



**HAL**  
open science

## Visibility Estimation and Joint Inpainting of Lidar Depth Maps

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

► **To cite this version:**

Marco Bevilacqua, Jean-François Aujol, Mathieu Brédif, Aurélie Bugeau. Visibility Estimation and Joint Inpainting of Lidar Depth Maps. IEEE International Conference on Image Processing (ICIP), Sep 2016, Phoenix, AZ, United States. hal-01316719

**HAL Id: hal-01316719**

**<https://hal.science/hal-01316719>**

Submitted on 17 May 2016

**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.

# VISIBILITY ESTIMATION AND JOINT INPAINTING OF LIDAR DEPTH MAPS

Marco Bevilacqua<sup>1,2,3</sup>, Jean-François Aujol<sup>1</sup>, Mathieu Brédif<sup>3</sup>, Aurélie Bugeau<sup>2</sup>

<sup>1</sup>Université de Bordeaux, IMB, CNRS UMR 5251, INP, 33400 Talence, France.

<sup>2</sup>Université de Bordeaux, LaBRI, CNRS UMR 5800, 33400 Talence, France.

<sup>3</sup>Université Paris-Est, IGN, SRIG, MATIS, 73 Avenue de Paris, 94160 Saint-Mandé, France.

## ABSTRACT

This paper presents a novel variational image inpainting method to solve the problem of generating, from 3-D lidar measures, a dense depth map coherent with a given color image, tackling visibility issues. When projecting the lidar point cloud onto the image plane, we generally obtain a sparse depth map, due to undersampling. Moreover, lidar and image sensor positions generally differ during acquisition, such that depth values referring to objects that are hidden from the image view point might appear with a naive projection. The proposed algorithm estimates the complete depth map, while simultaneously detecting and excluding those hidden points. It consists in a primal-dual optimization method, where a coupled total variation regularization term is included to match the depth and image gradients and a visibility indicator handles the selection of visible points. Tests with real data prove the effectiveness of the proposed strategy.

**Index Terms**— Inpainting, Total Variation, Depth Maps, Lidar, Point Cloud, Visibility

## 1. INTRODUCTION

This work is particularly driven by the exploitation of images and lidar point clouds acquired by Mobile Mapping Systems (MMS, such as [1]). These vehicles provide high quality calibrated and precisely geolocated images, together with coherent lidar point clouds acquired while driving through the streets at regular traffic speeds. Beyond all the modelization and urban planning applications, one might be interested in visualizing directly these raw datasets, without having to perform an error-prone reconstruction of a globally coherent 3D surface textured with the images. If all image pixels had a good depth estimate, depth-image-based rendering techniques (e.g. [2, 3, 4]) might be used to generate nearby views by blending warped images according to their depth information. Likewise, having the depth of each pixel at hand enables the insertion of virtual elements into the image such as pedestrians or vehicles generated by a traffic simulation [5]. MMS datasets however do not provide directly depth images perfectly aligned with the color images. While it is possible to project the lidar point clouds to each image, this yields a sparse depth map with the following three difficulties.

- Undersampling: lidar and image acquisition geometries and characteristics dictate that the sparse depth map resulting from the projection of the points onto the calibrated image is irregular. No points are present in the sky or reflective

surfaces and the projected point density depends on the distances between the surface of the image and the positions of the lidar sensor.

- Visibility: even when using a Z-buffer approach [6] that only keeps the closest depth for each pixel, points of background layers may be visible in between projected points of a foreground layer.
- Disocclusion and moving objects: lidar and image sensors not being co-located, an image may view surface patches that were not visible by the lidar sensor due to occlusions or the scene dynamics.

Our work lies in this context. Given an input sparse depth map, our algorithm aims at tackling the first two of the above-mentioned issues: the input depth map is fully inpainted (except for those areas largely uncovered, e.g. the sky), and points from background layers yet visible are automatically discarded. This is done in an optimization framework by solving an image inpainting problem, where a visibility indicator is simultaneously estimated to enable only the visible points. The inpainting process is also guided by the available color image, in the sense that a coupled total variation (TV) regularizer is included in the model to couple the depth and image gradients. Fig. 1 reports an example of the data involved in this problem: an incomplete depth map (c), which is subsequently inpainted by our method, is obtained by projecting real 3-D lidar measures (a) w.r.t. an existing image view point (b). The boxes superposed on the depth map indicate some of the issues that depth maps computed in this way may suffer.

The remainder of the paper is organized as follows. Section 2 revises the works related to the targeted problem and summarizes our contributions. Section 3 details the algorithm proposed. Before drawing conclusions, Section 4 describes the tests carried out to validate our approach.

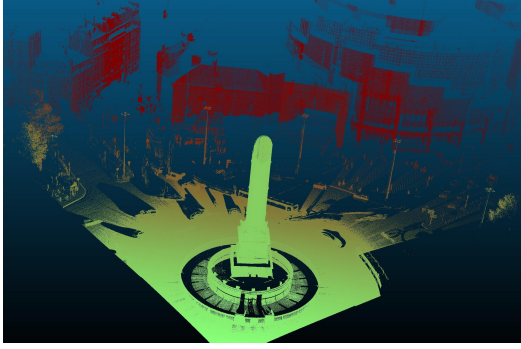
## 2. RELATED WORKS AND CONTRIBUTIONS

### 2.1. Related works

This work deals with the inpainting of depth maps extracted from lidar measures in a realistic scenario. While there exist several methods in the literature in the field of image-guided upsampling/super-resolution of low-resolution depth images, often obtained with Time of Flight (ToF) cameras (e.g., [7, 8, 9, 10]), few methods deal with the problem of upsampling depth images forming a sparse irregular grid (which can be referred to as an inpainting problem). Among the latter, in [11] a method to assign image pixel with a range value, using both image appearance and sparse laser data, is proposed. The problem is posed as an optimization of a cost function encapsulating a spatially varying smoothness cost and measurement compatibility. Similarly, the authors in [12] propose a depth upsampling method

---

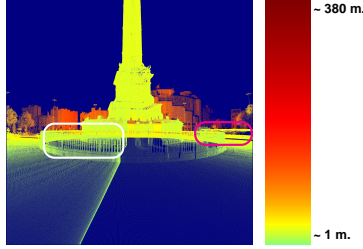
This study has been carried out with financial support from the French State, managed by the French National Research Agency (ANR) in the frame of the Investments for the future Programme IdEx Bordeaux (ANR-10-IDEX-03-02). J.-F. Aujol also acknowledges the support of the Institut Universitaire de France.



(a) Input lidar point cloud



(b) Intensity image



(c) Input depth map

**Fig. 1.** Example of test data case. Given a 3-D point cloud (a) and a reference image (b), a corresponding input depth map (c) is obtained by simple projection. The boxes highlight issues present in the depth map, i.e. hidden points appearing (white box) and moving objects (magenta box).

that results in an optimization problem with an Anisotropic Total Generalized Variation (ATGV) term. However, in none of the two methods the visibility issue is tackled, i.e. there is no estimation of input depth measures to possibly remove.

The hidden point removal (HPR) problem of assessing which is the subset of visible points from a given view point is typically addressed as an independent step. For HPR the state of the art is represented by variations of [13] that relate the visible point set to the convex hull of a viewpoint-dependent transformation of it, discarding points based on a concavity threshold as seen from the view point. While this approach is effective, there is in general no globally satisfactory concavity threshold that would both correctly detect hidden surfaces and keep background points close to foreground silhouettes. Our work is a step forward, in the sense that it attempts at integrating the HPR task into the depth inpainting problem.

## 2.2. Contributions

Our method is the first one to fuse two problems: hidden point removal and depth image inpainting. To this end, we propose a novel model that originally encodes a visibility attribute for each pixel, and leads to an optimization problem where the output depth map is estimated accordingly. Another contribution relates to the employment of a coupled TV regularizer [14], to guide the depth inpainting according to the available color image.

## 3. METHOD

In this section we present our inpainting method to generate a complete depth map corresponding to a known intensity image. Here-

inafter, we use the following notation. Let  $\Omega \subseteq \mathbb{R}^2$  be the “full” image support, and  $\Omega_S \subseteq \Omega$  the sparse image support where the input depth map is defined. Given the initial map  $y : \Omega_S$  and its corresponding intensity image  $w : \Omega$ , the goal is then to fully inpaint the depth measures available to obtain  $u : \Omega$ , and simultaneously estimate a visibility indicator  $v : \Omega_S$ , which, for each input point, indicates whether it is visible from the image view point and should thus be taken into account in the inpainting process.

### 3.1. Model proposed

The optimization problem we propose, to concurrently estimate an inpainted depth map  $u$  and a visibility indicator  $v$  consists of three parts (1): a data term  $F(u, v)$ , a “removal cost” term  $G(v)$ , and a TV-like regularizer.

$$\min_{\substack{u \in [u_m, u_M] \\ v \in [0, 1]}} F(u, v) + G(v) + \beta \text{TV}_\lambda(u, w) \quad (1)$$

For  $u$ , a lower and upper bound to the depth,  $u_m$  and  $u_M$  respectively, can be considered, whereas  $v \in [0, 1]$ , where  $v = 0$  stands for “hidden” and  $v = 1$  tells that the point is visible.

To address the problem of detecting visible points, we introduce the following data term:

$$\begin{aligned} F(u, v) &= \int_{\Omega_S} (\max(0, u - y))^2 dx_1 dx_2 + \int_{\Omega_S} v (\max(0, y - u))^2 dx_1 dx_2 \\ &= F_1(u) + F_2(u, v) \end{aligned} \quad (2)$$

where we treat separately over- and under-estimated depths (in  $F_1$  and  $F_2$ , respectively). If we suppose that the lidar measures are sufficiently dense, and thus background depths are still surrounded by correct depth values referring to closer visible objects, we can assume that for background points the estimated surface will be in front ( $u < y$ ). Conversely, for the correct foreground depths we will have over-estimated values ( $u > y$ ), i.e. the estimated surface will lie behind the related points. The proposed data matching term fully penalizes the latter case, in order the estimated surface to pass as close as possible by the visible points, whereas under-estimated depths are penalized according to the visibility indicator  $v$ . Ideally, a fraction of them, the most “problematic” points, will be classified as hidden ( $v = 0$ ) and thus be excluded from the inpainting process.

The second term of our model (3) is meant to penalize the number of removed points:

$$G(v) = \int_{\Omega_S} \alpha(y)(1 - v) dx_1 dx_2. \quad (3)$$

The cost of each point removal is weighted by a coefficient dependent on the original depth,  $\alpha(y)$ . We choose  $\alpha = (ky)^2$ . In this way, maybe counter-intuitively, it costs more to classify as hidden faraway points, since these are the points for which depth estimations are more susceptible to errors. The quadratic dependence of  $\alpha$  on the depth balances the terms  $F(u, v)$  and  $G(v)$  and make the parameter depend on the choice of an a-dimensional constant  $k$ .

Finally, for the regularization term, we adopt the coupled total variation [14]:

$$\text{TV}_\lambda(u, w) = \int_{\Omega} \sqrt{(\partial_{x_1} u)^2 + (\partial_{x_2} u)^2 + \lambda^2 (\partial_{x_1} w)^2 + \lambda^2 (\partial_{x_2} w)^2} dx_1 dx_2. \quad (4)$$

Here, the depth and image gradients are encouraged to align. As the coupling parameter  $\lambda$  increases, the gradient “jumps” in the two images will occur at the same locations.

### 3.2. Primal-dual optimization scheme

The optimization problem (1) is convex but not smooth due to the coupled TV term (4). To find a fast solution we adopt a primal-dual optimization scheme, as proposed in [14, 15]. Let  $M$  and  $N$  be the image dimensions in the discrete setting. We then have  $u, y, v, w, \alpha \in X = \mathbb{R}^{MN}$ , and we can express the coupled TV in terms of a dual variable  $q \in Y = \mathbb{R}^{4MN}$ , as follows:

$$\begin{aligned} \text{TV}(u, w) &= \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} \sqrt{(\nabla u_{i,j}^H)^2 + (\nabla u_{i,j}^V)^2 + \lambda^2 (\nabla w_{i,j}^H)^2 + \lambda^2 (\nabla w_{i,j}^V)^2} \\ &= \max_{q \in Y} \langle K^e u, q \rangle_Y - \delta_Q(q) \end{aligned} \quad (5)$$

where  $K^e$  is the ‘‘coupled gradient’’ operator with the following element-wise definition

$$(K^e u)_{i,j} = ((\nabla u^H)_{i,j}, (\nabla u^V)_{i,j}, \lambda(\nabla w^H)_{i,j}, \lambda(\nabla w^V)_{i,j}), \quad (6)$$

and  $q$  is a twofold dual variable ( $q_{i,j} = (p_{i,j}, \tilde{p}_{i,j})$ ,  $p, \tilde{p} \in \mathbb{R}^{2MN}$ ) defined by the feasibility set

$$Q = \{q \in Y \mid \|q_{i,j}\| \leq 1, \forall i, j\}. \quad (7)$$

Given the previous definitions, we can see our problem (1) as a particular case of the following saddle-point problem:

$$\min_{u \in \mathcal{X}} \min_{v \in \mathcal{X}} \max_{q \in \mathcal{Y}} \langle Ku|q \rangle - F^*(q) + A(u) + b(u, v) + C(v). \quad (8)$$

The following equivalences stand:

- $F^*(q) = \delta_Q(q)$ ;
- $A(u) = F_1(u) + \chi_{u \in [u_m, u_M]}$ ;
- $b(u, v) = F_2(u, v)$ ;
- $C(v) = G(v) + \chi_{v \in [0, 1]}$ .

The generic procedure to solve (8) is provided in Algorithm 1. The algorithm consists in a unique loop, where  $u$  and  $v$  are alternatively updated ( $u_0$  is an initial estimate of the output depth map, e.g. obtained via interpolation). The proximal operators involved are easily derivable as closed-form expressions [14]. The algorithm also requires four parameters:  $\theta$  is a relaxation parameter (typically  $\theta = 1$ ),  $\sigma$  and  $\tau$  are related to each other by the relation  $16\tau\sigma \leq 1$ , and  $\rho$  is a parameter that regulates the update speed of  $v$ .

---

**Algorithm 1** Primal-dual based algorithm for lidar depth map inpainting.

---

- 1: **Inputs:**  
 $u_0, w, \sigma, \rho, \tau, \theta$
  - 2: **Initialize:**  
 $u^0, \bar{u}^0 \leftarrow u_0, v_{i,j}^0 \leftarrow 0.5, q^0 \leftarrow (\nabla u_0, \lambda \nabla w)$
  - 3: **for**  $n = 0, 1, \dots$  **do**
  - 4:  $q^{n+1} \leftarrow \text{prox}_{\sigma F^*}(q^n + \sigma K \bar{u}^n)$
  - 5:  $v^{n+1} \leftarrow \text{prox}_{\rho A + \rho b(\bar{u}^n, \cdot)}(v^n)$
  - 6:  $u^{n+1} \leftarrow \text{prox}_{\tau C + \tau b(\cdot, v^{n+1})}(u^n - \tau K^* p^{n+1})$
  - 7:  $\bar{u}^{n+1} \leftarrow u^{n+1} + \theta(u^{n+1} - u^n)$
  - 8: **end for**
- 

## 4. EVALUATION

We evaluate our algorithm in terms of quality of the generated depth maps and w.r.t. the hidden point removal problem.

### 4.1. Inpainting of lidar depth maps

We consider a data set acquired by a MMS system [1], consisting of one lidar point cloud in the order of one billion of points and hundreds of optical image simultaneously acquired by the 5 cameras mounted on the vehicle. Given a chosen reference image, we project the lidar points available to form the initial depth map, only a fraction of them effectively ending up on the image plane. This image, along with the image gradient, represents the input of the algorithm.

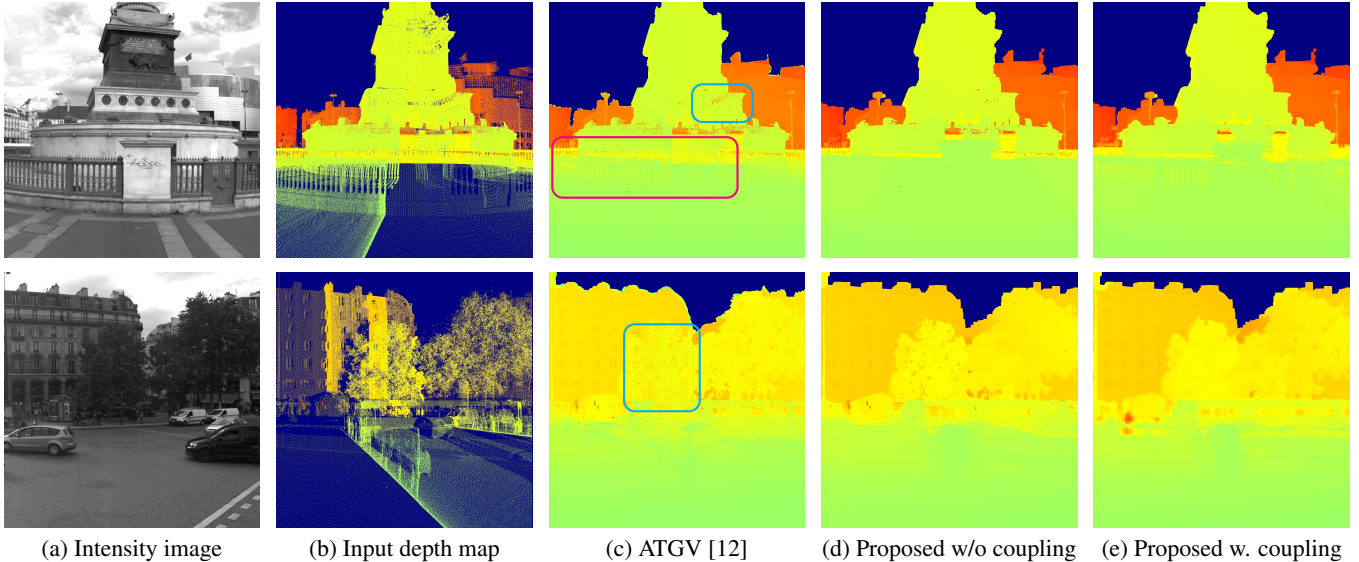
Fig. 2 reports results for two images, cropped to better highlight the details. Columns (a) and (b) show the intensity of the reference color image and the associated input lidar-based depth map, respectively. Our complete method is visually compared with the ATGV method [12] and with a variant of our algorithm that does not consider any coupling in the TV term (i.e we have  $\lambda = 0$  in (4)). As for the complete algorithm, in all the evaluations of this paper the following parameters have been employed:  $\beta = 0.05$ ,  $\lambda = 15$ , and  $k = 0.03$  (remind that the coefficient of the removal cost term  $G(v)$  is  $\alpha = (ky)^2$ ). In the first example of Fig. 2, the main difficulties are represented by the ‘‘fence’’, where multiple depth layers are mixed. The ATGV method [12] fails at resolving the conflicts in this area. With our method we are instead able to exclude most of the hidden background points (see particularly the points in the top part of the magenta box), and the inpainted map is more coherent with the image view point chosen. The need for coupling in the TV regularizer is also clear. In the result without coupling, we lose some details in the depth image (e.g. see the fences of the column), which are retrieved in our final result. By coupling the depth map with the intensity image, we are also able to better detect hidden points. In correspondence of the cyan box, we can see how depth values related to hidden points progressively disappear from (c) to (e). As for the second example of Fig. 2, a significant area to look at is in correspondence of the tree in the middle (cyan box). With our method we are able to correctly show the foreground depths of this area. This does not imply any detectable loss of details in other regions where the multiple layer issue does not occur.

Both images however present some issues related to disocclusions and moving objects. In the first example depth values related to invisible objects still remain (see around the middle of the image). In these small areas, the total absence of first-layer depths makes the removal of hidden point depths difficult. Similarly, the second example suffers of the presence of mobile objects. Depth values related to a moving bus in fact appear, although it is not present in the original image. While efficiently handling the visibility issue, our algorithm does not (yet) deal with these difficulties.

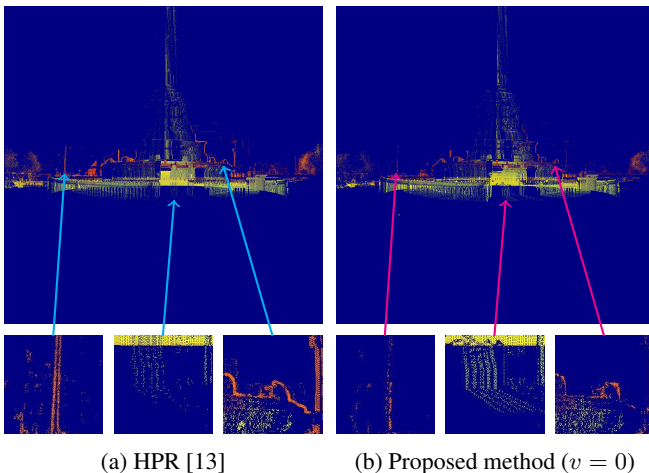
### 4.2. Performance on the hidden point removal task

As pointed out in Section 2, our algorithm attempts at fusing the hidden point removal (HPR) and the inpainting problems in an original way. To prove the effectiveness of the designed strategy w.r.t. the HPR task, we compare with the well-established method of [13], which removes the hidden points directly in the 3-D geometry according to a global concavity threshold. Fig. 3 reports the depth values of the detected hidden points for the column image, in the case of [13] and with our algorithm. In our case, we consider hidden points those depth values that are assigned  $v = 0$  at the end. These points are effectively discarded in the inpainting process.

Fig. 3 shows remarkable differences between the two methods (see the zoomed-in areas). The method of [13], in fact, wrongly selects points around the silhouettes (see first and third patch), while being more conservative in selecting actual hidden points (see patch in the middle). Our method performs better in both cases. As for



**Fig. 2.** Generated complete depth maps, for two reference images and related input depth maps, with: ATGV [12] (c), our method without coupling with the image (d), and our complete method (e). In the first-row example, we can see that the proposed method removes all the misleading depths referring to hidden points (cyan box), and better estimates depth values in mixed-layer areas (magenta box). In the second-row example, we can observe that our method better retrieves the depths of the tree in front (cyan box).



**Fig. 3.** Detected hidden points in the case of the “column image”, by the state-of-the-art method of [13] and our method. The three patches below each image represent zoomed-in areas of the images themselves at same locations. With the proposed algorithm we are able to detect more hidden points (see the center patch), while being more careful around the silhouettes (first and third patch).

[13], a concavity parameter equal to 4 has been chosen after tuning. The change of this parameter could improve one of the two aspects, but would inevitably make the second one worse. Therefore, we can say that the “quality” of the hidden point selection achieved by our algorithm is higher, thus suggesting that performing a soft selection of the points while inpainting is preferable.

## 5. CONCLUSION

In this paper we presented a novel strategy to generate a complete depth map associated to a color image, given as input a sparse depth map. The proposed method is particularly suited for addressing practical issues with depth maps obtained from 3-D lidar measures acquired by Mobile Mapping Systems (MMS). When projecting 3-D lidar points onto a reference image, the computed map typically suffers of three problems: under-sampling, visibility incoherences (i.e. the appearance of points that are not visible from the image view point), and presence of mobile objects. Our method efficiently tackles the first two issues. The produced depth map has the same pixel resolution as the reference image, and possible hidden points are sought and removed. This is done by solving an optimization problem that concurrently estimates the complete depth map and a visibility indicator. Coupled total variation is also used as a regularization term to align the depth and image gradients. Tests with real datasets show that the designed model leads to improved inpainted depth maps. In particular, when looking at the quality of the hidden point removals, we observe that we reach more satisfactory results than the state of the art [13]. Our method better preserves the image contours, while detecting more hidden points in other regions. As this work is a first step into the problem of image inpainting of lidar-based depth maps, there are several future directions. Future work will include the implementation of strategies to handle moving objects present in the scene (the third issue) and the simultaneous synthesis of depth maps from multiple views. As lidar measures carry out also information on the reflectance of the objects, one might think about inpainting a reflectance image as well, and use the inpainted reflectance in the coupling term. Finally, another modification to the model could involve the use of the  $l_1$ -norm instead of the  $l_2$ -norm in the data term.

## 6. REFERENCES

- [1] Nicolas Paparoditis, Jean-Pierre Papelard, Bertrand Cannelle, Alexandre Devaux, Bahman Soheilian, Nicolas David, and Erwann Houzay, “Stereopolis II: A multi-purpose and multi-sensor 3D mobile mapping system for street visualisation and 3D metrology,” *Revue Française de Photogrammétrie et de Télédétection*, vol. 1, no. 200, pp. 69–79, 2012.
- [2] Sveta Zinger, Luat Do, and PHN de With, “Free-viewpoint depth image based rendering,” *Journal of Visual Communication and Image Representation*, vol. 21, no. 5, pp. 533–541, 2010.
- [3] Wan-Yu Chen, Yu-Lin Chang, Shyh-Feng Lin, Li-Fu Ding, and Liang-Gee Chen, “Efficient Depth Image Based Rendering with Edge Dependent Depth Filter and Interpolation,” in *IEEE International Conference on Multimedia and Expo (ICME)*, 2005, pp. 1314–1317.
- [4] Michael Schmeing and Xiaoyi Jiang, “Depth Image Based Rendering,” in *Pattern Recognition, Machine Intelligence and Biometrics*, pp. 279–310. Springer, 2011.
- [5] Mathieu Brédif, “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*, vol. 1, no. 3, pp. 7–11, 2013.
- [6] Ned Greene, Michael Kass, and Gavin Miller, “Hierarchical Z-buffer visibility,” in *20th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*. ACM, 1993, pp. 231–238.
- [7] Frederic Garcia, Bruno Mirbach, Bjorn Ottersten, Frédéric Grandidier, and Angel Cuesta, “Pixel weighted average strategy for depth sensor data fusion,” in *17th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2010, pp. 2805–2808.
- [8] Jaesik Park, Hyeonwoo Kim, Yu-Wing Tai, Michael S Brown, and Inso Kweon, “High Quality Depth Map Upsampling for 3D-TOF Cameras,” in *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2011, pp. 1623–1630.
- [9] Benjamin Huhle, Timo Schairer, Philipp Jenke, and Wolfgang Straßer, “Fusion of range and color images for denoising and resolution enhancement with a non-local filter,” *Computer Vision and Image Understanding*, vol. 114, no. 12, pp. 1336–1345, Dec. 2010.
- [10] Qingxiong Yang, Ruigang Yang, James Davis, and David Nistér, “Spatial-Depth Super Resolution for Range Images,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2007, pp. 1–8.
- [11] Alastair Harrison and Paul Newman, “Image and Sparse Laser Fusion for Dense Scene Reconstruction,” in *Field and Service Robotics (FRS)*. Springer, 2010, pp. 219–228.
- [12] David Ferstl, Christian Reinbacher, Rene Ranftl, Matthias Rütther, and Horst Bischof, “Image Guided Depth Usampling using Anisotropic Total Generalized Variation,” in *IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 993–1000.
- [13] Sagi Katz, Ayellet Tal, and Ronen Basri, “Direct Visibility of Point Sets,” *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, pp. 24, July 2007.
- [14] Fabien Pierre, Jean-François Aujol, Aurélie Bugeau, Nicolas Papadakis, and Vinh-Thong Ta, “Luminance-Chrominance Model for Image Colorization,” *SIAM Journal on Imaging Sciences (SIIMS)*, vol. 8, no. 1, pp. 536–563, 2015.
- [15] Antonin Chambolle and Thomas Pock, “A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging,” *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.