Bayesian 3D Reconstruction of Complex Scenes from Single-Photon Lidar Data.
— Source link ↗

Julián Tachella, Yoann Altmann, Ximing Ren, Aongus McCarthy ...+3 more authors

Related papers:

- Single-photon three-dimensional imaging at up to 10 kilometers range.

- Three-dimensional single-photon imaging through obscurants

- A Few Photons Among Many: Unmixing Signal and Noise for Photon-Efficient Active Imaging

- Real-time 3D reconstruction from single-photon lidar data using plug-and-play point cloud denoisers.

- Lidar Waveform-Based Analysis of Depth Images Constructed Using Sparse Single-Photon Data

This is a Publisher's version published in: **http://oatao.univ-toulouse.fr/24125**

To cite this version:

Tachella, Julian and Altmann, Yoann and Ren, Ximin and McCarthy, Aongus and Buller, Gerald S. and Mclaughlin, Stephen and Tourneret, Jean-Yves *Bayesian 3D Reconstruction of Complex Scenes from Single-Photon Lidar Data.* (2019) SIAM Journal on Imaging Sciences, 12 (1). 521-550. ISSN 1936-4954

Bayesian 3D Reconstruction of Complex Scenes from Single-Photon Lidar Data[*]

Julián Tachella[†], Yoann Altmann[†], Ximing Ren[†], Aongus McCarthy[†], Gerald S. Buller[†],
Stephen McLaughlin[†], and Jean-Yves Tourneret[‡]

Abstract. Light detection and ranging (Lidar) data can be used to capture the depth and intensity profile of a 3D scene. This modality relies on constructing, for each pixel, a histogram of time delays between emitted light pulses and detected photon arrivals. In a general setting, more than one surface can be observed in a single pixel. The problem of estimating the number of surfaces, their reflectivity, and position becomes very challenging in the low-photon regime (which equates to short acquisition times) or relatively high background levels (i.e., strong ambient illumination). This paper presents a new approach to 3D reconstruction using single-photon, single-wavelength Lidar data, which is capable of identifying multiple surfaces in each pixel. Adopting a Bayesian approach, the 3D structure to be recovered is modelled as a marked point process, and reversible jump Markov chain Monte Carlo (RJ-MCMC) moves are proposed to sample the posterior distribution of interest. In order to promote spatial correlation between points belonging to the same surface, we propose a prior that combines an area interaction process and a Strauss process. New RJ-MCMC dilation and erosion updates are presented to achieve an efficient exploration of the configuration space. To further reduce the computational load, we adopt a multiresolution approach, processing the data from a coarse to the finest scale. The experiments performed with synthetic and real data show that the algorithm obtains better reconstructions than other recently published optimization algorithms for lower execution times.

Key words. Bayesian inference, 3D reconstruction, Lidar, low-photon imaging, Poisson noise

AMS subject classifications. 62F15, 62H12, 62H35, 62P30, 62P12, 65C40

DOI. 10.1137/18M1183972

1. Introduction. Reconstruction and analysis of 3D scenes have a variety of applications, spanning earth monitoring [16, 35, 28], underwater imaging [27, 18], automotive [36, 41], and defense [14]. Single-photon light detection and ranging devices acquire range measurements by illuminating a 3D scene with a train of laser pulses and recording the time-of-flight (TOF)

of the photons reflected from the objects in the illuminated scene. Using a time correlated single-photon counting (TCSPC) system, a histogram of time delays between emitted and reflected pulses is constructed for each pixel. For a given pixel, the presence of an object is associated with a characteristic distribution of photon counts in the histogram. The position and number of counts provide depth and reflectivity information, respectively. In scenarios where the light goes through a semitransparent material (e.g., windows or camouflage) or when the laser beam is wide enough with respect to the object size (e.g., distant objects), it is possible to record two or more surfaces in a single pixel. The recovery of multiple objects per pixel is thus very important in many applications, such as tree layer analysis [48] or detection of hidden targets behind camouflage [19].

In order to reconstruct the 3D scene from single-photon Lidar data, it is necessary to discriminate the photon counts associated with each surface from the ones linked to the background illumination. When the background level can be neglected, the traditional approach consists, under the single-peak assumption, of log-match filtering the Lidar waveforms and finding the maximum of the filtered data for each pixel [44], which is the maximum likelihood (ML) solution for a Poisson noise assumption (a matched filter is used for Gaussian noise). While this method obtains good results for high photon counts, it gives poor estimates when the background illumination is high or the number of recorded photons is low. Several studies have focused on improving the ML estimates in the single-depth estimation problem. Altmann et al. [5] proposed a Bayesian approach, whereas Shin et al. [42], Halimi et al. [17], and Rapp and Goyal [38] suggested three different optimization alternatives. The method introduced in [42] estimates the reflectivity and depth information independently, considering a rank-ordered mean censoring of background photons as a preprocessing step. The optimization method in [17] assumes a negligible background and estimates the depth and reflectivity jointly using an alternating direction method of multipliers (ADMM) algorithm. The algorithm proposed in [38] uses an adaptive superpixel approach to censor background photons and improve depth and reflectivity estimates. In the multiple-surface-per-pixel configuration, Hernandez-Marin, Wallace, and Gibson [23] proposed a pixelwise reversible jump Markov chain Monte Carlo (RJ-MCMC) algorithm. While this approach is able to find an a priori unknown number of surfaces and compute associated uncertainty intervals, it involves a prohibitive computation time. Moreover, it performs poorly when photon counts are relatively low, as it does not account for spatial correlation between neighboring pixels. In later work, Hernandez-Marin, Wallace, and Gibson [24] proposed an extension to the latter algorithm, where a Potts model was used to regularize spatially the number of surfaces per pixel. However, the computational load of their algorithm was prohibitive for large images and the correlation between the amplitude and position of each object was not modelled a priori. There have been other attempts to derive statistical models for Lidar waveforms with an unknown number of objects per pixel, such as Mallet et al. [29] with full waveform topographic Lidar, where a marked point process was considered for each pixel separately. While they defined interactions between pulses in the same pixel, no spatial interaction between points of neighboring pixels was considered. Recently, new optimization approaches have been proposed to tackle the multiple-object-per-pixel problem: Shin et al. [43] introduced an ℓ_1 norm regularization for the recovered peak positions, followed by a postprocessing of the 3D point cloud. Halimi et al. [19] improved it by considering a total variation (TV) operator and the ℓ_{21} norm.
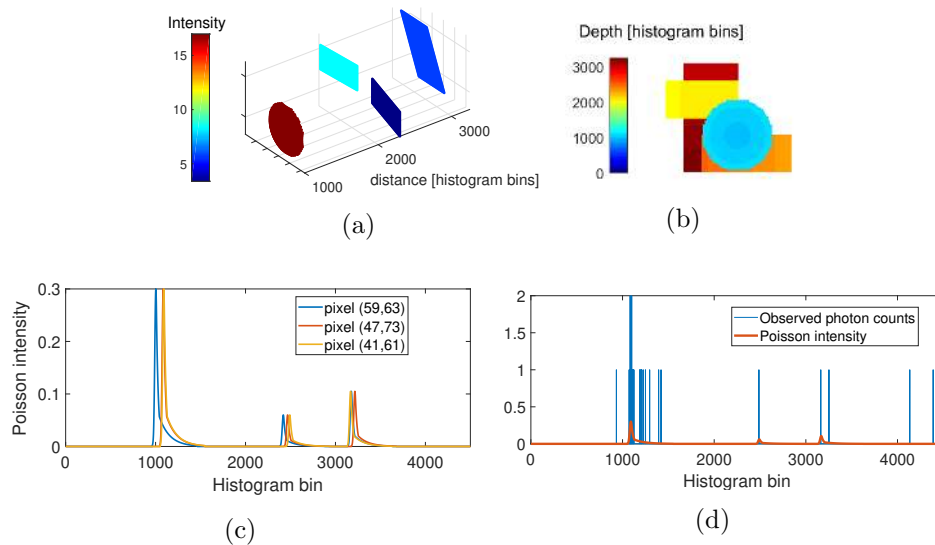
Figure 1. (a) *depicts a synthetic 3D point cloud with $N_r = 99$ rows, $N_c = 99$ columns, and $T = 4500$ bins. The scene consists of three plates with different sizes and orientations and one ball-shaped object. The intensity represents the mean number of photons associated with each 3D point.* (b) *illustrates the depth of the first object for each pixel.* (c) *shows the intensity of three different pixels. The observed photon counts and underlying Poisson intensity of a pixel with three surfaces are shown in* (d).

In this work, we introduce a new spatial point process within a Bayesian framework for modelling single-photon Lidar data. This novel approach considers interactions between points at a pixel level and also at an interpixel level, in a variable dimension configuration. Here, we consider each surface within a pixel as a point in the 3D space, which has a mark that indicates its intensity. Natural Lidar point clouds exhibit strong spatial clustering, as points belonging to the same surface tend to be close in range. Conversely, points in a given pixel tend to be separated as they correspond to different surfaces. Figure 1 shows an example of a synthetic Lidar 3D point cloud to illustrate this phenomenon. This prior information is added to our model using spatial point processes: repulsion between points at a pixel level is achieved with a hard constraint Strauss process, and attraction among points in neighboring pixels is attained by an area interaction process, as defined in [47]. Moreover, the combination of these two processes implicitly defines a connected-surface structure that is used to efficiently sample the posterior distribution. To promote smoothness between reflectivities of points in the same surface, we define a nearest neighbor Gaussian Markov random field (GMRF) prior model, similar to the one proposed in [31]. Inference about the posterior distribution of points, their marks, and the background level is done by an RJ-MCMC algorithm [10, Chapter 9], with carefully tailored moves to obtain high acceptance rates, ensuring better mixing and faster convergence rate. In addition to traditional birth/death, split/merge, shift, and mark moves, new dilation/erosion moves are introduced, which add and remove new points by extending or shrinking a connected surface, respectively. These moves lead to a much higher acceptance rate than those obtained for birth and death updates, as they propose moves to and within regions of high posterior probability. To further reduce the transient regime

of the Markov chains and reduce the computational time of the algorithm, we consider a multiresolution approach, where the original Lidar 3D data is binned into a coarser resolution data cube with higher signal power, lower number of points, and same data statistics. An initial estimate obtained from the downsampled data is used as the initial configuration for the finer scale, thus reducing the number of burn-in iterations needed for the Markov chains to convergence. We assess the quality of reconstruction and the computational complexity in several experiments based on synthetic Lidar data and three real Lidar datasets. The algorithm leads to new efficient 3D reconstructions with processing times similar to those of other existing optimization-based methods. This method can be successfully applied to scenes where there is only one object per pixel, thus generalizing other single-depth algorithms [42, 5, 19, 38]. Moreover, the proposed algorithm can also be applied in scenes where each pixel has at most one surface and it generalizes other target detection methods [6]. We refer to the proposed method as ManiPoP, as it aims to represent 2D manifolds with a 3D point process. In summary, the main contributions of this paper are

1. a new Bayesian model based on a marked point process prior for modelling spatially correlated 3D point clouds,
2. new reversible jump moves proposed for sampling the posterior distribution more efficiently,
3. a multiresolution processing approach to improve the convergence rate, which also allows for a rapid information extraction using only the coarser scales.

The remainder of this work is organized as follows. Section 2 presents the Bayesian model considered for the analysis of multiple-depth Lidar data. Section 3 details the sampling strategy using an RJ-MCMC algorithm. Section 4 discusses the proposed multiresolution approach and other implementation details to reduce the computational load of the algorithm. Results of experiments conducted on synthetic and real data are presented in section 5. Finally, section 6 summarizes our conclusions and discusses future work.

2. Proposed Bayesian model. Recovering the position and intensity of the objects from the raw Lidar data is an ill-posed problem, as the solution is not uniquely identified given the data (e.g., the histogram of Figure 1d). This problem can be tackled in a Bayesian framework, where the data generation mechanism is modelled through a set of parameters $\boldsymbol{\theta}$ that can be inferred using the available data $\boldsymbol{Z}$. The probability of observing a Lidar cube $\boldsymbol{Z}$ is given by the likelihood $p(\boldsymbol{Z}|\boldsymbol{\theta})$. The a priori knowledge of the unknown parameters $\boldsymbol{\theta}$ is embedded in the prior distribution $p(\boldsymbol{\theta}|\boldsymbol{\Psi})$ given a set of hyperparameters $\boldsymbol{\Psi}$. Following Bayes' theorem, the posterior distribution of the model parameters is

$$(2.1) \qquad p(\boldsymbol{\theta}|\boldsymbol{Z}, \boldsymbol{\Psi}) = \frac{p(\boldsymbol{Z}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{\Psi})}{\int p(\boldsymbol{Z}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{\Psi})d\boldsymbol{\theta}}.$$

2.1. Likelihood. A 3D point cloud is represented by an unordered set of points

$$(2.2) \qquad \boldsymbol{\Phi} = \{(\boldsymbol{c}_n, r_n), n = 1, \ldots, N_\Phi\},$$

where N_Φ is the total number of points, $\boldsymbol{c}_n = (x_n, y_n, t_n)^T \in \mathbb{R}^3$ is a coordinate vector, and $r_n \in \mathbb{R}^+$ is the intensity[1] of the nth point. For clarity in the notation, we will also denote the set of point coordinates as $\boldsymbol{\Phi}_c = \{\boldsymbol{c}_n, n = 1, \ldots, N_\Phi\}$ and the set of intensity values as $\boldsymbol{\Phi}_r = \{r_n, n = 1, \ldots, N_\Phi\}$.

According to [23], in the presence of distributed objects, the observed photon count in bin t and pixel (i, j) follows a Poisson distribution, whose intensity is a mixture of the pixel background level $b_{i,j}$ and the responses of the surfaces present in that pixel, i.e.,

$$(2.3) \qquad z_{i,j,t}|(\boldsymbol{\Phi}, b_{i,j}) \sim \mathcal{P}\left(\sum_{n:(x_n, y_n)=(i,j)} g_{i,j} r_n h(t - t_n) + g_{i,j} b_{i,j}\right),$$

where $t \in \{1, \ldots, T\}$, T is the number of histogram bins, $h(\cdot)$ is the known temporal instrumental response, and $g_{i,j}$ is a scaling factor that represents the gain/sensitivity of the detector in pixel (i, j). Assuming mutual independence between the noise realizations in different time bins and pixels, the full likelihood can be written as

$$(2.4) \qquad p(\boldsymbol{Z}|\boldsymbol{\Phi}, \boldsymbol{B}) = \prod_{i=1}^{N_c} \prod_{j=1}^{N_r} \prod_{t=1}^{T} p(z_{i,j,t}|\boldsymbol{\Phi}, b_{i,j}),$$

where $\boldsymbol{Z}$ is the full Lidar cube with $[\boldsymbol{Z}]_{i,j,t} = z_{i,j,t}$, $\boldsymbol{B}$ is the background 2D image, and N_r and N_c are the numbers of pixels in the vertical and horizontal axes, respectively. Note that $p(z_{i,j,t}|\boldsymbol{\Phi}, b_{i,j})$ in (2.4) is the Poisson distribution associated with (2.3).

2.2. Markov marked point process. The set of points $\boldsymbol{\Phi}$ is defined inside the 3D space $\mathcal{T} = [0, N_r] \times [0, N_c] \times [0, T]$. Interactions between points can be characterized by defining densities with respect to the Poisson reference measure, i.e.,

$$f(\boldsymbol{\Phi}_c) \propto f_1(\boldsymbol{\Phi}_c) \ldots f_r(\boldsymbol{\Phi}_c),$$

where $\propto$ means "proportional to." A more detailed definition of the point process theory can be found in section SM1. In this work, we only consider Markovian interactions between points. The benefits of this property are twofold: (a) Markovian interactions are well suited to describe the spatial correlations in natural 3D scenes [32] and (b) inference is performed using only local updates, which leads to a low computational complexity. We can constrain the minimum distance between two different surfaces in the same pixel using the hard object process with density

$$(2.5) \qquad f_1(\boldsymbol{\Phi}_c) \propto \begin{cases} 0 & \text{if } \exists\, n \neq n' : x_n = x_{n'},\, y_n = y_{n'}, \\ & \text{and } |t_n - t_{n'}| < d_{\min}, \\ 1 & \text{otherwise}, \end{cases}$$

which is a special case of the repulsive Strauss process [47], where $d_{\min}$ is the minimum distance between two points in the same pixel. Attraction between points of the same surface in

---

[1]The reflectivity of the point, limited to $(0, 1]$, can be obtained as $\max\{1, r_n/(\eta N_{\mathrm{rep}} \sum_t h(t))\}$, where $\eta \in [0, 1]$ is the quantum efficiency of the detector and N_{rep} is the number of laser pulses sent per pixel.

neighboring pixels cannot be modelled with another Strauss process, due to a phase transition of extremely clustered realizations, as explained in [47, 32]. However, a smoother transition into clustered configurations can be achieved by the area interaction process, introduced by Baddeley and van Lieshout in [8]. In this case, the density is defined as

$$(2.6) \qquad f_2(\boldsymbol{\Phi}_c | \gamma_a, \lambda_a) = k_1 \lambda_a^{N_\Phi} \gamma_a^{-m\left(\bigcup_{n=1}^{N_\Phi} S(\boldsymbol{c}_n)\right)},$$

where λ_a is a positive parameter that controls the total number of points, $\gamma_a \geq 1$ is a parameter adjusting the attraction between points, and k_1 is an intractable normalizing constant. The exponent of γ_a in (2.6) is the measure $m(\cdot)$ over the union of convex sets $S(\boldsymbol{c}_n) \subseteq \mathcal{T}$ centered around each point $\boldsymbol{c}_n$. In this way, the density is bigger when the intersection of the convex sets around two interacting points is closer to the union of them, i.e., if the points are clustered together. The special case $\gamma_a = 1$ corresponds to a Poisson point process (without considering a Strauss process) with an intensity proportional to $\lambda_a \lambda(\cdot)$ (see section SM2 for details). In the rest of this work, we fix $\lambda(\mathcal{T}) = 1$ and control the number of points with the parameter λ_a. The set $S(\boldsymbol{c}_n)$ is defined as a cuboid with a face of $N_p \times N_p$ squared pixels and a depth of $2N_b + 1$ histogram bins, and $m(\cdot)$ is the Lebesgue measure on $\mathcal{T}$. This set determines a cuboid of influence around each point, allowing interactions up to a distance of $\lfloor N_p/2 \rfloor$ pixels and N_b bins. As two points in the same pixel generally correspond to different surfaces, we set $d_{\min} > 2N_b$, thus constraining the minimum distance between two surfaces in the same pixel. The combination of the Strauss process and the area interaction process implicitly defines a connected-surface structure.
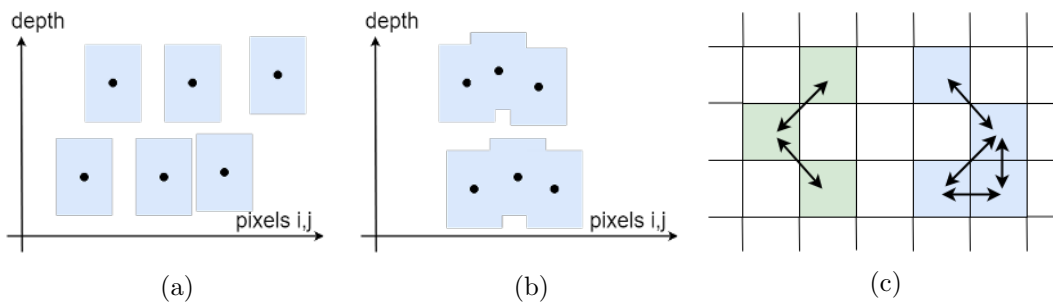

Figure 2. (a) and (b) show two different point configurations. Each point $\boldsymbol{c}_n$ is denoted by a black dot, and the corresponding blue rectangle depicts the area of the convex set $S(\boldsymbol{c}_n)$. The configuration shown in (a) has a lower prior probability than the one shown in (b), as the union of all sets $S(\boldsymbol{c}_n)$ is smaller in (b) with respect to the Lebesgue measure. (c) shows the connectivity at an interpixel level when $N_p = 3$. The green and blue squares correspond to pixels with points associated with two different surfaces, while the white squares denote pixels without points. For simplicity, in this example all points are considered to be present at the same depth. Note that each pixel can be connected with at most eight neighbors.

Figures 2 and 3 illustrate the connected-surface structure via several examples. The hyperparameters γ_a and λ_a of the area interaction process are difficult to estimate, as there is an intractable normalizing constant in the density of (2.6) and standard MCMC methods cannot be directly applied. Although there exist ways of bypassing this problem (e.g., [33]), we fixed these hyperparameters in all our experiments to ensure a reasonable computational complexity.
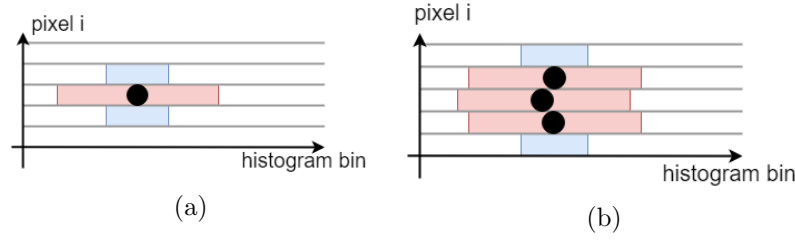
Figure 3. *In both figures, the red color denotes the space where no other points can be found (Strauss process), whereas the blue color denotes the volume where other points are likely to appear (area interaction process with $N_p = 3$). (a) Example of configuration with one point. (b) Example of configuration with three points.*

After defining the spatial priors, the marked point process is constructed by adding the intensity marks r_n to the set $\boldsymbol{\Phi}_c$ with the density detailed in the next section. An illustration of the proposed prior can be found in section SM4.

2.3. Intensity prior model. In natural scenes, the intensity values of points within the same surface exhibit strong spatial correlation. Following the Bayesian paradigm, this prior knowledge can be integrated into our model by defining a prior distribution over the point marks. Gaussian processes are classically used in spatial statistics. However, the underlying covariance structure needs to consider too many neighboring points to attain sufficient smoothing, which involves a prohibitive computational load. In order to obtain similar results with a lower computational burden, we propose to exploit the connected-surface structure to define a nearest neighbor Gaussian Markov random field (GMRF), similar to the one used by McCool et al. in [31]. First, we alleviate the difficulties induced by the positivity constraint of the intensity values by introducing the following change of variables, which is a standard choice in spatial statistics dealing with Poisson noise (see [39, Chapter 4]):

$$(2.7) \qquad m_n = \log(r_n), \quad n = 1, \ldots, N_{\Phi_c}.$$

Second, spatial correlation is promoted by defining the following conditional distribution of the log-intensities:

$$(2.8) \qquad p(m_n | \mathcal{M}_{pp}(\boldsymbol{c_n}), \sigma^2, \beta) \propto \exp\left(-\frac{1}{2\sigma^2} \left(\sum_{n' \in \mathcal{M}_{pp}(\boldsymbol{c_n})} \frac{(m_n - m_{n'})^2}{d(\boldsymbol{c_n}; \boldsymbol{c_{n'}})} + m_n^2 \beta \right) \right),$$

where $\mathcal{M}_{pp}(\boldsymbol{c_n})$ is the set of neighbors of $\boldsymbol{c_n}$, $d(\boldsymbol{c_n}; \boldsymbol{c_{n'}})$ denotes the distance between the points $\boldsymbol{c_n}$ and $\boldsymbol{c_{n'}}$, and β and σ^2 are two positive hyperparameters. The set of neighbors $\mathcal{M}_{pp}(\boldsymbol{c_n})$ is obtained using the connected-surface structure, where each point can have at most $N_p^2 - 1$ neighbors, as illustrated in Figure 2. The distance between two points is computed according to

$$d(\boldsymbol{c_n}; \boldsymbol{c_{n'}}) = \sqrt{(y_n - y_{n'})^2 + (x_n - x_{n'})^2 + \left(\frac{t_n - t_{n'}}{l_z} \right)^2}$$

with $l_z = \Delta_p / \Delta_b$, which normalizes the distance to have a physical meaning, where Δ_p and Δ_b are the approximate spatial resolutions of one pixel and one histogram bin, respectively. This

prior promotes a linear interpolation between neighboring[2] intensity values, as explained in [39]. In this work, we assume that Δ_p is constant throughout the scene. If the scene presents significant distortion, i.e., objects separated by a significant distance in depth, Δ_p should depend on the position by computing the projective transformation between world coordinates and Lidar coordinates (a detailed explanation can be found in [13, 22]). Following the Hammersley–Clifford theorem [20], the joint intensity distribution is given by the multivariate Gaussian distribution

$$(2.9) \qquad\qquad\qquad \boldsymbol{m}|\sigma^2, \beta, \boldsymbol{\Phi}_c \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{P}^{-1}),$$

where $\boldsymbol{P}$ is the unscaled precision matrix of size $N_\Phi \times N_\Phi$ with the following elements:

$$(2.10) \qquad\qquad [\boldsymbol{P}]_{n,n'} = \begin{cases} \beta + \sum_{\tilde{n} \in \mathcal{M}_{pp}(\boldsymbol{c}_n)} \frac{1}{d(\boldsymbol{c}_n; \boldsymbol{c}_{\tilde{n}})} & \text{if } n = n', \\ -\frac{1}{d(\boldsymbol{c}_n; \boldsymbol{c}_{n'})} & \text{if } \boldsymbol{c}_n \in \mathcal{M}_{pp}(\boldsymbol{c}_{n'}), \\ 0 & \text{otherwise.} \end{cases}$$

The parameter σ^2 controls the surface intensity smoothness, and $\frac{\beta}{\sigma^2}$ is related to the intensity variance of a point without any neighbor. In addition, the parameter β ensures a proper joint prior distribution, as $\boldsymbol{P}$ is diagonally dominant, thus full rank [39].

2.4. Background prior model. Noncoherent illumination sources, such as the solar illumination in outdoor scenes or room lights in the indoor case, are related to arrivals of photons at random times (uniformly distributed in time) to the single-photon detector. The level of these spurious detections is modelled as a 2D image of mean intensities $b_{i,j}$ with $i = 1, \ldots, N_r$ and $j = 1, \ldots, N_c$. If the transceiver system of the Lidar is monostatic[3] (e.g., the system described in [30]), the background image is usually similar to the objects present in the scene and exhibits spatial correlation, as background photons generally arise from the ambient light reflecting from parts of the targets and being collected by the system. Hence, we use a hidden gamma Markov random field prior distribution for $\boldsymbol{B}$ that takes into account the background positivity and spatial correlation. This prior was introduced by Dikmen and Cemgil in [12] and applied in many image processing applications with Poisson likelihood [2, 3]. In [12], the distribution of $b_{i,j}$ is defined via auxiliary variables $[\boldsymbol{W}]_{i,j} = w_{i,j}$ such that

$$(2.11) \qquad\qquad\qquad b_{i,j}|\mathcal{M}_B(b_{i,j}), \alpha_B \sim \mathcal{G}\left(\alpha_B, \frac{\overline{b}_{i,j}}{\alpha_B}\right),$$

$$(2.12) \qquad\qquad\qquad w_{i,j}|\mathcal{M}_B(w_{i,j}), \alpha_B \sim \mathcal{IG}(\alpha_B, \alpha_B \overline{w}_{i,j}),$$

where $\mathcal{M}_B$ denotes the set of five neighbors as shown in Figure 4, $\mathcal{G}$ and $\mathcal{IG}$ indicate gamma and inverse gamma distributions, α_B is a hyperparameter controlling the spatial regulariza-

---

[2]The combination of a local Euclidean distance with a nearest neighbors definition can be seen to approximate the manifold metrics [45].

[3]The transceiver system is monostatic when the transmit and receive channels are coaxial and thus share the same objective lens aperture.
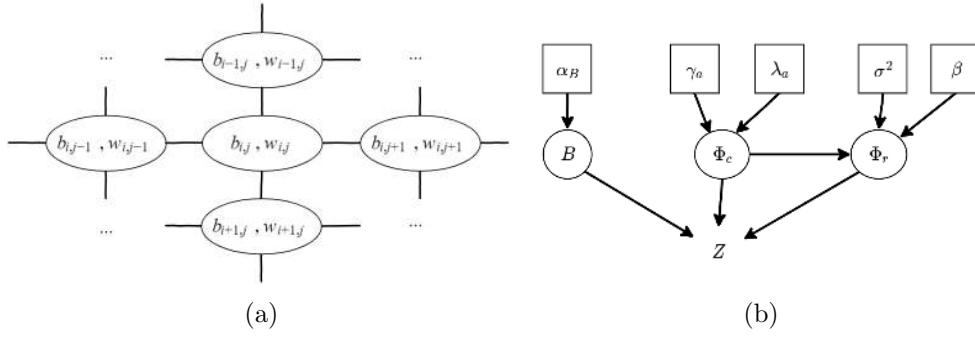
Figure 4. (a) *illustrates the gamma Markov random field neighboring structure* $\mathcal{M}_B$. *Each* $b_{i,j}$ *is connected to five auxiliary variables* $w_{i',j'}$ *as depicted by the continuous lines, including the one with the same subscript. Similarly, each* $w_{i,j}$ *is also connected to five other variables* $b_{i',j'}$ *as indicated by the continuous lines.* (b) *shows the directed acyclic graph of the proposed hierarchical Bayesian model. The variables inside squares are fixed, whereas the variables inside circles are estimated.*

tion, and

$$
(2.13) \qquad \overline{b}_{i,j} = \left(\frac{1}{4} \sum_{(i',j')\in\mathcal{M}_B(b_{i,j})} w_{i',j'}^{-1} \right)^{-1},
$$

$$
(2.14) \qquad \overline{w}_{i,j} = \frac{1}{4} \sum_{(i',j')\in\mathcal{M}_B(w_{i,j})} b_{(i',j')}.
$$

We are interested in the marginal distribution of the GMRF $p(\boldsymbol{B}|\alpha_B)$ that integrates over all possible realizations of the auxiliary variables $w_{i,j}$. The expression of this marginal density can be obtained analytically (as detailed in section SM3) as

$$
(2.15) \qquad p(\boldsymbol{B}|\alpha_B) \propto \int p(\boldsymbol{B},\boldsymbol{W}|\alpha_B)d\boldsymbol{W}
$$

$$
(2.16) \qquad \propto \prod_{i=1}^{N_c}\prod_{j=1}^{N_r} \frac{b_{i,j}^{\alpha_B-1}}{\left(\sum_{(i',j')\in\mathcal{M}_B(w_{i,j})} b_{i',j'}\right)^{\alpha_B}}.
$$

In this work, we fix the value of α_B, even if it could also be estimated using a stochastic gradient procedure as explained in [37], at the expense of an increase in the computational load. If the system is not monostatic, i.e., there is no prior assumption of smoothness in the background image, the value of α_B is set to 1.

2.5. Posterior distribution. The joint posterior distribution of the model parameters is given by

$$
(2.17) \quad p(\boldsymbol{\Phi}_c, \boldsymbol{\Phi}_r, \boldsymbol{B}|\boldsymbol{Z}, \Psi) \propto p(\boldsymbol{Z}|\boldsymbol{\Phi}_c, \boldsymbol{\Phi}_r, \boldsymbol{B})p(\boldsymbol{\Phi}_r|\boldsymbol{\Phi}_c, \sigma^2, \beta)
$$
$$
\times f_1(\boldsymbol{\Phi}_c|\gamma_a, \lambda_a) f_2(\boldsymbol{\Phi}_c|\gamma_{st})\pi(\boldsymbol{\Phi}_c)p(\boldsymbol{B}|\alpha_B),
$$

where Ψ denotes the set of hyperparameters $\Psi = \{\gamma_a, \lambda_a, \gamma_s, \sigma^2, \beta, \alpha_B\}$. Figure 4 shows the directed acyclic graph associated with the proposed hierarchical Bayesian model.

3. Estimation strategy. Bayesian estimators associated with the full posterior in (2.17) are analytically intractable. Moreover, standard optimization techniques cannot be applied due to the highly multimodality of the posterior distribution. However, we can obtain numerical estimates using samples generated by a Monte Carlo method denoted as

$$(3.1) \qquad \{\boldsymbol{\Phi}^{(s)}, \boldsymbol{B}^{(s)} \quad \forall s = 0, 1, \dots, N_i - 1\},$$

where N_i is the total number of samples. In this work, we will focus on the maximum a posteriori (MAP) estimator of the point cloud positions and intensity values, i.e.,

$$(3.2) \qquad \hat{\boldsymbol{\Phi}} = \arg \max_{\boldsymbol{\Phi}} p(\boldsymbol{\Phi}, \boldsymbol{B} | \boldsymbol{Z}, \Psi),$$

which is approximated by

$$(3.3) \qquad \hat{\boldsymbol{\Phi}} \approx \arg \max_{s=0,\dots,N_i-1} p(\boldsymbol{\Phi}^{(s)}, \boldsymbol{B}^{(s)} | \boldsymbol{Z}, \Psi).$$

In our experiments, we found that the minimum mean squared error estimator of $\boldsymbol{B}$, i.e.,

$$(3.4) \qquad \hat{\boldsymbol{B}} = \mathbb{E}\{\boldsymbol{B} | \boldsymbol{Z}, \Psi\},$$

achieves better background estimates than the MAP estimator. This estimator can be approximated by the empirical mean of the posterior samples of $\boldsymbol{B}$, that is,

$$(3.5) \qquad \hat{\boldsymbol{B}} \approx \frac{1}{N_i} \sum_{s=N_{\mathrm{bi}}+1}^{N_i} \boldsymbol{B}^{(s)},$$

where $N_{\mathrm{bi}} = N_i/2$ is the number of burn-in iterations. In many applications, assessing the presence or absence of a target at a pixel level can be of special interest (e.g., [19, 6]). Here, we can use the Monte Carlo samples to estimate the probability of having k objects present in pixel (i, j) as

$$(3.6) \qquad P(k \text{ returns in } (i,j) | \boldsymbol{Z}, \Psi) = \frac{1}{N_i} \sum_{s=N_{\mathrm{bi}}+1}^{N_i} \mathbb{1}_{k \text{ points in } (i,j)}(\boldsymbol{\Phi}^{(s)}).$$

Remark. If more detailed posterior statistics are needed, it is possible to fix the dimensionality of the problem using the estimate $\hat{\boldsymbol{\Phi}}$ and run a fixed dimensional sampler for additional N_i iterations (see section SM5).

Many samplers capable of exploring different model dimensions, i.e., different numbers of points, are available in the point process literature (a complete summary can be found in [10, Chapter 9]). The continuous birth-death chain method builds a continuous-time Markov chain that converges to the posterior distribution of interest. Alternatively, perfect sampling approaches generate samples using a rejection sampling scheme, which incurs a bigger computational load. Finally, the RJ-MCMC sampler, introduced by Green in [15], constructs a discrete time Markov chain, where moves between different dimensions are proposed and

accepted or rejected in order to converge to the posterior distribution of interest. In this work, we choose an RJ-MCMC sampler, as this option allows us to design application-specific proposals that speed up the convergence rate.

In addition, we propose a data augmentation scheme to sample the background levels. This technique introduces extra auxiliary (latent) variables $\boldsymbol{u}$ and generates samples in this augmented model space, $(\boldsymbol{B}^{(s)}, \boldsymbol{u}^{(s)}) \sim p(\boldsymbol{B}, \boldsymbol{u} | \boldsymbol{Z}, \boldsymbol{\Phi}, \alpha_B)$, which is easier than sampling the marginal distribution $p(\boldsymbol{B} | \boldsymbol{Z}, \boldsymbol{\Phi}, \alpha_B)$. The resulting samples $\boldsymbol{B}^{(s)}$ are distributed according to the desired marginal density (detailed theory and applications of data augmentation can be found in [10, Chapter 10]).

3.1. Reversible jump Markov chain Monte Carlo.
RJ-MCMC can be seen as a natural extension of the Metropolis–Hastings algorithm for problems with an unknown a priori dimensionality. Given the actual state of the chain $\boldsymbol{\theta} = \{\boldsymbol{\Phi}, \boldsymbol{B}\}$ of model order N_Φ, a random vector of auxiliary variables $\boldsymbol{u}$ is generated to create a new state $\boldsymbol{\theta}' = \{\boldsymbol{\Phi}', \boldsymbol{B}'\}$ of model order $N_{\Phi'}$, according to an appropriate deterministic function $\boldsymbol{\theta}' = g(\boldsymbol{\theta}, \boldsymbol{u})$. To ensure reversibility, an inverse mapping with auxiliary random variables $\boldsymbol{u}'$ has to exist such that $\boldsymbol{\theta} = g^{-1}(\boldsymbol{\theta}', \boldsymbol{u}')$. The move $\boldsymbol{\theta} \to \boldsymbol{\theta}'$ is accepted or rejected with probability $\rho = \min\{1, r(\boldsymbol{\theta}, \boldsymbol{\theta}')\}$, where $r(\cdot, \cdot)$ satisfies the so-called dimension balancing condition

$$(3.7) \qquad r(\boldsymbol{\theta}, \boldsymbol{\theta}') = \frac{p(\boldsymbol{\theta}' | \boldsymbol{Z}, \Psi) K(\boldsymbol{\theta} | \boldsymbol{\theta}') p(\boldsymbol{u}')}{p(\boldsymbol{\theta} | \boldsymbol{Z}, \Psi) K(\boldsymbol{\theta}' | \boldsymbol{\theta}) p(\boldsymbol{u})} \left| \frac{\partial g(\boldsymbol{\theta}, \boldsymbol{u})}{\partial(\boldsymbol{\theta}, \boldsymbol{u})} \right|,$$

where $K(\boldsymbol{\theta}' | \boldsymbol{\theta})$ is the probability of proposing the move $\boldsymbol{\theta} \to \boldsymbol{\theta}'$, $p(\boldsymbol{u})$ is the probability distribution of the random vector $\boldsymbol{u}$, and $\left| \frac{\partial g(\boldsymbol{\theta}, \boldsymbol{u})}{\partial(\boldsymbol{\theta}, \boldsymbol{u})} \right|$ is the Jacobian of the mapping $g(\cdot)$. All the terms involved in (3.7) have a complexity that depends only on the size of the neighborhood, except the prior distribution of the intensity values defined in (2.9). Note that (3.7) involves the computation of the ratio of determinants of the precision matrices $\boldsymbol{P}$ and $\boldsymbol{P}'$, which have a global dependency on all the points in Φ_r. To keep the computational complexity low, we address this difficulty by only considering a block diagonal approximation of $\boldsymbol{P}$, which includes only points in local neighborhoods (see section SM7 for more details). The RJ-MCMC algorithm performs birth, death, dilation, erosion, spatial shift, mark shift, split, and merge moves with probabilities p_{birth}, p_{death}, p_{dilation}, p_{erosion}, p_{shift}, p_{mark}, p_{split}, and p_{merge}. These moves are detailed in the following subsections. For ease of reading we summarize the key aspects of each move, without specifying the full acceptance rate expression of (3.7), which can be found in section SM8.

3.1.1. Birth and death moves.
The birth move proposes a new point $(\boldsymbol{c}_{N_\Phi+1}, r_{N_\Phi+1})$ uniformly at random in $\mathcal{T}$. The intensity of the new point is computed according to the following scheme:

$$(3.8) \qquad \begin{cases} u \sim \mathcal{U}(0, 1), \ b'_{i,j} = u b_{i,j}, \\ e^{m_{N_\Phi+1}} = (1 - u) b_{i,j} \dfrac{T}{\sum_{t=1}^{T} h(t)}. \end{cases}$$

This mapping preserves the total posterior intensity of the pixel, since

$$(3.9) \qquad e^{m_{N_\Phi+1}} \sum_{t=1}^{T} h(t) + b'_{i,j} T = b_{i,j} T,$$

thus yielding a relatively high acceptance probability. Its reversible pair, the death move, proposes to remove one point randomly. In this case, the inverse mapping is given by

$$(3.10) \qquad b'_{i,j} = b_{i,j} + e^{m_{N_\Phi+1}} \frac{\sum_{t=1}^{T} h(t)}{T}.$$

The acceptance ratio for the birth move reduces to $\rho = \min\{1, C_1\}$ with C_1 given by (3.7), where the posterior ratio is computed according to (2.17), $K(\boldsymbol{\theta}'|\boldsymbol{\theta}) = p_{\text{birth}}$, $K(\boldsymbol{\theta}|\boldsymbol{\theta}') = p_{\text{death}}$, $p(\boldsymbol{u}) = \frac{\lambda(\cdot)}{\lambda(\mathcal{T})}$ and $p(\boldsymbol{u}') = \frac{1}{N_\Phi+1}$, and a Jacobian equal to $\frac{1}{1-u}$. The death move is accepted or rejected with probability $\rho = \min\{1, C_1^{-1}\}$, modifying $p(\boldsymbol{u})$ accordingly (i.e., changing $\frac{1}{N_\Phi+1}$ to $\frac{1}{N_\Phi}$).

3.1.2. Dilation and erosion moves.

Standard birth and death moves yield low acceptance rates, because the probability of proposing a point in a likely position is relatively low, as the detected surfaces only occupy a small subset of the full 3D volume $\mathcal{T}$. To overcome this problem, we propose new RJ-MCMC moves that explore the target distribution by dilating and eroding existing surfaces. The dilation move randomly picks a point $\boldsymbol{c}_n$ that has fewer than $N_p^2 - 1$ neighbors, and then proposes a new neighbor $\boldsymbol{c}_{N_\Phi+1}$ with uniform probability across all possible pixel positions (where a point can be added). The new intensity can be sampled from the Gaussian prior, taking into account the available information from the neighbors, i.e., u is sampled from the conditional distribution specified in (2.8) and $m_{N_\Phi+1} = u$. The background level is adjusted to keep the total intensity of the pixel unmodified:

$$(3.11) \qquad b'_{i,j} = b_{i,j} - e^{m_{N_\Phi+1}} \frac{\sum_{t=1}^{T} h(t)}{T}.$$

If the resulting background level in (3.11) is negative, the move is rejected. The complementary move (named erosion) proposes to remove a point $\boldsymbol{c}_n$ with one or more neighbors. In a similar fashion to the birth move, a dilation is accepted with probability $\rho = \min\{1, C_2\}$, with C_2 computed according to (3.7). In this case, $p(\boldsymbol{u}) = p(u_1)p(u_2)$ with

$$(3.12) \qquad p(u_1) = \frac{1}{N_\Phi(2N_b + 1)} \sum_{m \in \mathcal{M}_{pp}(\boldsymbol{c}_{N_\Phi+1})} \# \mathcal{M}_{pp}(\boldsymbol{c}_m),$$

where $0 \leq \# \mathcal{M}_{pp}(\boldsymbol{c}_m) \leq N_p^2 - 1$ denotes the number of neighboring points of $\boldsymbol{c}_m$. The expression of $p(u_2)$ is given by the conditional distribution defined in (2.8), and the Jacobian term equals 1. The probability of u' is given by

$$(3.13) \qquad p(u') = \frac{1}{\sum_{m=1}^{N_\Phi+1} \mathbb{1}_{\mathbb{Z}_+}(\# \mathcal{M}_{pp}(\boldsymbol{c}_m))},$$

and the transition probabilities are $K(\boldsymbol{\theta}'|\boldsymbol{\theta}) = p_{\text{dilation}}$ and $K(\boldsymbol{\theta}|\boldsymbol{\theta}') = p_{\text{erosion}}$. An erosion move is accepted with probability $\rho = \min\{1, C_2^{-1}\}$.
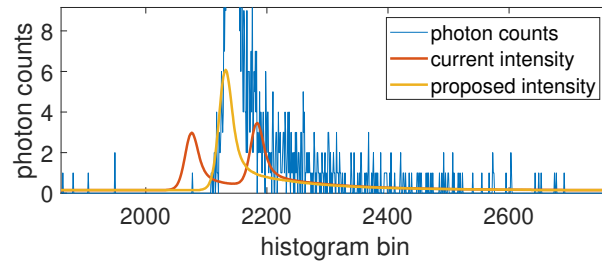
Figure 5. *In scenarios where the sampler proposes two points (red line) instead of one (yellow line), the probability of killing one of them and shifting the other is very low. However, accepting a merge move has high probability.*

3.1.3. Shift move. The shift move modifies the position of a given point. The point is chosen uniformly at random and a new position inside the same pixel is proposed using a random walk Metropolis proposal defined as

$$(3.14) \qquad u \sim \mathcal{N}\left(t_n, \delta_t\right),$$

and $t'_n = u$. The resulting acceptance ratio is $\rho = \min\{1, C_3\}$, with C_3 computed according to (3.7), where $K(\boldsymbol{\theta}'|\boldsymbol{\theta}) = K(\boldsymbol{\theta}|\boldsymbol{\theta}') = p_{\text{shift}}$, $p(u) = p(u')$ are given by the Gaussian distribution of (3.14), and the Jacobian term equals 1. The value of δ_t is set to $(\frac{N_b}{3})^2$ to obtain an acceptance ratio close to 41%, which is the optimal value, as explained in [10, Chapter 4].

3.1.4. Mark move. Similarly to the shift move, the mark move refines the intensity value of a randomly chosen point. The corresponding proposal is a Gaussian distribution with variance δ_m,

$$(3.15) \qquad u \sim \mathcal{N}\left(m_n, \delta_m\right),$$

and $m'_n = u$. In this move, the acceptance ratio is $\rho = \min\{1, r(\boldsymbol{\theta}, \boldsymbol{\theta}')\}$, where $K(\boldsymbol{\theta}'|\boldsymbol{\theta}) = K(\boldsymbol{\theta}|\boldsymbol{\theta}') = p_{\text{mark}}$, $p(u) = p(u')$ are given by (3.15), and the Jacobian term equals 1. As in the shift move, we set the value of δ_m to $(0.5)^2$ to obtain an acceptance ratio close to 41%.

3.1.5. Split and merge moves. In Lidar histograms with many photon counts per pixel, the likelihood function becomes very peaky and the nonconvexity of the problem becomes more difficult to handle. This nonconvexity is related to the discrete nature of the point process, similar to problems where the l_0 pseudo-norm regularization is used, as discussed in [49]. In such cases, when one true surface is associated with two points, as illustrated in Figure 5, the probability of performing a death move followed by a shift move is very low. To alleviate this problem, we propose a merge move and its complement, the split move. A merge move is performed by randomly choosing two points $\boldsymbol{c}_{k_1}$ and $\boldsymbol{c}_{k_2}$ inside the same pixel $(x_{k_1} = x_{k_2}$ and $y_{k_1} = y_{k_2})$ that satisfy the condition

$$(3.16) \qquad d_{\min} < |t_{k_1} - t_{k_2}| \leq \text{attack}_{h(t)} + \text{decay}_{h(t)},$$

where $\text{attack}_{h(t)}$ is the length of the impulse response until the maximum and $\text{decay}_{h(t)}$ is the length after the maximum until the value where $h(t)$ is negligible. The merged point $(\boldsymbol{c}'_n, r'_n)$

is finally obtained by the mapping

$$
(3.17) \quad
\begin{cases}
e^{m'_n} = e^{m_{k_1}} + e^{m_{k_2}}, \\[2mm]
t'_n = t_{k_1} \dfrac{e^{m_{k_1}}}{e^{m_{k_1}} + e^{m_{k_2}}} + t_{k_2} \dfrac{e^{m_{k_2}}}{e^{m_{k_1}} + e^{m_{k_2}}}
\end{cases}
$$

that preserves the total pixel intensity and weights the spatial shift of each peak according to its relative amplitude. For instance, if two peaks of significantly different amplitudes are merged, the resulting peak will be closer to the original peak which presents the highest amplitude. The split move randomly picks a point $(\boldsymbol{c}'_n, r'_n)$ and proposes two new points, $(\boldsymbol{c}_{k_1}, r_{k_1})$ and $(\boldsymbol{c}_{k_2}, r_{k_2})$, following the inverse mapping

$$
(3.18) \quad
\begin{cases}
u \sim \mathcal{U}(0,1), \\
\Delta \sim \mathcal{U}(d_{\min}, \text{attack}_{h(t)} + \text{decay}_{h(t)}), \\
m_{k_1} = m'_n + \log(u), \\
m_{k_2} = m'_n + \log(1-u), \\
t_{k_1} = t'_n - (1-u)\Delta, \\
t_{k_2} = t'_n + u\Delta,
\end{cases}
$$

which is based on the auxiliary variables u and Δ. This proposal verifies (3.17), ensuring reversibility. The acceptance ratio for the split move is $\rho = \min\{1, C_4\}$, with C_4 computed according to (3.7), where the Jacobian is $1/u(1-u)$, $K(\boldsymbol{\theta}'|\boldsymbol{\theta}) = p_{\text{shift}}$, $K(\boldsymbol{\theta}|\boldsymbol{\theta}') = p_{\text{merge}}$, $p(u) = \frac{1}{N_\Phi}(d_{\min} + \text{attack}_{h(t)} + \text{decay}_{h(t)})^{-1}$, and $p(u')$ is the inverse of the number of points in $\boldsymbol{\Phi}$ that verify (3.16). The acceptance probability of the merge move is simply $\rho = \min\{1, C_3^{-1}\}$.

3.2. Sampling the background. In the presence of at least one peak in a given pixel, Gibbs updates cannot be directly applied to obtain background samples, as the linear combination of objects and background level in (2.3) cancels the conjugacy between the Poisson likelihood and the gamma prior. However, this problem can be overcome by introducing auxiliary variables in a data augmentation scheme. In a similar fashion to [50], we propose to augment (2.3) as

$$
z_{i,j,t} = \sum_{n:(x_n,y_n)=(i,j)} \tilde{z}_{i,j,t,n} + \tilde{z}_{i,j,t,b},
$$

$$
\tilde{z}_{i,j,t,b} \sim \mathcal{P}(g_{i,j} b_{i,j}),
$$

$$
\tilde{z}_{i,j,t,n} \sim \mathcal{P}(g_{i,j} r_n h(t - t_n)),
$$

where $\tilde{z}_{i,j,t,n}$ are the photons in bin $\#t$ associated with the kth surface and $\tilde{z}_{i,j,t,b}$ are the ones associated with the background. If we also add the auxiliary variables $w_{i,j}$ of the GMRF (as explained in subsection 2.4), we can construct the following Gibbs sampler:

$$
(3.19) \quad
\begin{cases}
\tilde{z}_{i,j,t,b} \sim \mathcal{B}\left(z_{i,j,t}, \dfrac{b_{i,j}}{\sum_{n:(x_n,y_n)=(i,j)} \exp(m_n) h(t - t_n)} \right), \\[4mm]
w_{i,j} \sim \mathcal{IG}(\alpha_B, \alpha_B \overline{w}_{i,j}), \\[2mm]
b_{i,j} \sim \mathcal{G}\left(\alpha_B + \displaystyle\sum_{t=1}^{T} \tilde{z}_{i,j,t,b}, \dfrac{1}{T + \frac{\alpha_B}{\bar{b}_{i,j}}} \right),
\end{cases}
$$

p_{birth}	1/24	p_{death}	1/24	p_{dilation}	5/24	p_{erosion}	5/24
p_{shift}	5/24	p_{mark}	5/24	p_{split}	1/24	p_{merge}	1/24

where $\mathcal{B}(\cdot)$ denotes the binomial distribution, and $\overline{w}_{i,j}$ and $\overline{b}_{i,j}$ are defined according to (2.14) and (2.13), respectively. The transition kernel defined by (3.19) produces samples of $b_{i,j}$ distributed according to the marginal distribution of (2.15). In practice, we use only one iteration of this kernel.

3.3. Full algorithm. The RJ-MCMC algorithm alternates between birth, death, dilation, erosion, shift, mark, split, and merge moves with probabilities as reported in Table 1. A complete background update is done every $N_B = N_r N_c$ iterations. After each accepted update, we compute the difference in the posterior density δ_{map} in order to keep track of the maximum density map_{max}. After $N_{\text{bi}} = N_i/2$ burn-in iterations, we save the set of parameters $\mathbf{\Phi}$ that yield the highest posterior density, and we also accumulate the samples of $\boldsymbol{B}$ to compute (3.5). Algorithm 3.1 shows a pseudo-code of the resulting RJ-MCMC sampler.

Algorithm 3.1. ManiPoP.

1: **Input:** Lidar waveforms $\boldsymbol{Z}$, initial estimate $(\mathbf{\Phi}^{(0)}, \boldsymbol{B}^{(0)})$, and hyperparameters Ψ
2: **Initialization:**
3: $(\mathbf{\Phi}, \boldsymbol{B}) \leftarrow (\mathbf{\Phi}^{(0)}, \boldsymbol{B}^{(0)})$
4: $s \leftarrow 0$
5: **Main loop:**
6: **while** $s < N_i$ **do**
7:    **if** $\text{rem}(s, N_B) == 0$ **then**
8:       $(\mathbf{\Phi}, \boldsymbol{B}, \delta_{\text{map}}) \leftarrow$ sample $\boldsymbol{B}$ using (3.19)
9:    **end if**
10:    move $\sim \text{Discrete}(p_{\text{birth}}, \ldots, p_{\text{merge}})$
11:    $(\mathbf{\Phi}, \boldsymbol{B}, \delta_{\text{map}}) \leftarrow$ perform selected move
12:    $\text{map} \leftarrow \text{map} + \delta_{\text{map}}$
13:    **if** $s \geq N_{\text{bi}}$ **then**
14:       $\hat{\boldsymbol{B}} \leftarrow \hat{\boldsymbol{B}} + \boldsymbol{B}$
15:       **if** $\text{map} > \text{map}_{\text{max}}$ **then**
16:          $\hat{\mathbf{\Phi}} \leftarrow \mathbf{\Phi}$
17:          $\text{map}_{\text{max}} \leftarrow \text{map}$
18:       **end if**
19:    **end if**
20:    $s \leftarrow s + 1$
21: **end while**
22: $\hat{\boldsymbol{B}} \leftarrow \hat{\boldsymbol{B}}/(N_i - N_{\text{bi}})$
23: **Output:** Final estimates $(\hat{\mathbf{\Phi}}, \hat{\boldsymbol{B}})$

---

4. Efficient implementation. In order to achieve a computational performance similar to that of other optimization-based approaches, while allowing a more complex modelling of the input data, we have considered the following implementation aspects.

1. Recently, the algorithm reported in [7] showed that state-of-the-art denoising of images corrupted with Poisson noise can be obtained by starting from a coarser scale and progressively refining the estimates in finer scales. We propose a similar multiscale approach to achieve faster processing times and better scalability with the total data size. The proposed sequential procedure is detailed in subsection 4.1.

2. In the photon-starved regime considered in this work, the recorded histograms are generally extremely sparse, meaning that more than 95% of the time bins are empty. Therefore, a histogram representation is inefficient, in terms of both likelihood evaluation and memory requirements. In [42], the authors replaced the histograms by modelling directly each detected photon. Similarly, we represent the Lidar data by using an ordered list of bins and photon counts, only considering bins with at least one count (see section SM6 for more details).

3. In order to avoid finding neighbors of a point to be updated at each iteration, we store and update an adjacency list for each point. This list allows the neighbor search only during the creation or shift of a point.

4. To reduce the search space, we add a preprocessing step that computes the matched-filter response at the coarsest resolution. The time bins whose values are below a threshold (equal to $\frac{0.05}{T} \sum_{t=1}^{T} z_{i,j,t} \sum_{t=1}^{T} \log h(t)$) are assigned zero intensity in the point process prior, i.e., $\lambda(\cdot) = 0$. In this way, the search includes with high probability objects in pixels with signal-to-background ratio (SBR) higher than 0.05 (see section SM11 for a more detailed explanation).

5. When the number of photons per pixel is very high, the binomial sampling step of (3.19) is replaced by a Poisson approximation, i.e.,
$\sum_{t=1}^{T} \tilde{z}_{i,j,t,b} \sim \mathcal{P}(\sum_{t=1}^{T} \frac{b_{i,j} z_{i,j,t}}{\sum_{n:(x_n,y_n)=(i,j)} r_n h(t-t_n)+b_{i,j}})$.

4.1. Multiresolution approach. We downsample the input 3D data by summing the contents over $N_{\text{bin}} \times N_{\text{bin}}$ windows. This aggregation results in a smaller Lidar image that keeps the same Poisson statistics, where each bin can present an intensity N_{bin}^2 bigger (on average). Hence, a Lidar data cube with higher signal-to-noise ratio, approximately N_{bin}^2 fewer points to infer, and a similar observational model (if the broadening of the impulse response can be neglected) is obtained. In this way, we run Algorithm 3.1 on the downsampled data to get an initial coarse estimate of the 3D scene. This estimate is then upsampled and used as the initial condition for the finer resolution data. The point cloud $\mathbf{\Phi}$ is upsampled using a linear interpolator for fast computation. Following the connected-surface structure of ManiPoP, each of the estimated surfaces is upsampled independently of the rest. However, more elaborate algorithms can be also used, such as moving least squares, as detailed in [26]. These two steps can be performed in K scales, whereby, for each scale, the Lidar data $\mathbf{Z}_k$ is obtained by aggregating $\mathbf{Z}_{k+1}$. Algorithm 4.1 summarizes the proposed sequential multiscale approach.

Algorithm 4.1. Multiresolution ManiPoP.

> **Input:** Lidar scene $\boldsymbol{Z}$, hyperparameters Ψ, window size N_{binning}, and number of scales K
> **Initialization:**
> $\boldsymbol{\Phi}_1^{(0)} \leftarrow \emptyset$
> $\boldsymbol{B}_1^{(0)} \leftarrow$ sample from (3.19)
> **Main loop:**
> **for** $k = 1, \ldots, K$ **do**
>   **if** $k > 1$ **then**
> $(\boldsymbol{\Phi}_k^{(0)}, \boldsymbol{B}_k^{(0)}) \leftarrow \text{upsample}(\hat{\boldsymbol{\Phi}}_{k-1}, \hat{\boldsymbol{B}}_{k-1})$
>   **end if**
> $(\hat{\boldsymbol{\Phi}}_k, \hat{\boldsymbol{B}}_k) \leftarrow \text{ManiPoP}(\boldsymbol{Z}_k, (\boldsymbol{\Phi}_k^{(0)}, \boldsymbol{B}_k^{(0)}), \Psi)$
> **end for**
> **Output:** $(\hat{\boldsymbol{\Phi}}_K, \hat{\boldsymbol{B}}_K)$

---

5. Experiments. The proposed method was evaluated with synthetic and real Lidar data. In all experiments, we denote the bin length as $\Delta_b = \frac{T_b c}{2}$, where c is the speed of light in the scene medium and T_b is the bin width used in the TCSPC timing histogram. We also indicate the mean number of photons per pixel as $\bar{\lambda}_p$, which is proportional to the per-pixel acquisition time. Our method is compared with the classical log-matched filtering solution and two recent algorithms. The first is referred to as SPISTA [43] and considers an ℓ_1 regularization to promote sparsity in the recovered peaks. The second algorithm, the method presented in [19], is referred to as ℓ_{21}+TV. It considers an ℓ_{21} and total variation regularizations to promote smoothness between points in neighboring pixels. In our experiments, we have slightly modified both SPISTA and ℓ_{21}+TV to attain better results, as explained in subsection 5.2. The RJ-MCMC algorithm proposed in [24] was not considered in this work, as its computational complexity is hardly compatible with large images (for a scene of $N_r = 100 = N_c = 100$ pixels and $T = 4500$ bins, the algorithm takes more than a day of computation). The log-matched filtering solution is the depth ML estimator when the background is negligible and in the presence of a single peak, i.e., $\hat{t}_{i,j} = \arg\max_{t_{i,j} \in [1,T]} \sum_{t=1}^T z_{i,j,t} \log[h(t - t_{i,j})]$. The intensity estimator can then be obtained as $\hat{r}_{i,j} = \sum_{t=1}^T z_{i,j,t} / (g_{i,j} \sum_{t=1}^T h(t))$. In order to infer the background levels, we constrain the intensity estimate to the support of $h(t)$ leading to $\tilde{r}_{i,j} = \sum_{t=\hat{t}_{i,j}-\text{attack}}^{\hat{t}_{i,j}+\text{decay}} z_{i,j,t} / (g_{i,j} \sum_{t=1}^T h(t))$. The background components can then be computed using the residual photons as $\hat{b}_{i,j} = \sum_{t=1}^T z_{i,j,t} \mathbb{1}_{h(t-\hat{t}_{i,j}^k)=0}(t) / (g_{i,j} \sum_{t=1}^T \mathbb{1}_{h(t-\hat{t}_{i,j}^k)=0}(t))$. The corrected intensity estimate is finally computed as $\hat{r}_{i,j} = \min\{\tilde{r}_{i,j} - \hat{b}_{i,j}, 0\}$. For visualization purposes, all the intensity results obtained by different algorithms were normalized (postprocessing step) under the condition $\sum_{t=1}^T h(t) = 1$, such that the estimated intensity has a value that reflects the number of signal photons attributed to the corresponding 3D location. In the experiments, we used only two scales, a coarse one using a binning window of $N_{\text{bin}} = 3$ pixels and the full resolution. The hyperparameters were adjusted with the following considerations:

- The cuboid length N_b should be fixed according to the relative scale between the bin

Table 2
Hyperparameter values.

Hyperparameters	γ_a	λ_a	N_p	N_b	$d_{\min}$	σ^2	β	α_B
Coarse scale	e^2	$(N_r N_r / N_b^2)^{1.5}$	3	$3\Delta_p/\Delta_{\text{bin}}$	$2N_b + 1$	0.6^2	$\sigma^2/100$	2
Fine scale	e^3	$(N_r N_r)^{1.5}$	3	$3\Delta_p/\Delta_{\text{bin}}$	$2N_b + 1$	$0.6^2/3$	$\sigma^2/100$	2

width and the pixel resolution.
- The minimum distance between two points in the same pixel can be set as $d_{\min} = 2N_b + 1$, thus verifying the condition $d_{\min} > 2N_b$.
- The parameters controlling the number of points and the spatial correlation were set by cross-validation using many Lidar datasets leading to $\gamma_a = e^2$ and $\lambda_a = (N_r N_c)^{1.5}$.
- For each scale, we scaled the impulse response $h'(t) = h(t)\frac{\bar{\lambda}_p}{5\sum_t h(t)}$, where $h(t)$ is the unit gain impulse response, such that all intensity values lie approximately in the interval $[0, 10]$. The regularization parameters were then fixed to $\sigma^2 = 0.6^2$ and $\beta = \sigma^2/100$ by cross-validation in order to obtain smooth estimates.
- The hyperparameter controlling the smoothness in the background image $\boldsymbol{B}$ was also adjusted by cross-validation yielding $\alpha_B = 2$.

Table 2 summarizes the different hyperparameter values for the coarse and fine scales. All the experiments were performed using $N_i = 25N_r N_c$ iterations in the coarse scale and fine scale.

5.1. Error metrics. Three different error metrics are used to evaluate the performance of the proposed algorithm. We compare the percentage of true detections $F_{\text{true}}(\tau)$ as a function of the distance τ, considering an estimated point as a true detection if there is another point in the ground truth/reference point cloud in the same pixel ($x_n^{\text{true}} = x_{n'}^{\text{est}}$ and $y_n^{\text{true}} = y_{n'}^{\text{est}}$) such that $|t_n^{\text{true}} - t_{n'}^{\text{est}}| \leq \tau$. We also consider the number of points that were falsely created, denoted as $F_{\text{false}}(\tau)$ (i.e., the estimated points that cannot be assigned to any true point at a distance of τ). Regarding the intensity estimates, we focus on targetwise comparison, by gating the 3D reconstruction between the ranges where a specific target can be found, keeping only the point with biggest intensity and assigning zero intensity to the empty pixels. We compute the normalized mean squared error (NMSE) of the resulting 2D intensity image as

$$(5.1) \qquad \text{NMSE}_{\text{target}} = \frac{\sum_{i=1}^{N_r} \sum_{j=1}^{N_r} (r_{i,j}^{\text{true}} - \hat{r}_{i,j})^2}{\sum_{i=1}^{N_r} \sum_{j=1}^{N_r} \left(r_{i,j}^{\text{true}}\right)^2}.$$

Finally, we consider the NMSE metric for the background image

$$(5.2) \qquad \text{NMSE}_{\boldsymbol{B}} = \frac{\sum_{i=1}^{N_r} \sum_{j=1}^{N_r} (b_{i,j}^{\text{true}} - \hat{b}_{i,j})^2}{\sum_{i=1}^{N_r} \sum_{j=1}^{N_r} \left(b_{i,j}^{\text{true}}\right)^2}.$$

5.2. Synthetic data. We evaluated the algorithm in two synthetic datasets: a simple one, containing basic geometric shapes, and a complex one, based on a scene from the Middlebury dataset [40]. Both scenes present multiple surfaces per pixel. The first scene, shown in Figure 6, has dimensions $N_r = N_c = 99$, $T = 4500$, $\Delta_b = 1.2$ mm, and $\Delta_p \approx 8.5$ mm. The
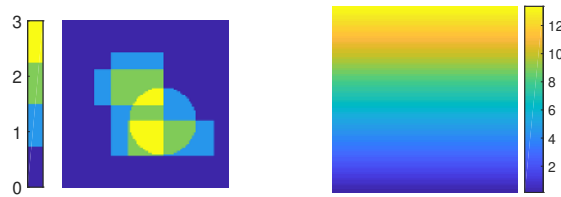
Figure 6. *The* 3*D scene depicted in* Figure 1 *consists of three plates with different sizes and orientations and one paraboloid-shaped object. Left: Number of objects per pixel. Right: Mean background photon count* $T\boldsymbol{B}$.

impulse response used in our experiments was obtained from real Lidar measurements, with attack = 58 bins and decay = 460 bins. The background was created using a linear intensity profile, as shown in Figure 6. The resulting mean intensity per pixel was $\bar{\lambda}_p = 11$, meaning that 99.75% of the bins are empty and approximately 4 photons per pixel are due to 3D objects. First we evaluated the performance with and without the proposed priors to show their effect on the final estimates. The algorithm was tested in the following conditions:

1. with all the priors as reported in Table 2,
2. without spatial regularization ($\gamma_a = 1$),
3. with a weak intensity regularization ($\sigma^2 = 100^2$),
4. with a softer spatial regularization for the background levels ($\alpha_B = 1$),
5. without erosion and dilation moves,
6. only using the finest scale, adjusting the number of iterations to yield the same computing time.

The total execution time for all cases was approximately 120 seconds. Figure 7a shows $F_{\text{true}}(\tau)$ and $F_{\text{false}}(\tau)$ for all the configurations. The number of false points increases dramatically when the area interaction process is not considered, as the sampler tends to create many points of low intensity, mistaking background counts as false surfaces. The background regularization does not affect the detected points significantly, but yields a better estimation of $\boldsymbol{B}$, leading to NMSE = 0.107 for $\alpha_B = 1$ and NMSE = 0.0912 for $\alpha_B = 2$. The number of true points detected without dilation and erosion moves or using only one scale decreases dramatically to 44% and 80%, respectively. Figure 7b compares the estimated intensity of the biggest plate with different values of σ^2. The NMSE obtained with $\sigma^2 = 0.6^2$ is 0.058, compared to 0.399 in the absence of correlation (i.e., when $\sigma^2 = 100^2$). Section SM12 shows the performance of ManiPoP for different SBRs and mean photons per pixel for this specific synthetic scene.

The second dataset was created with the "Art" scene from [40]. In order to have multiple surfaces per pixel, we added a semitransparent plane in front of the scene. We simulated the Lidar measurements, as if they were taken by the system described in [30]. The scene consists of $N_r = 183$, $N_c = 231$ pixels, and $T = 4500$ histogram bins. The bin width is $\Delta_b = 0.3$ mm and the pixel size is $\Delta_p \approx 1.2$ mm. In this complex scene, we compared the proposed method with the optimization algorithms SPISTA and ℓ_{21}+TV. SPISTA relies on the specification of a background level that was set to the true background value. It is important to note that this information is not available in real Lidar applications, as the background levels depend on the imaged scene. We also show the results for the regularization
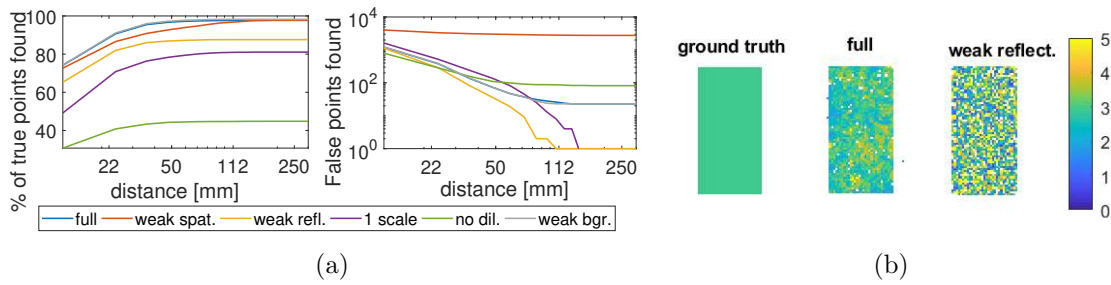
(a)   (b)

Figure 7. (a) *shows the percentage of true (left) and false (right) detections.* (b) *shows the intensity estimates for the vertical plate: Ground truth (left), estimates with $\sigma^2 = 0.6^2$ (center), and $\sigma^2 = 100^2$ (right).*

parameter that attained best results among many trials (the empirical rule for setting this parameter provided in [43] achieved worse results). We noticed that SPISTA provides large errors in the intensity estimates, as the gradient of the Poisson likelihood is not Lipschitz continuous and the gradient step iterations may diverge in very low photon scenarios [21, 11]. This problem can be solved by using the SPIRAL [21] inner loop to compute the step size (see section SM9 for details), yielding a new algorithm, which we name SPISTA+. The ℓ_{21}+TV algorithm has two regularization parameters that were adjusted in order to obtain the best results. It also relies on a thresholding step on the final estimates, as the output of the optimization method is not sparse. Again, the thresholding constant was adjusted to achieve the best results. To further improve the results of ℓ_{21}+TV, we included a grouping step, similar to the one described in [43], which reduces the number of false detections by pairing similar ones in the same pixel. Instead of taking the maximum intensity as in [43], we summed the intensities of the grouped detections, as this achieved better intensity estimates. Figure 8 shows the 3D point clouds obtained for each algorithm, whereas Figure 9 shows $F_{\text{true}}(\tau)$ and $F_{\text{false}}(\tau)$. SPISTA finds 18% of the true points and around 5033 false detections, whereas SPISTA+ improves the detection to 34% and 4267 false detections. ℓ_{21}+TV improves the detection rate to 57%, but also increases the false detections to 10^6. The grouping technique improves the results provided by ℓ_{21}+TV, reducing the false detections by a factor of 200. The proposed method obtains the best results, finding 92% of all the true points and 1852 false detections. As shown in Table 3, the proposed algorithm yields the best intensity estimates with the lowest execution time. Figure 10 shows the intensity estimate of the scene behind the semitransparent plane for each algorithm. SPISTA fails to provide meaningful intensity results, whereas SPISTA+ yields better estimates. As all the points behind the plane are grouped to yield a 2D intensity image, there is no difference between the ℓ_{21}+TV and ℓ_{21}+TV with grouping. Both SPISTA+ and ℓ_{21}+TV with grouping show a negative bias in the mean intensity, which may be attributed to the effect of the ℓ_1 and ℓ_{21} regularizations, respectively. As both SPISTA+ and ℓ_{21}+TV with grouping improve the results of the original algorithms in all the evaluated datasets, we show only their results in the rest of the experiments.

5.3. Real Lidar data. We assessed the proposed algorithm using three different Lidar datasets: the multilayered scene provided in [43, 1] recorded at the Massachusetts Institute of Technology, the polystyrene target imaged at Heriot-Watt University [5], and the camouflage scene from [19].
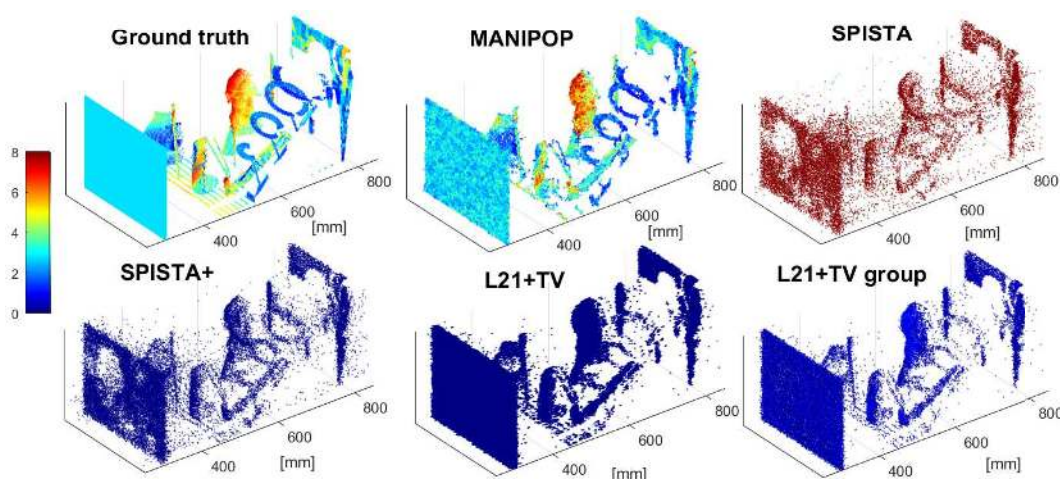
Figure 8. *Estimated 3D point cloud by the proposed algorithm, SPISTA, SPISTA+, $\ell_{21}+TV$, and $\ell_{21}+TV$ with grouping.*
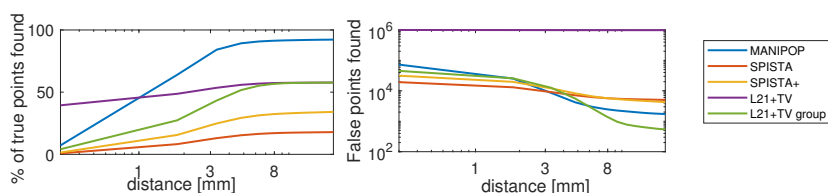

Figure 9. *Upper row: Percentage of true detections for different algorithms as a function of maximum distance τ, $F_{true}(\tau)$. Bottom row: Number of false detections, $F_{true}(\tau)$.*

Table 3
Performance of the proposed method, SPISTA, SPISTA+, $\ell_{21}+TV$, and $\ell_{21}+TV$ with grouping on the synthetic data.

Method	Total time [seconds]	NMSE intensity
SPISTA [43]	712	> 1
SPISTA+	8161	0.993
$\ell_{21}+TV$ [19]	2453	0.845
$\ell_{21}+TV$ group	2455	0.845
ManiPoP	**630**	**0.0999**
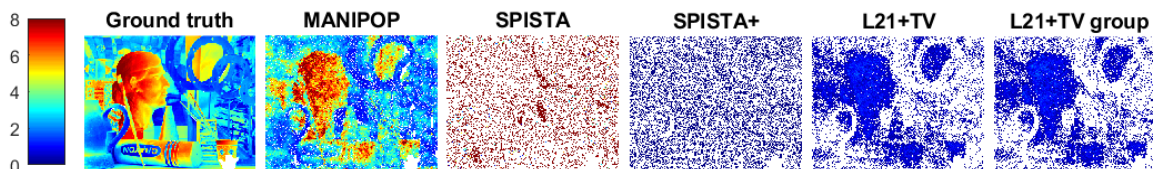
Figure 10. *Intensity estimates of the surfaces behind the semitransparent object.*
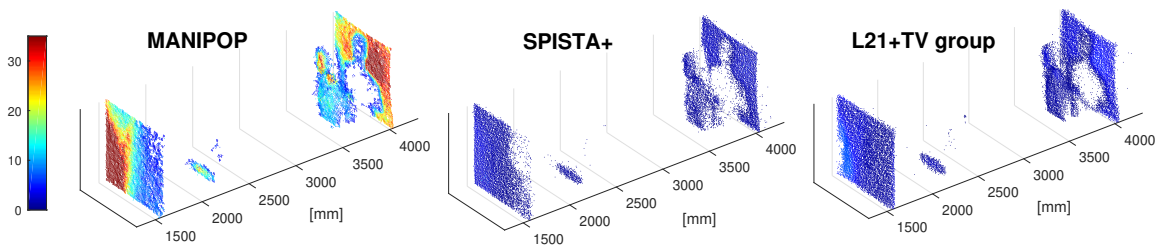
Figure 11. *Estimated 3D point cloud by ManiPoP, SPISTA, $\ell_{21}+TV$, and $\ell_{21}+TV$ with grouping.*
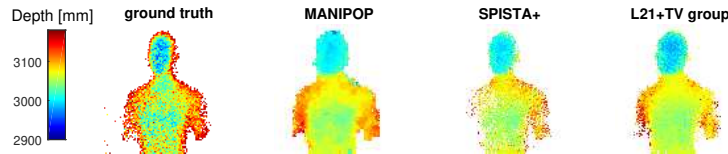
Figure 12. *Depth estimates of the mannequin. From left to right: Long acquisition reference, ManiPoP, SPISTA+, and $\ell_{21}+TV$ with grouping estimates.*

5.3.1. Mannequin behind a scattering object. The first scene consists of a mannequin located 4 meters behind a partially scattering object, with $N_r = N_c = 100$ pixels and $T = 4000$ bins. This Lidar scene is publicly available online [1]. The mean photon count per pixel is $\bar{\lambda}_p = 45$, and the dimensions are $\Delta_p \approx 8.4$ mm and $\Delta_b = 1.2$ mm. In [43], a Gaussian-shaped impulse response is suggested. However, we used a data-driven impulse response that yields better results (see section SM10 for a detailed explanation). Figure 11 shows the reconstructed point clouds for each algorithm. ManiPoP achieves a sparse and smooth solution, whereas the estimate of SPISTA presents more random scattering of points. The $\ell_{21}+TV$ output presents more spatial structure than SPISTA, but also fails to find the border of the mannequin. The dataset contains a reference depth of the mannequin obtained using a long acquisition time. This reference was computed using the log-matched filtering solution of a cropped Lidar cuboid where only the mannequin is present. Figure 12 shows the ground truth depth and the estimates obtained by ManiPoP, SPISTA+, and $\ell_{21}+TV$ with grouping. The proposed method outperforms the SPISTA+ and $\ell_{21}+TV$ outputs, finding 97.9% of the reference detections, whereas SPISTA+ only detects 74.8% and $\ell_{21}+TV$ with grouping finds 92.8%, as shown in Figure 13. The SPISTA+ and $\ell_{21}+TV$ with grouping algorithms detect 225 and 206 false points, respectively, compared to the 432 points found by ManiPoP. This increase in false detections can be attributed to the scattering object that was (probably) removed when the reference dataset was obtained. The scattering effect can be also seen in Figure 11, as it is possible to find some parts of the low intensity surface behind the mannequin. Despite not having a reference for reflectivity values of the target, we can say that the proposed method attains significantly better visual results, as shown in Figure 14. Both SPISTA+ and $\ell_{21}+TV$ with grouping underestimate the mean intensity. The total execution time of ManiPoP (146 seconds) was around 20 times less than SPISTA+ (2871 seconds) and slightly shorter than $\ell_{21}+TV$ with grouping (202 seconds).
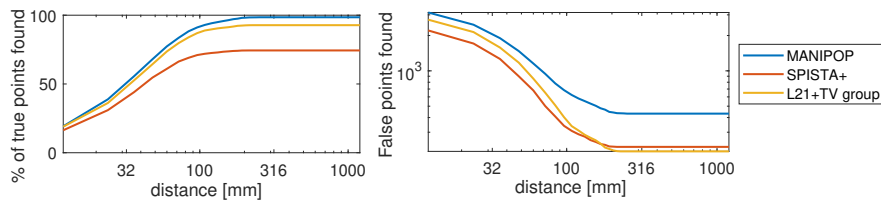
Figure 13. *Percentage of true detections at a maximum distance τ, $F_{true}(\tau)$, for ManiPoP, SPISTA+, and $\ell_{21}+TV$ with grouping. The number of false detections, $F_{false}(\tau)$, is shown in (b).*

Figure 14. *Estimated intensity by ManiPoP, SPISTA+, and $\ell_{21}+TV$ with grouping. The colorbar illustrates the number of photons assigned to each point. Both SPISTA+ and $\ell_{21}+TV$ show a negative bias in the mean intensity.*
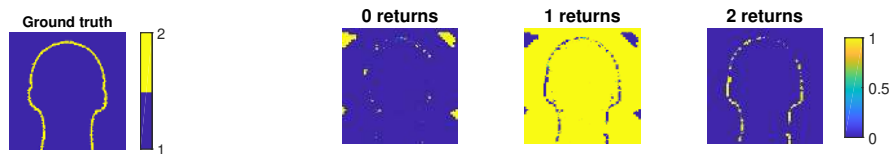

Figure 15. *From left to right: True number of surfaces per pixel and probability of having $k = 0, 1, 2$ objects per pixel for an acquisition time of 1 ms, respectively.*

5.3.2. Polystyrene head. The second dataset was obtained in Heriot-Watt University and consists of a life-sized polystyrene head at 40 meters from the imaging device (an image can be found in [5]). The data cuboid has size $N_r = N_c = 141$ pixels and $T = 4613$ bins. The physical dimensions are $\Delta_p \approx 2.1$ mm and $\Delta_{\text{bin}} = 0.3$ mm. A total acquisition time of 100 milliseconds was used for each pixel, yielding $\lambda_p = 337$ with approximately 23 background photons per pixel. The scene consists mainly of one object per pixel, only with two surfaces per pixel around the borders of the head. We compare the proposed method with the log-matched filtering solution and the SPISTA+ algorithm for different acquisition times, i.e., many values of λ_p. As no ground truth is available, we used as reference the log-matched filter solution, manually dividing the Lidar cube into segments with only one surface, using the largest acquisition time (100 ms). Although the dataset seems to have only one active depth per pixel, two surfaces per pixel can be found in the borders of the head, as shown in Figure 15. As only a few pixels contain two surfaces, we also compared with [38], which is a state-of-the-art 3D reconstruction algorithm under a single-surface-per-pixel assumption. Figure 16 shows the reconstructed 3D point clouds for an acquisition time of 1 ms, whereas Figure 17 shows $F_{\text{true}}(\tau)$ and $F_{\text{false}}(\tau)$ for acquisition times of 10, 1, and 0.2 ms. In the 10 and 1 ms cases, ManiPoP outperforms the other methods, finding almost all true points and

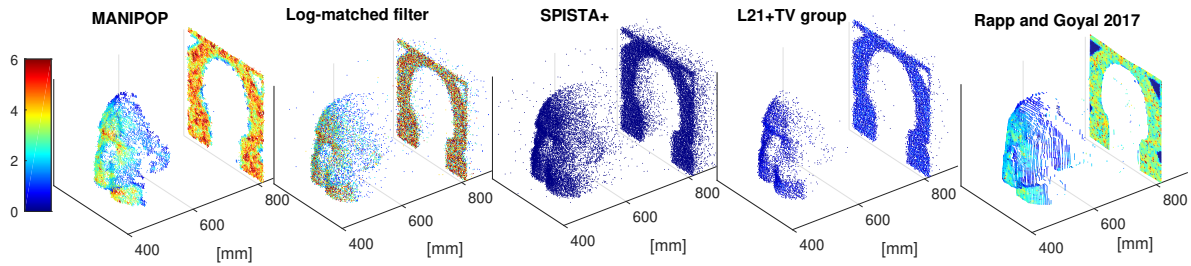**Figure 16.** *Estimated 3D point clouds using the polystyrene head dataset with an acquisition time of 1 ms. SPISTA+ and ℓ_{21}+TV underestimate the mean intensity, whereas ManiPoP, the log-matched filter solution, and [38] obtain a similar intensity mean.*
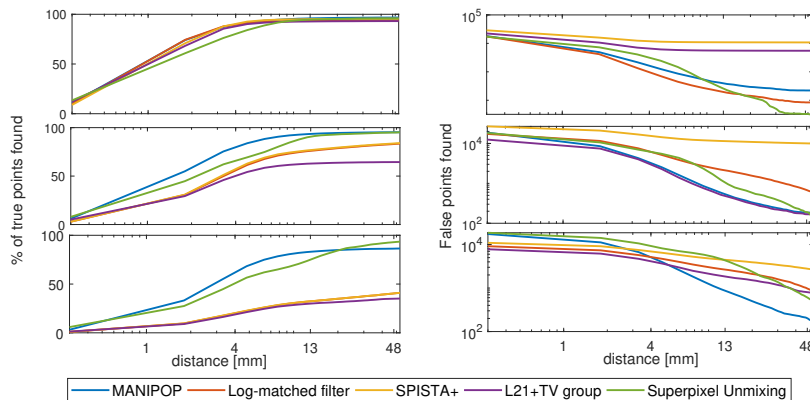

Figure 17. *$F_{true}(\tau)$ and $F_{false}(\tau)$ for the polystyrene head using acquisition times of 10 ms (top), 1 ms (middle), and 0.2 ms (bottom). While all methods obtain good reconstructions in the 10 ms case, ManiPoP and [38] also achieve good reconstructions with acquisition times of 1 ms and 0.2 ms.*

providing relatively few false estimates. The log-matched filter solution (of the complete Lidar cube) shows a significant error in depth estimates and fails to find 10% of true points, as it is only capable of finding one object per pixel. In the 0.2 ms case, there are only $\lambda_p = 0.7$ photons per pixel on average. Thus, the best performing algorithm is [38], as the single-surface assumption plays a fundamental role in inpainting the missing depth information. ManiPoP performs in second place, finding 14% fewer true points than [38].

As shown in Table 4, the fastest algorithm is the log-matched filtering solution with less than 20 seconds in all cases. However, ManiPoP still requires less computing time than SPISTA+ and ℓ_{21}+TV with grouping. It is worth noticing that the ℓ_{21}+TV algorithm has a memory requirement proportional to 6 times the whole data cube due to the ADMM algorithm, which can be prohibitively large when the Lidar cube is relatively big. The sparse nature of the ManiPoP algorithm only requires an amount of memory proportional to the number of bins with one photon or more plus the number of 3D points to infer.

To further demonstrate the generality of the proposed method, we studied the case where only one surface is present per pixel, but not all the pixels contain surfaces, which occurs in most outdoor measurements. If a single-surface-per-pixel algorithm is used, such as [25, 5, 42,

Table 4

Computing time of the proposed method, SPISTA+, ℓ_{21}+TV, log-matched filtering, and [38] on the polystyrene head dataset.

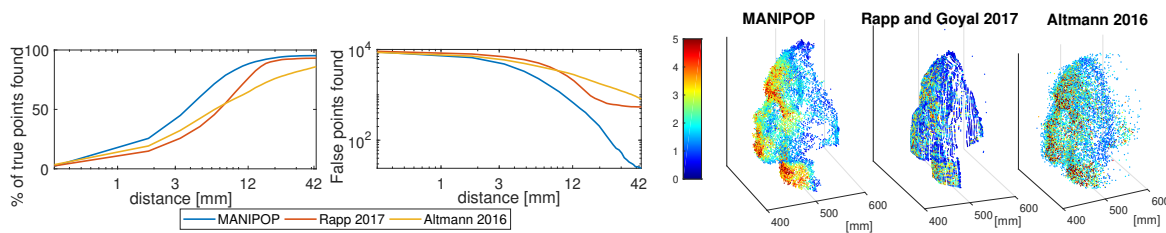| Algo./Acq. time	100 ms ($\lambda_p = 337$)	10 ms ($\lambda_p = 33.7$)	1 ms ($\lambda_p = 3.4$)	0.2 ms ($\lambda_p = 0.7$)
SPISTA+ [43]	6769	6981	7191	8461
ℓ_{21}+TV group [19]	793	697	705	535.4
ManiPoP	322	229	201	173.4
Log-matched filter	**18**	**11**	**7.8**	**5.6**
Rapp and Goyal 2017 [38]	196.87	40	37	38.4

Figure 18. $F_{true}(\tau)$ and $F_{false}(\tau)$ for the polystyrene head without backplane using an acquisition time of 1 ms are shown on the left. The 3D reconstructions are shown on the right.

17, 38], a nontrivial postprocessing step is necessary to discriminate which pixels have active depths. We also included the results obtained by the Bayesian target detection algorithm [6], which assumes at most one surface per pixel. To recreate this case using the polystyrene head dataset, we removed the backplane from the 1 ms dataset, obtaining a new 3D Lidar cube that only contains the polystyrene head. Figure 18 shows the results obtained using ManiPoP, [6], and [38]. In the latter, we applied a global thresholding based on the recovered reflectivity values, such that only the target would be present in the final results. The value of the threshold was manually chosen to obtain the best results. ManiPoP obtains the best results, finding 95.2% of the points with only 24 false detections, whereas [38] finds 93.1% of the points and 542 false detections and [6] obtains 86.0% of the points and 849 false detections. As shown in subsection 5.3.2, the estimates of [38] degrade significantly towards the borders of the target, as the single-surface assumption imposes a false correlation with the background photons in neighboring pixels where no surface is present. While [6] performs similarly to ManiPoP in terms of true and false point detections, the depth and reflectivity estimates are worse. This result can be attributed to the lack of prior spatial correlation for the depth and reflectivity values in [6].

Note that the samples generated by the proposed RJ-MCMC method are asymptotically distributed according to the posterior (2.17) and can thus be used to compute various uncertainty measures. For instance, Figure 15 shows the probability of having $k = 0, 1, 2$ peaks for an acquisition time of 1 ms, computed according to (3.6). Another example is displayed in Figure 19, which shows the position and log-intensity histograms that were computed using the samples from additional $N_i = 400 N_r N_c$ iterations in a fixed dimension (only allowing mark and shift moves).
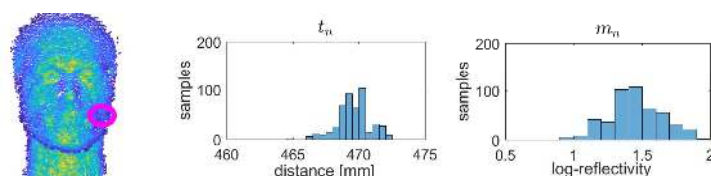
Figure 19. *The center and right plots show the position and log-intensity histograms for the point encircled in violet in the left plot, using an acquisition time of 1 ms.*
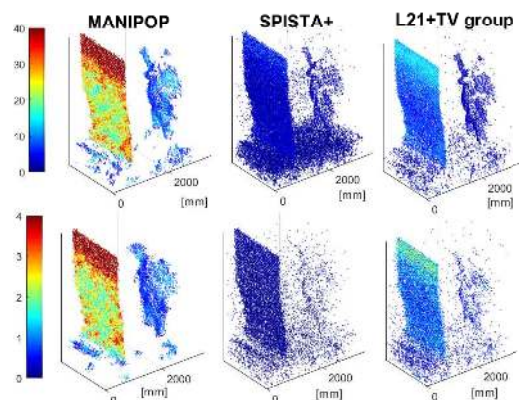

Figure 20. *Estimated 3D point clouds using the camouflage dataset for per-pixel acquisition times of 3.2 ms (top row) and 0.32 ms (bottom row).*

5.3.3. Human behind camouflage. The last dataset consists of a man standing behind camouflage at a stand-off distance of 230 meters from the Lidar system. An in-depth description of the scene can be found in [46, 19]. An acquisition time of 3.2 ms was used for each pixel, obtaining $\lambda_p = 44.6$ photons per pixel on average, where approximately 13.3 photons correspond to background levels. The Lidar cube has $N_r = 159$ and $N_c = 78$ pixels and $T = 550$ histogram bins. The physical dimensions are $\Delta_p \approx 2.1$ mm and $\Delta_{\text{bin}} = 5.6$ mm. We evaluated the performance of the algorithms for the per-pixel acquisition times of 3.2 ms and 0.32 ms. Figure 20 shows the reconstructions obtained by ManiPoP, SPISTA+, and TV+ℓ_{21} with depth grouping. In both cases, ManiPoP obtains a more structured reconstruction, without spurious detections and more dense reconstructions in the regions where the target is present.

6. Conclusions and future work. In this paper, we proposed a new Bayesian spatial point process model for describing single-photon depth images. This model promotes spatially correlated and sparse structures, which can be interpreted as a structured l_0 pseudo-norm regularization. From a compressive sensing viewpoint, structured sparsity priors can yield the lowest number of necessary measurements to reconstruct a signal [9]. Finding the MAP estimate of the proposed model is an NP-hard problem [34]. We overcame this problem by developing a stochastic RJ-MCMC algorithm with new moves that find a solution relatively fast. In addition, a multiresolution approach improved the estimates and reduced the execution time. The proposed method yielded good 3D reconstructions, with better depth

and intensity estimates. In our experiments, we noted that for each dataset, a different set of hyperparameters and thresholding values is needed for both SPISTA and ℓ_{21}+TV, thus making user supervision compulsory, whereas the proposed algorithm uses the same set of hyperparameters across all datasets. In extremely low photon cases, i.e., less than one photon per pixel on average, ManiPoP might fail to recover the surface, thus performing worse than other single-surface 3D reconstruction algorithms [38]. As shown in Figure SM9, the algorithm performs relatively well for an SBR higher than 1 and more than one photon per pixel. Excluding the aforementioned extremely low photon or extremely low SBR cases, ManiPoP generalizes other single-surface-per-pixel and target detection algorithms, as it can provide accurate estimates in scenes with only one surface per pixel and scenes where the target is present in a subset of pixels. The algorithm requires less execution time when compared to other optimization [43, 19] and RJ-MCMC approaches [23, 24]. Although there is a significant increase in computational time with respect to the classic log-matched filtering solution, a C++ implementation with efficient handling of the connected-surface structure would reduce the computing time considerably. A profiling analysis of the present code shows that around 70% of the total computational time is due to these computations. In addition, the Markovian structure of the algorithm could be further exploited to perform multiple parallel moves.

Scenes containing scattering media may present a broadening of the impulse response. Moreover, surfaces with normals that have a significant angle with respect to the laser beam might also show a broadening of $h(t)$. In such cases, the proposed method might have a reduced performance. Future work will be devoted to estimating the degree of broadening of each point. Moreover, the hard constraint on the minimum distance between two surfaces within a pixel (2.5) may not apply in some scenes, such as dense foliage or scenes with extremely close objects. In this setting, the hard constraint Strauss process should be modified for a soft constraint process [47]. Another important direction of future work is the extension of the proposed model to handle multispectral Lidar data, by considering jointly $L > 1$ bands and classifying the 3D point cloud according to different materials. Our model is easily extendible to this configuration, as we can add a mark to each point that labels the spectral signature of the object. We note that the presented model can be also used for single-photon imaging [25, 4]. Finally, we note that ManiPoP can be used as a first processing step to recover denoised 3D point clouds from raw Lidar data to then perform other higher-level computer vision tasks.

REFERENCES

[1] D. Shin, F. Xu, F. N. C. Wong, J. H. Shapiro, and V. K. Goyal, Code and data used for "Computational multi-depth single-photon imaging," https://github.com/photon-efficient-imaging/full-waveform (last accessed January 2019).

[2] Y. ALTMANN, R. ASPDEN, M. PADGETT, AND S. MCLAUGHLIN, *A Bayesian approach to denoising of single-photon binary images*, IEEE Trans. Comput. Imaging, 3 (2017), pp. 460–471.

[3] Y. ALTMANN, A. MACCARONE, A. MCCARTHY, G. NEWSTADT, G. BULLER, S. MCLAUGHLIN, AND A. HERO, *Robust spectral unmixing of sparse Lidar waveforms using gamma Markov random fields*, IEEE Trans. Comput. Imaging, 3 (2017), pp. 658–670.

[4] Y. ALTMANN, S. MCLAUGHLIN, AND M. PADGETT, *Unsupervised restoration of subsampled images constructed from geometric and binomial data*, in 2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), IEEE, 2018, pp. 1–5.

[5] Y. ALTMANN, X. REN, A. MCCARTHY, G. S. BULLER, AND S. MCLAUGHLIN, *Lidar waveform-based analysis of depth images constructed using sparse single-photon data*, IEEE Trans. Image Process., 25 (2016), pp. 1935–1946.

[6] Y. ALTMANN, X. REN, A. MCCARTHY, G. S. BULLER, AND S. MCLAUGHLIN, *Robust Bayesian target detection algorithm for depth imaging from sparse single-photon data*, IEEE Trans. Comput. Imaging, 2 (2016), pp. 456–467.

[7] L. AZZARI AND A. FOI, *Variance stabilization for noisy+estimate combination in iterative Poisson denoising*, IEEE Signal Process. Lett., 23 (2016), pp. 1086–1090.

[8] A. J. BADDELEY AND M. N. M. VAN LIESHOUT, *Area-interaction point processes*, Ann. Inst. Statist. Math., 47 (1995), pp. 601–619.

[9] R. G. BARANIUK, V. CEVHER, M. F. DUARTE, AND C. HEGDE, *Model-based compressive sensing*, IEEE Trans. Inform. Theory, 56 (2010), pp. 1982–2001.

[10] S. BROOKS, A. GELMAN, G. JONES, AND X.-L. MENG, *Handbook of Markov Chain Monte Carlo*, CRC Press, Boca Raton, FL, 2011.

[11] P. L. COMBETTES AND J.-C. PESQUET, *Proximal splitting methods in signal processing*, in Fixed-Point Algorithms for Inverse Problems in Science and Engineering, Springer, New York, 2011, pp. 185–212.

[12] O. DIKMEN AND A. T. CEMGIL, *Gamma Markov random fields for audio source modeling*, IEEE Trans. Audio Speech Language Process., 18 (2010), pp. 589–601.

[13] D. FERSTL, C. REINBACHER, R. RANFTL, M. RUETHER, AND H. BISCHOF, *Image guided depth upsampling using anisotropic total generalized variation*, in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2013.

[14] J. GAO, J. SUN, J. WEI, AND Q. WANG, *Research of underwater target detection using a slit streak tube imaging lidar*, in Proceedings of the Academic International Symposium on Optoelectronics and Microelectronics Technology (AISOMT), Harbin, China, 2012, pp. 240–243.

[15] P. J. GREEN, *Reversible jump Markov chain Monte Carlo computation and Bayesian model determination*, Biometrika, 82 (1995), pp. 711–732.

[16] T. HAKALA, J. SUOMALAINEN, S. KAASALAINEN, AND Y. CHEN, *Full waveform hyperspectral lidar for terrestrial laser scanning*, Opt. Express, 20 (2012), pp. 7119–7127.

[17] A. HALIMI, Y. ALTMANN, A. MCCARTHY, X. REN, R. TOBIN, G. S. BULLER, AND S. MCLAUGHLIN, *Restoration of intensity and depth images constructed using sparse single-photon data*, in Proceedings of the Signal Processing Conference (EUSIPCO), Budapest, Hungary, 2016, pp. 86–90.

[18] A. HALIMI, A. MACCARONE, A. MCCARTHY, S. MCLAUGHLIN, AND G. S. BULLER, *Object depth profile and reflectivity restoration from sparse single-photon data acquired in underwater environments*, IEEE Trans. Comput. Imaging, 3 (2017), pp. 472–484.

[19] A. HALIMI, R. TOBIN, A. MCCARTHY, S. MCLAUGHLIN, AND G. S. BULLER, *Restoration of multilayered single-photon 3d lidar images*, in Proceedings of the 2017 25th European Signal Processing Conference (EUSIPCO), 2017, pp. 708–712.

[20] J. M. HAMMERSLEY AND P. CLIFFORD, *Markov Fields on Finite Graphs and Lattices*, manuscript, 1971.

[21] Z. T. HARMANY, R. F. MARCIA, AND R. M. WILLETT, *This is SPIRAL-TAP: Sparse Poisson Intensity Reconstruction ALgorithms—Theory and Practice*, IEEE Trans. Image Process., 21 (2012), pp. 1084–1096.

[22] R. HARTLEY AND A. ZISSERMAN, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2003.

[23] S. HERNANDEZ-MARIN, A. M. WALLACE, AND G. J. GIBSON, *Bayesian analysis of Lidar signals with multiple returns*, IEEE Trans. Pattern Anal. Mach. Intell., 29 (2007), pp. 2170–2180.

[24] S. Hernandez-Marin, A. M. Wallace, and G. J. Gibson, *Multilayered 3D Lidar image construction using spatial models in a Bayesian framework*, IEEE Trans. Pattern Anal. Mach. Intell., 30 (2008), pp. 1028–1040.

[25] A. Kirmani, D. Venkatraman, D. Shin, A. Colaço, F. N. Wong, J. H. Shapiro, and V. K. Goyal, *First-photon imaging*, Science, 343 (2014), pp. 58–61.

[26] P. Lancaster and K. Salkauskas, *Surfaces generated by moving least squares methods*, Math. Comp., 37 (1981), pp. 141–158.

[27] A. Maccarone, A. McCarthy, X. Ren, R. E. Warburton, A. M. Wallace, J. Moffat, Y. Petillot, and G. S. Buller, *Underwater depth imaging using time-correlated single-photon counting*, Opt. Express, 23 (2015), pp. 33911–33926.

[28] C. Mallet and F. Bretar, *Full-waveform topographic lidar: State-of-the-art*, ISPRS J. Photogramm. Remote Sens., 64 (2009), pp. 1–16.

[29] C. Mallet, F. Lafarge, M. Roux, U. Soergel, F. Bretar, and C. Heipke, *A marked point process for modeling lidar waveforms*, IEEE Trans. Image Process., 19 (2010), pp. 3204–3221.

[30] A. McCarthy, R. J. Collins, N. J. Krichel, V. Fernández, A. M. Wallace, and G. S. Buller, *Long-range time-of-flight scanning sensor based on high-speed time-correlated single-photon counting*, Appl. Opt., 48 (2009), pp. 6241–6251.

[31] P. McCool, Y. Altmann, A. Perperidis, and S. McLaughlin, *Robust Markov random field outlier detection and removal in subsampled images*, in Proceedings of the Statistical Signal Processing Workshop (SSP), Palma de Mallorca, Spain, 2016, pp. 1–5.

[32] J. Møller and R. P. Waagepetersen, *Modern statistics for spatial point processes*, Scand. J. Statist., 34 (2007), pp. 643–684.

[33] I. Murray, Z. Ghahramani, and D. MacKay, *MCMC for Doubly-Intractable Distributions*, preprint, https://arxiv.org/abs/1206.6848, 2012.

[34] B. K. Natarajan, *Sparse approximate solutions to linear systems*, SIAM J. Comput., 24 (1995), pp. 227–234, https://doi.org/10.1137/S0097539792240406.

[35] M. Nilsson, *Estimation of tree heights and stand volume using an airborne lidar system*, Remote Sens. Environ., 56 (1996), pp. 1–7.

[36] T. Ogawa and K. Takagi, *Lane recognition using on-vehicle lidar*, in Proceedings of the Intelligent Vehicles Symposium, Tokyo, Japan, 2006, pp. 540–545.

[37] M. Pereyra, N. Whiteley, C. Andrieu, and J.-Y. Tourneret, *Maximum marginal likelihood estimation of the granularity coefficient of a Potts-Markov random field within an MCMC algorithm*, in Proceedings of the IEEE Workshop on Statistical Signal Processing (SSP), Gold Coast, Australia, 2014, pp. 121–124.

[38] J. Rapp and V. K. Goyal, *A few photons among many: Unmixing signal and noise for photon-efficient active imaging*, IEEE Trans. Comput. Imaging, 3 (2017), pp. 445–459.

[39] H. Rue and L. Held, *Gaussian Markov Random Fields: Theory and Applications*, CRC Press, Boca Raton, FL, 2005.

[40] D. Scharstein and C. Pal, *Learning conditional random fields for stereo*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07), 2007, pp. 1–8.

[41] B. Schwarz, *LIDAR: Mapping the world in 3D*, Nature Photonics, 4 (2010), pp. 429–430.

[42] D. Shin, A. Kirmani, V. K. Goyal, and J. H. Shapiro, *Photon-efficient computational 3-D and reflectivity imaging with single-photon detectors*, IEEE Trans. Comput. Imaging, 1 (2015), pp. 112–125.

[43] D. Shin, F. Xu, F. N. Wong, J. H. Shapiro, and V. K. Goyal, *Computational multi-depth single-photon imaging*, Opt. Express, 24 (2016), pp. 1873–1888.

[44] D. L. Snyder and M. I. Miller, *Random Point Processes in Time and Space*, Springer Science & Business Media, 2012.

[45] J. B. Tenenbaum, V. de Silva, and J. C. Langford, *A global geometric framework for nonlinear dimensionality reduction*, Science, 290 (2000), pp. 2319–2323.

[46] R. Tobin, A. Halimi, A. McCarthy, X. Ren, K. J. McEwan, S. McLaughlin, and G. S. Buller, *Long-range depth profiling of camouflaged targets using single-photon detection*, Opt. Eng., 57 (2017), 031303.

[47] M. N. M. van Lieshout, *Markov Point Processes and Their Applications*, World Scientific, 2000.

[48] A. M. Wallace, A. McCarthy, C. J. Nichol, X. Ren, S. Morak, D. Martinez-Ramirez, I. H. Woodhouse, and G. S. Buller, *Design and evaluation of LiDAR for the recovery of arboreal parameters*, IEEE Trans. Geosci. Remote Sens., 52 (2014), pp. 4942–4954.

[49] M. Zhou, H. Chen, L. Ren, G. Sapiro, L. Carin, and J. W. Paisley, *Non-parametric Bayesian dictionary learning for sparse image representations*, in Advances in Neural Information Processing Systems 22 (NIPS 2009), Curan Associates, 2009, pp. 2295–2303.

[50] M. Zhou, L. Hannah, D. B. Dunson, and L. Carin, *Beta-negative binomial process and Poisson factor analysis*, in Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics (AISTATS), Vol. 22, La Palma, Canary Islands, 2012, pp. 1462–1471.