Free-running vs. Synchronous: Single-Photon Lidar for High-flux 3D Imaging

Supplementary Material

A. Derivation of Likelihood Functions

A.1. Ideal Detector

The log likelihood for an ideal detector was previously derived in [2]. We include a derivation here for completeness. Without dead time, the detector is always armed, and detection times follow an inhomogeneous Poisson process with intensity function $\lambda(t)$ described in (1) in the main document. Let $(T_i)_{i=1}^N$ denote the sequence of absolute detection times. Then, following [11, eq. (2.33)], the log likelihood function is

$$\mathcal{L}^{\text{ideal}} = -\int_0^{n_r t_r} \lambda(t)\mathrm{d}t + \sum_{i=1}^N \log \lambda(T_i) \tag{1}$$

$$\overset{(a)}{=} -n_r \int_0^{t_r} \widetilde{\lambda}(t)\mathrm{d}t + \sum_{i=1}^N \log \widetilde{\lambda}(X_i) \tag{2}$$

$$= -n_r \Lambda + \sum_{i=1}^N \log \widetilde{\lambda}(X_i), \tag{3}$$

where step (a) follows from $\lambda(t) = \widetilde{\lambda}(t \bmod t_r)$, where $\widetilde{\lambda}(t)$ is the single-period intensity (2) in the main document. We remark that the log likelihood $\mathcal{L}^{\text{ideal}}$ is that of *absolute* detection times $(T_i)_{i=1}^N$, but it can be expressed in terms of *relative* detection times $\{X_i\}_{i=1}^N$.

A.2. Synchronous Detector

A synchronous detector detects only the first arriving photon in a repetition period. According to [11, eq. (2.20)], the probability density of the first detection time $X \in [0, t_r)$ is

$$p_X(x) = \widetilde{\lambda}(x) \exp\left(-\int_0^x \widetilde{\lambda}(t)\mathrm{d}t\right) \tag{4}$$

$$= \widetilde{\lambda}(x) \exp(-\widetilde{\Phi}(x)). \tag{5}$$

The probability that there is a photon detection in a given repetition period is

$$\mathbb{P}(0 \le X < t_r) = \int_0^{t_r} p_X(x) = 1 - e^{-\Lambda}. \tag{6}$$

Hence, the conditional probability density of X given that there is a detection is

$$p_{X|N}(x|1) = \frac{1}{1-e^{-\Lambda}} \widetilde{\lambda}(x) \exp(-\widetilde{\Phi}(x)). \tag{7}$$

The synchronous detector will be inactive in some repetition periods when there are carried-over dead times from previous repetition periods. Suppose there are $N_r' = n_r - \sum_{i=1}^N \mathbf{1}_{[t_r - t_d, t_r)}(X_i)$ active repetition periods. Since the probability of detecting a photon in a given repetition period is $1 - e^{-\Lambda}$, the number of detections N follows a binomial distribution:

$$N \sim \text{Binomial}(N_r', 1 - e^{-\Lambda}). \tag{8}$$

Since the detection times are independent and identically distributed, the probability density of all detection times is

$$p_{X_1,\dots,X_N,N|N_r'}(x_1,\dots,x_n,n|n_r') \tag{9}$$

$$= p_{N|N_r'}(n|n_r') p_{X_1,\dots,X_N|N,N_r'}(x_1,\dots,x_n|n,n_r') \tag{10}$$

$$= p_{N|N_r'}(n|n_r') \prod_{i=1}^n p_{X_i|N,N_r'}(x_i|n,n_r') \tag{11}$$

$$= \binom{N_r'}{n} (1-e^{-\Lambda})^n e^{-(N_r'-n)\Lambda} \prod_{i=1}^n \frac{1}{1-e^{-\Lambda}} \widetilde{\lambda}(x_i) \exp(-\widetilde{\Phi}(x_i)) \tag{12}$$

$$= \binom{N_r'}{n} e^{-(N_r'-n)\Lambda} \prod_{i=1}^n \widetilde{\lambda}(x_i) \exp(-\widetilde{\Phi}(x_i)). \tag{13}$$

The log likelihood conditioned on N_r' is therefore

$$\mathcal{L}^{\text{sync}} = -(N_r' - N)\Lambda + \sum_{i=1}^N \log \widetilde{\lambda}(X_i) - \widetilde{\Phi}(X_i) + \text{ const.} \tag{14}$$

The constant does not depend on the parameters of interest (S, B, and z), so it is dropped for the sake of ML estimation.

A.3. Free-running Detector

The free-running detector is unarmed for t_d whenever it detects a photon. The absolute detection times follow a self-exciting point process (SEPP) whose intensity becomes zero during dead times [11]. We denote the SEPP by $\{N(t), t \ge 0\}$, where $N(t)$ is the number of detections up to time t. The SEPP's intensity, which itself is a stochastic process, is

$$\mu(t) = \begin{cases} \lambda(t), & \text{if } t > T_{N(t)} + t_d, \\ 0, & \text{if } T_{N(t)} < t \le T_{N(t)} + t_d, \end{cases} \tag{15}$$

where $T_{N(t)}$ is the most recent detection time at time t. According to [11, Theorem 6.2.2], the log likelihood of the *absolute* detection times $(T_i)_{i=1}^N$ over the acquisition period $[0, n_r t_r)$ is

$$\mathcal{L}^{\text{free}} = -\int_0^{n_r t_r} \mu(t)\mathrm{d}t + \sum_{i=1}^N \log \lambda(T_i) \tag{16}$$

$$= -\int_0^{n_r t_r} \mu(t)\mathrm{d}t + \sum_{i=1}^{N} \log \widetilde{\lambda}(X_i). \tag{17}$$

We approximate the integral:

$$\begin{aligned}\int_0^{n_r t_r} \mu(t)\mathrm{d}t &= \int_0^{T_1} \lambda(t)\mathrm{d}t + \sum_{i=1}^{N-1} \int_{T_i+t_d}^{T_{i+1}} \lambda(t)\mathrm{d}t \\ &\quad + \mathbf{1}_{[0,n_r t_r)}(T_N + t_d) \int_{T_N+t_d}^{n_r t_r} \lambda(t)\mathrm{d}t \end{aligned} \tag{18}$$

$$\begin{aligned}&= \Phi(T_1) + \sum_{i=1}^{N-1} \left(\Phi(T_{i+1}) - \Phi(T_i + t_d)\right) \\ &\quad + \mathbf{1}_{[0,n_r t_r)}(T_N + t_d)\left(\Phi(n_r t_r) - \Phi(T_N + t_d)\right)\end{aligned} \tag{19}$$

$$\begin{aligned}&\overset{(a)}{\approx} \Phi(T_1) + \sum_{i=1}^{N-1} \left(\Phi(T_{i+1}) - \Phi(T_i + t_d)\right) \\ &\quad + \left(\Phi(n_r t_r) - \Phi(T_N + t_d)\right)\end{aligned} \tag{20}$$

$$= n_r\Lambda + \sum_{i=1}^{N} \Phi(T_i) - \Phi(T_i + t_d) \tag{21}$$

$$\begin{aligned}&\overset{(b)}{=} n_r\Lambda + \sum_{i=1}^{N} \Phi(T_i \bmod t_r) + \lfloor T_i/t_r \rfloor \Lambda \\ &\quad - \Phi(T_i \bmod t_r + t_d) - \lfloor T_i/t_r \rfloor \Lambda\end{aligned} \tag{22}$$

$$= n_r\Lambda + \sum_{i=1}^{N} \Phi(X_i) - \Phi(X_i + t_d). \tag{23}$$

In step (a), we replace the indicator function $\mathbf{1}_{[0,n_r t_r)}(T_N + t_d)$ with one. If the number of detections N is large, then the approximation will have a relatively small effect on the integral. Moreover, if the detector is active at $t = n_r t_r$, the indicator function takes the value one, and the approximation becomes an equality. Step (b) follows from $T_i = \lfloor T_i/t_r \rfloor t_r + T_i \bmod t_r$ and $\Phi(nt_r) = n\Lambda$ for any $n \in \{0, \ldots, n_r\}$. Substituting (23) into (17) completes the derivation:

$$\mathcal{L}^{\text{free}} \approx -n_r\Lambda + \sum_{i=1}^{N} \log \widetilde{\lambda}(X_i) + \Phi(X_i + t_d) - \Phi(X_i). \tag{24}$$

To write the log likelihood more explicitly in terms of the unknown parameters, we remark that $\Phi(X_i) = \widetilde{\Phi}(X_i)$, where $\widetilde{\Phi}(t) = SF(t - 2z/c) + bt$ is the single-period cumulative flux defined in Section 3.4. For the other term, $\Phi(X_i + t_d) = \widetilde{\Phi}((X_i + t_d) \bmod t_r) + \mathbf{1}_{[t_r,\infty)}(X_i + t_d)\Lambda$, which can then be written in terms of S, B, and z.

B. Derivation of ML Depth Estimators

B.1. Ideal Detector

The ML depth estimator has been previously shown to be a log matched filter [2]. We include a derivation of the filter here for completeness. We rewrite the ideal log likelihood as

$$\mathcal{L}^{\text{ideal}} = -n_r\Lambda + \sum_{i=1}^{N} \log\left(Sf(X_i - \tau) + b\right), \tag{25}$$

where $\tau = 2z/c$. Maximizing $\mathcal{L}^{\text{ideal}}$ results in

$$\widehat{\tau} = \arg\max_{\tau} \sum_{i=1}^{N} \log\left(Sf(X_i - \tau) + b\right) \tag{26}$$

$$= \arg\max_{\tau} \sum_{i=1}^{N} \log(Sf(\tau) + b) \oplus \delta(\tau - X_i) \tag{27}$$

$$= \arg\max_{\tau} \log(Sf(\tau) + b) \oplus \sum_{i=1}^{N} \delta(\tau - X_i) \tag{28}$$

$$= \arg\max_{\tau} w(\tau) \oplus h(\tau), \tag{29}$$

where $\oplus$ denotes correlation, the term

$$w(t) := \log(Sf(\tau) + b) \tag{30}$$

is the matched filter, and $h(t)$ is defined in (8) in the main paper.

B.2. Synchronous Detector

We find the time-of-flight τ maximizing $\mathcal{L}^{\text{sync}}$:

$$\widehat{\tau} = \arg\max_{\tau} \sum_{i=1}^{N} \log \widetilde{\lambda}(X_i) - \widetilde{\Phi}(X_i) \tag{31}$$

$$\begin{aligned}&= \arg\max_{\tau} \sum_{i=1}^{N} \log(Sf(X_i - \tau) + b) \\ &\qquad - \left(SF(X_i - \tau) + bX_i\right)\end{aligned} \tag{32}$$

$$= \arg\max_{\tau} \sum_{i=1}^{N} u(\tau) \oplus \delta(\tau - X_i) \tag{33}$$

$$= \arg\max_{\tau} u(\tau) \oplus h(\tau). \tag{34}$$

We recall that the filter $u(t)$ is

$$u(t) = \log(Sf(\tau) + b) - SF(t). \tag{35}$$

B.3. Free-running Detector

We first rewrite the approximate free-running log likelihood (24) as

$$\mathcal{L}^{\text{free}} \approx -n_r\Lambda + \sum_{i=1}^{N} \widetilde{\lambda}(X_i) + \Big(\widetilde{\Phi}((X_i+t_d) \bmod t_r) + \mathbf{1}_{[t_r,\infty)}(X_i+t_d)\Lambda\Big) - \widetilde{\Phi}(X_i). \tag{36}$$

The terms $-n_r\Lambda$ and $\mathbf{1}_{[t_r,\infty)}(X_i+t_d)\Lambda$ do not depend on τ, so they can be dropped for ML depth estimation. To make the equations concise, we define $Y_i := (X_i+t_d) \bmod t_r$. We maximize this approximate log likelihood with respect to τ as follow,

$$\widehat{\tau} = \arg\max_{\tau} \sum_{i=1}^{N} \log \widetilde{\lambda}(X_i) + \widetilde{\Phi}(Y_i) - \widetilde{\Phi}(X_i) \tag{37}$$

$$= \arg\max_{\tau} \sum_{i=1}^{N} \log(Sf(X_i-\tau)+b) + SF(Y_i-\tau) + bY_i - SF(X_i-\tau) - bX_i \tag{38}$$

$$\overset{(a)}{=} \arg\max_{\tau} \sum_{i=1}^{N} u(\tau) \oplus \delta(\tau - X_i) + v(\tau) \oplus \delta(\tau - Y_i) \tag{39}$$

$$= \arg\max_{\tau} u(\tau) \oplus h(\tau) + v(\tau) \oplus g(\tau), \tag{40}$$

where the filter $v(t)$ is

$$v(t) = SF(t). \tag{41}$$

B.4. Quantization

In practical SPL systems, the detection times are quantized to a predefined temporal resolution, and the data are stored as a histogram. Even in simulation, an efficient implementation of the ML depth estimators uses discrete-time correlation, which requires quantized detection times. We will show that the histogram is equivalent to $h(t)$ defined in (8) of the main document.

Suppose the period $[0, t_r)$ is partitioned into M histogram bins $\{[m\Delta, (m+1)\Delta)\}_{m=0}^{M-1}$, where the bin size is $\Delta = t_r/M$. Let us consider a set of relative detection times $\{X_i\}_{i=1}^{N}$. Each detection time is assigned to a bin center:

$$X_i \in \left\{(m+1/2)\Delta \middle| m \in \{0,\dots,M-1\}\right\}. \tag{42}$$

We can express $\{X_i\}_{i=1}^{N}$ in terms of the histogram:

$$h_\Delta[m] = \sum_{i=1}^{N} \mathbf{1}_{\{(m+1/2)\Delta\}}(X_i). \tag{43}$$

Now, let us consider $h(t)$ in the main paper defined as

$$h(t) = \sum_{i=1}^{N} \delta(t - X_i). \tag{44}$$

If the detection times are discretized, then

$$h(t) = \sum_{m=1}^{M-1} \delta(t-(m+1/2)\Delta) \sum_{i=1}^{N} \mathbf{1}_{\{(m+1/2)\Delta\}}(X_i) \tag{45}$$

$$= \sum_{m=1}^{M-1} \delta(t-(m+1/2)\Delta) h_\Delta[m]. \tag{46}$$

At time $t = (m'+1/2)\Delta$ for some $m' \in \{0,\dots,M\}$, which is the m'-th bin center of the histogram $h_\Delta[m]$, we have

$$h(t) = \delta(t-(m'+1/2)\Delta) h_\Delta[m']. \tag{47}$$

This shows that there is a one-to-one correspondence between $h(t)$ and the histogram $h_\Delta[m]$ when $\{X_i\}_{i=1}^{N}$ is quantized according to (42).

Even if $\{X_i\}_{i=1}^{N}$ is not quantized, we need to compute the quantized histogram $h_\Delta[m]$ with a bin size Δ in order to implement the depth estimators efficiently as discrete-time correlation. For example, the ML estimator of τ for an ideal detector is implemented as

$$\widehat{\tau} = (m^* + 1/2)\Delta \tag{48}$$

$$\text{where } m^* = \arg\max h_\Delta[m] \oplus w_\Delta[m] \tag{49}$$

$$= \arg\max \sum_{m'=-\infty}^{\infty} h_\Delta[m'] w_\Delta[m'-m], \tag{50}$$

and $w_\Delta[m] = w((m+1/2)\Delta)$ is the discrete-time filter. In simulation where we have access to $\{X_i\}_{i=1}^{N}$ with floating-point precision, we may further improve τ estimate by maximizing the log likelihood using gradient-based methods.

C. Initializing the ML Estimator

We initialize the alternating maximization algorithm (1) for ML estimation using the following estimators. For free-running measurements, we use the estimators for ideal measurements described below in Suppl. C.1.

C.1. Ideal Detector

We use the traditional log matched filtering method [11] to obtain an initial time-of-flight estimate:

$$\widehat{\tau}_{\text{init}}^{\text{ideal}} = \arg\max_{\tau} h(t) \oplus \log f(\tau). \tag{51}$$

Inspired by the Neyman-Pearson censoring estimator [9], we assume that the detection times in the window of size

t_{win} around $\widehat{\tau}_{\text{init}}$ are mostly due to the signal. In this paper, we set $t_{\text{win}} = 4w$, where w is the pulse width. Let N_{cl} denote the number of detections in this window, *i.e.*, $N_{\text{cl}} = \sum_{i=1}^{N} \mathbf{1}_{[\widehat{\tau}_{\text{init}} - t_{\text{win}}/2, \widehat{\tau}_{\text{init}} + t_{\text{win}}/2]}(X_i)$. Then, the signal flux initial estimate is

$$\widehat{S}_{\text{init}}^{\text{ideal}} = \max\left(\frac{N_{\text{cl}}}{n_r}, \epsilon\right), \tag{52}$$

where we lower bound the signal flux estimate by a small positive number $\epsilon > 0$ to ensure that arguments of the log terms in the log likelilhoods are positive. In our simulations and experiments, we use $\epsilon = 10^{-5}$. We estimate the background flux as

$$\widehat{B}_{\text{init}}^{\text{ideal}} = \max\left(\frac{N}{n_r} - \widehat{S}_{\text{init}}, \epsilon\right), \tag{53}$$

since N/n_r is an estimate of the total flux Λ, and $\Lambda = S + B$.

C.2. Synchronous Detector

We first use Coates's correction to estimate the photon arrival intensity [4]. Suppose the detection times $\{X_i\}_{i=1}^{N}$ are quantized into a histogram $h_\Delta[m]$ with bin size Δ. The Coates-corrected histogram, which is the ML estimate of the photon arrival intensity [5], for the m^{th} time bin is

$$\widehat{\lambda}[m] = \log\left(\frac{N - \sum_{m'=1}^{m-1} h_\Delta[m']}{N - \sum_{m'=1}^{m} h_\Delta[m']}\right). \tag{54}$$

Then, we apply the log-matched filter [11] to estimate the time-of-flight:

$$\widehat{\tau}_{\text{init}}^{\text{sync}} = (m^* + 1/2)\Delta, \tag{55}$$

$$\text{where } m^* = \arg\max \widehat{\lambda}[m] \oplus s[m] \tag{56}$$

$$s[m] = \log f((m + 1/2)\Delta), \tag{57}$$

where $f(t)$ is the pulse's temporal profile. Again, the signal and background flux estimates follow from censoring of the estimated photon arrival intensity:

$$\widehat{S}_{\text{init}}^{\text{sync}} = \max\left(\sum_{k \in \mathcal{K}_{\text{win}}} \widehat{\lambda}[k], \epsilon\right), \tag{58}$$

$$\widehat{B}_{\text{init}}^{\text{sync}} = \max\left(\sum_{k=1}^{N} \widehat{\lambda}[k] - \widehat{S}_{\text{init}}^{\text{sync}}, \epsilon\right). \tag{59}$$

D. Depth Score from Point Cloud Score

In the SSDR algorithm, the depth score $\sigma_p \in \mathbb{R}$ is obtained by projecting the point cloud score $\boldsymbol{s}_p \in \mathbb{R}^3$ onto the detector's line of sight, which is determined by the scan angles (θ_p, ϕ_p). The trained model's point cloud score approximates the Stein score, i.e., the gradient of the log density of the distribution of 3D points [7]. In this section, we show that σ_p relates to the Stein score of the depth distribution corresponding to the implicit prior on 3D points in the trained model.

D.1. From Cartesian to Spherical

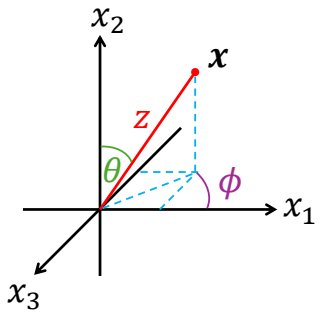


Figure 1. Relationship between Cartesian coordinate $\boldsymbol{x} = (x_1, x_2, x_3)$ and spherical coordinate $\widetilde{\boldsymbol{x}} = (z, \theta, \phi)$.

We first describe the transformation between the Cartesian coordinate $\boldsymbol{x} = (x_1, x_2, x_3)$ and the spherical coordinate $\widetilde{\boldsymbol{x}} = (z, \theta, \phi)$, where z is the depth from the detector, assumed to be at the origin. According to Fig. 1,

$$x_1 = z \sin\theta \cos\phi, \tag{60}$$

$$x_2 = z \cos\theta, \tag{61}$$

$$x_3 = -z \sin\theta \sin\phi. \tag{62}$$

We denote this transformation by T, *i.e.*, $\widetilde{\boldsymbol{x}} = T(\boldsymbol{x})$, and the inverse map by T^{-1}.

D.2. Spherical Coordinate Prior Distribution

Consider a prior distribution of a 3D point in the Cartesian coordinate with density $\pi(\boldsymbol{x})$, representing the implicit prior in the trained point cloud score model for a single point conditioned on all others. Since $\widetilde{\boldsymbol{x}} = T(\boldsymbol{x})$ and T is bijective, we can compute the density of the distribution of $\widetilde{\boldsymbol{x}}$, denoted by $\widetilde{\pi}(\widetilde{\boldsymbol{x}})$, using the change-of-variable formula:

$$\widetilde{\pi}(\widetilde{\boldsymbol{x}}) = \pi(T^{-1}(\widetilde{\boldsymbol{x}}))|\det J_{T^{-1}}(\widetilde{\boldsymbol{x}})|, \tag{63}$$

where $J_{T^{-1}}$ is the Jacobian of T^{-1}. We now compute the Jacobian:

$$J_{T^{-1}}(\widetilde{\boldsymbol{x}}) = \begin{bmatrix} \frac{\partial x_1}{\partial z} & \frac{\partial x_1}{\partial \theta} & \frac{\partial x_1}{\partial \phi} \\ \frac{\partial x_2}{\partial z} & \frac{\partial x_2}{\partial \theta} & \frac{\partial x_2}{\partial \phi} \\ \frac{\partial x_3}{\partial z} & \frac{\partial x_3}{\partial \theta} & \frac{\partial x_3}{\partial \phi} \end{bmatrix} \tag{64}$$

$$= \begin{bmatrix} \sin\theta\cos\phi & z\cos\theta\cos\phi & -z\sin\theta\sin\phi \\ \cos\theta & -z\sin\theta & 0 \\ -\sin\theta\sin\phi & -z\cos\theta\sin\phi & -z\sin\theta\cos\phi \end{bmatrix}. \tag{65}$$

The determinant is

$$\det J_{T^{-1}}(\widetilde{\boldsymbol{x}}) = z^2 \sin\theta. \tag{66}$$

Therefore, the density of the point in spherical coordinate is

$$\widetilde{\pi}(z, \theta, \phi) = \pi(x_1, x_2, x_3) \cdot z^2 \sin\theta. \tag{67}$$

D.3. Deriving the Depth Score

At pixel p, the detector's line of sight is defined by the scan angles (θ_p, ϕ_p). Given a point cloud score in Cartesian coordinate $\boldsymbol{s}_p = [\partial \log \pi / \partial x_1, \partial \log \pi / \partial x_2, \partial \log \pi / \partial x_3]$, the Stein score of the depth z can be computed from the density (67) evaluated at these angles:

$$\frac{\partial}{\partial z} \log \widetilde{\pi}(z, \theta_p, \phi_p) = \frac{\partial}{\partial z} \log(z^2 \sin\theta_p) + \frac{\partial \log \pi}{\partial z} \tag{68}$$

$$= \frac{2}{z} + \frac{\partial \log \pi}{\partial x_1}\frac{\partial x_1}{\partial z} + \frac{\partial \log \pi}{\partial x_2}\frac{\partial x_2}{\partial z} + \frac{\partial \log \pi}{\partial x_3}\frac{\partial x_3}{\partial z} \tag{69}$$

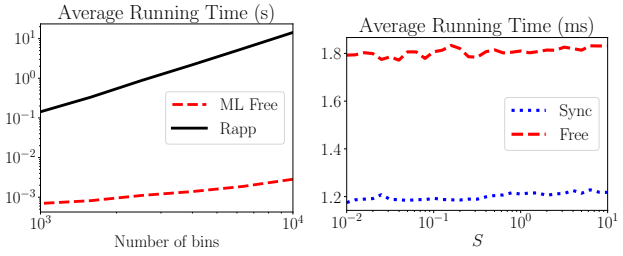
$$= \frac{2}{z} + \begin{bmatrix} \frac{\partial \log \pi}{\partial x_1} \\ \frac{\partial \log \pi}{\partial x_2} \\ \frac{\partial \log \pi}{\partial x_3} \end{bmatrix} \cdot \begin{bmatrix} \sin\theta_p \cos\phi_p \\ \cos\theta_p \\ -\sin\theta_p \sin\phi_p \end{bmatrix} \tag{70}$$

$$= \frac{2}{z} + \boldsymbol{s}_p \cdot \widehat{\boldsymbol{r}}_p = \frac{2}{z} + \sigma_p, \tag{71}$$

where $\widehat{\boldsymbol{r}}_p$ is the unit vector pointing in the direction of the scan angles (θ_p, ϕ_p). In SSDR, we use only $\sigma_p = \boldsymbol{s}_p \cdot \widehat{\boldsymbol{r}}_p$ as the depth score, since we find that $2/z$ often dominates the other term and does not provide good guidance.

E. Additional Simulation Results

E.1. Running time for Free-running Estimators



(a) ML Free vs Rapp (b) ML Free vs ML Sync

Figure 2. (a) Running time of the free-running ML depth estimator and Rapp et al. [10]'s method with different bin sizes averaged over 100 Monte Carlo trials. The setting matches that in Figure 2 of the main paper with $S = 0.1$ and $B = 1$, but the bin size and thus the number of bins vary. (b) Running time of free-running and synchronous ML estimators with the bin size of 0.01 ns, *i.e.*, 10000 bins as $t_r = 100$ ns.

We provide additional results regarding the running time of depth estimator for the free-running mode. Rapp et al. [10]'s method requires computing the stationary distribution of the relative detection times, which involve an expensive eigenvector computation. In contrast, the proposed ML depth estimator for the free-running mode can be implemented efficiently as matched filtering. Both methods operate on the histogram. However, the running time of Rapp et al. [10]'s method scales poorly with the number of time bins, as shown in Fig. 2a. At 10000 bins, the ML estimator takes 2.84 ms on average, while Rapp et al. [10]'s method takes 14.3 s, or approximately 5000x that of the ML estimator. Additionally, the ML estimator allows continuous refinement by maximizing the log likelihood with gradient-based optimization if we have access to detection times with higher numerical precision.

In Fig. 2b, we compare free-running and synchronous ML depth estimators under the same scene parameters and number of time bins. Both estimators are matched filters with access to ground truth S and B. Although the free-running ML estimator is slower, it is still computationally efficient, taking only 1.8 µs for 10000 time bins.

E.2. Hyperparameter Tuning for SSDR

SSDR has four hyperparameters: the depth score thresholds for iterative updates (ϵ) and median smoothing (ϵ_{init}), the step size (γ), and the regularization weight (α). We tune them using the *Optuna* package [1] with the Tree-structured Parzen Estimator [3], minimizing the RMSE of z estimates over 200 SSDR iterations for 100 hyperparameter samples. The tuning dataset is the free-running measurement of the "Mario" scene in Fig. 3, yielding the reported hyperparameters.

E.3. Additional 3D Imaging Results

The simulated 3D imaging results in Fig. 3 extend the main paper by including reconstructions from ideal measurements, SSDR-regularized results for ideal and synchronous measurements, and additional error metrics such as the RMSE of background flux estimates and the RMSE of depth estimates.

For pixel-wise ML estimates, free-running measurements consistently outperform synchronous ones in estimating S, B, and z, while ideal measurements provide the most accurate results. However, for SSDR-regularized reconstructions, the trend is less clear. In the "Man" and "Mario" scenes, both $\text{MAE}(\widehat{z}, z)$ and $\text{RMSE}(\widehat{z}, z)$ are higher for free-running than for synchronous measurements, despite significant improvement from pixel-wise ML estimates. We attribute this to randomness in the SSDR algorithm.

E.4. Ablation Study for SSDR

For the ablation study, we remove three key features from SSDR—one at a time: median smoothing for initialization, depth score hard-thresholding, and noise in the iterative update (turning it into standard gradient descent). Hyperparameters remain the same as in the main paper. Fig. 4 shows the RMSE of depth estimates at each SSDR iteration with and without each component. The tested scenes ("Duck," "Man," and "Mario") and measurement types ("Ideal," "Synchronous," and "Free-running") match those in Fig. 3.

Depth score thresholding has the greatest impact on RMSE reduction; without it, RMSE increases over iterations. Median smoothing is the next most influential—without it,

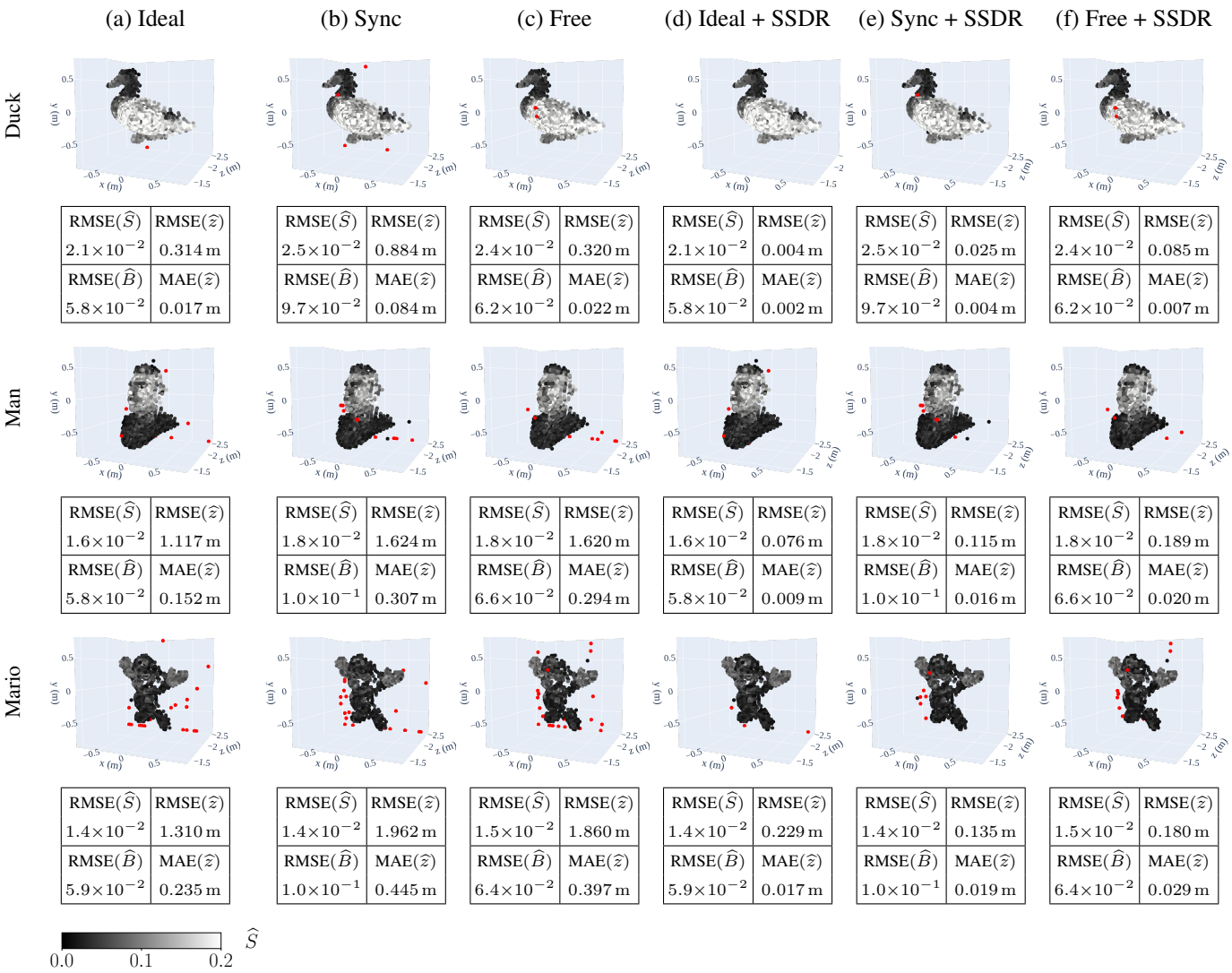


Figure 3. Reconstructions from simulated SPL measurements using pixel-wise ML estimators for **(a) ideal**, **(b) synchronous**, and **(c) free-running** measurements. Depth-regularized SSDR reconstructions are shown in **(d)**, **(e)**, and **(f)**. Point color represents the signal flux estimate, while red points indicate those projected onto the plotting space due to axis limits. Error statistics, including RMSEs of signal flux, background flux, and depth estimates, as well as the mean absolute error (MAE) of depth estimates, are provided beneath each reconstruction.

RMSE initially decreases but then plateaus, likely due to local optima in the log-posterior density. The iterative update noise has the least effect on final RMSEs, though in some cases (e.g., "Mario" scene with ideal measurements), it helps avoid local optima, leading to improved results. SSDR, following the Plug-and-Play Monte Carlo framework [12], also enables uncertainty quantification through multiple reconstruction samples.

E.5. SPL Regularization Comparison

We compare the performance of the proposed SSDR algorithm to ManiPoP, a Bayesian approach to SPL regularization [14], for ideal measurements. While ManiPoP can provide more accurate reconstruction and reduce outlier points, it can introduce structured artifacts and distort the point cloud's shape.

In Fig. 5, the scene consists of the "Duck" model from the Greyc 3D Colored Mesh Database [8] with an added back pane. The detector raster scans a 99×99 grid, using SPL settings identical to Section 4.4 of the main paper, with signal flux scaled to a maximum of 0.1. SSDR slightly reduces depth errors from the pixel-wise ML reconstruction but retains some outlier points. We remark that the score model is trained on point clouds of isolated 3D models [7], so scenes with targets at different depths can be challenging. ManiPoP achieves lower depth MAE but higher RMSE

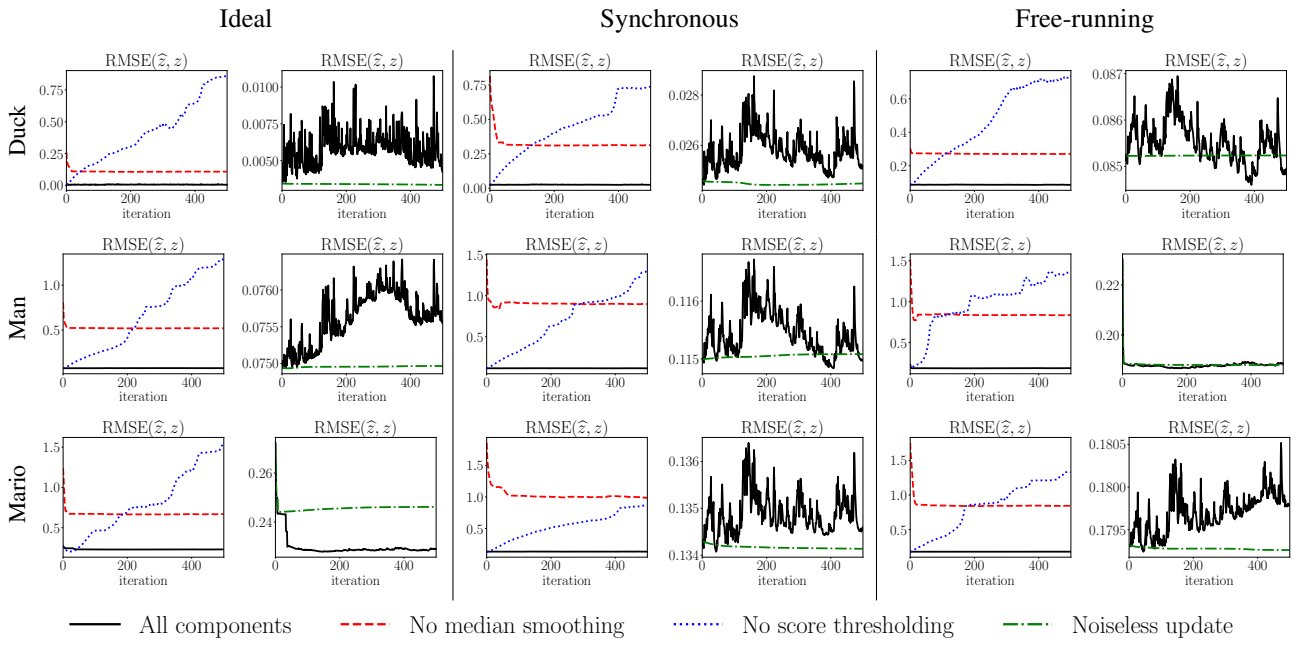


Figure 4. Ablation study: Depth RMSE per iteration of the SSDR algorithm under four conditions: (1) full SSDR algorithm, (2) without median smoothing, (3) without hard-thresholding the depth score, and (4) using noiseless iterative updates (i.e., standard gradient descent). Results are shown for ideal, synchronous, and free-running measurements across the three scenes from Fig. 3.

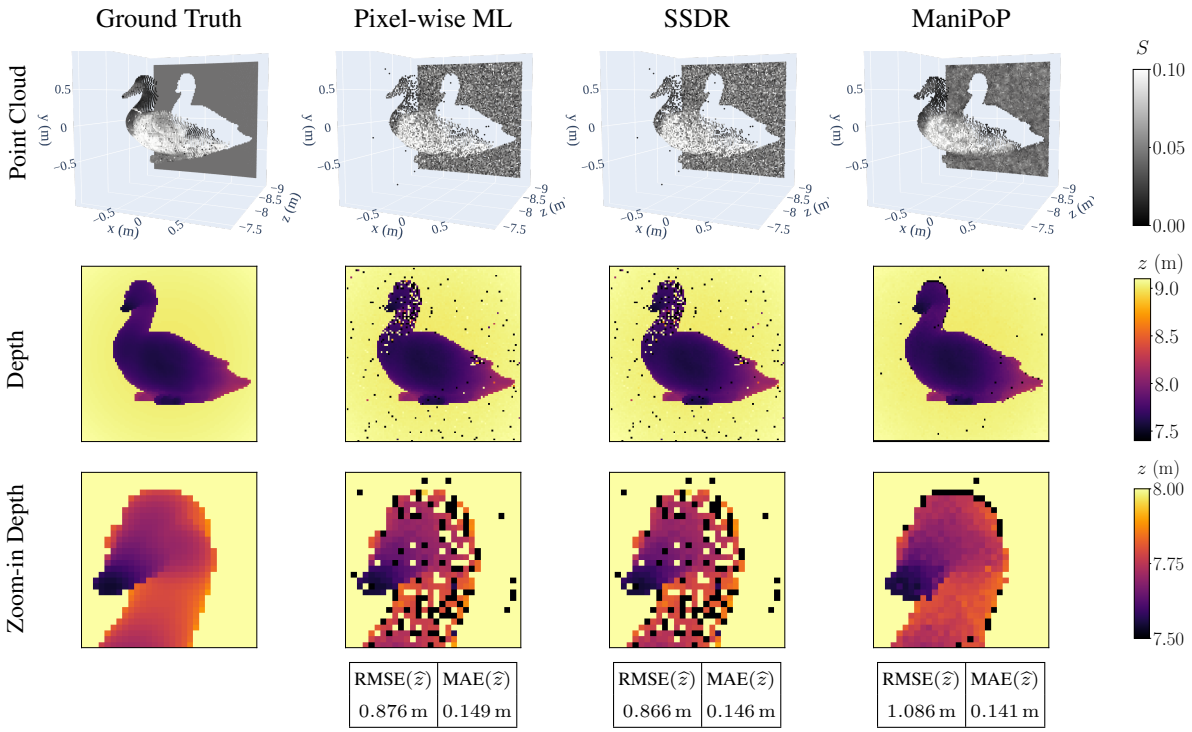


Figure 5. Comparison between SSDR and ManiPoP [14]: ManiPoP produces smoother reconstructions with lower error and regularizes signal flux estimates. However, it introduces distortions, evident in the duck's bill in the zoomed-in depth maps and depth underestimation at boundary between the duck and the background pane.

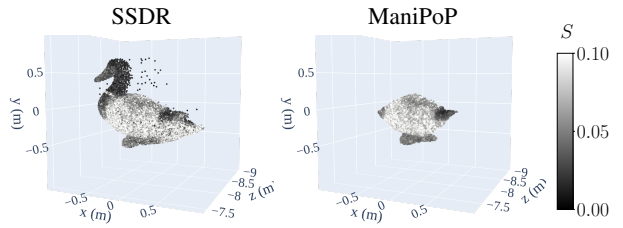


Figure 6. Comparison between SSDR and ManiPoP [14] without a back pane: ManiPoP can introduce large distortions. In this case, it discards regions with low signal flux and warping the shape of the remaining structure.

due to artifacts, including severe depth underestimation in a row at the bottom of the scene. The zoomed-in depth map reveals shape distortions in ManiPoP's reconstruction, whereas SSDR preserves structure more faithfully.

ManiPoP supports multi-depth imaging, detecting whether a pixel contains one or multiple targets. However, this can be a disadvantage, as shown in Fig. 6. In a scene without a back pane, ManiPoP eliminates the duck's head due to its lower signal flux, also distorting the remaining structure. Finally, we remark that the derived log likelihoods for synchronous and free-running modes can be integrated into regularization methods such as TV regularization [6], ManiPoP [14] and RT3D [13] to made them applicable to these detector modes.

F. Additional Experimental Details

F.1. Experimental Setup

The setup includes a pulsed laser (PicoQuant LDH-P-C-640-B) operating at $10\,\mathrm{MHz}$, resulting in $100\,\mathrm{ns}$ repetition period, and $231\,\mathrm{ps}$ pulse width. The laser is raster-scanned over a 64×64 grid using a Thorlabs GVS 202 galvo system. A FastGatedSPAD (Micro Photon Devices) detects photons at an avalanche threshold of $16\,\mathrm{mV}$ and bias current of $50\,\mathrm{mA}$.

In synchronous mode, the SPAD gate remains open for $t_{\mathrm{on}} = 60\,\mathrm{ns}$ in each repetition period, with an $81\,\mathrm{ns}$ hold-off time ensuring at most one detection per period. This slightly modifies the detection model, treating t_{on} as the effective period in ML estimation while using the full $100\,\mathrm{ns}$ repetition period to count active repetitions N_r'. In free-running mode, the dead time is $48\,\mathrm{ns}$. The SPAD output is recorded by a Swabian Instruments Time Tagger Ultra with a $4\,\mathrm{ps}$ bin width over a $10\,\mathrm{ms}$ acquisition time per pixel. Two objects at approximately $2.4\,\mathrm{m}$ depth are measured with tunable ambient light levels using two LED lamps. For the low-flux setting, both LED lamps and the room light are turned off. For high-flux synchronous and free-running acquisition, both LED lamps are turned on, although the room light is still off.

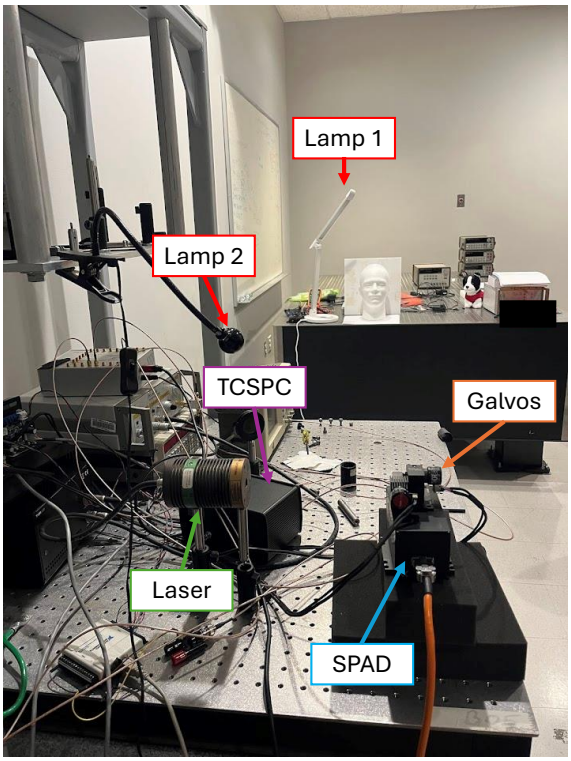


Figure 7. The experimental setup for acquiring SPL measurements. All lights are turned off for the low-flux setting. The two lamps are turned on while the room light is off for the synchronous and free-running high-flux settings.

F.2. Additional Experimental Results

Fig. 8 shows additional 3D reconstructions from experiments. We introduce another scene and include background flux estimates, along with statistics such as the average background flux estimate and the MAE of depth estimates. Reconstructions from low-flux measurements serve as the reference for computing signal flux and depth errors.

Consistent with simulation trends, free-running measurements yield more accurate signal flux and depth estimates. In contrast, synchronous mode fails to reconstruct the scene, as pile-up causes depth underestimation, resulting in a point cloud near the detector. For the "Mannequin" scene, SSDR reduces MAE of depth estimates by 28%. Background estimates for high-flux measurements remain uniform across pixels, as expected, while in low-flux measurements, they correlate with signal flux estimates due to laser multiple scattering. In the "Dog" scene, low reflectivity in dark regions makes estimation difficult, even in low-flux conditions. Both synchronous and free-running high-flux estimates remain unreliable in those regions. For low-flux measurements, some estimated points on the Dog's dark regions appear on the cardboard, because the laser spot has a finite size and it partially illuminates the more reflective cardboard.

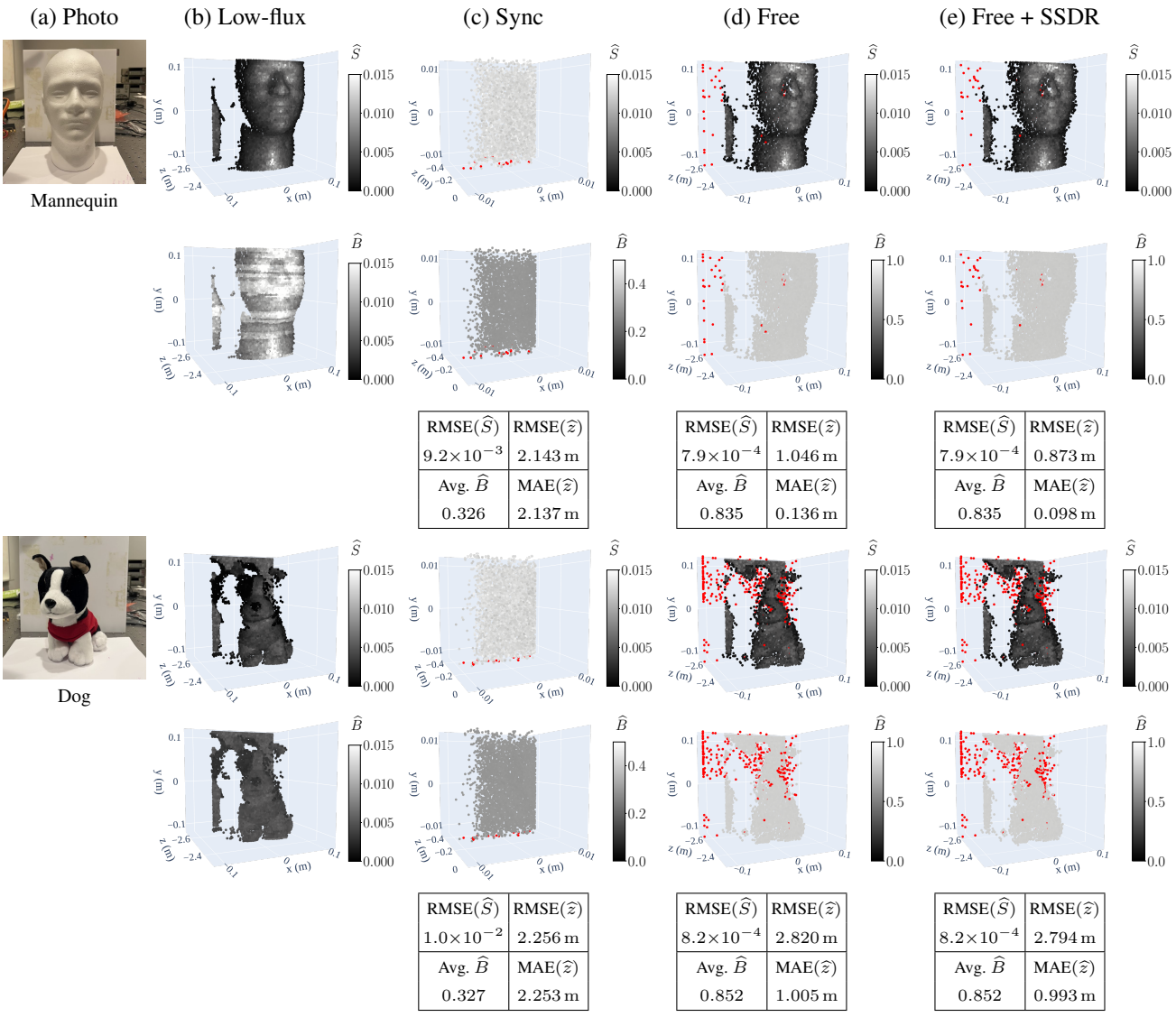


Figure 8. Additional 3D reconstruction results from experimentally collected SPL measurements. We include an additional scene and provide background flux estimates for each pixel, along with extra statistics such as the average background flux estimate and the MAE of depth estimates. Low-flux estimates serve as the ground truth for error computation.

References

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019. 5

[2] Israel Bar-David. Communication under the Poisson regime. *IEEE Transactions on Information Theory*, 15(1):31–37, 1969. 1, 2

[3] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Adv. Neural Inform. Process. Syst.*, 24, 2011. 5

[4] P. B. Coates. The correction for photon 'pile-up' in the measurement of radiative lifetimes. *Journal of Physics E: Scientific Instruments*, 1(8):878–879, 1968. 4

[5] Anant Gupta, Atul Ingle, Andreas Velten, and Mohit Gupta. Photon-flooded single-photon 3D cameras. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6770–6779, 2019. 4

[6] Felix Heide, Steven Diamond, David B. Lindell, and Gordon Wetzstein. Sub-picosecond photon-efficient 3D imaging using single-photon sensors. *Scientific Reports*, 8(1):17726, 2018. 8

[7] Shitong Luo and Wei Hu. Score-based point cloud denoising. In *Int. Conf. Comput. Vis.*, pages 4563–4572, 2021. 4, 6

[8] Anass Nouri, Christophe Charrier, and Olivier Lézoray. Technical report: Greyc 3D colored mesh database. Technical

report, Normandie Université, Unicaen, EnsiCaen, CNRS, GREYC UMR 6072, 2017. 6

[9] Joshua Rapp and Vivek K Goyal. A few photons among many: Unmixing signal and noise for photon-efficient active imaging. *IEEE Transactions on Computational Imaging*, 3 (3):445–459, 2017. 3

[10] Joshua Rapp, Yanting Ma, Robin M. A. Dawson, and Vivek K. Goyal. Dead time compensation for high-flux ranging. *IEEE Transactions on Signal Processing*, 67(13):3471–3486, 2019. 5

[11] Donald L. Snyder and Michael I. Miller. *Random Point Processes in Time and Space*. Springer Science & Business Media, 2012. 1, 3, 4

[12] Yu Sun, Zihui Wu, Yifan Chen, Berthy T. Feng, and Katherine L. Bouman. Provable probabilistic imaging using score-based generative priors. *IEEE Transactions on Computational Imaging*, 10:1290–1305, 2024. 6

[13] Julián Tachella, Yoann Altmann, Nicolas Mellado, Aongus McCarthy, Rachael Tobin, Gerald S. Buller, Jean-Yves Tourneret, and Stephen McLaughlin. Real-time 3D reconstruction from single-photon lidar data using plug-and-play point cloud denoisers. *Nature Communications*, 10(1):4984, 2019. 8

[14] Julián Tachella, Yoann Altmann, Ximing Ren, Aongus McCarthy, Gerald S. Buller, Stephen McLaughlin, and Jean-Yves Tourneret. Bayesian 3D reconstruction of complex scenes from single-photon lidar data. *SIAM Journal on Imaging Sciences*, 12(1):521–550, 2019. 6, 7, 8