Fixed Viewpoint Mirror Surface Reconstruction under an Uncalibrated Camera
Kai Han, Miaomiao Liu, Dirk Schnieders, and Kwan-Yee K. Wong

Abstract—This paper addresses the problem of mirror surface reconstruction, and proposes a solution based on observing the reflections of a moving reference plane on the mirror surface. Unlike previous approaches which require tedious calibration, our method can recover the camera intrinsics, the poses of the reference plane, as well as the mirror surface from the observed reflections of the reference plane under at least three unknown distinct poses. We first show that the 3D poses of the reference plane can be estimated from the reflection correspondences established between the images and the reference plane. We then form a bunch of 3D lines from the reflection correspondences, and derive an analytical solution to recover the line projection matrix. We transform the line projection matrix to its equivalent camera projection matrix, and propose a cross-ratio based formulation to optimize the camera projection matrix by minimizing reprojection errors. The mirror surface is then reconstructed based on the optimized cross-ratio constraint. Experimental results on both synthetic and real data are presented, which demonstrate the feasibility and accuracy of our method.

Index Terms—Mirror surface, reconstruction, reflection, light path.

I. INTRODUCTION

3D reconstruction of diffuse surfaces has enjoyed tremendous success. These surfaces reflect light from a single incident ray to many rays in all directions, resulting in a constant appearance regardless of the observer’s viewpoint. Methods for diffuse surface reconstruction can therefore rely on the appearance of the object.

This paper considers mirror surfaces, which exhibit specular reflections and whose appearances are a reflection of the surrounding environment. Under (perfect) specular reflection, an incoming ray is reflected to a single outgoing ray. This special characteristic makes the appearance of a mirror surface viewpoint dependent, and renders diffuse surface reconstruction methods useless. Meanwhile, there exist many objects with a mirror surface in the man-made environment. The study of mirror surface reconstruction is therefore an important problem in computer vision.

In this paper, we assume the mirror surface reflects a light ray only once, and tackle the mirror surface reconstruction problem by adopting a common approach of introducing motion to the environment. Unlike previous methods which require a fully calibrated camera and known motion, we propose a novel solution based on observing the reflections of a reference plane undergoing an unknown motion with a stationary uncalibrated camera (see Fig. 1(a)). Note that the checkerboard pattern in Fig. 1(a) is used here only to demonstrate the overall reflection effects of the mirror surface. In our experiment, we display vertical and horizontal sweeping lines (e.g., the bottom row in Fig. 18) to establish reflection correspondences between images and the reference plane.

We first show that the relative poses of the reference plane can be estimated from the reflection correspondences established between the images and the reference plane under three unknown distinct poses. This enables us to construct a 3D ray piercing the reference plane at specific positions under different poses for each image point on the mirror surface. Given the set of 3D rays and their corresponding image points, we derive an analytical solution to recover the camera projection matrix through estimating the line projection matrix, which can be easily transformed back into a corresponding camera projection matrix. To make our solution more robust to noise, we use this closed-form solution as an initialization and optimize the camera projection matrix by minimizing reprojection errors computed based on a cross-ratio formulation for the mirror surface (see Fig. 1(b)). The mirror surface is then reconstructed based on the optimized cross-ratio constraint. The key contributions of this work are

1. To the best of our knowledge, the first mirror surface reconstruction solution under an unknown motion of a reference plane and an uncalibrated camera.
2. A closed-form solution for recovering the relative 3D poses of the reference plane from reflection correspondences.
3. A closed-form solution for estimating the camera pro-
Adato et al. provided a theoretical basis for the later methods. In [7], [8], with a parametric representation (e.g., sphere), their work a mirror surface. Although they only recovered a surface specular flow and derived its relation with the 3D shape of an object. Roth and Black [6] introduced the concept of environment and those assuming a partial differential equations (PDEs). Vasilyev et al. further suggested that it is possible to reconstruct a smooth surface from one specular flow by inducing integrability constraints on the surface normal field. In [9], Canas et al. reparameterized the nonlinear PDEs as linear equations and derived a more manageable solution. Although SFSF achieves a theoretical breakthrough in shape recovery of mirror surfaces, the issues in tracking dense specular flow and solving PDEs still hinder their practical use. In [11], Sankaranarayanan et al. developed an approach that uses sparse specular reflection correspondences instead of specular flow to recover a mirror surface linearly. Their proposed method is more practical than the traditional SFSF methods. Nevertheless, their method requires quite a number of specular reflection correspondences across different views, which are difficult to obtain due to the distorted reflections on the mirror surface.

Under a known nearby environment, a different set of methods for shape recovery of mirror surfaces can be derived. The majority of these methods are based on the smoothness assumption on the mirror surface. Under this assumption, one popular way is to formulate the surface recovery into the problem of solving PDEs. In [12], Savarese and Perona demonstrated that local surface geometry of a mirror surface can be determined by analyzing the local differential properties of the reflections of two calibrated lines. Following the same fashion, Rozenfeld et al. [14] explored the 1D homography relation between the calibrated lines and the reflections using sparse correspondences. Depth and first order local shape are estimated by minimizing a statistically correct measure, and a dense 3D surface is then constructed by performing a constrained interpolation. In [15], Liu et al. proved that a smooth mirror surface can be determined up to a twofold ambiguity from just one reflection view of a calibrated reference plane. Another way to formulate the mirror surface recovery is by employing normal consistency property to refine visual hull and/or integrate normal field. In [16], Bonfort and Sturm introduced a voxel carving method to reconstruct a mirror surface using a normal consistency criterion derived from the reflections of some calibrated reference planes. In order to get a better view for shape recovery, they further proposed that the camera does not need to face the reference plane, and the shape can be well recovered by using a mirror to calibrate the poses of the reference plane [17], [18]. In [19], Nehab et al. formulated the shape recovery as an image matching problem by minimizing a cost function based on normal consistency. In [20], Weinmann et al. employed a turntable setup with multiple cameras and displays, which enables the calculation of the normal field for each reflection view. The 3D surface is then estimated by a robust multi-view normal field integration technique. In [21], Balzer et al. deployed a room-sized cube consisting of six walls that encode/decode specular correspondences based on a phase shift method. The surface is then recovered by integration of normal fields. Tin et al. [22] introduced a two-layer LCD setup, which contains a pair of perpendicular linear polarizers for establishing correspondences between the illumination rays and camera rays. After calibrating the camera and LCDs, the surface can be reconstructed by solving a joint optimization problem. In [23], Lu et al. introduced a setup to generate a polarization field using a commercial LCD with the top polarizer removed and modeled the liquid crystals as polarization rotators. Another approach is to reconstruct the individual light
paths based on the law of reflection. Kutulakos and Steger [24] showed that a point on a mirror surface can be recovered if the positions of two reference points are known in space and reflected to the same image point in a single view, or the positions of two reference points are known and are reflected by the same surface point to two different views.

Note that calibration plays an important role in all the above methods that assume a known nearby environment. Commonly, a reference plane with a known pattern is used as the known environment. In order to produce a good view of its reflections on the specular surface, the reference plane is often placed side-by-side with the camera. This results in the camera not being able to see the reference plane directly, making the calibration of the setup non-trivial. Traditional methods calibrate the poses of the reference plane by introducing an extra reference plane in the field of view of the camera, and an extra camera looking at both reference planes. In [18], Sturm and Bonfort used a planar mirror to allow the camera to see the reference plane through reflection. The pose of the reference plane can be obtained by placing the auxiliary mirror in at least three different positions. Generally, multiple reference plane poses are needed for recovering a large area of the mirror surface. However, the literature becomes comparatively sparse when it comes to automatic pose estimation of the reference plane in mirror surface recovery. Liu et al. [25] proposed an automatic motion estimation method by constraining the motion of the reference plane to a pure translation. Although they can achieve a simple closed-form solution for the motion estimation problem, their method cannot handle general motion. Besides, their method requires calibrating the intrinsics of the camera as well as the initial pose of the reference plane. In fact, most, if not all, of the methods that assume a known nearby environment require the camera(s) to be fully calibrated. In contrast, we neither require the calibration of the reference plane poses, nor require the calibration of the camera, and make no assumption on the smoothness of the mirror surface. Our proposed approach can automatically calibrate the setup as well as reconstruct the mirror surface using the observed reflections of the reference plane.

A cross-ratio constraint has been used to estimate mirror position and camera pose for axial non-central catadioptric systems [26], [27], and produce more point correspondences in the context of 3D reconstruction [28]. In this work, we incorporate cross-ratio constraint in our formulation to simultaneously optimize the camera projection matrix and recover the mirror surface.

III. ACQUISITION SETUP AND PLANE POSE ESTIMATION

A. Acquisition Setup

Figure 2 shows the setup used for mirror surface reconstruction. Consider a pinhole camera centered at C observing the reflections of a moving reference plane on a mirror surface $S$. Let $\mathbf{X}_0$ be a point on the plane at its initial pose, denoted by $P_0$, which is reflected by a point $\mathbf{M}$ on $S$ to a point $\mathbf{m}$ on the image plane $I$. Suppose the reference plane undergoes an unknown rigid body motion, and let $P_1$ and $P_2$ denote the plane at its two new poses. Let $\mathbf{X}_1$ and $\mathbf{X}_2$ be points on $P_1$ and $P_2$, respectively, which are both reflected by $\mathbf{M}$ on $S$ to the same image point $\mathbf{m}$ on $I$. $\mathbf{X}_0$, $\mathbf{X}_1$ and $\mathbf{X}_2$ are referred to as reflection correspondences of the image point $\mathbf{m}$. Since reflection correspondences must lie on the same incident ray, it follows that they must be colinear in 3D space. This property will be used to derive a constraint for computing the poses of the moving reference plane relative to its initial pose (see Section III-B).

B. Plane Pose Estimation

Referring to the setup shown in Fig. 2. Let the rigid body motion between $P_0$ and $P_1$ be denoted by $(\mathbf{R}^1, \mathbf{T}^1)$, where $\mathbf{R}^1$ and $\mathbf{T}^1$ are the rotation (matrix) and translation (vector) respectively. Similarly, let $(\mathbf{R}^2, \mathbf{T}^2)$ denote the rigid body motion between $P_0$ and $P_2$. Let $\mathbf{X}_0$, $\mathbf{X}_1$, and $\mathbf{X}_2$ be points on $P_0$, $P_1$ and $P_2$, respectively, that lie on the same incident light path. The 2D coordinates of $X_i$ on the plane are $\mathbf{X}_i^p = [x_i^p \ y_i^p \ 0]^T$, where $i \in \{0, 1, 2\}$. Their 3D coordinates, $\mathbf{X}_i = [x_i \ y_i \ z_i]^T$, $i \in \{0, 1, 2\}$, w.r.t $P_0$ can be written as

\[
\mathbf{X}_0 = \mathbf{X}_0^p = [x_0 \ y_0 \ z_0]^T = [x_0 \ y_0 \ 0]^T,
\]

\[
\mathbf{X}_1 = \mathbf{R}^1 \mathbf{X}_1^p + \mathbf{T}^1 = \mathcal{M} \mathbf{X}_1^p,
\]

\[
\mathbf{X}_2 = \mathbf{R}^2 \mathbf{X}_2^p + \mathbf{T}^2 = \mathcal{N} \mathbf{X}_2^p,
\]

where $\mathcal{M} = [\mathbf{R}_{x1}^1 \mathbf{R}_{x2}^1 \ \mathbf{T}^1]$, $\mathcal{N} = [\mathbf{R}_{x1}^2 \mathbf{R}_{x2}^2 \ \mathbf{T}^2]$, $\mathbf{X}_i^p = [x_i^p \ y_i^p \ 1]^T$, and $\mathbf{R}_{xj}^i$ denotes the $j$th column of $\mathbf{R}^i, i \in \{1, 2\}, j \in \{1, 2\}$. The unknown motion parameters are now embedded in $\mathcal{M}$ and $\mathcal{N}$. Since $\mathbf{X}_0$, $\mathbf{X}_1$ and $\mathbf{X}_2$ are colinear, it follows that

\[
\frac{x_1 - x_0}{x_2 - x_0} = \frac{y_1 - y_0}{y_2 - y_0} = \frac{z_1 - z_0}{z_2 - z_0},
\]

\[
\frac{\mathcal{M}_{1x} \mathbf{X}_1^p - \mathbf{x}_0}{\mathcal{N}_{1x} \mathbf{X}_2^p - \mathbf{y}_0} = \frac{\mathcal{M}_{2x} \mathbf{X}_1^p}{\mathcal{N}_{2x} \mathbf{X}_2^p - \mathbf{y}_0} = \frac{\mathcal{M}_{3x} \mathbf{X}_1^p}{\mathcal{N}_{3x} \mathbf{X}_2^p - \mathbf{y}_0} = 1
\]

(1)

where $\mathcal{M}_{ix}$ and $\mathcal{N}_{ix}$ denote the $i$th row of $\mathcal{M}$ and $\mathcal{N}$ respectively. The following two constraints can be derived from Equation (1):

\[
\begin{bmatrix}
(\mathbf{X}_2^p)^T \\
(\mathbf{X}_2^p)^T
\end{bmatrix} \begin{bmatrix}
\mathcal{A} \\
\mathcal{B}
\end{bmatrix} \begin{bmatrix}
\mathbf{X}_1^p - \mathbf{x}_0^p \\mathbf{X}_1^p - \mathbf{y}_0^p
\end{bmatrix} + \begin{bmatrix}
\mathcal{A}^T \\
\mathcal{B}^T
\end{bmatrix} \begin{bmatrix}
\mathcal{M}_{1x} \mathbf{X}_1^p \\
\mathcal{N}_{1x} \mathbf{X}_2^p
\end{bmatrix} \mathcal{M}_{3x}^T = 0,
\]

\[
\begin{bmatrix}
(\mathbf{X}_2^p)^T \\
(\mathbf{X}_2^p)^T
\end{bmatrix} \begin{bmatrix}
\mathcal{A} \\
\mathcal{B}
\end{bmatrix} \begin{bmatrix}
\mathbf{X}_1^p - \mathbf{x}_0^p \\mathbf{X}_1^p - \mathbf{y}_0^p
\end{bmatrix} \mathcal{N}_{3x}^T = 0,
\]

(2)
where $A = N_{3x}^T M_{1s} - N_{1x}^T M_{3s}$ and $B = N_{3x}^T M_{2s} - N_{2s}^T M_{3s}$.

Given $3 \times m$ points $\bar{X}_ij = [x_{ij}^p \ y_{ij}^p \ z_{ij}^p]^T$, where $0 \leq i \leq 2$, $1 \leq j \leq m$, $z_0 = 0$ and $z_{e(1,2)} = 1$, we can formulate the problem as solving a linear system

$$\mathbf{E} \mathbf{W} = \mathbf{0},$$

where

$$\mathbf{E} = \begin{bmatrix}
(x_{11}^p)^T (x_{11}^p) & \mathbf{0}^T & -x_{11}^p (x_{11}^p)^T & -x_{11}^p (x_{11}^p)^T \\
(x_{21}^p)^T (x_{11}^p) & \mathbf{0}^T & -x_{21}^p (x_{11}^p)^T & -x_{21}^p (x_{11}^p)^T \\
(x_{2m}^p)^T (x_{1m}^p) & \mathbf{0}^T & -x_{2m}^p (x_{1m}^p)^T & -x_{2m}^p (x_{1m}^p)^T \\
(x_{2m}^p)^T (x_{1m}^p) & \mathbf{0}^T & -x_{2m}^p (x_{1m}^p)^T & -x_{2m}^p (x_{1m}^p)^T
\end{bmatrix},$$

$$\mathbf{W} = \begin{bmatrix}
A_{1s} & A_{2s} & A_{3s} & B_{1s} & B_{2s} & B_{3s} & N_{1s} & M_{1s} & M_{2s}
\end{bmatrix}^T,$$

$A_{is}$ and $B_{is}$ denote the $i$th row of $A$ and $B$ respectively, and $\otimes$ denotes Kronecker tensor product. $\mathbf{W}$ contains 24 unknowns in total. Since each incident ray provides two constraints, we need at least 12 incident rays (i.e., $3 \times 12$) to solve the two scale parameters. We use the Symbolic Math Toolbox in Matlab to solve them (refer to the supplementary for more details).

IV. PROJECTION MATRIX ESTIMATION

In this section, we first briefly review the line projection matrix. We then derive a linear method for obtaining a closed-form solution to the line projection matrix of a camera from the reflection correspondences.

A. Line Projection Matrix

Using homogeneous coordinates, a linear mapping can be defined for mapping a point $\mathbf{x}$ in 3D space to a point $\mathbf{X}$ in a 2D image, i.e.,

$$\mathbf{x} = \mathbf{P} \mathbf{X},$$

where $\mathbf{P}$ is a $3 \times 4$ matrix known as the camera (point projection matrix). Similarly, using Plücker coordinates\footnote{A brief review of Plücker coordinates is given in the supplementary.}, a linear mapping can be defined for mapping a line $\mathcal{L}$ in 3D space to a line $\mathbf{l}$ (in homogeneous coordinates) in a 2D image, i.e.,

$$\mathbf{l} = \mathcal{P} \mathcal{L},$$

where $\mathcal{P}$ is a $3 \times 6$ matrix known as the line projection matrix and $\mathcal{L}$ is the dual Plücker vector\footnote{Given a Plücker vector $\mathcal{L} = [I_1 \ l_2 \ l_3 \ l_4 \ l_5 \ l_6]^T$, its dual Plücker vector is $\mathcal{L} = [I_5 \ l_6 \ l_4 \ l_3 \ l_1 \ l_2]^T$.} of $\mathcal{L}$. Note that each row $\mathcal{P}_s$ $(i \in \{1, 2, 3\})$ of $\mathcal{P}$ represents a plane (in homogeneous coordinates) that intersects at the optical center. Dually, each row $\mathcal{P}_s$ $(i \in \{1, 2, 3\})$ of $\mathcal{P}$ represents a line that intersects at the optical center (see Fig. 3). It follows that a valid line projection matrix must satisfy

$$\mathcal{P}_s \cdot \mathcal{P}_j = 0 \ \forall \ i, j \in \{1, 2, 3\} \iff \mathcal{P} \mathcal{P}^T = 0_{3,3},$$

where $\mathcal{P} = [\mathcal{P}_{1s}^T \mathcal{P}_{2s}^T \mathcal{P}_{3s}^T]^T$.

![Fig. 3. Visualization of the point and line projection matrices. (a) Rows of a point projection matrix represent planes that intersect at the optical center $C$ of the camera. (b) Dually, rows of a line projection matrix represent lines that intersect at the optical center.

A line projection matrix can be converted to its corresponding camera (point) projection matrix, and vice versa. Details of the conversion are given in the supplementary.

B. Estimating the Line Projection Matrix

To estimate the line projection matrix of the camera, we first apply our method described in Section III-B to recover the relative poses of the reference plane under three distinct poses using reflection correspondences established between the images and the reference plane. We can then form a 3D Plücker line $\mathcal{L}$ from the reflection correspondences of each observed point $\mathbf{x}$ in the image. Note that, by construction, $\mathbf{x}$ must lie on the projection of $\mathcal{L}$, i.e.,

$$\mathbf{x}^T \mathcal{P} \mathcal{L} = 0.$$
Given a set of 3D space lines \( \{L_1, ..., L_n\} \) constructed for a set of image points \( \{x_1, ..., x_n\} \), the constraint derived in (9) can be arranged into
\[
Zp = 0,
\]
where \( p = [P_{1s}^T \ P_{2s}^T \ P_{3s}^T]^T \) and
\[
Z = \begin{bmatrix}
x_1^T \otimes \bar{L}_1^T \\
\vdots \\
x_n^T \otimes \bar{L}_n^T
\end{bmatrix}.
\]
(11)

The line projection matrix of the camera can then be estimated by solving
\[
\arg\min_p \|Zp\|^2
\]
subject to \( \|p\| = 1 \). The line projection matrix thus obtained can be transformed into a point projection matrix and vice versa. Note that, however, (12) minimizes only algebraic errors and does not enforce (8). The solution to (12) is therefore subject to numerical instability and not robust in the presence of noise. Instead of solving (12), we can minimize the geometric distance from each image point to the projection of the corresponding 3D line. Let \( I = [a, b, c]^T = P \bar{L} \) be the projection of the 3D line \( \bar{L} \) corresponding to an image point \( x = [x_1, x_2, x_3]^T \). \( P \) can be estimated by solving
\[
\arg\min_p \sum_{i=1}^{n} (x_i^T P \bar{L}_i)^2
\]
subject to \( \|p\| = 1 \), where \( \|p\| \) is the Frobenius norm of \( P \). A straightforward approach to enforce (8) is by incorporating it as a hard constraint in (13). However, experiments using a number of state-of-the-art optimization schemes show that such a solution often converges to local minima.

C. Enforcing Constraints

Given a proper camera projection matrix, the corresponding line projection matrix will automatically satisfy (8). However, given an improper \( 3 \times 6 \) line projection matrix not satisfying (8), the corresponding camera projection matrix cannot be decomposed into one with proper intrinsic and extrinsic parameters. Based on this observation, we propose to enforce (8) through ensuring a proper decomposition of the camera projection matrix.

Consider a simplified scenario where the principal point \((u_0, v_0)\) (which is often located at the image centre) is known. After translating the image origin to the principal point, the camera projection matrix can be expressed as
\[
P = \begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix} R T
\]
and the corresponding line projection matrix (refer to the supplementary for the conversion) can be expressed as
\[
P' = \begin{bmatrix} f_y & 0 & 0 \\ 0 & f_x & 0 \\ 0 & 0 & f_x f_y \end{bmatrix} \cdot
\]

where
\[
P_{i*} = \begin{bmatrix} \rho_{i1} \\ \rho_{i2} \\ \rho_{i3} \\ \rho_{i4} \\ \rho_{i5} \\ \rho_{i6} \end{bmatrix} = (-1)^{(i+1)} \begin{bmatrix} r_{j3} t_k - t_j r_{k3} \\ t_j r_{k2} - r_{j2} t_k \\ r_{j2} t_k - r_{j3} t_k \\ r_{j1} t_k - t_j r_{k1} \\ r_{j1} t_k - r_{j2} t_k \\ r_{j3} t_k - r_{j1} t_k \end{bmatrix},
\]
with \( i \neq j \neq k \in \{1, 2, 3\} \) and \( j < k \).

In Section V, we tackle the problem of unknown camera intrinsics by formulating the problem into a nonlinear optimization by minimizing reprojection errors computed based on a cross-ratio formulation for the mirror surface. For initialization purpose, we assume \((u_0, v_0)\) being located at the image center, and \( f_x = f_y = f \). We choose a rough range of \( f \) and for each sample value of \( f \) within the range, we estimate \( R \) and \( T \) by solving (16). The point to line distance criterion in (13) is applied to find the best focal length \( f' \). A camera projection matrix can then be constructed using \( f' \), \((u_0, v_0)\), \( R \) and \( T \) that satisfies all the above mentioned constraints.

V. CROSS-RATIO BASED FORMULATION

In this section, we obtain the camera projection matrix and the mirror surface by minimizing reprojection errors. We will derive a cross-ratio based formulation for recovering a 3D point on the mirror surface from its reflection correspondences. Note that minimizing point-to-point reprojection errors can provide a stronger geometrical constraint than minimizing the point-to-line distances in (13) (see Fig. 4).

Consider a point \( M \) on the mirror surface (see Fig. 5). Let \( X_0, X_1 \) and \( X_2 \) be its reflection correspondences on the reference plane under three distinct poses, denoted by \( P_0, P_1 \) and \( P_2 \), respectively. Suppose \( M, X_0, X_1 \) and \( X_2 \) are projected to the image as \( \bar{m}, \bar{x}_0, \bar{x}_1 \) and \( \bar{x}_2 \) respectively. We observe that the cross-ratios \( [M, X_0; X_1, X_2] \) and \( [m, x_0; x_1, x_2] \) must be identical, i.e.,
\[
\frac{||\bar{X}_1\bar{M}|| ||\bar{X}_2\bar{X}_0||}{||\bar{X}_1\bar{X}_0|| ||\bar{X}_2\bar{M}||} = \frac{||\bar{x}_1\bar{m}|| ||\bar{x}_2\bar{x}_0||}{||\bar{x}_1\bar{x}_0|| ||\bar{x}_2\bar{m}||},
\]
(17)
where \( \bar{AB} \) denotes the directed ray (vector) from \( A \) to \( B \) and \( ||\bar{AB}|| \) is the length of the vector. Let \( s \) be the distance between \( X_2 \) and \( M \) (i.e., \( s = ||\bar{X}_2\bar{M}|| \)), from (17)
\[
s = \frac{||\bar{x}_2\bar{X}_0|| ||\bar{x}_2\bar{x}_0|| ||\bar{x}_1\bar{m}|| ||\bar{x}_2\bar{m}||}{||\bar{x}_1\bar{X}_0|| ||\bar{x}_2\bar{X}_0|| ||\bar{x}_1\bar{m}|| ||\bar{x}_2\bar{m}||}. \]
(18)
where $m$ denote the observation of $M$ denoted in Fig. 5. Cross-ratio constraint. Camera projection matrix and mirror surface points are recovered by minimizing reprojection errors computed from the cross-ratio constraint $\{M, X_0; X_1, X_2\} = \{m, x_0; x_1, x_2\}$, where $X_0, X_1, X_2$ are the reflection correspondences of $M$ on the reference plane under three different poses (i.e., $P_0, P_1$ and $P_2$) and $m, x_0, x_1, x_2$ are their projections on the image plane. Note that $X_0, X_1, X_2$ may not be visible to the camera.

Given the projection matrix, $x_0, x_1, x_2$ and $m$, the surface point $M$ can be recovered as

$$M = X_2 + s\frac{\overrightarrow{X_2 X_0}}{\|\overrightarrow{X_2 X_0}\|}. \quad (19)$$

We optimize the projection matrix by minimizing the reprojection errors, i.e.,

$$\arg\min_{\theta} \sum_{i=1}^{n} (m_i - m_i')^2, \quad (20)$$

where $m_i$ is the observation of $M_i$, $m_i' = P(\theta)M_i$, and $\theta = \left[f_x, f_y, u_0, v_0, r_x, r_y, r_z, t_x, t_y, t_z\right]^T$. To obtain $s$ in (18), we also project the three reflection correspondences of $m$, to the image using $P(\theta)$. We initialize $\theta$ using the method proposed in Section IV, and solve the optimization problem using the Levenberg-Marquardt method. Given the estimated projection matrix, the mirror surface can be robustly reconstructed by solving (17)-(19).

VI. DISCUSSION

A. Object Analysis

We analyze the surface points that can be reconstructed by our proposed method using an example as shown in Fig. 6, where a fixed camera centered at $C$ is viewing a mirror surface $S$, and a reference plane $XY$ is placed beside the surface. Consider the visual ray $\overrightarrow{CM}$ of a pixel $q$ for the surface point $M$. The angle between the incident ray and the reflected ray (visual ray) will restrict the surface point that can be reconstructed by our method. Given a reference plane $XY$ as in Fig. 6, the minimum angle between the incident ray and the reflected ray is $\angle CMX$ leading to a surface normal $\overrightarrow{MU}$, while the maximum angle is $\angle CMY$ leading to a surface normal $\overrightarrow{MV}$. The surface point $M$ can be reconstructed by our method when its normal lies in the range between $\overrightarrow{MU}$ and $\overrightarrow{MV}$. This range can be represented as the angle $\Delta = \angle UMV$, which is related to the distance between the surface and the reference plane, the size of the reference plane, and the relative pose between the camera and the reference plane.

![Fig. 6. A camera centered at $C$ observes a mirror surface $S$ and a reference plane $XY$ placed beside the surface.](image)

Let $\angle CMO = \theta$, $\|\overrightarrow{MO}\| = h_1$, $\|\overrightarrow{XO}\| = w_1$, and $\|\overrightarrow{YO}\| = w_2$. According to the law of reflection, we have

$$\angle CMU = \frac{1}{2}(\theta - \tan^{-1}\frac{w_1}{h_1}), \quad (21)$$

$$\angle CMV = \frac{1}{2}(\theta + \tan^{-1}\frac{w_2}{h_1}). \quad (22)$$

We used angle-axis representation for rotation, i.e., $[r_x, r_y, r_z]^T = \tau e$, where $\tau$ is the rotation angle and $e$ is the unit rotation axis.
It follows that
\[
\Delta = \angle CMV - \angle CMU = \tan^{-1}\frac{w_1}{h} + \tan^{-1}\frac{w_2}{h}. \tag{23}
\]
From (23), we can see that a larger size of the reference plane and/or a closer distance between the surface and the reference plane will result in a larger surface region that can be reconstructed by our method. Note that however according to (21), it is not necessary for \(w_1\) to be infinite large as we have to ensure \(\angle CMU \geq 0\) (i.e., \(w_1 \leq h \times \tan \theta\)) otherwise the camera will not perceive the reflected ray CM.

Thus, we can conclude that if the surface normals are distributed within the range defined by \(\Delta\), the whole surface can be reconstructed by our method. For a surface with a broader normal distribution, we can recover it by increasing the \(\Delta\) range via using a large enough reference plane and placing the reference plane at a location not too far away from the surface. In addition, placing the reference plane at multiple locations facing different (visible) regions of the object can also help enlarge the region that can be reconstructed. Alternatively, a cubic room like the one built in [21] can be used to enlarge the normal range that can be reconstructed by our method. However, our method requires moving each side of the room to three different locations, which makes it difficult to build such a room.

In addition, we discuss the relationship among the object size, distance from the camera to the object, and distance from the reference plane to the object as shown in Fig. 7. A camera centered at \(C\) observes the mirror surface \(S\). \(M_1\) and \(M_2\) are the normals at surface points \(M_1\) and \(M_2\). For simplicity, let \(\overrightarrow{M_1M_2} \parallel \overrightarrow{XY}\). We then construct points \(A\) and \(B\) with \(\overrightarrow{CA} \perp \overrightarrow{M_1M_2}\) and \(\overrightarrow{AB} \perp \overrightarrow{XY}\). A virtual camera center at \(C'\), can be formed by extending \(\overrightarrow{YM_1}\) and \(\overrightarrow{XM_2}\). Let \(\|\overrightarrow{CA}\| = h_1\), \(\|\overrightarrow{AB}\| = h_2\), \(\|\overrightarrow{M_1M_2}\| = s_1\), and \(\|\overrightarrow{XY}\| = s_2\), we have
\[
\frac{\|\overrightarrow{C'A}\|}{\|\overrightarrow{CA}\| + \|\overrightarrow{AB}\|} = \frac{\|\overrightarrow{CA}\|}{\|\overrightarrow{CA}\| + \|\overrightarrow{AB}\|} = \frac{h_1}{h_1 + h_2} = \frac{\|\overrightarrow{M_1M_2}\|}{\|\overrightarrow{XY}\|} = \frac{s_1}{s_2}. \tag{24}
\]
The maximum size of the surface that can be reconstructed by our method is then approximated by \(s_1 = \frac{h_1 s_2}{h_1 + h_2}\). Note that we use a planar surface \(S\) here only for discussion, while planar surfaces correspond to the degenerate case of our method as will be discussed next. As long as there are some normal variations between \(M_1\) and \(M_2\) on \(S\), the above discussion is still a valid approximation.

Lastly, it is worth noting that we assume the surface is perfectly reflective and our formulation depends on the law of reflection. Therefore, our method may not work for other mirror-like surfaces that do not follow the law of reflection. Meanwhile, for some mirror objects, the energy of the incident rays are not fully reflected due to the surface material. For such cases, the brightness of the sweeping stripes for correspondence estimation will be low, bringing more difficulties in identifying the peak from the intensity profile (see Fig. 8), thus reducing the accuracy of camera projection matrix estimation and shape reconstruction.

![Fig. 7. The relationship among the object size, distance from the camera to the object, and distance from the reference plane to the object.](image)

**B. Degeneracy**

In order to obtain 3D line correspondences, we proposed the method in Section III-B to estimate relative poses of the reference plane. However, there do exist degenerate cases. In particular, the relative poses of the reference plane cannot be uniquely determined in the following cases: (1) The specular object is of a planar, elliptical, parabolic or hyperbolic mirror; and (2) the arrangement of such mirror surfaces and the pinhole camera forms a central catadioptric system; namely the pinhole camera and the mirror surfaces form a single effective viewpoint. If the unknown mirror surface and the camera form a central catadioptric system with a single effective viewpoint, the relative pose estimation between reference planes (e.g., \(P_0\), \(P_1\), \(P_2\)) is equivalent to the camera calibration process described in [29]. As demonstrated in [29], the poses cannot be determined if the reference plane undergoes pure translation without knowing the translation direction. If the reference plane is placed at three general positions, the poses are estimated up to one scale ambiguity. As analysed in [30], a pinhole camera viewing a sphere does not form a central catadioptric system in general if the distance between the pinhole and effective viewpoint is larger than the radius. If the camera is located at the centre of the sphere, it indeed forms a central catadioptric system, which however differs from our experimental setup.

In addition, when \(\{m,x_0; x_1, x_2\}\) in Fig. 5 are close to each other, while \(\{M, X_0; X_1, X_2\}\) are further away to each other, our cross-ratio based formulation will be more sensitive to noise and a small error in the pixel domain will result in a large error in 3D points estimation. Actually, this is on par with small baseline triangulation as described in [31], which is generally more sensitive to noise.
VII. Evaluation

To demonstrate the effectiveness of our method, we evaluate it using both synthetic and real data. We first evaluate the effectiveness of our proposed approach for estimating the relative poses of the reference plane (Section III-B). We then show that our approach can faithfully reconstruct the mirror surfaces. The source code of our method can be found at https://github.com/k-han/mirror.

A. Relative Pose Estimation of the Reference Plane

To evaluate the performance of our reference plane pose estimation presented in Section III-B, we generated synthetic data using two spheres. The spheres have a radius of 300 mm. The reference plane used has a dimension of 2000×2000 mm² and was placed at three different poses denoted by P₀, P₁, and P₂ respectively. The reflection correspondences were obtained via ray tracing. To evaluate the robustness of our method, we added Gaussian noise to the reflection correspondences on the reference plane with standard deviations ranging from 0 to 3.0 mm. The errors were reported as the average value over 50 trials of the experiments for each noise level (Gaussian noise standard deviation). Specifically, we reported the errors in the relative rotation matrix R in terms of the angle of the rotation induced by R$_{gt}$R$^{T}$, where R$_{gt}$ denotes the ground truth rotation matrix. We reported the errors in the translation vector T in terms of the angle (T$^{deg}$) between T and T$_{gt}$, where T$_{gt}$ denotes the ground truth translation vector and T$^{scale} = ||T_{gt} - T||$. In Fig. 9, we show the estimation errors for (R$^{T}$, T$^{1}$) and (R$^{2}$, T$^{2}$). It can be seen that the errors increase with the noise level, while the magnitude of the errors are quite small, demonstrating that our relative pose estimation method is robust to the noise.

B. Recovery of Projection Matrix and Mirror Surface

1) Synthetic Data Experiments: We employed a reflective Stanford bunny and a reflective engine hood created by [21] to generate our synthetic data. The bunny has a dimension of 880×680×870 mm³ and 208,573 surface points, while the hood has a dimension of 2120×1180×270 mm³ and 38,546 surface points. The images have a resolution of 960×1280 pixels. Figure 10(a) shows the reflective appearances of the bunny and the hood. In their original data, the bunny (hood) was placed in a cubic room, with each side of the room working as a reference plane. The reference plane has a dimension of 3048×3048 mm². The center of the room was defined as the world origin. A camera was placed in the room viewing the bunny (hood). Since our method requires reflection correspondences under three distinct poses of the reference plane, we introduced two additional planes for each side of the room and obtained the reflection correspondences through ray tracing.

![Fig. 10. RMS errors for the mirror Stanford bunny (first row) and engine hood (second row). (a) An image of the mirror Stanford bunny and an image of the mirror engine hood. (b) RMS reprojection errors (computed against ground truth image points). (c) RMS reconstruction errors (computed against ground truth 3D surface points).](image)

![Fig. 9. Noise sensitivity analysis for relative pose estimation of the reference plane. The red curves and blue curves are the pose estimation errors for P₁ and P₂, respectively, under different noise levels. (a) Relative rotation error. (b) Translation angular error. (c) Translation scale error.](image)

<table>
<thead>
<tr>
<th>TABLE I</th>
<th>ESTIMATION ERROR UNDER NOISE LEVEL σ = 2.0 [mm] ON BUNNY.</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>f_u [%]</td>
</tr>
<tr>
<td>L</td>
<td>1.89</td>
</tr>
<tr>
<td>EL</td>
<td>0.94</td>
</tr>
<tr>
<td>CR</td>
<td>0.11</td>
</tr>
</tbody>
</table>

L: linear solution in Section IV-B; EL: constrained linear solution with strategy in Section IV-C; CR: estimation using cross-ratio formulation initialized with EL.

To evaluate the performance of our method, we added Gaussian noise to the 2D reflection correspondences on the reference plane with standard deviations ranging from 0 to 3.0 mm. The reflection correspondences are the input of our method and essential to the performance. It is difficult to obtain high quality reflection correspondences due to the distortion caused by the reflection of the mirror surface. Thus we added the noise to the reflection correspondences. The typical intensity profile for the x- or y-coordinate is a bell-shaped curve (see Fig. 8 for an example), which has a similar characteristic as Gaussian distribution, hence we choose Gaussian noise to model the possible noises in the 2D correspondences. We initialized the projection matrix using the method described in Section IV. The optimized projection matrix together with the 3D surface points were obtained by minimizing reprojection errors computed based on our cross-ratio formulation. In Table I, we compared: 1) linear solution by solving (12) in Section IV-B, denoted as L; 2) linear solution by solving (16) after enforcing the constraints in Section IV-C, denoted as EL; and 3) cross-ratio based non-linear solution by solving (20) in Section V, denoted as
Fig. 11. Reconstruction results of bunny and hood shapes. First and fourth rows: reconstructed point clouds under different noise levels of the bunny and the hood, respectively. Coordinates are w.r.t world and colors are rendered w.r.t z coordinates. Note that the missing regions of the bunny are due to the lack of correspondences in the original data set. Second and fifth rows: surfaces of the bunny and the hood generated using screened Poisson surface reconstruction method in [32]. Third and last rows: visualization of surface point reconstruction errors.

TABLE II

<table>
<thead>
<tr>
<th>noise lv</th>
<th>( f_x [\text{pixel}] )</th>
<th>( f_y [\text{pixel}] )</th>
<th>( u_0 [\text{pixel}] )</th>
<th>( v_0 [\text{pixel}] )</th>
<th>( R [\text{°}] )</th>
<th>( T_{ho0} [\text{°}] )</th>
<th>( T_{scale} [\text{mm}] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>bunny</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \sigma = 0.5 )</td>
<td>0.22(0.02%)</td>
<td>0.22(0.02%)</td>
<td>0.11(0.06%)</td>
<td>0.00(0.01%)</td>
<td>0.02</td>
<td>0.02</td>
<td>0.53(0.03%)</td>
</tr>
<tr>
<td>( \sigma = 1.0 )</td>
<td>0.33(0.02%)</td>
<td>0.33(0.02%)</td>
<td>0.41(0.06%)</td>
<td>0.05(0.01%)</td>
<td>0.02</td>
<td>0.02</td>
<td>0.52(0.03%)</td>
</tr>
<tr>
<td>( \sigma = 1.5 )</td>
<td>0.50(0.04%)</td>
<td>0.51(0.04%)</td>
<td>0.56(0.09%)</td>
<td>0.90(0.20%)</td>
<td>0.04</td>
<td>0.05</td>
<td>1.74(0.10%)</td>
</tr>
<tr>
<td>( \sigma = 2.0 )</td>
<td>1.52(0.11%)</td>
<td>1.52(0.11%)</td>
<td>1.16(0.18%)</td>
<td>1.22(0.25%)</td>
<td>0.08</td>
<td>0.07</td>
<td>2.42(0.13%)</td>
</tr>
<tr>
<td>( \sigma = 2.5 )</td>
<td>4.36(0.31%)</td>
<td>4.36(0.31%)</td>
<td>2.11(0.32%)</td>
<td>2.36(0.49%)</td>
<td>0.37</td>
<td>0.56</td>
<td>7.33(0.40%)</td>
</tr>
<tr>
<td>( \sigma = 3.0 )</td>
<td>10.15(0.73%)</td>
<td>10.11(0.73%)</td>
<td>5.76(0.90%)</td>
<td>3.08(0.64%)</td>
<td>0.85</td>
<td>0.70</td>
<td>13.29(0.73%)</td>
</tr>
<tr>
<td>hood</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \sigma = 0.5 )</td>
<td>0.25(0.04%)</td>
<td>0.30(0.04%)</td>
<td>0.18(0.03%)</td>
<td>0.71(0.15%)</td>
<td>0.15</td>
<td>0.29</td>
<td>1.07(0.10%)</td>
</tr>
<tr>
<td>( \sigma = 1.0 )</td>
<td>1.43(0.10%)</td>
<td>1.43(0.10%)</td>
<td>1.14(0.18%)</td>
<td>0.72(0.15%)</td>
<td>0.20</td>
<td>0.26</td>
<td>2.45(0.23%)</td>
</tr>
<tr>
<td>( \sigma = 1.5 )</td>
<td>2.53(0.18%)</td>
<td>2.56(0.18%)</td>
<td>1.10(0.17%)</td>
<td>1.43(0.30%)</td>
<td>0.17</td>
<td>0.34</td>
<td>5.01(0.47%)</td>
</tr>
<tr>
<td>( \sigma = 2.0 )</td>
<td>4.87(0.35%)</td>
<td>4.87(0.18%)</td>
<td>3.08(0.48%)</td>
<td>2.66(0.55%)</td>
<td>0.33</td>
<td>0.35</td>
<td>8.38(0.79%)</td>
</tr>
<tr>
<td>( \sigma = 2.5 )</td>
<td>7.32(0.52%)</td>
<td>7.32(0.52%)</td>
<td>4.74(0.74%)</td>
<td>5.37(1.12%)</td>
<td>0.53</td>
<td>0.55</td>
<td>14.01(1.32%)</td>
</tr>
<tr>
<td>( \sigma = 3.0 )</td>
<td>11.53(0.82%)</td>
<td>11.53(0.82%)</td>
<td>6.55(1.02%)</td>
<td>7.71(1.61%)</td>
<td>0.54</td>
<td>0.86</td>
<td>17.57(1.66%)</td>
</tr>
</tbody>
</table>

The ground truth for the intrinsic parameters are \( f_x = 1400, f_y = 1400, \) and \( (u_0, v_0) = (639.5, 479.5) \). The norm of translation vectors are 1811.2 and 1057.9 for bunny and hood, respectively.
CR. It can be seen that our cross-ratio based formulation can effectively improve the linear solutions. Figure 10(b) and (c) depict the root mean square (RMS) reprojection errors and reconstruction errors, respectively, under different noise levels. It can be seen that the reprojection errors and the reconstruction errors increase linearly with the noise level. The magnitude of the reconstruction errors is relatively small compared to the size of the object. Figure 11 shows the reconstructed point clouds w.r.t. the ground truth. Among all noise levels, the errors are below 1% for \( f_u, f_v, u_0, v_0 \) and \( T_{\text{scale}} \), and angular errors are below 1° for \( R \) and \( T \).

Moreover, as the quantization noise, which obeys the uniform distribution, is also typical in real situations, we further verify the robustness of our approach to the quantization noise by adding the uniform noise in the range of \([-\gamma, \gamma]\) to the 2D pixel locations. The results are reported in Table III and Fig. 12. The errors increase linearly with the noise level while the estimation errors are small, revealing that our approach is robust to quantization noise.

![Table III: Estimation errors on bunny and hood under quantization noise](image)

**Table III**: Estimation errors on bunny and hood under quantization noise \( \gamma = 2.0 \) [pixel].

<table>
<thead>
<tr>
<th></th>
<th>( f_u ) [%]</th>
<th>( f_v ) [%]</th>
<th>( u_0 ) [%]</th>
<th>( v_0 ) [%]</th>
<th>( R ) [%]</th>
<th>( T_{\text{deg}} ) [%]</th>
<th>( T_{\text{scale}} ) [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bunny</td>
<td>0.12</td>
<td>0.12</td>
<td>0.20</td>
<td>0.72</td>
<td>0.15</td>
<td>0.20</td>
<td>0.36</td>
</tr>
<tr>
<td>Hood</td>
<td>0.08</td>
<td>0.08</td>
<td>0.11</td>
<td>0.27</td>
<td>0.03</td>
<td>0.07</td>
<td>0.17</td>
</tr>
</tbody>
</table>

![Fig. 12: Reconstruction results under different quantization noise.](image)

Besides, we compared our method with state-of-the-art mirror surface reconstruction method in [15] under smooth surface assumption and calibrated setup. Note that [15] assumes the mirror surface is \( C^2 \) continuous. In order to make a fair comparison, we performed the experiment on a sphere patch under the same setup with the Stanford bunny and engine hood datasets. Fig. 13 depicts the comparison between fully calibrated ([15]) and uncalibrated (proposed) methods. The overall reconstruction accuracies are similar. While our result is not as smooth as that of [15] due to our point-wise reconstruction. Their result shows a global reconstruction bias due to the B-spline parameterization for the surface (see Fig. 13).

![Fig. 13: Comparison with a fully calibrated method of Liu et al. [15]. Upper left: ground truth. Lower left: RMS reconstruction errors. Upper right ([15]) & lower right (ours): reconstruction (blue) against ground truth (red) under \( \sigma = 2.0 \). Our uncalibrated approach achieves comparable accuracy with that of the fully calibrated method in [15].](image)

![Fig. 14: Reconstruction results under different distortion coefficients.](image)

We consider the ideal pinhole camera model, therefore the potential lens distortion is not taken into account in our formulation. In practice, the lens may have radial and tangential distortion [33]. The one-parameter radial distortion is the most common form of distortion, which can be described by

\[
\begin{align*}
x_2 &= x_1 \left(1 + k_1 r^2 \right) + \frac{k_2 r^2}{1 + k_1 r^2} + x_0 \\
y_2 &= y_1 \left(1 + k_1 r^2 \right) + \frac{k_2 r^2}{1 + k_1 r^2} + y_0
\end{align*}
\]

(24)

where \((x_1, y_1)\) and \((x_2, y_2)\) are the undistorted and distorted pixel locations in normalized image coordinates \([-1, 1]\), \(k_1\) is the radial distortion coefficient (positive/negative radial distortion) of the lens and \(r = \sqrt{x_1^2 + y_1^2}\). To verify the robustness of our method to the lens distortion, we thoroughly evaluate the performance of our method on bunny and hood shapes by introducing radial

![Fig. 15: Surface reconstruction errors of bunny. “init” indicates ray triangulation based on the initial camera projection matrix by our method in Section IV-C; “triangulation” indicates ray triangulation based on the camera projection matrix by our cross-ratio based formulation; “cross-ratio” indicates our cross-ratio based method.](image)
better performance. Although ray triangulation based formulation can provide better performance than our cross-ratio based formulation, leading to a more accurate reconstruction result, it includes more geometric regularization than ray triangulation based formulation, leading to a more accurate reconstruction result. As shown in Section IV-C, our cross-ratio based method can significantly improve the reconstruction results. In addition, we also explored the triangulation counterpart for the optimization problem in (20), by simply replacing the 3D point formulation in (19) with ray triangulation (i.e., triangulating the incident and the reflected (visual) rays). The results are reported in Table V and Fig. 17. Meanwhile, comparing with the ray triangulation results using our cross-ratio based method in Section IV-C, our cross-ratio based method can significantly improve the reconstruction results. In addition, we also explored the triangulation counterpart for the optimization problem in (20), by simply replacing the 3D point formulation in (19) with ray triangulation (i.e., triangulating the incident and the reflected (visual) rays). The results are reported in Table V and Fig. 17. Though the triangulation based formulation can also improve the results obtained by our constrained linear solution introduced in Section IV-C (Table I vs Table V), the cross-ratio formulation performs notably better. Both formulations aim to minimize the point-to-point reprojection error. However, as our cross-ratio based formulation also enforces the cross-ratio constraint during optimization, it includes more geometric regularization than ray triangulation based formulation, leading to better performance.

### TABLE IV

<table>
<thead>
<tr>
<th>Bunny</th>
<th>δ [pix.]</th>
<th>f_u [%]</th>
<th>f_v [%]</th>
<th>τ_u [%]</th>
<th>τ_v [%]</th>
<th>R [°]</th>
<th>Tdeg [°]</th>
<th>Tscale [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.22/2.21</td>
<td>0.94/0.95</td>
<td>0.93/0.95</td>
<td>1.12/1.41</td>
<td>1.09/1.88</td>
<td>0.78/0.79</td>
<td>0.83/0.81</td>
<td>6.01/5.73</td>
<td></td>
</tr>
</tbody>
</table>

δ denotes the mean pixel offsets introduced by the lens distortion.

distortion of different levels. After introducing distortion, the pixel locations are denormalized to the original magnitude for camera pose estimation and surface reconstruction. We vary the distortion coefficient \( k_1 \) and measure the reconstruction errors on bunny and hood (see Fig. 14). In Table IV, we show the results of our approach under with \( k_1 = 0.02/-0.02 \). It can be seen that our method is generally robust to the lens distortion.

We further compare the surface reconstruction results between the ray triangulation and our cross-ratio based method in Fig. 15, given the camera projection matrix estimated by our method. It can be seen that with the estimated camera projection matrix, the reconstruction accuracy of ray triangulation is on par with our cross-ratio based formulation. As the surface points can be obtained as the by-product of our cross-ratio based methods, the extra ray triangulation is no longer needed. Meanwhile, comparing with the ray triangulation results using our camera projection matrix initialization method in Section IV-C, our cross-ratio based method can significantly improve the reconstruction results. In addition, we also explored the triangulation counterpart for the optimization problem in (20), by simply replacing the 3D point formulation in (19) with ray triangulation (i.e., triangulating the incident and the reflected (visual) rays). The results are reported in Table V and Fig. 17. Though the triangulation based formulation can also improve the results obtained by our constrained linear solution introduced in Section IV-C (Table I vs Table V), the cross-ratio formulation performs notably better. Both formulations aim to minimize the point-to-point reprojection error. However, as our cross-ratio based formulation also enforces the cross-ratio constraint during optimization, it includes more geometric regularization than ray triangulation based formulation, leading to better performance.

### TABLE V

<table>
<thead>
<tr>
<th>bunny</th>
<th>( f_u ) [%]</th>
<th>( f_v ) [%]</th>
<th>( τ_u ) [%]</th>
<th>( τ_v ) [%]</th>
<th>( R ) [°]</th>
<th>( T_{\text{deg}} ) [°]</th>
<th>( T_{\text{scale}} ) [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>TR</td>
<td>0.82</td>
<td>0.82</td>
<td>1.02</td>
<td>1.09</td>
<td>0.69</td>
<td>0.46</td>
<td>0.16</td>
</tr>
<tr>
<td>CR</td>
<td>0.11</td>
<td>0.11</td>
<td>0.18</td>
<td>0.25</td>
<td>0.08</td>
<td>0.07</td>
<td>0.13</td>
</tr>
</tbody>
</table>

After obtaining the camera projection matrix and surface points by our cross-ratio based method, we can obtain the normal by simply calculating the bisector of the visual ray and incident ray. Figure 16 shows the normal estimation results on hood. It can be seen that our method can estimate accurate normals. In this work, we focus more on recovering the point cloud because it contains the actual scale information of the object.

![Normal reconstruction results of hood shape](Image)

Fig. 16. Normal reconstruction results of hood shape. Left to right: ground truth, normal estimation w/ correspondence noise \( \sigma = 2.0 \), normal estimation w/ correspondence noise \( \sigma = 2.0 \), normal estimation w/ distortion coefficient \( k_1 = 0.02 \), normal estimation error w/ distortion coefficient \( k_1 = 0.02 \). The color in error maps indicates error in degree.

![Surface reconstruction errors of bunny](Image)

Fig. 17. Surface reconstruction errors of bunny using cross-ratio and triangulation based optimization formulations respectively.

After obtaining the camera projection matrix and surface points by our cross-ratio based method, we can obtain the normal by simply calculating the bisector of the visual ray and incident ray. Figure 16 shows the normal estimation results on hood. It can be seen that our method can estimate accurate normals. In this work, we focus more on recovering the point cloud because it contains the actual scale information of the object.

![Real data experiments](Image)

Fig. 18. Sauce boat and two spheres in real data experiments. Top row: sauce boat and two spheres in real experiments. Bottom row: a sweeping line is reflected by two spheres under three distinct positions of the LCD monitor while the camera and mirror surfaces are stationary.

2) Real Data Experiments: We evaluated our method on a sauce boat and two spheres respectively (see Fig. 18). We captured images using a Canon EOS 40D digital camera with a 24-70 mm lens. A 19 inch LCD monitor was used as a reference plane and was placed at three different locations. We follow [24], [34], [35] to display sweeping stripes on the monitor to establish reflection correspondences. In particular,
at each location, the LCD monitor shows a thin stripe sweeping across the screen in vertical direction and then in horizontal direction. For each position of the sweeping stripe, we capture an image. We then obtain a sequence of images w.r.t the vertically sweeping stripe and a sequence of images w.r.t the horizontally sweeping stripe. We can then obtain the correspondence on the reference plane for each image point by examining its corresponding two sequences of intensity values. The position of the stripe that produces the peak intensity value in each sweeping direction then gives the position of the correspondence on the reference plane. In order to improve the accuracy of the peak localization, we fit a quadratic curve to the intensity profile in the neighborhood of the sampled peak value, and solve for the exact peak analytically.

After establishing reflection correspondences, we first estimated the relative poses of the reference plane using our method described in Section III-B. We then formed 3D lines from the reflection correspondences on the reference plane under the two poses that were furthest apart (e.g., \( P_0 \) and \( P_2 \) in Fig. 5). These 3D lines were used to obtain a preliminary solution of projection matrix using the linear method in Section IV, which was then used to initialize the nonlinear optimization described in Section V.

To evaluate our method, we calibrated the camera and uncalibrated plane poses, respectively. The ground truth for the intrinsic parameters are

\[
\begin{align*}
(B_{uu} & = 36.70(0.63\%) \quad f_u = 21.99(0.38\%) \quad u_0 = 99.10(5.03\%) \quad v_0 = 100.00(8.13\%) \quad R = 9.12 \quad T_{deg} = 1.00 \quad T_{scale} = 19.16(8.23\%) \quad S_{rms} = 2.55) \\
(B_{cu} & = 101.70(1.75\%) \quad f_c = 86.90(1.49\%) \quad u_c = 112.10(5.69\%) \quad v_c = 113.00(9.19\%) \quad R = 9.86 \quad T_{deg} = 1.99 \quad T_{scale} = 17.94(7.34\%) \quad S_{rms} = 2.71) \\
(S_{cu} & = 63.38(1.09\%) \quad f_c = 68.01(1.17\%) \quad u_c = 61.49(3.18\%) \quad v_c = 42.7(3.47\%) \quad R = 6.67 \quad T_{deg} = 1.78 \quad T_{scale} = 33.83(8.96\%) \quad S_{rms} = 1.78) \\
(S_{uu} & = 81.38(1.40\%) \quad f_u = 86.02(1.48\%) \quad u_u = 81.67(4.14\%) \quad v_u = 56.70(4.61\%) \quad R = 7.17 \quad T_{deg} = 2.13 \quad T_{scale} = 37.69(9.98\%) \quad S_{rms} = 2.03)
\end{align*}
\]

\( B \) and \( S \) denote the results of sauce boat and two spheres respectively. The subscripts \( uc, cu, \) and \( uu \) stand for experiments under (1) an uncalibrated camera and calibrated plane poses, (2) a calibrated camera and uncalibrated plane poses, and (3) an uncalibrated camera and uncalibrated plane poses, respectively. The ground truth for the intrinsic parameters are \( f_u = 5812.86, \ f_c = 5812.82, \) and \( (u_0, v_0) = (1971.95, 1230.02) \). \( S_{rms} \) stands for the RMS reconstruction error.

<table>
<thead>
<tr>
<th>( B_{uu} )</th>
<th>( B_{cu} )</th>
<th>( S_{cu} )</th>
<th>( S_{uu} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_u ) [pixel]</td>
<td>( f_u ) [pixel]</td>
<td>( u_c ) [pixel]</td>
<td>( v_c ) [pixel]</td>
</tr>
<tr>
<td>36.70(0.63%)</td>
<td>101.70(1.75%)</td>
<td>63.38(1.09%)</td>
<td>81.38(1.40%)</td>
</tr>
<tr>
<td>21.99(0.38%)</td>
<td>86.90(1.49%)</td>
<td>68.01(1.17%)</td>
<td>86.02(1.48%)</td>
</tr>
<tr>
<td>99.10(5.03%)</td>
<td>112.10(5.69%)</td>
<td>61.49(3.18%)</td>
<td>81.67(4.14%)</td>
</tr>
<tr>
<td>100.00(8.13%)</td>
<td>113.00(9.19%)</td>
<td>42.7(3.47%)</td>
<td>56.70(4.61%)</td>
</tr>
<tr>
<td>9.12</td>
<td>9.86</td>
<td>6.67</td>
<td>7.17</td>
</tr>
<tr>
<td>1.00</td>
<td>1.99</td>
<td>1.78</td>
<td>2.13</td>
</tr>
<tr>
<td>19.16(8.23%)</td>
<td>17.94(7.34%)</td>
<td>33.83(8.96%)</td>
<td>37.69(9.98%)</td>
</tr>
<tr>
<td>2.55</td>
<td>2.71</td>
<td>1.78</td>
<td>2.03</td>
</tr>
</tbody>
</table>

Fig. 19. Reconstruction results of sauce boat and two spheres. (a)-(d): reconstructions of sauce boat. Results are obtained under (a) calibrated camera with calibrated plane poses (this result is treated as ground truth and overlaid in (b), (c) and (d) for comparison (red)); (b) uncalibrated camera with calibrated plane poses (blue); (c) calibrated camera with uncalibrated plane poses (blue); (d) uncalibrated camera with uncalibrated plane poses (ours, blue). Note the missing regions (in red rectangle) in the reconstructed point clouds are filled by the mesh generation algorithm and should be ignored in comparing the surface meshes. (e)-(h): reconstructions of two spheres.
reference plane poses using the Camera Calibration Toolbox for Matlab [36]. We used the calibration result to estimate the surface and treated it as the ground truth. This result was compared against the result obtained using 1) uncalibrated camera and calibrated plane poses, 2) calibrated camera and uncalibrated plane poses, and 3) uncalibrated camera and uncalibrated plane poses. Fig. 19 shows the reconstructed surfaces and Table VI shows the numerical errors. We aligned each estimated surface with the ground truth by a rigid body transformation using the method in [37] before computing the reconstruction error. The RMS reconstruction errors are below 3 mm. $f_u$ and $f_v$ errors are below 2%. $u_0$, $v_0$ and $T_{scale}$ errors are below 10%. The angular errors are below $10^\circ$ for $R$ and below $3^\circ$ for $T$. The errors in intrinsics and extrinsics are larger than those in the synthetic experiments.

This is reasonable since accurate reflection correspondences in real cases are difficult to obtain due to the large and complex distortion caused by the mirror surface and varying lighting condition. Besides, degradation from optics, such as optics aberrations, will also introduce errors for establishing reflections correspondences. The quality of the reflection correspondences is also a key factor for other existing mirror surface reconstruction methods. Note that existing methods are designed under certain assumptions (e.g., convex, $C^n$ continuity, etc), and their setups are carefully tailored or require special equipments. Besides, there is no publicly available dataset that can serve as a benchmark for existing methods.

As a result, it is challenging to make a fair comparison with existing methods on real data. Therefore, we do not include comparison with other existing methods on real data.

VIII. CONCLUSIONS

A novel method is introduced for mirror surface reconstruction. Our method works under an uncalibrated setup and can recover the camera intrinsics and extrinsics, along with the surface. We proposed an analytical solution for the reference plane relative pose estimation, a closed-form solution for camera projection matrix estimation, and a cross-ratio based formulation to achieve a robust estimation of both camera plane relative pose estimation, a closed-form solution for surface. We proposed an analytical solution for the reference plane should be moved three times at each location. For mirror surfaces with complex normal distribution, the surface is likely to happen for regions with relatively sharp changes.

This is reasonable since accurate reflection correspondences in real cases are difficult to obtain due to the large and complex distortion caused by the mirror surface and varying lighting condition. Besides, degradation from optics, such as optics aberrations, will also introduce errors for establishing reflections correspondences. The quality of the reflection correspondences is also a key factor for other existing mirror surface reconstruction methods. Note that existing methods are designed under certain assumptions (e.g., convex, $C^n$ continuity, etc), and their setups are carefully tailored or require special equipments. Besides, there is no publicly available dataset that can serve as a benchmark for existing methods.

As a result, it is challenging to make a fair comparison with existing methods on real data. Therefore, we do not include comparison with other existing methods on real data.

VIII. CONCLUSIONS

A novel method is introduced for mirror surface reconstruction. Our method works under an uncalibrated setup and can recover the camera intrinsics and extrinsics, along with the surface. We proposed an analytical solution for the reference plane relative pose estimation, a closed-form solution for camera projection matrix estimation, and a cross-ratio based formulation to achieve a robust estimation of both camera plane relative pose estimation, a closed-form solution for surface. We proposed an analytical solution for the reference plane should be moved three times at each location. For mirror surfaces with complex normal distribution, the surface is likely to happen for regions with relatively sharp changes.

This is reasonable since accurate reflection correspondences in real cases are difficult to obtain due to the large and complex distortion caused by the mirror surface and varying lighting condition. Besides, degradation from optics, such as optics aberrations, will also introduce errors for establishing reflections correspondences. The quality of the reflection correspondences is also a key factor for other existing mirror surface reconstruction methods. Note that existing methods are designed under certain assumptions (e.g., convex, $C^n$ continuity, etc), and their setups are carefully tailored or require special equipments. Besides, there is no publicly available dataset that can serve as a benchmark for existing methods. This greatly simplifies the challenging problem of mirror surface recovery. We believe our work can provide a meaningful insight towards solving this problem.

Our method does have a few limitations. We assume the surface is perfectly reflective and our formulation depends on the law of reflection. Thus our method may not work for other mirror-like surfaces that do not follow the law of reflection, or mirror surfaces that do not reflect the full energy of the incident rays. Although our cross-ratio based formulation does not encounter degenerate cases, degenerate cases may occur in the relative pose estimation of the reference plane. Efforts are needed to explore methods without degeneracy for estimating the relative poses of the reference plane from reflection correspondences. Meanwhile, the region that can be reconstructed by our method depends on the size of the reference plane. For mirror surfaces with complex normal distribution, the reference plane may need to be carefully placed at different locations facing different regions of the mirror surface, and the reference plane should be moved three times at each location.

Despite an increased number of images, extra efforts are also required to fuse these reconstructed regions together. Besides, in our formulation, we did not consider inter-reflection, which is likely to happen for regions with relatively sharp changes in the mirror surface. In the future, we would like to develop methods to recover complete surfaces and further investigate inter-reflection cases.

ACKNOWLEDGEMENT

This project is supported by a grant from the Research Grant Council of the Hong Kong (SAR), China, under the project HKU 17203119 and a grant (DE180100628) from Australian Research Council (ARC).

REFERENCES


[34] K. Han, K.-Y. K. Wong, and M. Liu, “A fixed viewpoint approach for dense reconstruction of transparent objects,” in CVPR, 2015, pp. 4001–4008.


Kai Han received the Ph.D. degree from the Department of Computer Science, The University of Hong Kong, in 2018. He was a Postdoctoral Researcher with the Visual Geometry Group (VGG), Department of Engineering Science, University of Oxford, from 2018 to 2020. He was a Research Intern with the WILLOW Group, INRIA / Ecole Normale Supérieure in 2016. He is currently a Lecturer (Assistant Professor) in Computer Vision with the Department of Computer Science at University of Bristol. His research interests include computer vision and deep learning.

Miaomiao Liu (Member, IEEE) received the B.Eng degree from Yantai Normal University, Yantai, China, in 2004, the M.Eng. degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2007, and the Ph.D. degree from The University of Hong Kong, Hong Kong, China, in 2012. She worked as a Researcher with the Computer Vision Group, NICTA, from 2012 to 2016, and a Research Scientist with the CSIRO’s Data61, Canberra, ACT, Australia, from 2017 to 2018. She joined the Australian National University (ANU) as an ARC DECRA fellow in 2018, where she is currently holding a Tenure-track Senior Lecturer position.

Dirk Schnieders graduated with a Diplom-Informatiker (FH) degree from the Fachhochschule Dortmund (Germany) in 2004 and received his MSc and PhD in Computer Science from The University of Hong Kong in 2007 and 2011. Currently, he is a Lecturer at the Department of Computer Science, The University of Hong Kong.

Kwan-Yee K. Wong (Senior Member, IEEE) received the B.Eng. degree (Hons.) in computer engineering from The Chinese University of Hong Kong in 1998, and the M.Phil. and Ph.D. degrees in computer vision (information engineering) from the University of Cambridge in 2000 and 2001, respectively. Since 2001, he has been with the Department of Computer Science, The University of Hong Kong, where he is currently an Associate Professor. His research interests include computer vision and machine intelligence. He is currently an editorial board member of International Journal of Computer Vision (IJCV).