Enhancing Autonomous Navigation by Imaging Hidden Objects using Single-Photon LiDAR

Aaron Young[*1], Nevindu M. Batagoda[*2], Harry Zhang[2], Akshat Dave[1],
Adithya Pediredla[3], Dan Negrut[2], Ramesh Raskar[1]

Abstract— Robust autonomous navigation in environments with limited visibility remains a critical challenge in robotics. We present a novel approach that leverages Non-Line-of-Sight (NLOS) sensing using single-photon LiDAR to improve visibility and enhance autonomous navigation. Our method enables mobile robots to "see around corners" by utilizing multi-bounce light information, effectively expanding their perceptual range without additional infrastructure. We propose a three-module pipeline: (1) Sensing, which captures multi-bounce histograms using SPAD-based LiDAR; (2) Perception, which estimates occupancy maps of hidden regions from these histograms using a convolutional neural network; and (3) Control, which allows a robot to follow safe paths based on the estimated occupancy. We evaluate our approach through simulations and real-world experiments on a mobile robot navigating an L-shaped corridor with hidden obstacles. Our work represents the first experimental demonstration of NLOS imaging for autonomous navigation, paving the way for safer and more efficient robotic systems operating in complex environments. We also contribute a novel dynamics-integrated transient rendering framework for simulating NLOS scenarios, facilitating future research in this domain.

I. INTRODUCTION

Robust autonomous navigation remains a critical challenge in robotics as mobile robots become increasingly prevalent, from industrial warehouses to urban environments. Robust navigation systems have the potential to save lives by preventing accidents, reduce financial losses by protecting human and robotic assets, and enhance operational efficiency by optimizing operation speed and energy consumption.

A fundamental challenge in achieving robust autonomous navigation lies in the inherent limits of perception systems. A robot's actions are ultimately constrained by what it can perceive. When unseen threats or obstacles suddenly appear in the robot's path, safe maneuvers become challenging. For example, consider a pedestrian emerging from a blind corner in front of a self-driving car. The autonomy stack must be capable of rapid detection and response, despite the threat being previously outside its field of view. The multi-bounce LiDAR approach outlined in this paper aims to address such challenges by expanding the robot's perceptual capabilities,

*These authors contributed equally.
[1]MIT Media Lab, Massachusetts Institute of Technology, 75 Amherst St, Cambridge, MA 02139 `aryoung`, `ad74`, `raskar@mit.edu`
[2]Department of Mechanical Engineering, University of Wisconsin - Madison, 1513 University Ave, Madison, WI 53706. `batagoda`, `hzhang699`, `negrut@wisc.edu`
[3]Department of Computer Science, Dartmouth College, 15 Thayer Dr, Hanover, NH 03755. `adithya.k.pediredla@dartmouth.edu`
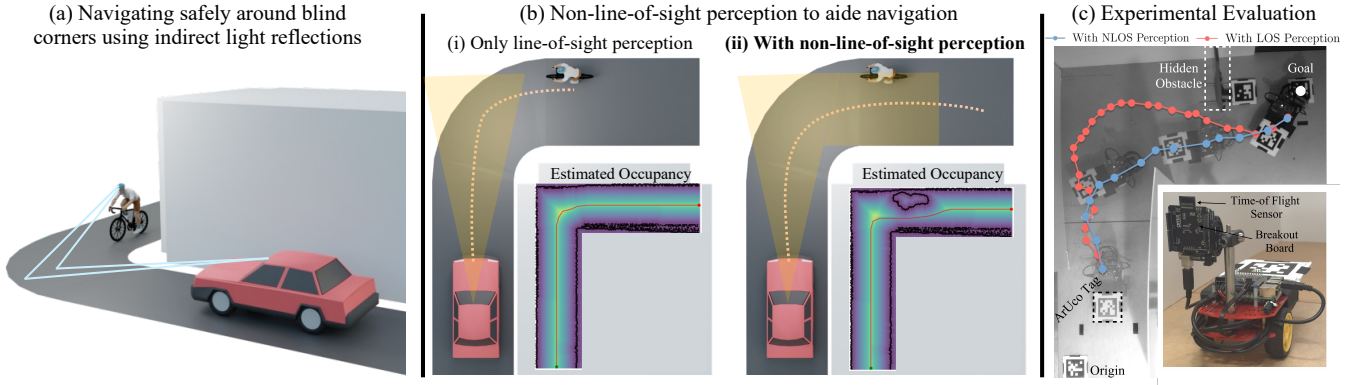
potentially allowing it to "see around corners" and react to threats before they become imminent dangers.

Existing solutions for autonomous navigation under limited visibility come with trade-offs. One common approach is to simply reduce speed at blind corners or in areas with potential hidden hazards. While this strategy improves safety, it comes at the cost of reduced efficiency, wasting both time and energy. Another solution is to use Vehicle-to-Everything (V2X) communication, which can provide additional information about the environment beyond the robot's immediate field of view. However, such solutions often require the installation and maintenance of additional infrastructure, which can be costly and may not be feasible in all environments.

We propose a novel solution that addresses the limitations of both speed reduction and additional infrastructure requirements. Drawing inspiration from the concept of safety mirrors used at blind intersections, we leverage Non-Line-of-Sight (NLOS) sensing to enhance autonomous navigation capabilities. The key insight of our work is that multi-bounce LiDAR technology can effectively transform ordinary walls and surfaces into virtual safety mirrors, expanding the robot's perceptual range without the need for additional infrastructure.

NLOS sensing utilizes the reflections of light from surfaces to detect objects that are not directly visible to the sensor. LiDAR emits laser pulses that bounce off walls and other surfaces before returning to the sensor, carrying information about occluded areas. A LiDAR sensor measures the time-of-flight of the returned light pulses. From the time-of-flight of multi-bounce light, one can reconstruct the presence and movement of objects around corners or behind obstacles.

We propose a novel pipeline that leverages single-photon LiDAR to enable non-line-of-sight perception in autonomous navigation, see Figure 1. Our approach has three key modules operating in a loop: 1) Sensing, which captures multi-bounce histograms from a SPAD-based LiDAR; 2) Perception, which estimates occupancy maps of hidden regions from these histograms; and 3) Control, which allows an agent to follow safe paths based on the estimated occupancy maps.

We evaluated the effectiveness of our NLOS perception approach for autonomous navigation through simulation and real-world experiments. We develop a simulation framework that couples transient rendering with vehicle dynamics within the multi-physics engine Project Chrono[1][2]. This allows us to co-simulate and model the multi-bounce histograms

(a) Navigating safely around blind corners using indirect light reflections

(b) Non-line-of-sight perception to aide navigation
(i) Only line-of-sight perception
(ii) With non-line-of-sight perception
Estimated Occupancy

(c) Experimental Evaluation
With NLOS Perception — With LOS Perception
Hidden Obstacle — Goal
Time-of Flight Sensor
Breakout Board
ArUco Tag
Origin

Fig. 1. **Autonomous navigation using single-photon LiDAR.** (a) Autonomous navigation is challenging under unseen threats such as a cyclist emerging from a blind corner. We leverage single-photon LiDAR for autonomous navigation by seeing around corners using indirect reflections of light. (b) With only line-of-sight perception, the location of hidden objects around the corner is unknown to the agent, resulting in collisions. Our approach enables non-line-of-sight (NLOS) perception with the estimated occupancy map accurately estimating the hidden object location. (c) We evaluate our NLOS-based navigation approach experimentally using a commercial-grade single-photon LiDAR mounted on a two-wheeled robot. Incorporating NLOS perception results in a smoother trajectory while avoiding the hidden obstacle.

of single-photon LiDARs with vehicle/robot dynamics simulations. Our simulation experiments demonstrate NLOS perception aids in navigating an L-turn corner with reduced chances of collision with hidden obstacles compared to LOS-only perception. We implemented our system on a physical two-wheeled robot equipped with a commercial-grade single-photon LiDAR. In a blind corner navigation task, the NLOS-capable robot demonstrated superior performance, generating and following more efficient paths around hidden objects. Quantitatively, the NLOS approach reduced travel time by more than half and shortened the trajectory by 33% compared to LOS-only navigation.

Our contributions are summarized as follows:

- Ours is the first work to explore the role of non-line-of-sight perception for autonomous navigation.
- We propose a novel pipeline for incorporating multi-bounce histograms containing NLOS information into a navigation framework, including a data-driven NLOS perception approach.
- We develop a transient rendering engine within the open-source multi-physics engine Project Chrono to simulate single-photon LiDARs and robot dynamics in conjunction. Through simulations, we demonstrate our approach reduces the likelihood of collisions compared to LOS-only perception.
- We experimentally evaluate our NLOS-aided approach on a novel setup leveraging commercial-grade single-photon LiDAR mounted on a two-wheeled robot.

II. RELATED WORK

A. Single-photon LiDAR

Single-photon LiDAR has emerged as a promising sensing technology for robotics applications. Unlike traditional LiDAR systems that typically record only the first return, single-photon LiDARs can detect and time-stamp individual photons with picosecond-level precision [3], [4]. Single-photon LiDAR has higher sensitivity and allows one to capture not just the direct reflections from visible surfaces, but also the indirect reflections that have undergone multiple bounces before returning to the sensor. Single-photon LiDAR has shown great promise in depth sensing [5], [6], 3D surface mapping [7], [8], handling specular surfaces [9], [10], long-range detection [11], [12], airborne applications such as snow depth mapping [13], [14], [9], forest surveying [15], and 3D terrain mapping [16].

Our system specifically utilizes *flash* single-photon LiDARs that are now commercially available and inexpensive [17], [18]. These LiDARs feature flash illumination that floods the scene, coupled with an array of SPAD pixels. This configuration enables simultaneous capture of spatio-temporal multi-bounce information.

B. Non-Line-Of-Sight Imaging

Non-line-of-sight imaging has emerged as a promising application by utilizing single-photon LiDAR that can measure indirect reflections and image objects hidden from the direct line of sight. Recent works demonstrate 3D reconstruction of occluded scenes from single-photon LiDAR at high spatial resolution and computational efficiency [19], [20], [21], [22], [23], [24]. While the majority of existing works focus on 3D imaging, there are also notable contributions to 3D pose estimation [25], [26] and 3D tracking [4] using NLOS information. Our paper presents the first experimental demonstration of NLOS imaging for robotics applications, paving the way for novel applications in autonomous navigation and sensing.

C. Safety-Critical Control

Developing autonomous systems capable of collision-free obstacle avoidance is challenging and remains an area of on-going research. One approach is to integrate vehicle path tracking, stabilization, and collision avoidance, temporarily prioritizing collision avoidance over stabilization [27]. Model predictive control (MPC) is a proven technique which has been shown to be an effective controller for vehicles at high speeds [28], [29], in highway scenarios [30], and for collision

avoidance [31], [32]. While MPC-based algorithms couple planning and tracking, treating them as separate problems has proven simple and reliable. Common planning algorithms employed in mobile robotics include Dijkstra's algorithm, A*, and the Artificial Potential Field (APF) method [33], [34], [35], [36], [37], amongst others. While many control-focused strategies aim to improve obstacle avoidance through control design, e.g, predictive control and control barrier functions, we approach this problem by enhancing perception to increase visibility, giving control algorithms more reaction time and space.

III. NLOS-aided Autonomous Navigation

In this section, we present our pipeline for incorporating non-line-of-sight perception into autonomous navigation, see also Fig. 2. We utilize single-photon LiDAR to enhance autonomous agent's perception beyond its line of sight. Our pipeline consists of three key modules that operate in a continuous loop: Sensing, Perception, and Control. In the *Sensing* phase, the agent captures multi-bounce histograms from a SPAD-based LiDAR, gathering information about the hidden region. Subsequently, the *Perception* module uses these histograms to estimate an occupancy map of the occluded areas. Finally, the *Control* module plans a path based on the estimated occupancy. This process repeats at each time step, allowing the agent to effectively plan for and navigate around obstacles in blind corners. The following subsections describe each of the modules in detail.

A. Sensing histograms using single-photon LiDAR

For the single-photon LiDARs, the multi-bounce information is captured in the form of histograms, which record the number of photon arrivals at different time bins. These histograms encode both the direct and indirect light transport. In a typical histogram, the first peak corresponds to the direct component – light that has reflected directly off visible surfaces. Subsequent peaks typically represent light that has bounced three times before reaching the sensor: 1) from the light source to the visible surface; 2) from the visible surface to the hidden surface; and 3) from the hidden surface to the visible surface before reaching the Lidar detector. The position of the three-bounce peaks in the histogram encodes location and size of the hidden or occluded objects.

We utilize *Flash* single-photon LiDARs [17], [18], which comprises of an array of SPAD pixels that each generates its own time-of-flight histogram. In this paper, we consider that the SPAD LiDAR is pointed at a planar floor, and that there is single hidden object of unknown location (x_h, y_h) and size s_h. The spatio-temporal histogram array obtained $\mathbf{H}$ depends on x_h, y_h and s_h.

B. Perception of the hidden region

Estimating size and location of a hidden object from the histogram array is challenging. The first challenge is tied to the ambiguity of the signal signature. There could be multiple hidden object positions that could have the same 3-bounce time-of-flight. The second challenge is that of low signal

quality. Commercial LiDARs operate with eye-safe lasers imposing a limit on the emitted laser power. Three bounces on surfaces severely attenuates the signal reaching back to the sensor resulting in very low signal-to-noise ratio (SNR).

We tackle the ambiguity and SNR challenges of commercial-grade LiDAR sensing using a data-driven approach. We train a convolutional neural network (CNN) that inputs the histogram array and outputs the size and location of the hidden object. This network is trained on a dataset of histogram arrays captured with varying hidden object location and size along with the LiDAR location.

$$(\hat{x}_h, \hat{y}_h, \hat{s}_h) = f_\theta \left(\mathbf{H} \left(x_h, y_h, s_h \right) \right) \tag{1}$$

The neural network is comprised of two convolutional layers, followed by two fully connected layers. The fully connected layers take the output of the convolutions concatenated with the ground truth position of the robot at the current state. We use L2 loss with respect to the object's ground truth position and orientation.

C. Control based on the estimated occupancy

We decompose the control task into two stages: path planning and actuation. This two-stage approach allows us to build a controller agnostic to the underlying vehicle in use. We use a simple global path planning algorithm (specifically A* [37]), and iteratively optimize the path as the agent moves within the environment. The path produced by A* doesn't take vehicle dynamics under consideration, so we smooth the output path using a Bezier curve. To determine the next actuation signal sent to the vehicle, we sample the first control point and compute the heading error. Based on a proportional feed-forward calculation, we compute a throttle and steering command which is sent to the vehicle.

IV. Simulation Results

In this section, we describe our simulation environment, which was developed to test the integration of multi-bounce sensor information for robotic navigation. To accomplish this task, we require two simulators: 1) a dynamics engine capable of simulating a virtual world and virtual agents, and 2) a rendering engine that can model transient data, i.e., the time-of-flight of light within the virtual world.

A. Dynamics-Integrated Transient Rendering

To design robotic controllers that utilize transient data in-the-loop, we have developed a transient renderer within the open-source Chrono multi-physics engine [1]. With native integration of other sensors [2], vehicles [38], and ROS [39], [40], Chrono enables the simulation of a wide range of robotic applications. Transient rendering is the simulation of the propagation of light at very high time resolutions. As opposed to standard path tracing, transient rendering models the light at a point in the spatial and time domains (x, y, t). Using transient rendering, we can model the amount of light arriving at a time-of-flight detector often at time resolutions on the order of picoseconds or less and record it
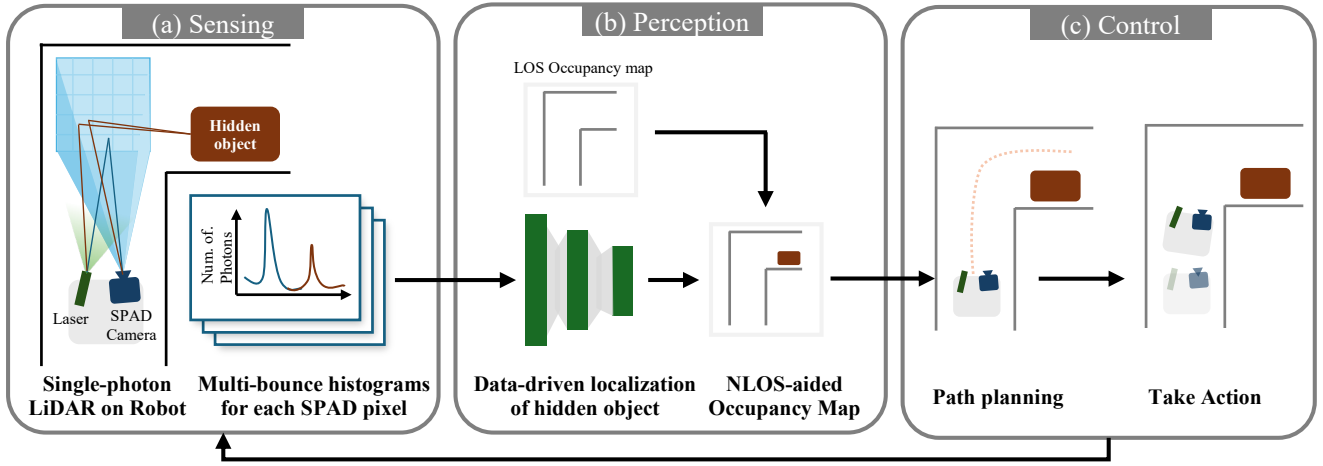
Fig. 2. **Our pipeline for NLOS-aided Autonomous Navigation.** (a) *Sensing*: We capture multi-bounce histograms using single-photon LiDAR to gather information about hidden regions. (b)*Perception*: Captured histograms are then processed using a data-driven approach to estimate the occupancy map of the occluded region. (c) *Control*: Then the optimal path is planned based on the estimated occupancy map for navigating around obstacles.
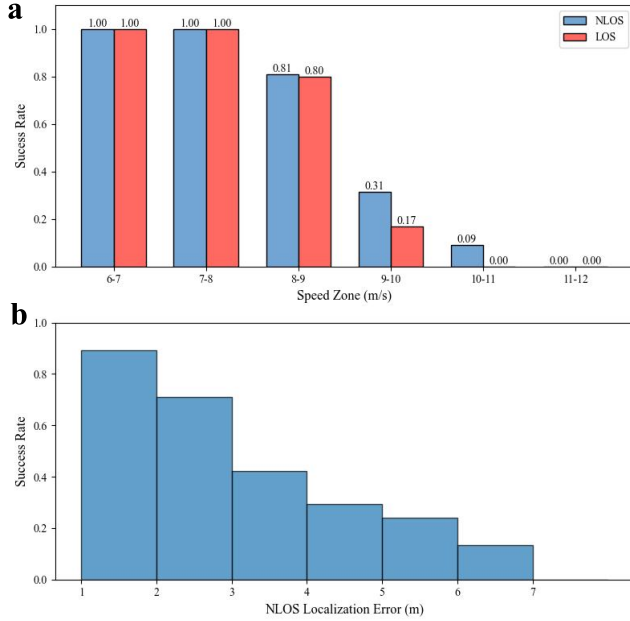

Fig. 3. **Simulation Results for navigating a L-turn corner using line-of-sight and non-line-of-sight perception. a)** Comparison of collision avoidance success rate between LOS and NLOS perception. NLOS has an edge in performance over LOS at higher speeds where path planning cannot react on time. **b)** And, we find that higher localization error leads to worse performance in avoiding obstacles. Which matches intuition.

as a collection of histograms for each time interval. Using these high-time-resolution histograms, we can distinguish light coming from multiple scattering/bounce events in the scene. This enables us to infer information about the scene even beyond the line of sight. Our transient renderer is based on Royo et al.'s work on non-line-of-sight transient rendering [41]. Since NLOS transient rendering is computationally expensive, we use GPU hardware accelerated ray tracing and special sampling methods [41] to improve rendering speeds. Our simulation framework can model vehicle dynamics and

render an 8×8 multi-bounce transient sensor at 10 FPS.

B. High-speed Hallway Navigation with Occluded Obstacles

We view the use of multi-bounce light as an enhancement to existing perception-based approaches for robotic navigation. Our simulated experiments, shown in Figure 1, deploy a wheeled vehicle in a corridor environment with a hidden retro-reflective object of diameter of 2m randomly placed around the corner. The vehicle maintains a constant speed throughout the maneuver. The vehicle is then either provided sensor data from a single-bounce LiDAR (Figure 1a), or from a multi-bounce LiDAR (Figure 1b). The perceived area is conceptualized as an occupancy map, where obstacles and walls are considered "occupied" and everywhere else is "unoccupied". Training data for the NLOS object detection model is acquired by running the vehicle using a privileged controller through 30 different environments with unique object positions and saving the transient information. With the trained model, we run a total of 200 trials with uniformly sampled speed values for both LOS and NLOS perception. Analysis of the NLOS vs LOS-based navigation is shown in Figure 3a.

In the simulated environment, we found improvement for vehicles capable of perceiving multi-bounce light versus only line-of-sight information. In the $8 - 9ms^{-1}$ speed zone, the collision rate decreased by 1%, by 14% in the $9 - 10ms^{-1}$ zone, and by 9% in the $10 - 11ms^{-1}$ zone as compared to the LOS case. This indicates that being able to detect hidden objects is beneficial when navigating corners while maintaining higher speed profiles. In fact, we see that LOS completely fails to avoid the obstacle at speeds higher than $10ms^{-1}$. For this particular scenario, the maximum reasonable speed a vehicle could take this corner at was calculated to be $12ms^{-1}$ based on the formula,

$$v_{max} = \sqrt{\mu g r_{max}} \tag{2}$$

where r_{max} is the maximum turn radius of the corner ($15m$), mu is the coefficient of friction of the vehicle's tires (estimated to be 0.9), and g is acceleration due to gravity. As such, for our simulation experiments we set $12ms^{-1}$ as the maximum velocity of the vehicle.

We examine the effect of localization error on collision rate, where the localization error is the mean of all the errors between ground truth obstacle position and a predicted measurement at a given sensor capture time. When localizing the object position, we define a radius of uncertainty ($r = 1m$ in our trials) as the model doesn't output object size. We found that the localization error did not have a considerable effect until the radius of uncertainty was larger than the object itself (Figure 3b). As such, we see a degrading performance in obstacle avoidance with high localization errors. We expect simple filtering techniques of the noisy localization values to improve this result.

V. Experimental Results

To validate our findings from simulation and assess the viability of multi-bounce LiDAR measurements in robotic navigation, we test our method using a real-world setup. This section describes the results of these experiments, demonstrating the potential of three-bounce measurements for enhancing autonomous navigation systems.

A. Experimental Setup

Our experimental setup is composed of a two wheeled mobile robot navigation around an L-shaped corridor. We utilize the AMS TMF8828 SPAD-based LiDAR module and their Arduino evaluation kit as the sensor system [17]. The TMF8828 sensor is comprised of 18×12 individual SPAD pixels, which is aggregated into 3×3 measurement zones. In this configuration, the sensor has an effective FoV of $41° \times 52°$ and can measure light traveling a total of 10 m. Ground truth position measurements are obtained via four ArUco tags [42] – placed on the scene, robot, object, and goal respectively – and imaged with a monochrome camera mounted with a birds-eye view of the entire scene. The robot is a small, two wheeled mobile robot [43] equipped with a Raspberry Pi 5 for processing. Training of the underlying perception model is performed on an external system, but data acquisition and evaluation takes place on the robot with ground truth positions streamed over the network.

B. Data Acquisition and Training

Data acquisition is performed for 400 unique object positions and 5 unique camera positions within the scene for a total of 2000 samples. A 2-axis gantry system (Figure 4a) is used to repeatedly position the object, and the mobile robot is used to position the camera during data collection. Each data capture contains 10 exposures which are averaged to obtain the final measurement value for that state. The object itself is covered in a retro-reflective material to enhance third-bounce measurements (Figure 4b). This setup, using a fixed sensor, is shown in Figure 4a.
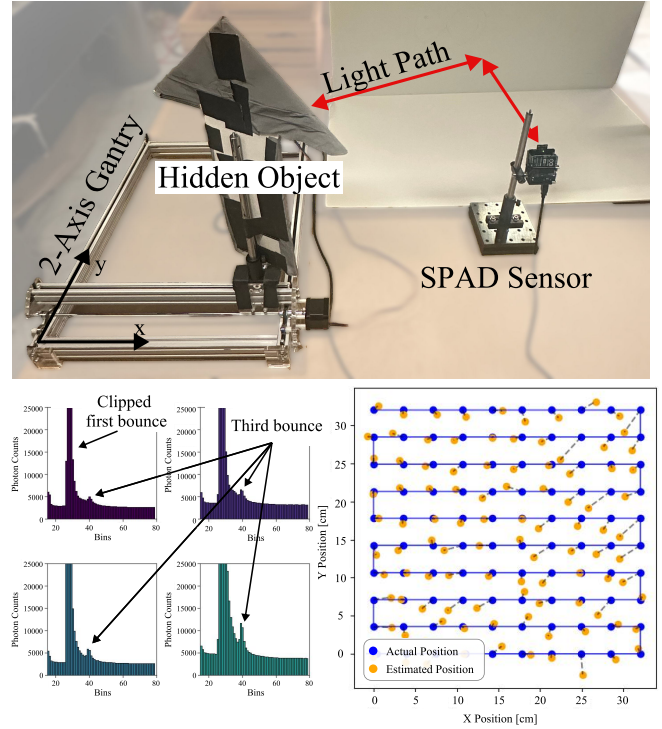

Fig. 4. **Capture and evaluation setup.** a) To validate the perception algorithm and generate a dataset to train the CNN, we utilize a 2-axis gantry to precisely position a hidden object (in this case, a retro-reflective arrow) in some region outside the FoV of the SPAD sensor. For training, we generate data for many positions and orientations of both the sensor and object to produce a diverse dataset. b) Multi-bounce histograms of four of the output zones from the SPAD sensor. The first bounce is the primary spike, and in later bins (i.e. with a longer time-of-flight), the third bounce reflection from the object can be observed. Different zones have a different view of the scene, and so the relative intensity and peak position encode the object's state. This information is leveraged by the model to predict the position of the hidden object. c) The estimates of our model on an evaluation set (i.e., data held out from training) which shows our model estimates object position with sub-10 cm accuracy. We found that evaluation performance decreased substantially when testing on states (i.e., camera/object positions) not seen during training.

The acquired dataset is subsequently used to train the CNN. The model takes as input the full, uncropped histogram and the ground truth camera position, and is trained to predict the object's x, y, and yaw. During training, the dataset is split 80/20 into a training and evaluation set, respectively. We use a batch size of 16, a learning rate of 1×10^{-3}, and train for ~ 2000 epochs. An example evaluation scenario is shown in Figure 4c, where the model is capable of predicting object states not in it's training dataset to within $5cm$.

We found our method was effective for multiple sensor placements (i.e. height relative to the relay surface) and orientations. During training we use multiple sensor poses, such as oblique views of the ground and perpendicular views of a vertical relay wall. This produced a more robust policy that was effective in interpolating between training locations. However, this required a considerable amount of data. Additionally, object or camera positions out of the bounds of the training dataset produced erroneous predictions. We believe the is due in part to the small size of the CNN in use.

a Line-of-sight

b Non-line-of-sight

Ground truth object position

Initial estimated object position

---- Calculated path using initial LOS occupancy

—•— Final path taken

---- Calculated path using initial NLOS occupancy
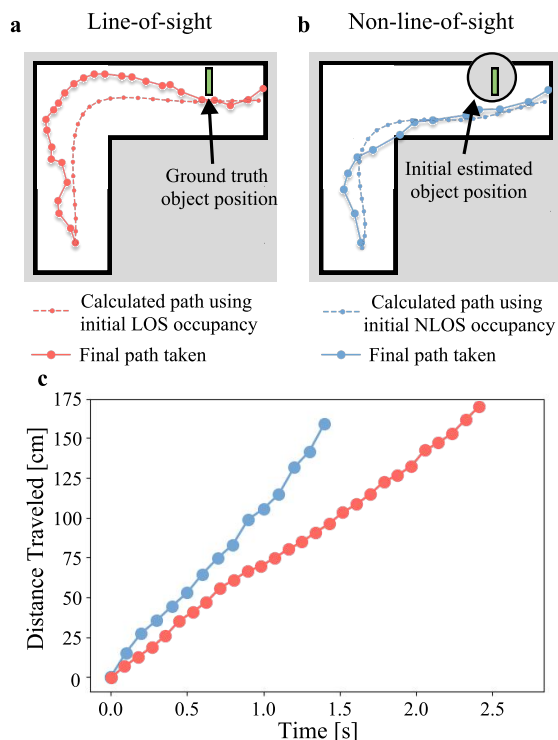
—•— Final path taken

c

Fig. 5. **Blind corner navigation using non-line-of-sight perception on a physical robot.** To test the effectiveness of non-line-of-sight imaging for autonomous navigation, we ran experiments on a physical robot in an L-shaped blind corner. An object was placed in an area hidden from view of the sensor. a) The agent with only a LOS view couldn't prepare for the object before rounding the corner, and had to overshoot back inside to reach the goal. b) On the other hand, when NLOS information was used, the robot could successfully generate and follow a path which reduced the wasted travel time. c) A quantitative depiction of the efficiency improvements for an NLOS-capable robot. With only LOS information, the robot took more than $2\times$ longer and required a 33% longer trajectory.

C. Blind Corner Navigation

To demonstrate the usefulness of third-bounce light for robotic navigation, we deploy a mobile robot in an L-shaped corridor. We then compare its navigation performance using strictly line-of-sight information versus non-line-of-sight measurements. As shown in Figure 5a and Figure 5b, without information about the hidden object (i.e. LOS only), the algorithm essentially produces a centerline path through the corridor. This approach is suboptimal, as the robot encounters the object as soon as it rounds the corner, thus needing costly maneuvers later on. Ideally, the agent should anticipate the obstacle and take a sharper path closer to the inside of the corner. In the non-line-of-sight case (Figure 5b), the robot localizes the hidden object from the outset, enabling it to plan an efficient path that immediately avoids the obstacle.

By effectively seeing around the corner, the agent is able to produce more desireable control outcomes. As shown in Figure 5c, by planning a route that avoids sharp maneuvers later in the experiment, the NLOS-enabled agent follows a route 33% shorter than if it were using strictly LOS information. Furthermore, by following a longer route, the robot that uses LOS information takes nearly $2\times$ the amount of time than in the NLOS case.

VI. CONCLUSION AND FUTURE WORK

In this paper, we demonstrate the utility of NLOS perception using single-photon LiDAR for autonomous navigation. Our novel pipeline estimates the occupancy map of the hidden region from multi-bounce transients and then utilizes the occupancy map for autonomous navigation. In simulations, we show that our approach, with NLOS perception, reduces the likelihood of collisions with the hidden target at high speeds compared to traditional LOS-only perception. Through experiments with a commercial-grade single-photon LiDAR mounted on a two-wheeled robot, we demonstrate a smoother trajectory in navigating around hidden objects.

Our work opens up new avenues for sensing, perception, and control systems that leverage indirect reflections for safer navigation around hidden obstacles. There are several areas for future improvement and investigation. On the sensing front, addressing the low signal-to-noise ratio of commercial-grade single-photon LiDARs in measuring multi-bounce light remains challenging. While our initial evaluation utilized retro-reflective surfaces, developing robust approaches for diffuse surfaces is a crucial next step. Regarding perception, our data-driven approach could be extended by exploring the use of dynamics-integrated transient renderers for training on simulated data, potentially reducing the need for extensive physical data collection. Finally, there is significant potential to enhance our control framework to handle more complex environments and multiple hidden targets, further improving the versatility and applicability of NLOS perception in autonomous navigation.

VII. ACKNOWLEDGMENTS

REFERENCES

[1] A. Tasora, R. Serban, H. Mazhar, A. Pazouki, D. Melanz, J. Fleischmann, M. Taylor, H. Sugiyama, and D. Negrut, "Chrono: An open source multi-physics dynamics engine," in *High Performance Computing in Science and Engineering – Lecture Notes in Computer Science*, T. Kozubek, Ed. Springer International Publishing, 2016, pp. 19–49.

[2] A. Elmquist, R. Serban, and D. Negrut, "A Sensor Simulation Framework for Training and Testing Robots and Autonomous Vehicles," *Journal of Autonomous Vehicles and Systems*, vol. 1, no. 2, p. 021001, 02 2021. [Online]. Available: https://doi.org/10.1115/1.4050080

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

[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 (TOG)*, vol. 40, no. 4, pp. 1–12, 2021.

[5] D. B. Lindell, M. O'Toole, and G. Wetzstein, "Single-photon 3d imaging with deep sensor fusion." *ACM Trans. Graph.*, vol. 37, no. 4, p. 113, 2018.

[6] J. Rapp, Y. Ma, R. M. Dawson, and V. K. Goyal, "High-flux single-photon lidar," *Optica*, vol. 8, no. 1, pp. 30–39, 2021.

[7] C. Sifferman, W. Sun, M. Gupta, and M. Gleicher, "Using a distance sensor to detect deviations in a planar surface," *arXiv preprint arXiv:2408.03838*, 2024.

[8] 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," 2024. [Online]. Available: https://arxiv.org/abs/2403.17801

[9] C. Henley, S. Somasundaram, J. Hollmann, and R. Raskar, "Detection and mapping of specular surfaces using multibounce lidar returns," *Optics Express*, vol. 31, no. 4, pp. 6370–6388, 2023.

[10] T.-H. Lin, C. Henley, S. Somasundaram, A. Dave, M. Laifenfeld, and R. Raskar, "Handheld mapping of specular surfaces using consumer-grade flash lidar," in *2024 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2024, pp. 1–12.

[11] R. H. Hadfield, J. Leach, F. Fleming, D. J. Paul, C. H. Tan, J. S. Ng, R. K. Henderson, and G. S. Buller, "Single-photon detection for long-range imaging and sensing," *Optica*, vol. 10, no. 9, pp. 1124–1141, 2023.

[12] S. Chan, A. Halimi, F. Zhu, I. Gyongy, R. K. Henderson, R. Bowman, S. McLaughlin, G. S. Buller, and J. Leach, "Long-range depth imaging using a single-photon detector array and non-local data fusion," *Scientific reports*, vol. 9, no. 1, p. 8075, 2019.

[13] F. King, R. Kelly, and C. G. Fletcher, "Evaluation of lidar-derived snow depth estimates from the iphone 12 pro," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[14] ——, "New opportunities for low-cost lidar-derived snow depth estimates from a consumer drone-mounted smartphone," *Cold Regions Science and Technology*, vol. 207, p. 103757, 2023.

[15] A. Swatantran, H. Tang, T. Barrett, P. DeCola, and R. Dubayah, "Rapid, high-resolution forest structure and terrain mapping over large areas using single photon lidar," *Scientific reports*, vol. 6, no. 1, p. 28277, 2016.

[16] Y. Hong, S. Liu, Z.-P. Li, X. Huang, P. Jiang, Y. Xu, C. Wu, H. Zhou, Y.-C. Zhang, H.-L. Ren *et al.*, "Airborne single-photon lidar towards a small-sized and low-power payload," *Optica*, vol. 11, no. 5, pp. 612–618, 2024.

[17] AMS OSRAM AG, "TMF882X Datasheet," https://ams-osram.com/, n.d.

[18] STMicroelectronics, "VL53L4CX Time-of-Flight Ranging Sensor Datasheet," https://www.st.com/, n.d.

[19] M. O'Toole, D. B. Lindell, and G. Wetzstein, "Confocal non-line-of-sight imaging based on the light-cone transform," *Nature*, vol. 555, no. 7696, pp. 338–341, 2018.

[20] F. Heide, M. O'Toole, K. Zang, D. B. Lindell, S. Diamond, and G. Wetzstein, "Non-line-of-sight imaging with partial occluders and surface normals," *ACM Transactions on Graphics (ToG)*, vol. 38, no. 3, pp. 1–10, 2019.

[21] D. B. Lindell, G. Wetzstein, and M. O'Toole, "Wave-based non-line-of-sight imaging using fast fk migration," *ACM Transactions on Graphics (ToG)*, vol. 38, no. 4, pp. 1–13, 2019.

[22] X. Liu, I. Guillén, M. La Manna, J. H. Nam, S. A. Reza, T. Huu Le, A. Jarabo, D. Gutierrez, and A. Velten, "Non-line-of-sight imaging using phasor-field virtual wave optics," *Nature*, vol. 572, no. 7771, pp. 620–623, 2019.

[23] A. Pediredla, A. Dave, and A. Veeraraghavan, "Snlos: Non-line-of-sight scanning through temporal focusing," in *2019 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2019, pp. 1–13.

[24] B. Ahn, A. Dave, A. Veeraraghavan, I. Gkioulekas, and A. C. Sankaranarayanan, "Convolutional approximations to the general non-line-of-sight imaging operator," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7889–7899.

[25] M. Isogawa, Y. Yuan, M. O'Toole, and K. M. Kitani, "Optical non-line-of-sight physics-based 3d human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7013–7022.

[26] P. Liu, Y. Yu, Z. Pan, X. Peng, R. Li, Y. Wang, J. Yu, and S. Li, "Hiddenpose: Non-line-of-sight 3d human pose estimation," in *2022 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2022, pp. 1–12.

[27] J. Funke, M. Brown, S. M. Erlien, and J. C. Gerdes, "Collision avoidance and stabilization for autonomous vehicles in emergency scenarios," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 4, pp. 1204–1216, 2017.

[28] C. Kim, Y. Yoon, S. Kim, M. J. Yoo, and K. Yi, "Trajectory planning

and control of autonomous vehicles for static vehicle avoidance in dynamic traffic environments," *IEEE Access*, vol. 11, pp. 5772–5788, 2023.

[29] F. Micheli, M. Bersani, S. Arrigoni, F. Braghin, and F. Cheli, "Nmpc trajectory planner for urban autonomous driving," *Vehicle system dynamics*, vol. 61, no. 5, pp. 1387–1409, 2023.

[30] S. Li and Q. Zhao, "Research on the emergency obstacle avoidance strategy of intelligent vehicles based on a safety distance model," *IEEE Access*, vol. 11, pp. 7124–7134, 2023.

[31] Y. S. Quan, J. S. Kim, and C. C. Chung, "Robust model predictive control with control barrier function for nonholonomic robots with obstacle avoidance," in *2021 21st International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 2021, pp. 1377–1382.

[32] R. Grandia, A. J. Taylor, A. D. Ames, and M. Hutter, "Multi-layered safety for legged robots via control barrier functions and model predictive control," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 8352–8358.

[33] A. N. A. Rafai, N. Adzhar, and N. I. Jaini, "A review on path planning and obstacle avoidance algorithms for autonomous mobile robots," *Journal of Robotics*, vol. 2022, no. 1, p. 2538220, 2022.

[34] H. Zhang, Y. Tao, and W. Zhu, "Global path planning of unmanned surface vehicle based on improved a-star algorithm," *Sensors*, vol. 23, no. 14, p. 6647, 2023.

[35] D. Zhang, C. Chen, and G. Zhang, "Agv path planning based on improved a-star algorithm," in *2024 IEEE 7th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 7. IEEE, 2024, pp. 1590–1595.

[36] Y. Ji, L. Ni, C. Zhao, C. Lei, Y. Du, and W. Wang, "Tripfield: A 3d potential field model and its applications to local path planning of autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 3541–3554, 2023.

[37] R. Kabir, Y. Watanobe, M. R. Islam, and K. Naruse, "Enhanced robot motion block of a-star algorithm for robotic path planning," *arXiv preprint arXiv:2312.15738*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2312.15738

[38] R. Serban, M. Taylor, D. Negrut, and A. Tasora, "Chrono::Vehicle

[39] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA workshop on open source software*, vol. 3. Kobe, Japan, 2009, p. 5.

[40] A. Elmquist, A. Young, I. Mahajan, K. Fahey, A. Dashora, S. Ashokkumar, S. Caldararu, V. Freire, X. Xu, R. Serban *et al.*, "A software toolkit and hardware platform for investigating and comparing robot autonomy algorithms in simulation and reality," *arXiv preprint arXiv:2206.06537*, 2022.

[41] D. Royo, J. García, A. Muñoz, and A. Jarabo, "Non-line-of-sight transient rendering," *Computers & Graphics*, vol. 107, pp. 84–92, 2022.

[42] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.

[43] L. Paull, J. Tani, H. Ahn, J. Alonso-Mora, L. Carlone, M. Cap, Y. F. Chen, C. Choi, J. Dusek, Y. Fang, D. Hoehener, S.-Y. Liu, M. Novitzky, I. F. Okuyama, J. Pazis, G. Rosman, V. Varricchio, H.-C. Wang, D. Yershov, H. Zhao, M. Benjamin, C. Carr, M. Zuber, S. Karaman, E. Frazzoli, D. Del Vecchio, D. Rus, J. How, J. Leonard, and A. Censi, "Duckietown: An open, inexpensive and flexible platform for autonomy education and research," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1497–1504.

template-based ground vehicle modeling and simulation," *International Journal of Vehicle Performance*, vol. 5, no. 1, pp. 18–39, 2019.