Fast Multiscale 3D Reconstruction Using Single-Photon Lidar Data

FAST MULTISCALE 3D RECONSTRUCTION USING SINGLE-PHOTON LIDAR DATA

Sandor Plosz[1]*, Istvan Gyongy*[2]*, Jonathan Leach*[1]*,*
Steve McLaughlin[1]*, Gerald S. Buller*[1]*, Abderrahim Halimi*[1]

(1) School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, UK
(2) School of Engineering, University of Edinburgh, Edinburgh, UK

ABSTRACT

Time-correlated single-photon technology is emerging as an important approach to 3D Imaging. This paper presents a reconstruction algorithm that exploits data statistics and multiscale information to deliver clean depth and reflectivity images together with associated uncertainty maps. The statistical method has been implemented to run on graphics processing units (GPUs) that enable real-time reconstruction of moving scenes at more than 1000 depth frames per second on the 32×64 pixels real Quantic4x4 sensor data. Comparisons with state-of-the-art algorithms on simulated and real data demonstrate the robust and efficient performance of the proposed method.

Index Terms— 3D reconstruction, Lidar, real-time estimation, GPUs, parallel coding, Poisson noise.

1. INTRODUCTION

Single-photon LiDaR uses a high repetition pulsed laser source to illuminate the scene. The returned photons are then counted using a picosecond resolution single-photon avalanche diode (SPAD) detector and their time-of-flight (ToFs) measured with the time-correlated single photon counting (TCSPC) technique [1, 2]. The photons reflected from a target contain depth and reflectivity information allowing the 3D reconstruction of the scene.

Single-photon LiDaR has several key advantages including its excellent surface-to-surface resolution in time of flight (ToF) mode, and its high optical sensitivity. However, there are still limitations to allow full deployment of this technology, namely, (i) the reconstruction can be corrupted due to collecting insufficient photons when imaging at high frame rates, or too high level of background counts when imaging in bright or scattering environments [3–5]; (ii) existing reconstruction methods do not allow real-time video processing or lack uncertainty quantification as required by critical applications; (iii) the Lidar systems can output data in different formats, such as binary frames with (at most) one ToF value per pixel [5, 6], histograms of photon counts with respect to ToFs [1,2], or pre-estimated depth and reflectivity maps as for commercial Lidar systems [1].

These challenges have stimulated the design of multiple algorithms to process single-photon Lidar data [7–10]. These methods could be grouped into statistical based approaches exploiting known physical based data statistics [3, 4, 11–13], or learning based ones exploiting available databases to learn useful features [14–18]. Both strategies use spatial correlations and multiscale information as priors to regularise the estimated parameters, and improve robustness to noise or lack of information due to photon sparsity.

We propose a fast statistically based reconstruction algorithm for 3D Lidar data. This approach operates on either histogram of photon counts, or raw depth and reflectivity images provided by commercial systems. It comprises two steps. If histogram timing data is available, the first step pre-estimates noisy depth and reflectivity images using the maximum likelihood estimator. The second denoising step uses multi-scale information to reject outliers or to fill empty pixels. The proposed formulation involves independent updates, which have been implemented using CUDA C++ on GPUs (graphics processing unit). This enables the reconstruction of a 32×64 depth scene in less than 1 millisecond, which will help the deployment of single-photon LiDaR to real-world applications. The method is evaluated on simulated and real data confirming its benefits compared to existing state-of-the-art methods.

The paper is structured as follows. Section 2 introduces the Poisson-based observation model. Section 3 presents the proposed approximated multiscale model and its denoising algorithm. Section 4 describes the parallel implementation on GPUs. Results on simulated and real data are reported and analysed in Sections 5 and 6. Finally, conclusions and future work are presented in Section 7.

2. OBSERVATION MODEL

The time-tagged photon events are usually accumulated into a histogram of counts with respect to their ToF, denoted $y_{n,t}$ for the nth pixel and tth ToF. This counting data is assumed

This work was supported by the UK Royal Academy of Engineering under the Research Fellowship Scheme (RF/201718/17128), the DASA project (DSTLX1000147844); and EPSRC Grants EP/T00097X/1, EP/S026428/1.

[1] https://velodynelidar.com/

to follow a Poisson distribution $\mathcal{P}(.)$ as follows [13, 19, 20]:

$$y_{n,t} \sim \mathcal{P}\left(s_{n,t}\right), \quad s_{n,t} = r_n f_s\left(t - d_n\right) + b_{n,t} \tag{1}$$

where we assume the presence of at most one target per pixel with a depth d_n and a reflectivity r_n, f_s represents the system impulse response (SIR), and $b_{n,t}$ represents the background counts originating from the scattering environment or the detector's dark counts. This background can be non-uniform when imaging through obscurants [3], thus we assume it depends on t in our model. The joint likelihood is obtained by assuming independence of the different pixels

$$P(\boldsymbol{Y}|\boldsymbol{d}, \boldsymbol{r}, \boldsymbol{B}) = \prod_{n=1}^{N}\prod_{t=1}^{T} \frac{s_{n,t}^{y_{n,t}}}{y_{n,t}!} \exp\left(-s_{n,t}\right) \tag{2}$$

where $\boldsymbol{d}, \boldsymbol{r}, \boldsymbol{B}$ gather the depth, reflectivity and background values for all pixels, respectively. The goal is to estimate the depth and reflectivity parameters $d_n, r_n, \forall n$ while accounting for data statistics, for example using maximum likelihood (ML) estimators. However, there are two challenges: the ML estimates are poor for noisy real data, and the computational cost of the estimator algorithm needs to be minimal. To improve the parameter estimates, the above model is usually extended to incorporate multi-scale information, as used recently in algorithms [4, 13, 19, 21, 22]. This is performed by low-pass filtering the histogram of counts to obtain new histograms, from which the depth and reflectivity images present a lower level of noise at the cost of a reduced spatial resolution. However, this process is computationally expensive as it requires filtering large histogram data. To enable fast processing as required by real-world applications, recent approaches tend to approximate the likelihood term using simpler models, and to recover the lost information by regularizing the estimated parameters [4, 23]. This approach is adopted in this paper to implement a fast approximated multiscale model.

3. APPROXIMATED MULTISCALE MODEL FOR DENOISING

Building multiscale histograms requires filtering a large histogram cube with a computational cost which scales linearly with the large number of time bins $\mathcal{O}(N\log(N)T)$. This represents the computational bottleneck in multiscale based models [4, 18]. In addition, histograms of counts are not always available, as for commercial systems that pre-process the raw data to only output depth and reflectivity estimates (such as Velodyne or Kinect[2]). This highlights the importance of designing a robust method operating only on depth and reflectivity images to reduce the computational cost of the multiscale model and ensure generalisability to commercial systems, as described in the following section.

[2] https://learn.microsoft.com/en-us/azure/kinect-dk/depth-camera

3.1. Inferring multiscale depths from estimated maps

This section introduces an approximate model allowing the inference of the multiscale images $\hat{d}_n^{(\ell)}, \hat{r}_n^{(\ell)}$, for the scale $\ell \in \{1, \cdots, L\}$, in absence of histogram data, while only assuming the presence of the first scale of depth and reflectivity estimates $\hat{d}_n^{(\ell=1)}, \hat{r}_n^{(\ell=1)}$ from a pre-processing step. Akin to [4], assuming the absence of background and a Gaussian SIR $f\left(\mu - d_n\right) \approx \mathcal{N}(d_n; \mu, \sigma^2)$ in (2) simplifies the first scale likelihood to the following ($\ell = 1$ is omitted for brevity)

$$\begin{aligned} p\left(\boldsymbol{y}_n \mid r_n, d_n\right) \propto \mathcal{G}\left(r_n; 1 + \bar{y}_n, 1\right) Q\left(\boldsymbol{y}_n\right) \\ \times \mathcal{N}(d_n; \hat{d}_n, \bar{\sigma}_n^2), \end{aligned} \tag{3}$$

where $\hat{d}_n = \arg\max_d p\left(\boldsymbol{y}_n \mid r_n, d\right)$ denotes the maximum likelihood depth estimate, $\bar{\sigma}^2 := \sigma^2/\bar{y}_n$, Q is a parameter-free function only dependent on $\boldsymbol{y}_n$, and $\mathcal{G}(x\,; \cdot, \cdot)$ denotes the gamma distribution with shape and scale parameters. Therefore, in the case of a Gaussian SIR, this equation shows that the first scale maximum likelihood estimates of depth and reflectivity can be expressed as $d_n^{(\ell=1)} = \frac{\sum_{t=1}^{T} t y_{nt}}{\sum_{t=1}^{T} y_{nt}}$ and $r_n^{(\ell=1)} = \bar{y}_n$. Extending this analysis to L low-pass filtered versions of the histogram of counts, and assuming independence between the observations leads to

$$\begin{aligned} P(\boldsymbol{y}_n^{(\ell)}|\boldsymbol{r}_n^{(\ell)}, d_n^{(\ell)}) \quad &\propto \quad \mathcal{G}\left(r_n^{(\ell)}; 1 + \bar{y}_n^{(\ell)}, q_\ell^{-1}\right) \bar{Q}\left(\boldsymbol{y}_n^{(\ell)}\right) \\ &\times \quad \mathcal{N}\left(d_n^{(\ell)}; \hat{d}_n^{(\ell)}, \left(\bar{\sigma}^{(\ell)}\right)^2\right) \end{aligned} \tag{4}$$

$\forall \ell \in 1, \cdots, L$, where $\ell = 1$ is the original cube and $\ell > 1$ represents scales obtained by filtering with uniform kernels (e.g., $q_{\ell=2} = 3 \times 3, q_3 = 5 \times 5$, etc), $y_n^{(\ell)} = \sum_{n' \in \nu_n} y_n$, with ν_n representing the neighbours of the nth pixel. Comparing (4) and (3) shows that the multiscale depth and reflectivity estimates could be expressed using the original first scale estimates as follows: $\hat{d}_n^{(\ell)} = \frac{\sum_{t=1}^{T} t y_{nt}^{(\ell)}}{\sum_{t=1}^{T} y_{nt}^{(\ell)}} = \frac{\sum_{n' \in \nu_n} \hat{r}_{n'}^{(\ell=1)} \hat{d}_{n'}^{(\ell=1)}}{\sum_{n' \in \nu_n} \hat{r}_{n'}^{(\ell=1)}}$ and $r_n^{(\ell)} = \sum_{n' \in \nu_n} \hat{r}_{n'}^{(\ell=1)}/q_\ell$. Note that the depth estimates represent the mode of the Gaussian distribution in (4). In presence of outliers, it is better approximated by

$$\hat{d}_n^{(\ell)} = \arg\max_d \sum_{n'} \hat{r}_{n'}^{(\ell=1)} \log\left[f_s(\hat{d}_{n'}^{(\ell=1)} - d)\right]. \tag{5}$$

The reflectivity estimator represents the average photon counts received per pixel. This is justified in the absence of background, since all photons are reflected by a target. However, this estimator can be improved by only summing photons around the estimated depth $\hat{d}_n^{(\ell)}$ to reject outliers.

3.2. Denoising algorithm

The approximate multiscale model delivers L depth and reflectivity maps. These maps need to be combined efficiently

to associate single depth and reflectivity values per X-Y location or pixel. The proposed denoising process discourages outliers having few spatial neighbours and prioritizes values from lower scales as higher scales might lead to blurred depth maps. Formally, this is obtained by

$$\bar{d}_n = \frac{\sum_{n' \in \nu'_n} d_{n'}^{(\bar{\ell}_n)}}{\text{card}(\nu'_n)}, \tag{6}$$

with $\bar{\ell}_n = \min \ell$ subject to $\sum_{n' \in \nu_n} \delta\left(|d_n^{(\ell)} - d_{n'}^{(\ell)}| < \epsilon_d\right) \geq \sqrt{q_L}$, $\nu'_n = \left\{n' \in \nu_n, \text{such that } |d_n^{\bar{\ell}_n} - d_{n'}^{\bar{\ell}_n}| < \epsilon_d\right\}$, card(.) denoting the cardinality of a set, and the coefficient ϵ_d is easily fixed based on physical considerations related to the impulse response width and time bin resolution. This equation shows that the valid pixels with sufficient number of neighbours (e.g., those with more than $\sqrt{q_L}$ close depth values) are detected first, and the new depth is obtained as the average of the surrounding valid points.

Aiming for efficient reflectivity denoising, and akin to the bilateral filter [24], the proposed approach exploits the cleaned depth map and average the reflectivity of the good points as follows

$$\bar{r}_n = \frac{\sum_{n' \in \nu'_n} r_{n'}^{(\bar{\ell}_n)}}{\text{card}(\nu'_n)}. \tag{7}$$

Note however that other algorithms could be used to update reflectivity based on chosen priorities, including Poisson-based algorithms to deal with photon sparse regimes [25–27], or other state-of-the-art denoising algorithms [28, 29] for dense photon regimes. The proposed method allows depth uncertainty quantification which helps with decision making. Inspired from [4], the depth variance ϵ_n of the nth pixel could be approximated by

$$\hat{\epsilon}_n = \frac{\beta_d + \sum_{\ell, n' \in \nu_n} w_{n',n}^{(\ell)} |\bar{d}_n - d_{n'}^{(\ell)}|}{L + \text{card}(\nu_n) + \alpha_d + 1} \tag{8}$$

where α_d, β_d are small constants and $w_{n',n}^{(\ell)}$ are positive normalized weights given by

$$w_{n,n'}^{(\ell)} = \exp\left(-\frac{|d_n^{(\ell)} - \underline{d}_{n'}^{(\ell)}|}{2\epsilon_d}\right) \tag{9}$$

with ϵ_d a positive constant and $\underline{\boldsymbol{d}}$ contains the valid multiscale depth values after zeroing the outliers.

4. PARALLEL IMPLEMENTATION ON GRAPHICS PROCESSING UNIT

The proposed algorithm is modelled using a Data-flow Diagram (DFD) that depicts individual functional units, and the flow of data between them. A high-level overview is shown in Fig. 1. This model helps identifying possible parallel execution paths for maximum processing speed performance.

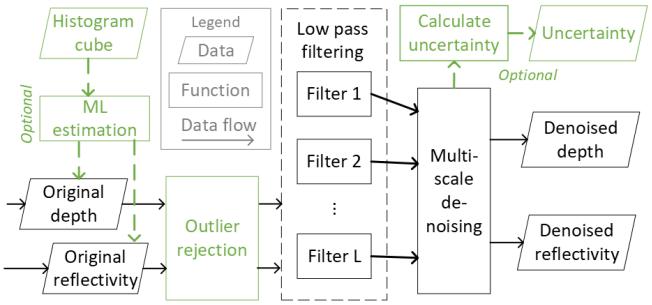

Fig. 1. Modular structure of the implementation

.

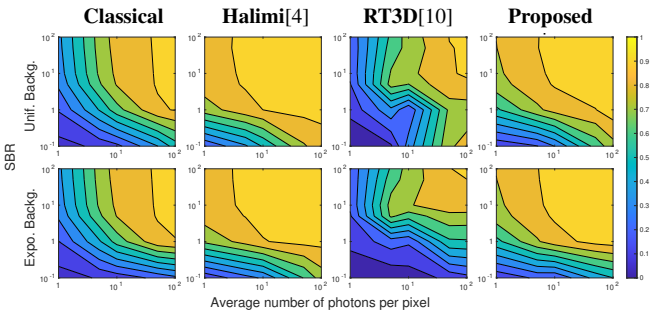

Fig. 2. Probability of good detection ($\tau = 0.45$ bins) for the investigated algorithms in case of (top) uniform and (bottom) exponential shaped background. Higher values are better.

The individual kernels for the application were implemented using the CUDA framework in C++. While multiple general parallel processing frameworks are available, we found that best computational performance are obtained with a native CUDA implementation [30]. More precisely, real-time performance was achieved by exploiting features of modern C++ such as compile-time polymorphism, statically initializing the required memory, transforming the data to be properly aligned in the device memory, and by using CUDA streams for parallel execution paths. We have also used the primitives of the CUB[3] library for thread-level cooperative operations. In addition, the cross-correlation used for depth estimation was computed in the frequency domain using the cuFFT library. In the denoiser, the filtering kernel is realized by first using 2D CUDA textures to get the neighbouring depth and reflectivity values for each pixel. This matrix is computed once for the largest filter size, and then exploited for smaller filter sizes in the multiscale case.

5. RESULTS ON SIMULATED DATA

This section evaluates the performance of the proposed algorithm on the simulated Reindeer scene from the Middlebury

[3] https://nvlabs.github.io/cub/

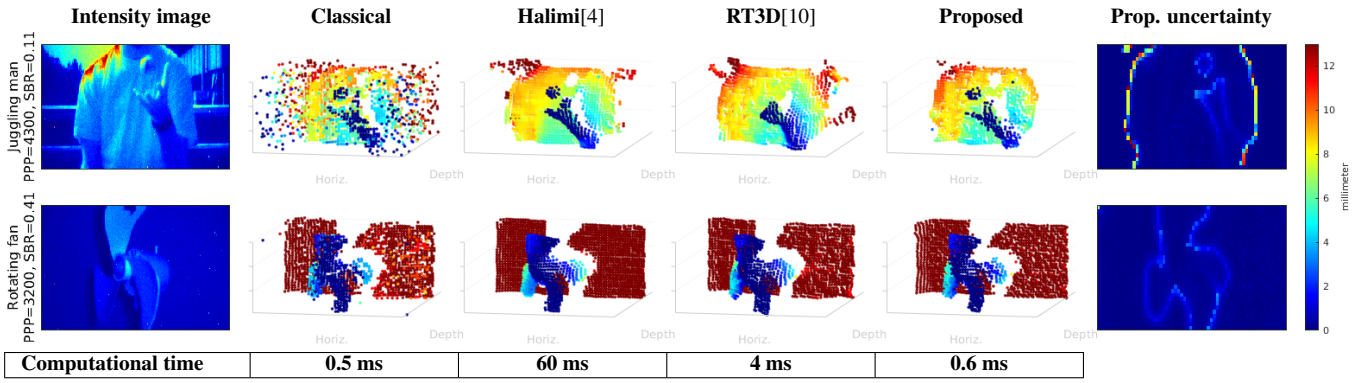

Fig. 3. 3D reconstructions of two frames from the Juggling and Fan scenes with different algorithms. The proposed algorithm only considers initial depth and reflectivity images as input while others use full histogram data.

Table 1. Computational time (CT) of the proposed reconstruction algorithm with respect to pixel sizes (for filters $[3, 7]$) and filter sizes (for 64×64 pixels).

Pixels	32^2	64^2	128^2	256^2	512^2
CT (ms)	0.4	0.9	3.1	10.3	37
Filters	3	$3, 5$	$3, 7, 9$	$3, 7, 11$	$3, 7, 13$
CT (ms)	0.4	0.6	2.1	2.4	3.9

dataset[4]. The data has been generated according to (1) with 139×168 pixels, $T = 20$ bins, Gaussian SIR with $\sigma = 0.6$, while varying the average photon per pixel (PPP) in [1, 100] and the signal to background ratio in [0.1, 100]. The proposed algorithm is compared with the classical ML, Halimi [4], and RT3D [10] algorithms. Both the proposed and Halimi used 2 scales of 5×5 and 7×7. Fig. 2 shows the probability of good detection obtained for the studied algorithms when varying SBR and PPP levels, where a point is considered as a good detection if a reference point exists within a threshold of $\tau = 0.45$ bins. This figure shows better results for all algorithms compared to classical ML even in presence of non-uniform background with decreasing exponential temporal shape. The computational cost of the proposed algorithm was also evaluated when varying the number of pixels and filter sizes for the proposed algorithm. Table 1 shows millisecond processing times with a 1000 frame per second (fps) for a size of 64×64 pixels, which will help deployment to real-world applications. Results on reflectivity were not included due to space limitations. Except for RT3D, all simulations were performed on a HP ZBook Studio G8 laptop with Intel i7-11850H CPU, 32GB RAM with an NVIDIA GeForce RTX 3070 GPU with 8GB memory. The available RT3D was pre-compiled on a windows machine, therefore we used a PC with Intel i5-6600 CPU, 24 GB RAM and a GeForce GTX 1060 6GB GPU under Windows 10.

[4]Available on: http://vision.middlebury.edu/stereo/data/

6. RESULTS ON REAL DATA

This section evaluates the proposed algorithms on real data acquired in Heriot-Watt University using the Quantic4x4 sensor [31]. Successive histograms of counts (32×64 pixels and $T = 16$ bins) were acquired at fast speed with 500 histogram per second. We consider two scenes, a fast rotating fan acquired in indoor conditions, and a juggling man acquired in bright outdoor conditions. Fig. 3 compares the results of the proposed algorithm with Classical, Halimi [4], and RT3D [10] algorithms. This highlights that the proposed algorithm provides detailed 3D reconstructions while preserving small features (e.g., see ball in Juggling data) and removing outliers. The processing time of these data is also depicted. Accordingly, the proposed algorithm processed the data in 0.6ms which is faster than the sensor's output of 500 fps. In addition, the proposed algorithm provides an estimate of the depth variance, which indicates higher uncertainty around object's edges or in bright conditions due to higher background levels.

7. CONCLUSIONS

This paper proposed a robust real-time reconstruction algorithm for 3D single-photon LiDaR data. The approach adopted an approximated multiscale model that enables robustness to false detections, while only using initial depth and reflectivity estimates, as commonly delivered by commercial systems. The estimation of the parameters and their uncertainty was implemented in parallel using graphics processing units, which led to a fast and robust algorithm showing real-time processing performance for the Quantic4x4 sensor with more than 1000 fps for the 32×64 pixels. Comparisons with state-of-the-art algorithms on simulated and real data highlight the benefit of the proposed strategy in terms of robustness and computational cost. Future work will generalize this approach to perform robust real-time guided super-resolution on depth maps.

8. REFERENCES

[1] A. M. Wallace, A. Halimi, and G. S. Buller, "Full waveform lidar for adverse weather conditions," *IEEE Trans. Vehicular Tech.*, vol. 69, no. 7, pp. 7064–7077, 2020.

[2] J. Rapp, J. Tachella, Y. Altmann, S. McLaughlin, and V. K Goyal, "Advances in single-photon lidar for autonomous vehicles: Working principles, challenges, and recent advances," *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 62–71, 2020.

[3] G. Satat, M. Tancik, and R. Raskar, "Towards photography through realistic fog," in *Computational Photography (ICCP), 2018 IEEE International Conference on*, 2018, pp. 1–10.

[4] A. Halimi, A. Maccarone, R. A. Lamb, G. S. Buller, and S. McLaughlin, "Robust and guided bayesian reconstruction of single-photon 3D lidar data: Application to multispectral and underwater imaging," *IEEE Trans. Comput. Imaging*, vol. 7, pp. 961–974, 2021.

[5] R. Tobin, A. Halimi, A. McCarthy, P. Soan, and G. S. Buller, "Robust real-time 3D imaging of moving scenes through atmospheric obscurant using single-photon LiDAR," *Scientific Reports*, vol. 11, pp. 11236, 2021.

[6] A. Maccarone, F. M. Della Rocca, A. McCarthy, R. Henderson, and G. S. Buller, "Three-dimensional imaging of stationary and moving targets in turbid underwater environments using a single-photon detector array," *Opt. Express*, vol. 27, no. 20, pp. 28437–28456, Sep 2019.

[7] D. Shin, F. Xu, D. Venkatraman, R. Lussana, F. Villa, F. Zappa, V. K. Goyal, Franco N. C. Wong, and J. H. Shapiro, "Photon-efficient imaging with a single-photon camera," *Nature Communications*, vol. 7, 2016.

[8] A. M. Pawlikowska, A. Halimi, R. A. Lamb, and G. S. Buller, "Single-photon three-dimensional imaging at up to 10 kilometers range," *Opt. Express*, vol. 25, no. 10, pp. 11919–11931, May 2017.

[9] Z-P. Li, X. Huang, Y. Cao, B. Wang, Y.-H. Li, W. Jin, C. Yu, J. Zhang, Q. Zhang, C.-Z. Peng, F. Xu, and J.-W. Pan, "Single-photon computational 3d imaging at 45 km," *Photon. Res.*, vol. 8, no. 9, pp. 1532–1540, Sep 2020.

[10] J. Tachella, Y. Altmann, N. Mellado, Aongus McCarthy, R. Tobin, G. S. Buller, J.-Y. Tourneret, and S. Journal McLaughlin, "Real-time 3d reconstruction from single-photon lidar data using plug-and-play point cloud denoisers," *Nature Communications*, vol. 10, no. 1, pp. 4984, 2019.

[11] 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.*, vol. 30, no. 6, pp. 1028–1040, June 2008.

[12] J. Tachella, Y. Altmann, M. Márquez, H. Arguello-Fuentes, J. Y. Tourneret, and S. McLaughlin, "Bayesian 3d reconstruction of subsampled multispectral single-photon lidar signals," *IEEE Trans. Comput. Imaging*, vol. 6, pp. 208–220, 2020.

[13] J. Rapp and V. K. Goyal, "A few photons among many: Unmixing signal and noise for photon-efficient active imaging," *IEEE Trans. Comput. Imaging*, vol. 3, no. 3, pp. 445–459, Sept. 2017.

[14] David B. Lindell, Matthew O'Toole, and Gordon Wetzstein, "Single-photon 3d imaging with deep sensor fusion," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 113:1–113:12, July 2018.

[15] Z. Sun, D. B. Lindell, O. Solgaard, and G. Wetzstein, "Spadnet: deep rgb-spad sensor fusion assisted by monocular depth estimation," *Opt. Express*, vol. 28, no. 10, pp. 14948–14962, May 2020.

[16] J. Peng, Z. Xiong, X. Huang, Z.-P. Li, D. Liu, and F. Xu, "Photon-efficient 3D imaging with a non-local neural network," in *Computer Vision – ECCV 2020*, Cham, 2020, pp. 225–241, Springer International Publishing.

[17] A. Ruget, S. McLaughlin, R. K. Henderson, II. Gyongy, A. Halimi, and J. Leach, "Robust super-resolution depth imaging via a multi-feature fusion deep network," *Opt. Express*, In press.

[18] J. Koo, A. Halimi, and S. McLaughlin, "A bayesian based deep unrolling algorithm for single-photon lidar systems," *Selected Topics in Signal Process., IEEE Journal of*, vol. 16, no. 4, pp. 762–774, 2022.

[19] A. Halimi, R. Tobin, A. McCarthy, J. Bioucas-Dias, S. McLaughlin, and G. S. Buller, "Robust restoration of sparse multidimensional single-photon lidar images," *IEEE Trans. Comput. Imaging*, vol. 6, pp. 138–152, 2020.

[20] 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.*, vol. 25, no. 5, pp. 1935–1946, Mar. 2015.

[21] W. Marais and R. Willett, "Proximal-gradient methods for poisson image reconstruction with bm3d-based regularization," in *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2017, pp. 183–187.

[22] J. Tachella, Y. Altmann, X. Ren, A. McCarthy, G. S. Buller, S. McLaughlin, and J.-Y. Tourneret, "Bayesian 3d reconstruction of complex scenes from single-photon lidar data," *SIAM Journal on Imaging Sciences*, vol. 12, no. 1, pp. 521–550, 2019.

[23] Q. Legros, J. Tachella, R. Tobin, A. Mccarthy, S. Meignen, G. S. Buller, Y. Altmann, S. Mclaughlin, and M. E. Davies, "Robust 3d reconstruction of dynamic scenes from single-photon lidar using beta-divergences," *IEEE Trans. Image Process.*, vol. 30, pp. 1716–1727, 2021.

[24] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, 1998, pp. 839–846.

[25] L. Azzari and A. Foi, "Variance stabilization for noisy+estimate combination in iterative poisson denoising," *IEEE Signal Processing Letters*, vol. 23, no. 8, pp. 1086–1090, 2016.

[26] J. Salmon, Z. Harmany, C.-A. Deledalle, and R. Willett, "Poisson noise reduction with non-local PCA," *Journal of Mathematical Imaging and Vision*, vol. 48, no. 2, pp. 279–294, 2014.

[27] 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 *Proc. EUSIPCO*, 2016, pp. 86–90.

[28] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug 2007.

[29] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[30] M. Khalilov and A. Timoveev, "Performance analysis of CUDA, OpenACC and OpenMP programming models on TESLA v100 GPU," *Journal of Physics: Conference Series*, vol. 1740, no. 1, pp. 012056, jan 2021.

[31] I. Gyongy, S. W. Hutchings, A. Halimi, M. Tyler, S. Chan, F. Zhu, S. McLaughlin, R. K. Henderson, and J. Leach, "High-speed 3d sensing via hybrid-mode imaging and guided upsampling," *Optica*, vol. 7, no. 10, pp. 1253–1260, Oct 2020.