 2 illustrates the benefits of our proposed confidence term. All the pixels belonging to the interior boundary of the mask and that are only surrounded by foreground pixels have their confidences equal to zero. These pixels are then inpainted at the very end of the process, which is a desired behavior.

Moreover, by avoiding additional parameters in our proposed priority term, its overall sensitivity is reduced.

### 2.4 Depth-aware search scheme

The proposed lookup strategy to find the best patch that match a target patch \(\Psi_t\) is composed of 2 components:

- Since \(I_s\) is obtained from \(I_o\) by warping, we can safely claim that most of data of \(I_s\) can also be found in \(I_o\) (data(\(I_s\) \(\subset\) data(\(I_o\)). Hence, candidate patches are searched (Eq. 1) only in \(I_o\). Thanks to the offsets used for the warping, search areas can be easily defined on \(I_o\) at the right places by taking the inverse of the offsets.

- Only patches that have their depth at the same depth as \(t\) (w.r.t \(\lambda\)) are candidate patches (step similar to the one used in [Yoon et al. 2014]):

\[
\Psi_t = \left\{ \Psi_p \in I_o \mid \arg\min_{|J_x(p) - J_x(t)| < \lambda} d_{SSD}(\Psi_t, \Psi_p) \right\}
\]
2.5 Depth-aware patch copy

Once the best patch $\Psi_t$ has been found, masked pixels of $\Psi_t$ are filled with values from $\Psi_t$ (Sec. 2.2, point 3). To our knowledge, all the literature methods use this equation to fill in the masked part of $\Psi_t$. While it works well in practice when the hole has to be filled with background values only (since the foreground is unmasked), this copy scheme has a major flaw when dealing with multiple foreground objects (i.e., a foreground object which is masked by another). Figure 4 (second column) illustrates such a case, where multiple foreground objects (in addition to the background) overlap. In such a case, some foreground objects are masked, and this simple copy strategy is clearly not optimal.

To tackle these not-so-rare cases, we propose a depth-aware copy scheme, which copies values from $\Psi_t$ to $\Psi_t$ only if the adjoining depths are equal (w.r.t. $\lambda$):

$$\Psi_t(q) = \Psi_t(p) \mid q \in N_t \cap \Omega \mid |J_s(p) - J_s(q)| < \lambda$$  \hspace{1cm} (6)

Figure 4 illustrates the benefits of this depth-aware copy. The masked part is not considered as background only, and one can recover the masked foreground objects. To our knowledge, such a mask can not be properly inpainted with the state-of-the-art methods.

2.6 Dealing with object aliasing

One important problem when processing both color image and depth map, is that the color of the objects often bleeds into other objects or into the background. While the frontier between two objects is well defined in the depth map, it can be several pixels wide in the color image (Fig. 5, first row). If these so-called narrow-bands are not taken into account during the process, the inpainting algorithm is likely to copy pixels belonging to background that have such foreground color components (Fig. 5, third row, middle column).

To tackle this problem, we propose to compute the morphological gradient obtained via a dilation of the original (resp. synthesized) depth map. This gradient image is then thresholded to $\lambda (T(\cdot, \lambda))$ to obtain the narrow-bands image $NB_o$ (resp. $NB_s$) of the original (resp. synthesized) view:

$$\left\{ \begin{array}{l}
NB_o = \mathcal{T} (\delta(J_o) - J_o, \lambda) \\
NB_s = \mathcal{T} (\delta(J_s) - J_s, \lambda)
\end{array} \right.$$  \hspace{1cm} (7)

where $\delta(\cdot)$ is the dilation operator whose size essentially depends on the resolution of the image. Figure 5 (second row) displays the superposition of the narrow-bands images (in green) on the color images for both the synthesized (left) and original (right) views. These narrow-bands are then incorporated to the inpainting process. Given a target pixel $t$, two different cases appear:

- If $t \notin NB_s$, the search of $\Psi_t$ is restricted to $\Psi_p \in I_o$ such that $\Psi_p$ contains no pixels of $NB_s$.
- If $t \in NB_s$, the search of $\Psi_t$ is processed without restrictions on $NB_o$.

This simple mechanism avoids (1) the copy of bleeding pixels into unwanted parts of the mask, while (2) authorizing it at the frontier between the background and foreground objects. Figure 5 (third...
Figure 6: Inpainting comparisons. From left to right: masked image, our result, and inpainting result excerpts with [Gautier et al. 2011], with [Yoon et al. 2014], and with our method. For information, the mask sizes are 402986 pixels for the conesF image (first row), 401987 pixels for the Art image (second row), and 157257 pixels for the Midd2 image (third row). While the depth maps used for the first two rows are the groundtruth (available in the Middlebury data set), the depth map of the third row has been inpainted with our dedicated depth map inpainting algorithm (the corresponding article is under submission).

3 Evaluation

The evaluation is performed on the Middlebury stereo dataset [Scharstein et al. 2014] which offers a collection of pairs of stereo images whose size vary from 1.4M to 6M pixels. The synthesis is performed from view1 to view5 for the 2014 sub-dataset, and from view5 to view1 for the rest. For all the images, we set $\lambda = 3$.

Note that some authors [Reel et al. 2013], [Yoon et al. 2014] use the PSNR value between the inpainting result and the groundtruth as a measure of efficiency. In this paper, we provide only visual comparisons since the PSNR is definitely not suited for inpainting (because of course, a pleasant reconstruction may be very different from the groundtruth while leading to a low PSNR, and vice versa).

Figure 6 shows some of our reconstruction results in addition to inpainting result excerpts obtained with the methods of [Gautier et al. 2011] and [Yoon et al. 2014]. For the first two rows, we use the depth groundtruth to guide the inpainting, since this paper deals only with color image disocclusion. The third row shows a result with a depth map inpainted with our dedicated depth map inpainting algorithm (under submission). Thanks to the proposed depth-aware patch-based inpainting method, our inpainting results show no important leaks nor major reconstruction artifacts. It offers visually equal or better results than the state-of-the-art methods on many difficult cases implying numerous objects of different depths. Moreover, our proposed method is quite fast in practice (inpainting at roughly 1500px/s), and can be easily parallelized.

References


