

Efficient Detection of Objects Near a Robot Manipulator via Miniature Time-of-Flight Sensors

Carter Sifferman¹, Mohit Gupta¹, and Michael Gleicher¹

Abstract—We provide a method for detecting and localizing objects near a robot arm using arm-mounted miniature time-of-flight sensors. A key challenge when using arm-mounted sensors is differentiating between the robot itself and external objects in sensor measurements. To address this challenge, we propose a computationally lightweight method which utilizes the raw time-of-flight information captured by many off-the-shelf, low-resolution time-of-flight sensor. We build an empirical model of expected sensor measurements in the presence of the robot alone, and use this model at runtime to detect objects in proximity to the robot. In addition to avoiding robot self-detections in common sensor configurations, the proposed method enables extra flexibility in sensor placement, unlocking configurations which achieve more efficient coverage of a radius around the robot arm. Our method can detect small objects near the arm and localize the position of objects along the length of a robot link to reasonable precision. We evaluate the performance of the method with respect to object type, location, and ambient light level, and identify limiting factors on performance inherent in the measurement principle. The proposed method has potential applications in collision avoidance and in facilitating safe human-robot interaction.

Project Page: https://cpsiff.github.io/efficient_detection/

I. INTRODUCTION

DETECTION of objects near a robot arm is useful for tasks such as collision avoidance [1], [2] or to enable proximity-based human-robot interactions [3]. Externally mounted cameras are one way of detecting such objects, but they suffer from occlusion and require the robot to remain in view of the cameras, limiting their practicality when used with mobile manipulators. Therefore, we seek a solution which uses sensors mounted on the robot. Miniature time-of-flight (ToF) sensors [4], [5], [6], [7] are particularly attractive for mounting on-robot because of their small size and low power consumption. In order to cover the space around the robot with a small number of sensors, we need the ability to choose efficient sensor configurations, such as placing the sensor peering down the length of an arm segment, as shown in Fig. 1F. However, miniature ToF sensors have very low pixel counts (*e.g.*, 3x3) with a wide field-of-view (FoV) per-pixel (5° - 40°), meaning that in such configurations, the robot is constantly detected, and other objects can only be detected if they are closer than the robot is to the ToF sensor. As illustrated in Fig. 2, simple filtering of pixels which view the

Manuscript received: May 12, 2025; Revised August 17, 2025; Accepted September 14, 2025.

This paper was recommended for publication by Editor Pascal Vasseur upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by NSF Grant 2152163.

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

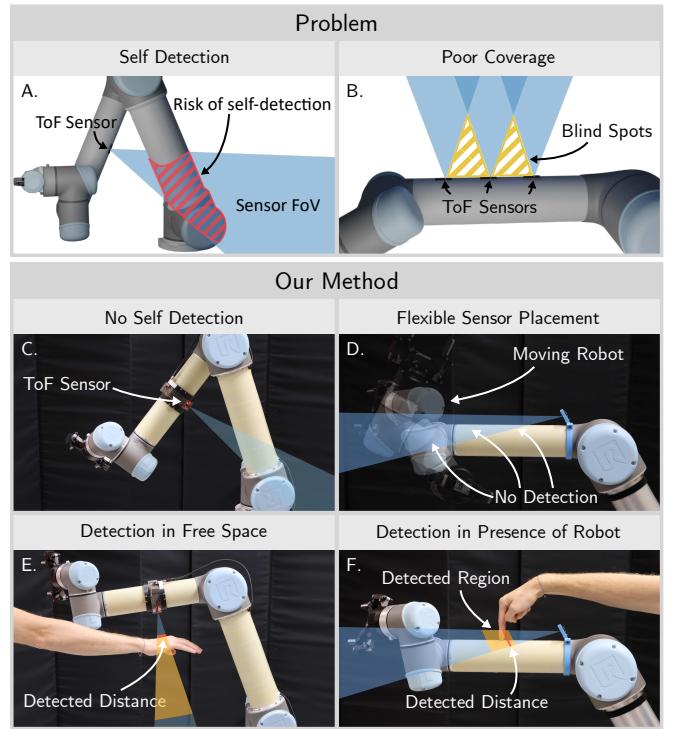


Fig. 1: Time-of-flight sensors attached to robot arms are prone to self-detection (A), and typical configurations provide inefficient coverage of the radius near the robot surface (B). Our method enables self-detection free proximity sensing, which enables new sensor configurations that provide more efficient coverage of a radius around the robot surface (C-F).

robot is ineffective in this case; therefore, to enable such sensor configurations, any approach to object detection must be able to differentiate between *self-detections* (the robot itself) and external object detections *within each pixel*.

In this work, we provide a method for detecting and localizing objects near a robot arm using miniature ToF sensors. Our method allows for flexibility in sensor placement and avoids self-detections of the robot. We address the key challenge of differentiating robot self-detections from other objects by implicitly modeling the expected appearance, reflectance, and geometry of the robot through sampling of raw ToF measurements. We utilize the raw ToF data captured by commonly used off-the-shelf sensors; at runtime our method finds differences between the measured ToF data and the expected appearance of the robot. Therefore, our method can detect objects even in sensor pixels for which the robot is prominent in view. This enables configurations such as that shown in Fig. 1F, which provide coverage of a small radius around the robot surface using few sensors. Our method also prevents self-detection in more typical outward facing sensor configurations as shown in Fig. 1C.

Our contributions are: 1) a method for detection and localization of objects near a robot arm with a known joint state using a miniature ToF sensor while ignoring robot self-detections; 2) experiments demonstrating that our method is effective at detecting and estimating distance to objects with the configuration shown in Fig. 1F, and experiments investigating the limits and inherent constraints on the performance of our method; and 3) a live demonstration.

Scope and Limitations. While our method can scale to multiple sensors, in this work we build a prototype which includes one sensor at a time. Our demonstration shows live output of our method, but is not integrated with robot control for *e.g.*, collision avoidance, and the sensor frame rate is limited to 3.5 Hz by the data interface of currently available sensors. Our method enables sensor configurations which efficiently cover a small radius around the robot surface, and is computationally efficient at runtime, but requires one-time overnight reference data capture per sensor position. Additionally, our method foregoes the need for any geometric calibration of sensor position, which is required by most alternative methods.

II. RELATED WORK

A. Whole-Robot Proximity Detection

Research on robotic “sensitive skin” comprised of touch or proximity sensors dates back to the 1980s [8], [9]. While tactile sensors [10], [11], [12] are useful for collision detection and human-robot interaction, the ability to sense objects before touch occurs (proximity detection) enables a different set of applications including collision avoidance and safety in human-robot interaction. Systems have been proposed for whole-robot proximity detection, including those based on optical ToF [13], [14], [15], [16], or other sensing principles [17], [18]. These works do not address the problem of self-detection directly. Therefore, they are limited to outward facing sensor configurations. Our work demonstrates using flexibility in sensor placement to enable novel configurations that cover regions efficiently.

B. Avoiding Self-Detection

Self-detection, when the robot itself is detected as an external object, is a challenging problem for robot manipulator perception systems due to the manipulator’s dynamic shape during operation. There exists work on avoiding the self-detection problem when using an external depth camera which provides a 3D point cloud, by filtering out points belonging to the robot. Some approaches rely on extrinsic calibration between the camera and the robot, and use a 3D model of the robot to simulate its expected signal in the point cloud [19], [20]. Other works do not rely on extrinsic calibration, recognizing and removing the robot from the point cloud directly [21], or using temporal cues along with proprioception [22], [23]. These approaches are a reasonable solution for high resolution point clouds; when the robot points are removed, there is still sufficient information remaining to avoid collisions. However, with a low resolution sensor, filtering point clouds is not effective because the robot may be

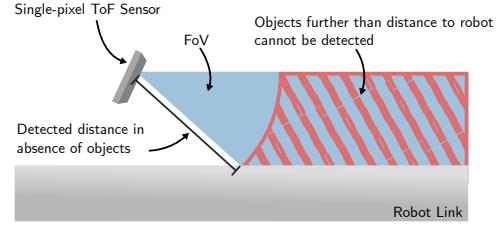


Fig. 2: When sensor measurements are treated as single distances per-pixel, objects further than the detected distance to the robot cannot be detected. This leads to significant blind spots, precluding sensor configurations such as the one shown. Our work enables such sensor configurations.

visible in all or nearly-all pixels. Therefore objects can only be detected when they are closer than the detected distance to the robot in a given pixel, severely limiting the effective detection area for some sensor configurations. A visualization of this limitation is shown in Fig. 2.

There is little prior work on avoiding self-detection with arm-mounted proximity sensors. The works which address the problem directly follow an approach similar to that used for point clouds. Avanzini *et al.* [24] place distance sensors on the links of a robot arm. To avoid self-detections, they use a 3D model of the robot to simulate the expected distance reading if only the robot were present. If the distance reading is less than the simulated reading, an object is detected. Himmelsbach *et al.* [25] use a similar approach. Such an approach means the sensor will not see objects within its FoV which are further than the simulated distance estimate, again being prone to the problem shown in Fig. 2.

C. Miniature Time-of-Flight Sensors

Miniature ToF sensors are widely used in robotics due to their small size and low power requirements. Applications have been developed for the sensors mounted on miniature drones [26], [27], [28], [29], [30]. Other works place a sparse set of sensors around a robot [3], [16] or at a robot wrist [31], and build applications for collision avoidance. There exist methods which use the sensors to detect the 3DoF pose of a planar surface [5], and methods for calibrating the extrinsic position of a sensor attached to a robot arm [32]. This work builds on previous work which provides a method for detecting geometric deviations on a planar surface [33]. In contrast to this previous work, the method presented in this paper works for articulated robots and non-planar surfaces, and is able to determine the distance to unknown objects.

There is a body of research which aims to make use of the raw ToF data from low-cost sensors akin to the one used in this work. Callenberg *et al.* [4] demonstrate in-contact material classification and, utilizing additional hardware, high-resolution imaging and non-line-of-sight tracking. Other works look to recover more detailed 3D information from the low-resolution measurements of these sensors. There exist approaches for recovering 3D human pose [34], high resolution depth images [35], and general 3D reconstruction (from a distributed set of sensors) [36]. Miniature ToF sensors have also been used to refine monocular depth estimates [37], [38] and augment RGB SLAM [39]. Aforementioned work in robotics also takes advantage of raw ToF data [5], [33].

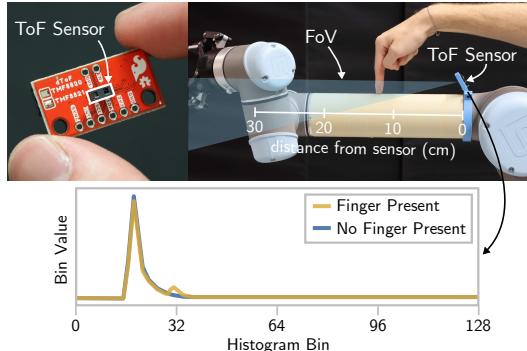


Fig. 3: AMS TMF8820 sensor used in this work, and an example histogram captured by the sensor mounted on a robot arm. The large, leftmost peak is due to the surface of the robot link. The presence of the finger causes the appearance of an additional peak.

III. PROBLEM OVERVIEW

A. Background: Direct Time-of-Flight

The miniature direct ToF sensors that we utilize operate by illuminating a patch of the scene with a pulse of (typically infrared) light and directly measuring the time of travel of the returning light at high (nano- to picosecond) time resolution. The returning light waveform is called the *scene transience*, and the quantized version of that waveform recorded by the sensor is the *transient histogram* [37], [40]. The transient histogram is a function of scene geometry and reflectance properties (in addition to sensor and light-source characteristics) integrated over the FoV of the sensor which, for miniature sensors, is typically between 10 and 40 degrees per pixel. Single photon avalanche diodes (SPADs) [41], [42] are the most mature and widely available technology enabling direct ToF, and the basis of currently available miniature direct ToF sensors. These sensors are very small ($<20 \text{ mm}^3$), lightweight ($<1 \text{ gram}$), and power efficient ($<10 \text{ milliwatts}$ per measurement) [6], [7], [43].

Typically, miniature ToF sensors use an onboard algorithm to calculate a distance estimate from the transient histogram, which is reported. The goal of this work is to recover the distance to the closest point on unknown geometry while ignoring the robot itself. To accomplish this, our method utilizes the transient histogram directly, rather than on-sensor distance estimates. An example of a transient histogram is shown in Fig. 3. Using the transient histogram directly allows us to pick out subtle variations in the measured ToF signal in an individual pixel which are not contained in on-sensor distance estimates alone. In §V-E, we demonstrate that on-sensor distance estimates are not sufficient.

B. Problem Analysis

In order to estimate the distance to unknown objects in a transient histogram while ignoring the robot itself, we first must identify and remove the signal caused by the robot. In the case of an articulated robot, the geometry of the robot varies with respect to the robot joint state. Our goal is then to create a mapping from robot joint state to a probabilistic model of the expected transient histogram, as it would appear if only the robot were present. This mapping could be achieved

in multiple ways. Previous work [36], [5] demonstrated an effective forward model for miniature direct ToF sensors, which allows simulation of a histogram measurement given scene geometry and reflectance (*i.e.*, surface albedo and specularity). However, in order to accurately simulate sensor measurements of the robot, the spatially varying reflectance properties of the robot would need to be known. Gathering such a measurement requires highly specialized equipment, making a simulation-based approach impractical. Further, previous work has established that detecting objects based on known geometry but unknown reflectance is fundamentally ambiguous under many settings [33], [37].

Rather than modeling the ToF signal of the robot explicitly, we utilize a data-driven approach in which measurements from the sensor are sampled at many robot states with only the robot present. Those measurements are used to create a probabilistic model of the expected transient histogram for any single joint state within the sampled range. Our method is applicable to any direct ToF sensor which reports a transient histogram. We evaluate our method using the AMS TMF8820, shown in Fig. 3.

IV. METHOD

Given a b -bin transient histogram $\mathbf{h}_{\text{obs}} \in \mathbb{N}^b$ captured by a miniature ToF sensor attached to an n degree-of-freedom robot with joint state $\mathbf{q} \in \mathbb{R}^n$, we aim to recover the distance d to the point nearest the sensor on any object in the sensor FoV excluding the robot itself (and any attached accessories, *e.g.*, a gripper). In practice, depending on the mounting position of the sensor, the sensor may not be able to see every link of the robot. In this case only degrees-of-freedom which affect sensor readings are included in \mathbf{q} .

A. Histogram Pre-Processing

Transient histograms are affected by ambient light, which manifests as a DC offset in the captured signal [44]. To avoid falsely detecting changes in ambient light as objects, we pre-process histograms by subtracting the DC offset and normalizing the area under the signal, following the approach of previous work [33]. For the histogram \mathbf{h} the DC offset h_{offset} induced by ambient light is approximated by finding the maximum kernel density on the values of \mathbf{h} , which acts as a robust way of estimating the modal value of \mathbf{h} . The kernel bandwidth σ is a tuneable parameter, which can vary by sensor model:

$$h_{\text{offset}} = \underset{x}{\operatorname{argmax}} \sum_{h_i \in \mathbf{h}} \mathcal{N}(x; h_i, \sigma) \quad (1)$$

The area under the signal is normalized after h_{offset} is subtracted. The pre-processed histogram $\tilde{\mathbf{h}}$ is given by:

$$\tilde{\mathbf{h}} = \frac{\mathbf{h} - h_{\text{offset}}}{\|\mathbf{h} - h_{\text{offset}}\|_1} \quad (2)$$

B. Modeling Known Objects

To detect the distance to unknown objects in the FoV imaged by \mathbf{h}_{obs} , we rely on a probabilistic model of the per-bin mean $\mu_{\mathbf{q}} \in \mathbb{R}_+^b$ and per-bin variance $\sigma_{\mathbf{q}} \in \mathbb{R}_+^b$ of the expected

histogram if only known objects were in the sensor FoV, given the current joint state \mathbf{q} . For reasons explained in §III-B, we approximate $\mu_{\mathbf{q}}$ and $\sigma_{\mathbf{q}}$ by interpolating between real samples over a range of possible \mathbf{q} rather than an analytical approach.

Our method requires a set of per-bin histogram means \mathbf{M} , variances \mathbf{V} and corresponding joint angles \mathbf{J} which sample the robot configuration space. In practice, this dataset can be captured by, *e.g.*, grid search or random sampling in configuration space. For each joint position sampled, multiple histograms are captured to generate a good approximation of \mathbf{V} to capture sensor noise. Details of how we perform this sampling for real-world experiments are given in §V. To approximate $\mu_{\mathbf{q}}$ and $\sigma_{\mathbf{q}}$ from samples in \mathbf{M} and \mathbf{V} , we perform barycentric interpolation, which requires finding a convex hull of $n + 1$ points around \mathbf{q} in \mathbf{J} . $\mu_{\mathbf{q}}$ and $\sigma_{\mathbf{q}}$ are then interpolated between the corresponding values in \mathbf{M} and \mathbf{V} . The importance of interpolation and the density of samples needed to achieve good approximation of $\mu_{\mathbf{q}}$ and $\sigma_{\mathbf{q}}$ is investigated in §V-B.

C. Detecting Distance to Unknown Objects

Given $\mu_{\mathbf{q}}$ and $\sigma_{\mathbf{q}}$, we calculate the normalized probability vector $\mathbf{p} \in \mathbb{R}_+^b$ for \mathbf{h}_{obs} , which encodes the per-bin likelihood that a given bin is in the distribution expected when the robot alone is present, normalized so that for a given bin index i , when $h_{\text{obs},i} = \mu_{\mathbf{q},i}$ the likelihood is 1:

$$p_i = e^{-\frac{(h_{\text{obs},i} - \mu_{\mathbf{q},i})^2}{2(\sigma_{\mathbf{q},i})^2}} \quad (3)$$

To detect objects and estimate their distance, we transform \mathbf{p} to a binary vector \mathbf{g} which encodes bins that are likely to contain an unknown object. The threshold for detection t is a hyper-parameter which can be tuned to adjust sensitivity:

$$g_i = \begin{cases} 1 & \text{if } p_i < t \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

We then search for segments of the value 1 in \mathbf{g} which span c or more contiguous bins, each of which corresponds to one detected object. For each segment, we extract the values of \mathbf{h} over the corresponding range. We find the peak in these extracted values of \mathbf{h} . The position (bin index) of this peak corresponds to the distance to the detected object. We convert from bin index i_{peak} to distance using the conversion from bin index to distance for the TMF8820 sensor established by previous work [5]: distance (m) = $0.01387i_{\text{peak}} - 0.1825$. We empirically observe that this calibration is stable between multiple instances of the same sensor model.

D. TMF8820 Calibration

We observe that a varying bias is applied to TMF8820 histogram measurements between sensor power cycles. This bias can lead to false positives if the sensor is cycled off after the reference measurements (\mathbf{M} , \mathbf{V} and \mathbf{J}) are captured. While we do not know the exact cause of this effect, or if it applies to other ToF sensors, we are able to mitigate it by performing a one-off calibration step every time the sensor

is powered on after reference capture. We move the robot to the first reference joint position \mathbf{J}_1 and capture a set of 50 measurements, which we average per-bin and store as \mathbf{h}_{ref} . We then calculate $\mathbf{h}_{\text{calib}} = \mathbf{M}_1 - \mathbf{h}_{\text{ref}}$. $\mathbf{h}_{\text{calib}}$ is stored and at query time is added to \mathbf{h} prior to Eqn. 1.

V. EXPERIMENTAL RESULTS

A. Implementation Details

We perform a series of experiments in which a TMF8820 sensor is attached to link two of a Universal Robots UR5 robot arm. The sensor is positioned facing the end effector, as shown in Fig. 3. In this position, sensor readings are invariant to movement in the three most proximal joints. Thus, we only sample the 3DoF of the three most distal joints (*i.e.*, those comprising the wrist) to capture reference histograms. Each experiment aside from §VI-I relies on the same reference dataset, which is captured over a 3D grid in joint space, in which $\mathbf{q}_4 \in [-\pi, -\pi/12]$, $\mathbf{q}_5 \in [-5\pi/6, 5\pi/12]$, $\mathbf{q}_6 \in [-\pi/2, 5\pi/12]$. Joint positions are sampled in $\pi/12$ radian increments, for a total of 2,304 joint positions. It takes ~ 10 hours to programmatically capture 50 measurements per joint position. The brushed aluminum surface of the UR5 robot is highly specular. While our data-driven approach models the effect of the specular surface when the robot alone is present, when other objects are present outside of the sensor field-of-view the surface acts like a mirror. This leads to false detections when objects outside of the sensor FoV are detected via three-bounce paths. We cover the metallic surfaces of the robot in masking tape to minimize this effect, and further investigate in §V-G

We capture data from an AMS TMF8820 sensor connected to an Arduino microcontroller, using the microcontroller code provided by prior works [5], [36], [33] to extract both distance and histogram measurements. The TMF8820 reports 9 histograms over 9 non-overlapping zones, for a total FoV of 30° diagonally. For simplicity and to limit the FoV to avoid unwanted detections, we utilize only one zone of the sensor, yielding a 10° diagonal FoV. Sensor frame rate is limited by the speed at which the I²C interface, which is not designed for histogram data capture, can transmit histograms, so we modify the microcontroller code to only report bins 1-80 to increase frame rate. This means the maximum range of the sensor as configured is ~ 90 cm. The sensor reports measurements at 3.5 FPS. The execution of our algorithm takes 0.35 ms (2803 FPS) on a mid-range laptop CPU (Intel i5 1340P), and the runtime scales linearly with the number of sensors used. In our testing, interference between multiple TMF8820 sensors is minimal, making them well-suited to future systems with many sensors.

Unless otherwise stated, we set the probability threshold $t = 0.001$, and minimum segment size $c = 4$. These values were manually tuned to create a reasonably low false positive rate for the main experiments. We investigate the effect of changing these parameters in §V-H. A peak in bin ~ 14 from the TMF8820 corresponds to an object at distance zero; therefore we trim the histogram $\tilde{\mathbf{h}}$ to bin range (15, 80) before applying Eqn. 3. We set σ in Eqn. 1 to 5, following

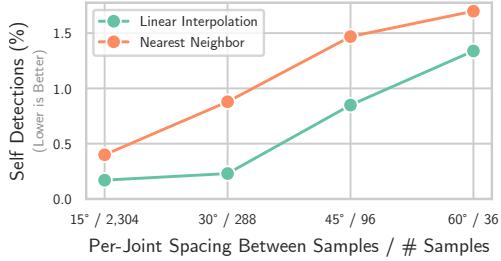


Fig. 4: Effect of joint-space sampling density on self-detection (false positive) rate. Linear interpolation of background histograms between nearby joint states outperforms nearest neighbor interpolation (lower is better). 3 DoF of robot wrist joints are sampled.

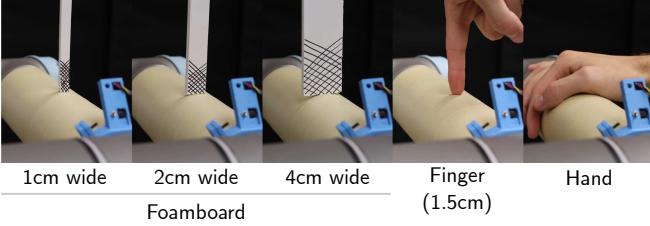


Fig. 5: Objects used in experiments, shown as placed on the robot for data capture. Pieces of foamboard have a hatching pattern applied to provide visual features for ground-truth depth-from-stereo camera.

previous work [33]. In each experiment, we only consider the *closest* detection. All captures are in a windowless room with fluorescent lights (~ 500 lux).

B. Self-Detection Rate

We perform an experiment to understand the effect of joint-space sampling density on the rate at which the robot is falsely detected. The results of this experiment are highly dependent on robot geometry and sensor position. The experiment serves to provide a rough approximation of performance in general, and provides context for other experiments, which use the same robot and the same sensor position.

We capture a dataset of ToF measurements from 1000 uniformly sampled random joint positions within the joint range of the reference dataset, with only the robot present. The self-detection rate (false positive rate) is the rate at which detections occur in this dataset. To investigate the effect that sampling density has on self-detection rate, we sub-sample the reference dataset by a factor of 2, 3, and 4 per dimension, creating a coarser grid of samples on which the model is built, and plot the effect on self-detection rate in Fig. 4. Self-detection rate increases as the density of joint-space samples decreases, and linear interpolation leads to a lower self-detection rate than nearest neighbor interpolation. Self-detection rate levels off at high sampling densities; we hypothesize that this is because in rare cases measurement noise leads to self-detections, and measurement noise is constant regardless of sample density. A coarser sampling of every 30° achieves similar performance to 15° while requiring an order of magnitude fewer samples. Coupled with future sensors with a higher frame rate, this means that reference data capture could be made orders of magnitudes faster for little performance penalty.

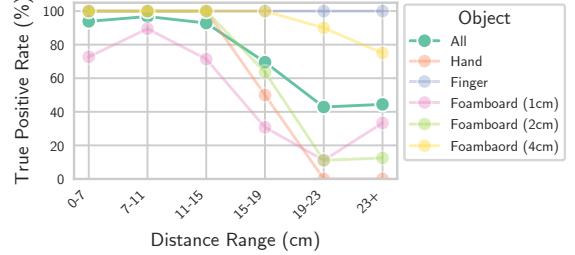


Fig. 6: True positive rate as a factor of the distance from the sensor to the object, broken down by object type. Note that due to random object placement, each data point may not represent the same number of samples.

C. True Positive Rate

To evaluate the true positive rate of our method (*i.e.*, the rate at which an object is detected when one is present), we capture a dataset of ToF measurements in which objects are touching or nearly touching the robot arm. Between each measurement, the robot is moved to a random uniformly sampled joint state within the bounds of the reference dataset, and moved the object to a random distance from the sensor along the robot arm (1-28cm from the sensor); random distances are used to make data capture faster, ultimately allowing a larger dataset. We utilize five objects: a human pointer finger, a human hand, and long pieces of white foamboard cut to 1cm, 2cm, and 4cm in width. The objects are shown in Fig. 5. The finger and hand were captured touching the robot arm, while each piece of foamboard was captured at 0cm, 1cm, and 2cm proximity from the arm itself. In total, the dataset contains 275 captures, each with varying conditions (object, distance to sensor, and proximity to arm).

We achieve a true positive rate of 78.9%; this is broken down by object and distance to the sensor in Fig. 6. The 4cm wide foamboard and pointer finger are the easiest to detect at all distances, while the narrower foamboard and hand are the most difficult. The hand, 1cm, and 2cm-wide foamboard are rarely recognized beyond 19cm from the sensor. We hypothesize that the difference between objects is due to a difference in their cross-sectional area and geometric deviation from the robot. While the hand is a large object, its cross sectional area is small from the point-of-view of the sensor when the hand is resting flat on the robot, and the hand does not extend far above the robot surface, leading to a relatively small change in the histogram. Object albedo might also play a factor; an object which is much brighter or darker than the robot will cause a larger change in the measured histogram than one with the same albedo. Lastly, non-line-of-sight effects (see §V-G) caused by the rest presence of the wrist and rest of the arm above the finger may make it easier to detect than the narrow foamboard. Future work should aim to isolate and negate the cause of the performance gap between objects.

D. Distance Estimation

We use the same dataset as in the previous subsection (§V-C) to evaluate the accuracy of our distance estimate. Ground truth distance labels are captured via an Intel Realsense D405 depth-from-stereo camera positioned next to the ToF sensor. The closest point on each object is labeled, and the depth estimate extracted from the D405 depth image. Objects

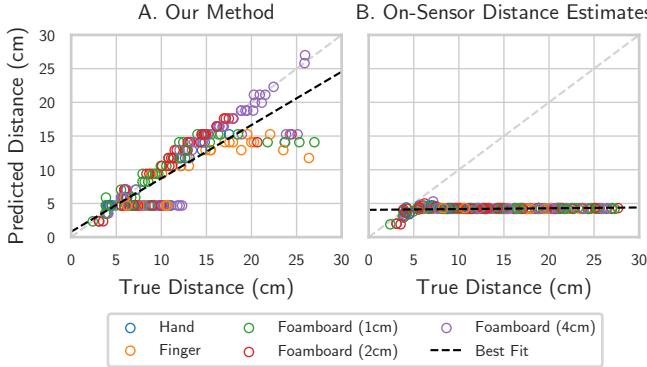


Fig. 7: A. Actual distance vs. distance predicted by our method. We achieve an average distance error of 2.08cm. B. Actual distance vs. distance predicted by a baseline method which utilizes on-sensor distance estimates. The baseline method never estimates a distance further than 5cm due to the limitation illustrated in Fig. 2.

closer than the minimum depth range of the camera are labeled by overlaying an image of the robot arm with ruler marks onto the captured RGB image of the object.

Fig. 7A compares the distance estimate from our method to the actual distance. We achieve an average absolute error of 2.08cm. Our method under-estimates the distance to objects in some cases. One case is when a nearby object fills a large portion of the FoV, completely changing the shape of the histogram. When this happens, it is difficult to align the observed histogram to the reference to localize the deviation. Our method also sometimes under-estimates the distance to far away objects. We hypothesize that this could be due to a low signal-to-noise ratio, and/or the presence of the dynamic robot wrist links at those distances.

E. On-Sensor Distance Estimation

We compare the distance estimation results achieved by on-sensor distance estimates to our method. The TMF8820 reports up to two distance estimates per-zone, corresponding to up to two objects in the FoV. For each sample we choose the distance estimate of the two which achieves the lowest absolute error from the ground truth to demonstrate the best case for on-sensor distance estimates. In Fig. 7B, we show the distance estimated by this method compared to the true distance. The distance estimates never exceed 5cm, roughly the distance to the nearest point on the robot. This experiment demonstrates the limitation illustrated in Fig. 2 and makes it clear that the results achieved by our method cannot be achieved using on-sensor distance estimates alone.

F. Sensor Field-of-View

We characterize the FoV of the sensor by placing the 4cm and 2cm foamboard at varying distance from the sensor and proximity to the arm and plot the TPR per object position in Fig. 8. The sensor is 4cm from the robot surface, with the top edge of the FoV aligned with the surface. Accordingly, we see that detection is less likely at 4cm from the surface and much less likely at 6cm+. The maximum detection proximity is consistent across all distances from the sensor, making the sensor configuration effective for detecting objects that come within 4cm of the robot surface.

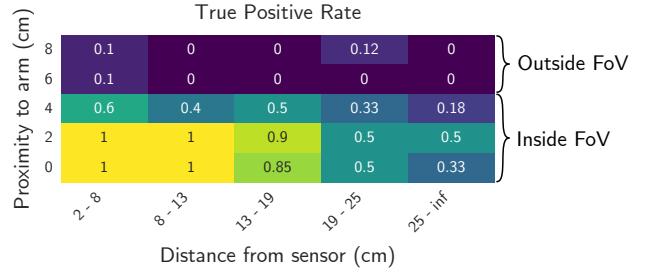


Fig. 8: Detection rate as a factor of object distance from the sensor and proximity to the arm. We see that objects beyond 4cm proximity to the arm are rarely detected, and that the proximity of detection is consistent regardless of distance from the sensor. This well-defined field-of-view is desirable for downstream applications.

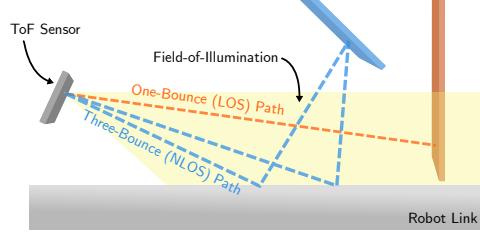


Fig. 9: There is an ambiguity between distant line-of-sight (LOS) objects and nearer non-line-of-sight (NLOS) objects outside of the sensor field-of-illumination. This limits the performance of our method on distant objects. If the detection threshold is lowered to determine distance LOS objects, nearer NLOS objects will also be detected.

G. Non-Line-of-Sight Objects

Direct ToF sensors are subject to non-line-of-sight (NLOS) effects, which occur when photons bounce multiple times before returning to the sensor, as illustrated by the blue path in Fig. 9. As previously noted, this effect is very noticeable with the brushed aluminum robot surface. We cover the arm with masking tape to lower this effect for our experiments, but it is still present. In this subsection, we investigate the prevalence of NLOS effects in our setting.

We investigate NLOS effects over the previously captured dataset (§V-F). We treat objects outside of the direct FoI of the sensor (*i.e.*, , 6cm and 8cm proximity to the arm; the top two rows in Fig. 8) as NLOS objects. We compare false positive rate between this set of objects and the case where only the robot is present. As shown in Fig. 10, false positives are much more likely when an NLOS object is present. The NLOS effect limits the performance of our system when the detection threshold is limited to not detect NLOS objects; increasing the sensitivity to detect more distant objects comes at the expense of increased detection of NLOS objects. On the other hand, for some applications, detection of NLOS objects may be welcome. In such cases the sensitivity can be increased to detect more distant line-of-sight objects with little increase in false positives in the absence of any objects.

H. Parameter Tuning and Ablation Study

Tuning the parameters t and c (see §IV-C) allows for a tradeoff between false positive rate and true positive rate. We use the dataset from §V-B to test false positive rate and that from §V-C (containing objects 0cm, 1cm, and 2cm from the robot surface) to test true positive rate. The operating points achieved by varying t and c are shown in Fig. 11. Varying these

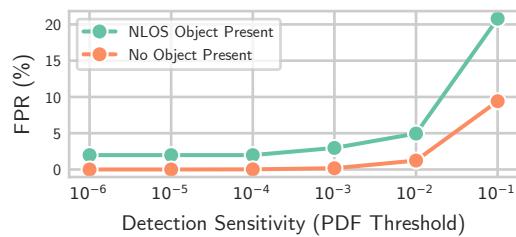


Fig. 10: When there is a large object outside of the sensor field-of-view, false positives are more likely due to three-bounce paths from the object.

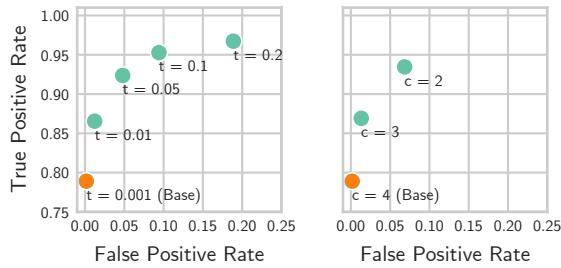


Fig. 11: Changing method parameters (t or c described in §IV-C) leads to different operating points on the ROC curve.

parameters allows for tuning of the FPR/TPR trade-off for different scenarios. However, as shown in Fig. 10, increasing the sensitivity of the method also increases the detection rate of non-line-of-sight objects.

We ablate components of our method and show the effect on true positive rate and false positive rate in Tab. I. We find that calibration (§IV-D), and bin trimming are necessary to prevent a high false positive rate. Normalization (Eqn. 1) has no significant effect on the results because the test dataset does not exhibit strong changes in ambient light.

I. Performance Under Varying Ambient Light

Ambient light affects the histogram, potentially leading to a false positive if ambient light level changes after reference capture. To evaluate our method under such changes, we capture new reference captures at the same joint locations as previous experiments. For each of four ambient light levels, 100 measurements of the robot are captured at random joint positions with only the robot present. We calculate the false positive rate of our method at each of these four light levels. The results are shown in Tab. II. Moderate changes in ambient light from the reference capture (which was taken under the “Dim LED” lighting) do not lead to an increase in false positive rate. The bright, IR-heavy halogen light source leads to a sharp increase in false positive rate. This means our method is impractical when ambient light changes significantly (*e.g.*, moving from indoors to sunlight). Additionally, we observe that the pre-processing steps described in §IV-A are somewhat

Condition	TPR (\uparrow)	FPR (\downarrow)
Base	0.789	0.002
No Preprocessing	0.785	0.002
No Calibration	0.996	0.917
No Bin Trimming	0.938	0.706

TABLE I: Ablation study. Calibration and bin trimming are necessary to avoid high false positives. Preprocessing has no effect as the test dataset does not contain changes in ambient light.

Lighting	Lux	FPR (\downarrow)	
		Base Method	No Pre-Processing
Dark	< 0.1	0.0099	0.0198
Dim LEDs*	100	0.0198	0.0198
Flourescent Lights	500	0.0099	0.1089
Halogen Lights	1000	0.1782	1.0000

TABLE II: Evaluation of the false positive rate of our method under varying levels of ambient light. Poor performance is observed under bright halogen lights, and performance is worse with no pre-processing (Eqn. 2). * Dim LED lights match the lighting used during reference capture.

effective at improving performance under high ambient light. It is possible that more sophisticated methods for reversing the effect of ambient light [44] could improve the performance of our method in these scenarios, but they require an accurate model of sensor hardware which is not available with current commercially available sensors.

VI. DEMONSTRATION

We demonstrate our method on a UR5 robot arm. As objects approach the arm, they are detected and their position shown overlaid on an image of the robot as a yellow region corresponding to the range of the detected segments, and a red line for the detected distance, shown in Fig. 1E and Fig. 1F. In addition to the configuration used for experiments (§V), we demonstrate an additional configuration in which the sensor is positioned orthogonal to the robot surface. The only change made to accommodate this configuration is that joints 2 and 3 of the robot are sampled for reference measurements rather than joints 4, 5, and 6. A live demonstration of both sensor configurations is provided in the supplementary video. This demonstration shows that our method is effective under multiple sensor configurations with minimal adjustments.

VII. CONCLUSION AND FUTURE WORK

This work demonstrates that it is possible to use raw ToF measurements to extract information about objects near a robot arm that would be impossible to obtain from distance estimates alone. We see poor performance on small objects when they are placed beyond $\sim 15\text{cm}$ from the sensor, limited partially by non-line-of-sight effects. Future work should investigate fusing sensor measurements to resolve the ambiguity between line-of-sight and non-line-of-sight objects. Our method could also be made more practical. We sample joint space uniformly to capture reference data, which is inefficient; future work should explore adaptive sampling or vary sampling density based on robot geometry. Creating a robot system which utilizes our method will require improved sensing hardware design (*i.e.*, PCBs and integrated wiring) to decrease footprint and increase frame rate, in addition to integration with kinematics and control algorithms, akin to [3], [16], [45], to enable human-robot interaction and collision avoidance. A full system could additionally leverage temporal filtering to improve detection stability and performance. Further work should also further investigate performance as a factor of the sensor placement and the geometry and reflectance properties of detected objects. We believe the method presented in this work is a step towards whole-body proximity sensing with minimal hardware sensing cost.

REFERENCES

- [1] P. Svarny, M. Tesar, J. K. Behrens, and M. Hoffmann, "Safe physical hri: Toward a unified treatment of speed and separation monitoring together with power and force limiting," in *International Conference on Intelligent Robots and Systems*, 2019, pp. 7580–7587.
- [2] P. A. Lasota, G. F. Rossano, and J. A. Shah, "Toward safe close-proximity human-robot interaction with standard industrial robots," in *International Conference on Automation Science and Engineering*. IEEE, 2014, pp. 339–344.
- [3] 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 *International Conference on Intelligent Robots and Systems*. IEEE, 2021, pp. 7255–7262.
- [4] C. Callenberg, Z. Shi, F. Heide, and M. B. Hullin, "Low-cost spad sensing for non-line-of-sight tracking, material classification and depth imaging," *ACM Transactions on Graphics*, vol. 40, no. 4, 2021.
- [5] C. Sifferman, Y. Wang, M. Gupta, and M. Gleicher, "Unlocking the performance of proximity sensors by utilizing transient histograms," *Robotics and Automation Letters*, vol. 8, no. 10, pp. 6843–6850, 2023.
- [6] AMS OSRAM AG, *TMF882X Datasheet*, AMS OSRAM AG.
- [7] ST Microelectronics, *VL53L8CH Datasheet*, ST Microelectronics.
- [8] D. Um, B. Stankovic, K. Giles, T. Hammond, and V. Lumelsky, "A modularized sensitive skin for motion planning in uncertain environments," in *International Conference on Robotics and Automation*, vol. 1. IEEE, 1998, pp. 7–12.
- [9] V. Lumelsky, M. S. Shur, S. Wagner, and M. Ding, *Sensitive skin*. World Scientific, 2000, vol. 18.
- [10] N. Yogeswaran, W. Dang, W. T. Navaraj, D. Shakthivel, S. Khan, E. O. Polat, S. Gupta, H. Heidari, M. Kaboli, L. Lorenzelli, et al., "New materials and advances in making electronic skin for interactive robots," *Advanced Robotics*, vol. 29, no. 21, pp. 1359–1373, 2015.
- [11] F. Xia, B. Bahreyni, and F. Campi, "Multi-functional capacitive proximity sensing system for industrial safety applications," in *Sensors*. IEEE, 2016, pp. 1–3.
- [12] F. Liu, S. Deswal, A. Christou, Y. Sandamirskaya, M. Kaboli, and R. Dahiya, "Neuro-inspired electronic skin for robots," *Science robotics*, vol. 7, no. 67, p. eabf7344, 2022.
- [13] S. Tsuji and T. Kohama, "Proximity skin sensor using time-of-flight sensor for human collaborative robot," *Sensors*, vol. 19, no. 14, pp. 5859–5864, 2019.
- [14] F. Giovinazzo, F. Grella, M. Sartore, M. Adami, R. Galletti, and G. Cannata, "From cyskin to proxyskin: Design, implementation and testing of a multi-modal robotic skin for human–robot interaction," *Sensors*, vol. 24, no. 4, p. 1334, 2024.
- [15] Y. Zhou, J. Zhao, P. Lu, Z. Wang, and B. He, "Tacsuit: A wearable large-area, bioinspired multi-modal tactile skin for collaborative robots," *Transactions on Industrial Electronics*, 2023.
- [16] D. Kim, M. Srouji, C. Chen, and J. Zhang, "ARMOR: Egocentric Perception for Humanoid Robot Collision Avoidance and Motion Planning," Nov. 2024, arXiv:2412.00396 [cs].
- [17] X. Fan, R. Simmons-Edler, D. Lee, L. Jackel, R. Howard, and D. Lee, "Aurasense: Robot collision avoidance by full surface proximity detection," in *International Conference on Intelligent Robots and Systems*, 2021, pp. 1763–1770.
- [18] S. E. Navarro, S. Mühlbacher-Karrer, H. Alagi, H. Zangl, K. Koyama, B. Hein, C. Duriez, and J. R. Smith, "Proximity perception in human-centered robotics: A survey on sensing systems and applications," *Transactions on Robotics*, vol. 38, no. 3, pp. 1599–1620, 2021.
- [19] P. Rakprayoon, M. Ruchanurucks, and A. Coundoul, "Kinect-based obstacle detection for manipulator," in *International Symposium on System Integration*, Dec. 2011, pp. 68–73.
- [20] W. Sukmanee, M. Ruchanurucks, and P. Rakprayoon, "Obstacle modeling for manipulator using iterative least square (ILS) and iterative closest point (ICP) base on Kinect," in *International Conference on Robotics and Biomimetics*, Dec. 2012, pp. 672–676.
- [21] X. Wang, C. Yang, Z. Ju, H. Ma, and M. Fu, "Robot manipulator self-identification for surrounding obstacle detection," *Multimedia Tools and Applications*, vol. 76, no. 5, pp. 6495–6520, Mar. 2017.
- [22] N. Lyubova, D. Filliat, and S. Ivaldi, "Improving object learning through manipulation and robot self-identification," in *International Conference on Robotics and Biomimetics*, Dec. 2013, pp. 1365–1370.
- [23] P. Michel, K. Gold, and B. Scassellati, "Motion-based robotic self-recognition," in *International Conference on Robotics and Systems*, vol. 3, Sept. 2004, pp. 2763–2768 vol.3.
- [24] G. Buizza Avanzini, N. M. Ceriani, A. M. Zanchettin, P. Rocco, and L. Bascetta, "Safety Control of Industrial Robots Based on a Distributed Distance Sensor," *Transactions on Control Systems Technology*, vol. 22, no. 6, pp. 2127–2140, Nov. 2014.
- [25] U. B. Himmelsbach, T. M. Wendt, N. Hangst, and P. Gawron, "Single pixel time-of-flight sensors for object detection and self-detection in three-sectional single-arm robot manipulators," in *International Conference on Robotic Computing*. IEEE, 2019, pp. 250–253.
- [26] S. Tsuji and T. Kohama, "Omnidirectional proximity sensor system for drones using optical time-of-flight sensors," *Transactions on Electrical and Electronic Engineering*, vol. 17, no. 1, pp. 19–25, 2022.
- [27] H. Müller, V. Niculescu, T. Polonelli, M. Magno, and L. Benini, "Robust and efficient depth-based obstacle avoidance for autonomous miniaturized uavs," *Transactions on Robotics*, 2023.
- [28] N. Zimmerman, H. Müller, M. Magno, and L. Benini, "Fully onboard low-power localization with semantic sensor fusion on a nano-uav using floor plans," *arXiv preprint arXiv:2310.12536*, 2023.
- [29] V. Niculescu, T. Polonelli, M. Magno, and L. Benini, "Nanoslam: Enabling fully onboard slam for tiny robots," *Internet of Things Journal*, 2023.
- [30] S. Karam, F. Nex, B. T. Chidura, and N. Kerle, "Microdrone-based indoor mapping with graph slam," *Drones*, vol. 6, no. 11, p. 352, 2022.
- [31] 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 *International Conference on Automation Science and Engineering*. IEEE, 2019, pp. 1230–1236.
- [32] C. Sifferman, D. Mehrotra, M. Gupta, and M. Gleicher, "Geometric calibration of single-pixel distance sensors," *Robotics and Automation Letters*, vol. 7, no. 3, pp. 6598–6605, 2022.
- [33] C. Sifferman, W. Sun, M. Gupta, and M. Gleicher, "Using a distance sensor to detect deviations in a planar surface," *Robotics and Automation Letters*, vol. 9, no. 10, pp. 8515–8522, 2024.
- [34] 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.
- [35] Y. Li, X. Liu, W. Dong, H. Zhou, H. Bao, G. Zhang, Y. Zhang, and Z. Cui, "Deltar: Depth estimation from a light-weight tof sensor and rgb image," in *European Conference on Computer Vision*. Springer, 2022, pp. 619–636.
- [36] F. Mu, C. Sifferman, S. Jungerman, Y. Li, M. Han, M. Gleicher, M. Gupta, and Y. Li, "Towards 3d vision with low-cost single-photon cameras," in *Computer Vision and Pattern Recognition*, 2024, pp. 5302–5311.
- [37] S. Jungerman, A. Ingle, Y. Li, and M. Gupta, "3d scene inference from transient histograms," in *European Conference on Computer Vision*. Springer, 2022, pp. 401–417.
- [38] M. Nishimura, D. B. Lindell, C. Metzler, and G. Wetzstein, "Disambiguating monocular depth estimation with a single transient," in *European Conference on Computer Vision*. Springer, 2020, pp. 139–155.
- [39] L. Xinyang, L. Yijin, T. Yanbin, B. Hujun, Z. Guofeng, Z. Yinda, and C. Zhaopeng, "Multi-modal neural radiance field for monocular dense slam with a light-weight tof sensor," in *International Conference on Computer Vision*, 2023.
- [40] F. Gutierrez-Barragan, A. Ingle, T. Seets, M. Gupta, and A. Velten, "Compressive single-photon 3d cameras," in *Computer Vision and Pattern Recognition*, 2022, pp. 17854–17864.
- [41] C. Niclass, A. Rochas, P.-A. Besse, and E. Charbon, "Design and characterization of a cmos 3-d image sensor based on single photon avalanche diodes," *Journal of Solid-State Circuits*, vol. 40, no. 9, pp. 1847–1854, 2005.
- [42] F. Zappa, S. Tisa, A. Tosi, and S. Cova, "Principles and features of single-photon avalanche diode arrays," *Sensors and Actuators A: Physical*, vol. 140, no. 1, pp. 103–112, 2007.
- [43] ST Microelectronics, *VL6180X Proximity and Ambient Light Sensing Module Datasheet*, ST Microelectronics.
- [44] A. Gupta, A. Ingle, A. Velten, and M. Gupta, "Photon-flooded single-photon 3d cameras," in *Computer Vision and Pattern Recognition*, 2019, pp. 6770–6779.
- [45] D. Rakita, H. Shi, B. Mutlu, and M. Gleicher, "CollisionIK: A Per-Instant Pose Optimization Method for Generating Robot Motions with Environment Collision Avoidance," in *2021 International Conference on Robotics and Automation*, May 2021, pp. 9995–10001.