2024 IEEE International Conference on Computational Photography (ICCP) | 979-8-3503-6155-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCP61108.2024.10644634

Handheld Mapping of Specular Surfaces using Consumer-Grade Flash LiDAR

Tsung-Han Lin, Connor Henley, Siddharth Somasundaram, Akshat Dave, Moshe Laifenfeld, Ramesh Raskar

Abstract—We propose an approach to leverage multi-bounce returns of a flash LiDAR on portable smartphones for 3D specular surface reconstruction. Traditional LiDAR systems assume that all returns are one-bounce returns, which can lead to an overestimation of the true mirror surface and cause it to appear as if there is a hole. However, in reality, returns from mirror surfaces follow multi-bounce paths. We operate with a consumer-grade, coarse multi-beam flash LiDAR, enabling real-time mapping on an affordable and portable smartphone. To address the challenges posed by the coarse setup, where the transmitter and receiver are co-located, we propose solving the association problem using the 'reciprocal pair' algorithm. This algorithm can distinguish between different types of bounces from multi-bounce returns. We have demonstrated detection over multiple consecutive frames for dense mirror mapping. In addition to 3D reconstruction, we show that multi-bounce returns enhance performance in applications such as segmentation and novel view synthesis. Our method can be integrated with state-of-the-art learned-based models, enhancing their robustness in discerning ambiguous scenarios. Importantly, our approach can map various specular surfaces like mirrors and glasses without assuming specific shapes, and it can operate on non-perpendicular specular-diffuse surface pairs.

Index Terms—3D Reconstruction, Specular Surfaces, Multi-bounce Returns, Segmentation, Novel View Synthesis

◆

1 INTRODUCTION

DETECTION and mapping of specular surfaces like mirrors and glass remains a challenging unsolved problem in computer vision. These surfaces are ubiquitous indoors, yet most vision algorithms fail to handle their presence, often resulting in 3D reconstructions containing holes along specular surfaces, as shown in Fig. 1. The ability to properly detect, segment, and map mirror-like surfaces is critical for performance in downstream applications (e.g. robotics). For example, robots can collide with specular surfaces or fail to properly grasp them if they are not properly detected and mapped. In this work, we present a practical specular surface mapping approach that utilizes emerging consumer-grade flash LiDAR cameras.

1.1 Ambiguities with Conventional RGB and LiDAR.

Existing RGB and depth cameras suffer from inherent ambiguities that make specular detection challenging. Each pixel in a camera measures light traveling in a scene along a certain ray. If this ray intersects a diffuse object, the pixel measurement will encode information about the surface of the object. However, if this ray intersects a specular surface, the surface will strongly reflect light from other parts of the scene along the camera ray. In this case, the light arriving at the pixel encodes information about other portions of the scene, as shown in Fig. 2(a). The problem of determining whether a pixel is measuring a specular reflection from a different part of the scene or a diffuse reflection along the camera ray is fundamentally ambiguous. As a result, 3D mapping is challenging under the presence of specular objects using RGB cameras alone.

Fig. 1. **Mirror Reconstruction with Conventional LiDAR.** Conventional LiDAR systems fail to map the 3D shape of mirrors because mirrors reflect light away from the sensor. As a result, holes appear in the reconstruction.

This ambiguity persists with scanning LiDAR systems as well. If a LiDAR scanner emits a laser spot towards a specular point, the system will measure the time of flight (ToF) of three-bounce light traveling from the camera → specular point → a diffuse point in the scene → specular point → camera. Determining if this returned light is three-bounce light or one-bounce light that traveled past the mirror is also challenging, as shown in Fig. 2(b). The same issue holds: there exists an ambiguity as to whether the light remained along the same camera ray or whether it interacted with other parts of the scene. This effect can also be seen in Fig. 1, where the camera mistakenly places points behind the mirror plane.

1.2 Importance of Two-Bounce Returns.

Conventional scanning LiDAR only measures light that is emitted and reflected along the same ray, leading to ambiguities between one-bounce and three-bounce light. In this work, we leverage ToF of two-bounce returns to resolve

---

- *T.-H.L., C.H., S.S., A.D., R.R. are with Massachusetts Institute of Technology.*
- *M.L. is with Apple.*

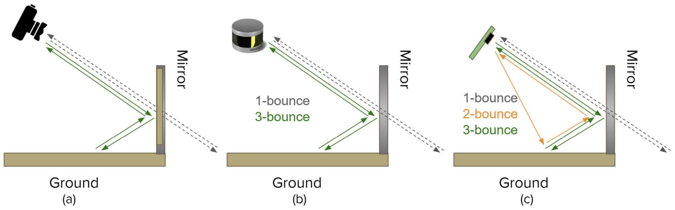


Fig. 2. **(a) RGB camera.** RGB cameras struggle to differentiate real vs virtual images. **(b) Conventional Scanning LiDAR.** Scanning LiDAR cannot differentiate one-bounce vs three-bounce returns. **(c) Flash LiDAR.** Flash LiDAR can detect two-bounce returns from co-linear paths, as well as one-bounce and three-bounce returns.

these ambiguities, as shown in Fig. 2(c). Two-bounce light paths occur when light is emitted along a ray that is different from the ray that the light returns from, as shown in Fig. 2(c). Measuring two-bounce light allows us to measure light bouncing either from specular to diffuse surface or from diffuse to specular surface. The presence of two-bounce light indicates the presence of a specular surface, enabling detection of real and virtual objects.

1.3 Two-bounce Flash LiDAR.

Flash LiDAR emits multiple laser spots at once and enables detection of two-bounce returns. Henley et al. [1] demonstrate proof-of-concept results of utilizing a simulated flash LiDAR for specular surface mapping. However, their approach relies on a non-confocal illumination setup, where the receiver is able to densely sample the scene spatially using mechanical scanning. As shown in Fig. 3, dense sampling of the receiver enables classification of two-bounce returns by using the fact that non-confocal returns must be two-bounce. In practical scenarios, however, it is infeasible to have such high-resolution non-confocal captures on consumer devices, which are currently often limited to 12×12 spatial resolution, making two-bounce classification challenging. In our work, we show how to disambiguate one-bounce, two-bounce, and three-bounce returns for specular surface mapping.

1.4 Key Insight: Reciprocal Pairs and Multiple Frames.

To overcome the low spatial resolution of consumer-grade LiDAR sensors, we introduce the reciprocal pair constraints to classify one-, two-, and three-bounce light. The reciprocal pair constraints is based on the observation that a specular surface can observe two- and three-bounce light, and a diffuse point can observe one- and two-bounce light. Combining this observation with Helmholtz reciprocity, we are able to derive conditions to detect pixels that measured light traversing the same paths, but in opposite directions. These reciprocal pair pixels can then be used to triangulate the position of diffuse and specular points in the scene. By moving the handheld device and scanning the scene in an unstructured manner, we can detect more reciprocal pair pixels and gradually accumulate a point cloud of the mirror. The reciprocal pair constraints and camera motion enable us to solve the mirror mapping problem even with low spatial resolution and multiplexed illumination on consumer-grade smartphone LiDAR devices.

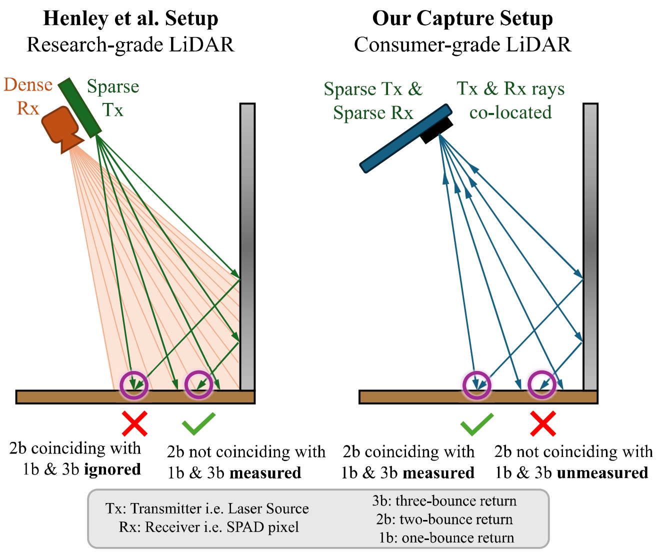


Fig. 3. **Comparing our consumer-grade capture setting with prior research-grade capture work.** Henley et al. [1] consider a research-grade setup with receiver (Rx) having a dense grid of SPAD pixels and thus can measure two-bounce returns that do not coincide with one-bounce return, enabling a straightforward separation of one-, two- and three-bounce returns. In this work, we rely on a sparse grid of 12×12 pixels in smartphone LiDAR. In this setting, we can only measure two-bounce and three-bounce returns that coincide with one-bounce returns making it challenging to identify two-bounce returns.

1.5 Our Contributions.

The key challenge with our problem statement is that we are relying on consumer-grade LiDAR cameras that have low spatial resolution and flash illumination. As a result, detection and classification of one-, two-, and three-bounce signals becomes challenging, unlike the setup used by Henley et al. [1], which had a dense grid of pixels. Our key contribution is to derive a set of reciprocal pair constraints that enable this classification, from which we can identify diffuse and specular pixels and map their 3D locations. To the best of our knowledge, we are the first to develop an algorithm that utilizes multi-beam returns for mirror mapping on consumer-grade LiDAR. Finally, we demonstrate that harnessing multi-bounce returns offers benefits beyond 3D reconstruction, extending to applications like segmentation and novel view synthesis. We improved over other state-of-art learned-based models on these applications, particularly in handling ambiguous scenes. Code will be released.

Our contributions are summarized as follows:

- **Specular Mapping with Consumer-Grade Devices:** We demonstrate specular mapping with consumer-grade LiDARs that have low spatial resolution and flash illumination.
- **Reciprocal Pair Constraints:** We derive constraints that enable classification of one-, two-, and three-bounce light and use these constraints to map diffuse and specular surfaces.
- **Downstream Performance Improvement:** Our multi-bounce LiDAR specular mapping technique enhances the performance of existing specular segmentation, reconstruction, and novel view synthesis techniques that use RGB information.

2 RELATED WORKS

2.1 RGB-based Mirror Mapping

Early works in specular surface estimation from standard RGB cameras use geometric techniques to leverage the optical flow of reflections under a moving camera [2], [3], [4], the observation of specular highlights in varying light or camera position [5], [6], [7], and the distortions of known printed [8], [9], [10] or projected patterns [11], [12], [13], [14], [15], [16]. Supervised, or learned-based techniques have been proposed for per-pixel segmentation of mirrors [17], [18]. These methods rely on RGB image cues, such as the mirror frame, texture discontinuity inside and outside the mirror, that may be confusing in some scenarios. Recently neural radiance field based-techniques [19], [20], [21] have focused on novel view synthesis in the presence of specular and gloss surfaces by modeling multi-view reflections. MS-NeRF [19] cannot estimate the mirror depth correctly, and MirrorNeRF [20] requires correct mirror segmentation as inputs. Our method is physic-based that computes the correct segment and depth to the mirror, and so is more generalized to different scenes.

2.2 Dense-depth-based Mirror Mapping

Mei et al. [22] fuse RGB and ToF cameras, using ToF depth discontinuities as a mirror cue. However, this approach struggles to distinguish actual open spaces in the scene (e.g. doorways) from virtual holes caused by mirrors. Our multi-bounce modeling provides a physics-based distinction between real scene holes and virtual depth holes caused by mirror reflections being mistaken as direct line-of-sight.

2.3 LiDAR-based Mirror Mapping

Other LiDAR methods rely on glare [23] or edge [24] detection, while we do not require these potentially ambiguous cues. Yang et al. [25] exploit LiDAR symmetry, but only model a specific three-bounce corridor case. We handle more general layouts analyzing one, two, and three bounces. Raskar et al. [26] propose two-bounce ToF to map specular surfaces, but without direct scene illumination. Kutulakos et al. [27] reconstruct specular surfaces through triangulation, they assume a known 3D reference point, whereas we make no such assumptions. O'Toole et al. [28] reconstruct specular geometries, but possess full control over the illumination pattern via a projector and can capture multiple images, making reconstruction well-posed. We show a method that can identify reciprocal pairs with a fixed illumination pattern and a single capture, which is more ill-posed.

Henley et al. also exploit multi-bounce returns [1], by using a single-beam and multi-beam flash to scan glass or mirror surfaces. However, their method assumes two-bounce returns can only arrive at the receiver from directions that are not coincident with any transmitted light rays, by interpolating between one-bounce and three-bounce returns. Their method lacks the ability to distinguish two-bounce returns detected by typical, consumer-grade flash lidars that can only receive along the directions of transmitted beams. As a result, their method is unsuitable for baseline comparison as it renders no detection on our device. Our method, by contrast, was designed to be used on consumer-grade flash lidar systems with coarse, low-resolution receivers. Using reciprocal pair constraints, we are able to distinguish between one-, two-, and three-bounce signals that return along transmitted beam paths.

3 RECIPROCAL PAIR CONSTRAINTS

Scanning a specular surface with conventional lidar systems is challenging because mirrors only reflect light in one direction, often away from the lidar sensor. As a result, the sensor doesn't receive any physical information that can be used to estimate the mirror geometry. In this section, we will describe how we can exploit multi-bounce signals from a consumer-grade handheld flash lidar. We first derive the reciprocal pair constraints, which enables classification of one-, two-, and three-bounce light. We then used the detected reciprocal pair pixels and the corresponding bounce classification to map the 3D geometry of the specular and diffuse surfaces.

3.1 Experimental Setup.

Our lidar system consists of a pulsed laser and SPAD array sensor. The pulsed laser simultaneously illuminates a grid of 12 × 12 laser spots, and each pixel in a 12 × 12 grid in the SPAD array is viewing one of these 144 spots. The laser spots and pixels have same viewing directions, and thus illuminate and image the same scene point. For consumer-grade devices, we assume that the sensor and laser are roughly co-located. This property enables us to derive convenient geometric relationships between the time-of-flight measurements certain sets of pixels, referred to as reciprocal pairs.

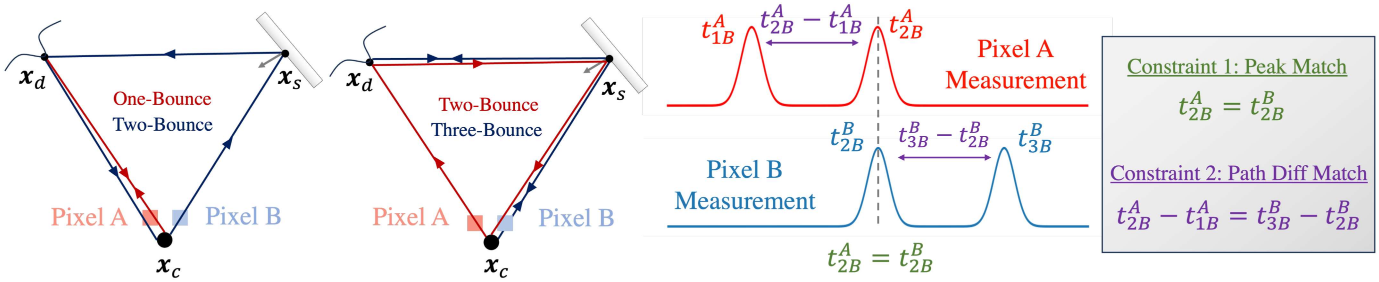
3.2 Reciprocal Pair Pixels.

Reciprocal pairs are pairs of pixels that share the same multi-bounce light paths. Consider the scene in Fig. 4. Pixel A is observing a point $\mathbf{x}_d$ on the diffuse surface and pixel B is observing a point $\mathbf{x}_s$ on the specular surface. Suppose that pixel B emits light towards $\mathbf{x}_s$. By Snell's Law, the incident light will be reflected by angle θ relative to the surface normal towards $\mathbf{x}_d$. The light from $\mathbf{x}_d$ will then be measured by pixel A because diffuse surfaces reflect light in all directions. The position of $\mathbf{x}_d$ relative to the surface normal at $\mathbf{x}_s$ ensures that the incident laser pulse will be reflected to $\mathbf{x}_d$.

Pixel A Measurement. Pixel A is observing a diffuse point $\mathbf{x}_d$, and therefore will measure one-bounce and two-bounce light. The one-bounce light travels the path $\mathbf{x}_c^A \to \mathbf{x_d} \to \mathbf{x}_c^A$. The two-bounce light originates from the laser at pixel B and travels the path $\mathbf{x}_c^B \to \mathbf{x}_s \to \mathbf{x}_d \to \mathbf{x}_c^A$.

Pixel B Measurement. Pixel B is observing a specular point $\mathbf{x}_s$, and therefore will measure two-bounce and three-bounce light, but not one-bounce light. The two-bounce light travels the path $\mathbf{x}_c^A \to \mathbf{x}_d \to \mathbf{x}_s \to \mathbf{x}_c^B$. The three-bounce light travels the path $\mathbf{x}_c^B \to \mathbf{x}_s \to \mathbf{x}_d \to \mathbf{x}_s \to \mathbf{x}_c^B$.

Estimating Geometry from Reciprocal Pairs. Using the one-bounce and two-bounce signals measured at pixel A, we can estimate the positions of $\mathbf{x}_d$ and $\mathbf{x}_s$. The distance from the



(a) Measurement at Pixel A. (b) Measurement at Pixel B. (c) Reciprocal Pair Constraints.

Fig. 4. **Multi-Bounce Flash Lidar. (a)** Pixel A, observing a diffuse point $\mathbf{x}_d$, measures one- and two-bounce light. **(b)** Pixel B, observing a specular point $\mathbf{x}_s$, measures two- and three-bounce light. **(c)** Reciprocal Pair Constraints. For **(a)** and **(b)** *only*, rays in red correspond to laser spots emitted by pixel A, and rays in blue correspond to laser spots emitted by pixel B.

camera to the diffuse point $|\mathbf{x}_d - \mathbf{x}_c|$ can be estimated using the one-bounce signal as

$$r_d = |\mathbf{x}_d - \mathbf{x}_c| = \frac{ct^A_{1B}}{2}, \tag{1}$$

where c is the speed of light, t^A_{1B} is the time-of-flight of the one-bounce light measured at pixel A. Once $\mathbf{x}_d$ is known, the position of the specular point $\mathbf{x}_s$ can be computed as a ray-ellipsoid intersection.

$$\mathbf{x}_s = \textsc{Intersect}(\mathbf{r}^B, \mathcal{E}(\mathbf{x}_c, \mathbf{x}_d, ct^A_{2B} - r_d)), \tag{2}$$

where $\mathbf{r}^B$ corresponds to the viewing direction of pixel B and $\mathcal{E}(\mathbf{f}_1, \mathbf{f}_2, a)$ is a 3D spheroid ellipsoid with focus at $\mathbf{f}_1$ and $\mathbf{f}_2$, major axis length a, minor axes length $b = \sqrt{a^2 - f^2}$, and focal length $f = |\mathbf{f}_1 - \mathbf{f}_2|$. An analytical expression for $\mathbf{x}_s$ is provided in the Supplementary.

Problem Statement. The key insight is that reciprocal pair pixels enable mapping of a single point on a specular surface. However, there are two key challenges with using consumer-grade hardware for such techniques. (1) It is challenging to determine which two pixels are reciprocal pairs due to flash illumination. (2) Due to the sparse number of pixels, there is limited multi-bounce information. In the following subsection, we will discuss how to algorithmically mitigate these two challenges.

3.3 Detecting Reciprocal Pairs in Flash Lidar

In practice, it is unknown which two pixels in a 12 × 12 array are reciprocal pairs, how many reciprocal pairs there are, and if any reciprocal pairs exist at all. In this section, we will introduce techniques to detect reciprocal pairs using two constraints: peak match and path diff match. Both constraints hold true under the assumption that the laser and SPAD sensor are roughly co-located. The constraints are visually explained in Fig. 4(c).

3.3.1 Constraint 1: Peak Match

The peak match constraint is derived from Helmholtz reciprocity. Consider the two-bounce returns that arrive at pixel A and pixel B in Fig. 4. The two-bounce light arriving at pixel A travels along the path $\mathbf{x}^B_c \rightarrow \mathbf{x}_s \rightarrow \mathbf{x}_d \rightarrow \mathbf{x}^A_c$. Similarly, the two-bounce light arriving at pixel B travels along the path $\mathbf{x}^A_c \rightarrow \mathbf{x}_s \rightarrow \mathbf{x}_d \rightarrow \mathbf{x}^B_c$. Because $\mathbf{x}^A_c = \mathbf{x}^B_c = \mathbf{x}_c$, these traversed paths are precisely the same. By Helmholtz Principle, the two traveling rays experience the same optical "adventure" (i.e. path), but in reverse directions. Therefore, they also share the same pathlength. The peak match constraint for reciprocal pair pixels can be expressed as

$$t^A_{2B} = t^B_{2B}. \tag{3}$$

In Fig. 4, this equation can be understood by analyzing the blue paths in (a) and the red paths in (b). They are the same paths, but in opposite directions.

3.3.2 Constraint 2: Path Diff Match

While the peak match constraint is derived with respect to two-bounce light, the path diff match constraint is derived with respect to one-bounce, two-bounce, and three-bounce light. Recall that a reciprocal pair contains one pixel observing a specular surface and one pixel observing a diffuse surface. The pixel observing the diffuse surface measures one- and two-bounce light, and the pixel observing the specular surface observes two- and three-bounce light. Mathematically, the path diff match constraint can be expressed as

$$t^A_{2B} - t^A_{1B} = t^B_{3B} - t^B_{2B}. \tag{4}$$

This constraint can be better understood in the context of Fig. 4(a) and (b) by subtracting the three-bounce pathlength from the two-bounce pathlength, and the two-bounce pathlength and the one-bounce pathlength. The resulting pathlength is $|\mathbf{x}_c - \mathbf{x}_d| + |\mathbf{x}_d - \mathbf{x}_s| + |\mathbf{x}_s - \mathbf{x}_c|$, which forms a triangle between $\mathbf{x}_c$, $\mathbf{x}_d$, and $\mathbf{x}_s$.

3.4 Specular Surface Estimation

To reconstruct specular surface with a consumer-grade lidar sensor, we (1) detect reciprocal pair pixels (if any are detected) using the constraints in 3.3, (2) determine which pixel is observing a specular point and which pixel is observing a diffuse point, and (3) analytically compute the distance using the equations in 6. Each pixel measurement will contain up to two echos. An ECHO is defined as a measured pulse return. If a pixel only receives one ECHO, that pixel is assumed to be viewing a diffuse surface and the echo is one-bounce light. If a pixel receives two echos, then it measures two returning pulses. In this case, the first echo is either one-bounce or two-bounce light and the second echo

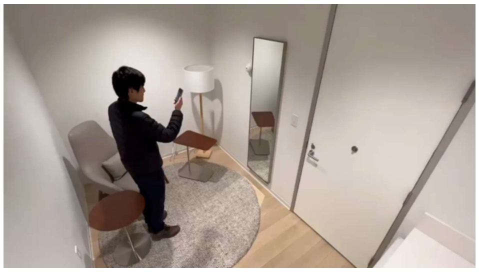


Fig. 5. **Experimental Setup**. We use a handheld device for 3D room scanning, eliminating the need for complex research-grade equipment or manual calibration.

is either two-bounce or three-bounce light, as shown by the possible scenarios in Fig. 4(c).

Resolving Echo Ambiguity. To resolve this echo ambiguity, we first apply the peak match constraint by computing pairwise differences between ECHO_1 and ECHO_2 for all pixels. Any pixel pair that satisfies $\mathrm{ECHO}_1^B - \mathrm{ECHO}_2^A = 0$ satisfies the peak match constraint, as shown in Fig. 4(c). Then, we test the path diff match constraint for all pixel pairs (A, B) that satisfied the peak match constraint. We perform this by computing $|\mathrm{ECHO}_2^A - \mathrm{ECHO}_1^A| - |\mathrm{ECHO}_2^B - \mathrm{ECHO}_1^B|$. Pixel pairs that satisfy this second condition are reciprocal pairs. For these reciprocal pair pixels, we can determine which echoes correspond to the one-, two-, and three-bounce signals by referring to the time-of-flight ordering in Fig. 4(c).

Estimating the Scene. Once we determine the one-, two-, and three-bounce signals in the reciprocal pair, we perform step (2) listed above. Step (2) can be accomplished by looking at the relative time-of-flight of the two reciprocal pair pixels. As shown in Fig. 4, one-bounce light arrives before three-bounce light. Therefore, by identifying which pixel measures one-bounce and three-bounce light, we can determine the diffuse and specular pixel. Lastly, once we determine the diffuse and specular pixel, we can compute the 3D position of diffuse and specular points using eq. (1) and eq. (2). Note that many captured frames will not have reciprocal pairs due to the happenstance of such reciprocal pairs. However, we show experimentally that these reciprocal pairs will appear periodically, and are sufficient to estimate the mirror geometry.

4 RESULTS AND ANALYSIS

4.1 Experimental Setup

We use a consumer-grade smartphone LiDAR device. This device contains a low-spatial-resolution array of SPAD pixels. Each Rx pixel of the sensor array has a corresponding Tx laser emitter, such that the Tx and Rx are both pointing in the same direction and are approximately co-located. The viewing direction of each pixel (i.e. camera rays) is known through prior camera calibration. Each pixel measures a histogram of intensities. These histograms are pre-processed such that each pixel outputs (up to) two echos. The pre-processing step consists of peak finding and computation of a confidence score based on SNR for each peak (or echo). From this pre-processing step, our method receives three inputs: (1) camera rays for each pixel, (2) detected echos for each pixel, and (3) the confidence score for each echo. We filter out echoes with lower confidence, with the threshold treated as a hyperparameter.

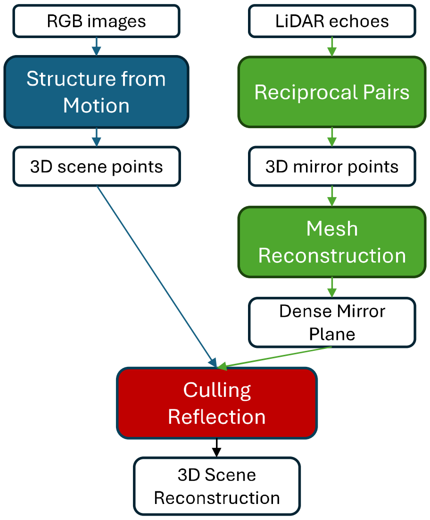


Fig. 6. **Dense 3D Mirror Reconstruction Pipeline.** The dense 3D scene points and camera poses are computed from RGB images via structure-from-motion (SfM). 3D mirror surface points are computed using the reciprocal pair algorithm from the LiDAR. Spurious mirror points are filtered and densified with Poisson mesh reconstruction. After mirror surface is reconstructed, incorrect reflection points from SfM that are behind the mirror surfaces are removed with a culling algorithm.

We conduct scans of real-world rooms by freely moving around with the handheld device, without need for complex research-grade equipment or manual calibration, as shown in Fig. 5. The rooms are naturally illuminated at full brightness by the existing room lights. We present the results of mirror surface reconstruction and demonstrate how our method enhances segmentation and novel view synthesis techniques.

4.2 Dense 3D Mirror Reconstruction

In this section, we outline the procedure for dense 3D mirror reconstruction. Mirror detection is accomplished using the reciprocal pair algorithm, and the scene is generated from structure-from-motion (SfM). Accurately locating the mirror surface allows for the removal of incorrect reflections.

Dense Scene Map from RGB Refer to Fig. 6 for the pipeline. From RGB images, we use structure-from-motion (SfM) to reconstruct dense 3D scene points. SfM is readily available on smartphones either with ARKit or ARCore libraries, or COLMAP [29] on personal computers. Due to the low spatial resolution of the LiDAR, we resort to dense scene reconstruction from RGB images for better visualization and pose estimation. Attempts have been made to reconstruct the scene solely with LiDAR 3D points. However, the scanned data quality is too sparse and noisy, leading LiDAR-based pose registration algorithms like Iterative Closest Point (ICP) [30] to drift after only a few frames.

3D Mirror Points from LiDAR Given the input of two echoes per pixel of the Rx, we first resolve the association problem via the reciprocal pair constraints to determine which echo is one-, two-, or three-bounce return. Next,

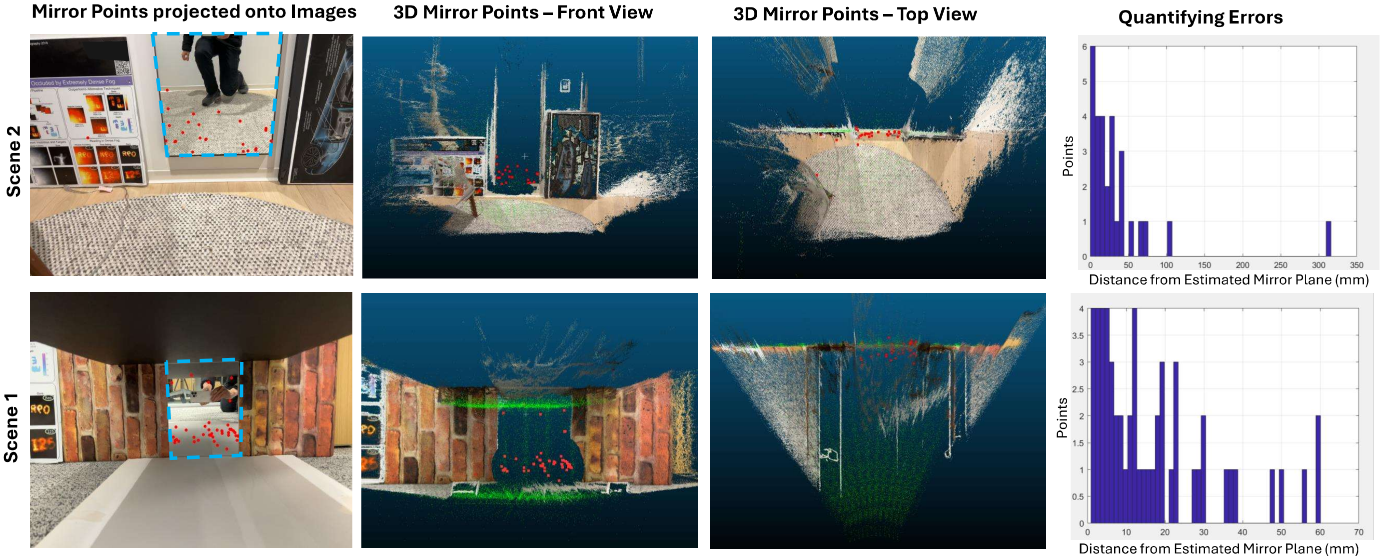
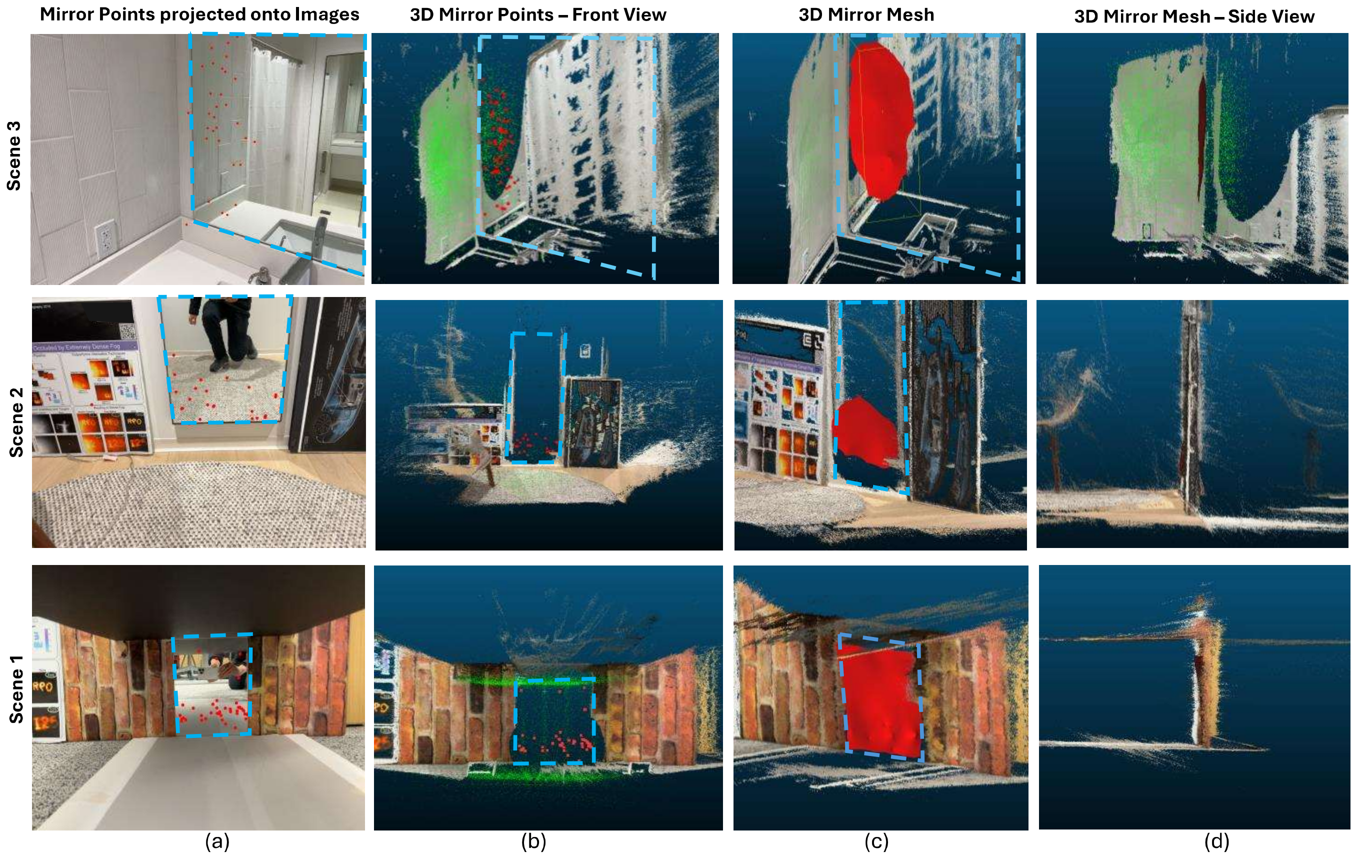


Fig. 7. **Quantitative Analysis of Detected 3D Mirror Points.** First column shows the detected mirror points projected onto an image, shown as red points. The true mirror surface is highlighted with dashed blue lines. Second and third columns show the detected points in 3D, for front and top views respectively. The dense 3D scene is generated from RGB images via structure-from-motion. LiDAR's first echo is projected onto the 3D scene as green points for visualization. Forth column shows histograms of the distance of each 3D mirror points to an estimated mirror plane.



(a) (b) (c) (d)

Fig. 8. **Dense 3D Reconstruction of Mirror Mesh Surface.** Scene 3 uses the sidewall for mirror mapping. Scene 2 uses the floor to map the large mirror. Scene 3 uses both the floor and ceiling to map mirror points on the top and bottom of the mirror, to reconstruct nearly all of the mirror surface. Blue dashed lines outlines the true mirror, detected 3D mirror points are shown as red points, and mirror surface is shown as red surface.

based on the analytical equations, we compute the distance to the diffuse surface r_{dc} using eq. (5). Given r_{dc}, we then compute the distance from LiDAR to the specular surfaces r_{sc} using eq. (6). Finally, given r_{sc}, the final 3D mirror points are reconstructed using eq. (7). At this stage, we need to integrate the 3D mirror points (from LiDAR) with the 3D scene points (from RGB images). Since SfM operates in a different world frame compared to LiDAR, we begin by projecting the 3D mirror points onto the 2D image plane using the known calibration between the LiDAR and camera sensors. Subsequently, the 2D mirror points are unprojected to 3D into the SfM world frame, with depth values up to a scale factor.

Dense Mirror Surface Reconstruction Due to the limited spatial resolution, typically only a few (0-4) reciprocal pairs (and mirror points) are detected per frame. Therefore, we accumulate points over multiple frames to create a denser mirror surface. The poses of each frame are based on SfM estimation. We refine the 3D mirror points into a denser mirror surface using Poisson mesh reconstruction, which is more versatile compared to plane fitting. The reciprocal pair algorithm does not assume a planar specular surface, and similarly, we avoid assuming a planar mirror surface during dense surface reconstruction. Another benefit of the reciprocal pair algorithm is that it allows derivation of the normal for each mirror point, although our current results do not yet include this feature.

Culling Reflection The scene generated from SfM incorrectly places reflections behind the mirror surface, assuming all RGB camera rays are one-bounce. Our method accurately locates the true 3D mirror surface, enabling the removal of incorrect reflections, illustrated by Fig. 9. The core concept involves checking if incorrectly placed reflection points, when projected onto the LiDAR's view frustum, intersect with the true mirror plane. If they do, these points are identified as reflections and not part of the diffuse scene points. In practice, we first compute the convex hull polygon of the detected mirror points in the 2D image. This convex hull serves as the LiDAR's frustum counterpart. By projecting all the 3D scene points onto the image plane, only the reflection points would be inside the convex hull and so are removed.

4.2.1 Quantitative Analysis

Fig. 7 shows the mirror reconstruction quantitative results for two scenes. Scene 1 features a small mirror approximately 1 foot tall, accompanied by a synthetic ceiling. The objective is to detect points on the top of the mirror using multi-bounce returns from the ceiling, alongside the lower part of the mirror from the floor. Scene 2 features a large mirror hanging on the walls, using only multi-bounce returns from the floor. Poster boards were positioned along the wall surface to assist with SfM pose registration and do not affect mirror point detection. First column shows the 3D mirror points projected onto the image (red points), most of which lie within the true mirror surface (dashed blue lines). Second column shows the quality of the 3D mirror points in a front-view, while the third column shows the top view. The fourth column quantifies the errors of the detected mirror points with a histogram showing the distances from the mirror plane.

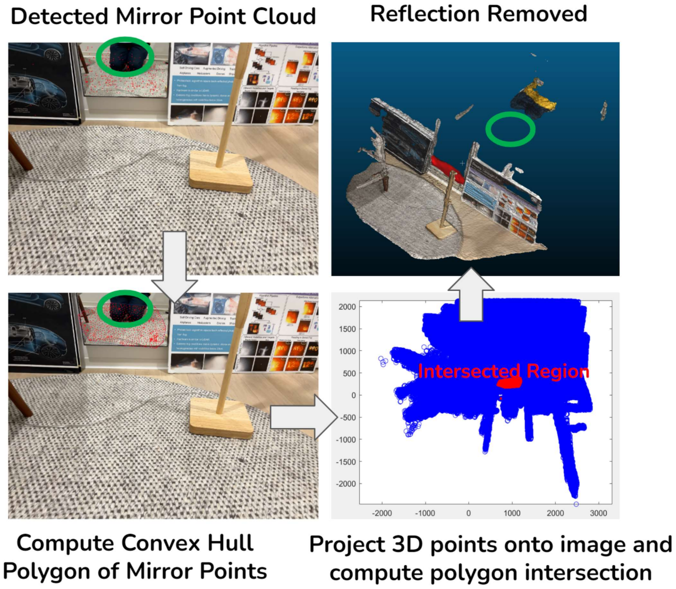


Fig. 9. **Culling Reflection with 3D Mirror Reconstruction.** Green circle shows the removal of incorrectly reflection points constructed from RGB-based SfM that are behind the true mirror surface reconstructed by our method.

To estimate the ground truth for the 3D mirror surface plane, we assume the mirror plane aligns with the back wall it hangs on. We employ RANSAC [31] plane fitting on the first echo (which resembles the back wall) to estimate this plane. We plot the histogram showing the error, which is the shortest distance from each 3D point to the estimated plane. Excluding the outliers, both scenes indicate that the points are within 60mm of the mirror plane, with the majority being within 20mm of the mirror plane. Other sources of errors include LiDAR sensor noise, causing inaccuracies in mirror plane estimation (particularly in scenes with few points from the back wall), errors in SfM registration, depth scaling discrepancies between LiDAR and SfM coordinate frames, and floating-point precision issues related to the equality constraints of reciprocal pairs (where values under 10mm are considered equal).

4.2.2 Qualitative Analysis

For scene 1 and 2, we include the dense mirror surface reconstruction from the mesh filtering, in Fig. 8(c). In scene 1, the mirror surface nearly covers the entire mirror as it has detection from top and bottom of the mirror. In scene 2, most of the bottom of the large mirror is covered. Scene 3 utilizes the left side wall of the mirror to reconstruct most of its left side. Notably, in scene 3, there is no back wall aligned with the mirror for quantifying errors. Note that all results are shown with the incorrect reflections behind the mirror removed.

4.3 Mapping Glass

We demonstrate our reciprocal pair algorithm on other specular surfaces such as glass. In Fig. 10, we map glass doors and glass walls. Glass is more challenging to detect due to its imperfectly specular nature, resulting in weaker and noisier signals. In (a), we map the bottom of the glass


(a) Glass Door

(b) Glass Wall

Fig. 10. **Mapping Glass Surfaces.** We show that our algorithm can detect other specular points such as (a) glass door and (b) glass wall. Glass door leverages multi-bounce returns from the floor, and Glass wall from both the floor and ceiling. Note that glass is more challenging than mirrors because of weaker and noisier signals. Detected points are shown as red points.

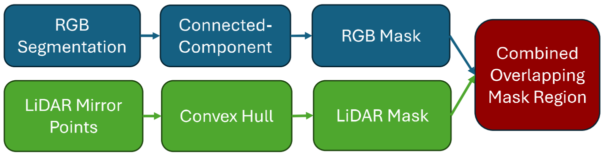


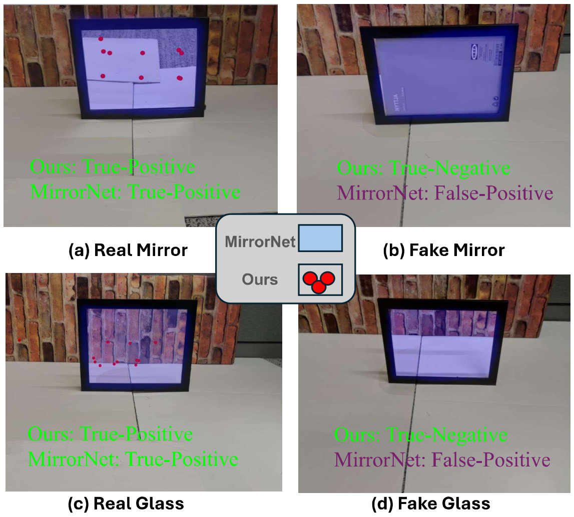
Fig. 11. **Pipeline for refining RGB mirror segmentation with LiDAR.**

door by bouncing off the floor surface. In (b), we map the top and bottom sections of the glass wall on the right side of the door, bouncing off the floor and the ceiling.

4.4 Enhancing RGB-based Mirror Segmentation with LiDAR

In addition to 3D reconstruction, we demonstrate how leveraging multi-bounce returns from LiDAR can help in mirror segmentation. Fig. 11 shows the segmentation pipeline for combining both RGB and LiDAR results. For RGB, we use learned-based model MirrorNet [32] as demonstration. First, we binarize the probability output mask from MirrorNet and then use connected-components to separate different mask objects. From the LiDAR 3D mirror points, we apply convex-hull to obtain the polygon outline of the mask. Subsequently, we determine the true mirror object mask by overlapping the RGB mask with the LiDAR mask. Finally, we apply a fill-in polygon algorithm to address any holes within the mask.

Fig. 12. **Refining RGB Segmentation with LiDAR.** LiDAR corrects the incorrect mirror mask from RGB segmentation. First column shows the incorrect mirror mask detected by RGB-based segmentator (MirrorNet), indicated by the yellow arrows. Second column shows masks from LiDAR mirror detection (mesh reconstruction from the detected points). Third column shows the combined mirror masks from both sensors. RGB gives a denser detection, while LiDAR can identifies the true mirror segment.



(a) Real Mirror

(b) Fake Mirror

(c) Real Glass

(d) Fake Glass

Fig. 13. **Enhancing Robustness of Segmentation Detector. (a) True Mirror:** Our algorithm successfully detects the mirror, shown as red points. MirrorNet [32] correctly identifies the mirror, shown in blue tint. **(b) Fake Mirror:** The frame contains only paper. MirrorNet misidentifies the paper as a mirror, while our algorithm rejects it. **(c) True Glass:** Glass is present inside the frame. Both our algorithm and MirrorNet detect it accurately. **(d) Fake Glass:** The frame is empty inside. MirrorNet incorrectly classifies the empty frame as a mirror, whereas our algorithm correctly rejects it.

Fig. 12 illustrates how we correct incorrect mirror segmentation from any RGB-based detector. The first column shows ambiguous mirror detection from MirrorNet, possibly due to the absence of most of the ‘mirror frame’ cue in the image. Incorrect detections are indicated by the yellow arrows. The second column displays the LiDAR mask, which is more accurate but may only capture part of the mirror due to the reciprocal pair algorithm requiring adjacent surfaces for multi-bounce returns. By leveraging the strengths of both methods, the LiDAR mask identifies the true-positive areas, while the RGB mask helps detect a denser mirror segment, resulting in a clean segmentation as shown in the third column.

For our analysis, we set up scenarios involving ambiguous objects such as real and fake mirrors and glass. As depicted in Fig. 13, we test: (a) a true mirror, (b) a fake mirror with paper inside mimicking a mirror texture, (c) a true glass, and (d) a fake glass with nothing inside the frame, resembling transparent glass.

Our algorithm correctly detects the true mirror (a) and true glass (c), as indicated by the red points in the detections. Additionally, we successfully reject the fake mirror (b) and fake glass (d) cases. MirrorNet [32], which primarily relies on cues from RGB images, fails on the fake mirror and fake glass scenarios (although to its credit, it is not trained on glass). It mistakenly detects glass as mirror, possibly basing off the cue of ‘mirror frame’.

The model [22], which trains on dense depth images using depth discontinuities as cues, would also fail on the

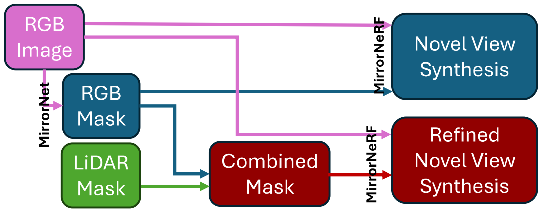


Fig. 14. **Pipeline for enhancing novel view synthesis with LiDAR.** MirrorNeRF requires mirror mask as input for novel view synthesis. We show the combined mask of from RGB and LiDAR masks produces better result from pure RGB mask of MirrorNet [32].

empty frame case (d), as it does not effectively distinguish between one-bounce and three-bounce returns. This limitation was shown in the doorway scene of their paper. Our physics-based 3D point detection can be seamlessly integrated with any RGB-based learned model for joint feature learning [33], harnessing the strengths of both approaches.

4.5 Enhancing Novel View Synthesis with LiDAR

We demonstrate how multi-bounce LiDAR returns can enhance a novel view synthesis model. The RGB-based learned model MirrorNeRF [20] relies on the accurate mirror mask as input for its Neural Radiance Field (NeRF) model. Creating a mirror mask manually involves labor-intensive labeling, so we have developed an automated solution for generating mirror masks. we present two pipelines: one using the result generated from the RGB-based mask using MirrorNet, and another using the combined mask from RGB and LiDAR masks, as discussed in Section 4.4.

The results are shown in Fig. 15. First column shows the input sensor modalities. Second column shows the generated masks, where the RGB-based method incorrectly identifies part of the poster as the mirror, indicated by the yellow arrow. Given the mirror mask as input, third column shows the novel view synthesis results from MirrorNeRF. Qualitatively, the part of the poster that is incorrectly identified as mirror by RGB-based method appears blurrier than the combined RGB+LiDAR method. Quantitatively, the PSNR values of RGB+LiDAR method is also higher than RGB method, 24.1 vs 22.5 respectively.

The PSNR improvement for the novel view synthesis result appears to be marginal. However, our results can handle edge cases better, which the PSNR metric cannot capture. Detecting specular points reduces the chances of detecting diffuse surface as a specular one and vice versa and avoids artifacts in rendering due to these false detections. The artifacts are sparse and localized. Thus our improvements have marginal effect on the overall PSNR, but enhance the robustness of novel view rendering techniques under specular surfaces.

5 DISCUSSION AND LIMITATIONS.

5.0.1 *Single Reciprocal Pair and Duplicates.*

We analyze reciprocal pair detection in a single frame in Fig. 17. Visually, the mirror points (red triangles) correctly lie in the empty space along the wall. The two-bounce points (cyan) and three-bounce points (red $+$) all align in a single line directed towards a single pixel of the receiver. This alignment adheres to the reciprocal pair assumption, where the two-bounce and three-bounce returns are echoes at the same pixel spot. As shown in (b), the three-bounce point is the 'mirror image' of the one-bounce point, symmetric over the plane of the mirror, at the same distance d. The two-bounce point appears off-plane from the diffuse surface.

Due to the thresholding of floating-point number equality with the reciprocal pair constraints, we may get duplicates for the same diffuse point Fig. 18(a). We remove duplicate pair by choosing the pair with minimum overall errors from the constraints Fig. 18(b).

5.0.2 *Motion and Angle of Scanning*

We investigate our ability to recover reciprocal pairs from various angles during scanning. When scanning, we try to keep the mirror surface and diffuse surface as 50/50 split in the receiver FoV. Using different motions (translation and rotation), we demonstrate our capability to recover reciprocal pairs from different angles. Fig. 16 shows a heatmap of number of detections of each pixel in the Rx. In cases of nearly static motion (left), detections are concentrated mostly in the middle rows from similar angles. With larger motions (right), we observe a wider spread of detections across the array, indicating successful capture of reciprocal pairs from diverse angles.

5.0.3 *Limitations*

Our method only reconstructs a sparse set of points and has limited accuracy. These limitations arise due to the limited sensor spatio-temporal resolution because (a) fewer reciprocal pairs are detected, (b) ambiguity between reciprocal pair candidates increases, and (c) 3D ranging resolution worsens. These effects will be mitigated and reconstructions will improve as the quality of commercial-grade LiDAR sensors improves, which is likely as they are becoming more ubiquitous. However, the goal of this paper was *not* to demonstrate high-quality 3D reconstruction, but rather to demonstrate that the use of multi-bounce signals (even with coarse resolution) can be useful for handling specularity in vision tasks.

The reciprocal pair constraints necessitate the presence of reciprocity between incoming and outgoing lights within the sensor's field-of-view (FoV). This implies that our algorithm requires both specular and bounce surfaces to be within the FoV. However, this assumption is valid because many sensors have wide FOVs (45°) and many real-world scenes are $>$ 1-2 meter away, including those in our capture setups. In the scenario where large mirrors extend beyond the FoV, our method primarily captures the edges of the mirrors. Additionally, for multi-bounce triangulation to be effective, both specular and diffuse surfaces must be within a geometry where their angles are (90°) or less. However, we do not assume that the specular surfaces need to be planar.

Lastly, spurious detections can occur due to sensor noise or inter-reflection light paths that undergoes more than three bounces from nearby objects in the scene. In Fig. 19 (c) and (d), returns from sofa give erroneous mirror point detection, where the lights undergo more than three bounces between sofa, floor, and the mirror surfaces.

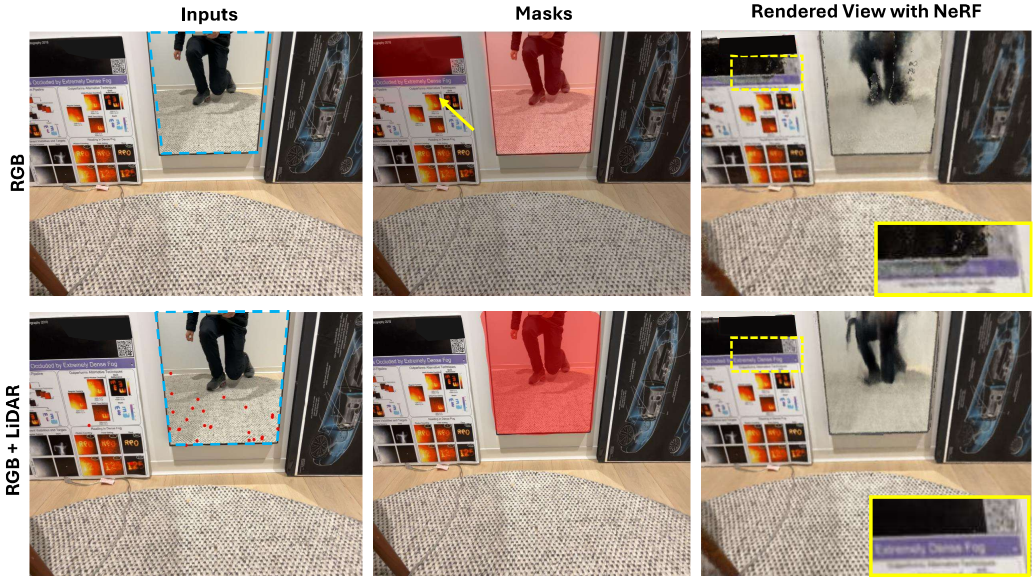


Fig. 15. **Enhancing Novel View Synthesis (NeRF) with LiDAR.** First column shows input sensor. RGB-only for the first row, and RGB+LiDAR for the second row. The mirror surface is highlighted with dashed blue lines. Red points shows the LiDAR mirror points. Second column shows the masks of mirror segmentation. RGB detector (first row) incorrectly segments part of the left poster as mirror, shown by the yellow arrow. By combining RGB and LiDAR masks, the correct mirror segmentation is constructed (second row). Third column shows the novel view synthesis with Mirror-NeRF that requires mirror masks as input. Incorrect masking from RGB segmentation results in worse rendered view comapred to the corrected mask from RGB+LiDAR segmentation. Yellow boxes show the zoomed-in regions. RGB PSNR: 22.5, RGB+LiDAR PSNR: 24.1.

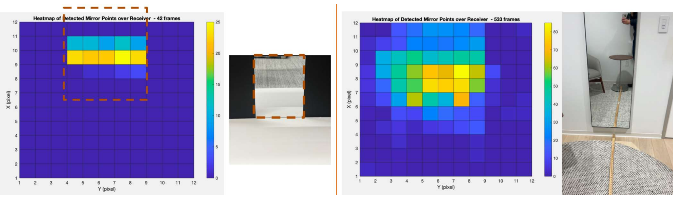
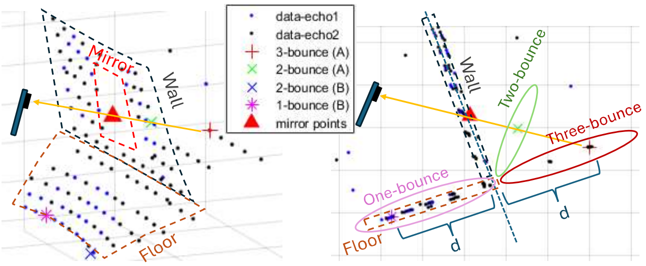


Fig. 16. **Larger Motion Increases Number of Reciprocal Pair Detections.** Heatmap showing detection of the 12×12 SPAD array, where each cell indicates number of detections. The top half of the array corresponds to a mirror, and the bottom corresponds to the floor. (Left) controlled scene with minor phone movements, where the detection is mostly concentrated in the middle rows of the mirror. (Right) scene with large phone movements, we can observe the detections are more widely spread out. This indicates that our reciprocal pair method can effectively detect across a wider region of the receiver at various angles with more motions.



(a) 45^{0} View

(b) Side View

Fig. 17. **Reciprocal Pair Detection in a Single Frame.** The three-bounce, two-bounce, and mirror points lie along the same line (yellow arrow) toward the sensor. (b) The three-bounce point is the 'mirror image' of the one-bounce point, symmetric over the plane of the mirror, with the same distance d. Two-bounce appears as off-plane from the diffuse surface.

(a) Duplicates

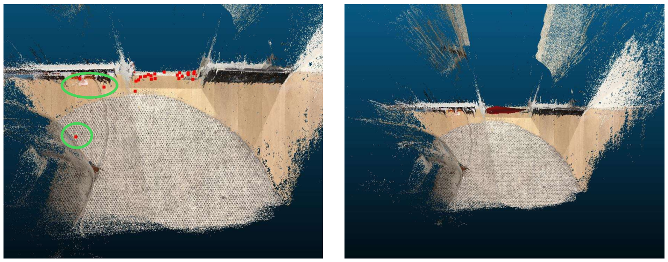
(b) Duplicates Removed

Fig. 18. **Removing Duplicates.** Due to the thresholding of floating point number equality with the reciprocal pair constraints, we may get duplicates for the same diffuse point (a). We remove duplicate pair by choosing the pair with minimum overall errors from the constraints (b).

6 CONCLUSION

The handheld smartphone setup poses challenges due to its use of a coarse emitter and corresponding receiver for real-time operation. We introduce the 'reciprocal pair' method to address the association problem, allowing us to distinguish between one-, two-, or three-bounce returns at these coarse scanned spots where the transmitter (Tx) and receiver (Rx) are co-located. Our results demonstrate the capability to reconstruct true specular surfaces in 3D and effectively cull reflections in the scene.

We show our approach effectively enhances the robustness of state-of-the-art learning-based methods in ambiguous scenarios, including segmentation and novel view synthesis applications. This not only paves the way for future applications but also highlights the simplicity and effectiveness of our methods for integration with various algorithms. We anticipate that leveraging multi-bounce returns will unlock new possibilities in fields and applications that were previously unattainable.



(a) False-positives Points **(b) Filtered Mesh Reconstruction**

(c) Nearby objects inter-reflections **(d) Nearby objects inter-reflections**

Fig. 19. **False Positive Detection.** (a) Potential false-positives (green circles) can occur due to sensor noise or inter-reflections from nearby objects in the scene. (b) Mesh reconstruction and morphological filtering can remove spurious detections over time. (c) and (d) shows the inter-reflection from the sofa. One-bounce is shown as purple point, and mirror point is shown as red triangles.

REFERENCES

[1] 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.

[2] S. Roth and M. J. Black, "Specular flow and the recovery of surface structure," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1869–1876.

[3] Y. Adato, Y. Vasilyev, T. Zickler, and O. Ben-Shahar, "Shape from specular flow," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 11, pp. 2054–2070, 2010.

[4] A. C. Sankaranarayanan, A. Veeraraghavan, O. Tuzel, and A. Agrawal, "Specular surface reconstruction from sparse reflection correspondences," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 1245–1252.

[5] A. Zisserman, P. Giblin, and A. Blake, "The information available to a moving observer from specularities," *Image and vision computing*, vol. 7, no. 1, pp. 38–42, 1989.

[6] T. Chen, M. Goesele, and H.-P. Seidel, "Mesostructure from specularity," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1825–1832.

[7] K. Ikeuchi, "Determining surface orientations of specular surfaces by using the photometric stereo method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 661–669, 1981.

[8] Bonfort and Sturm, "Voxel carving for specular surfaces," in *Proceedings Ninth IEEE International Conference on Computer Vision*. IEEE, 2003, pp. 591–596.

[9] S. Savarese, M. Chen, and P. Perona, "Local shape from mirror reflections," *International Journal of Computer Vision*, vol. 64, pp. 31–67, 2005.

[10] T. Whelan, M. Goesele, S. J. Lovegrove, J. Straub, S. Green, R. Szeliski, S. Butterfield, S. Verma, R. A. Newcombe, M. Goesele *et al.*, "Reconstructing scenes with mirror and glass surfaces." *ACM Trans. Graph.*, vol. 37, no. 4, p. 102, 2018.

[11] M. Tarini, H. P. Lensch, M. Goesele, and H.-P. Seidel, "3d acquisition of mirroring objects using striped patterns," *Graphical Models*, vol. 67, no. 4, pp. 233–259, 2005.

[12] Y. Francken, T. Cuypers, T. Mertens, J. Gielis, and P. Bekaert, "High quality mesostructure acquisition using specularities," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–7.

[13] D. Nehab, T. Weyrich, and S. Rusinkiewicz, "Dense 3d reconstruction from specularity consistency," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[14] M. Yamazaki, S. Iwata, and G. Xu, "Dense 3d reconstruction of specular and transparent objects using stereo cameras and phase-shift method," in *Computer Vision–ACCV 2007: 8th Asian Conference on Computer Vision, Tokyo, Japan, November 18-22, 2007, Proceedings, Part II 8*. Springer, 2007, pp. 570–579.

[15] J. Balzer, "Shape from specular reflection in calibrated environments and the integration of spatial normal fields," in *2011 18th IEEE International Conference on Image Processing*. IEEE, 2011, pp. 21–24.

[16] M. Weinmann, A. Osep, R. Ruiters, and R. Klein, "Multi-view normal field integration for 3d reconstruction of mirroring objects," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 2504–2511.

[17] X. Yang, H. Mei, K. Xu, X. Wei, B. Yin, and R. W. Lau, "Where is my mirror?" in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[18] T. Huang, B. Dong, J. Lin, X. Liu, R. W. Lau, and W. Zuo, "Symmetry-aware transformer-based mirror detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 1, 2023, pp. 935–943.

[19] Z.-X. Yin, J. Qiu, M.-M. Cheng, and B. Ren, "Multi-space neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 12 407–12 416.

[20] J. Zeng, C. Bao, R. Chen, Z. Dong, G. Zhang, H. Bao, and Z. Cui, "Mirror-nerf: Learning neural radiance fields for mirrors with whitted-style ray tracing," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 4606–4615.

[21] Y.-C. Guo, D. Kang, L. Bao, Y. He, and S.-H. Zhang, "Nerfren: Neural radiance fields with reflections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 409–18 418.

[22] H. Mei, B. Dong, W. Dong, P. Peers, X. Yang, Q. Zhang, and X. Wei, "Depth-aware mirror segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3044–3053.

[23] P. Foster, Z. Sun, J. J. Park, and B. Kuipers, "Visagge: Visible angle grid for glass environments," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 2213–2220.

[24] H. Tibebu, J. Roche, V. De Silva, and A. Kondoz, "Lidar-based glass detection for improved occupancy grid mapping," *Sensors*, vol. 21, no. 7, p. 2263, 2021.

[25] S.-W. Yang and C.-C. Wang, "On solving mirror reflection in lidar sensing," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 2, pp. 255–265, 2010.

[26] R. Ramesh and J. Davis, "5d time-light transport matrix: What can we reason about scene properties?" Tech. Rep., 2008.

[27] K. Kutulakos and E. Steger, "A theory of refractive and specular 3d shape by light-path triangulation," in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 2, 2005, pp. 1448–1455 Vol. 2.

[28] M. O'Toole, F. Heide, L. Xiao, M. B. Hullin, W. Heidrich, and K. N. Kutulakos, "Temporal frequency probing for 5d transient analysis of global light transport," *ACM Trans. Graph.*, vol. 33, no. 4, jul 2014.

[29] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[30] P. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[31] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981. [Online]. Available: /brokenurl#http://publication.wilsonwong.me/load.php?id=233282275

[32] X. Yang, H. Mei, K. Xu, X. Wei, B. Yin, and R. W. Lau, "Where is my mirror?" in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8809–8818.

[33] T.-H. Lin and C.-C. Wang, "Deep learning of spatio-temporal features with geometric-based moving point detection for motion segmentation," in *2014 IEEE International Conference on Robotics and Automation*. IEEE, 2014, pp. 3058–3065.

7 SUPPLEMENTARY

Here we derive the analytical expression for $\mathbf{x}_s$ that is used in our experimental results. For a more precise measurement of the one-bounce return in eq. (1), we introduce $\mathbf{x}_l$ as a point on the LiDAR transmitter, that has a baseline b from the receiver $\mathbf{x}_c$. Based on Fig. 4, the two-bounce path of pixel B is $\mathbf{x}_l \to \mathbf{x}_d \to \mathbf{x}_s \to \mathbf{x}_c^B$, and the one-bounce path of pixel A is $\mathbf{x}_l \to \mathbf{x_d} \to \mathbf{x}_c^A$. Here we compute the range r_{dc}, which is the range from point $\mathbf{x}_d$ to $\mathbf{x}_c$. Using the spherical coordinate, let θ be the polar angle and ϕ be the azimuthal angle, and with a coordinate frame where X, Y, and Z axes point left, up, and forward with respect to the LiDAR, r_{dc} is computed as:

$$r_{dc} = \frac{1}{2}\frac{c^2 t_1^2 - b^2}{ct_1 - b\sin(\theta_{dc})\sin(\phi_{dc})}, \tag{5}$$

where c is the speed of light, and the ToF of the one-bounce return is $t_1 = \frac{1}{c}(r_{dl} + r_{dc})$.

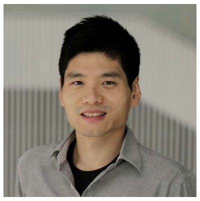
Analytical expression of eq. (2): With the ToF of two-bounce $t_2 = \frac{1}{c}(r_{dl} + r_{ds} + r_{sc})$, and the relationship $r_{ds} = c(t_2 - t_1) + r_{dc} - r_{sc}$, we can substitute the expression using law of cosine for the triangle with $\mathbf{x}_d, \mathbf{x}_c, \mathbf{x}_s$ endpoints, and compute r_{sc} as:

$$r_{sc} = \frac{c}{2}\frac{\Delta t_{12}[\Delta t_{12} + \frac{2r_{dc}}{c}]}{\Delta t_{12} + (1 - \cos\delta)\frac{r_{dc}}{c}}, \tag{6}$$

where $\Delta t_{12} = t_2 - t_1$, the time difference between two-bounce and one-bounce returns, and δ is the angle between the direction of arriving echoes at pixel A and B, such that $\cos(\delta) = AB/|A||B|$.

Finally, the 3D coordinate X, Y, and Z of the point $\mathbf{x}_s$ on the specular surface is:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = r_{sc} \begin{bmatrix} cos(\theta) \\ sin(\theta)sin(\phi) \\ sin(\theta)cos(\phi) \end{bmatrix}, \tag{7}$$



Tsung-Han Lin is currently a S.M. student in the Camera Culture group at MIT Media Lab. He received his B.S. degree from University of California, San Diego in 2010, and M.S. from National Taiwan University in 2013, both in Computer Science. Afterwards, he spent a few years in autonomous driving industry, working at Uber ATG and various startups. His past work spanned topics in robotics, computer vision, and motion prediction. His current research interests include time-of-flight and non-line-of-sight imaging.

Connor Henley recently received his PhD from the Massachusetts Institute of Technology, where he was a member of the Camera Culture group in the MIT Media Lab. His research focuses on developing new imaging and remote sensing methods that rely on the information captured by lidar sensors and time-of-flight cameras. Prior to attending MIT, Connor was an assistant technical staff member at MIT Lincoln Laboratory. He received his BS in Physics and Mathematics from The Johns Hopkins University.

Siddharth Somasundaram is a PhD student in the Camera Culture group at MIT Media Lab. Previously, he completed his M.S. in 2024 and worked as research staff from 2021 to 2022 there. He received his B.S. degree in electrical engineering from University of California, Los Angeles in 2021. His past work spanned topics in photonics, optics, and computational imaging. His current research interests are in computational imaging, computer vision, time-of-flight imaging, and single-photon sensing. He is a recipient of the NSF Graduate Research Fellowship Program (2024).

Akshat Dave is a Postdoctoral Associate in the Camera Culture group at MIT Media Lab. He received his Bachelors and Masters degree in Electrical Engineering from Indian Institute of Technology Madras in 2017 and Ph.D. in Electrical and Computer Engineering from Rice University in 2023. His research interests include time-of-fight, polarization, computational imaging, inverse rendering, and neural rendering. He is a recipient of the Ralph Budd Thesis Award (2024), Lodieska Stockbridge Vaughn Fellowship (2023), COSI Best Paper Award (2020) and Texas Instruments Fellowship (2017).

Moshe Laifenfeld received his BSc, MSc, and PhD degrees from the Technion, Tel-Aviv University, and Boston University, in 1992, 1998, and 2008, respectively, all in electrical and computer engineering. After a joint post-doctoral position with MIT and Boston University, he joined General Motors RD, focusing on in-vehicle wireless communications. More recently he lead the research and development of dToF LiDAR sensors, and ML based point cloud - RGB fusion at Apple. In his past, he led the algorithms development of a 3rd generation UMTS transceiver, and held several RD positions in medical devices startups.

Ramesh Raskar is an Associate Professor at MIT Media Lab where he directs the Camera Culture research group. His focus is on AI and Imaging for health and sustainability. They span research in physical (e.g., sensors, health-tech), digital (e.g., automated and privacy-aware machine learning) and global (e.g., geomaps, autonomous mobility) domains. He received the Lemelson Award (2016), ACM SIGGRAPH Achievement Award (2017), DARPA Young Faculty Award (2009), Alfred P. Sloan Research Fellowship (2009), TR100 Award from MIT Technology Review (2004) and Global Indus Technovator Award (2003). He has worked on special research projects at Google [X] and Facebook and co-founded/advised several companies.