

# Seeing the Unseen: Locating Objects from Reflections

Jing Wu<sup>[0000–0001–5123–9861]</sup> and Ze Ji<sup>[0000–0002–8968–9902]</sup>

Cardiff University, Cardiff CF24 3AA, UK,  
{wuj11, jiz1}@cardiff.ac.uk

**Abstract.** Inspired by the ubiquitous use of reflections in human vision system, in this paper, we present our first step exploration of using reflections to extend the FOV of cameras in computer vision applications. We make use of a stereo camera, and establish mathematical models for locating objects from their mirror reflections. We also propose a pipeline to track and locate moving objects from their reflections captured in videos. Experimental results demonstrated the efficiency and effectiveness of the proposed method, and verified the potential use of reflections in locating non-line of sight (NLOS) objects.

**Keywords:** Mirror reflection, object tracking, stereo vision

## 1 Introduction

Reflections are ubiquitous in our environments, and are utilized by our vision system to make us aware of our surroundings. A typical use of reflections is to extend the field of view (FOV) of our eyes to make scenes and objects, which are not directly in line of sight (NLOS), visible. For example, we use rear view mirrors of cars to tell the traffic behind, and detect oncoming vehicles at junctions by their reflections from parked cars or windows. There is little effort in the area of robotics using reflections as another means of robot perception. Inspired by such empirical use of reflections in human vision, in this paper, we present our first step exploration of using reflections to extend the FOV of cameras for not only general-purpose computer vision applications, but also specifically for applications with vision-based robot perception, such as vSLAM (Visual Simultaneous Localization and Mapping) and situation awareness.

There have been works in computer vision that try to extract useful information from reflections. Yang et.al. [18] regarded a water reflection image as a special case of two-view stereo and recovered the depth of the scene using stereo matching. Depth information can also be recovered from the replica of reflections presented on transparent surfaces [19]. For a transparent surface, the reflected scenes are often mixed up with the scenes transmitted from the other side. Then it becomes a popular topic how to separate and restore the two scenes [3,4,14,13,15,5]. Reflections in eyes have also attracted research interests. Nishino et.al. [10,9] estimated lighting from corneal reflections, which were then used for relighting applications [10] and robust face recognition [9]. Jenkins and Kerr [7] extracted faces of bystanders from corneal reflections, and proposed its possible application in criminal investigations. There are other interesting works making use of reflections, such as Sparkle Vision [20] – a work from MIT which aims to

find images hidden in reflections from random shining micro-surfaces, and the use of inconsistent reflections to expose manipulation of photos [11].

Existing works demonstrated the diverse use of reflections in computer vision applications. Reflections provide a view of the scene from a different angle, which on one hand can reveal depth information when combined with the direct view; on the other hand can extend the FOV to expose NLOS scenes and objects. Our work focuses on the latter, and differs from relevant existing works [7] in that 1) instead of restoration of static scenes, we emphasize on tracking moving NLOS objects from reflections captured in videos; and 2) we make use of a stereo camera to help locating the object.

There have been works using ultra-fast time-of-flight imaging to detect motion of NLOS objects [12] and recover their 3D shapes [17]. However this imaging relies on the use of a femtosecond laser and a streak camera which are expensive (hundreds of thousands dollars), and slow (one hour or so capturing time) [6]. Our work overcomes these limitations by using visual reflections coupled with real-time processing algorithms.

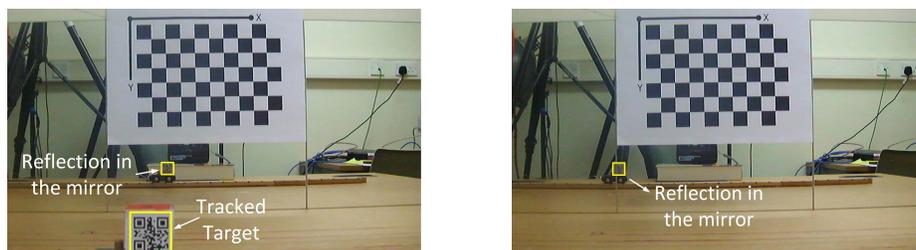
The contributions of this paper are two folds. Firstly, this paper presents the first exploration into the novel use of visual reflections in estimating movements of NLOS objects. Secondly, in a simplified lab setting, we 1) establish the mathematical model that relates the locations of the object, the reflective surface, and the observer, and 2) based on the model, propose a processing pipeline for tracking and locating the moving NLOS object.

## 2 Problem Overview

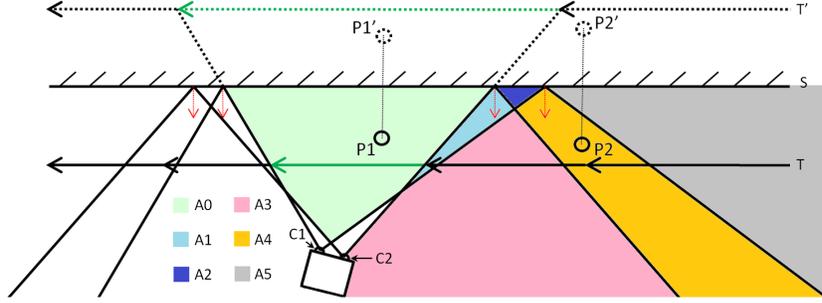
Fig. 1 illustrates the idea of using reflections to reveal NLOS objects. While the target object is already out of FOV in Fig. 1 right, its reflection from the mirror is still visible to the camera. Our idea is to infer the 3D location of the object from its reflection. Although reflection is a complex phenomenon in real world, in this initial exploration, we make a few simplifications in our laboratory set-up, which include:

- a planar mirror reflector to simplify the mathematical model and liberate the attention from image distortion and blurring;
- a binary QR code on the object to facilitate object detection and tracking.

Fig. 2 draws the simplified scene from top view.  $S$  is the planar reflector.  $C1$  and  $C2$  are the two lenses of the stereo camera. The use of reflections from  $S$  extends the FOV



**Fig. 1.** Example frames of the moving object and its reflection, and their tracking results



**Fig. 2.** Illustration of the simplified scenario and the reflections.

of  $C1$  and  $C2$ . Without loss of generality, we analyse the field on the right. Look at  $C1$  for example, objects in regions  $A0 - 2$ , e.g.  $P1$ , are directly visible to  $C1$ . Objects in regions  $A3$  and  $A4$ , e.g.  $P2$ , are invisible to  $C1$  themselves, but their reflections from  $S$  are visible to  $C1$ . The FOV of  $C1$  is thus extended to  $A3$  and  $A4$ . Similarly, the FOV of  $C2$  is extended to  $A1$  and  $A3$ . Based on the visible contents (the object  $P$  or its reflection  $P'$ ), the field on the right can be segmented into six regions as colour coded in Fig. 2. Table 1 lists the visible contents for each region.

**Table 1.** Visibility of the object  $P$  and its reflection  $P'$  to camera  $C1$  and  $C2$ .

|      | $A0$    | $A1$    | $A2$    | $A3$ | $A4$ | $A5$ |
|------|---------|---------|---------|------|------|------|
| $C1$ | $P, P'$ | $P, P'$ | $P, P'$ | $P'$ | $P'$ |      |
| $C2$ | $P, P'$ | $P'$    |         | $P'$ |      |      |

When an object moving along a trajectory  $T$ , its reflection moves along the mirrored trajectory  $T'$ . As shown in Fig. 2, the visible section on  $T'$  has extended range than that on  $T$ . The visible sections of  $T$  and  $T'$  can be recovered using stereo vision. And the extended part on  $T'$  can be used to infer the locations on  $T$  even they are not directly visible. The method is described in the following section.

### 3 Method

Fig. 3 illustrates the pipeline of our method to recover an object's moving trajectory. The videos are captured in three sections. The calibration section captures a set of checkerboard patterns and is used to calibrate the stereo camera and estimate the location and orientation of the mirror. The movement section records the moving object, from which we detect/track the target object in 2D and reconstruct their 3D locations. The parameters estimated from the calibration section are used in the 3D reconstruction. The sequence of reconstructed 3D locations forms the recovered trajectory which is then compared with the ground-truth obtained from the evaluation section. In the fol-

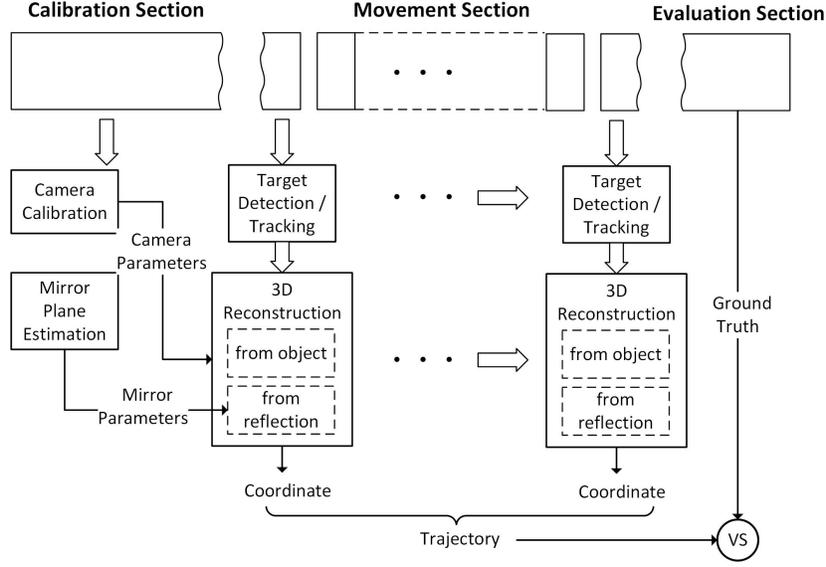


Fig. 3. Processing pipeline.

lowing, we describe in detail the two main functional modules, i.e., target detection and tracking, and 3D reconstruction.

### 3.1 Detection and Tracking

The detection and tracking step is to find the 2D pixel locations of both the actual object and its mirror reflection in the video frames. The experiment considers only one known object. A template matching method is used for the initial detection of the object in the scene. Since the real object and its reflection are not always observable to the camera (Fig. 2), template matching will continue running with every new frame until the first detection of the target.

A registered template  $t$  is first pre-determined for the search of the initial position of the object in the frames. The template matching is based on the fast normalized cross-correlation [8] that can be formulated as:

$$r(u, v) = \frac{\sum_{x,y} [f(x, y) - \bar{f}_{u,v}] \cdot [t(x - u, y - v) - \bar{t}]}{\sqrt{\sum_{x,y} [f(x, y) - \bar{f}_{u,v}]^2 \sum_{x,y} [t(x - u, y - v) - \bar{t}]^2}} \quad (1)$$

where  $f(x, y)$  is the intensity value at  $(x, y)$ ,  $\bar{f}_{u,v}$  is the mean of the region shifted to position  $[u, v]$  being compared with the template  $t$ , and  $\bar{t}$  is the mean of the template.

With the first object detected, image tracking will be activated using the initial detected image patch as the tracking target. Due to various other uncontrollable factors, such as deformation caused by uneven reflective surfaces, blurred images due to poor

focusing, occlusions, as well as rotations and distances from the camera, a robust image tracker is therefore needed. Correlation filter based trackers are considered highly efficient and robust in tracking through the correlation operations between the frames and the correlation filter, which is trained and updated at each subsequent time step [2].

Considering robustness and efficiency, we use the MOSSE (Minimum Output Sum of Squared Error) method [2]. Below is a summary of MOSSE. A filter  $h$  is needed to map the input  $x$  with a desired output  $y$  by minimizing the sum of squared error between actual correlation outputs and desired correlation outputs. The correlation operations are performed in the DFT (Discrete Fourier Transform) space with high efficiency compared to the exhausted element-wise convolution computation. This can be formulated as below.

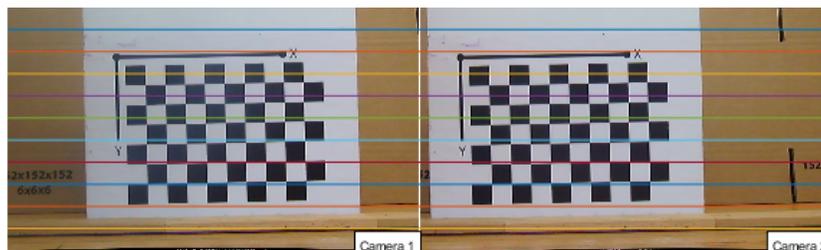
$$\min_{\hat{h}^*} \sum \|\hat{h}^* \odot \hat{x} - \hat{y}\|^2 \quad (2)$$

Fig. 1 shows the tracking results, where both the reflection of the object and the actual object are tracked and highlighted with bounding boxes. As can be seen, to increase the robustness of tracking, we used a binary QR code to enhance the discrimination of the target from the background.

### 3.2 3D Reconstruction

This step is to reconstruct the 3D location (i.e., the world coordinate) of the tracked object at each frame. Camera calibration and mirror plane estimation are carried out beforehand to provide the required parameters in the reconstruction.

**Camera calibration** is to find the camera's intrinsic and extrinsic parameters, and to establish the projection from world coordinates to pixel locations. In this work, we define the origin of the world coordinate system to be at the optical centre of  $C1$ , and the  $x$ ,  $y$ ,  $z$ -axis pointing to the right, down, and away from the camera respectively. We make use of the stereo calibration app in Matlab [1] to calibrate  $C1$  and  $C2$ , and rectify the images they captured. Fig. 4 shows a pair of rectified left/right images after the stereo calibration.



**Fig. 4.** A pair of rectified images after stereo camera calibration. The lines show that the rectified images become undistorted and row-aligned.

**Stereo Vision** is a standard method to locate an object in the 3D world when the object is visible to both cameras. Suppose  $(x_{p1}, y_{p1})$  and  $(x_{p2}, y_{p2})$  are a pair of cor-

responding pixel locations of an object in the rectified left/right images. The disparity between the two pixels  $d = x_{p1} - x_{p2}$  is inversely proportional to the depth of the object, and the world coordinate of the object  $(X_P, Y_P, Z_P)$  can be computed using triangulation, i.e.,

$$Z_P = bf/d, \quad X_P = Z_P(x_{p1} - x_{01})/f, \quad Y_P = Z_P(y_{p1} - y_{01})/f \quad (3)$$

where  $f$ ,  $b$ , and  $(x_{01}, y_{01})$  are the focal length, the baseline, and the pixel location of the camera's optical centre, and are obtained from calibration.

**Stereo vision via reflection** is an extension of standard stereo vision, and can locate an object when its reflected image is visible to both cameras. An example is when the object is in regions  $A0$ ,  $A1$ , and  $A3$  in Fig 2. Assuming a planar reflecting surface, the reconstruction via reflection can be carried out in two steps as shown in Fig. 5. Since the virtual object  $P'$  can be seen as a fixed point behind the reflector, we can first recover its world coordinate  $\mathbf{X}_{P'} = (X_{P'}, Y_{P'}, Z_{P'})$  using Eq.(3). Then the coordinate of the actual object  $P$ :  $\mathbf{X}_P = (X_P, Y_P, Z_P)$  is the mirror inversion to  $\mathbf{X}_{P'}$ , i.e.,

$$\mathbf{X}_P = \mathbf{X}_{P'} + 2 \cdot dist \cdot \mathbf{n}_S, \quad (4)$$

where  $\mathbf{n}_S = (n_x, n_y, n_z)$  is the unit normal vector of the planar reflective surface,  $dist = |\mathbf{n}_S \cdot \mathbf{X}_{P'} + h|$  is the distance from  $P'$  to the plane, and  $h$  the distance from the origin of the world coordinate system to the plane.

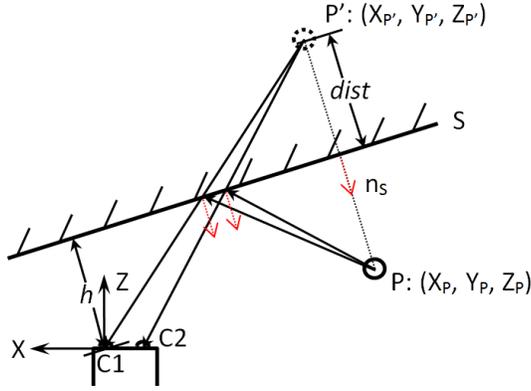
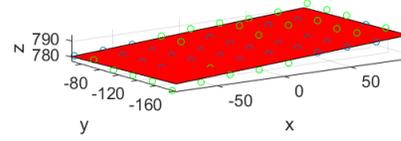
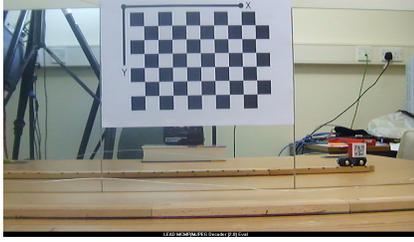


Fig. 5. Stereo vision via reflection.  $P'$  is visible to both  $C1$  and  $C2$ .

**Locating the mirror plane** is important in locating an object from its mirror reflection, as the position of the virtual object depends on the position and orientation of the reflector. In this work, we simplify the mirror plane estimation by using a checkerboard pattern attached on it, as shown in Fig. 6 left. The world coordinates of the checkerboard points are recovered using stereo vision (Eq.(3)), and the position  $h$  and orientation  $\mathbf{n}_S$  of the mirror plane are found using the M-estimator Sample Consensus (MSAC) algorithm [16] on the detected points. Fig. 6 right shows a recovered mirror plane.



**Fig. 6.** Locating the mirror plane from checkerboard points.

The orientation of the mirror plane is determined by the normal's elevation angle  $\theta$  and azimuth angle  $\phi$ . Representing the normal using spherical coordinate  $\mathbf{n}_S = (\cos \theta \cos \phi, \cos \theta \sin \phi, \sin \theta)$ , from Eq.(4), the world coordinate of the real object  $X_P$  can be expressed as a function

$$X_P = f(X_{P'}, \theta, \phi, h) \quad (5)$$

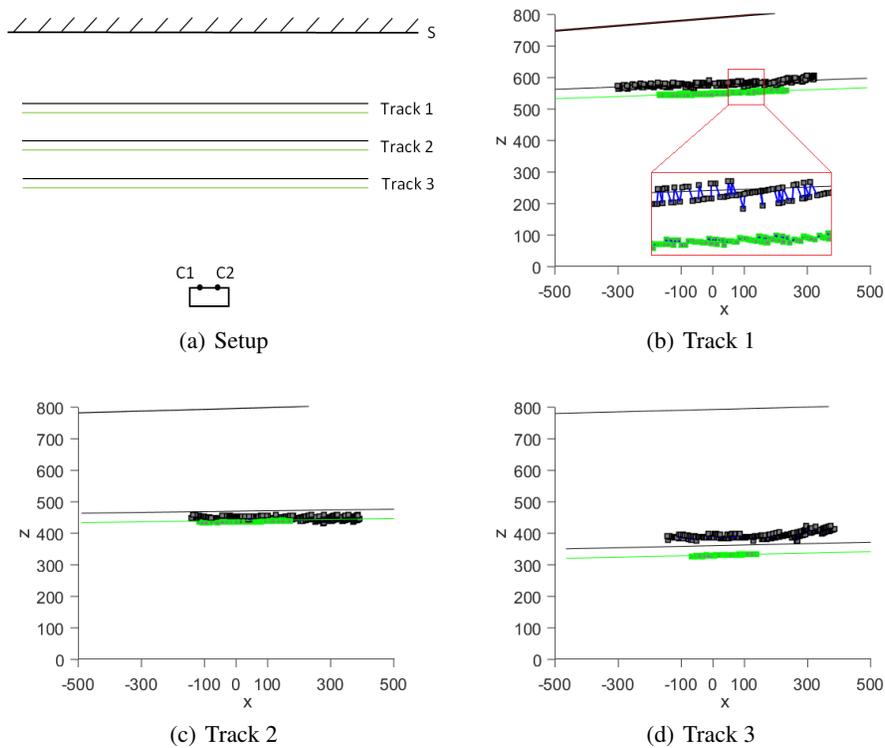
Analysing the partial derivatives of  $f$  with respect to the mirror parameters gives important information how the estimation of the mirror affect locating the object via its reflection. In experiments, we carry out numerical analysis of this influence.

## 4 Results and Analysis

To compare the trajectories recovered from the actual object and its reflection, we move the object along tracks at different distances toward the mirror. Fig. 7 (a) shows the setup of the experiments. Because the object has a thickness, the images of the object and its reflection show two opposite sides of the object, and the moving trajectories for the two sides are separated. As a result, for each track, two ground-truth trajectories are used. Fig. 7 (b-d) visualize the recovered trajectories for each of the three tracks. Each recovered trajectory consists of a sequence of 3D locations reconstructed frame-wise from the stereo video. The sequences in black are recovered using images of the object's reflection while the green ones are recovered from images of the actual object. Both types of trajectories are compared with the according ground-truth shown as a straight line in the same colour.

From Fig. 7, we observed that both the recovered trajectories are generally consistent with the ground-truth. The trajectories from reflections extend the according trajectories from objects, demonstrating the benefit of using reflections to extend the FOV of cameras. Then at a closer look, we noticed that the trajectories from reflections deviate slightly from their ground-truth, especially in Fig. 7 (c) and (d). Moreover, amplifying the recovered trajectories (as shown in the red box in Fig. 7 (b)), we noticed bigger variance within the trajectory from reflections than that within the trajectory from objects.

To quantitatively analyse the results, for each recovered trajectory, we measure its length in  $x$  direction, its reconstruction error, and its variability. The reconstruction error is calculated as the average distance from the recovered locations to the corresponding



**Fig. 7.** Shown in  $xz$  plane, the recovered trajectories (sequences of points) compared with ground-truth (straight lines). Black: trajectories from images of reflection; Green: trajectories from images of object. The top line in each sub-figure shows the recovered mirror plane.

ground-truth line. To compute the variability, we first use SVD (singular value decomposition) to find the best fit line to the recovered sequence of locations. The variability of the trajectory is calculated as the standard deviation from the locations to the line. The measurements for each recovered trajectory are shown in Table 2.

From the table, the lengths of trajectories verified the extended FOV of cameras using reflections. In theory, the extension becomes more significant when object is closer to the camera. The ratio of the length from reflection to that from object shows the trend (although not accurate, as we didn't track the reflections for the whole visible length). We also found that when the object moves away from the mirror toward the camera, the recovered trajectory from reflection has increased reconstruction error and variability, but the recovered trajectory from object maintains and even decreases the reconstruction error with decreased variability. There are several reasons behind this observation. Firstly, it is known that the accuracy of stereo vision in the near range far exceeds that in the far range. From Track 1 to 3, while the object gets closer to the camera, its reflection gets further away, which explains the trends in the accuracy changes. Secondly, a

**Table 2.** Quantitative analysis of recovered trajectories

| Trajectory from           | Track 1 |            | Track 2 |            | Track 3 |            |
|---------------------------|---------|------------|---------|------------|---------|------------|
|                           | object  | reflection | object  | reflection | object  | reflection |
| Length (mm)               | 402.2   | 622.1      | 288.1   | 533.4      | 205.3   | 530.2      |
| Reconstruction error (mm) | 3.7     | 6.4        | 4.5     | 24.0       | 3.2     | 30.7       |
| Standard deviation (mm)   | 1.3     | 4.6        | 0.8     | 5.2        | 0.5     | 7.9        |

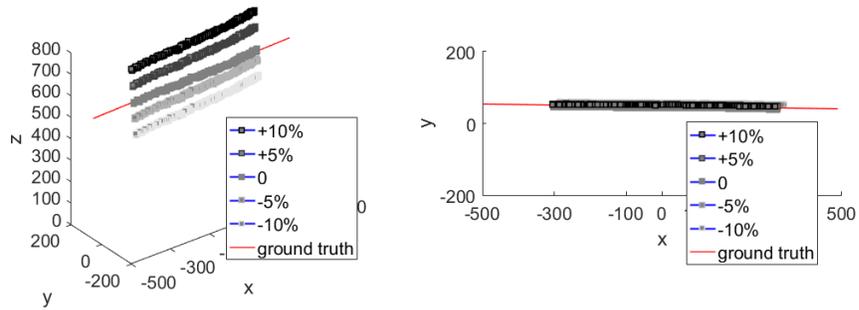
faraway reflection increases errors in tracking, which further adversely affects the accuracy of reconstruction. Finally, reconstruction from reflection inherently depends on the location and orientation of the reflective surface. Deviations of mirror estimation will cause further inaccuracies in the recovered trajectory from reflection.

To better understand the influence from estimation of mirror parameters, we use the recovered trajectory for Track 1 as an example and examine the changes of the trajectory from reflection by varying each of the mirror parameters. Fig. 8 shows the changes by varying (a) the distance of the origin to the mirror  $h$ , (b) the elevation angle of the surface normal  $\theta$ , and (c) the azimuth angle of the surface normal  $\phi$ . From the two views in Fig. 8 (a), we can see that estimation errors in  $h$  mainly influence the depth of the trajectory, and have limited influence to  $x, y$  locations. A positive deviation in  $h$  results in an overestimation of the depth while a negative deviation results in an underestimation. Turning our attention to Fig. 8 (b), estimation errors in  $\theta$  have influence to locations in all three directions. Deviating  $\theta$  from  $+30^\circ$  to  $-30^\circ$  moves the trajectory along a sheared valley shaped surface. Compared with  $h$  and  $\theta$ , the influence from errors in  $\phi$  is relatively small. We log scaled  $y$ -axis in the second view in Fig. 8 (c) for a better visualization, and noticed a slight movement of trajectory in  $y$  direction when deviating  $\phi$  from  $+30^\circ$  to  $-30^\circ$ . This numerical analysis gives a rough idea of the influence from estimation of mirror parameters. A more thorough analysis can be done analytically by deriving the Jacobian matrix of Eq.(5), and will be covered in our future work.

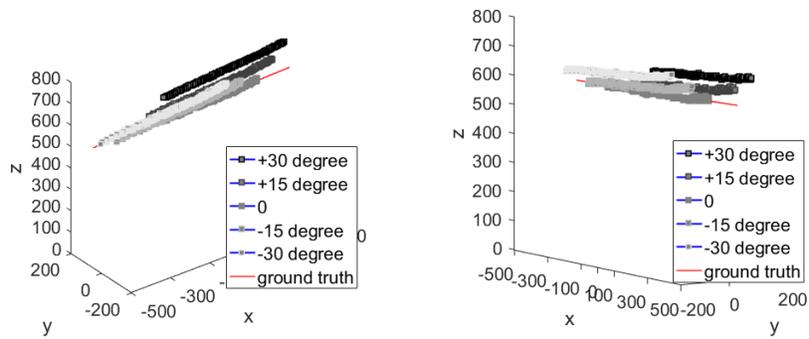
Fig. 9 shows the running time of the proposed method in tracking and locating the moving object on Track 1 from its reflection. The program is run on a PC with Intel Core i7-7500U CPU at 2.70GHz and 8GB memory. From the figure, we notice that tracking in the first frame takes the longest time of 0.035 seconds because of the detection in the whole image, and then drops to below 0.001 seconds, while the reconstruction step is relatively stable around 0.008 seconds. Tracking and reconstruction together takes an average of 0.009 seconds for each frame (110 fps), which demonstrates the real-time processing ability of the proposed method.

## 5 Conclusion and Discussion

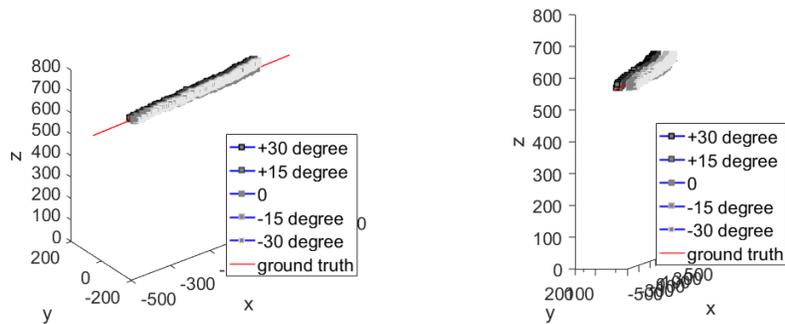
In this paper, we proposed the idea of making use of reflections to extend the FOV of cameras, and carried out preliminary work verifying our idea in a simplified lab setting. We established the mathematical model in the simplified scene, and proposed a processing pipeline to recover an object’s moving trajectory reaching out of the camera’s FOV. We carried out experiments to analyse the accuracies of recovered trajectories, and how the recovery is influenced by the estimation of the reflective surface. The results verified



(a) Varying  $h$



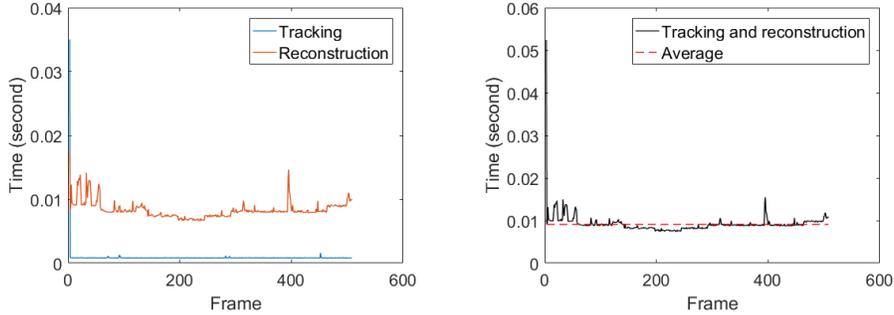
(b) Varying  $\theta$



(c) Varying  $\phi$

**Fig. 8.** Influence from estimation errors of mirror parameters  $h$ ,  $\theta$ , and  $\phi$ . For each parameter, two views are provided to give better illustration.

the potential use of reflections in locating NLOS objects, and inspired more ideas for future work which will be discussed below.



**Fig. 9.** Time efficiency of tracking and reconstruction.

As mentioned in section 4, the trajectory from reflection depends on the mirror parameters. Moreover the recovery of both trajectories from reflection and object depends on the camera’s intrinsic/extrinsic parameters. In order to have a thorough understanding of how the recovered trajectory is affected by all these factors, we will carry out analytical error analysis by deriving the Jacobian matrix of Eq.(5), and this will be our immediate work in the future.

A robust and accurate tracking algorithm is required to handle multiple objects and reflections, occlusions, blurry images, and deformation. We will first deploy other advanced image trackers, including the popular kernel-based KCF method [2] and its derivatives. Also, we will also implement the re-detection function to handle situations when the target is lost due to occlusions or ambiguous features. One long term challenge is automatic association of the object and its corresponding reflection, hence, allowing tracking the object by not only the registered patterns, but also fusing the reconstructed trajectories from both the reflection and object.

Ground-truth trajectories are necessary to evaluate the proposed method. However, it is difficult to extract accurate ground-truth in the current primitive lab setting with limited equipments. Currently, we obtain the ground-truth trajectories by combining detection of checkerboard points and manual measurements, which are inaccurate and affect the evaluation of reconstruction accuracy. In the future, it is desired to develop approaches to achieve more accurate ground-truth data, possibly from using more advanced equipments or using computer simulations. In the long term, creating such a benchmark dataset will benefit future research into the use of reflections as well.

An interesting direction to carry forward is to extend the mathematical model and evaluation to non-planar reflective surfaces which are more common in real world. The non-planarity is sometimes beneficial in revealing NLOS objects. An example is the convex rear view mirror on cars which gives much wider FOV than a planar mirror. Stereo vision via reflection from such non-planar surfaces is much more complex, but is more practical in real world applications, and thus will be an interesting direction to explore in our future work.

In addition to non-planar reflective surfaces, there are many other challenges in application of the proposed method to real world scenarios. For example, many surfaces

are not perfectly specular, from which reflections can be dim and blurry, and makes it difficult for detection and tracking. Another fundamental question is how to differentiate images of reflections from images of objects. And the time efficiency is crucial in many applications. However, we believe the use of reflections has the potential to improve many computer vision tasks, and hope our preliminary work would inspire more interest into this challenging field.

## References

1. Matlab stereo calibration app, <https://uk.mathworks.com/help/vision/ug/stereo-camera-calibrator-app.html>, online; accessed 13 Feb 2018 **5**
2. Chen, Z., Hong, Z., Tao, D.: An experimental survey on correlation filter-based tracking. *CoRR* **abs/1509.05520** (2015) **5, 11**
3. Diamant, Y., Schechner, Y.Y.: Overcoming visual reverberations. In: Proc. CVPR (2008) **1**
4. Farid, H., Adelson, E.: Separating reflections from images by use of independent component analysis. *Journal of the Optical Society of America A* **16**(9), 2136–2145 (1999) **1**
5. Guo, X., Cao, X., Ma, Y.: Robust separation of reflection from multiple images. In: Proc. CVPR. pp. 2195–2202 (2014) **1**
6. Heide, F., Hullin, M.B., Gregson, J., Heidrich, W.: Low-budget transient imaging using photonic mixer devices. *ACM Transactions on Graphics (TOG)* **32**(4) (July 2013) **2**
7. Jenkins, R., Kerr, C.: Identifiable images of bystanders extracted from corneal reflections. *PLOS ONE* **8**(12) (December 2013) **1, 2**
8. Lewis, J.P.: Fast normalized cross-correlation (1995), <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.6062> **4**
9. Nishino, K., Belhumeur, P., Nayar, S.: Using eye reflections for face recognition under varying illumination. In: Proc. ICCV (2005) **1**
10. Nishino, K., Nayar, S.K.: Eyes for relighting. *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2004* **23**(3), 704–711 (August 2004) **1**
11. O’Brien, J.F., Farid, H.: Exposing photo manipulation with inconsistent reflections. *ACM Transactions on Graphics (TOG)* **31**(1) (January 2012) **2**
12. Pandharkar, R., Velten, A., Bardagjy, A., Lawson, E.: Estimating motion and size of moving non-line-of-sight objects in cluttered environments. In: Proc. CVPR. pp. 265–272 (2011) **2**
13. Sarel, B., Irani, M.: Separating transparent layers through layer information exchange. In: Proc. ECCV. vol. IV, pp. 328–341 (2004) **1**
14. Schechner, Y.Y., Nayar, S.K.: Generalized mosaicing: Polarization panorama. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* **27**(4), 631–636 (April 2005) **1**
15. Szeliski, R., Avidan, S., Anandan, P.: Layer extraction from multiple images containing reflections and transparency. In: Proc. CVPR (2000) **1**
16. Torr, P.H.S., Zisserman, A.: Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding* **78**(1), 138–156 (2000) **6**
17. Velten, A., Willwacher, T., Gupta, O., Veeraraghavan, A., Bawendi, M.G., Raskar, R.: Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nature Communications* **3** (March 2012) **2**
18. Yang, L., Liu, J., Tang, X.: Depth from water reflection. *IEEE TRANSACTIONS ON IMAGE PROCESSING* **24**(4), 1235–1243 (April 2015) **1**
19. Yano, T., Shimizu, M., Okutomi, M.: Image restoration and disparity estimation from an uncalibrated multi-layered image. In: Proc. CVPR. pp. 247–254 (2010) **1**
20. Zhang, Z., Isola, P., Adelson, E.H.: Sparklevision: Seeing the world through random specular microfacets. In: Proc. CVPR Workshops (2015) **1**