A Novel Catadioptric Ray-Pixel Camera Model and its Application to 3D Reconstruction

RYO KAWAHARA¹,a) SHOHEI NOBUHARA¹,b)

Abstract: This paper proposes a new ray-pixel camera model for image-based 3D measurement by a catadioptric imaging system. The key idea of this paper is to employ a virtual camera model that describes ray-pixel mappings with exploiting an axially-symmetric structure of the ray distribution of the system. Our contributions include, structured ray-pixel camera models which handle refractive and reflective projection rays efficiently, and practical calibration algorithms for them. Evaluations with real images prove the concept of our measurement system.

1. Introduction

3D shape acquisition has been an important topic in computer vision as an essential factor for interacting with real world, and there are a large number of studies in particular for capturing human[1], [2], [3] and buildings[4], [5]. Applications of unconstrained and noninvasive image-based 3D shape capturing include digital archiving, navigation in surgery, SLAM, industrial inspection, virtual reality, surveying and measurement, etc.

Most of the 3D shape capture studies in literature utilize regular perspective cameras. Catadioptric system with additional lenses and mirrors, however, can also be a practical solution for particular targets and scenes. For example, catadioptric system has been widely studied for panoramic imaging[6], [7]. Besides, in the case of underwater 3D capture, measurement through waterproof housings or aquarium surfaces can also be considered as catadioptric system.

For such catadioptric measurements, the rays captured by the system do not form a pencil due to reflections and refractions, and the optical system as a whole cannot be modeled as perspective. Therefore the 3D recovery via triangulation becomes a non-trivial process as a result. The goal of this paper is to propose a camera model that can handle such projections in catadioptric system efficiently.

Our approach is to employ a ray-pixel (raxel) camera model which focuses on the mapping from each pixel to the scene ray[8]. It implements 2D-3D projection just by storing the pixel-ray correspondences without tracing the optical path between them, and therefore it is flexible enough to model the light-field captured by the catadioptric system.

The general ray-pixel representation, however, cannot provide 3D-2D projection in a straightforward manner. Given a 3D point to be projected, it needs to find a ray stored in the model that intersects with the point. This can be an exhaustive search if the rays are unstructured, while the 3D-2D projection is an essential process in 3D recovery in handling occlusion and view dependent color consistency analysis as done in space carving[9] for example.

The key idea of this paper is to propose a structured ray-pixel model which focuses on modeling the distribution of the rays in catadioptric systems. For modeling a general distribution of the rays, Grossberg and Nayar utilized a unique ray-surface called caustics [8], to which all the incoming scene rays tangent. In contrast, to realize a compact description and an efficient 3D-2D projection, we exploit an axially-symmetric structure of the rays and employ a virtual camera model to describe such a 1D pixel-to-ray mapping as a relationship between a virtual focal length and a virtual pixel.

Based on our new camera model, we propose two practical catadioptric systems for 3D measurement, one for an underwater active stereo with flat water-proof housings, and the other for a microscale 3D reconstruction with teleidoscopic system. In the first system, projectors are utilized as reverse cameras[10], [11], [12] in order to improve the stereo matching for poorly-textured underwater objects by attaching artificial textures onto the target surface. We show our ray-pixel camera can correctly handle the underwater scene ray geometry and show that our efficient 3D-2D projection realizes a practical 3D capture of underwater objects such as swimming fish.

The second teleidoscopic system has three planar mirrors and a monocentric lens similarly to teleidoscopes. The planar mirrors virtually define multiple viewpoints, and the monocentric lens realizes a high magnification with less blurry and surround view even in closeup imaging. We show our camera model can handle the refractive and reflective projection of the rays and show that the system realizes a 3D shape capture of microscale objects.

2. Related Work

Catadioptric system have been utilized for capturing objects in
a wide range of scales and media, such as capturing objects in water. Several studies have been proposed for handling scattering or absorption[13, 14, 15, 16, 17, 18, 19], refraction at water surface boundary[20, 21, 22] or refraction by housing of underwater camera[23, 24, 25, 26, 27, 28, 29, 30].

3D shape capturing with refraction

As well known, Snell’s law describes the refraction process at the boundary of medium. Therefore, the process of observing underwater scene by a perspective camera via refractive medium such as waterproof housing can be expressed by tracing the light paths by the law of refraction[24, 25, 26]. This is an analytical process, however, its 3D-2D projection is defined as a solution of a 12th-degree polynomial[24] and is time-consuming for dense 3D geometry recovery.

On the other hand, the ray-pixel approach proposed by Grossberg and Nayar describes such projection process as a mapping between them regardless of the intermediate projection process[8]. While this pixel-wise representation has a great flexibility to describe complicated projections[31, 32, 33, 34], 3D-2D projection cannot be provided explicitly. That is, it requires finding a ray in the mapping function that intersects with the 3D point in question.

Therefore, in both analytical and general ray-pixel approaches, 3D-2D projections involve a time-consuming numerical optimization process. Instead of directly using 3D-2D projection, Sedlazeck and Koch propose a virtual camera whose projection centers are on a ray-surface called caustic, and they provide 2D-3D based reprojection error modeling for handing refraction[27], [35].

3D shape capturing with mirrors

While catadioptric system has a wide variety of applications such as omnidirectional observation and panoramic stereo[6, 7, 36, 37], the following reviews studies with multiple flat mirrors.

A fundamental motivation of introducing mirrors in observation system is to increase the number of viewpoints without installing additional cameras for multi-view capture of a target[38, 39, 40, 41, 42, 43]. Takahashi et al.[40] have proposed a kaleidoscopic imaging system and demonstrated a 3D shape reconstruction using multiple reflections. Tagawa et al. [42] have proposed a multi-facet imaging system for that observes a target from an equally distributed virtual cameras reflectance analysis.

These studies can be categorized into two groups: virtual camera approaches or virtual object approaches. The former utilizes the mirrors to define virtual cameras capturing the original object from different directions. The latter considers that the mirrors define virtual objects in the original camera image. While the latter allows modeling the entire light field by a single ray-pixel camera, we follow the former approach to exploit an axially-symmetric structure found in the rays of each virtual view.

3D shape capturing with monocentric lens

In the context of imaging system, the monocentric lens is often used to obtain a wide field-of-view[44], in particular for endoscopes, as a short focal length lens. The monocentric lens, however, has additional useful characteristics: its symmetric structure and magnifying power.

By exploiting the symmetric structure of the monocentric lens, Cossairt et al. [45] have proposed a camera array which captures a same scene through a single monocentric lens so that the images can be stitched into a single high-resolution image. Similarly, Dansereau et al. [46] have proposed a lightfield camera which captures omnidirectional lightfield images through a single monocentric lens with a camera orbiting around it.

A typical use of the monocentric lens as a magnifier can be found in the Leeuwenhoek’s microscope in the 17th century. It utilizes a single monocentric lens and realized over 100× magnification[47].

3. 3D Reconstruction through Planar Refraction

Projection paths to a perspective camera via a flat housing is shown in Fig.1-(a). Unlike the case in the air, the ray in water is not described as straight line from a common projection center because of the refraction. This fact indicates that conventional triangulation methods in the air cannot return the correct location and that refraction effects cannot simply be modeled by radial distortions.

The key idea of our approach on modeling such ray distribution is to exploit the axially-symmetric structure of the ray distribution by the flat housing.

3.1 Measurement through Planar Refraction

Measurement model through planar refraction is shown in Fig.2. A pinhole camera $C$ at $o_0$ observes the underwater scene via flat housing. A point $p_w$ in water is projected to $p_p$ and $o_0$ along the segments $\ell_a-\ell_q-\ell_u$. ©2016, Elsevier [28].
Assuming that the two surfaces $S_a$ and $S_b$ of the housing are flat and parallel, the segments $\ell_a-\ell_b-\ell_a$ and housing normal $n$ are always on a single plane-of-refraction and have axially symmetric structure around the normal $n$. Hence, we employ the axial camera concept proposed by Agrawal et al. [24] to simplify this model without loss of generality.

Consider a virtual camera $C_v$ such that its projection center is placed at $o_0$ and its direction of optical axis is identical to the normal $n$ of the flat housing (Fig. 3). If the pose of camera $C$ w.r.t. the housing is calibrated beforehand, the mapping from a pixel of virtual camera $C_v$ to the corresponding pixel of $C$ is given by a homography matrix $H_C$ derived from the intrinsic and the pose of $C$, where we simply assume the cameras $C$ and $C_v$ share the same intrinsic parameter $A$ and assume the radial distortion caused by the internal lens of the camera is already rectified.

Therefore, instead of $C$, we can use $C_v$ which has a radially symmetric structure of refractive path around the $Z$-axis without loss of generality. In the coordinate of $C_v$, any continuous segments $\ell_a-\ell_b-\ell_a$ are observed as a single line on the image plane, because the segments are on a single plane-of-refraction. Hence we employ the $(r, z)$ coordinate system hereafter.

Let $r_o$ and $z_o$ denote the $r$ and $z$ elements of vector $o$ in general. For example, point $p_p$ is described as $p_p = (r_p, z_p)$. Also let $v_X = (r_X, z_X)$ denote the direction vector of line $\ell_X$ towards the water from the camera, where $X$ is each medium (Fig. 3).

The light path $\ell_a-\ell_b-\ell_a$ follows Snell’s law which is expressed as $\mu_a r_v a = \mu_b r_v b = \mu_a r_v a$, where $\mu_a$, $\mu_b$, and $\mu_a$ are the refractive indices of the air, housing and water. Using Snell’s law, the light path $\ell_a-\ell_b-\ell_a$ is given as

$$v_a = \left(\frac{p_o}{\sqrt{r_o^2 + f_c^2}}, \frac{f_i}{\sqrt{r_o^2 + f_c^2}}\right)^T,$$  \hspace{1cm} (1)

$$p_a = \frac{d_a}{z_v} v_a,$$ \hspace{1cm} (2)

$$v_g = \left(\frac{\mu_a}{\mu_g} r_v g, \sqrt{1 - r_v g^2}\right)^T,$$ \hspace{1cm} (3)

$$p_g = p_o + \frac{d_g}{z_v} v_g,$$ \hspace{1cm} (4)

$$v_w = \left(\frac{\mu_g}{\mu_w} r_v w, \sqrt{1 - r_v w^2}\right)^T,$$ \hspace{1cm} (5)

$$p_w = p_g + \frac{z_{w o} - z_{v o}}{z_v} v_w.$$ \hspace{1cm} (6)

where $A_i$ is the focal length of the camera. These equations allow computing direction of a ray in water $v_w$ and a position $p_w$, given a pixel $p_p$ on the image plane. Similarly, computing $p_p$ from $v_w$ can be done by applying Snell’s law inversely because $v_w$ can be derived from $v_o$ by Eqs. (3) and (5).

This indicates that the inverse process of Eqs. (1)-(5) requires the direction $v_o$, but in the case of 3D-2D projection, only given is 3D point $p_o$. It requires solving a 12th degree equation and becomes a time-consuming process[24]. To realize a practical projection through planar refraction, we employ a new camera model that exploits the radial structure of the rays in $C_v$, based on the ray-pixel camera model[8] and Eqs. (1)-(5).

3.2 Planar Refraction Ray-Pixel Camera Model

Let us consider the representation of scene rays $\ell_o$. On the assumption of the straightness of light, a scene ray $\ell_o$ is simply described by a set of a starting point and a direction. That is, when we set an arbitrary starting point $o_0$ along the $\ell_o$, the geometric representation of the ray $\ell_o$ is defined by the set $(o_0, v_0)$, where the point $o_0$ is generally $\mathbb{R}^3$ and the direction $v_0$ is $\mathbb{R}^2$ by two angles similarly to the geometric part of the light field of plenoptic camera by Adelson and Bergen[48]. In addition, the starting point $o_0$ can be on a surface such as a unique caustic formed by the rays $\ell_o$[35] or a 2D plane in front of the camera.

Fig. 5 shows our new virtual camera model called planar refraction ray-pixel camera model. Notice that each extended rays of $\ell_o$ intersects a common axis whose direction is identical to the housing normal $n$. This fact indicates that we can compactly represent the ray $\ell_o$ by the position of the intersection point on the axis.

In particular, we introduce a virtual camera whose image plane is on the outer housing surface $S_a$ associated with a pixel-wise virtual focal length $f_i(p_p) \in \mathbb{R}$ as follows.

© 2013 Information Processing Society of Japan
A pixel-ray mapping for generalized ray-pixel model. A pixel \( p_g \) is associated with a point on ray-surface \( o_g \in \mathbb{R}^3 \) and the direction \( n_g \in \mathbb{R}^2 \).

<table>
<thead>
<tr>
<th>pixel</th>
<th>( p_{g0} )</th>
<th>( p_{g1} )</th>
<th>( \ldots )</th>
<th>( p_{gN} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ray</td>
<td>( (o_{g0}, n_{g0}) )</td>
<td>( (o_{g1}, n_{g1}) )</td>
<td>( \ldots )</td>
<td>( (o_{gN}, n_{gN}) )</td>
</tr>
</tbody>
</table>

In addition, it is obvious that in the 1D ordered structure easily provide the differential \( \frac{df}{df} \) for the mapping.

\[
\begin{align*}
\text{Table 2} & \quad \text{A pixel-ray mapping of our planar refraction ray-pixel camera model. A pixel } p_g \text{ in } (r, z) \text{ coordinate in } C \text{ is associated with a pixel-wise focal length } f(r_p) \text{ in } \mathbb{R}. \text{ The 1D ordered structure easily provide the differential } \frac{df}{df} \text{ for the mapping.} \\
| \text{pixel} | \text{ } p_{g0} | \text{ } p_{g1} | \text{ } \ldots | \text{ } p_{gN} | \\
| \text{ray} | f(r_{p0}) | f(r_{p1}) | \text{ } \ldots | \text{ } f(r_{pN}) | \\

\text{\( f_p \)} & \text{\( f_{p1} \)} & \text{\( \ldots \)} & \text{\( f_{pN} \)} \\

\\end{align*}
\]

- The virtual image screen coincides with the housing surface \( S_g \). The virtual pixel \( p_v \) is associated with a pixel \( p_g \) in \( C \) by Eq. (4) and the homography \( H \) between \( C \) and \( C_v \).
- The virtual optical axis (Z-axis) is identical to the housing normal \( n \).
- The pixel-wise projection center \( o_{p_v} \) of \( p_g \) is defined simply by connecting \( \ell_\omega \) to the virtual optical axis (Fig. 5, the green straight line). The distance between \( o_{p_v} \) and \( S_g \) is denoted as the pixel-wise focal length \( f_\omega(p_g) \).

From the radially symmetric structure of the ray distribution, the pixel-wise focal length \( f_\omega(p_g) \) can be expressed as a function of the radial distance \( r_{p_g} \) of \( p_g \) from the optical axis without loss of generality:

\[ f_\omega(p_g) = f_\omega(r_{p_g}). \quad (7) \]

In addition, it is obvious that \( f_\omega(r_{p_g}) \) is a monotonically increasing function of \( r_{p_g} \) from Fig. 5.

By the definition, the pixel-wise virtual focal length \( f_\omega(r_{p_g}) \) is derived as follows,

\[
\begin{align*}
\text{\( o_{p_v} = \ell_\omega + n_{p_v} \).} \\
\text{where,} \\
\text{Eq. (9), Eq. (11), and Eq. (13),}
\end{align*}
\]

These representation leads to the compact description of the rays \( \ell_\omega \) as a ray-pixel camera model.

3.2.1 The Ray-Pixel Mapping

As shown in Table 1, the generalized ray-pixel camera model such as [8] associates fully-described scene rays in \( \mathbb{R}^3 \) by the mapping of ray-surface such as caustic. Compared to such generalized ray-pixel camera model, our planar refraction ray-pixel camera model stores focal lengths for each radial distance in a 1D array as shown in Table 2.

In addition, our mapping also has the derivative \( \frac{df}{df} \) (the last row of Table 2), since \( f(r_{p_g}) \) is a smooth and monotonic function. This \( \frac{df}{df} \) is the key to realize our efficient 3D-2D projection (Section 3.2.3).

In practice the mapping table can be obtained as follows.

- Sample the radial distance \( r_{p_g} \) of virtual pixels at equal intervals so that the interval is smaller than 1px in the image plane of corresponding real camera \( C \).
- Compute the pixel-wise virtual focal length \( f_\omega \) given by \( r_{p_g} \).

3.2.2 2D-3D Projection

For a pixel \( p_g \) on virtual image plane \( S_g \), 2D-3D projection is trivially defined like a perspective projection from virtual focus \( o_{p_v} \) which corresponds to \( r_{p_g} \) by Table 2. That is, the 2D-3D projection is represented by using virtual focus \( o_{p_v} = (0, -f_\omega(r_{p_g}))^T \) as

\[
\ell_\omega : o_{p_v} + t_\omega n_w = o_{p_v} + t_\omega \parallel p_g - o_{p_v} \parallel^1, \quad (10)
\]

where \( t_\omega \) is the \( p_g - p_0 \) distance.

3.2.3 3D-2D projection

The 3D-2D projection process can be realized by searching the mapping table for the ray intersecting with the point \( p_0 \) to be projected. As shown in Fig. 6, let us hypothesize a ray \( \ell_v \) from \( p_0 \) which intersects with the axis at \( \lambda_0(0, -\lambda_0) \). If the ray \( \ell_v \) is identical to \( \ell_\omega \) stored in the mapping table, we can conclude that the hypothesized ray was the correct projection direction from \( p_0 \). Otherwise, we can refine the hypothesis in the following two approaches.

By Gauss-Newton method

Utilizing the ray-pixel mapping and the derivative of \( f_\omega \) shown in Table 2, we can solve the 3D-2D projection by Gauss-Newton method. When the ray \( \ell_4 \) is identical to \( \ell_\omega \), following equation is satisfied,

\[
G(r_{p_g}) = f_\omega(r_{p_g}) - \lambda_4(r_{p_g}), \quad (11)
\]

where, \( G(r_{p_g}) \) is a monotonic function and hence we can compute \( r_{p_g} \) by iterative process as

\[
\begin{align*}
\lambda_4(r_{p_g}) &= \frac{r_{p_g}^{(k+1)}}{r_{p_g}} - G(r_{p_g}) \frac{\Delta r_{p_g}}{\Delta G} = \frac{f_\omega(r_{p_g}) - \lambda_4(r_{p_g})}{f_\omega(r_{p_g}) - \lambda_4(r_{p_g})},
\end{align*}
\]

where \( \lambda_4^{(k)} \) denotes the element with the notation identifier \( X \), in this case, \( \lambda_4^{(k)} \) denotes \( r \) element of the \( X \)-th iteration. The derivative \( \lambda_4^{(k)} \) is obtained by Table 2 computed beforehand. \( \lambda_4(0, -\lambda_4) \) and its derivative are functions of \( r_{p_g}^{(k)} \),

\[
\lambda_4 = \frac{r_{p_g}(G(r_{p_g}) - \lambda_4(r_{p_g}))}{G(r_{p_g}) - \lambda_4(r_{p_g})}, \quad \lambda_4^{(k+1)} = \frac{r_{p_g}^{(k)} - \lambda_4^{(k)}}{r_{p_g}^{(k)} - \lambda_4^{(k)}}, \quad (13)
\]

The global convergence of this process is ensured from the characteristics of \( G(r_{p_g}) \). From Eq. (9), Eq. (11), and Eq. (13), the form of \( G(r_{p_g}) \) becomes the same as Eq. (6) and known as a twice continuously differentiable, increasing, and convex function having a zero[28].

© 2013 Information Processing Society of Japan
By Recurrence Relation

As shown in Fig. 6, we can also obtain the corresponding ray without using the differential \( f'_r \) as follows:

\[
\begin{align*}
r_{p_r}^{(k)} &= \frac{r_{p_r}^{(k)} \Lambda_v^{(k)}}{z_{p_r} + \Lambda_v^{(k)}}, \quad \theta^{(k+1)} = f_\theta(r_{p_r}^{(k)}) \\
&\Rightarrow r_{p_r}^{(k+1)} = \frac{r_{p_r} f_\theta(r_{p_r}^{(k)})}{z_{p_r} + f_\theta(r_{p_r}^{(k)})}.
\end{align*}
\]  

(14)

(15)

The method guarantees global convergence because of the following reasons. From Eq. (9), \( f_\theta \) is a monotonic function of \( r_{p_r} \) satisfying

\[
\theta^{(k+1)} > \theta^{(k)} \Leftrightarrow f_\theta^{(k+1)} > f_\theta^{(k)}.
\]

(16)

Therefore, from Eq. (15),

\[
\theta^{(0)} < \theta^{(1)} \Rightarrow r_{p_r}^{(0)} < r_{p_r} \frac{r_{p_r} f_\theta(r_{p_r}^{(1)})}{z_{p_r} + f_\theta(r_{p_r}^{(1)})} < r_{p_r}.
\]

(17)

That is, the following monotonicity is satisfied:

\[
r_{p_r}^{(1)} > r_{p_r}^{(0)} \Rightarrow r_{p_r}^{(k+1)} > r_{p_r}^{(k)} \quad r_{p_r}^{(1)} < r_{p_r}^{(0)} \Rightarrow r_{p_r}^{(k+1)} < r_{p_r}^{(k)}.
\]

(18)

As a result, the iteration converges to

\[
r_{p_r}^{(k)} = r_{p_r}, \quad \theta^{(k+1)} = f_\theta^{(k+1)} = f_\theta^{(k)}.
\]

(19)

Notice that a reasonable initial guess for \( \theta^{(0)} \) can be given by projecting the point in water to the projection center \( \omega_0 \) of the real camera \( C \) without considering the refraction.

3.2.4 Efficiency of 3D-2D Projection of a Planar Refractive Ray-Pixel Camera

This section evaluates our 3D-2D projection computation in terms of efficiency. Notice that the proposed model does not improve the projection accuracy in comparison with the conventional analytical approach by definition.

Rate of Convergence

To evaluate the rate of convergence of our iterative methods for the 3D-2D projection in Section 3.2.3, Fig. 8 shows the projection error \( E_p \) against the number of iterations \( k \). By using a synthesized data set, the reprojection error is defined as

\[
E_p = |P'(\tilde{r}_{p_r}) - P'(r_{p_r}^{(k)})|,
\]

(20)

where \( \tilde{r}_{p_r} \) is the ground-truth and \( r_{p_r}^{(k)} \) is the value returned by the algorithm at the \( k \)-th iteration in \( C_r \). \( P'(r_{p_r}^{(k)}) \) denotes the pixel position in the original image of \( C' \) corresponding to \( r_{p_r}^{(k)} \) in \( C_r \). Notice that \( P'(\cdot) \) is employed only for evaluating \( E_p \) in pixels, and is not required for the 3D-2D projection to \( C_r \).

From these results, we can observe that (1) the rates of convergence of the recurrence relation and the Newton-based ones are linear and quadratic respectively, and (2) our Newton-based algorithm with iteration \( k = 3 \) achieve a sub-pixel accuracy.

Computational Efficiency

Table 3 reports computational costs of our methods computing up to the subpixel accuracy and those of the state-of-the-art solving a 12th degree of equation analytically[24]. They are the average values of 6400 3D-2D projections run in Matlab on an Intel Core-i5 2.5GHz PC.

From these results, we can conclude that our method runs much faster than the analytical method while achieving the sub-pixel accuracy. In other words, our method achieves a comparable accuracy as other methods in practical image-based analysis.

The linear and quadratic rates of convergence shown in Fig. 8 do not immediately indicate that the Newton-based method is always better. That is because their computation costs for one iteration step is different as shown in Table 3. One practical option is that combining these two methods by updating with the recurrence relation method first and then by switching to the Newton-based one for fine tuning, because the computation cost of recurrence relation method is smaller but linear convergence requires many iteration steps in larger angle of refraction as shown in Fig. 8.

3.3 Underwater 3D Reconstruction

This section introduces our underwater active stereo system in which both cameras and projectors are modeled by our planar refraction ray-pixel camera model.

The main difficulty in underwater stereo is its depth-dependent distortion on the image plane caused by refraction. This distortion deforms the epipolar line and invalidates stereo methods which provide dense 3D shape by template matching.

We show that our model can realize an efficient implementation of underwater active stereo utilizing the 3D-2D projection introduced in the last section.

Table 3. Average computational costs of single 3D-2D projections.

<table>
<thead>
<tr>
<th>Method</th>
<th>Analytical[24]</th>
<th>By Recurrence</th>
<th>By Newton</th>
</tr>
</thead>
<tbody>
<tr>
<td>Runtime</td>
<td>1.39 msec</td>
<td>0.14 msec</td>
<td>0.27 msec</td>
</tr>
</tbody>
</table>

3.3.1 Calibration

Single Camera Calibration in Water

Single underwater camera is calibrated based on a conventional method using a reference object of known geometry (such as a chessboard) in water\[24\], [49\]. Once the parameters describing the refraction process are obtained, we can build the ray-pixel mapping table as shown in Table 2.

Linear Extrinsics of Underwater Cameras

Relative posture of planar refraction ray-pixel cameras can be estimated from corresponding pairs of projections of scene points.Suppose each of the cameras are calibrated as a planar refraction ray-pixel camera beforehand. Given a pair of corresponding points in two such cameras, we can back-project the rays in water as shown in Fig. 9. Since these two lines intersect, the following coplanarity constraint holds.

\[ e_x^\top \left( (R_a p_a + t - a_p) \times (R_b p_b) \right) = 0. \] (21)

By rewriting this in a bilinear form, we have

\[
\begin{pmatrix}
  x_a \\ y_a \\ z_a \\
- \frac{1}{f_a} y_a \\ \frac{1}{f_a} x_a \\ 0
\end{pmatrix}
\begin{pmatrix}
  (t_x R) \\ R \\ 0
\end{pmatrix}
\begin{pmatrix}
  x_b \\ y_b \\ z_b \\
- \frac{1}{f_b} y_b \\ \frac{1}{f_b} x_b \\ 0
\end{pmatrix} = 0,
\] (22)

where \([X]_b\) denotes the 3×3 skew-symmetric matrix defined by a 3D vector \(X\). Now we can rewrite the equation in a Plücker forms as

\[ \ell_a E_1 \ell_b = 0. \] (23)

Since \(\ell_a\) and \(\ell_b\) are given by each of the corresponding pairs, we can linearly estimate 17 unknown (9 for \([X]_2 R\), 8 for \(R_{3,3}\)) elements of \(E_1\) up to a scale using 16 or more corresponding pairs for Eq. (23).

Notice that Eq. (23) has a trivial solution \(R_a p_a + t - a_p = 0\). This indicate that two virtual optical centers coincide. As mentioned in [50],

\[ E_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \] (24)

exists iff \(f_a = f_b\) and these virtual focal lengths are changed for each pixel.

Calibration of Projector-Camera System in Water

This section describes our underwater projector-camera calibration as originally presented in [51]. Same as the case of single underwater camera calibration, what required for the parameters of underwater projector is the corresponding pairs of a 3D scene point and its projection \((p_c, p_v)\). However, since the position of 3D point \(p_c\) projected by the underwater projector cannot be obtained directly, we estimate \(p_c\) by an underwater camera calibrated beforehand.

As illustrated in Fig. 10, there are underwater projector \(C\), underwater camera \(C^\prime\), and reference plane \(\Pi\) in water. Let us assume underwater camera \(C^\prime\) is calibrated as \(C^\prime\) beforehand.

Suppose reference points of known geometry are printed on the plane \(\Pi\) as \(p_{1i} = (x_{1i}, y_{1i}, 0)^\top\), for the pose estimation of camera \(C^\prime\) w.r.t. \(\Pi\) by capturing pattern \(p_{1i}\). On the other hand, unknown points are projected on the plane \(\Pi\) as \(p_{vi} = (x_{vi}, y_{vi}, 0)^\top\) by the projector \(C\). Our goal is to estimate \(p_v\) and calibrate underwater projector as \(C^\prime\).

To sum up, our calibration consists of the following steps.

**Step 1.** Camera \(C^\prime\) pose estimation w.r.t. \(\Pi\) by capturing pattern \(p_{1i}\).

**Step 2.** Estimation of 3D geometry of a pattern \(p_v\) projected by \(C\) on \(\Pi\) using \(C^\prime\).

**Step 3.** Projector calibration of \(C^\prime\) and its pose estimation w.r.t. \(\Pi\) using 2D-3D correspondence of pattern \(p_v\).

**Step 1. Pose Estimation of Camera using Planner Pattern in Water**

The camera pose \(R_{1i}'\) and \(t_{1i}'\) w.r.t. \(\Pi\) can be estimated using the flat refraction constraint[24]. That is, the direction \(e_{w}^{(1i)}\) of the ray \(\ell_{a}'\) as a backprojection is identical to the vector from the incident point \(p_{c}^{(1i)}\) to \(p_{c}^{(1i)}\), where the known point \(p_{1i} = (x_{1i}, y_{1i}, 0)^\top\) is described as \(p_{c}^{(1i)}\) in the coordinate system of camera \(C^\prime\) (Fig. 10).

\[ e_{w}^{(1i)} \times \left( (R_{1i}' p_{1i}' + t_{1i}' - p_{c}^{(1i)}) \right) = 0 \] (25)

\[ \Rightarrow \begin{pmatrix} x_{p_{c}^{(1i)}} & y_{p_{c}^{(1i)}} & 1 \end{pmatrix}^{\top} \begin{pmatrix} t_{1i}' \\ t_{1i}' \\ t_{1i}' \end{pmatrix} = e_{w}^{(1i)} \times p_{c}^{(1i)}. \]
where \( r_{Xi} \) denotes the \( i \)th column vector of \( R_X \) with notation identifier \( X \). Since this equation provides three constraints for 9 unknowns \( r_{c1}, r_{c2}, \) and \( t_{c} \), we can solve this system of equations linearly by using at least three points. Once \( r_{c1} \) and \( r_{c2} \) are obtained, \( r_{c3} \) is given by their cross product.

**Step 2. Estimation of 3D Geometry of Projected Pattern**

The goal here is to estimate \( p_v = (x_p, y_p, 0)^T \) from its projection \( p^v \) in the camera \( C_v \) image in order to establish 2D-3D correspondences between 2D projector pixels \( p^v \) and 3D points \( p_v \) on \( \Pi \).

Since \( p_v \) is on \( t'_{w} \), we can represent its 3D position to \( C_v' \) by 2D-3D projection (mentioned in the section 3.2.2) with a scale parameter \( t_{w} \) as

\[
\begin{align*}
p_v &= R_v^w (p^v_w - t'_{w}) \\
&= R_v^w (t_{w} p^v_v + a^v_{p_v} - t'_{w}).
\end{align*}
\]

Since we know \( z \) of \( p_v \) is equal to 0, it is trivial to determine the unknowns \( x_p, y_p, \) and \( t_{w} \).

**Step 3. Calibration of Projector using 2D-3D Correspondences**

Given a set of correspondences between 2D projector pixels and its projection on the plane \( \Pi \) \( (p^v, p_v) \) in the previous step, the pose of the real projector \( R_v \) and \( t_v \) w.r.t. \( \Pi \), and its housing parameters can be calibrated by the conventional method[24]. Once obtained these parameters, we can build a table representing the virtual focal length as done for underwater camera.

Notice that the 3D points \( p_v \) are not necessarily from a single \( \Pi \). In fact, by capturing the panel \( \Pi \) with different poses in water, they can cover a larger area of the scene and contribute to improve the accuracy and robustness of the parameter estimation as pointed out in [24].

### 3.3.2 Triangulation by Planar Refraction Ray-Pixel Cameras

Towards the 3D reconstruction, we introduce the triangulation method by multiple planar refraction ray-pixel cameras. The basic idea is to form triangle as with the case of linear extrinsic calibration of underwater cameras in Section 3.3.1.

Instead of using the plane-of-refraction constraint, the triangulation process utilizes the two way of a light path description. That is, in Fig. 9, outgoing ray direction \( v_o = (r_{w_o}, z_{w_o})^T \) of \( t_{w} \) is also described as \( a_{p_v}(0, -f_{p} b_{p_v} r_{p_v}, z_{p_v}) \) direction.

Therefore, the following relationship (flat refraction constraint) holds.

\[
\begin{align*}
v_o \times (p_v - a_{p_v}) &= 0 \\
\Leftrightarrow r_{p_v} - \frac{r_{w_o}}{z_{w_o}} z_{p_v} = \frac{r_{w_o}}{z_{w_o}} f_{w_o}.
\end{align*}
\]

For multiple viewpoints, we rewrite this in \((x, y, z)\) expression as

\[
\begin{align*}
\begin{bmatrix} 1 & 0 & -\frac{r_{w_o}}{z_{w_o}} \\
1 & 0 & \frac{z_{w_o}}{z_{p_v}} \\
0 & 1 & z_{p_v}
\end{bmatrix}
\begin{bmatrix} x_p \\
y_p \\
z_{p_v}
\end{bmatrix} = \frac{f_{p}}{z_{w_o}} \begin{bmatrix} x_{w_o} \\
y_{w_o} \\
z_{w_o}
\end{bmatrix},
\end{align*}
\]

\[
\Leftrightarrow A_0 p_v = b_0,
\]

and for \( C_v' \).
corners are different from the corners used for calibration step and the distance between the nearest and the farthest panels was roughly 400 mm.

The blue points are the ground truth calculated by the reference cameras in the air (Camera 3, Camera 4). The cyan ones are the points by our underwater camera system (Camera 1, Camera 2), and the average error of these 200 points was 2.43 mm. The red ones are points calculated by assuming the perspective projection without refraction, and its average error was 31.11 mm. The result shows our calibration of underwater cameras obviously provides a better result of 3D measurement quantitatively and qualitatively.

**Evaluation of Underwater Projector and Camera**

Figs. 12(b) and (c) show the estimated chess corner positions on three panels at every 200 mm using the Structured Lighting[53], [54] conventional Gray code pattern. The yellow points and the plane are the ground truth calculated by the reference cameras in the air (Camera 3, Camera 4). The blue dots denote the points by our underwater projector and camera system (Camera 1, Projector). The red dots denote the points by assuming the perspective projection without refraction. These figures qualitatively visualize that our method better reconstructs the 3D points of different distances from the camera and the projector.

The quality of the calibration is assessed by measuring the distance from the ground truth plane to the estimated 3D points. The average errors of the blue points on the three panels were, from near to far, 1.90 mm, 1.59 mm, and 4.01 mm respectively. Those of the red ones were 34.36 mm, 9.53 mm, and 89.06 mm respectively.

From these results we can conclude that our method realized a practical underwater projector-camera calibration in a reasonable accuracy for a wide range of distance from the cameras.

### 3.3.4 Dynamic 3D Capture of Swimming Fish

As shown in Fig. 13, we used five underwater cameras and two underwater projectors, and captured a swimming goldfish. Each of the projector casts a pattern in different color channels (red and blue) for avoiding interference. The system ran at 15 fps in recording, and took about 30 sec per frame to reconstruct the 3D shape by our underwater space carving using 4 mm voxel resolution.

The three columns in the left of Fig. 14 show the captured images, and the three columns in the right show rendered images of the reconstructed 3D shapes by our method and by the conventional space carving with perspective projection[9]. As the left column of the rendered images shows, we can virtually observe the object appearance even from the viewpoint where the real camera does not exist (left column of Rendered Images), and the conventional space carving cannot produce a comparable result since it ignores refraction (right most column). ©2016, Elsevier [28].

**4. Catadioptric Ray-Pixel Camera Model**

In this section, we introduce the idea and description of our new ray-pixel camera model for catadioptric system with a front lens and planar mirrors.

#### 4.1 Monocentric Lens

Monocentric lens is a spherical and homogeneous optical lens which often has a high refraction index. As shown in Fig. 15, rays from a single point (blue) towards a monocentric lens diverge at a wide angle on the other side.

This indicates that a perspective camera located at the point can use the monocentric lens as a conversion lens to obtain a wider field-of-view. It approximately has a short focal length as a thick lens, however, it does not have a single focus strictly[45], as illustrated as the caustic by the red lines in Fig. 15. In order to model such rays efficiently, we first model the ray-pixel mapping through the monocentric lens.
Fig. 15 Refraction by monocentric lens. The blue, green, and red lines indicate incident, refracted, and emergent rays through a monocentric lens respectively. Notice that the emergent rays have a wider field-of-view than that of the incident rays, while they form a caustic.

Fig. 16 Measurement through monocentric lens. The segments of projection paths $\ell_0, \ell_0', \ell_0''$ have an axially symmetric structure around the axis which directs to the monocentric lens center $\mathbf{0}_0$.

4.1.1 Measurement through a Monocentric Lens

Suppose a perspective camera whose camera center is located at $\mathbf{0}_0$ observes the scene through a monocentric lens located at $\mathbf{0}_D$ as illustrated in Fig. 15. Here the perspective camera can be assumed to be directed to the the center of the monocentric lens $\mathbf{0}_D$ without loss of generality, so we can calibrate the rotation to align the optical axis of the camera to the line $\mathbf{0}_D - \mathbf{0}_0$ as described later. We denote this normalized camera by $C_0$.

Obviously, the rays back-projected through pixels of $C_0$ have an axially symmetric structure around the optical axis, and an incident ray $\ell_0$ through a pixel $\mathbf{p}$, and its refraction $\ell_0'$ and emergent ray $\ell_0''$ are on a single plane-of-refraction[55]. Therefore, we can describe the ray through a single pixel by a 2D $(r, z)^\top$ coordinate system centered at $\mathbf{0}_0$ as shown in Fig. 15.

Consider the projection path $\ell_0, \ell_0', \ell_0''$ from $\mathbf{0}_0$ through the point $\mathbf{p}$ in Fig. 15. The incident point $\mathbf{p}$ on the sphere is described as a function of $\theta_p = \tan^{-1}(l_p/f_p)$ as

$$z_{pi} = \frac{d_D - \sqrt{d_D^2 - (1 + \tan^2\theta_p)(D_0^2 - r_{pi}^2)}}{1 + \tan^2\theta_p},$$

$$r_{pi} = z_{pi}\tan\theta_p.$$

The refraction angle $\theta_p$ is then given by Snell’s law with assuming the refractive indices of the air $\mu_a$ and the lens $\mu_y$ are known:

$$\mu_a\sin\theta_p = \mu_y\sin(\theta_p + \theta_D),$$

$$\Leftrightarrow \sin\theta_p = \frac{\mu_y}{\mu_a} \frac{d_p l_{pi}}{r_{D} \sqrt{z_{pi}^2 + r_{pi}^2}}$$

The outgoing point $\mathbf{p}_o$ is derived as the intersection of the path $\ell_y$ and sphere boundary. It is also obtained as the mirror of the point $\mathbf{p}_i$ because of the symmetrical relationship between incident path $\ell_i$ and outgoing point $\ell_o$ w.r.t. the monocentric lens.

As shown in Fig. 16, $\ell_i$ and $\ell_o$ are in line symmetry to the line that is perpendicular to the vector $\mathbf{n}_o$ through the point $\mathbf{0}_0$. That is, given the point $\mathbf{p}_i$, its reflection $\mathbf{p}_o$ is described as

$$\mathbf{p}_o = H_p \mathbf{p}_i + \mathbf{t}_p,$$

$$\Leftrightarrow \mathbf{p}_o = (I - 2n_o n_o^\top) \mathbf{p}_i + 2(a_o^\top n_o) n_o,$$

where $H_p$ is the Householder matrix and $\mathbf{t}_p$ denotes the center of reflection. Besides, the direction $\mathbf{n}_p$ is given as

$$\mathbf{n}_p = \begin{pmatrix} \cos(\frac{\pi}{2} + \theta_D) - \theta_p \\ \sin(\frac{\pi}{2} + \theta_D) - \theta_p \\ \cos(\theta_D - \theta_y) - \theta_p \end{pmatrix}.$$  

Therefore, $H_p$ and $\mathbf{t}_p$ are rewrite to

$$H_p = \begin{pmatrix} \cos(2\theta_D - \theta_y) \\ \sin(2\theta_D - \theta_y) \\ -\cos(2\theta_D - \theta_y) \end{pmatrix},$$

$$\mathbf{t}_p = \begin{pmatrix} -d_D\sin2(\theta_D - \theta_y) \\ d_D\cos2(\theta_D - \theta_y) + d_D \end{pmatrix}.$$  

The direction $\mathbf{v}_o$ is also given with Householder matrix as

$$\mathbf{v}_o = H_p \mathbf{v}_i.$$  

As a result, the intersection $\mathbf{o}_{i_0} = (0, f_{i_0})^\top$ of the ray $\ell_i$ and the optical axis is given as follows:

$$\mathbf{o}_{i_0} = \mathbf{u}_0 \mathbf{v}_i + \mathbf{p}_{i_0},$$

$$\begin{pmatrix} 0 \\ f_{i_0} \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{u_0} \\ \mathbf{z}_{p_{i_0}} \end{pmatrix} + \begin{pmatrix} \mathbf{r}_p \\ \mathbf{z}_{p_{i_0}} \end{pmatrix},$$

$$\Leftrightarrow \mathbf{f}_{i_0} = \mathbf{z}_{p_{i_0}} + -\mathbf{e}_{u_0}^{-1} \mathbf{r}_p.$$  

The inverse process is similarly obtained for a given path $\ell_o$. The analytical 3D-2D projection, however, requires solving a 10th-degree equation[55]. The next section introduces a ray-pixel camera which exploits the axial symmetric structure of the rays to provide an efficient numerical 3D-2D projection.

4.2 Spherical Refraction Ray-Pixel Camera Model

Up to this point, we observed that the rays emitted from a perspective camera through a monocentric lens show an axially symmetric structure around the line from the camera center to the lens center. As illustrated in Fig. 15, given a pixel $\mathbf{p}_i$ by specifying $\theta_p$, the corresponding ray $\ell_i$ can be determined uniquely.

This section introduces our spherical refraction ray-pixel camera model by representing the ray-pixel mapping as follows.

Fig. 17 illustrates the light path from a pixel $\mathbf{p}_i$ on the plane-of-refraction. Suppose the emergent ray $\ell_y$ intersects with the optical
axis at \( \alpha_i = (0, f_q) \) with angle \( \theta_q \). Obviously, changing the position of \( p_p \) in \( r \) direction results in changing the corresponding \( \ell_q \), i.e., \( (\ell_q, \theta_q) \) pair.

That is, since the mapping between the pixel \( p_p \) and the ray \( \ell_q \) is bijective because of the reversibility of light, representing the light paths from the pixels in \( r \) space is identical to knowing all possible \((f_q, \theta_q)\) pairs. In other words, the system of Fig. 17 as a whole can be seen as a pixel-wise varifocal camera which changes the focal length \( f_q \) for each virtual pixel parameterized by \( \theta_q \). In fact, the mapping \( \theta_q \mapsto f_q \) is a monotonic function due to the spherical structure of the lens.

Table 4 shows our ray-pixel mapping. Due to the spherical structure of the lens, this is a discretization of the monotonic function \( \theta_q \mapsto f_q \) by \( \theta_q \). In practice, we sample \( \theta_q \) so that their interval results in a sub-pixel sampling in the original image domain. Notice that the derivative \( \frac{df_q}{d\theta_q} \) is also stored for our numerical 3D-2D projection as described later.

### 4.2.1 2D-3D Projection

Given a virtual pixel \( \theta_q \), the corresponding light path \( \ell_q \) is simply given by

\[
\ell_q : \alpha_i + t_q v_o = \alpha_i + t_q \begin{pmatrix} -\sin(\theta_q) \\ \cos(\theta_q) \end{pmatrix},
\]

where the parameter \( t_q \) represents the depth. The mapping between \( \theta_q \) and the camera pixel can be provided by the measurement model in Section 4.1, which can also be stored in the ray-pixel mapping table (Table 4) in practice.

### 4.2.2 3D-2D Projection

Instead of solving a 10th-degree equation[55] analytically, this section introduces a numerical 3D-2D projection using our ray-pixel mapping table (Table 4).

Similar to the case of planar refraction ray-pixel cameras in Section 3.2.3, the key idea of our numerical 3D-2D projection is to hypothesize a projection line \( \ell_q \) in Fig. 17 first, and check if it intersects with the optical axis at the identical virtual focal length stored in the ray-pixel mapping table. If the focal lengths are not identical, then \( \ell_q \), i.e., the virtual pixel \( \theta_q \) equivalently, is refined to minimize the difference.

We can formulate this process as a Gauss-Newton optimization as follows.

### Gauss-Newton Method

As shown in Fig. 17, let us consider the ray \( \ell_q \) from \( p_w \) which intersects the axis at \( \Lambda_q(0, \lambda_q) \) with an angle \( \theta_q \), i.e., by hypothesizing that the 3D point is projected to a virtual pixel \( \lambda_q \), we can compute the intersection \( \Lambda_q(\theta_q) = (0, \lambda_q(\theta_q)) \) of \( \ell_q \) and the optical axis. If \( \lambda_q \) is equal to the virtual focal length \( f_q(\theta_q) \) stored in the ray-pixel mapping, the ray \( \ell_q \) is identical to \( \ell_q \), and hence that can intersect with the optical axis at the camera center \( \alpha_i \).

That is, the numerical 3D-2D projection can be achieved by solving the following optimization:

\[
\theta_q = \arg\min_{\theta_q} G(\theta_q) = \arg\min_{\theta_q} \left( f_q(\theta_q) - \lambda_q(\theta_q) \right).
\]

Here \( G(\theta_q) \) is a monotonic function and hence we can refine \( \theta_q \) iteratively as

\[
\theta_q^{(k+1)} = \theta_q^{(k)} - \frac{\Delta \theta_q}{\Delta G} = \frac{f_q(\theta_q^{(k)}) - \lambda_q(\theta_q^{(k)})}{f'_q(\theta_q^{(k)}) - \lambda'_q(\theta_q^{(k)})},
\]

where \( \theta_q^{(k)} \) denotes \( \theta_q \) of \( k \)th iteration.

Therefore, if we compute the derivative \( f'_q \) beforehand as shown in the third row of Table 4, then this 3D-2D projection can be computed efficiently.

### Computational Efficiency

| & Analytical[55] & By Newton with LUT (ours) |
|-----------------|-------------------|-------------------|
| Runtime         | 3.93 msec         | 0.62 msec         |

As shown in Table 5, our 3D-2D projection is much faster than the analytical way while maintaining the sub-pixel accuracy. They are the average values of 100 trials, 10K points 3D-2D projections run in Matlab on an Intel Core-i7 2.6GHz PC. The same as the case of planar refraction, the result shows our Spherical Refraction Ray-Pixel Camera compactly realize the 3D-2D projection, while the analytical way requires the process that choosing the solution from the roots of the higher-degree equation for each point.

### 4.3 Multifacet Mirror

As is well known, observing the scene via a multifacet mirror or a kaleidoscope is identical to observing the scene by virtual multi-view cameras, and in particular, kaleidoscopes with three mirrors are known to be reasonable in terms of less overlaps of mirrored images called discontinuities[56], [57]. In our teleidoscopic imaging system, we use a three-facet mirror which defines reflections of the spherical refraction ray-pixel camera introduced in Section 4.2.

The reflection \( p' \) of a 3D point \( p \) by a mirror of normal \( n \) and distance \( d \) is given by

\[
p' = H_l p + t_l \Leftrightarrow p' = (I - 2n_n^2) p + 2d_n^2,
\]

where \( H_l \) is the Householder transformation matrix.

In the case of kaleidoscopic imaging, the mirrors generate bouncing reflections as shown in Fig. 18. The reflection of \( p' \) by another mirror of normal \( n_j \) and distance \( d \) is given simply by...
4.4 Depth of Field

This section describes analytical evaluations on the depth-of-field of a catadioptric imaging system. We first review the depth-of-field of the thin lens camera model, and then introduces the monocentric lens.

4.4.1 Depth-of-Field of Thin Lens Camera

Ideally all the incident light rays from a subject point to the lens focuses at a common point. As shown in Fig. 19, suppose that a point $s_c$ is focused on the image plane of a camera $C$ through its lens $L$. Then the following thin lens formula for paraxial ray holds:

$$\frac{1}{s} + \frac{1}{f} = \frac{1}{L},$$

(43)

The depth-of-field is defined as the backprojection of the permissible circle-of-confusion centered at the focused point. If the subject distance $s_c$ is not long enough, the near and the far depth-of-focus $\epsilon_N$ and $\epsilon_F$ are given by aperture size $\Phi$ and $t_e$ as

$$\epsilon_N = \frac{\delta t_e}{\Phi - \delta} = \frac{\delta s_c f_e}{(\Phi - \delta)(s_c - f_e)},$$

(44)

$$\epsilon_F = \frac{\delta t_e}{\Phi + \delta} = \frac{\delta s_c f_e}{(\Phi + \delta)(s_c - f_e)}.$$

The near and the far depth-of-field $D_N$ and $D_F$ corresponding to $\epsilon_N$ and $\epsilon_F$ are then obtained by Eq. (43) as

$$D_N = \frac{\epsilon_N}{f^2} = \frac{\epsilon_N(s_c - f_e)}{f^2}, \quad D_F = \frac{\epsilon_F(s_c - f_e)}{f^2}.$$

As a result, $D_N$ and $D_F$ are described as follows:

$$D_N = \frac{\delta s_c f_e (s_c - f_e)}{(\Phi - \delta)(f_e - f_c)} = \frac{\delta s_c f_e (s_c - f_e)}{(\Phi - \delta)(f_e - f_c)}.$$

(46)

Eq. (46) indicates that the depth-of-field of camera $C$ depends on the subject distance $s_c$, the aperture size $\Phi$, and the permissible circle-of-confusion $\delta$.

4.4.2 Depth of Field with a Monocentric Lens

Fig. 20 illustrates the back-projection of the permissible circle-of-confusion through a thin lens and a monocentric camera. The light path through the lens center $\theta_0$ is identical to the path illustrated in Fig. 16.

The key point is that an on-focus scene point at the distance $s_D$ is also on-focus at the distance $s_c$ between the two lenses. That is, depth-of-field with a monocentric lens can be simply defined as an extension of the path of the thin lens camera. This is because the points in $L_N$ and $L_F$ are projected within the permissible circle-of-confusion even though the monocentric lens itself introduces the spherical aberration.

Hence, we define the depth-of-field with a monocentric lens $D_N$ and $D_F$ as the intersection of the backprojection path through the edge of aperture (the led and blue dash lines in Fig. 21) and the backprojection path through the lens center $\theta_0$.

Fig. 22 shows the changes in the depth-of-field by the mono-
section, we aimed at proposing a catadioptric imaging system for microscopic object capture. Unlike conventional microscopic imaging system such as differential phase contrast microscopy\cite{58}, \cite{59} and multi-focus approaches\cite{60}, \cite{61}, \cite{62}, our method realizes a multi-view capture of the target from a single physical viewpoint which can contribute to free-viewpoint rendering, 3D shape reconstruction, and reflection analysis.

The main challenges in image-based microscopic 3D shape measurement is its shallow Depth-of-Field and camera arrangement in the closeup scenario. Applying conventional multiple camera system designed for human-size capture\cite{63}, \cite{64} cannot be a feasible solution due to limitations on camera placement. Conventional multiple mirror system\cite{56} also have difficulties inevitably in depth-of-focus due to differences in their optical paths with varying numbers of bounces.

The key idea to solve these problems is to employ a catadioptric imaging system which realizes a practical closeup multi-view imaging. The point of our design is that the system has a monocentric front lens like a telescoposcope, instead of using microscopic imaging system shown in Fig. 25, we introduce a kaleidoscoposcopic multi-facet mirror between the front lens and the camera. As discussed later, this design realizes a deeper depth-of-field and results in less blurring imaging.

We call our system teleidoscopic imaging system and show that the system can be compactly modeled by a structured ray-pixel camera model\cite{8}.

5.1 Calibration of Teleidoscopic Imaging System

This section introduces our calibration algorithm of teleidoscopic imaging system which requires capturing a single reference planar patterns. Fig. 26 shows the measurement model where the real camera $C$ observes reference points $p_n$ such as chessboard corners on a reference board $\Pi$ via a monocentric lens and three mirrors. Our calibration estimates the mirror normals $n_i$, their distances $d_i$ from the camera, and the position of the monocentric lens, with assuming that the 2D positions of the reference points $p_n$ on the reference plane $\Pi$, the intrinsic parameters of the camera, and the refraction index of the lens are given beforehand.

A challenge in this calibration is the fact that the mirrors require observing 3D points and their reflections to estimate their poses\cite{56}, while the observation in the teleidoscopic system does not include such mirrored points that follow Eq. (42) due to the refraction by the monocentric lens. Similarly, the rays reflected and then refracted through the projection in teleidoscopic imag-
The axis to the center of the mirrored monocentric lens \( a^{(i)}_v \) can be obtained by the mirrored points \( p^{(i)}_v \).

### 5.1.2 Mirror Normals

The axis to the center of monocentric lens \( a^{(0)}_v = (x^{(0)}, y^{(0)}, z^{(0)})^\top \) and its mirror \( a^{(i)}_v = (x^{(i)}, y^{(i)}, z^{(i)})^\top \) are formed in reflection satisfies

\[
\mathbf{a}^{(0)}_v [\mathbf{n}_i], \mathbf{a}^{(i)}_v = 0, \tag{49}
\]

where \( \mathbf{n}_i \) denotes the skew-symmetric matrix defined by the normal \( \mathbf{n} = (x_n, y_n, z_n)^\top \) of the mirror \( i \).

The same constraint holds for each of the first-second reflection pairs \( \mathbf{a}^{(i)}_v - \mathbf{a}^{(0)}_v \) about the same mirror normal \( \mathbf{n}_i \) [56]. Therefore \( \mathbf{n}_i \) can be obtained linearly only from the axes to the centers of the monocentric lenses.

### 5.1.3 Mirror Distances

Once the mirror normals are estimated, we can utilize the kaleidoscopic triangulation [56] to obtain linear constraints on the mirror distances \( d_i \). That is, for the first and the second reflections such as \( C_i \) and \( C_j \), following equation holds:

\[
\mathbf{a}^{(0)}_v \times (\mathbf{H}_i \mathbf{a}^{(i)}_v)^\top (\mathbf{0} - \mathbf{t}_i) = 0 \tag{50}
\]

By integrating Eq. (50) for \( ij = [12, 13, 21, 23, 31, 32] \) as a set of linear equations of \( d_i \) \( (i = 1, 2, 3) \), we linearly obtain \( d_1, d_2, d_3 \) up to scale.

### 5.1.4 Pose of Reference Plane

Similarly to Eq. (47), the plane-of-refraction constraint holds for the mirrored cameras \( C_i \):

\[
(\mathbf{p}^{(i)}_v)^\top \left( \mathbf{a}^{(i)}_v \times (\mathbf{H}_i(\mathbf{R}_i \mathbf{p}_w + \mathbf{t}_i) + \mathbf{t}_i) \right) = 0. \tag{51}
\]

This constraint allows us estimating the pose of the reference plane \( \mathbf{R}_i \) and \( \mathbf{t}_i \) linearly.

### 5.1.5 Monocentric Lens Parameters

The calibration algorithm up to this point does not require the monocentric lens parameters \( d_p, r_p, \) and \( \mu_p \). We estimate these parameters by the coplanarity constraint of the ray through \( \mathbf{p}^{(i)}_v \) of \( C_i \):

\[
\mathbf{a}^{(i)}_v \times (\mathbf{p}^{(i)}_v - \mathbf{p}^{(0)}_v) = 0. \tag{52}
\]

This is a nonlinear constraint for the monocentric lens parameters as described in Section 4.1.1 and we solve this as a nonlinear optimization problem with assuming their rough estimates are available in practice.

### 5.1.6 Bundle Adjustment

The last step of our calibration is to refine the parameters \( \mathbf{a}^{(i)}_v, d_p, r_p, \mu_p, \mathbf{n}_i, d_i, \mathbf{R}_i, \mathbf{t}_i \) \( (i = 1, 2, 3) \) by minimizing the re-projection errors of the reference points \( \mathbf{p}_w \) as a nonlinear optimization problem. On computing the 3D-2D projection, we used the analytical solution by [55].

### 5.2 Evaluation

In this section, we evaluate the calibration of our catadioptric ray-pixel camera.
5.2.1 Quantitative Evaluation using Synthesized Data

Fig. 27 shows the measurement environment which simulates the real capture system used in Section 5.4. The system has a kaleidoscope with three $10 \times 30$mm mirrors in front of the camera C. The mirrors are at slightly off-perpendicular angle of $1.4^\circ$ to the camera image plane so that the mirrors define virtual cameras around the target with less overlaps of the mirrored images. The system also has a monocentric lens of 10mm diameter in front of the mirrors, at 40mm distance from the camera. The refraction index $\mu_g$ is set to 2.0.

Fig. 28 Reprojection errors at different noise levels. The bars denote the standard deviation of the errors.

The system captures 48 reference points (blue dots in Fig. 27) to calibrate its parameters. By injecting Gaussian noise of different standard deviations $\sigma$ to the 2D positions of their projections, we evaluate the robustness of our calibration procedure.

Fig. 29 shows average reprojection errors in pixel of 100 trials at each pixel noise level $\sigma$. We can observe that the reprojection errors increases linearly against the pixel noise level.

Figs. 29, 30, 31 show the estimation errors of the monocentric lens parameters, the mirror parameters, and the reference plane parameters respectively. These results indicate that our calibration algorithm performs reasonably under realistic observation noise.

5.3 Teleidoscopic Triangulation

Since our Teleidoscopic Imaging System has multiple ray-pixel cameras, the manner of the triangulation is the same as described in Section 3.3.2.

In the case of three or more viewpoints, depending on the number of the cameras sharing an intersection of the depth-of-fields (Fig. 24), we can add equations in the same form into this system for triangulation.

5.4 Teleidoscopic 3D Shape Reconstruction

To evaluate the proposed teleidoscopic system as a multi-view camera system for 3D shape reconstruction, this section demonstrates a 3D reconstruction of a small object of approximately 5mm size shown in Fig. 32(a).

The system consists of a FLIR Flea3 FL3-U3-88S2C-C camera (4000×3000 resolution, pixel size 1.55μm) with an S-mount lens (focal length 3mm, F8), three $10 \times 30$mm mirrors, and a monocentric lens of 10mm diameter whose refraction index is $\mu_g = 2.0$.

Figs. 32(b) and (c) show images captured by our teleidoscopic system and by its kaleidoscopic part without the monocentric lens. These images clearly demonstrates that the use of the monocentric lens realizes a denser close-up surrounding multi-view capture of the target.

In order to demonstrate the idea of our teleidoscopic multi-view capture, we introduce a focus-free laser projector Sony MP-CL1A to cast structured light patterns[53], [54] to minimize er-
rors in the stereo correspondence search process as shown in Figs. 34(b) and (c).

Figs. 34(a) and (b) show the result of our 3D reconstruction as a point cloud. This result demonstrates that our system realizes a closeup and surround-view capturing successfully.

6. Conclusion

This paper proposed a catadioptric ray-pixel camera model exploiting an axially-symmetric structure of the rays captured by the system. Our model describes the ray associated with each pixel by a simple 1D mapping, and realizes an efficient forward 3D-2D projection.

The proposed system realized a closeup, semi-surround view, and less blurring imaging system for microscale objects. Our future work includes an extension to fully-surround view 3D capture of microscale underwater objects, with immersing the front monocentric lens in water.

Acknowledgments

This research is partially supported by JSPS KAKENHI 26240023, 15J07706, and 18K19815.

References
