Unlocking the Performance of Proximity Sensors by Utilizing Transient Histograms

Carter Sifferman[1], Yeping Wang[1], Mohit Gupta[1], and Michael Gleicher[1]

***Abstract*—We provide methods which recover planar scene geometry by utilizing the *transient histograms* captured by a class of close-range time-of-flight (ToF) distance sensor. A transient histogram is a one dimensional temporal waveform which encodes the arrival time of photons incident on the ToF sensor. Typically, a sensor processes the transient histogram using a proprietary algorithm to produce distance estimates, which are commonly used in several robotics applications. Our methods utilize the transient histogram directly to enable recovery of planar geometry more accurately than is possible using only proprietary distance estimates, and consistent recovery of the albedo of the planar surface, which is not possible with proprietary distance estimates alone. This is accomplished via a differentiable rendering pipeline, which simulates the transient imaging process, allowing direct optimization of scene geometry to match observations. To validate our methods, we capture 3,800 measurements of eight planar surfaces from a wide range of viewpoints, and show that our method outperforms the proprietary-distance-estimate baseline by an order of magnitude in most scenarios. We demonstrate a simple robotics application which uses our method to sense the distance to and slope of a planar surface from a sensor mounted on the end effector of a robot arm.**

***Index Terms*—RGB-D Perception, Range Sensing**

I. Introduction

OPTICAL time-of-flight proximity sensors which measure scene *transients* have recently become widely available. These sensors operate by illuminating the scene with a pulse of light, and measuring the *shape* of that pulse over time as it returns back from the scene in a *transient histogram*, as shown in Figure 1. These *transient sensors* have seen use in robotics due to their ability to reliably report a distance estimate over a wide range (1cm - 5m) while being small (< 20 mm^3), lightweight, and low-power (on the order of milliwatts per measurement) [1], [2]. Because of their form factor, transient sensors can be placed in locations where higher resolution 3D sensors cannot, such as on the gripper or links of a robot manipulator, or on very small robots. While these sensors have many desirable properties, existing robotics applications do not utilize the transient histograms, instead relying on low-resolution (at most 4×4 pixel) proximity measurements generated onboard the sensor. Due to the coarseness of their measurements, these sensors are presently only used in robotics for coarse sensing, *e.g.,* detecting the presence of obstacles or distance to a target.

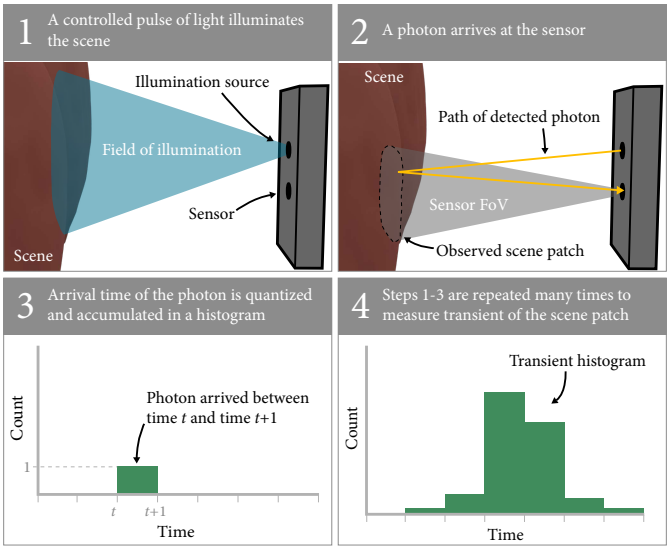

Fig. 1: **The transient histogram is a temporal waveform which records the response of a scene patch when exposed to a pulsed light source.** Presently available commodity sensors estimate the transient histogram through a repeated process.

In this work, we utilize transient histograms directly to recover accurate planar scene geometry, and consistent planar albedo from a single 3×3 transient sensor measurement. Planar geometry is an initial use case for our methods, and is a special case of 3D sensing that has many applications in robotics. A robot interacting directly with any planar surface will benefit from sensing the geometry of that surface accurately and at a close range. For example: a robot arm placing an object on a tabletop, sweeping a floor, or writing on a flat surface; a mobile robot localizing the floor and walls of a room; or a drone finding a safe spot to land. Our method enables accurate recovery of this planar geometry that otherwise would have required multiple proximity sensors or a depth camera, while maintaining the same very small form factor and operating at ranges as low as 1cm.

This work is the first to demonstrate that utilizing transient histograms can improve the performance of proximity sensors over utilizing proprietary on-sensor distance estimates. To achieve this, our contributions are 1) an effective *forward imaging model* for commodity proximity sensors, 2) a *differentiable rendering pipeline* which implements the forward

Manuscript received: May 31, 2023; Revised August 19, 2023; Accepted August 23, 2023.

This paper was recommended for publication by Editor Sven Behnke upon evaluation of the Associate Editor and Reviewers comments. This work was supported by Los Alamos National Laboratory and the Department of Energy, a University of Wisconsin Vilas Award, and National Science Foundation awards 1830242, 1943139, 2107060, and 2003129.

[1]The authors are with the Department of Computer Sciences, University of Wisconsin-Madison, Madison 53706, USA [sifferman|yeping|mohitg|gleicher]cs.wisc.edu

Digital Object Identifier (DOI): see top of this page.

imaging model and utilizes it to recover planar geometry and albedo directly from transient histograms, 3) an *empirically calibrated approach* which approximates the performance of the differentiable rendering pipeline and acts as a baseline, and 4) *empirical evidence* that our approaches outperform alternative methods which do not utilize transient histograms.

We present two methods for recovery of planar geometry, one of which can also be used to consistently recover the albedo of the planar surface. To evaluate our methods, we gather thousands of measurements of eight planar surfaces with a commodity transient sensor from a range of angles-of-incidence and distances. We find that our methods which utilize the transient histogram are more accurate and robust than those which rely on proprietary distance estimates. We also find that our method recovers consistent planar albedo, which is not possible to recover from proprietary distance estimates, as they do not encode intensity information. We build a demonstration application which takes advantage of the small size of a transient sensor by mounting the sensor to the gripper of a robot arm. Measurements from the sensor are used to measure the distance to the surface below the gripper and to ensure that the surface is level before placing an object.

II. Background: Transient Histograms

A *transient* is a one dimensional temporal waveform which measures the light reflected from a scene over time in response to a pulsed light source. Recently, sensors which are able to capture a transient quantized over short (picosecond) time scales have become available for distance/range measurement using the time-of-flight principle. We refer to these sensors as *transient sensors*. These sensors come in a range of form factors: from high resolution lab-grade arrays to mobile device LiDAR modules, to very small proximity sensors. Most notable of the currently available transient sensors is the single photon avalanche diode (SPAD) [3], [4], which is inexpensive and commonly used in robotics (see Section III-A).

As shown in Figure 1, a SPAD-based sensor approximates the transient histogram through a repeated process. A controlled pulse of light (typically infrared) flood-illuminates the scene in front of the sensor. Each sensor pixel records the elapsed time between this pulse being sent and a single photon arriving at the pixel. This arrival time is quantized to a discrete bin and accumulated in a *transient histogram*. Over many photon arrivals, this histogram approximates the true transient. In practice, a commodity sensor may record millions of photon arrivals to form a transient histogram. In sensors with an array of pixels, a transient histogram is generated for each pixel.

Currently available commodity transient sensors have many desirable properties. Many are capable of gathering transient histograms at 30 frames per second. Maximum range varies by model, but may be as high as 5m, with a typical minimum range of 1cm. There exist techniques for mitigating the effects of high ambient light on these sensors, enabling their operation in diverse environments [5], [6].

In this work, we evaluate our method using the SPAD-based TMF8820 sensor manufactured by AMS. We choose this sensor because it 1) allows access to transient histograms through an official driver, 2) captures a 3×3 grid of transient histograms at a time, each from a different region of its field-of-view, and 3) provides access to a "reference histogram" which encodes the intensity of the laser pulse over time. In the sensor's default configuration, transient histograms are summarized onboard the sensor via a proprietary algorithm, and a distance and confidence estimate are reported for each field-of-view region. We reconfigure the sensor to report a transient histogram *and* proprietary distance estimate for each FoV region. While we utilize the TMF8820 in this work, the methods we propose can be applied to any sensor which reports a transient histogram.

III. Related Work

A. Transient Sensors in Robotics

Transient sensors are widely used in robotics applications as they provide highly reliable distance measurements, while being lightweight, low-cost and low-power. Tsuji and Kohama [7] demonstrate a "sensitive skin" for a robot arm consisting of many single-pixel transient sensors. Similarly, Adamides et al. [8] propose an array of transient sensors mounted around a robot wrist to achieve safe human-robot collaboration. Escobedo et al. place transient sensor on robot joints and use them to actively avoid collisions [9]. Transient sensors have been used to detect obstacles when mounted on a drone [10]. Our previous work characterized two transient sensors and demonstrated a method for extrinsically calibrating their position relative to a robot arm to which they are attached [11]. Outside of robotics, commodity transient sensors have seen use in wearable computing [12] and inspection applications [13]. In these prior works, only the sensor's proprietary distance estimates are utilized. To our knowledge, our work is the first to utilize the transient histogram in a robotics setting.

B. Inference from Transient Histograms

Our differentiable rendering pipeline and forward imaging model are heavily inspired by prior work in imaging. Photon arrival times, like those encoded by a transient histogram, are heavily utilized in non-line-of-sight (NLOS) imaging, pioneered by Velten et al. [14]. In NLOS imaging, scene geometry is recovered from around the corner by reflecting a powerful pulsed laser off a diffuse surface. Recent NLOS works utilize the same single photon avalanche diode (SPAD) technology as the sensor that we use in this work [15], [16]. However, the imaging setup used in NLOS imaging requires a high-powered laser and relatively large, expensive laboratory grade SPAD sensors, which have thousands of histogram bins and very precise timing. In contrast, the sensor that we use in this work is readily available, small, lightweight, and eye safe, but reports only 128 histogram bins, and has less precise timing and optical characteristics.

A number of recent papers have utilized transient histograms from commodity SPADs to perform scene inference. Each of these works uses sensors which are very similar to the one used in this paper in terms of technology, form factor, and cost. Callenberg et al. [17] propose the use of transient

histograms from a SPAD to classify materials based on sub-surface scattering (with the sensor placed in direct contact), generate higher resolution depth imagery, and perform non-line-of-sight imaging (with additional hardware). Becker and Koerner [18] also classify materials, but in a non-contact setting. Ruget et al. [19] perform super resolution and use supervised machine learning to estimate human poses from transient histograms. Other works also perform super resolution to resolve higher resolution depth images from relatively few transient histograms [20], [21].

The differentiable rendering approach used in this work is inspired by Jungerman et al. [22], who use differentiable rendering to recover partial plane parameters from a single transient histogram. Because the sensor used by Jungerman et al. reports only a single transient histogram, only two of the three planar degrees of freedom could be recovered from a single sensor measurement. In contrast, the multiple transient histograms reported by the sensor used in this work enable recovery of all plane parameters from a single measurement, and our work is the first to do so.

IV. Forward Imaging Model

An accurate forward imaging model is crucial to enabling our differentiable rendering method. In this section, we give an overview of our forward imaging model, which is designed for the TMF8820 sensor, but can in principle be adapted to other sensors. Our model assumes planar scene geometry, with uniform albedo and reflectance model parameters per-plane. For a more general forward imaging model that is sensor agnostic, refer to previous work [22]. We consider a set of transient histograms which are simultaneously captured by a transient sensor over different fields-of-view. We refer to this set of histograms as an *image* Φ. Each image consists of n histograms $\varphi \in \Phi$. Each histogram consists of m bins, $\varphi_i : 1 \leq i \leq m$.

A. Surface Reflection Model

We utilize the Phong reflection model [23], in which a surface's reflection properties are parameterized by its albedo α, specular exponent k_e, and specular weight k_s. We assume that the light source and sensor are co-located, the pulsed laser source is the only light in the scene, and the strength of illumination is uniform over the field of view. The intensity I of incident light returned by a ray $\mathbf{r} \in \mathbb{R}^3$ intersecting with plane $\mathbf{a}\mathbf{x} + d = 0$ is given by:

$$I = \alpha * (1 - k_s)(\mathbf{r} \cdot \mathbf{a}) + k_s((2(\mathbf{r} \cdot \mathbf{a})\mathbf{a} + \mathbf{r}) \cdot \mathbf{r})^{k_e} \quad (1)$$

B. SPAD Saturation

The Phong reflection model alone does not take into account light falloff or unique properties of the SPAD sensor. Previous work [24] has established that, due to the nature of SPADs, the number of detected photons p follows a soft saturation curve in relation to the number of incident photons ϕ, given by $p = 1 - e^{-\phi}$. Due to the inverse-square law, a ray which travels distance r from the sensor before bouncing off the scene returns with an intensity of $1/r^2$. We incorporate the plane's albedo α, as well as the output I from the lighting model given in Equation (1). The asymptotic highest possible photon detection count σ is a property of the sensor. The sensor gain parameter g scales the intensity for an individual ray–this is included because in practice we do not simulate as many rays as photons are actually measured by the sensor. The number of detected photons p is then given by:

$$p = \sigma(1 - e^{-gI/(\sigma r^2)}) \quad (2)$$

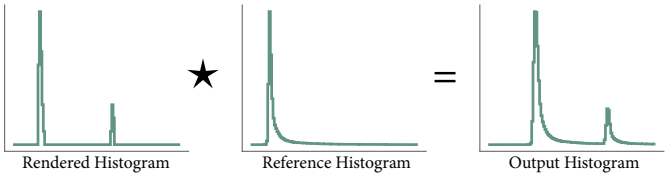

Fig. 2: The raw rendered histogram is cross-correlated with the reference histogram (which encodes the laser pulse intensity over time) to generate the output histogram of our forward imaging model.

C. Histogram Formation

Consider a histogram φ which images a plane given by $\mathbf{a}\mathbf{x}+d = 0$, with a uniform albedo and reflectance parameters. Let the sensor reside at the origin, and let R be a set of rays uniformly sampled from the field-of-view which φ images. If φ has n bins, a bin temporal "size" of t, and a bin offset ω (meaning a flight time of t is recorded as $t + \omega$), the value of an individual histogram bin is given by:

$$i_s = \omega + t(i-1) \qquad i_e = \omega + ti$$

$$\varphi_i^{raw} = \sum_{\mathbf{r} \in R} \begin{cases} p(\mathbf{r}) & \text{if } i_s \leq \dfrac{2||\texttt{isect}(\mathbf{r}, \mathbf{a}, d)||_2}{c} < i_e \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Where $p(\mathbf{r})$ is the intensity of light returned by ray $\mathbf{r}$, as given in Equation (2), c is the speed of light, and $\texttt{isect}(\cdot) \in \mathbb{R}^3$ is the intersection point of $\mathbf{r}$ with $\mathbf{a}\mathbf{x}+d = 0$. Because the sensor that we model (TMF8820) filters out ambient light on-sensor, we assume no ambient light in our imaging model.

D. Laser Impulse

The sensor which we model records the intensity of its laser impulse over time by piping the laser pulse directly to a SPAD [1], and reports the result as a "reference histogram". The captured transient histogram is effectively temporally blurred by a kernel matching the reference histogram. To replicate this effect, we cross-correlate the reference histogram with the generated histogram as a step in our forward process, as shown in Figure 2. In the case of the TMF8820 sensor that we utilize, the temporal scale (bin size) is not the same in the reference histogram δ as in the transient histogram φ, so we temporally scale δ by a factor s_δ before applying the cross-correlation. The histogram after correlation is given by

$$\varphi^{corr} = \varphi^{raw} \star \texttt{rescale}(\delta, s_\delta) \quad (4)$$

Where $\star$ denotes the cross-correlation operation, and the `rescale` function scales the function δ temporally by s_δ.

E. Inter-histogram Interference

The sensor that we model suffers from *inter-histogram interference*, meaning light detected in one histogram is also detected in other histograms, scaled by a factor. We assume that one histogram interferes with all other histograms with an equal magnitude ψ, meaning that a bin value of x in one histogram will manifest as ψx in all other histograms. Formally, for a histogram $\varphi \in \Phi$, where φ_i is the i^{th} bin of φ,

$$\varphi_i = \varphi_i^{corr} + \psi \sum_{\tilde{\varphi}^{corr} \in \Phi^{corr}} \tilde{\varphi}_i^{corr} \tag{5}$$

V. Differentiable Rendering Pipeline

Our differentiable rendering pipeline recovers plane parameters by minimizing the loss between an observed histogram and the output of a render function. The render function renders a histogram image Φ^r as a function of four sets of variables: the scene geometry G, reflectance model parameters F, the sensor's forward imaging model parameters C, and the sensor's reported laser impulse δ:

$$\Phi^r = R(G, F, C, \delta) \tag{6}$$

The render function assumes that the sensor is placed at the origin and the optical axis is aligned with the positive z axis. The planar geometry G is given by the angle of incidence θ of the optical axis to the plane, the intersection point of the plane with the z axis Z_0, and the azimuth angle ϕ, which denotes rotation about the optical axis.

The lighting parameters F are comprised of the Phong reflection model parameters (k_s, k_e, α). The camera parameters C are comprised of those described in Section IV $(n, t, g, \psi, s_\delta, \sigma)$, along with 36 scalar parameters which define the height, width, and center of each of the sensor's 9 FoV regions. These FoV parameters are derived from the TMF8820 specification sheet [1], and are not differentiable in our implementation. Also included in C is an integer which denotes the number of random ray samples used to render the transient histogram. We keep this fixed at 2304 per FoV region. The impulse response function δ is reported by the sensor along with every image.

To compare the rendered histogram Φ^r to the observed histogram Φ^o, the following loss function $\mathcal{L}$ is used:

$$\mathcal{L}(\Phi^r, \Phi^o) = \sum_{(\varphi^r, \varphi^o) \in \Phi^r, \Phi^o} \left|\left| \frac{\varphi^r}{\max(\varphi^o)} - \frac{\varphi^o}{\max(\varphi^o)} \right|\right|_2 \tag{7}$$

Dividing by the magnitude of the observed histogram ensures that high magnitude histograms do not dominate the loss. Unlike previous work [22], we do not use a Fourier transform-based loss function. In our tests, the L2-norm function above performed slightly better. We believe this is because we utilize a good initial estimate from the histogram peak based approach described in Section VI. A Fourier-based loss excels when the rendered and observed histograms are very different, but may not provide as strong of a signal when they are similar. We adapt Mu et al.'s Python implementation [25] of the algorithm given by Urea et al. [26] to uniformly sample rays from the rectangular FoV of the TMF8820.

The process of assigning a value to a histogram bin is inherently non-differentiable, as there is an instantaneous change in the output histogram as the input crosses a bin boundary. Following previous work [22], we make the render function differentiable via a soft binning process, in which a Gaussian kernel is generated centered at each input datapoint, and these Gaussians sampled at the bin centers and summed across the datapoint dimension to generate an approximation of the histogram. In our implementation, each Gaussian is also weighted according to its intensity, which is given by Equation (2). The same soft binning process is used to temporally scale the reference histogram δ.

To tune the parameters of our forward imaging model, we utilize a large dataset D of (Φ^o, δ) pairs, each with an associated ground truth geometry G. We minimize the reconstruction loss, as given in Equation (7) over the entire dataset to find the ideal camera parameters C^*:

$$C^* = \arg\min_C \sum_{(\Phi^o, G, \delta) \in D} \mathcal{L}(R(G, F, C, \delta), \Phi^o) \tag{8}$$

Where the reflectance model parameters F are free variables. This optimization only needs to be performed once, as C^* remains fixed for a given sensor.

To recover the geometric parameters G^* of a planar surface from a single image Φ^o with reference histogram δ, we use the optimized forward imaging parameters C^*, while allowing the scene geometry G and reflectance model parameters F to change:

$$G^*, F^* = \arg\min_{G,F} \quad \mathcal{L}(R(G, F, C^*, \delta), \Phi^o) \tag{9}$$

Performing this optimization also recovers the reflectance model parameters F^* of the surface. We evaluate the consistency of the surface albedo recovered by this approach in section Section VII-D. Recovery of other reflectance parameters is left for future work as it is outside of the scope of this paper, and evaluation of these parameters is difficult.

Optimization is performed via stochastic gradient descent using the Adam optimizer [27]. The render function R is implemented in PyTorch with gradients generated through automatic differentiation. To initialize G in Equation (9), we use the output of the histogram peak based approach described in Section VI. We observe that regardless of what reasonable starting estimate is used, the optimization tends to converge to the same solution for planar geometry.

VI. Histogram Peak Based Approach

We provide an empirically calibrated approach which is able to approximate the performance of differentiable rendering on the plane recovery task. This method operates by estimating the distance to the plane in each field of view, projecting outwards by the distance, and fitting a plane to the projected points. To tune the method, we optimize a linear mapping from histogram bin to distance (given by parameters m and b below). We also optimize the angle at which points are projected outwards; a different angle is used depending on whether the histogram corresponds to a field of view region

on the edge or corner of the overall 3×3 region field-of-view (s_e or s_c respectively). The algorithm for this approach is shown in Algorithm 1.

Algorithm 1 Empirically calibrated algorithm for recovering planar geometry from a set of transient histograms using histogram peaks

```
function RecoverPlane(Φ, m, b, s_e, s_c)
    pts ← [ ]
    for φ in Φ do
        i ← the temporal coordinate of the peak of φ
        dist ← i * m + b
        u ← unit vector pointing to center of FoV of φ
        if φ images an edge FoV region then
            Scale angle of u from optical axis by s_e
        else if φ images a corner FoV region then
            Scale angle of u from optical axis by s_c
        end if
        pt ← u * dist
        Append pt to pts
    end for
    Fit a plane to pts via SVD [28]
    return the parameters a, d of the fit plane
end function
```

To find the location of the peak in a histogram φ, we fit a piecewise cubic curve to the 128-bin histogram, and sample that curve at $10\times$ density around the highest individual bin. The temporal position of the highest point on the interpolated curve is the location of the peak. This process captures variations smaller than the $\sim$ 1.2cm equivalent bins of the histogram by using the relative intensity between adjacent bins. We find empirically that this approach outperforms picking the highest bin without interpolation.

To determine the optimal parameters m, b, s_e, and s_c to the RecoverPlane function, we find the parameters which minimize the error in the reconstructed plane over some calibration dataset D which contains images Φ along with ground truth planar geometry $\mathbf{a}, d$:

$$m^*, b^*, s_e^*, s_c^* = \underset{m,b,s_e,s_c}{\arg\min} \sum_{\Phi,\mathbf{a},d \in D} \epsilon_p(f(\Phi, m, b, s_e, s_c), \mathbf{a}, d) \tag{10}$$

where the ϵ_p is the point error between two planes, as defined in Equation (11), and f is the RecoverPlane function given in Algorithm 1. We perform this optimization using the Nelder-Mead method [29] with finite difference estimation of derivatives, via the SciPy Python library. As this optimization is performed only once, speed is not crucial.

VII. Experimental Results

A. Sensor Configuration

We run the TMF8820 in "short range, high accuracy" mode, in which it reports 128 bins with an individual bin size equivalent to $\sim$ 1.2cm of distance. We run the sensor in the default configuration of 4 million iterations (light pulses) per measurement, and use the default field-of-view configuration, which gives an FoV of $33° \times 34°$, divided into 3×3 regions, with a transient histogram reported for each region.

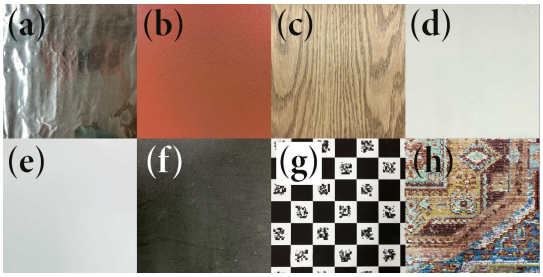

Fig. 3: **Materials on which we evaluate planar recovery.** (a) aluminum foil; (b) red painted drywall; (c) wooden table; (d) whiteboard; (e) white paper; (f) black fabric; (g) checkerboard; (h) patterned rug.

B. Metrics

We use three metrics to measure the accuracy of plane recovery. Assume that we are comparing two planes given by $\mathbf{a}_1\mathbf{x} + d_1 = 0$ and $\mathbf{a}_2\mathbf{x} + d_2 = 0$, where $d_1 > 0$, $d_2 > 0$, then the *angular error* $\epsilon_a = \arccos(\mathbf{a}_1 \cdot \mathbf{a}_2)$. *Linear error* is given by $\epsilon_l = |d_1 - d_2|$. These metrics are intuitive, but the trade-off between the two is not clear. To capture error with a single metric, we define point error ϵ_p. Given a random ray originating at the sensor and within the sensor's FoV, point error captures the expected difference between the intersection of that ray with the predicted plane and with the ground truth plane. Formally:

$$\epsilon_p = \frac{\sum_{\mathbf{r} \in R} ||\texttt{isect}(\mathbf{a}_1, d_1, \mathbf{r}) - \texttt{isect}(\mathbf{a}_2, d_2, \mathbf{r})||_2}{|R|} \tag{11}$$

where $\texttt{isect}(\mathbf{a}, d, \mathbf{r})$ returns the 3D point of intersection between plane $\mathbf{a}\mathbf{x} + d = 0$ and ray $\mathbf{r}$, and R is a randomly sampled set of rays originating at the sensor and within the sensor's FoV. In practice, we set R to be an 8×8 grid of rays which uniformly cover the sensor's FoV for repeatability.

C. Planar Recovery

We evaluate five different approaches for planar recovery:

1) Differentiable rendering, the optimization problem in Equation (9) is solved.
2) Peak finding - calibrated, the histogram peak based approach given by Algorithm 1 is performed with optimized parameters given by Equation (10).
3) Proprietary distances - calibrated, the same as 2), but utilizing distance estimates generated onboard the sensor rather than histogram peak locations.
4) Peak finding - naive, the histogram peak based approach is used, but without optimized parameters.
5) Proprietary distances - naive, the same as 4), but utilizing distance estimates generated onboard the sensor rather than histogram peaks.

To generate ground truth data, we mount a TMF8820 sensor to a custom 3D printed end effector for a Universal Robots UR5 robot arm. We manually calibrate the position of the sensor relative to the end effector, and use the robot's forward kinematics (which are quoted as precise to $\pm$ 0.1mm) to gather ground truth sensor poses. To determine the position of the plane relative to the robot, the end effector is touched to the

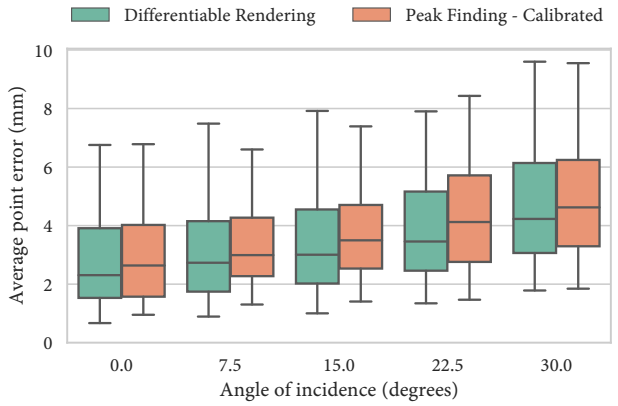

Fig. 4: **Higher angle-of-incidence leads to higher error in reconstruction.** Measurements of materials (c)-(h) cover distance range 0-30cm. Whiskers extend to 5th and 95th percentile.

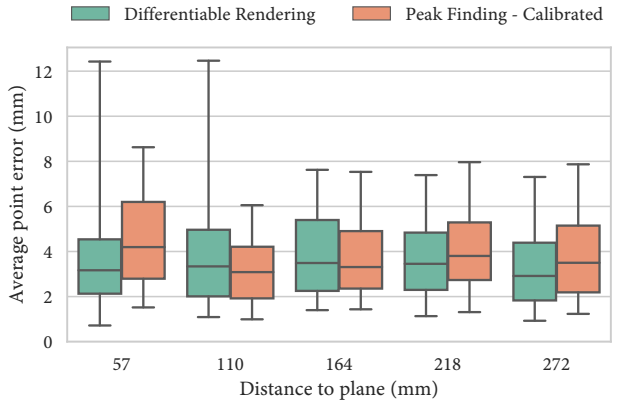

Fig. 5: **Distance to the planar surface has little effect on reconstruction error.** Measurements of materials (c)-(h) cover AoI range 0-30°. Whiskers extend to 5th and 95th percentile. Ticks on x axis denote center of 54mm bins.

plane at a number of points, and a ground truth plane is fit to these points. The robot is used to automatically move the sensor, allowing us to generate a large dataset of planar images (3,800 images total) from a variety of distances (Z_0), angles-of-incidence (θ), and azimuth angles (ϕ). All measurements were captured in an artificially lit room.

To ensure the validity of our results when comparing differentiable rendering to other approaches, we use the worst-performing naive proprietary distances approach as a starting estimate, and perform 100 iterations of gradient descent. One iteration takes about 0.05 seconds on a mid-range laptop (i7-10705H, NVIDIA GTX 1650Ti). In real-world operation, a better starting estimate could be used and fewer iterations performed. The peak-based approaches operate at 95Hz on the same hardware, exceeding the 30Hz at which the sensor reports data.

A comparison between the five approaches is given in Table I. Methods which utilize transient histograms consistently outperform those which rely on proprietary distance estimates. We see that the differentiable rendering approach, which utilizes the entirety of the information in all nine histograms, outperforms peak finding approaches, in which each histogram is reduced to a single value. Our peak finding approach comes close to the performance of differentiable rendering across the board, even outperforming it in some cases, offering speed at the expense of generality. We believe the large gap between the "peak finding" and "proprietary distances" approaches can partially be explained by a difference in interpolation method; the interpolation method used onboard the sensor may be less accurate than the one used in our peak finding method. However, in our testing we found that even when using *no interpolation at all*, our peak finding approach outperformed the proprietary distances approaches, necessitating an additional explanation.

We suspect that the proprietary algorithm onboard the sensor is not overly naive, but instead is designed to be more general purpose than our approach. Plane fitting is a special case; an algorithm which performs well for a variety of potential use cases may not be optimal for plane fitting. The peak finding method that we use was chosen *because* it is effective at recovering planar surfaces. By accessing the transient histograms directly, we were afforded the ability to make this choice. Results of planar recovery over a wider range of sensor poses are shown in Table II. The effect of angle-of-incidence (AoI) and distance on reconstruction error is shown in Figure 4 and Figure 5.

We evaluate our method on a range of surfaces and report the results in Table III. The parameters of the "calibrated" methods and imaging model parameters of the differentiable rendering methods were trained on measurements of the red painted drywall. We see that our methods are generally robust to this change from training to testing surface, particularly when that surface has a uniform texture and albedo. Our methods are slightly less robust to textured surfaces such as wood and a patterned rug. We see diminished performance with the slightly glossy whiteboard, and the checkerboard surface, which has spatially varying albedo. Performance is significantly diminished on the specular aluminum foil.

We observe that the proprietary distance based approach tends to overfit when calibrated to a dataset. There is evidence of this overfitting in the longer range test in Table II; the calibrated histogram approach improves over the naive approach, while the calibrated proprietary distance approach performs *worse* than the naive. This is because the parameters of the "calibrated" approaches were calibrated to recover planar geometry on images of a different surface over a different range of distances and angles of incidence. While the histogram based approaches, including differentiable rendering, are robust to this change in surface, the approaches which utilize proprietary distances are not.

D. Albedo Recovery

We evaluate the performance of our differentiable renderer for recovering surface albedo, as given in Equation (9) by recovering the albedo from images of three planar surfaces which have a uniform texture and albedo. We only evaluate the *consistency* of the recovered albedo, not the *accuracy*. Evaluating the accuracy would require an accurate characterization

Method	Angular Error (°)			Linear Error (mm)			Point Error (mm)		
	Mean	Median	95%	Mean	Median	95%	Mean	Median	95%
Differentiable Rendering*	**3.40**	**1.97**	**12.90**	**2.46**	**1.90**	**6.51**	**3.79**	**3.17**	8.46
Peak Finding - Calibrated†	3.57	2.22	13.44	2.67	2.11	7.13	3.94	3.52	**7.92**
Peak Finding - Naive	5.68	3.87	18.42	6.15	5.28	13.56	7.70	6.96	14.28
Proprietary Distances - Calibrated†	7.34	4.71	25.97	49.20	60.31	68.41	52.44	62.96	69.60
Proprietary Distances - Naive	8.87	4.71	30.06	60.31	71.96	78.14	65.45	76.26	83.15

TABLE I: **Methods which utilize the histogram outperform those which use proprietary distance estimates in all metrics.** Images in range 1-30cm to plane, 0-30°AoI on surfaces (c) - (h). 400 measurements per surface. Measurements of surface (b) from the same range were used optimize forward model of differentiable method (*) and calibrate "calibrated" methods (†). 95% refers to the 95th percentile of error. See Section VII-C for a description of methods.

Method	Point Error (mm)		
	Mean	Median	95%
Differentiable Rendering*	**6.26**	**3.52**	**22.31**
Peak Finding - Calibrated†	6.80	3.78	23.58
Peak Finding - Naive	15.84	11.45	44.56
Proprietary Distances - Naive	42.23	22.15	130.46
Proprietary Distances - Calibrated†	74.45	75.45	143.51

TABLE II: **Methods which utilize the histogram outperform those which use proprietary distance estimates in larger range of plane parameters.** Measurements cover range 1-70cm, 0-45°AoI of surface (b). Measurements of surface (e) from range 0-30cm, 0-30° AoI were used optimize forward model of renderer (*) and to calibrate "calibrated" methods (†).

Material	Mean Point Error (mm)		
	Diff. Render	Peak Find	Propr. Dist.
(b) Red drywall*	**2.05**	3.09	4.68
(e) White paper	**2.51**	3.45	63.0
(f) Black fabric	**2.51**	3.35	72.5
(h) Patterned rug	**2.69**	3.62	62.7
(c) Wood	4.19	**4.03**	60.7
(d) Whiteboard	**5.12**	5.82	54.9
(g) Checkerboard	6.94	**4.12**	61.1
(a) Aluminum foil	**12.7**	15.0	25.3

TABLE III: **Our methods are generally robust to surface properties, aside from highly specular aluminum foil.** Images in range 1-30cm, 0-30° AoI. *measurements of red drywall are used to optimize forward model of differentiable method and to calibrate peak finding and proprietary distance approaches.

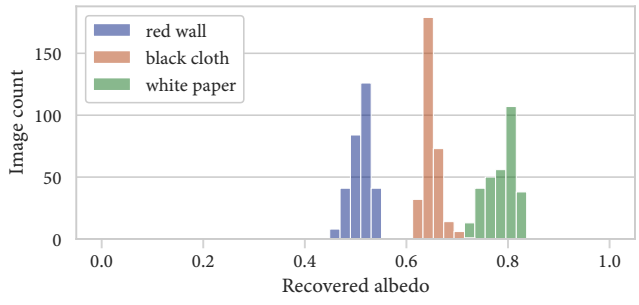

Fig. 6: **Our method recovers consistent surface albedo under various distances and angles-of-incidence.** Recovered albedo is in the wavelength of the sensor light source (940nm IR), and may vary significantly from the albedo as it appears to the human eye under visible light. Each surface is observed from 300 poses in range 7-40cm, 0-30° AoI.

of the wavelength of the sensor light source, and surfaces with a known albedo in that wavelength, which is outside the scope of this work. We find that this method recovers a consistent albedo per-surface relatively invariant to distance in the range 7-40cm and angle-of-incidence in the range 0-30°, allowing discrimination between surfaces, as shown in Figure 6.

VIII. Example Application

We build an application to showcase our methods, in which a robot arm is holding a cup of liquid. The robot's goal is to safely place the cup on a tabletop below, which is at an unknown distance and may have regions which are not level. In our application, we attach a TMF8820 transient sensor directly to the gripper of the robot arm. Due to its small size, the sensor can be placed centimeters away from the jaws of the gripper, where it senses the surface below directly, making it invulnerable to occlusions.

Using our approach for recovering planar geometry, the robot is able to sense the distance to and slope of the surface below the cup being held in the end effector, as shown in Figure 7. The robot uses this information to know when it is close enough to the surface to place the cup down, and to ensure that the surface is level enough to safely release the cup.

IX. Limitations

While the differentiable rendering method given in this work can in principle recover any unknown parameters to the render function, we only evaluate recovery of scene geometry and albedo. A next step is to investigate recovery of the reflectance model parameters of a surface. While our method in principle enables such recovery, evaluation is difficult. Another next step is recovering other types of geometry. Our approach can in principle easily be adapted to other parameterized surfaces, *e.g.*, a sphere or cube. Extending to arbitrary geometry would require a more general differentiable representation, *e.g.* a neural representation akin to NeRF [30]. As both of these tasks introduce extra degrees of freedom into the optimization process, they may require a more accurate and/or sophisticated model of the transient histogram imaging process to sufficiently constrain optimization.

One challenge for future work is the low bandwidth available on commodity sensors. In our test setup, histograms are read from the sensor at 4.5 frames per second (where one "frame" consists of nine histograms) despite the sensor generating proximity estimates at 150Hz. This is not a limitation of

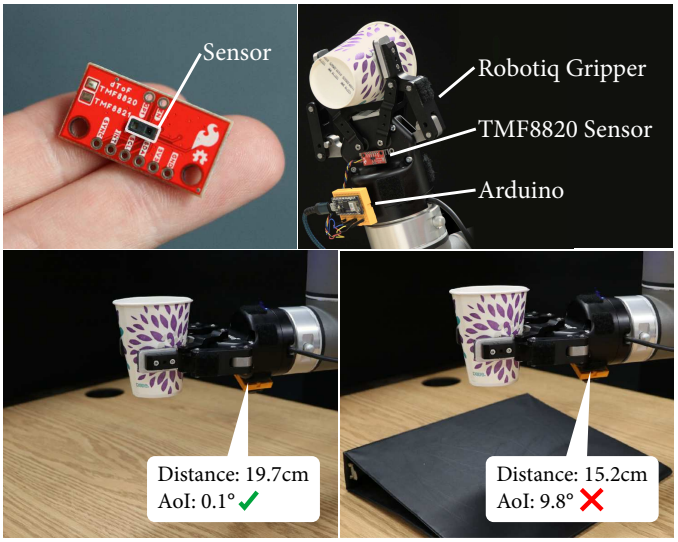

Fig. 7: We mount a proximity sensor on a robot gripper (top). The sensor detects when the surface below the gripper is level and safe to place a cup full of liquid (bottom left) or is not level and therefore unsafe (bottom right).

the sensor technology, but of the I^2C interface over which it transmits data. We hope that commodity SPAD sensors will in the future come packaged with high bandwidth interfaces to enable granular and high-speed sensing. Algorithms will also need to be optimized to perform inference quickly enough to keep pace with higher bandwidth sensors.

Lastly, we provide only a basic demonstration of utilizing transient histograms in a robotics setting. It is yet to be shown that utilizing these histograms leads to improvement in performance of downstream robotics tasks. An important next step is to build a complete robotics system which utilizes transient histogram data, and evaluate the system performance compared to alternative sensing modalities. We are hopeful that future robotics systems will harness the power of transient histograms to be highly aware of their environment on a low size, weight, and power budget.

REFERENCES

[1] A. O. AG, *TMF882X Datasheet*, AMS OSRAM AG. [Online]. Available: https://ams.com/tmf8820#tab/documents

[2] S. Microelectronics, *VL6180X Proximity and Ambient Light Sensing Module Datasheet*, ST Microelectronics. [Online]. Available: https://www.st.com/resource/en/datasheet/vl6180x.pdf

[3] S. Cova, M. Ghioni, A. Lacaita, C. Samori, and F. Zappa, "Avalanche photodiodes and quenching circuits for single-photon detection," *Applied optics*, vol. 35, no. 12, pp. 1956–1976, 1996.

[4] S. Pellegrini, G. S. Buller, J. M. Smith, A. M. Wallace, and S. Cova, "Laser-based distance measurement using picosecond resolution time-correlated single-photon counting," *Measurement Science and Technology*, vol. 11, no. 6, p. 712, 2000.

[5] A. K. Pediredla, A. C. Sankaranarayanan, M. Buttafava, A. Tosi, and A. Veeraraghavan, "Signal processing based pile-up compensation for gated single-photon avalanche diodes," *arXiv preprint arXiv:1806.07437*, 2018.

[6] A. Gupta, A. Ingle, and M. Gupta, "Asynchronous single-photon 3d imaging," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7909–7918.

[7] S. Tsuji and T. Kohama, "Proximity skin sensor using time-of-flight sensor for human collaborative robot," *IEEE Sensors Journal*, vol. 19, no. 14, pp. 5859–5864, 2019.

[8] O. A. Adamides, A. S. Modur, S. Kumar, and F. Sahin, "A time-of-flight on-robot proximity sensing system to achieve human detection for collaborative robots," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 1230–1236.

[9] C. Escobedo, M. Strong, M. West, A. Aramburu, and A. Roncone, "Contact anticipation for physical human–robot interaction with robotic manipulators using onboard proximity sensors," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 7255–7262.

[10] S. Tsuji and T. Kohama, "Omnidirectional proximity sensor system for drones using optical time-of-flight sensors," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 17, no. 1, pp. 19–25, 2022.

[11] C. Sifferman, D. Mehrotra, M. Gupta, and M. Gleicher, "Geometric calibration of single-pixel distance sensors," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6598–6605, 2022.

[12] M. Selvaraj, V. Baltzopoulos, A. Shaw, C. N. Maganaris, J. Cullen, T. OBrien, and P. Kot, "Stair fall risk detection using wearable sensors," in *2018 11th International Conference on Developments in eSystems Engineering (DeSE)*. IEEE, 2018, pp. 108–112.

[13] D. Marko and D. Hrubỳ, "Distance measuring in vineyard row using ultrasonic and optical sensors," in *Proceeding of 22 nd International Conference of Young Scientists. Praha: Česká zemědělská univerzita*, 2020, pp. 194–204.

[14] A. Velten, T. Willwacher, O. Gupta, A. Veeraraghavan, M. G. Bawendi, and R. Raskar, "Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging," *Nature communications*, vol. 3, no. 1, p. 745, 2012.

[15] M. Buttafava, J. Zeman, A. Tosi, K. Eliceiri, and A. Velten, "Non-line-of-sight imaging using a time-gated single photon avalanche diode," *Optics express*, vol. 23, no. 16, pp. 20 997–21 011, 2015.

[16] S. Shen, Z. Wang, P. Liu, Z. Pan, R. Li, T. Gao, S. Li, and J. Yu, "Non-line-of-sight imaging via neural transient fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 7, pp. 2257–2268, 2021.

[17] F. H. Clara Callenberg, Zheng Shi and M. B. Hullin, "Low-cost spad sensing for non-line-of-sight tracking, material classification and depth imaging," *ACM Trans. Graph. (SIGGRAPH)*, vol. 40, no. 4, 2021.

[18] C. N. Becker and L. J. Koerner, "Plastic classification using optical parameter features measured with the tmf8801 direct time-of-flight depth sensor," *Sensors*, vol. 23, no. 6, p. 3324, 2023.

[19] A. Ruget, M. Tyler, G. Mora Martín, S. Scholes, F. Zhu, I. Gyongy, B. Hearn, S. McLaughlin, A. Halimi, and J. Leach, "Pixels2pose: Super-resolution time-of-flight imaging for 3d pose estimation," *Science Advances*, vol. 8, no. 48, p. eade0123, 2022.

[20] L. Bian, H. Song, L. Peng, X. Chang, X. Yang, R. Horstmeyer, L. Ye, T. Qin, D. Zheng, and J. Zhang, "Large-scale single-photon imaging," *arXiv preprint arXiv:2212.13654*, 2022.

[21] A. Ruget, M. Tyler, G. M. Martín, S. Scholes, F. Zhu, I. Gyongy, B. Hearn, S. McLaughlin, A. Halimi, and J. Leach, "Real-time, low-cost multi-person 3d pose estimation," *arXiv preprint arXiv:2110.11414*, 2021.

[22] S. Jungerman, A. Ingle, Y. Li, and M. Gupta, "3d scene inference from transient histograms," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VII*. Springer, 2022, pp. 401–417.

[23] B. T. Phong, "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no. 6, pp. 311–317, 1975.

[24] A. Gupta, A. Ingle, A. Velten, and M. Gupta, "Photon-flooded single-photon 3d cameras," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6770–6779.

[25] F. Mu, S. Mo, J. Peng, X. Liu, J. H. Nam, S. Raghavan, A. Velten, and Y. Li, "Physics to the rescue: Deep non-line-of-sight reconstruction for high-speed imaging," *arXiv preprint arXiv:2205.01679*, 2022.

[26] C. Urea, M. Fajardo, and A. King, "An area-preserving parametrization for spherical rectangles," *Computer Graphics Forum*, vol. 32, no. 4, pp. 59–66, 2013. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12151

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[28] C. Brown, "Principal axes and best-fit planes, with applications," 1976.

[29] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Comput. J.*, vol. 7, pp. 308–313, 1965.

[30] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.