



Range-Image: Incorporating sensor topology for LiDAR point cloud processing

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

► To cite this version:

Pierre Biasutti, Jean-François Aujol, Mathieu Brédif, Aurélie Bugeau. Range-Image: Incorporating sensor topology for LiDAR point cloud processing. Photogrammetric engineering and remote sensing, 2018, 84 (6), pp.367–375. 10.14358/PERS.84.6.367 . hal-01756975

HAL Id: hal-01756975

<https://hal.science/hal-01756975>

Submitted on 3 Apr 2018

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.

RANGE-IMAGE: INCORPORATING SENSOR TOPOLOGY FOR LiDAR POINT CLOUD PROCESSING

P. Biasutti¹²³, J-F. Aujol¹, M. Brédif³, A. Bugeau²

¹ *Université de Bordeaux, IMB, CNRS UMR 5251, INP, 33400 Talence, France.*

² *Université de Bordeaux, LaBRI, CNRS UMR 5800, 33400 Talence, France.*

³ *Université Paris-Est, LASTIG MATIS, IGN, ENSG, F-94160 Saint-Mandé, France*
{pierre.biasutti, jean-francois.aujol}@math.u-bordeaux.fr,
mathieu.bredif@ign.fr,
aurelie.bugeau@labri.fr

Abstract

This paper proposes a novel methodology for LiDAR point cloud processing that takes advantage of the implicit topology of various LiDAR sensors to derive 2D images from the point cloud while bringing spatial structure to each point. The interest of such a methodology is then proved by addressing the problems of segmentation and disocclusion of mobile objects in 3D LiDAR scenes acquired via street-based Mobile Mapping Systems (MMS). Most of the existing lines of research tackle those problems directly in the 3D space. This work promotes an alternative approach by using this image representation of the 3D point cloud, taking advantage of the fact that the problem of disocclusion has been intensively studied in the 2D image processing community over the past decade. Using the image derived from the sensor data by exploiting the sensor topology, a semi-automatic segmentation procedure based on depth histograms is presented. Then, a variational image inpainting technique is introduced to reconstruct the areas that are occluded by objects. Experiments and validation on real data prove the effectiveness of this methodology both in terms of accuracy and speed.

1. INTRODUCTION

Over the past decade, street-based Mobile Mapping Systems (MMS) have encountered a large success as the onboard 3D sensors are able to map full urban environments with a very high accuracy. These systems are now widely used for various applications from urban surveying to city modeling (Serna and Marcotegui, 2013; Hervieu et al., 2015; El-Halawany et al., 2011; Hervieu and Soheilian, 2013; Goulette et al., 2006). Several systems have been proposed in order to perform these acquisitions. They mostly consist in optical

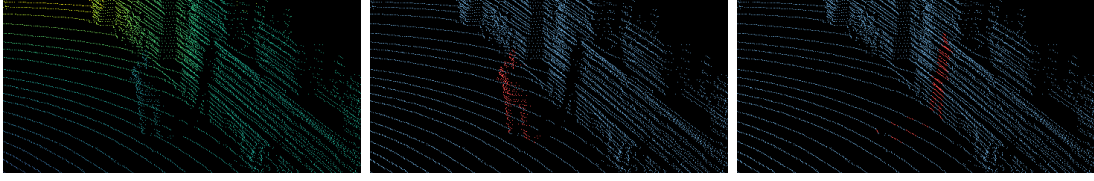


Figure 1: Result of the segmentation and the disocclusion of a pedestrian in a point cloud using range images. (left) original point cloud, (center) segmentation using range image, (right) disocclusion using range image. The pedestrian is correctly segmented and its background is then reconstructed in a plausible way.

cameras, 3D LiDAR sensor and GPS combined with Inertial Measurement Unit (IMU), built on a vehicle for mobility purposes (Paparoditis et al., 2012; Geiger et al., 2013). They provide multi-modal data that can be merged in several ways, such as LiDAR point clouds colored by optical images or LiDAR depth maps aligned with optical images. Although these systems lead to very complete 3D mapping of urban scenes by capturing optical and 3D details (pavements, walls, trees, etc.), providing billions of 3D points and RGB pixels per hour of acquisition, they often require further processing to suit their ultimate usage.

For example, MMS tend to acquire mobile objects that are not persistent to the scene. This often happens in urban environments with objects such as cars, pedestrians, traffic cones, etc. As LiDAR sensors cannot penetrate opaque objects, those mobile objects cast shadows behind them where no point has been acquired (Figure 1, left). Therefore, merging optical data with the point cloud can be ambiguous as the point cloud might represent objects that are not present in the optical image. Moreover, these shadows are also largely visible when the point cloud is not viewed from the original acquisition point of view. This might end up being distracting and confusing for visualization. Thus, the segmentation of mobile objects and the reconstruction of their background remain strategic issues in order to improve the understanding of urban 3D scans.

We argue that working on simplified representations of the point cloud enables specific problems such as disocclusion to be solved not only using traditional 3D techniques but also using techniques brought by other communities (image processing in our case). Exploiting the sensor topology also brings spatial structure into the point cloud that can be used for other applications such as segmentation, remeshing, colorization or registration.

The main contribution of this paper is a novel methodology for point cloud processing by exploiting the implicit topology of various LiDAR sensors that can be used to infer a simplified representation of the LiDAR point cloud while bringing spatial structure between every points. The utility of such a methodology is here demonstrated by two applications.

First, a fast segmentation technique for dense and sparse point clouds to extract full
35 objects from the scene is presented (Figure 1, center). Then, we introduce a fast and
efficient variational method for the disocclusion of a point cloud using the range image
representation while taking advantage of a horizontal prior without any knowledge of the
color or texture of the represented objects (Figure 1, right).

This paper is an extension of Biasutti et al. (2017) with improved technical details
40 of the methodology as well as a complete validation of the proposed applications and a
discussion about its limitations.

The paper is organized as follows: after a review on the state-of-the-art of the two
application scenarios (Section 2), we detail how the topology of various sensors can be
exploited to turn a regular LiDAR point cloud into a range image (Section 3). In Section 4,
45 a point cloud segmentation model using range images is introduced with corresponding
results and a validation on several datasets. Then, a disocclusion method for point clouds
is presented in Section 5 as well as results and validation on various datasets. Finally
conclusions are drawn and potential future work is identified in Section 6.

2. Related Works

50 The growing interest for MMS over the past decade has lead to many works and con-
tributions for solving problems that could be tackled using range images. In this part, we
present a state-of-the-art on both segmentation and disocclusion.

2.1. Point cloud segmentation

The problem of point cloud segmentation has been extensively addressed in the past
55 years. Three types of methods have emerged: geometry-based techniques, statistical tech-
niques and techniques based on simplified representations of the point cloud.

Geometry-based segmentation. The first well-known method in this category is region-
growing where the point cloud is segmented into various geometric shapes based on the
neighboring area of each point (Huang and Menq, 2001). Later, techniques that aim at
60 fitting primitives (cones, spheres, planes, cubes ...) in the point cloud using RANSAC
(Schnabel et al., 2007) have been proposed. Others look for smooth surfaces (Rabbani
et al., 2006). Although these methods do not need any prior about the number of objects,
they often suffer from over-segmenting the scene resulting in objects segmented in several
parts.

65 *Semantic segmentation.* The methods in this category analyze the point cloud characteristics (Demantke et al., 2011; Weinmann et al., 2015; Landrieu et al., 2017). They analyze the geometric neighborhood of each point in order to perform a point-wise classification, possibly with spatial regularisation, which, in turn, yields a semantic segmentation. It leads to a good separation of points that belong to static and mobile objects, but not to
70 the distinction between different objects of the same class.

Simplified model for segmentation. MMS LiDAR point clouds typically represent massive amounts of unorganized data that are difficult to handle. Different segmentation approaches based on a simplified representation of the point cloud have been proposed. Papon et al. (2013) propose a method in which the point cloud is first turned into a set
75 of voxels which are then merged using a variant of the SLIC algorithm for super-pixels in 2D images (Achanta et al., 2012). This representation leads to a fast segmentation but it might fail when the scale of the objects in the scene is too different. Gehring et al. (2017) propose to extract moving objects from MLS data by using a probabilistic volumetric representation of the MLS data in order to cluster points between mobile objects
80 and static objects. However this technique can only be used with 3D sensors. Another simplified model of the point cloud is presented by Zhu et al. (2010). The authors take advantage of the implicit topology of the sensor to simplify the point cloud in order to segment it before performing classification. The segmentation is done through a graph-based method as the notion of neighborhood is easily computable on a 2D image. Although the
85 provided segmentation algorithm is fast, it suffers from the same issues as geometry-based algorithms such as over-segmentation or incoherent segmentation. Finally, an approach for urban objects segmentation using elevation images is proposed in Serna and Marcotegui (2014). There, the point cloud is simplified by projecting its statistics onto a horizontal grid. Advanced morphological operators are then applied on the horizontal grid and objects
90 are segmented using a watershed approach. Although this method provides good results, the overall precision of the segmentation is limited by the resolution of the projection grid and leads to the occurrence of artifacts at object borders.

Moreover, all those categories of segmentation techniques are not able to treat efficiently both dense and sparse LiDAR point clouds *i.e.* point clouds acquired with high or low
95 sampling rates compared to the real-world feature sizes (*e.g.* macroscopic objects such as cars, pedestrians, etc.). For example, one sensor turn in the KITTI dataset (Geiger et al., 2013) corresponds to $\simeq 10^5$ points (sparse) whereas for a scene of similar size in

the Stereopolis-II dataset (Paparoditis et al., 2012), the scene contains more than $4 \cdot 10^6$ points (dense). In this paper, we present a novel simplified model for segmentation based
100 on histograms of depth in range images by leveraging grid-like topology without suffering from accuracy loss that is often caused by projection/rasterization.

2.2. Disocclusion

Disocclusion of a scene has only been scarcely investigated for 3D point clouds (Sharf et al., 2004; Park et al., 2005; Becker et al., 2009). These methods generally work on
105 complete point clouds (with homogeneous sampling) rather than LiDAR point clouds. This task, also referred to as inpainting, has been much more studied in the image processing community. Over the past decades, various approaches have emerged to solve the problem in different manners. Patch-based methods such as the one proposed by Criminisi et al. (2004) (and more recently Lorenzi et al. (2011) and Buyssens et al. (2015b)) have proven
110 their strengths. They have been extended for RGB-D images (Buyssens et al., 2015a) and to LiDAR point clouds (Doria and Radke, 2012) by considering an implicit topology in the point cloud. Variational approaches represent another type of inpainting algorithms (Weickert, 1998; Bertalmio et al., 2000; Bredies et al., 2010; Chambolle and Pock, 2011). They have been extended to RGB-D images by taking advantage of the bi-modality of the
115 data (Ferstl et al., 2013; Bevilacqua et al., 2017). Even if the results of the disocclusion are quite satisfying, these models require the point cloud to have color information as well as the 3D data. In this work, we introduce an improvement to a variational disocclusion technique by taking advantage of a horizontal prior.

3. Range images derived from the sensor topology

120 In this paper, we demonstrate that a simplified model of the point cloud can be directly derived from it using the intrinsic topology of the sensing pattern during acquisition. This section introduces this sensor topology and how it can be exploited on various kinds of sensors. Examples of its usages are presented.

3.1. Sensor topology

125 Most of modern LiDAR sensors offer an intrinsic 2D topology in raw acquisitions. However, this feature is rarely considered in recent works. Namely, LiDAR points may obviously be ordered along scanlines, yielding the first dimension of the sensor topology, linking each LiDAR pulse to the immediately preceding and succeeding pulses within the

same scanline. For most LiDAR devices, one can also order the consecutive scanlines. It
 130 amounts to considering a second dimension of the sensor topology across the scanlines as
 it can be seen in Figure 2.

3.2. From sensor topology to range image

The sensor topology often varies with the type of LiDAR sensor that is being used. 2D
 LiDAR sensors (*i.e.*, featuring a single simultaneous scanline acquisition) such as the one
 135 used in (Paparoditis et al., 2012) generally send an almost constant number H of pulses
 per scanline (or per turn for 360 degree 2D LiDARs) where each pulse was emitted at
 a certain θ angle value. Therefore, any measurement of the sensor might be organized
 in an image of size $W \times H$, where W is the number of consecutive scanlines and thus a
 temporal dimension. This is illustrated in Figure 3 in which one can see how the 2D image
 140 is spanned by the sensor topology. In this work, such images are only built using the range
 measurement as pixel intensity, later referred to as range images. Note that these range
 images differ from typical range images (Kinect, RGB-D) as the origin of acquisition is not
 the same for each pixel and the 3D directions of pixels are not regularly spaced along the
 image, but warped by the orientation changes of the sensor trajectory.

145 3D LiDAR sensors are based on multiple simultaneous scanline acquisitions (*e.g.* $H =$
 64 fibers) such as in the MMS proposed in (Geiger et al., 2013). Again, each scanline
 contains the same number of points and each scanline may be stacked horizontally to
 form the same type of structure, as illustrated in Figure 4. Note that Figures 3 and 4
 are simplified for better understanding, but that realistic cases can be more chaotic as
 150 discussed later in this section.

Whereas LiDAR pulses are emitted somewhat regularly, many pulses yield no range
 measurements due, for instance, to reflective surfaces, absorption or absence of target

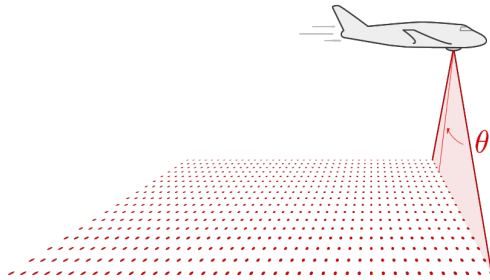


Figure 2: Example of the intrinsic topology of a 2D LiDAR sensor built on a plane

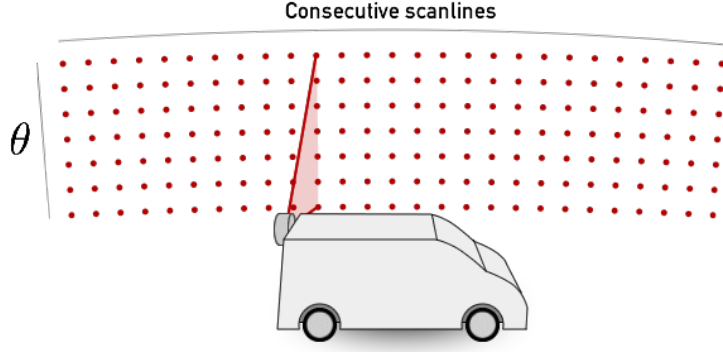


Figure 3: Example of 2D LiDAR sensor and the related topology

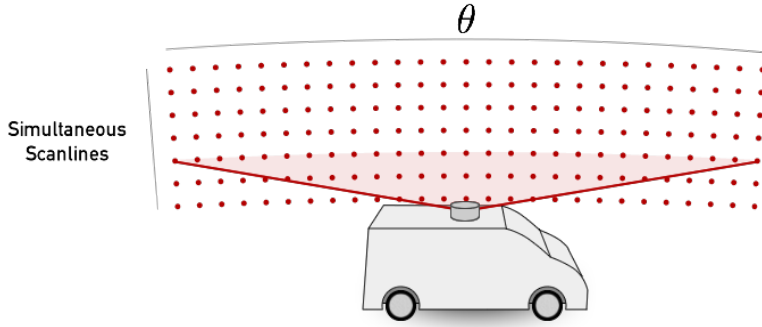


Figure 4: Example of 3D LiDAR sensor and the related topology

objects (*e.g.* in the sky direction) or an ignored measurement whenever the measure is too uncertain. Therefore the sensor topology is only a relevant approximation for emitted pulses but not for echo returns, such that the range image is sparse with undefined values where the sensor measured no echoes (or when further processing was performed on the acquisition, leading to the removal of points having a too incertain measurement). This is illustrated in Figure 5.b in which pulses with no echoes appear in dark. Note that considering multi-echo datasets as a multilayer depth image is beyond the scope of this paper, which only considers first returns.

This 2D sensor topology encodes an implicit neighborhood between LiDAR measurement pulses. Whereas the implicit topology of pixels in optical images is supported by a regular geometry of rays (shared origin and regular grid of directions if geometric distortion is neglected), the proposed 2D sensor topology for LiDAR point clouds is supported by the trajectory-warped geometry of 3D rays. However, it readily provides, with minimal effort, an approximation of the immediate 3D point neighborhoods, especially if the sensor moves or turns slowly compared to its sensing rate. We argue however that this approximation

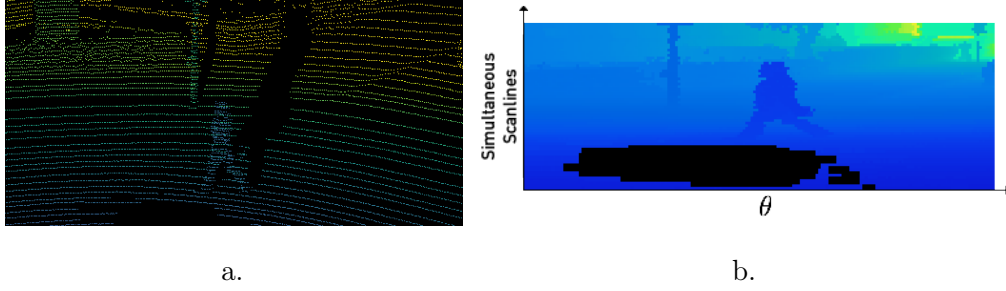


Figure 5: Example of a point cloud from the KITTI database (Geiger et al., 2013) (a) turned into a range image (b). Note that the dark area in (b) corresponds to pulses with no returns.

is sufficient for most purposes, as it has the added advantage of providing pulse neighborhoods that are reasonably local both in terms of space and time, thus being robust to misregistrations, and being very efficient to handle (constant time access to neighbors). Moreover, as LiDAR sensor designs evolve to higher sampling rates within and/or across scanlines, the sensor topology will better approximate spatio-temporal neighborhoods, even in the case of mobile acquisitions.

We argue that most raw LiDAR datasets contain all the information (scanline ordering, pulses with no echo, number of points per turn...) to enable the access to a well-defined implicit sensor topology. However it sometimes occurs that the dataset received further processings (points were reordered or filtered, or pulses with no return were discarded) or that the sensor does not acquire neighbouring points consecutively. Therefore, the sensor topology may then only be approximated using auxilliary point attributes (time, θ , fiber id...) and guesses about acquisition settings (*e.g.* guessing approximate Δtime or $\Delta\theta$ values between successive pulse emissions). Using this information, one can recreate the range map by stacking points even if some points were discarded. Defining a grid-like topology is a good approximation if the number of pulses per scanline/per turn is close to an integer constant with relatively stable rotation offsets between pulses.

3.3. Interest and applications

The use of range images as the simplified representation of a point cloud directly brings spatial structure to the point cloud. Therefore, retrieving neighbors of a point, which was formerly done using advanced data structures (Muja and Lowe, 2014), is now a trivial operation and is given without any ambiguities. This was proved to be very useful in applications such as remeshing since faces can be directly associated to the grid structure of the range image. As shown in this paper, considering a point cloud as a range image

supported by its implicit sensor topology enables the adaptation of the many existing image processing approaches to LiDAR point cloud processing (*e.g.*: segmentation, disocclusion). Moreover, when optical data was acquired along with LiDAR point clouds, the range image
 195 can be used for improving the point cloud colorization and the texture registration on the point cloud as the silhouettes present in the range image are likely to be aligned with the gradients of optical images.

In the following sections, the LiDAR measurements, equipped with this implicit 2D topology, are denoted as the sparse range image u^R .

200 4. Application to point cloud segmentation

In this section, a simple yet efficient segmentation technique that takes advantage of the range image will be introduced. Results will be presented and a quantitative analysis will be performed to validate the model.

4.1. Range Histogram Segmentation technique

205 We now propose a segmentation technique based on range histograms. For the sake of simplicity, we assume that the ground is relatively flat and we remove ground points, which are identified by plane fitting.

Instead of segmenting the whole range image u^R directly, we first split this image into S sub-windows $u_s^R, s = 1 \dots S$ of size $W_s \times H$ along the horizontal axis to prevent
 210 each sub-window from representing several objects at the same range. For each u_s^R , a depth histogram h_s of B bins is built. This histogram is automatically segmented into C_s classes using the a-contrario technique presented in Delon et al. (2007). This technique presents the advantage of segmenting a 1D-histogram without any prior assumption, *e.g.* the underlying density function or the number of objects. Moreover, it aims at segmenting
 215 the histogram following an accurate definition of an admissible segmentation, preventing over- and under-segmentation. An example of a segmented histogram is given in Figure 6.

Once the histograms of successive sub-images have been segmented, we merge together the corresponding classes by checking the distance between each of their centroids in order to obtain the final segmentation labels. Let us define the centroid \mathcal{C}_s^i of the i^{th} class C_s^i in the histogram h_s of the sub-image u_s^R as follows:

$$\mathcal{C}_s^i = \frac{\sum_{b \in C_s^i} b \times h_s(b)}{\sum_{b \in C_s^i} h_s(b)} \quad (1)$$

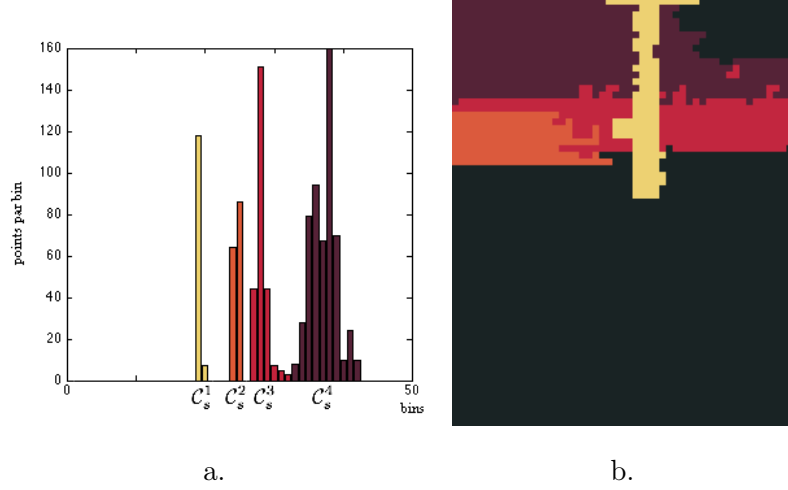


Figure 6: Result of the histogram segmentation using the approach of Delon et al. (2007). (a) segmented histogram (bins of 50cm), (b) result in the range image using the same colors. We can see how well the segmentation follows the different modes of the histogram.

where b are all bins belonging to class C_s^i . The distance between two classes C_s^i and C_r^j of two consecutive windows r and s can be defined as follows:

$$d(C_s^i, C_r^j) = |C_s^i - C_r^j| \quad (2)$$

Finally, we can set a threshold such that if $d(C_s^i, C_r^j) \leq \tau$, classes C_s^i and C_r^j should be merged (*e.g.* they now share the same label). If two classes of the same window are eligible to be merged with the class of an other window, then only the one with lower depth should be merged. Results of this segmentation procedure can be found in the next subsection. The choice of W_s , B and τ mostly depends on the type of data that is being treated (sparse or dense). For sparse point clouds (few thousand points per turn), B has to remain small (*e.g.* 50) whereas for dense point clouds ($> 10^5$ points per turn), this value can be increased (*e.g.* 200). In practice, we found out that good segmentations may be obtained on various kinds of data by setting $W_s = 0.5 \times B$ and $\tau = 0.2 \times B$. Note that the windows are not required to be overlapping in most cases, but for very sparse point clouds, an overlap of 10% is enough to achieve good segmentation. For example in our experiments on the KITTI dataset (Geiger et al., 2013), for range images of size 2215×64 px, $W_s = 50$, $B = 100$, $\tau = 20$ with no overlap.

4.2. Results & Analysis

Figure 7 shows two examples of segmentations obtained using our method on different point clouds from the KITTI dataset (Geiger et al., 2013). Each object, of different scale,

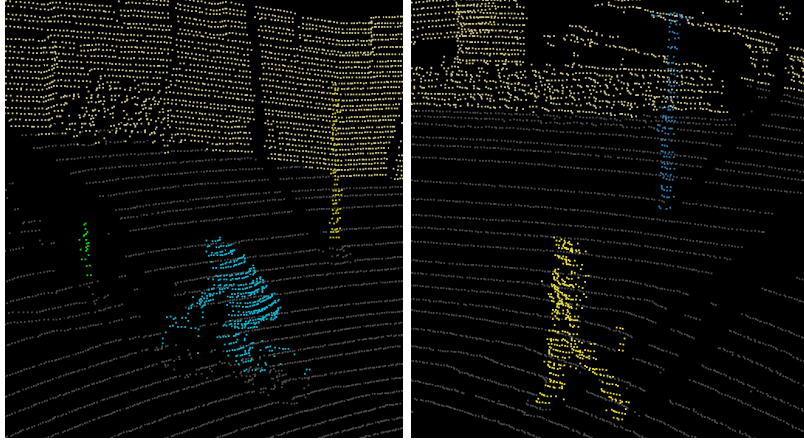


Figure 7: Example of point cloud segmentation using our model on various scenes. We can note how each label strictly corresponds to a single object (pedestrian, poles, walls).

is correctly distinguished from all others as an individual entity. Moreover, both results appear to be visually plausible.

235 Apart from the visual inspection, we also performed a quantitative analysis on the IQmulus dataset (Vallet et al., 2015). The IQmulus dataset consists of a manually annotated point cloud of 12 million points in which points are clustered into several classes corresponding to typical urban entities (cars, walls, pedestrian, etc.). Our aim is to compare the quality of our segmentation on several objects to the ground truth provided by this

240 dataset. First, the point cloud is segmented using our technique, using 100px wide windows with a 10px overlap and a threshold for merging set to 50. After that, we manually select labels that correspond to the wanted object (hereafter: cars). We then compare the result of the segmentation to the ground truth in the same area, and compute the Jac-

card distance (Intersection over Union) between our result and the ground truth. Figure 8

245 presents the result of such a comparison. The overall distance shows that the segmentation matches 97.09% of the ground truth, for a total of 59021 points. Although the result is very satisfying, our result differs in some ways from the ground truth. Indeed, in the first zoom of Figure 8, one can see that our model better succeeds in catching the points of the cars that are close to the ground (we remind here that the ground truth on IQmulus

250 was manually labelled and thus subject to errors). In the second zoomed-in part, one can see that points belonging to the windows of the car were not correctly retrieved using our model. This is due to the fact that the measure in areas where the beam was highly deviated (*e.g.* beams that were not reflected in the same direction as the one they were emitted along) is not reliable as the range estimation is not realistic. Therefore our model

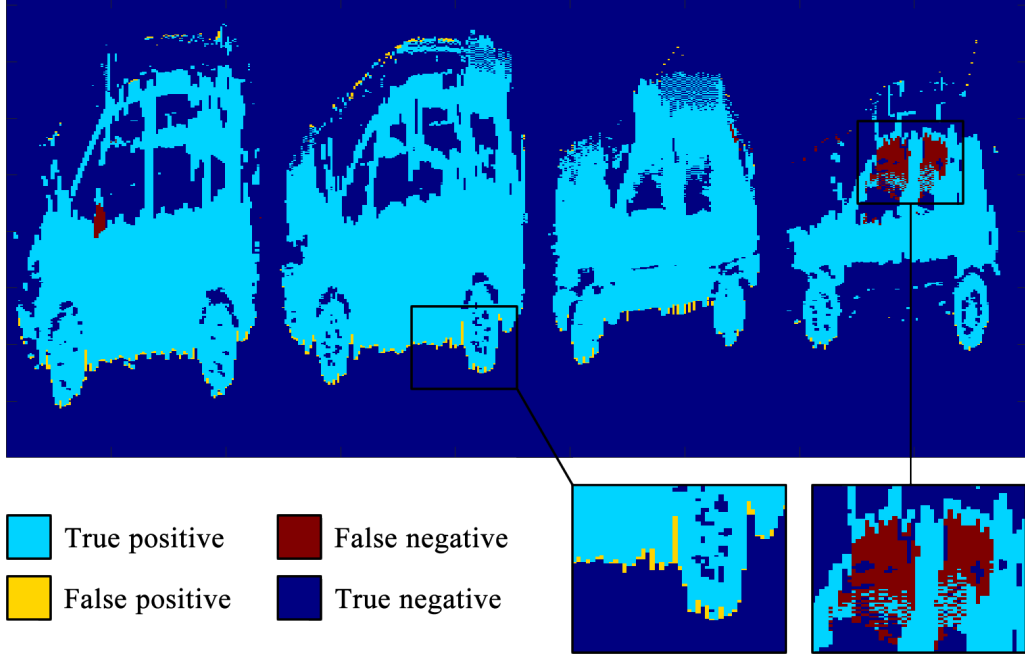


Figure 8: Quantitative analysis of the segmentation of cars. Our segmentation result only slightly differs from the ground truth in areas close to the ground or for points that were largely deviated such as points through windows.

255 fails in areas where the estimated 3D point is not close to the actual 3D surface. Note that a similar case appears for the review mirror (Figure 8, on the left) which is made of specular material that leads to bad measurements.

In some extreme cases, the segmentation is not able to separate objects that are too close from the sensor point of view. Figure 9.a shows a result of the segmentation in a scene
 260 where two cars that are segmented with the same label (symbolised by the same color). In order to better distinguish the different objects, one can simply compute the connected components of the points regarding their 3D neighborhood (that can be computed using K-NN for example). Figure 9.b shows the result of such post-processing on the same two cars. We can notice how both cars are distinguished from one other.

265 5. Application to disocclusion

In this section, we show that the problem of disocclusion in a 3D point cloud can be addressed using basic image inpainting techniques.

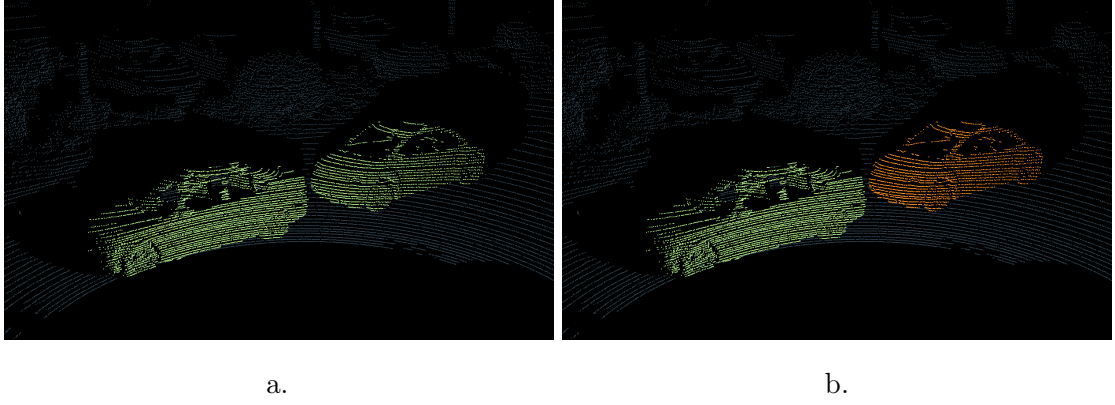


Figure 9: Result of the segmentation of a point cloud where two objects end up with the same label (a), and the labeling after considering the connected components (b).

5.1. Range map disocclusion technique

The segmentation technique introduced above provides labels that can be manually selected in order to build masks. As mentioned in the beginning, we propose a variational approach to the problem of disocclusion of the point cloud that leverages its range image representation. By considering the range image representation of the point cloud rather than the point cloud itself, the problem of disocclusion can be reduced to the estimation of a set of 1D ranges instead of a set of 3D points, where each range is associated with the ray direction of the pulse. The Gaussian diffusion algorithm provides a very simple algorithm for the disocclusion of objects in 2D images by solving partial differential equations. This technique is defined as follows:

$$\begin{cases} \frac{\partial u}{\partial t} - \Delta u = 0 \text{ in } (0, T) \times \Omega \\ u(t = 0, x, y) = u^R(x, y) \text{ in } \Omega \end{cases} \quad (3)$$

having u an image defined on Ω , t being a time range and Δ the Laplacian operator. As the
270 diffusion is performed in every direction, the result of this algorithm is often very smooth. Therefore, the result in 3D lacks of coherence as shown in Figure 10.b.

In this work, we show that the structures that require disocclusion are likely to evolve smoothly along the x_W and y_W axes of the real world as defined in Figure 11.a. Therefore, we set $\vec{\eta}$ for each pixel to be a unitary vector orthogonal to the projection of z_W in the
275 u^R range image (Figure 11.b). This vector defines the direction in which the diffusion should be done to respect this prior. Note that most MLS systems provide georeferenced coordinates of each point that can be used to define $\vec{\eta}$. For example, using a 2D LiDAR sensor that is orthogonal to the path of the vehicle, one can define $\vec{\eta}$ as the projection of the pitch angle of the acquisition vehicle.

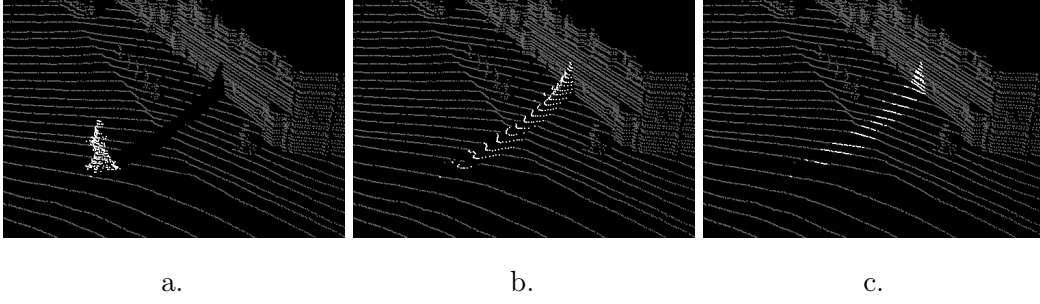


Figure 10: Comparison between disocclusion algorithms. (a) is the original point cloud (white points belong to the object to be disoccluded), (b) the result after Gaussian diffusion and (c) the result with our proposed algorithm (1500 iterations). Note that the Gaussian diffusion oversmooths the background of the object whereas our proposed model respects the coherence of the scene.

We aim at extending the level lines of u along $\vec{\eta}$. This can be expressed as $\langle \nabla u, \vec{\eta} \rangle = 0$. Therefore, we define the energy $F(u) = \frac{1}{2}(\langle \nabla u, \vec{\eta} \rangle)^2$. The disocclusion is then computed as a solution of the minimization problem $\inf_u F(u)$. The gradient of this energy is given by $\nabla F(u) = -\langle (\nabla^2 u) \vec{\eta}, \vec{\eta} \rangle = -u_{\vec{\eta}\vec{\eta}}$, where $u_{\vec{\eta}\vec{\eta}}$ stands for the second order derivative of u with respect to $\vec{\eta}$ and $\nabla^2 u$ for the Hessian matrix. The minimization of F can be done by gradient descent. If we cast it into a continuous framework, we end up with the following equation to solve our disocclusion problem:

$$\begin{cases} \frac{\partial u}{\partial t} - u_{\vec{\eta}\vec{\eta}} = 0 \text{ in } (0, T) \times \Omega \\ u(t = 0, x, y) = u^R(x, y) \text{ in } \Omega \end{cases} \quad (4)$$

280 using the notations introduced earlier. We recall that $\Delta u = u_{\vec{\eta}\vec{\eta}} + u_{\vec{\eta}^T \vec{\eta}^T}$, where $\vec{\eta}^T$ stands for a unitary vector orthogonal to $\vec{\eta}$. Thus, Equation (4) can be seen as an adaptation of the Gaussian diffusion equation (3) with respect to the diffusion prior in the direction $\vec{\eta}$.

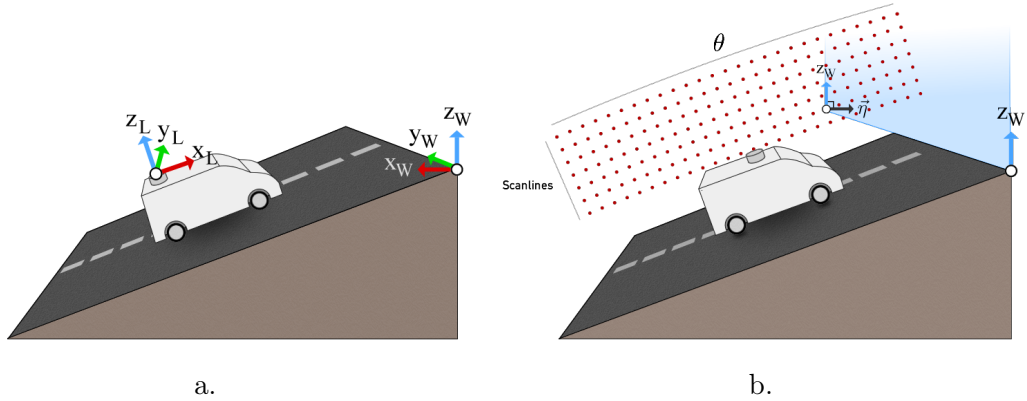


Figure 11: (a) is the definition of the different frames between the LiDAR sensor (x_L, y_L, z_L) and the real world (x_W, y_W, z_W) , (b) is the definition and the visualization of $\vec{\eta}$.

Figure 10 shows a comparison between the original Gaussian diffusion algorithm and our modification. The Gaussian diffusion leads to an over-smoothing of the scene, creating an aberrant surface, whereas our modification provides a result that is more plausible.

The equation proposed in (4) can be solved iteratively. The number of iterations simply depends on the size of the area that needs to be filled in.

5.2. Results & Analysis

In this part, the results of the segmentation of various objects and the disocclusion of their background are detailed.

Sparse point cloud. A first result is shown in Figure 12. This result is obtained for a sparse point cloud ($\approx 10^5$ pts) of the KITTI database. A pedestrian is segmented out of the scene using our proposed segmentation technique (using the parameters introduced in 4.1) and a manual selection of the corresponding label. This is used as a mask for the disocclusion of its background using our modified variational technique for disocclusion. Figure 12.a shows the original range image. In Figure 12.b, the dark region corresponds to the result of the segmentation step for the pedestrian. For practical purpose, a very small dilation is applied to the mask (radius of 2px in sensor topology) to ensure that no outlier points (near the occluder’s silhouette with low accuracy or on the occluder itself) bias the reconstruction. Finally, Figure 12.c shows the range image after the reconstruction. We can see that the disocclusion performs very well as the pedestrian has completely disappeared and the result is visually plausible in the range image. Notice how the implicit sensor topology of the range image has allowed here to use a standard 2D image processing technique from mathematical morphology to filter mislabelled and inaccurate points near silhouettes.

In this scene, $\vec{\eta}$ has a direction that is very close to the x axis of the range image and the 3D point cloud is acquired using a 3D LiDAR sensor. Therefore, the coherence of the reconstruction can be checked by looking how the acquisition lines are connected. Figure 13 shows the reconstruction of the same scene in three dimensions. This reconstruction simply consists in the projection of the depth of each pixel along the axis formed by each corresponding point and the sensor origin. We can see that the acquisition lines are properly retrieved after removing the pedestrian. This result was generated in 4.9 seconds using Matlab on a 2.7GHz processor. Note that a similar analysis can be done on the results presented in Figure 1.

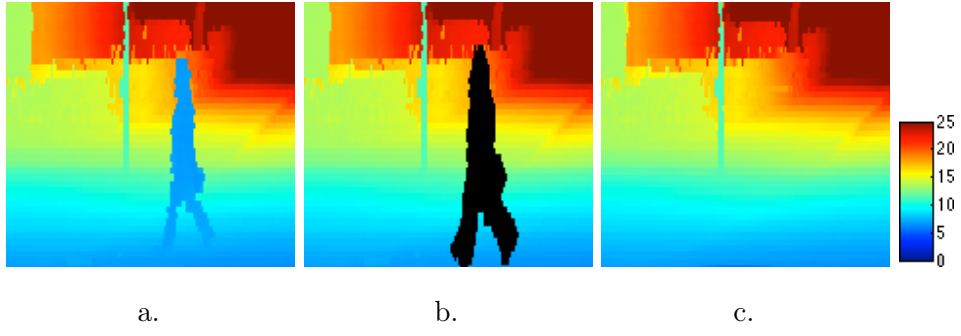
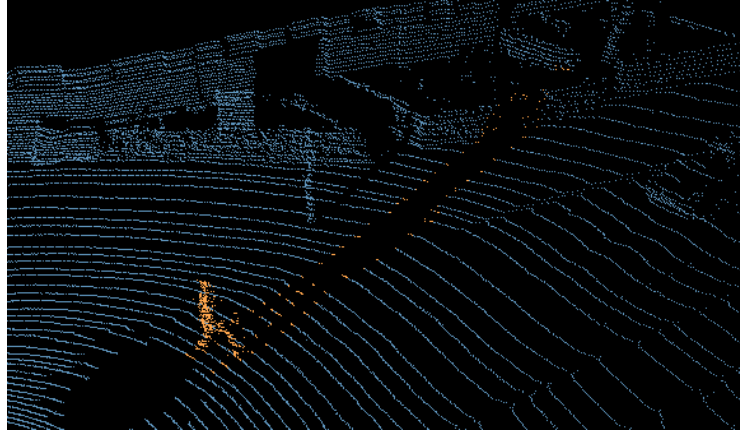
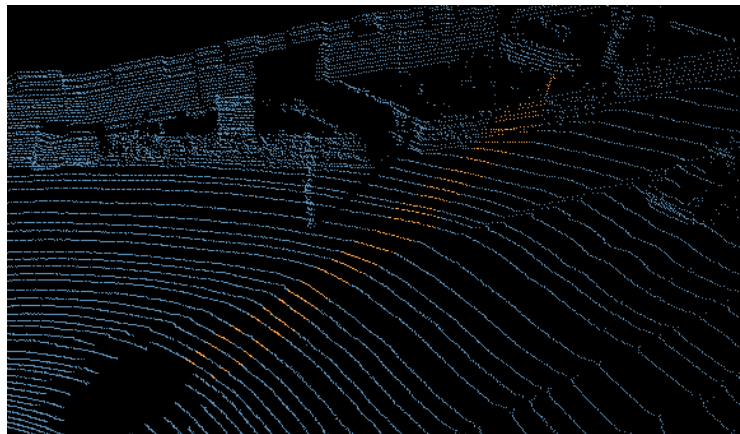


Figure 12: Result of disocclusion on a pedestrian on the KITTI database (Geiger et al., 2013). (a) is the original range image, (b) the segmented pedestrian (dark), (c) the final disocclusion. Depth scale is given in meters. After disocclusion, the pedestrian completely disappears from the image, and its background is reconstructed accordingly to the rest of the scene.



a.



b.

Figure 13: 3D representation of the disocclusion of the pedestrian presented in Figure 12. (a) is the original mask highlighted in 3D, (b) is the final reconstruction.



a.



b.



c.

Figure 14: Result of the disocclusion on a car in a dense point cloud. (a) is the original point cloud colorized with the reflectance, (b) is the segmentation of the car highlighted in orange, (c) is the result of the disocclusion. The car is entirely removed and the road is correctly reconstructed.

Dense point cloud. In this work, we aim at presenting a model that performs well on both
 315 sparse and dense data. Figure 14 shows a result of the disocclusion of a car in a dense
 point cloud. This point cloud was acquired using the Stereopolis-II system (Paparoditis
 et al., 2012) and contains over 4.9 million points. In Figure 14.a, the original point cloud is
 displayed with the color based on the reflectance of the points for a better understanding
 of the scene. Figure 14.b highlights the segmentation of the car using our model (with
 320 the same parameters as in Section 4.2), dilated to prevent aberrant points. Finally, Figure
 14.c depicts the result of the disocclusion of the car using our method.

We can note that the car is perfectly removed from the scene. It is replaced by the
 ground that could not have been measured during the acquisition. Although the recon-
 struction is satisfying, some gaps are left in the point cloud. Indeed, in the data used for
 325 this example, pulses returned with large deviation values were discarded. Therefore, the
 windows and the roof of the car are not present in the point cloud before and after the
 reconstruction as no data is available. We could have added these no-return pulses in the
 inpainting mask as well to reconstruct these holes as well.

Quantitative analysis. To conclude this section, we perform a quantitative analysis of our
 330 disocclusion model on the KITTI dataset. The experiment consists in removing areas of

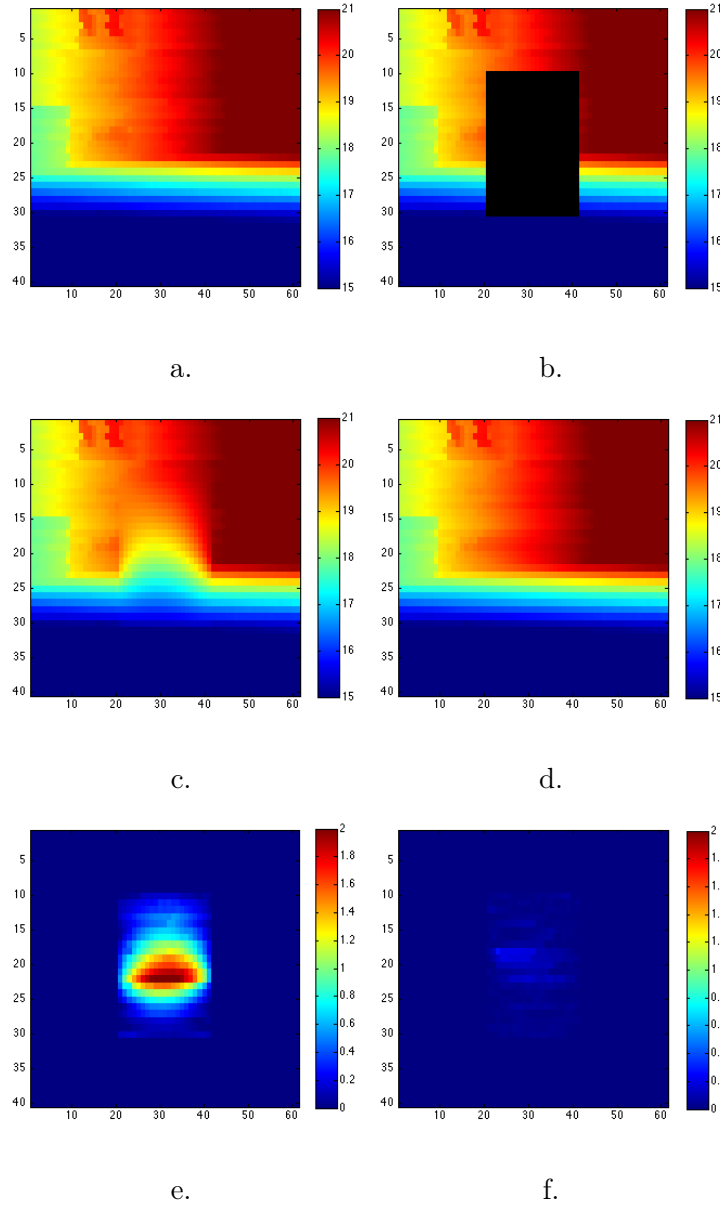


Figure 15: Example of results obtained for the quantitative experiment. (a) is the original point cloud (ground truth), (b) the artificial occlusion in dark, (c) the disocclusion result with the Gaussian diffusion, (d) the disocclusion using our method, (e) the Absolute Difference of the ground truth against the Gaussian diffusion, (f) the Absolute Difference of the ground truth against our method. Scales are given in meters.

various point clouds in order to reconstruct them using our model. Therefore, the original point clouds can serve as ground truth. Note that areas are removed while taking care that no objects are present in those locations. Indeed, this test aims at showing how the disocclusion step behaves when reconstructing backgrounds of objects. The size of the removed areas corresponds to an approximation of a pedestrian's size at 8 meters from the sensor in the range image ($20 \times 20\text{px}$).

The test was done on 20 point clouds in which an area was manually removed and then reconstructed. After that, we computed the MAE (Mean Absolute Error) between the ground truth and the reconstruction (where the occlusion was simulated) using both Gaussian disocclusion and our model. We recall that the MAE is expressed as follows:

$$\text{MAE}(u_1, u_2) = \frac{1}{N} \sum_{i,j \in \Omega} |u_1(i, j) - u_2(i, j)| \quad (5)$$

where u_1, u_2 are images defined on Ω with N pixels where each pixel intensity represents the depth value. Table 1 sums up the result of our experiment. We can note that our method provides a great improvement compared to the Gaussian disocclusion, with an
340 average MAE lower than 3cm. These results are obtained on scenes where objects are located from 12 to 25 meters away from the sensor. The result obtained using our method is very close to the sensor accuracy as mentionned by the manufacturer ($\simeq 2\text{cm}$).

Figure 15 shows an example of disocclusion following this protocole. The result of our proposed model is visually very plausible whereas the Gaussian diffusion ends up
345 oversmoothing the reconstructed range image which increases the MAE.

Overlapping objects. Although the proposed disocclusion method performs well in realistic scenarios as demonstrated above, in some specific contexts, the reconstruction quality can be debatable. Indeed, when two small objects (pedestrians, poles, cars, etc.) overlap in front of the 3D sensor (*e.g.* one object is in front of the other), the disocclusion of the
350 closest object may not fully recover the farthest object. Figure 16.a shows an example of such a scenario where the goal is to remove the cyclist (highlighted in green). In this case, a pole (Figure 16.a, in orange) is situated between the cyclist and the background. Figure 16.b presents the disocclusion of the cyclist. The background is reconstructed in a plausible way, however, details of the occluded part of the pole are not recovered.

Table 1: Comparison of the average MAE (Mean Absolute Error) on the reconstruction of occluded areas.

	Gaussian	Proposed model
Average MAE (meters)	0.591	0.0279
Standard deviation of MAEs	0.143	0.0232

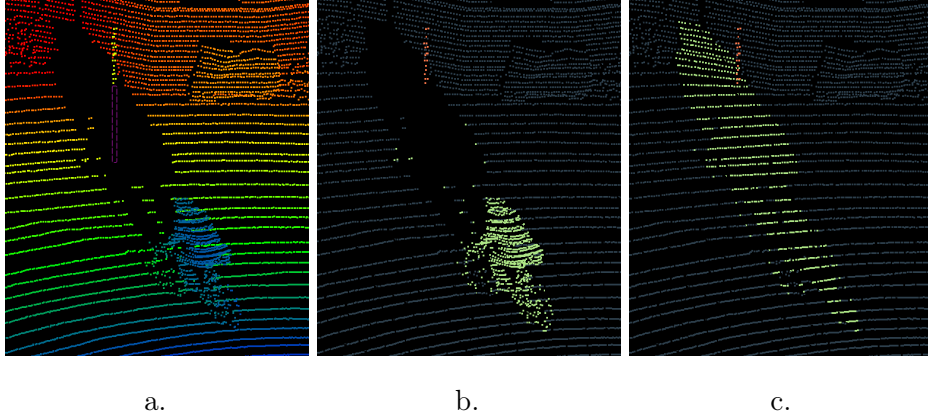


Figure 16: Example of a scene where two objects overlap in the acquisition. (a) is the original point cloud colored with depth towards sensor with the missing part of a pole highlighted with dashed pink contour, (b) shows the two objects that overlap: a pole (highlighted in orange) and a cyclist (highlighted in green), (c) shows the disocclusion of the cyclist. Although the background is reconstructed in a plausible way, details of the occluded part of the pole are missing.

355 6. Conclusion

In this paper, we have proposed a novel methodology for LiDAR point cloud processing that relies on the implicit topology that is brought by most recent LiDAR sensors. Considering the range image derived from the sensor topology has enabled a simplified formulation of the problem from having to determine an unknown number of 3D points to estimating only the 1D range in the ray directions of a fixed set of range image pixels.
 360 Beyond simplifying drastically the search space, it also provides directly a reasonable sampling pattern for the reconstructed point set. Moreover, it also directly provides a robust estimation of the neighborhood of each point according to the acquisition, while improving the computational time and the memory usage.

365 To highlight the relevance of this methodology, we have proposed novel approaches for the segmentation and the disocclusion of objects in 3D point clouds acquired using MMS. These models take advantage of range images. We have also proposed an improvement of a classical imaging technique that takes the nature of the point cloud into account (horizontality prior on the 3D embedding), leading to better results. The segmentation step can
 370 be done online any time a new window is acquired, leading to great speed improvement, constant memory requirements and the possibility of online processing during the acquisition. Moreover, our model is designed to work semi-automatically with using very few parameters in reasonable computational time. We have validated both the segmentation and the disocclusion methods by visual inspection as well as quantitative analysis against

375 ground truth and we have proved their effectiveness in terms of accuracy.

In the future, we will focus on extending the methodology to other point cloud processing tasks such as LiDAR point cloud colorization / registration using range images and optical images through variational models.

7. Acknowledgement

380 J-F. Aujol is a member of Institut Universitaire de France. This work was funded by the ANR GOTMI (ANR-16-CE33-0010-01) grant. We would like to thank the anonymous reviewer for his/her useful comments.

References

- Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (11), 2274–2282.
- 385
- Becker, J., Stewart, C., Radke, R. J., 2009. LiDAR inpainting from a single image. In: *IEEE International Conference on Computer Vision*. Vol. 1. pp. 1441–1448.
- Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C., 2000. Image inpainting. In: *ACM Computer graphics and interactive techniques*. Vol. 1. pp. 417–424.
- 390
- Bevilacqua, M., Aujol, J.-F., Biasutti, P., Brédif, M., Bugeau, A., 2017. Joint inpainting of depth and reflectance with visibility estimation. *ISPRS Journal of Photogrammetry and Remote Sensing* 125, 16–32.
- Biasutti, P., Aujol, J.-F., Brédif, M., Bugeau, A., 2017. Disocclusion of 3D LiDAR point clouds using range images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 4 (1), 75–82.
- 395
- Bredies, K., Kunisch, K., Pock, T., 2010. Total generalized variation. *SIAM Journal on Mathematical Analysis* 3, 492–526.
- Buyssens, P., Daisy, M., Tschumperlé, D., Lézoray, O., 2015a. Depth-aware patch-based image disocclusion for virtual view synthesis. In: *Proceedings of SIGGRAPH Asia Technical Briefs*. Vol. 34. pp. 2–6.
- 400
- Buyssens, P., Daisy, M., Tschumperlé, D., Lézoray, O., 2015b. Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions. *IEEE Transactions on Image Processing* 24 (6), 1809–1824.

- 405 Chambolle, A., Pock, T., 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Visison* 40 (1), 120–145.
- Criminisi, A., Pérez, P., Toyama, K., 2004. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing* 13 (9), 1200–1212.
- Delon, J., Desolneux, A., Lisani, J.-L., Petro, A. B., 2007. A nonparametric approach for histogram
410 segmentation. *IEEE Transactions on Image Processing* 16 (1), 253–261.
- Demantke, J., Mallet, C., David, N., Vallet, B., 2011. Dimensionality based scale selection in 3D LiDAR point clouds. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 38 (5), 97–102.
- Doria, D., Radke, R., 2012. Filling large holes in LiDAR data by inpainting depth gradients. In:
415 *International Conference on Pattern Recognition*. Vol. 1. pp. 65–72.
- El-Halawany, S., Moussa, A., Lichti, D., El-Sheimy, N., 2011. Detection of road curb from mobile terrestrial laser scanner point cloud. In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. 38. pp. 29–31.
- Ferstl, D., Reinbacher, C., Ranftl, R., Rütther, M., Bischof, H., 2013. Image guided depth up-
420 sampling using anisotropic total generalized variation. In: *IEEE International Conference on Computer Vision*. Vol. 1. pp. 993–1000.
- Gehring, J., Hebel, M., Arens, M., Stilla, U., 2017. An approach to extract moving objects from MLS data using a volumetric background representation. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 4, 107–114.
- 425 Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research* 32 (11), 1231–1237.
- Goulette, F., Nashashibi, F., Abuhadrous, I., Ammoun, S., Lourceau, C., 2006. An integrated on-board laser range sensing system for on-the-way city and road modelling. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 34 (A).
- 430 Hervieu, A., Soheilian, B., 2013. Semi-automatic road/pavement modeling using mobile laser scanning. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. 2.
- Hervieu, A., Soheilian, B., Brédif, M., 2015. Road marking extraction using a Model&Data-driven RJ-MCMC. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information
435 Sciences* 2 (3), 47–48.

- Huang, J., Menq, C.-H., 2001. Automatic data segmentation for geometric feature extraction from unorganized 3D coordinate points. *IEEE Trans. on Robotics and Automation* 17 (3), 268–279.
- Landrieu, L., Mallet, C., Weinmann, M., 2017. Comparison of belief propagation and graph-cut approaches for contextual classification of 3D LiDAR point cloud data. In: *IEEE International Geoscience and Remote Sensing Symposium*. Vol. 1. pp. 2768–2771.
- Lorenzi, L., Melgani, F., Mercier, G., 2011. Inpainting strategies for reconstruction of missing data in VHR images. *IEEE Geoscience and Remote Sensing Letters* 8 (5), 914–918.
- Muja, M., Lowe, D. G., 2014. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36 (11), 2227–2240.
- Paparoditis, N., Papelard, J.-P., Cannelle, B., Devaux, A., Soheilian, B., David, N., Houzay, E., 2012. 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* 200, 69–79.
- Papon, J., Abramov, A., Schoeler, M., Worgotter, F., 2013. Voxel cloud connectivity segmentation-supervoxels for point clouds. In: *IEEE Conference on Computer Vision and Pattern Recognition*. Vol. 1. pp. 2027–2034.
- Park, S., Guo, X., Shin, H., Qin, H., 2005. Shape and appearance repair for incomplete point surfaces. In: *IEEE International Conference on Computer Vision*. Vol. 2. pp. 1260–1267.
- Rabbani, T., Van Den Heuvel, F., Vosselmann, G., 2006. Segmentation of point clouds using smoothness constraint. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 36 (5), 248–253.
- Schnabel, R., Wahl, R., Klein, R., 2007. RANSAC based out-of-core point-cloud shape detection for city-modeling. *Proceedings of "Terrestrisches Laserscanning"* 26, 214–226.
- Serna, A., Marcotegui, B., 2013. Urban accessibility diagnosis from mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing* 84, 23–32.
- Serna, A., Marcotegui, B., 2014. Detection, segmentation and classification of 3D urban objects using mathematical morphology and supervised learning. *ISPRS Journal of Photogrammetry and Remote Sensing* 93, 243–255.
- Sharf, A., Alexa, M., Cohen-Or, D., 2004. Context-based surface completion. *ACM Transactions on Graphics* 23 (3), 878–887.
- Vallet, B., Brédif, M., Serna, A., Marcotegui, B., Paparoditis, N., 2015. TerraMobilita/IQmulus urban point cloud analysis benchmark. *Computers and Graphics* 49, 126–133.

Weickert, J., 1998. Anisotropic diffusion in image processing. Vol. 1. Teubner Stuttgart.

470 Weinmann, M., Jutzi, B., Hinz, S., Mallet, C., 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. ISPRS Journal of Photogrammetry and Remote Sensing 105, 286–304.

Zhu, X., Zhao, H., Liu, Y., Zhao, Y., Zha, H., 2010. Segmentation and classification of range image from an intelligent vehicle in urban environment. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. Vol. 1. pp. 1457–1462.