



# Joint inpainting of depth and reflectance with visibility estimation

Marco Bevilacqua, Jean-François Aujol, Pierre Biasutti, Mathieu Brédif,  
Aurélie Bugeau

## ► To cite this version:

Marco Bevilacqua, Jean-François Aujol, Pierre Biasutti, Mathieu Brédif, Aurélie Bugeau. Joint inpainting of depth and reflectance with visibility estimation. ISPRS Journal of Photogrammetry and Remote Sensing, 2017, 125, pp.16–32. 10.1016/j.isprsjprs.2017.01.005 . hal-01439299

**HAL Id: hal-01439299**

**<https://hal.science/hal-01439299>**

Submitted on 18 Jan 2017

**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.

# Joint Inpainting of Depth and Reflectance with Visibility Estimation<sup>☆</sup>

Marco Bevilacqua<sup>a,b,c,d</sup>, Jean-François Aujol<sup>a</sup>, Pierre Biasutti<sup>a,b,c</sup>, Mathieu Brédif<sup>c</sup>, Aurélie Bugeau<sup>b</sup>

<sup>a</sup>*Université de Bordeaux, IMB, CNRS UMR 5251, 33400 Talence, France.*

<sup>b</sup>*Université de Bordeaux, LaBRI, CNRS UMR 5800, 33400 Talence, France.*

<sup>c</sup>*Université Paris-Est, LASTIG MATIS, IGN, ENSG, F-94160 Saint-Mande, France*

<sup>d</sup>*Bordeaux INP, IMS, CNRS UMR 5218, 33400 Talence, France.*

---

## Abstract

This paper presents a novel strategy to generate, from 3-D lidar measures, dense depth and reflectance images coherent with given color images. It also estimates for each pixel of the input images a visibility attribute. 3-D lidar measures carry multiple information, e.g. relative distances to the sensor (from which we can compute depths) and reflectances. When projecting a lidar point cloud onto a reference image plane, we generally obtain sparse images, due to undersampling. Moreover, lidar and image sensor positions typically differ during acquisition; therefore points belonging to objects that are hidden from the image view point might appear in the lidar images. The proposed algorithm estimates the complete depth and reflectance images, while concurrently excluding those hidden points. It consists in solving a joint (depth and reflectance) variational image inpainting problem, with an extra variable to concurrently estimate handling the selection of visible points. As regularizers, two coupled total variation terms are included to match, two by two, the depth, reflectance, and color image gradients. We compare our algorithm with other image-guided depth upsampling methods, and show that, when dealing with real data, it produces better inpainted images, by solving the visibility issue.

---

<sup>☆</sup>This study has been carried out with financial support from the French State, managed by the French National Research Agency (ANR) in the frame of the Investments for the future Programme IdEx Bordeaux (ANR-10-IDEX-03-02). J.-F. Aujol also acknowledges the support of the Institut Universitaire de France.

*Keywords:* Inpainting, Total Variation, Depth Maps, Lidar, Reflectance, Point Cloud, Visibility

---

## 1. Introduction

Image-based 3D reconstruction of static and dynamic scenes (Herbort and Wöhler, 2011; Seitz et al., 2006; Stoykova et al., 2007) is one of the main challenges in computer vision nowadays. In the recent years many efforts have been made to elaborate configurations and approaches, possibly requiring the employment of multiple sensors, with the final goal of generating plausible and detailed 3D models of scenes. To this end, typical optical cameras are often combined with non-visual sensors. The intermediate outputs of these hybrid systems, prior to the final scene rendering, are in general depth or depth+color images (RGB-D). Among the non-visual sensors, we can find Time-of-Flight (ToF) cameras (Kolb et al., 2010), which acquire low-resolution co-registered depth and color images at a cheap cost, and the famous Kinect (Zhang, 2012), capable to extract depth information by exploiting structural light. Another possibility is represented by lidar devices, which are used in a variety of applications and provide as output point clouds with measures of distance and reflectivity of the sensed surfaces.

This work lies in the context described and is particularly driven by the exploitation of data acquired by Mobile Mapping Systems (MMS), such as (Paparoditis et al., 2012). MMS systems are vehicles equipped with high-resolution cameras and at least one lidar sensor: their contained dimensions allow them to be driven through regular streets and acquire data of urban scenes. The data acquired is a set of calibrated and geolocated images, together with coherent lidar point clouds. The interest towards them comes from the possibility of having available, at a relatively small processing cost, the combination of depth and color information, without having to perform explicit (error-prone) reconstructions. Having a good depth estimate at each pixel, for example, would enable the possibility to perform depth-image-based rendering algorithms, e.g. (Chen et al., 2005; Schmeing and Jiang, 2011; Zinger et al., 2010). Similarly, the availability of depth information allows the insertion of virtual elements into the image, such as pedestrians or vehicles generated by a traffic simulation (Brédif, 2013). While MMS data sets do not include directly depth images aligned with the available color images, it is easy, by exploiting the known geometry, to project the lidar

34 point clouds onto each image. This operation produces initial depth images,  
 35 which present three main issues (see Figure 1, where three parts of an input  
 depth image are shown, together with the corresponding image parts).

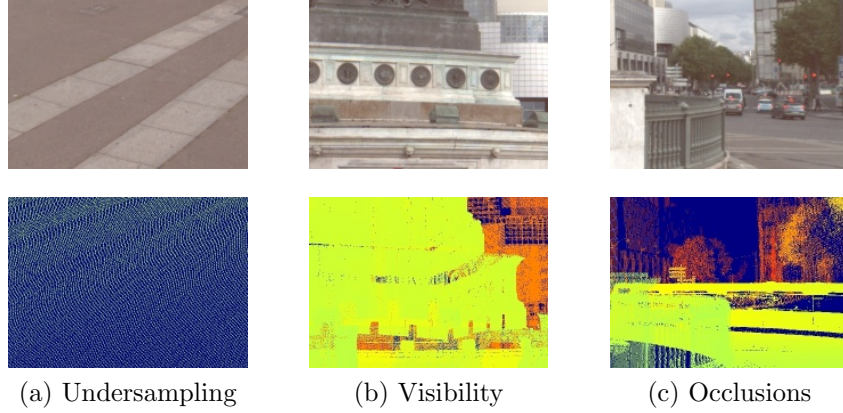


Figure 1: Examples of parts from a resulting input depth image (bottom row), with the corresponding parts from the reference color image (top row), showing the three issues mentioned: undersampling, appearance of hidden points, and presence of occlusions.

36

- 37 1. *Undersampling*: since lidar and image acquisitions are deeply different  
 38 in terms of geometry and characteristics, the resulting depth images  
 39 turn to be irregular. No points are present in the sky and on reflective  
 40 surfaces. Moreover, the point density, which depends on the variable  
 41 distances between the camera image plane and the positions of the lidar  
 42 sensor, is generally significantly smaller than the pixel resolution. We  
 43 can therefore talk about sparse input depth images (see for example  
 44 Figure 1a, showing the low density of lidar points from the ground).
- 45 2. *Visibility* (hidden parts appear): since points that are not visible from  
 46 the image view point (hidden points) can be occasionally “seen” by  
 47 the moving lidar sensor, erroneous values referring to such points can  
 48 appear in the input depth image. This occurs even when a Z-buffer  
 49 approach (Greene et al., 1993) is used, i.e. only the closest depth values  
 50 for each pixel are kept (in case multiple values end up in the same pixel  
 51 location). E.g., Figure 1b shows that depth values from the building  
 52 behind appear as foreground points.
- 53 3. *Occlusions* (visible parts disappear): for the same reason as above, i.e.  
 54 the different acquisition timing and geometry between image and lidar



sensors, surfaces normally visible from the image view point do not get a corresponding depth. This can happen when the lidar sensor suffers occlusions at a given instant or because of the scene dynamics. E.g., in Figure 1c, a moving bus that is not present at the moment of the image shot happens to appear in the depth image.

While there is a variety of methods in the literature that deal with the first issue, i.e. that aim at upscaling an irregular input depth image possibly with the guidance of a corresponding color image, little work has been performed to address the last two issues. In this paper, while inpainting the input depth image, we also intend to tackle the visibility problem. Moreover, we treat at the same time an additional input: a sparse reflectance image derived in the same way as the input depth image (i.e., by naively projecting the lidar point cloud, considering the reflectance information carried out by each point). We will show that the simultaneous use of a reflectance image, which is inpainted jointly with the depth, improves the quality of the produced depth image itself. To jointly inpaint depth and reflectance and concurrently evaluate the visibility of each point (i.e. establish if a single point is reliable or, since non-visible, must be discarded), we formulate an optimization problem with three variables to estimate: depth, reflectance and a visibility attribute per pixel. The inpainting process is also guided by the available color image, by means of a two-fold coupled total variation (TV) regularizer.

The remainder of the paper is organized as follows. In Section 2, we present our approach and mention the related works, in particular on the image-guided depth inpainting problem. In sections 3 and 4 we describe the model used and the primal-dual optimization algorithm that arises, respectively. Finally, in Section 5 we bring experimental evidence that proves the effectiveness of the proposed approach.

## 2. Problem addressed and related work

Figure 2 depicts the scheme of the proposed approach. Given an MMS data set consisting of a lidar point cloud and a set of camera images, we choose among the latter a reference color image ( $w$ ), and we obtain input depth ( $u_S$ ) and reflectance ( $r_s$ ) images by re-projecting the lidar points according to the image geometry. The two lidar-originated images are sparse images with irregular sampling and need to be inpainted. We propose to do that jointly and simultaneously estimate the visibility of the input points,

90 within a variational optimization framework. The output of the algorithm  
 91 are then three: the inpainted depth and reflectance ( $u$  and  $r$ , respectively),  
 92 and a binary image expressing the visibility at each point ( $v$ ).

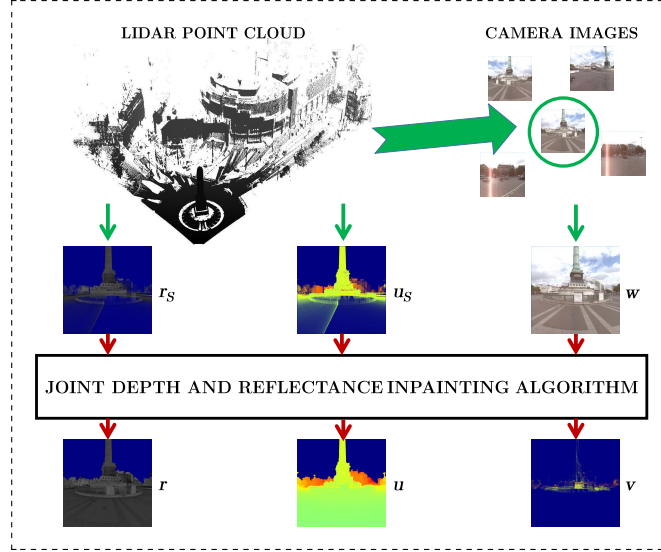


Figure 2: General scheme of the proposed approach. The final outputs of the algorithm are the inpainted reflectance and depth images,  $r$  and  $d$  respectively, and a binary visibility image  $v$ . To represent  $v$ , we show the original depth values that finally get  $v \simeq 0$ .

93 In the literature there is a variety of methods that aim at upscaling or  
 94 inpainting an original sparse depth image. Most of them are presented in the  
 95 context of ToF cameras; thus, a high quality color image is acquired at the  
 96 same time and can be exploited. We refer to this problem as image-guided  
 97 depth inpainting. The typical assumption, when exploiting the available im-  
 98 age, is that image edges are related to depth edges. Following this principle,  
 99 many approaches have been proposed, such as methods using different ver-  
 100 sions of multilateral filtering (Chan et al., 2008; Garcia et al., 2010; Yang  
 101 et al., 2013), methods based on Markov Random Fields (Diebel and Thrun,  
 102 2005), and methods using Non-Local Means (Huhle et al., 2010; Park et al.,  
 103 2011). Another family relates to recent methods that make use of optimiza-  
 104 tion (Ferstl et al., 2013; Harrison and Newman, 2010; Liu and Gong, 2013;  
 105 Schneider et al., 2016; Schwarz et al., 2012). Among these, in (Harrison  
 106 and Newman, 2010), a method to assign image pixel with a range value,  
 107 using both image appearance and sparse laser data, is proposed. The prob-  
 108 lem is posed as an optimization of a cost function encapsulating a spatially

109 varying smoothness cost and measurement compatibility. In the same spirit,  
 110 the authors of (Ferstl et al., 2013) present an optimization-based depth up-  
 111 sampling method, which uses an Anisotropic Total Generalized Variation  
 112 (ATGV) term to regularize the solution while exploiting the color image in-  
 113 formation. Another recent algorithm for the upsampling of sparse depth data  
 114 is presented in (Schneider et al., 2016). The key idea here is to exploit ad-  
 115 ditional object boundary cues (via structured edge detection and semantic  
 116 scene labelling) together with usual intensity cues in a unique optimization  
 117 framework.

118 While presenting good results on images that are not particularly “prob-  
 119 lematic”, in none of the mentioned methods the visibility issue is directly  
 120 tackled, i.e. there is no explicit estimation of input depth measures to pos-  
 121 sibly exclude from the inpainting process. We instead intend to estimate visi-  
 122 bility, to be able to cope with realistic depth images. To this end, we build on  
 123 our previous work on lidar-based depth inpainting (Bevilacqua et al., 2016).  
 124 W.r.t. the latter, the model is significantly modified to include a reflectance  
 125 image as well into a new optimization framework. We will show that depth  
 126 and reflectance mutually benefit of each other in the inpainting process, thus  
 127 leading to better output results for both. In the next section we present the  
 128 novel model.

### 129 3. Model

130 Let  $\Omega \subseteq \mathbb{R}^2$  be the “full” image support, and  $\Omega_S \subseteq \Omega$  the sparse im-  
 131 age support where the input images are defined (i.e., there is at least one  
 132 lidar point ending up there after projection). Given an input depth image  
 133  $u_S : \Omega_S \rightarrow \mathbb{R}$ , an input reflectance image  $r_S : \Omega_S \rightarrow \mathbb{R}$ , and the luminance  
 134 component of their corresponding color image  $w : \Omega \rightarrow \mathbb{R}$  (defined in the  
 135 complete domain), the goal is to fully inpaint the depth and reflectance in-  
 136 put images to obtain  $u : \Omega \rightarrow \mathbb{R}$  and  $r : \Omega \rightarrow \mathbb{R}$ , and concurrently estimate a  
 137 visibility attribute  $v : \Omega_S \rightarrow \mathbb{R}$ . For each input pixel,  $v$  indicates whether it  
 138 is visible from the image view point and should thus be taken into account in  
 139 the inpainting process. Figure 3 reports an example of three possible input  
 140 images - depth ( $u_S$ ), reflectance ( $r_S$ ) and camera images - and their respective  
 141 gradient images.

142 We model our joint inpainting problem as an optimization problem with  
 143 three variables,  $u$ ,  $r$ , and  $v$ , to be estimated. Lower and upper bounds for the  
 144 values of  $u$  and  $r$  are considered in the expression. The visibility attribute

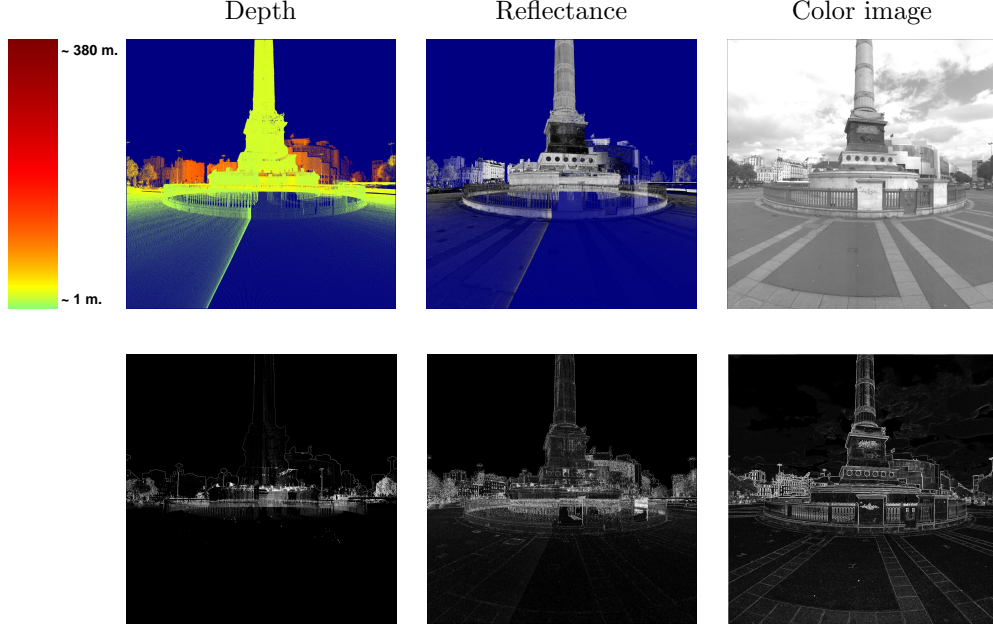


Figure 3: Example of input depth, reflectance and color images (top row), and their respective gradient images (bottom row). Besides the input depth image, the color map used to encode depth values is reported. Gradients of depth and reflectance are computed on the interpolated versions of the input sparse images, initially obtained by nearest neighbor interpolation.

145  $v$  takes values in  $[0, 1]$ , where  $v = 0$  stands for “hidden” and  $v = 1$  means  
 146 that the point is visible from the considered image view point. The model  
 147 considered consists of four terms:

$$\min_{\substack{u \in [u_m, u_M] \\ r \in [r_m, r_M] \\ v \in [0, 1]}} F(u, v|u_S) + G(r, v|r_S) + H(v|u_S, r_S) + R(u, r|w) . \quad (1)$$

148  $F(u, v|u_S)$  and  $G(r, v|r_S)$  are two data-fidelity terms, for depth and re-  
 149 flectance respectively. In both of them the visibility attribute  $v$  intervenes.  
 150  $H(v|u_S, r_S)$  is a term depending exclusively on  $v$ , which represents the total  
 151 cost of classifying input pixels as non-visible. Finally,  $R(u, r|w)$  is a regular-  
 152 ization term that penalizes the total variation of  $u$  and  $r$ , by also taking into  
 153 account the color image  $w$ . In the next sections we will detail all the terms  
 154 composing (1).

### 155 3.1. Visibility-weighted data-fidelity terms

156 The data-fitting terms in (1) are meant to enforce fidelity with the original  
 157 values of depth and reflectance,  $u_S$  and  $r_S$  respectively. Deviations from the  
 158 original values are more penalized if the points are considered “trustful”;  
 159 conversely, for erroneous original measures (e.g., referring to hidden points)  
 160 larger deviations are allowed. Therefore we use the visibility attribute  $v$  to  
 161 weight the data terms. For the reflectance data-fidelity term  $G(r, v|r_S)$  we  
 162 have the following expression:

$$G(r, v|r_S) = \eta_2 \int_{\Omega_S} v|r - r_S| dx_1 dx_2 , \quad (2)$$

163 where  $\eta_2$  is a coefficient weighting the term within the model, and  $dx_1$  and  
 164  $dx_2$  express the differential lengths in the two image directions. Note that in  
 165 (2) an  $\ell_1$ -norm error is used. The  $\ell_1$  norm is considered in substitution of the  
 166 classical  $\ell_2$  measure of the error for its effectiveness in implicitly removing  
 167 impulse noise with strong outliers (Nikolova, 2004) and its better contrast  
 168 preservation (Chan and Esedoglu, 2005). As said, weighting by  $v$  relaxes the  
 169 dependence on the input data for those points classified as hidden.

170 The depth data-fidelity term, weighted by the coefficient  $\eta_1$ , is further  
 171 divided into two terms, as follows:

$$\begin{aligned} F(u, v|u_S) &= \eta_1 \left( \int_{\Omega_S} \max(0, u - u_S) dx_1 dx_2 + \int_{\Omega_S} v(\max(0, u_S - u)) dx_1 dx_2 \right) \\ &= F_1(u|u_S) + F_2(u, v|u_S) . \end{aligned} \quad (3)$$

172 The basic idea behind this separation is to treat differently over- and under-  
 173 estimated depths. Points for which the estimated depth is greater than the  
 174 original value ( $u > u_S$ ) most likely correspond to correct input measures,  
 175 where the over-estimation would be due to the surrounding presence of larger  
 176 erroneous depths. The expression  $\max(0, u - u_S)$  is meant to select this  
 177 kind of points (over-estimated depths). As they are considered reliable, an  
 178 unweighted data-fitting term,  $F_1(u|u_S)$ , is imposed. It is easy to see that  
 179 for these points the visibility attribute  $v$  tends to converge to 1, i.e. they  
 180 are the best candidates for being classified as visible points. Conversely, the  
 181 hidden points to remove are sought among depth values which undergo under-  
 182 estimation ( $u < u_S$ ). These points are taken into account in the second term  
 183  $F_2(u, v|u_S)$ , where the  $\ell_1$  error is weighted by the visibility attribute. Ideally,  
 184 a fraction of them, the most “problematic” ones, will be classified as hidden

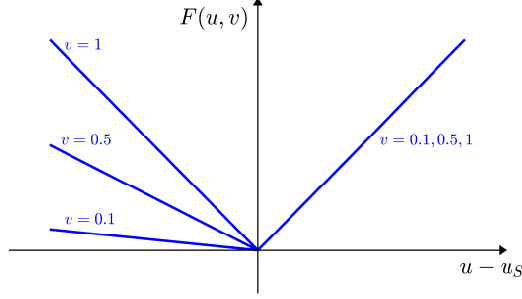


Figure 4: Depth data-fidelity cost  $F(u, v|u_S)$  as a function of  $u - u_S$  for different values of  $v$  ( $\eta_1 = 1$  for simplicity). For over-estimated depths ( $u - u_S > 0$ ) the cost is independent of  $v$ , whereas for  $u - u_S < 0$  we have different lines as  $v$  varies.

185 ( $v = 0$ ) and thus not considered in the data fitting cost. Figure 4 shows  
 186 graphically the depth data-fidelity cost as a function of  $u - u_S$ . Depending  
 187 on the value of the visibility attribute  $v$ , the  $\ell_1$ -type error  $|u - u_S|$  is relaxed  
 188 for negative depth deviations ( $u < u_S$ ).

### 189 3.2. Removal cost

190 The second term of the model (1) is meant to penalize the total number  
 191 of hidden points.

$$H(v|u_S, r_S) = \int_{\Omega_S} \alpha(u_S, r_S)(1 - v) dx_1 dx_2 . \quad (4)$$

192 The cost of a single pixel exclusion is proportional to  $1 - v$ , i.e. we have the  
 193 highest cost for an input pixel when it is totally excluded in the data-fitting  
 194 cost ( $v = 0$ ). We individually weight each removal cost, in order to give  
 195 different importance to each decision visible/hidden. Individual weighting is  
 196 given by a coefficient dependent on the original depth and reflectance values,  
 197  $\alpha(u_S, r_S)$ . We generally choose  $\alpha = k_1 u_S + k_2 r_S$ . The linear dependence of  $\alpha$   
 198 on the depth and the reflectance “balances” the three terms of (1) depending  
 199 on  $v$ , such that  $k_1$  and  $k_2$  appear to be constants. We will discuss their choice  
 200 later in this paper (Section 5.1).

### 201 3.3. Coupled Total Variation

202 Depth upsampling/inpainting methods that exploit corresponding camera  
 203 images often relate image edges to depth edges. This has been shown to  
 204 improve the quality of the reconstructed depth images.

205 To couple two images in a total variation framework, we adopt the *coupled*  
 206 total variation (coupled TV) of (Pierre et al., 2015):

$$\text{TV}_\lambda(a, b) = \int_{\Omega} \sqrt{(\partial_{x_1}a)^2 + (\partial_{x_2}a)^2 + \lambda^2(\partial_{x_1}b)^2 + \lambda^2(\partial_{x_2}b)^2} dx_1 dx_2 . \quad (5)$$

207 where  $\lambda$  is a coupling parameter. When  $\lambda \neq 0$  the minimization of  $\text{TV}_\lambda$   
 208 encourages the gradient “jumps” to occur at the same locations in  $a$  and  $b$ .  
 209 The coupled TV is then a way to align the edges of an image with those of  
 210 a given one.

211 In our problem we have three types of images: a color image  $w$ , a depth  
 212 image  $u$ , and a reflectance image  $r$ . Figure 3 reports in the bottom row  
 213 an example of gradient magnitudes related to three images. The gradients  
 214 of the input depth and reflectance images have been computed after initial  
 215 interpolation of the latter. As we can clearly see from the image, the color  
 216 image gradient particularly matches the reflectance one, while being rather  
 217 dissimilar to the depth gradient. In turn, the reflectance gradient shares  
 218 some patterns, yet less prominently, with the depth one. See, e.g., the area  
 219 at the base of the column, where multiple layers mix and produce a similar  
 220 effect in the two gradient images. We therefore propose to match the three  
 221 gradients two by two: depth with reflectance, and the same reflectance with  
 222 the fixed color image. By using the previous definition of coupled TV (5),  
 223 we express the regularization term as follows:

$$R(u, r|w) = \text{TV}_{\lambda_1}(u, r) + \text{TV}_{\lambda_2}(r, w) . \quad (6)$$

224 After detailing all the terms, our model (1) can therefore be rewritten as  
 225 follows, the four terms being still distinct:

$$\begin{aligned} \min_{\substack{u \in [u_m, u_M] \\ r \in [r_m, r_M] \\ v \in [0, 1]}} & \underbrace{\eta_1 \left( \int_{\Omega_S} \max(0, u - u_S) + \int_{\Omega_S} v(\max(0, u_S - u)) \right)}_{F: \text{Data-fidelity for Depth}} + \underbrace{\eta_2 \int_{\Omega_S} v|r - r_S|}_{G: \text{Data-fidelity for Reflectance}} \\ & + \underbrace{\int_{\Omega_S} \alpha(u_S, r_S)(1 - v)}_{H: \text{Removal cost}} + \underbrace{\text{TV}_{\lambda_1}(u, r) + \text{TV}_{\lambda_2}(r, w)}_{R: \text{TV regularization}} . \end{aligned} \quad (7)$$

226 In the next section we detail a primal-dual approach to solve (7).

## 227 4. Algorithm

228 The optimization problem (7) turns out to be convex, but not smooth,  
 229 due to  $\ell_1$ -type data-fidelity terms,  $F(u, v|u_S)$  and  $G(r, v|r_S)$ , and the total

230 variation regularization term  $R(u, r|w)$ . Recently, in (Chambolle and Pock,  
 231 2011) a primal-dual first-order algorithm has been proposed to solve such  
 232 problems. In Section 4.1 we provide the necessary definitions for the algo-  
 233 rithm, which is subsequently described in Section 4.2.

#### 234 4.1. Discrete setting and definitions

235 Images, considered in Section 3 as continuous functions in  $\mathbb{R}^2$ , are here  
 236 converted into real finite-dimensional vectors. Let  $M$  and  $N$  be the image  
 237 dimensions in this discrete setting, and  $(i, j)$  the indices denoting all possible  
 238 discrete locations in the Cartesian grid of size  $M \times N$  ( $1 \leq i \leq M$ ,  $1 \leq j \leq$   
 239  $N$ ). We then have  $u, u_S, r, r_S, v, w$ , and  $\alpha \in X = \mathbb{R}^{MN}$ , where  $X$  is a finite  
 240 dimensional vector space equipped with a standard scalar product:

$$\langle u, v \rangle_X = \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} u_{i,j} v_{i,j}, \quad u, v \in X. \quad (8)$$

241 The gradient of an image  $u \in X$ ,  $\nabla u$ , is a vector in the vector space  $X^2$   
 242 with two components per pixel:

$$(\nabla u)_{i,j} = ((\nabla_V u)_{i,j}, (\nabla_H u)_{i,j}). \quad (9)$$

243 We compute the gradient components via standard finite differences with  
 244 Neumann boundary conditions, i.e.:

$$\begin{aligned} (\nabla_V u)_{i,j} &= \begin{cases} u_{i+1,j} - u_{i,j} & i < M \\ 0 & i = M \end{cases} \\ (\nabla_H u)_{i,j} &= \begin{cases} u_{i,j+1} - u_{i,j} & j < N \\ 0 & j = N \end{cases} \end{aligned} \quad (10)$$

245 From the definition of gradient, it follows the expression of discrete cou-  
 246 pled total variation, which matches the continuous one (5):

$$\text{TV}_\lambda(a, b) = \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} \sqrt{(\nabla_H a_{i,j})^2 + (\nabla_V a_{i,j})^2 + \lambda^2 (\nabla_H b_{i,j})^2 + \lambda^2 (\nabla_V b_{i,j})^2}. \quad (11)$$

247 As first suggested by (Chan et al., 1999), a total variation optimization  
 248 problem can be recast into a primal-dual form that makes its solution eas-  
 249 ier, by rewriting the gradient norm by means of a vector-valued dual vari-  
 250 able. To this end, in our case we first define a “coupled gradient” operator



251  $\mathcal{K}_{\lambda b} : X \rightarrow Y$  ( $Y = X^4$ ), which, applied to an image  $a \in X$ , expands its  
 252 gradient to include the one of a reference image  $b$  according to a coupling  
 253 parameter  $\lambda$ . I.e., we have the following element-wise definition:

$$(\mathcal{K}_{\lambda b}a)_{i,j} = ((\nabla_H a)_{i,j}, (\nabla_V a)_{i,j}, \lambda(\nabla_H b)_{i,j}, \lambda(\nabla_V b)_{i,j}) . \quad (12)$$

254 The coupled gradient operator  $\mathcal{K}_{\lambda b}$  can be further decomposed as  $\mathcal{K}_{\lambda b} =$   
 255  $\tilde{\mathcal{K}} + \beta_\lambda(b)$ , according to the following element-wise definition:

$$\begin{aligned} (\mathcal{K}_{\lambda b}a)_{i,j} &= (\tilde{\mathcal{K}}a)_{i,j} + (\beta_\lambda(b))_{i,j} \\ &= ((\nabla_H a)_{i,j}, (\nabla_V a)_{i,j}, 0, 0) + (0, 0, \lambda(\nabla_H b)_{i,j}, \lambda(\nabla_V b)_{i,j}) . \end{aligned} \quad (13)$$

256  $\tilde{\mathcal{K}}$  is the usual gradient operator “padded” with two zero components and it  
 257 is linear in  $a$ ;  $\beta_\lambda(b)$  is a bias term, depending on the gradient of the fixed  
 258 variable  $b$ , which determines the last two components of the global coupled  
 259 gradient operator.

260 Thanks to the definitions above, we can express alternatively the coupled  
 261 total variation (11), by introducing the dual variable  $p \in Y$ :

$$\begin{aligned} \text{TV}_\lambda(a, b) &= \max_{p \in Y} \langle \mathcal{K}_{\lambda b}a, p \rangle_Y - \delta_P(p) \\ &= \max_{p \in Y} \langle \tilde{\mathcal{K}}a, p \rangle_Y + \langle \beta_\lambda(b), p \rangle_Y - \delta_P(p) , \end{aligned} \quad (14)$$

where the scalar product in  $Y$  is defined as

$$\begin{aligned} \langle p, q \rangle_Y &= \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} p_{i,j}^1 q_{i,j}^1 + p_{i,j}^2 q_{i,j}^2 + p_{i,j}^3 q_{i,j}^3 + p_{i,j}^4 q_{i,j}^4 , \\ p &= (p^1, p^2, p^3, p^4), \quad q = (q^1, q^2, q^3, q^4) \in Y \end{aligned}$$

262  $\delta_P$  denotes the indicator function of the set  $P$

$$\delta_P(p) = \begin{cases} 0 & \text{if } p \in P \\ +\infty & \text{if } p \notin P \end{cases} , \quad (15)$$

263 and the feasibility set  $P$  for the dual variable  $p$ , is defined as

$$P = \{p \in Y \mid \|p_{i,j}\|_2 \leq 1, \forall i, j\} , \quad (16)$$

264 i.e.  $\|p\|_\infty \leq 1$ .

265 We can now finally express the regularization term of our model  $R(u, r|w)$   
 266 (6) as the maximization over two dual variables. We then have:

$$\begin{aligned} R(u, r|w) &= \max_{p \in Y} \max_{q \in Y} \langle \mathcal{K}_{\lambda_1 r} u, p \rangle_Y + \langle \mathcal{K}_{\lambda_2 w} r, p \rangle_Y - \delta_P(p) - \delta_Q(q) \\ &= \max_{p \in Y} \max_{q \in Y} \langle \tilde{\mathcal{K}} u, p \rangle_Y + \langle \beta_{\lambda_1}(r), p \rangle_Y + \langle \tilde{\mathcal{K}} r, q \rangle_Y + \langle \beta_{\lambda_2}(w), q \rangle_Y - \delta_P(p) - \delta_Q(q) . \end{aligned} \quad (17)$$

267 This will let us formulate a discrete version of our joint inpainting problem  
 268 (7), which falls into the primal-dual optimization framework. As for the  
 269 other terms in (7), rewritten in discrete notation, we have:

$$\begin{aligned} F_1(u|u_S) &= \eta_1 \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} \Phi_{i,j} \max(0, u_{i,j} - u_{S,i,j}) \\ F_2(u, v|u_S) &= \eta_1 \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} \Phi_{i,j} v_{i,j} \max(0, u_{S,i,j} - u_{i,j}) \\ G(r, v|r_S) &= \eta_2 \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} \Phi_{i,j} v_{i,j} |r_{i,j} - r_{S,i,j}| \\ H(v|u_S, r_S) &= \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} \Phi_{i,j} \alpha_{i,j} (1 - v_{i,j}) \end{aligned} \quad (18)$$

270 where  $\Phi$  is a binary mask indicating the initial known pixels, i.e. belonging  
 271 to the sparse image support  $\Omega_S$ .

#### 272 4.2. A primal-dual algorithm

Thanks to the previous definitions, we can express our model (7) in the form of the following saddle-point problem, which is an extension (including two extra variables) of the one presented in (Pierre et al., 2015):

$$\begin{aligned} \min_{u \in X} \min_{r \in X} \min_{v \in X} \max_{p \in Y} \max_{q \in Y} \{ &\langle K_1 u, p \rangle + \langle K_2 r, q \rangle - D_1^*(p) - D_2^*(q) \\ &+ A(u) + B(r) + a(u, v) + b(r, v) + C(v) \} . \end{aligned} \quad (19)$$

273 It is a primal-dual problem with three primal variables ( $u$ ,  $r$ , and  $v$ ) and  
 274 two dual variables ( $p$  and  $q$ ) that evolve independently. Each dual variable  
 275 is particularly linked to the gradient of a primal variable, i.e.  $p$  to  $u$ , and  
 276  $q$  to  $r$ .  $D_1^*$ ,  $D_2^*$ ,  $A$ ,  $B$ , and  $C$  are convex functions;  $a$  and  $b$  are convex  
 277 w.r.t. each of its respective variables. Globally, the functional is not convex

278 w.r.t. the triplet  $(u, r, v)$ . By relating (7) and (19), and using the primal-dual  
 279 expression of the regularization term reported in (17), we have the following  
 280 equivalences:

$$\begin{aligned}
 281 \quad & \bullet K_1 u = \tilde{K}u; & \bullet K_2 r = \tilde{K}r; \\
 282 \quad & \bullet D_1^*(p) = -\langle \beta_{\lambda_1}(r), p \rangle_Y + \delta_P(p); & \bullet D_2^*(q) = -\langle \beta_{\lambda_2}(w), q \rangle_Y + \delta_Q(q); \\
 283 \quad & \bullet A(u) = F_1(u|u_S) + \delta_{[u_m, u_M]}(u); & \bullet B(r) = \delta_{[r_m, r_M]}(r); \\
 284 \quad & \bullet a(u, v) = F_2(u, v|u_S); & \bullet b(r, v) = G(r, v|r_S); \\
 285 \quad & \bullet C(v) = H(v|u_S, r_S) + \delta_{[0,1]}(v).
 \end{aligned}$$

286 An algorithm to solve (19) can be derived within the primal-dual opti-  
 287 mization framework of (Chambolle and Pock, 2011). It consists in a unique  
 288 loop, where all variables are alternatively updated via proximal operators  
 289 (see Algorithm 1). The algorithm takes as inputs the initial estimates of  
 290 the complete depth and reflectance images ( $u_0$  and  $r_0$ , respectively), and the  
 291 reference intensity image  $w$ . It also requires three parameters inherent to the  
 292 algorithm:  $\sigma$  and  $\tau$ , which are related to each other by the relation  $16\tau\sigma \leq 1$   
 293 (Chambolle and Pock, 2011), and  $\rho$ , which is a parameter regulating the  
 294 update speed of  $v$ .

295 Algorithm 1 involves the computation of the adjoints to the linear op-  
 296 erators  $K_1$  and  $K_2$  (the “zero-padded” gradient operators). It is known  
 297 that the adjoint of the gradient operator is the negative divergence operator  
 298 ( $\nabla^* = -\text{div}$ ). In our case, the adjoint to the operator  $K_1 : X \rightarrow Y$  is a  
 299 linear operator  $K_1^* : Y \rightarrow X$  consisting in the negative divergence computed  
 300 only on the two first components of a four-component dual variable  $p \in Y$ ,  
 301 and by taking finite differences in the opposite direction than the gradient  
 302 operator (10). These components are in fact the ones related to the primal  
 303 variable to which the coupled gradient operator has been applied. We then  
 304 have the following element-wise definition for  $K_1^*p$  (the same definition stands  
 305 for  $K_2^*q$ ):

$$(K_1^*p)_{i,j} = - \begin{cases} p_{i,j}^1 - p_{i-1,j}^1 & \text{if } 1 < i < M \\ p_{i,j}^1 & \text{if } i = 1 \\ -p_{i-1,j}^1 & \text{if } i = M \end{cases} - \begin{cases} p_{i,j}^2 - p_{i,j-1}^2 & \text{if } 1 < j < N \\ p_{i,j}^2 & \text{if } j = 1 \\ -p_{i,j-1}^2 & \text{if } j = N \end{cases}. \quad (20)$$

---

**Algorithm 1** Primal-dual based algorithm for depth and reflectance joint inpainting.

---

1: **Inputs:**

$$u_0, r_0, w, \sigma, \rho, \tau$$

2: **Initialize:**

$$u^0, \bar{u}^0 \leftarrow u_0, r^0, \bar{r}^0 \leftarrow r_0, v_{i,j}^0 \leftarrow 0.5, \\ p^0 \leftarrow (\nabla u_0, \lambda_1 \nabla r_0), q^0 \leftarrow (\nabla r_0, \lambda_2 \nabla w)$$

3: **for**  $n = 0, 1, \dots$  **do**

$$4: p^{n+1} \leftarrow \text{prox}_{\sigma D_1^*}(p^n + \sigma K_1 \bar{u}^n)$$

$$5: q^{n+1} \leftarrow \text{prox}_{\sigma D_2^*}(q^n + \sigma K_2 \bar{r}^n)$$

$$6: v^{n+1} \leftarrow \text{prox}_{\rho a(\bar{u}^n, \cdot) + \rho b(\bar{r}^n, \cdot) + \rho C}(v^n)$$

$$7: u^{n+1} \leftarrow \text{prox}_{\tau A + \tau a(\cdot, v^{n+1})}(u^n - \tau K_1^* p^{n+1})$$

$$8: r^{n+1} \leftarrow \text{prox}_{\tau B + \tau b(\cdot, v^{n+1})}(r^n - \tau K_2^* q^{n+1})$$

$$9: \bar{u}^{n+1} \leftarrow 2u^{n+1} - u^n$$

$$10: \bar{r}^{n+1} \leftarrow 2r^{n+1} - r^n$$

11: **end for**

---

306 Closed-form expressions for the update rules in Algorithm 1 can be easily  
 307 computed by applying the definition of proximal operator (see Appendix  
 308 A). The resulting expressions are reported here below, where  $\mathcal{P}$  denotes  
 309 the projection operation over a given real interval, i.e. values are clipped if  
 310 exceeding the interval limits. Details about the derivation of these expression  
 311 can be found in Appendix A.

$$\text{prox}_{\sigma D_1^*}(\tilde{p}) = \frac{\tilde{p} + \sigma \beta_{\lambda_1}(r)}{\max(1, \|\tilde{p} + \sigma \beta_{\lambda_1}(r)\|_2)} \quad (21)$$

$$\text{prox}_{\sigma D_2^*}(\tilde{q}) = \frac{\tilde{q} + \sigma \beta_{\lambda_2}(w)}{\max(1, \|\tilde{q} + \sigma \beta_{\lambda_2}(w)\|_2)} \quad (22)$$

$$\begin{aligned} \text{prox}_{\rho a(\bar{u}, \cdot) + \rho b(\bar{r}, \cdot) + \rho C}(\tilde{v}) = \\ \begin{cases} \mathcal{P}_{[0,1]}(\tilde{v}) & \text{if } \Phi_{i,j} = 0 \\ \mathcal{P}_{[0,1]}(\tilde{v} + \rho\alpha - \rho\eta_2|\bar{r} - r_S|) & \text{if } \Phi_{i,j} = 1, \bar{u}_{i,j} \geq u_{S\ i,j} \\ \mathcal{P}_{[0,1]}(\tilde{v} + \rho\alpha - \rho\eta_1(u_S - \bar{u}) - \rho\eta_2|\bar{r} - r_S|) & \text{if } \Phi_{i,j} = 1, \bar{u}_{i,j} < u_{S\ i,j} \end{cases} \quad (23) \end{aligned}$$

$$\text{prox}_{\tau A + \tau a(\cdot, v)}(\tilde{u}) = \begin{cases} \mathcal{P}_{[u_m, u_M]}(\tilde{u}) & \text{if } \Phi_{i,j} = 0 \\ \mathcal{P}_{[u_m, u_M]}(\tilde{u} - \tau\eta_1) & \text{if } \Phi_{i,j} = 1, \tilde{u}_{i,j} > u_{S\ i,j} + \tau\eta_1 \\ \mathcal{P}_{[u_m, u_M]}(\tilde{u} + v\tau\eta_1) & \text{if } \Phi_{i,j} = 1, \tilde{u}_{i,j} < u_{S\ i,j} - v\tau\eta_1 \\ \mathcal{P}_{[u_m, u_M]}(u_S) & \text{otherwise} \end{cases} \quad (24)$$

$$\text{prox}_{\tau B + \tau b(\cdot, v)}(\tilde{r}) = \begin{cases} \mathcal{P}_{[r_m, r_M]}(\tilde{r}) & \text{if } \Phi_{i,j} = 0 \\ \mathcal{P}_{[r_m, r_M]}(\tilde{r} - v\tau\eta_2) & \text{if } \Phi_{i,j} = 1, \tilde{r}_{i,j} > r_{S\ i,j} + v\tau\eta_2 \\ \mathcal{P}_{[r_m, r_M]}(\tilde{r} + v\tau\eta_2) & \text{if } \Phi_{i,j} = 1, \tilde{r}_{i,j} < r_{S\ i,j} - v\tau\eta_2 \\ \mathcal{P}_{[r_m, r_M]}(r_S) & \text{otherwise} \end{cases} \quad (25)$$

312 The operations indicated in the proximal operators are pixel-wise, al-  
 313 though the pixel coordinates have not been made explicit for clearer read-  
 314 ing.

## 315 5. Experimental results

316 The algorithm presented in Section 4 is evaluated with a new data set  
 317 acquired in an urban scenario by a Mobile Mapping System (MMS), com-  
 318 posed of lidar measures and camera-originated images. With this data set,  
 319 we provide a qualitative evaluation of our algorithm in comparison with other  
 320 methods, by showing the reconstructed depth and reflectance images, and we  
 321 assess the quality of the visibility estimation task, which is a crucial charac-  
 322 teristic of our algorithm. Moreover, we also provide a quantitative analysis  
 323 by means of a benchmark data set publicly available. Before showing results  
 324 and comparisons, in Section 5.1 we motivate some critical choices in terms  
 325 of model and algorithmic parameters.

### 326 5.1. Parameters of the algorithm and model choices

327 Our finally resulting joint inpainting model (7) consists of four terms:  
 328 two data-fidelity terms,  $F(u, v|u_S)$  and  $G(r, v|r_S)$ , a “removal” cost depend-  
 329 ing solely on the variable  $v$ ,  $H(v|u_S, r_S)$ , and the two-fold regularization  
 330 term  $R(u, r|w)$ . As discussed in Section 3.1, for the data-fidelity terms we  
 331 opt for an  $\ell_1$  measure of the error, in order to promote more contrasted so-  
 332 lutions (Chan and Esedoglu, 2005). The visibility attribute  $v$  weights the  
 333 data matching cost of each single pixel (data matching is more and more

334 relaxed, as  $v$  tends to zero, i.e. when that particular point is considered to  
 335 be excluded). However, over-estimated depths ( $u > u_S$ ) are not weighted by  
 336  $v$  but are fully penalized. These values relate to pixels where either there  
 337 is noise on a visible point that is slightly corrected ( $u - u_S$  is small), or the  
 338 value  $u_S$  represents an outlier (e.g. it is due to a mobile object). At present,  
 339 we do not have a way to handle the latter case.

340 In  $H(v|u_S, r_S)$  (4), each point removal cost is the product between  $(1 - v)$   
 341 (the level of “invisibility” of the point) and a coefficient  $\alpha$  depending on the  
 342 local input depth and reflectance:  $\alpha = k_1 u_S + k_2 r_S$ . This choice has been  
 343 made in order to balance all terms in (7) where  $v$  appears. Let us now  
 344 observe the “complete” update rule for  $v$  (last case of (23), i.e. for points  
 345 with under-estimated depth). According to it, we have that at each iteration  
 346  $v$  is incremented/decremented by a quantity  $\Delta v = \rho(\alpha - \eta_1 \Delta u - \eta_2 \Delta r)$ . Let  
 347 us suppose that the fluctuations on depth are significantly larger than the  
 348 fluctuations on reflectance (the appearance of a hidden point can cause a big  
 349 “jump” in depth, while the reflectance values might still be similar. For the  
 350 sake of simplicity we can then adjust the value of  $\alpha$  only on the basis of the  
 351 depth input value. The proposed simplified expression for  $\alpha$  is then:

$$\alpha = k u_S. \quad (26)$$

352 With the assumptions made we therefore have  $\Delta v \propto (k u_S - \eta_1 \Delta u)$ . The  
 353 attribute  $v$  for a certain pixel increases (it gets a higher confidence as a  
 354 visible point) if  $\frac{\Delta u}{u_S} < \frac{k}{\eta_1}$ , i.e. if the relative depth deviation is below a certain  
 355 threshold.  $k$  is an adimensional parameter that contributes determining this  
 356 threshold. Conversely,  $v$  decreases for relative depth deviations exceeding  
 357 the threshold. As for the update of  $v$  for points with over-estimated depths  
 358 (second case of (23)), if we hypothesize that  $\alpha$ , adjusted on depth, is large  
 359 enough w.r.t. the reflectance deviation, we have that  $v$  progressively tends  
 360 to one (unless large absolute reflectance deviations occur).

361 As for the regularization term  $R(u, r|w)$ , we proposed in Section 3.3 to  
 362 combine two distinct coupled total variation terms:  $\text{TV}_{\lambda_1}(u, r)$  (depth is  
 363 individually coupled with reflectance) and  $\text{TV}_{\lambda_2}(r, w)$  (reflectance is individ-  
 364 ually coupled with the color image). By having two separate coupled TV  
 365 terms, each one encoded by a dual variable that evolves independently from  
 366 the other one, the reflectance gradient is constantly brought back to the ref-  
 367 erence gradient of the color image. At the same time the “correct” gradient  
 368 information is transferred to the depth via the second term. Figure 5 shows

369 an example of results obtained with the algorithm for the same test case as  
 370 Figure 3.

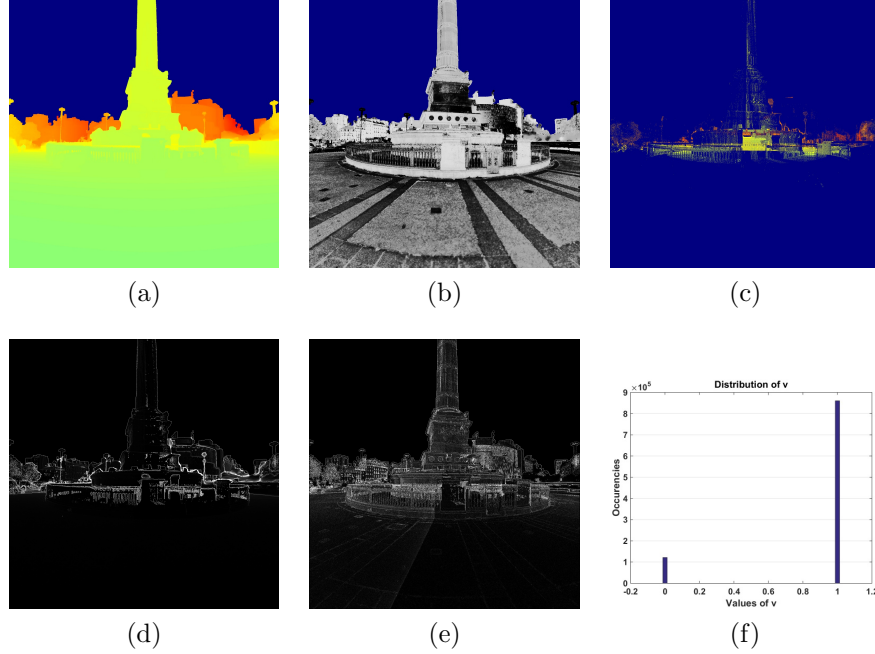


Figure 5: Output of the proposed algorithm for the image *Column1*: (a) Inpainted depth, (b) Inpainted reflectance, (c) Removed points ( $v = 0$ ), (d) Final depth gradient, (e) Final reflectance gradient, (f) Final histogram of  $v$ .

371 For the example test of Figure 5, as well as for all the results reported  
 372 hereinafter, the following parameters, found with multiple tests, have been  
 373 used to characterize the model (7):  $\eta_1 = 1.7$ ,  $\eta_2 = 50$ ,  $k = 0.05$  (the co-  
 374 efficient determining  $\alpha$  according to (26)),  $\lambda_1 = 0.5$ ,  $\lambda_2 = 1$ . These values  
 375 have been found empirically by letting them vary one by one and observing  
 376 the obtained visual results. The two data terms  $F(u, v|u_S)$  and  $G(r, v|r_S)$   
 377 are attributed different weights. The larger coefficient assigned to the re-  
 378 flectance data term ( $\eta_2 > \eta_1$ ) means that a greater data fidelity is imposed  
 379 on reflectance. Depth values have instead a greater “freedom” in deviating  
 380 from their original values. The two coupling parameters  $\lambda_1$  and  $\lambda_2$  being  
 381 in the same order of magnitude, it shows that the two coupling terms have  
 382 a similar importance. As for the parameters, inherent to the primal-dual  
 383 optimization scheme (Algorithm 1), the following values have been set after  
 384 testing:  $\rho = 10$ ,  $\tau = 0.004$ ,  $\sigma = 14$ .

385 If we observe the input sparse depth image of Figure 3, we see that  
 386 the major problems come from the fact that depth values referring to the  
 387 building behind the column appear mixed with foreground depths. With  
 388 our algorithm we are able to resolve these conflicts, as we can see in the  
 389 inpainted depth image (Figure 5a). Part of the input pixels have in fact  
 390 been removed, i.e. classified as non-visible ( $v = 0$ ). Figure 5c reports the  
 391 locations of such points in the original depth image. From the histogram of  
 392 the values of  $v$  (Figure 5f) it is evident that the algorithm produces a bi-  
 393 partition of the points according to their visibility attribute. Figure 5 shows  
 394 also the inpainted reflectance and the final depth and reflectance gradients.  
 395 By comparing the latter to the original gradients (Figure 3), we can observe  
 396 that they end up incorporating elements of the color image gradient, while  
 397 removing erroneous edges. In the next section we will present more results  
 398 obtained with our algorithm, also in comparison to other inpainting methods.

## 399 5.2. Results with urban data

400 We consider a data set acquired by a MMS system (Paparoditis et al.,  
 401 2012) at *Place de la Bastille*, Paris, consisting of one lidar point cloud in the  
 402 order of one billion of points and hundreds of optical images simultaneously  
 403 acquired by 5 cameras mounted on the vehicle. Given a reference optical  
 404 image, we project onto it the available lidar points to form the initial depth  
 405 and reflectance incomplete images. Note that not all the points are effectively  
 406 visible from the image view point. The incomplete depth and reflectance  
 407 images, along with the reference color image chosen, represent the input of  
 408 the algorithm ( $u_S$ ,  $r_S$ , and  $w$  respectively).

409 Figures 6–9 present results for four images (cropped w.r.t. the full size)  
 410 of the data set: *Column1*, *Column2*, *Buildings1*, *Buildings2*. For each ref-  
 411 erence image, the input sparse depth and reflectance images, obtained via  
 412 projection, are shown, as well as the inpainted depth and reflectance im-  
 413 ages, obtained with four different methods. For the output depth images of  
 414 Figure 8 and 9 we added some shading by modulating the color intensity  
 415 of each pixel based on the zenith angle of the normal vector, to emphasize  
 416 high-frequency changes. Moreover, for the inpainted depths, an alternative  
 417 view of the resulting 3-D point cloud is proposed, where the coordinates of  
 418 the points are retrieved thanks to the computed depths and color texture is  
 419 applied to enrich the points. A color box is overlaid to the first of these 3-D  
 420 views to highlight areas where the comparison between the different methods  
 421 is particularly significant.



Our algorithm, presented in Section 4, gives as output the two inpainted images  $u$  and  $r$ . As for the produced depth image, our algorithm is visually compared with nearest neighbor (NN) interpolation, the anisotropic total generalized variation (*ATGV*) method of (Ferstl et al., 2013), and our previous depth inpainting method (Bevilacqua et al., 2016), which does not rely on reflectance information. We refer to the latter as Depth Inpainting with Visibility Estimation (*DIVE*). The optimization problem of DIVE is the following:

$$\min_{\substack{u \in [u_m, u_M] \\ v \in [0, 1]}} \eta \int_{\Omega_S} (\max(0, u - y))^2 dx_1 dx_2 + \eta \int_{\Omega_S} v (\max(0, y - u))^2 dx_1 dx_2 \\ + \int_{\Omega_S} (ku_S)^2 (1 - v) dx_1 dx_2 + \text{TV}_\lambda(u, w) . \quad (27)$$

422 The DIVE problem can be related to our proposed model (7), if we consider  
 423 in the latter  $\eta_1 = \eta$ ,  $\eta_2 = 0$ ,  $\lambda_1 = \lambda$ , and we suppress the coupled TV term  
 424 related to the reflectance (depth is instead coupled directly with the color  
 425 image). Moreover, in (27) we have a  $\ell_2$ -norm data fidelity term; as a conse-  
 426 quence of that, the coefficient of the removal cost term follows a quadratic  
 427 law (we have  $\alpha = (ku_S)^2$ , instead of  $\alpha = ku_S$ , as in (7)).

428 As for the produced reflectance image, our algorithm is compared with  
 429 nearest neighbor (NN) interpolation, the ATGV method of (Ferstl et al.,  
 430 2013) applied to reflectance, and a reduced version of our model (7) limited to  
 431 reflectance. We refer to this method as Reflectance Inpainting with Visibility  
 432 Estimation (*RIVE*). The RIVE method is derived from the solution of the  
 433 following optimization problem:

$$\min_{\substack{r \in [r_m, r_M] \\ v \in [0, 1]}} \eta \int_{\Omega_S} v |r - r_S| dx_1 dx_2 + \int_{\Omega_S} (kr_S)(1 - v) dx_1 dx_2 + \text{TV}_\lambda(r, w) . \quad (28)$$

434 Also in this case we can derive the considered problem (RIVE) as a simplified  
 435 version of our proposed model (7), where  $\eta_1 = 0$ ,  $\eta_2 = \eta$ ,  $\lambda_2 = \lambda$ , and the  
 436 coupled TV term related to depth is suppressed. Moreover, the coefficient of  
 437 the removal cost, while still following a linear law, here depends on the input  
 438 reflectance  $r_S$ .

439 The four examples reported show the better performance of our algorithm  
 440 in generating complete depth and reflectance images from real lidar measures.  
 441 Results with the image *Column1*, reported in Figure 6, particularly prove the

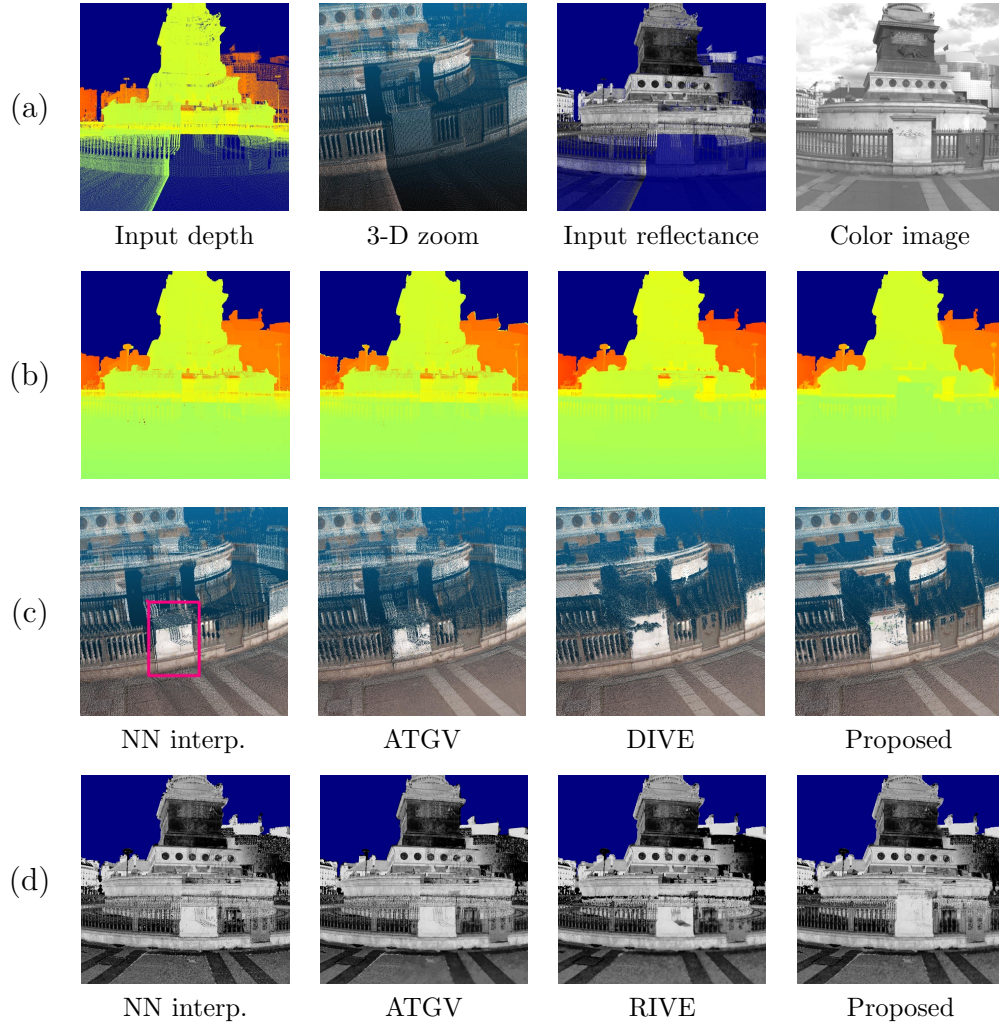


Figure 6: Visual results for the image *Column1*. Row (a) shows the related input images: depth (with a 3-D zoom), reflectance, and reference color image. Rows (b) and (c) report the results obtained in terms of inpainted depth images (with related 3-D zoomed-in view) with the algorithms indicated below. Row (d) shows the inpainted reflectance images obtained with different methods, our proposed method always reported as last.

effectiveness of our algorithm in detecting and removing hidden points ap-  
 pearing in the front, thus producing inpainted images correct from the image  
 view point. These points, in yellow/orange according to the color code used  
 for depth, appear mixed to visible points belonging to the column and the  
 fence. By looking at the depth images generated (row (b)), our algorithm

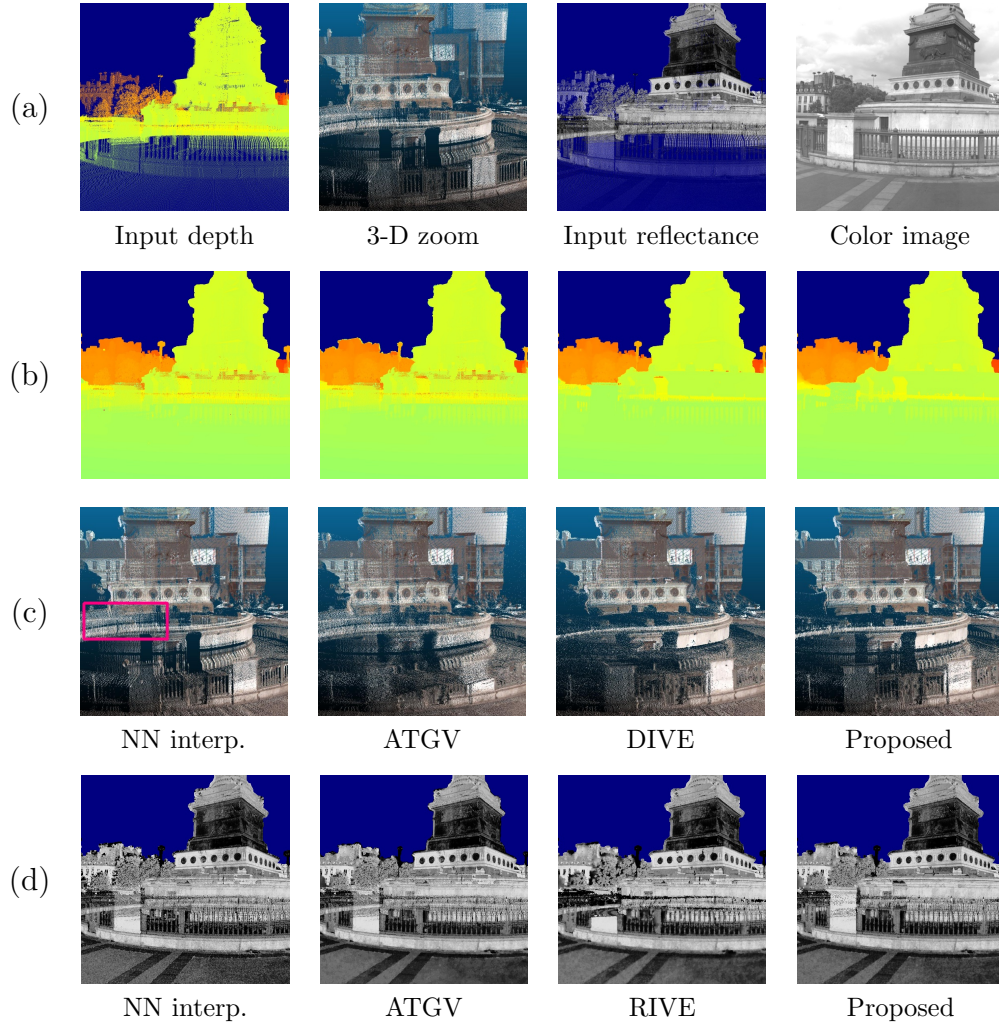


Figure 7: Visual results for the image *Column2*. Row (a) shows the related input images: depth (with a 3-D zoom), reflectance, and reference color image. Rows (b) and (c) report the results obtained in terms of inpainted depth images (with related 3-D zoomed-in view) with the algorithms indicated below. Row (d) shows the inpainted reflectance images obtained with different methods, our proposed method always reported as last.

447 is the only one which is able to remove the misleading points and correctly  
 448 reconstruct the foreground depth plane. This is even more visible by ob-  
 449 serving the main marble pole highlighted in the 3-D views (row (c)). While  
 450 other methods are not able to reconstruct the pole, since “distracted” by the  
 451 interfering background depths, the reconstruction is better performed in our

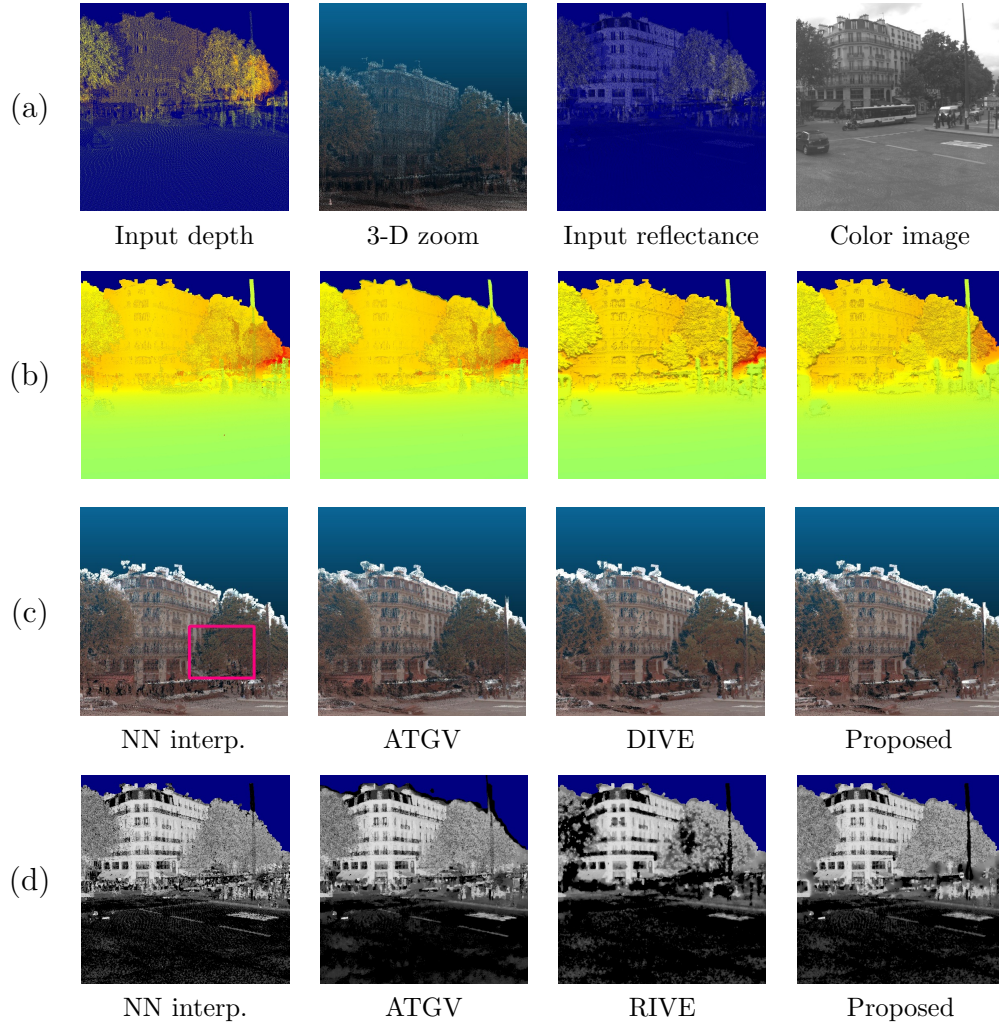


Figure 8: Visual results for the image *Buildings1*. Row (a) shows the related input images: depth (with a 3-D zoom), reflectance, and reference color image. Rows (b) and (c) report the results obtained in terms of inpainted depth images (with related 3-D zoomed-in view) with the algorithms indicated below. Row (d) shows the inpainted reflectance images obtained with different methods, our proposed method always reported as last.

case. Results on the reflectance image confirm the trend. By observing again  
the main marble pole, we clearly see that the reflectance is better inpainted.  
This is possible thanks to the joint use of depth information, which helps de-  
tecting hidden points by leveraging depth over- and under-estimations, and  
the coupling with the color image gradient, which helps correctly restoring



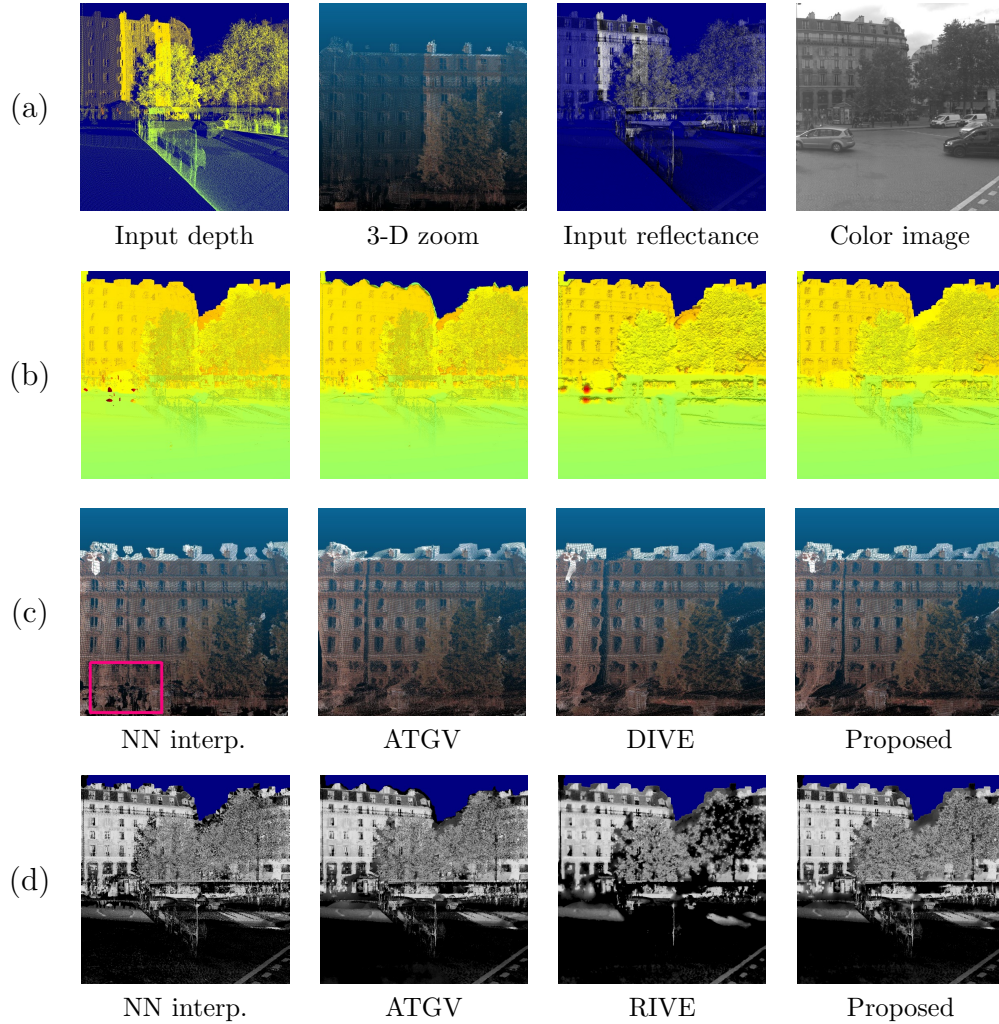


Figure 9: Visual results for the image *Buildings2*. Row (a) shows the related input images: depth (with a 3-D zoom), reflectance, and reference color image. Rows (b) and (c) report the results obtained in terms of inpainted depth images (with related 3-D zoomed-in view) with the algorithms indicated below. Row (d) shows the inpainted reflectance images obtained with different methods, our proposed method always reported as last.

457 the edges. Similar considerations can be made for the image *Column2* (visual  
 458 results are reported in Figure 7). Here the box overlaid on the 3-D views indi-  
 459 cates an area where points, non-visible from the reference image view point,  
 460 should be removed. The removal of these points, as well as the inpainting of  
 461 depth and reflectance, is performed more efficiently by our method.

462 Figures 8 and 9 show results w.r.t. two other images taken peripherally to  
 463 the scene. For the image *Buildings1*, we can observe that with our algorithm  
 464 the inpainted depth and reflectance images look more satisfactory, the pole on  
 465 the left being completely unveiled as a foreground element. The box overlaid  
 466 on the 3-D views highlights a part of the scene where the depth values of  
 467 two trees interfere. Our proposed algorithm (as well as the DIVE method  
 468 (Bevilacqua et al., 2016)) makes a correct distinction between the two depth  
 469 layers. Figure 9, reporting results related to the image *Buildings2*, presents  
 470 the problem of wrong lidar measures appearing in the front. Our method  
 471 turns out to be the most effective in clearing out these points, as also shown  
 472 in the area highlighted by the box.

### 473 5.3. Performance on visibility estimation

474 While in the previous section we evaluated the performance of the algo-  
 475 rithm in terms of produced inpainted images  $u$  and  $r$ , we now want to assess  
 476 the quality of the third output of the algorithm, i.e.  $v$ , the visibility attribute.

477 As visibility is estimated while performing the depth and reflectance es-  
 478 timation, we can say that our algorithm fuses two problems: hidden point  
 479 removal (HPR) and inpainting. Typically HPR is, instead, possibly per-  
 480 formed as a preliminary operation. For HPR “stand-alone” the state of the  
 481 art is represented by variations of (Katz et al., 2007) that relate the visible  
 482 point set to the convex hull of a viewpoint-dependent transformation of it,  
 483 discarding points based on a concavity threshold as seen from the view point.  
 484 While this approach is effective, there is in general no globally satisfactory  
 485 concavity threshold that would both correctly detect hidden surfaces and  
 486 keep background points close to foreground silhouettes. To compare the two  
 487 strategies for estimating visibility (the dedicated operation of (Katz et al.,  
 488 2007) and our “soft” estimation), we show an example in Figure 10, related  
 489 to the image *Column1*. In our case, we consider hidden points those depth  
 490 values that are assigned  $v = 0$  at the end of the algorithm. As for (Katz  
 491 et al., 2007), a concavity parameter equal to 4 has been chosen after tuning.

492 The images obtained show that the “quality” of the visibility estimation  
 493 process is comparable, if not higher with our method. If we observe closely  
 494 the zoomed-in areas in Figure 10, in fact, we can see that the HPR method  
 495 wrongly selects points around the silhouettes (see first patch), while some-  
 496 times missing the detection of actual hidden points (see last two patches).

497 As a further test, we also compare our method (which jointly performs  
 498 visibility estimation and inpainting), with a two-step approach, where visibil-

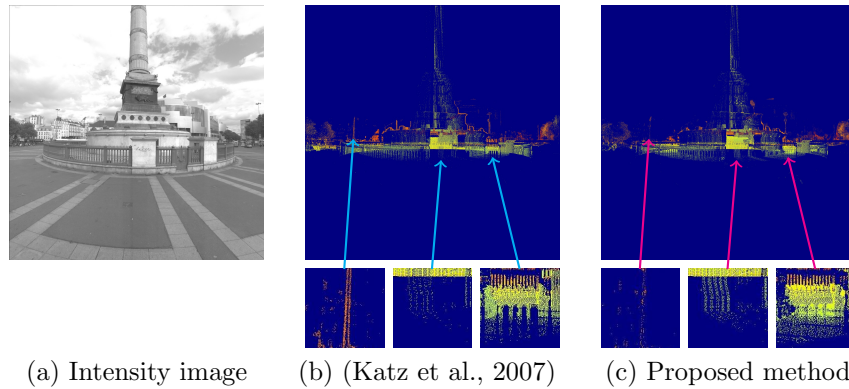


Figure 10: Detected hidden points in the case of the image *Column1*, by the state-of-the-art method of (Katz et al., 2007) and our method. The three patches below each image represent zoomed-in areas of the images themselves at same locations.

ity estimation (hidden point removal) is performed as a preliminary operation by the algorithm of (Katz et al., 2007). Depth is subsequently inpainted with the ATGV-based algorithm of (Ferstl et al., 2013). Figure 11 reports results for such comparison with two images, the two-step approach being denoted as “HPR + ATGV”.

In the two cases of Figure 11, we can observe a better outcome with our algorithm. For the image *Column1*, the preliminary point removal operation is not able to remove all the ambiguities in the central part of the image, where the depth values of the fence and the column are confused. For the image *Buildings2*, the HPR method of (Katz et al., 2007) exceeds in removing several points along the upper board of the image, causing blurred edges in the final reconstructed depth image. Besides the benefits observable in the qualitative assessment, the joint approach of our method has the advantage of not requiring an explicit parameter to be globally set (the concavity threshold in the case of (Katz et al., 2007)) to perform HPR. This is instead done in a “soft” way that adapts to the input image.

#### 5.4. Quantitative evaluation with a benchmark data set

In this section we perform a quantitative analysis of our algorithm by using the publicly available KITTI Vision Benchmark Suite (Geiger et al., 2013). The KITTI suite includes data acquired by a MMS similar to the one considered for our personal data set in Section 5.2, and allows to evaluate algorithms on several applications. The vehicle used is a standard car equipped

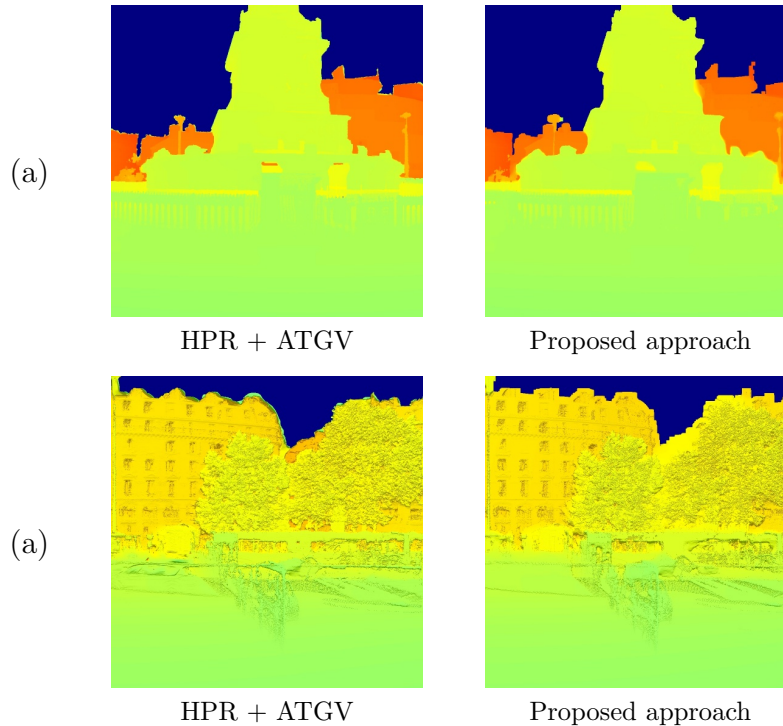


Figure 11: Comparison between our joint approach and a two-step approach, where visibility estimation and inpainting are performed separately, on the images *Column1* (a) and *Buldings2* (b).

with two high-resolution color and grayscale video cameras (enabling stereo vision), acquiring images with a pixel resolution equal to  $1242 \times 375$ . Accurate 3-D measures are provided by a Velodyne lidar scanner and a GPS localization system. Thus, the lidar measures are generally used as ground truth for algorithm evaluations. In (Menze and Geiger, 2015) a novel data set is presented for stereo benchmarking, which considers also moving objects. By making a special processing on the latter and manually removing erroneous points due to occlusions, ground truth disparity maps are obtained. These maps appear “cleaner” and denser than the input depth images that can be obtained with the raw lidar data, and can therefore be used to evaluate algorithm estimating disparity. To exploit this possibility, as described in (Schneider et al., 2016, Sec. 4.3), we use the ground truth maps of this stereo benchmark data set to have a quantitative evaluation of our depth+reflectance inpainting algorithm. As done by the authors of



(Schneider et al., 2016), we identify 82 frames (provided ground truth disparity maps) for which we can find correspondences in the raw data set, i.e. a corresponding color image and related lidar point cloud. We then use the raw data lidar to compute an input depth (e.g., Figure 12a) and use the provided ground truth map to compute a Mean Absolute Error (MAE). The ground truth maps, although denser than the input maps, are sparse, i.e. they are not defined for all pixels (only about 19% of the pixels have values). Thus, the MAE is computed only for those pixels which are defined in the respective ground truth map.

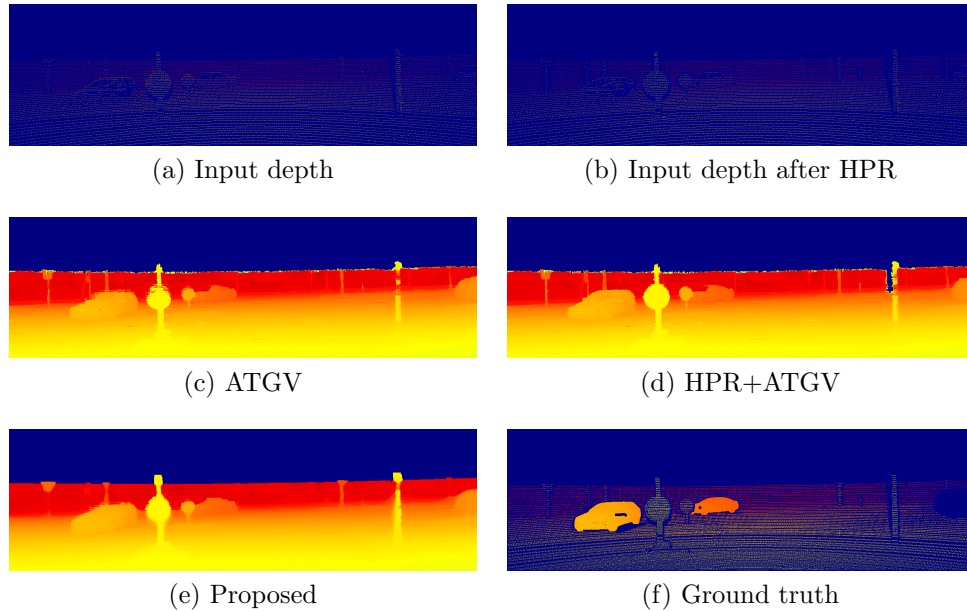


Figure 12: Case example from the 2015 KITTI stereo benchmark data set. For each input depth map (a), we have a ground truth disparity map available yet sparse (f), w.r.t. which it is possible to compute an error by only considering the pixels where it is defined. By applying a hidden point removal (HPR) algorithm to the input depth data it is possible to create a new input map where background hidden pixels have been removed (b). Results for different depth inpainting strategies are reported (c, d, e).

We computed the MAE for all 82 frames of the found correspondences, for our method and the ATGV-based algorithm of (Ferstl et al., 2013). As done in Section 5.3, we also compare with a two-step approach, where AGTV-based inpainting is preceded by a hidden point removal (HPR) operation, performed with the algorithm of (Katz et al., 2007). The resulting average MEAs, which are measured as the average pixel displacement between two

disparity maps, are reported in Table 1.

	ATGV	HPR+ATGV	Proposed
<b>Average MAE (<math>px.</math>)</b>	2.13	2.07	1.99

Table 1: Average Mean Absolute Error (MAE), i.e. average pixel displacement between ground truth and reconstructed disparity maps, obtained by averaging the results of 82 frames of the 2015 KITTI stereo benchmark data set.

When creating the ground truth maps, the authors of the KITTI benchmark data set have removed objects presenting particular issues in terms of visibility. Other objects are instead manually handled (they are removed from the scene and re-inserted after fitting a CAD model). Thus, the ground truth maps basically consist of the latter and fixed parts of the scene (e.g. streets and walls) that do not yield any ambiguity. Due to this relative “simplicity” of the data set, the performance in terms of average MAE are rather similar among the three methods (ATGV, HPR+ATGV, and proposed method), with our method obtaining a slightly lower error. Nevertheless, we can observe that the ATGV method of (Ferstl et al., 2013) produces more artifacts (see, for example, the reconstructed pole on the left in Figure 12c, in comparison to Figure 12e). Most of these artifacts can be removed by performing a preliminary HPR step (see, in Figure 12b, an example of input depth map cleaned out of ambiguous pixel). The combination of a HPR step and the ATGV-based depth upsampling algorithm of (Ferstl et al., 2013) yields inpainted depth maps with a visual quality comparable to the one of our approach. However, as stated in Section 5.3, with our approach we keep the advantage of having an all-in-one procedure performing jointly inpainting and “soft” visibility estimation (without the need of setting a per-image global threshold as requested by the algorithm of (Katz et al., 2007)). We also expect for our method a greater improvement of the MAE metric and the visual outcome on more complex scenes.

## 6. Conclusion

In this paper we presented a novel strategy to jointly inpaint depth and reflectance images with the guidance of a co-registered color image, and by simultaneously estimating a visibility attribute for each pixel. The problem studied and the proposed approach are particularly suited for data sets acquired by Mobile Mapping Systems (MMS): vehicles that can easily image

579 urban scenes by means of optical cameras and lidar sensors. By projecting  
 580 the 3D lidar points onto a chosen reference image, we obtain depth and re-  
 581 flectance images, which suffer of practical issues due to the big diversity of  
 582 the lidar and optical sensor acquisitions. By estimating visibility, we aim at  
 583 solving one of these issues, i.e. the appearance (in depth and reflectance) of  
 584 parts of objects non-visible from the image view point, but captured by the  
 585 lidar sensor. Those points are meant to be detected by our algorithm and  
 586 thus discarded in the inpainting process. The proposed approach consists in  
 587 a variational optimization problem, where three variables (depth, reflectance,  
 588 and visibility) are simultaneously estimated. As a regularization term, a two-  
 589 fold coupled total variation (TV) term is proposed, where the gradients of  
 590 depth, reflectance and color image are matched two by two, by leveraging  
 591 the inherent correlation between them. The proposed algorithm is compared,  
 592 in terms of inpainted images, to other inpainting algorithms, which do not  
 593 take into account the simultaneous detection of possibly erroneous measures.  
 594 The clear superiority of the proposed method w.r.t. the latter proves that  
 595 the visibility estimation is a necessary step. Another comparison is made  
 596 with a simplified version of the algorithm, which accounts for visibility but  
 597 considers alternatively either depth or reflectance. The worse performance of  
 598 the simplified algorithm indicates that the joint exploitation of depth and re-  
 599 flectance is a key aspect for the success of the algorithm. The mutual benefit  
 600 comes from the fact that depth is particularly important for the visibility es-  
 601 timation task; in turn, reflectance is crucial in restoring the correct edges, via  
 602 coupling with the color image. Future work will continue in the direction of  
 603 solving practical issues with lidar-based images to inpaint. Notably, another  
 604 problem is related to disocclusions: the detection of mobile objects is in this  
 605 case necessary to prevent occlusions in the produced depth and reflectance  
 606 images. In this case, we could not only use each point measurement as a vote  
 607 that there is a surface at that point, but we could also exploit the informa-  
 608 tion that the line of sight from the sensor to that point is unobscured, e.g.  
 609 as done in (Xiao et al., 2015). Other work, concerning modifications to the  
 610 model presented in this paper, could consist in considering multiple depth  
 611 candidates per pixel when several points project into the same 2-D location  
 612 (instead of the Z-buffer approach currently used).

## 613 Appendix A. Derivation of the proximal operators in Algorithm 1

614 In this section we detail the derivation of the closed-form expressions of  
 615 the proximal operators for the update of three primal variables ( $v$ ,  $u$ , and  $r$ )  
 616 in Algorithm 1, as listed in Section 4.2. Let  $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$  be a closed  
 617 proper convex function. The proximal operator or mapping  $\text{prox}_f : \mathbb{R}^n \rightarrow \mathbb{R}^n$   
 618 of  $f$  (Parikh and Boyd, 2013) is defined by:

$$\text{prox}_f(v) = \arg \min_{x \in \mathbb{R}^n} f(x) + \frac{1}{2} \|x - v\|_2^2 . \quad (\text{A.1})$$

619 Broadly speaking, the proximal operator of a function is a mathematical  
 620 tool that allows to make an approximation to a certain value, while making  
 621 a compromise between the accuracy of the approximation and a cost given  
 622 by the function itself.

623 Given the general definition (A.1), we can derive the expressions for the  
 624 proximal operators of the functions considered in our algorithm. We have  
 625 that the operations involved are independent for each coordinate of the pro-  
 626 cessed images. Therefore, the expressions reported below are to be intended  
 627 per coordinate, although the spatial indices indicating a particular pixel lo-  
 628 cation are not specified for brevity.

629 •  $\boxed{\text{prox}_1 = \text{prox}_{\sigma D_1^*}(\tilde{p})}$

$\Leftrightarrow$

$$\begin{aligned} \text{prox}_1 &= \arg \min_p \frac{1}{2} (p - \tilde{p})^\top (p - \tilde{p}) - \sigma \beta^\top p + \delta_P(p) \\ &= \arg \min_p \frac{1}{2} p^\top p - \tilde{p}^\top p - \sigma \beta^\top p + K + \delta_P(p) \\ &= \arg \min_p \frac{1}{2} p^\top p - (\tilde{p} + \sigma \beta)^\top p + K + \delta_P(p) \\ &= \arg \min_p \frac{1}{2} (p - (\tilde{p} + \sigma \beta))^\top (p - (\tilde{p} + \sigma \beta)) + K' + \delta_P(p) \\ &= \frac{\tilde{p} + \sigma \beta}{\max(1, \|\tilde{p} + \sigma \beta\|_2)} \end{aligned} \quad (\text{A.2})$$

630 (The variable  $p$ , even considered for a single coordinate, is vector-  
 631 valued.)

632 •  $\boxed{\text{prox}_2 = \text{prox}_{\rho a(\tilde{u}, \cdot) + \rho b(\tilde{r}, \cdot) + \rho C}(\tilde{v})}$

⇔

$$\begin{aligned} \text{prox}_2 = \arg \min_v & \frac{1}{2}(v - \tilde{v})^2 + \rho\eta_1\Phi v \max(0, u_S - \bar{u}) \\ & + \rho\eta_2\Phi v |\bar{r} - r_S| + \rho\alpha\Phi(1 - v) + \delta_{[0,1]}(v) \end{aligned} \quad (\text{A.3})$$

633

◆ If  $\Phi_{i,j} = 0$  (point out of the sparse domain), we trivially have:

$$\begin{aligned} \text{prox}_2 &= \arg \min_v \frac{1}{2}(v - \tilde{v})^2 + \delta_{[0,1]}(v) \\ &= \mathcal{P}_{[0,1]}(\tilde{v}) . \end{aligned} \quad (\text{A.4})$$

634

◆ If  $\Phi_{i,j} = 1$  and  $\bar{u}_{i,j} \geq u_{S\ i,j}$ , we have:

$$\begin{aligned} \text{prox}_2 &= \arg \min_v \frac{1}{2}(v - \tilde{v})^2 + \rho\eta_2v|\bar{r} - r_S| + \rho\alpha(1 - v) + \delta_{[0,1]}(v) \\ &= \arg \min_v \frac{1}{2}v^2 - v\tilde{v} + \rho\eta_2v|\bar{r} - r_S| - \rho\alpha v + K + \delta_{[0,1]}(v) \\ &= \arg \min_v \frac{1}{2}v^2 - v(\tilde{v} + \rho\alpha - \rho\eta_2|\bar{r} - r_S|) + K + \delta_{[0,1]}(v) \\ &= \arg \min_v \frac{1}{2}[v - (\tilde{v} + \rho\alpha - \rho\eta_2|\bar{r} - r_S|)]^2 + K' + \delta_{[0,1]}(v) \\ &= \mathcal{P}_{[0,1]}(\tilde{v} + \rho\alpha - \rho\eta_2|\bar{r} - r_S|) . \end{aligned} \quad (\text{A.5})$$

635

◆ If  $\Phi_{i,j} = 1$  and  $\bar{u}_{i,j} < u_{S\ i,j}$ , we have:

$$\begin{aligned} \text{prox}_2 &= \arg \min_v \frac{1}{2}(v - \tilde{v})^2 + \rho\eta_1v(u_S - \bar{u}) + \rho\eta_2v|\bar{r} - r_S| \\ &\quad + \rho\alpha(1 - v) + \delta_{[0,1]}(v) \\ &= \arg \min_v \frac{1}{2}v^2 - v\tilde{v} + \rho\eta_1v(u_S - \bar{u}) + \rho\eta_2v|\bar{r} - r_S| - \rho\alpha v \\ &\quad + K + \delta_{[0,1]}(v) \\ &= \arg \min_v \frac{1}{2}v^2 - v(\tilde{v} + \rho\alpha - \rho\eta_1(u_S - \bar{u}) - \rho\eta_2|\bar{r} - r_S|) \\ &\quad + K + \delta_{[0,1]}(v) \\ &= \arg \min_v \frac{1}{2}[v - (\tilde{v} + \rho\alpha - \rho\eta_1(u_S - \bar{u}) - \rho\eta_2|\bar{r} - r_S|)]^2 \\ &\quad + K' + \delta_{[0,1]}(v) \\ &= \mathcal{P}_{[0,1]}(\tilde{v} + \rho\alpha - \rho\eta_1(u_S - \bar{u}) - \rho\eta_2|\bar{r} - r_S|) . \end{aligned} \quad (\text{A.6})$$

636

◆ Summing up, we have:

$$\text{prox}_2 = \begin{cases} \mathcal{P}_{[0,1]}(\tilde{v}) & \text{if } \Phi_{i,j} = 0 \\ \mathcal{P}_{[0,1]}(\tilde{v} + \rho\alpha - \rho\eta_2|\bar{r} - r_S|) & \text{if } \Phi_{i,j} = 1, \bar{u}_{i,j} \leq u_{S\ i,j} \\ \mathcal{P}_{[0,1]}(\tilde{v} + \rho\alpha - \rho\eta_1(u_S - \bar{u}) - \rho\eta_2|\bar{r} - r_S|) & \text{if } \Phi_{i,j} = 1, \bar{u}_{i,j} > u_{S\ i,j} \end{cases} . \quad (\text{A.7})$$

637

$$\bullet \quad \boxed{\text{prox}_3 = \text{prox}_{\tau A + \tau a(\cdot, v)}(\tilde{u})}$$

$$\Leftrightarrow$$

$$\begin{aligned} \text{prox}_3 = \arg \min_u & \frac{1}{2}(u - \tilde{u})^2 + \tau\eta_1\Phi \max(0, u - u_S) \\ & + \tau\eta_1\Phi v \max(0, u_S - u) + \delta_{[u_m, u_M]}(u) \end{aligned} \quad (\text{A.8})$$

638

◆ If  $\Phi_{i,j} = 0$ , we trivially have:

$$\begin{aligned} \text{prox}_3 &= \arg \min_u \frac{1}{2}(u - \tilde{u})^2 + \delta_{[u_m, u_M]}(u) \\ &= \mathcal{P}_{[u_m, u_M]}(\tilde{u}) . \end{aligned} \quad (\text{A.9})$$

639

◆ If  $\Phi_{i,j} = 1$  and  $u_{i,j} > u_{S\ i,j}$ , we have:

$$\begin{aligned} \text{prox}_3 &= \arg \min_u \frac{1}{2}(u - \tilde{u})^2 + \tau\eta_1(u - u_S) + \delta_{[u_m, u_M]}(u) \\ &= \arg \min_u \frac{1}{2}u^2 - u\tilde{u} + \tau\eta_1u + K + \delta_{[u_m, u_M]}(u) \\ &= \arg \min_u \frac{1}{2}u^2 - u(\tilde{u} - \tau\eta_1) + K + \delta_{[u_m, u_M]}(u) \\ &= \arg \min_u \frac{1}{2}[u - (\tilde{u} - \tau\eta_1)]^2 + K' + \delta_{[u_m, u_M]}(u) \\ &= \mathcal{P}_{[u_m, u_M]}(\tilde{u} - \tau\eta_1) . \end{aligned} \quad (\text{A.10})$$

By substituting the optimal value found for  $u$  in the splitting condition, we have:

$$u_{i,j} > u_{S\ i,j} \Rightarrow \tilde{u}_{i,j} > u_{S\ i,j} + \tau\eta_1 .$$

640

◆ If  $\Phi_{i,j} = 1$  and  $u_{i,j} < u_{S\ i,j}$ , we have:

$$\begin{aligned} \text{prox}_3 &= \arg \min_u \frac{1}{2}(u - \tilde{u})^2 + \tau\eta_1v(u_S - u) + \delta_{[u_m, u_M]}(u) \\ &= \arg \min_u \frac{1}{2}u^2 - u\tilde{u} - \tau\eta_1vu + K + \delta_{[u_m, u_M]}(u) \\ &= \arg \min_u \frac{1}{2}u^2 - u(\tilde{u} + v\tau\eta_1) + K + \delta_{[u_m, u_M]}(u) \\ &= \arg \min_u \frac{1}{2}[u - (\tilde{u} + v\tau\eta_1)]^2 + K' + \delta_{[u_m, u_M]}(u) \\ &= \mathcal{P}_{[u_m, u_M]}(\tilde{u} + v\tau\eta_1) . \end{aligned} \quad (\text{A.11})$$

By substituting the optimal value found for  $u$  in the splitting condition, we have:

$$u_{i,j} < u_{S\ i,j} \Rightarrow \tilde{u}_{i,j} < u_{S\ i,j} - v\tau\eta_1 .$$

- 641     ♦ The remaining case is:  $\Phi_{i,j} = 1$  and  $u_{i,j} = u_{S\ i,j}$ . This directly  
642     implies the solution for the proximal operator:

$$\text{prox}_2 = \mathcal{P}_{[u_m, u_M]}(u_S) . \quad (\text{A.12})$$

From the previous cases, we can derive the related validity condition on the calculation point  $\tilde{u}_{i,j}$ , i.e.:

$$-v\tau\eta_1 < \tilde{u}_{i,j} - u_{S\ i,j} < \tau\eta_1 .$$

- 643     ♦ Summing up, we have:

$$\text{prox}_3 = \begin{cases} \mathcal{P}_{[u_m, u_M]}(\tilde{u}) & \text{if } \Phi_{i,j} = 0 \\ \mathcal{P}_{[u_m, u_M]}(\tilde{u} - \tau\eta_1) & \text{if } \Phi_{i,j} = 1, \tilde{u}_{i,j} > u_{S\ i,j} + \tau\eta_1 \\ \mathcal{P}_{[u_m, u_M]}(\tilde{u} + v\tau\eta_1) & \text{if } \Phi_{i,j} = 1, \tilde{u}_{i,j} < u_{S\ i,j} - v\tau\eta_1 \\ \mathcal{P}_{[u_m, u_M]}(u_S) & \text{otherwise} \end{cases} \quad (\text{A.13})$$

644     •  $\boxed{\text{prox}_4 = \text{prox}_{\tau B + \tau b(\cdot, v)}(\tilde{r})}$

⇔

$$\text{prox}_4 = \arg \min_r \frac{1}{2}(r - \tilde{r})^2 + \tau\eta_2\Phi v|r - r_S| + \delta_{[r_m, r_M]}(r) \quad (\text{A.14})$$

- 645     ♦ If  $\Phi_{i,j} = 0$ , we trivially have:

$$\begin{aligned} \text{prox}_4 &= \arg \min_r \frac{1}{2}(r - \tilde{r})^2 + \delta_{[r_m, r_M]}(r) \\ &= \mathcal{P}_{[r_m, r_M]}(\tilde{r}) . \end{aligned} \quad (\text{A.15})$$

646

◆ If  $\Phi_{i,j} = 1$  and  $r_{i,j} > r_{S\ i,j}$ , we have:

$$\begin{aligned}
\text{prox}_4 &= \arg \min_r \frac{1}{2}(r - \tilde{r})^2 + \tau\eta_2 v(r - r_S) + \delta_{[r_m, r_M]}(r) \\
&= \arg \min_r \frac{1}{2}r^2 - r\tilde{r} + v\tau\eta_2 r + K + \delta_{[r_m, r_M]}(r) \\
&= \arg \min_r \frac{1}{2}r^2 - r(\tilde{r} - v\tau\eta_2) + K + \delta_{[r_m, r_M]}(r) \\
&= \arg \min_r \frac{1}{2}[r - (\tilde{r} - v\tau\eta_2)]^2 + K' + \delta_{[r_m, r_M]}(r) \\
&= \mathcal{P}_{[r_m, r_M]}(\tilde{r} - v\tau\eta_2) .
\end{aligned} \tag{A.16}$$

By substituting the optimal value found for  $r$  in the splitting condition, we have:

$$r_{i,j} > r_{S\ i,j} \Rightarrow \tilde{r}_{i,j} > r_{S\ i,j} + v\tau\eta_1 .$$

647

◆ If  $\Phi_{i,j} = 1$  and  $r_{i,j} < r_{S\ i,j}$ , we have:

$$\begin{aligned}
\text{prox}_4 &= \arg \min_r \frac{1}{2}(r - \tilde{r})^2 + \tau\eta_2 v(r_S - r) + \delta_{[r_m, r_M]}(r) \\
&= \arg \min_r \frac{1}{2}r^2 - r\tilde{r} - v\tau\eta_2 r + K + \delta_{[r_m, r_M]}(r) \\
&= \arg \min_r \frac{1}{2}r^2 - r(\tilde{r} + v\tau\eta_2) + K + \delta_{[r_m, r_M]}(r) \\
&= \arg \min_r \frac{1}{2}[r - (\tilde{r} + v\tau\eta_2)]^2 + K' + \delta_{[r_m, r_M]}(r) \\
&= \mathcal{P}_{[r_m, r_M]}(\tilde{r} + v\tau\eta_2) .
\end{aligned} \tag{A.17}$$

By substituting the optimal value found for  $r$  in the splitting condition, we have:

$$r_{i,j} < r_{S\ i,j} \Rightarrow \tilde{r}_{i,j} < r_{S\ i,j} - v\tau\eta_1 .$$

648

◆ The remaining case is:  $\Phi_{i,j} = 1$  and  $r_{i,j} = r_{S\ i,j}$ . This directly implies the solution for the proximal operator:

649

$$\text{prox}_3 = \mathcal{P}_{[r_m, r_M]}(r_S) . \tag{A.18}$$

From the previous cases, we can derive the related validity condition on the calculation point  $\tilde{r}_{i,j}$ , i.e.:

$$|\tilde{r}_{i,j} - r_{S\ i,j}| < v\tau\eta_2 .$$



◆ Summing up, we have:

$$\text{prox}_3 = \begin{cases} \mathcal{P}_{[r_m, r_M]}(\tilde{r}) & \text{if } \Phi_{i,j} = 0 \\ \mathcal{P}_{[r_m, r_M]}(\tilde{r} - v\tau\eta_2) & \text{if } \Phi_{i,j} = 1, \tilde{r}_{i,j} > r_{S\ i,j} + v\tau\eta_2 \\ \mathcal{P}_{[r_m, r_M]}(\tilde{r} + v\tau\eta_2) & \text{if } \Phi_{i,j} = 1, \tilde{r}_{i,j} < r_{S\ i,j} - v\tau\eta_2 \\ \mathcal{P}_{[r_m, r_M]}(r_S) & \text{otherwise} \end{cases} \quad (\text{A.19})$$

## References

- Bevilacqua, M., Aujol, J.-F., Brédif, M., Bugeau, A., 2016. Visibility Estimation and Joint Inpainting of Lidar Depth Maps. In: IEEE International Conference on Image Processing (ICIP). pp. 1–5.
- Brédif, M., 2013. Image-Based Rendering of LOD1 3D City Models for traffic-augmented Immersive Street-view Navigation. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences 1 (3), 7–11.
- Chambolle, A., Pock, T., 2011. A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. Journal of Mathematical Imaging and Vision 40 (1), 120–145.
- Chan, D., Buisman, H., Theobalt, C., Thrun, S., 2008. A Noise-Aware Filter for Real-Time Depth Upsampling. In: ECCV Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (M2SFA2). pp. 1–12.
- Chan, T. F., Esedoglu, S., 2005. Aspects of Total Variation Regularized L1 Function Approximation. SIAM Journal on Applied Mathematics 65 (5), 1817–1837.
- Chan, T. F., Golub, G. H., Mulet, P., 1999. A Nonlinear Primal-Dual Method for Total Variation-Based Image Restoration. SIAM Journal on Scientific Computing 20 (6), 1964–1977.
- Chen, W.-Y., Chang, Y.-L., Lin, S.-F., Ding, L.-F., Chen, L.-G., 2005. Efficient Depth Image Based Rendering with Edge Dependent Depth Filter and Interpolation. In: IEEE International Conference on Multimedia and Expo (ICME). pp. 1314–1317.
- Diebel, J., Thrun, S., 2005. An application of Mmarkov random fields to range sensing. In: Advances in Neural Information Processing Systems (NIPS). Vol. 5. pp. 291–298.
- Ferstl, D., Reinbacher, C., Ranftl, R., Rütther, M., Bischof, H., 2013. Image Guided Depth Usampling using Anisotropic Total Generalized Variation. In: IEEE International Conference on Computer Vision (ICCV). pp. 993–1000.
- Garcia, F., Mirbach, B., Ottersten, B., Grandidier, F., Cuesta, A., 2010. Pixel weighted average strategy for depth sensor data fusion. In: 17th IEEE International Conference on Image Processing (ICIP). IEEE, pp. 2805–2808.
- Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets robotics: The KITTI dataset. The International Journal of Robotics Research, 1–6.
- Greene, N., Kass, M., Miller, G., 1993. Hierarchical Z-buffer visibility. In: 20th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). ACM, pp. 231–238.

- 685 Harrison, A., Newman, P., 2010. Image and Sparse Laser Fusion for Dense Scene Recon-  
686 struction. In: Field and Service Robotics (FRS). Springer, pp. 219–228.
- 687 Herbort, S., Wöhler, C., 2011. An introduction to image-based 3D surface reconstruction  
688 and a survey of photometric stereo methods. 3D Research 2 (3), 1–17.
- 689 Huhle, B., Schairer, T., Jenke, P., Straßer, W., Dec. 2010. Fusion of range and color images  
690 for denoising and resolution enhancement with a non-local filter. Computer Vision and  
691 Image Understanding 114 (12), 1336–1345.
- 692 Katz, S., Tal, A., Basri, R., Jul. 2007. Direct Visibility of Point Sets. ACM Transactions  
693 on Graphics (TOG) 26 (3), 24.
- 694 Kolb, A., Barth, E., Koch, R., Larsen, R., 2010. Time-of-Flight Cameras in Computer  
695 Graphics. In: Computer Graphics Forum. Vol. 29. Wiley Online Library, pp. 141–159.
- 696 Liu, J., Gong, X., 2013. Guided Depth Enhancement via Anisotropic Diffusion. In: Ad-  
697 vances in Multimedia Information Processing – PCM 2013. Springer International Pub-  
698 lishing, pp. 408–417.
- 699 Menze, M., Geiger, A., 2015. Object Scene Flow for Autonomous Vehicles. In: IEEE  
700 Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3061–3070.
- 701 Nikolova, M., 2004. A Variational Approach to Remove Outliers and Impulse Noise. Jour-  
702 nal of Mathematical Imaging and Vision 20 (1-2), 99–120.
- 703 Paparoditis, N., Papelard, J.-P., Cannelle, B., Devaux, A., Soheilian, B., David, N.,  
704 Houzay, E., 2012. Stereopolis II: A multi-purpose and multi-sensor 3D mobile mapping  
705 system for street visualisation and 3D metrology. Revue Française de Photogrammétrie  
706 et de Télédétection 1 (200), 69–79.
- 707 Parikh, N., Boyd, S., 2013. Proximal Algorithms. Foundations and Trends in Optimization  
708 1 (3), 123–231.
- 709 Park, J., Kim, H., Tai, Y.-W., Brown, M. S., Kweon, I., 2011. High Quality Depth Map  
710 Upsampling for 3D-TOF Cameras. In: IEEE International Conference on Computer  
711 Vision (ICCV). IEEE, pp. 1623–1630.
- 712 Pierre, F., Aujol, J.-F., Bugeau, A., Papadakis, N., Ta, V.-T., 2015. Luminance-  
713 Chrominance Model for Image Colorization. SIAM Journal on Imaging Sciences (SI-  
714 IMS) 8 (1), 536–563.
- 715 Schmeing, M., Jiang, X., 2011. Depth Image Based Rendering. In: Pattern Recognition,  
716 Machine Intelligence and Biometrics. Springer, pp. 279–310.
- 717 Schneider, N., Schneider, L., Pinggera, P., Franke, U., Pollefeys, M., Stiller, C., 2016. Se-  
718 mantically Guided Depth Upsampling. In: German Conference on Pattern Recognition  
719 (GCPR). Springer, pp. 37–48.

720 Schwarz, S., Sjöström, M., Olsson, R., 2012. Depth Map Upscaling Through Edge  
721 Weighted Optimization. In: Three-Dimensional Image Processing (3DIP) and Appli-  
722 cations II. Vol. 8290. Society of Photo-Optical Instrumentation Engineers (SPIE), pp.  
723 1–8.

724 Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R., Jun. 2006. A Comparison  
725 and Evaluation of Multi-View Stereo Reconstruction Algorithms. In: IEEE Computer  
726 Society Conference on Computer Vision and Pattern Recognition (CVPR). Vol. 1. pp.  
727 519–528.

728 Stoykova, E., Ayd, A., Benzie, P., Grammalidis, N., Malassiotis, S., Ostermann, J., Piekh,  
729 S., Sainov, V., Theobalt, C., Thevar, T., et al., 2007. 3-D time-varying scene capture  
730 technologies—A survey. IEEE Transactions on Circuits and Systems for Video Technol-  
731 ogy 17 (11), 1568–1586.

732 Xiao, W., Vallet, B., Brédif, M., Paparoditis, N., Sep. 2015. Street environment change  
733 detection from mobile laser scanning point clouds. ISPRS Journal of Photogrammetry  
734 and Remote Sensing 107, 38–49.

735 Yang, Q., Ahuja, N., Yang, R., Tan, K.-H., Davis, J., Culbertson, B., Apostolopoulos, J.,  
736 Wang, G., 2013. Fusion of median and bilateral filtering for range image upsampling.  
737 IEEE Transactions on Image Processing 22 (12), 4841–4852.

738 Zhang, Z., 2012. Microsoft Kinect sensor and its effect. IEEE MultiMedia 19 (2), 4–10.

739 Zinger, S., Do, L., de With, P., 2010. Free-viewpoint depth image based rendering. Journal  
740 of Visual Communication and Image Representation 21 (5), 533–541.