# A Unified Model for Line Projections in Catadioptric Cameras with Rotationally Symmetric Mirrors

Pedro Miraldo
Mitsubishi Electric Research Labs (MERL)
miraldo@merl.com

José Pedro Iglesias
Chalmers University of Technology
jose.iglesias@chalmers.se

## Abstract

*Lines are among the most used computer vision features, in applications such as camera calibration to object detection. Catadioptric cameras with rotationally symmetric mirrors are omnidirectional imaging devices, capturing up to a 360 degrees field of view. These are used in many applications ranging from robotics to panoramic vision. Although known for some specific configurations, the modeling of line projection was never fully solved for general central and non-central catadioptric cameras. We start by taking some general point reflection assumptions and derive a line reflection constraint. This constraint is then used to define a line projection into the image. Next, we compare our model with previous methods, showing that our general approach outputs the same polynomial degrees as previous configuration-specific systems. We run several experiments using synthetic and real-world data, validating our line projection model. Lastly, we show an application of our methods to an absolute camera pose problem.*

## 1. Introduction

Catadioptric cameras are imaging devices combining general mirrors with perspective cameras, [41]. These systems are used to get up to a 360-degree field of view with a single image and are suitable for applications ranging from robotics to medical imaging. In theory, particular cases of catadioptric cameras can be modeled by the central perspective model [3][1]. However, this modeling requires the use of specific mirrors and that the camera is perfectly aligned and at a specific distance concerning the mirror's positions. Therefore, even if possible in theory, as argued in [53], catadioptric camera systems are non-central cameras.

Due to their non-linear distortions, modeling point and line projections in catadioptric systems are significantly more complex than in the perspective cameras. Researchers made a big effort to derive a unified model for the projection of points, firstly only for central systems, [4, 18], and for general configurations, central and non-central, [1, 2]. On the other hand, although solved for some particular configurations, e.g., [1, 4, 7, 8], a unified model for line projection was never proposed.

Line projections have been used in many computer vision applications, such as: i) camera calibration [15, 28, 36, 46]; ii) epipolar geometry [5, 25, 31, 50, 52]; iii) the Structure-from-Motion, and pose estimation [12, 20, 26, 37, 43, 47]; iv) and 3D reconstruction [10, 19, 35, 38]. In all these examples, we assume the use of central systems. To develop a similar application using catadioptric systems, we need to derive the respective projection of straight lines. A couple of papers [35,56] are published on self-calibration of non-central catadioptric cameras using the epipolar geometry. However, the authors assume central approximations for computing epipolar lines, *i.e.*, approximation of the projection of 3D lines by a central camera model. On the other hand, imaging models for representing any camera (central and non-central) have been proposed; see [24, 40, 49]. [51, 59] list the advantages of non-central cameras compared to central. For example, when using a non-central camera, a single image of the environment is enough to recover the line's 3D parameters, *e.g.*, [11, 17, 29, 54]).

This paper presents the first unified model for the projection of 3D straight lines in general catadioptric cameras with rotationally symmetric mirrors; central and non-central systems. In Tab. 1, we show the degrees of the polynomials of prior specifically-derived methods and some results obtained from our derivations. Next, we present the related work. Secs. 3 and 4 define the problem and derive the proposed unified model. Section 5 presents the experiments and Sec. 6 concludes the paper.

## 2. Related Work

This section shows the related work on modeling point and line projections in omnidirectional cameras for central and non-central systems.

[1] Only two combinations of mirror settings and perspective camera positions are possible. Table 1 shows these two cases.

| System Setup | | | | Curve Degree | Previous? |
|---|---|---|---|---|---|
| **Mirror Type** | **Camera Position** | **Mirror Parameters** | **Central?** | | |
| General | Any | $A = *, B = *, C = *$ | No | 6 | – |
| General | Axial | $A = *, B = *, C = *$ | No | 6 | – |
| Sphere | Any | $A = 1, B = 0, C = *$ | No | 4 | [1] |
| Cone | Axial | $A = -1, B = 0, C = 0$ | No | 4 | [7, 8]† |
| Ellipsoidal | Axial | $A = \frac{2k}{c_3^2+2k}, B = -\frac{2c_3k}{c_3^2+2k}, C = -\frac{c_3^2k}{2c_3^2+4k} + \frac{k}{2}$ | No | 6 | – |
| Ellipsoidal | At the foci, *i.e.,* $[0\ 0\ c_3]^T$ | $A = \frac{2k}{c_3^2+2k}, B = -\frac{2c_3k}{c_3^2+2k}, C = -\frac{c_3^2k}{2c_3^2+4k} + \frac{k}{2}$ | Yes | 2 | [4, 18] |
| Hyperboloidal | Axial | $A = -\frac{2}{k-2}, B = \frac{2c_3}{k-2}, C = \frac{c_3^2k}{k-2} - \frac{c_3^2}{2k}$ | No | 6 | – |
| Hyperboloidal | At the foci, *i.e.,* $[0\ 0\ c_3]^T$ | $A = -\frac{2}{k-2}, B = \frac{2c_3}{k-2}, C = \frac{c_3^2k}{k-2} - \frac{c_3^2}{2k}$ | Yes | 2 | [4, 18] |

Table 1. We show a small set of specific configurations representing the degrees of the implicit equations derived in this paper against the previous techniques when there is one. We note that previous methods only consider specific camera configurations in mirror parameters and camera position. The parameter $k$ in the table is a general mirror parameter representing central catadioptric cameras, as defined in [3]. †indicates these polynomial degrees are not directly derived in [7, 8].

**Points and central systems:** This has been extensively studied for perspective cameras [25, 31]. For the conditions derived in [3], [18] proposes a unified projection model for central omnidirectional cameras. This method uses a two projections pipeline. It first consists of projecting the points into a sphere, then projecting them into the image using a changed perspective projection model. [33] adapts this model and considers planar grids for a flexible calibration technique. Other well-known works on the image formation of omnidirectional systems consider projection approximations, such as [16, 48]. [55] presents another interesting work. A point is projected into a line in the image, making it model-independent from distortion.

**Lines and central systems:** Line projection for perspective cameras is well defined; see [25, 31]. We also find their modeling for central catadioptric systems. [18] is the first work exploring this problem. The authors follow the same two projections strategy used to project 3D points. In [4], the authors further explore this line projection problem. They present some relevant properties and their advantages in camera calibration. Some other authors focus on line projection fitting; see for example [6, 9, 32]. Table 1 lists current central solutions.

**Points and non-central systems:** One of the first works addressing the image formation of non-central catadioptric cameras is [53], in which the authors use caustics on the modeling. Most authors use polynomials (implicit equations) to represent these 3D point projections at the mirror. [1] starts by proposing a polynomial equation of degree 6 to represent the projection of 3D points in axial non-central catadioptric cameras (camera aligned with the mirrors axis of symmetry). This work is extended for general camera positions in [2], with a polynomial equation of degree 8. In [21–23], the authors follow a different approach. They notice that the problem could not be solved using closed-form operations and focus on defining efficient iterative techniques.

**Lines and non-central catadioptric:** Analytical solutions to the line-image for non-central catadioptric cameras were only analyzed for two camera positions/mirror configurations. As in the latest works on modeling 3D projection of points, previous authors rely on modeling these projections using implicit equations in the form of polynomials. [1] proposes a solution to the 3D projection of lines into non-central catadioptric cameras with spherical mirrors. The authors obtain a 4-degree polynomial to represent the curve in the image. In [7, 8], the line projection model and their fitting for conical non-central catadioptric cameras are proposed. Concerning the model, the authors assume an axial system and get a 4-degree polynomial to represent the curve in the mirror. Other authors admit approximate solutions for this line projection. [14] study the projection of 3D lines using the Generalized Linear Camera model, showing good results for a small local window. Yang *et al.* [57] fit several lines using an approximate projection that makes use of a set of basis functions and a look-up table. Other authors present analytical solutions to different non-central systems; see [27, 34, 44, 58].

Unlike prior work, this paper offers a general and analytical model for the 3D projection of lines in the entire image space. Our solution can be used to model systems with any rotationally symmetric mirror and camera positions, i.e., central and non-central imaging devices.

## 3. Problem Definition

As in previous catadioptric image formation works, [1, 4, 7, 8, 18], we use rotationally symmetric mirrors, represented by $\Omega(\mathbf{m}) = 0$, where

$$\Omega(\mathbf{m}) = x^2 + y^2 + Az^2 + Bz - C, \tag{1}$$

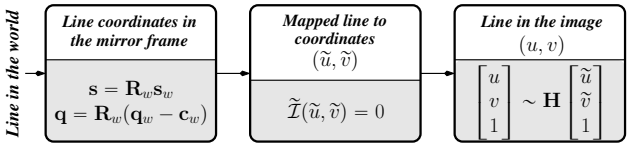



Figure 1. **Projection model:** The input is a line in the world ($\mathbf{q}_w$ and a direction $\mathbf{d}_w$). The first step transforms the line parameters to the mirror's reference frame by applying a rigid transformation $\mathbf{R}_w \in \mathcal{SO}(3)$ and $\mathbf{c}_w \in \mathbb{R}^3$ (the rotation and world frame position of the mirror, respectively). Next, we map the 3D line from the world into the normalized image space. Last step applies a standard collineation between two projective planes; $\mathbf{H} = \mathbf{KR}$.

and $\mathbf{m} = [x\ y\ z]^T$ is a point on the mirror's surface. From (1), the normal vector at the mirror point $\mathbf{m}$ is given by

$$\mathbf{n}(\mathbf{m}) = \nabla\Omega(\mathbf{m}) = [x \quad y \quad Az + B/2]^T. \tag{2}$$

Taking the perspective projection equation (see [25]), the image of a point in the mirror's surface is given by

$$\zeta \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{KR}\,(\mathbf{m} - \mathbf{c}) \tag{3}$$

where $\mathbf{K} \in \mathbb{R}^{3\times3}$ represents the camera intrinsics. $[u\ v]$ are an image point in pixel coordinates. $\mathbf{R} \in \mathcal{SO}(3)$ and $\mathbf{c} \in \mathbb{R}^3$ are the rotation of the camera and its center with respect to the mirror's reference frame, respectively. As in [1], without loss of generality, given its symmetry, one can rotate the mirror's coordinate system, getting $\mathbf{c} = [0\ c_2\ c_3]^T$.

To represent lines, we use a point $\mathbf{q} \in \mathbb{R}^3$ and a direction $\mathbf{s} \in \mathbb{R}^3$; any point on the line satisfies

$$\mathbf{p}(\lambda) = \mathbf{q} + \lambda\mathbf{s}, \text{ for any } \lambda \in \mathbb{R}. \tag{4}$$

Given the high degrees of some polynomials in the paper, we use $\kappa_i^j[.]$ to denote the $i^{\text{th}}$ polynomials $j^{\text{th}}$ total degree[2].

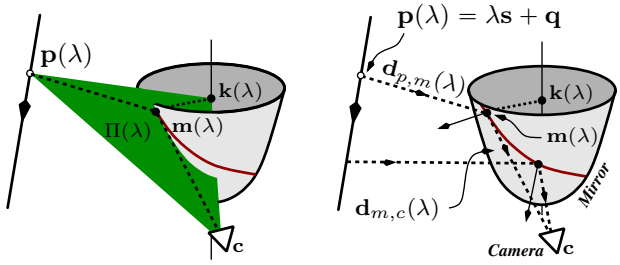
We conclude this section by describing our problem:

**Problem 1** (Line projection)**.** *The projection of lines in general catadioptric cameras with rotationally symmetric mirrors is given by an implicit equation (in polynomial form), $\mathcal{I}(u, v)$, with coefficients specified as a function of the 3D line, perspective camera, and mirror parameters.*

## 4. Line Projection in Catadioptric Cameras

Figure 1 shows the projection model used for catadioptric cameras. As in [1, 4], without loss of generality, we assume that the world reference frame is aligned with the mirror (*i.e.*, $\mathbf{R}_w = \mathbf{I}$ and $\mathbf{c}_w = \mathbf{0}$). Then, this section focuses on mapping the line coordinates.

[2]The derived polynomial coefficients are available for download at https://github.com/pmiraldo/line-projection-catadioptric.




(a) Reflection Plane (b) Snell Reflection Law

Figure 2. **Projection constraints:** At the left, we show the planar constraint defined by each point on the line, its reflection point on the mirror, and the perspective camera center. At the right, we show Snell's reflection constraint. The red curve in the images represents the line projection into the mirror.

Consider a 3D line, $\mathbf{p}(\lambda)$, represented in the mirror frame, as shown in (4). This section proposes a parametric representation for the line projection, $\mathcal{I}(u, v)$; *i.e.*, solves Problem 1. We start by defining some basic projection constraints in Sec. 4.1. The algebraic line-reflection constraint we use to model line projections is derived in Sec. 4.2. Section 4.3 presents the parameterization of the projection curve in normalized image coordinates. To conclude, the implications on the application of the standard collineation is shown in Sec. 4.4 (affine transformation $\mathbf{H}$ in Fig. 1).

### 4.1. Projection Constraints

We use two basic properties of the catadioptric image formation for deriving the line-projection constraint (to be presented in the following subsection), see [2, 21]:

**Definition 1.** *A point on the line, $\mathbf{p}(\lambda)$, its reflection point on the mirror, $\mathbf{m}(\lambda)$, and the perspective camera's center of projection, $\mathbf{c}$, define a plane $\Pi(\lambda)$ (see Fig. 2(a)); and*

**Definition 2.** *The incoming and reflected rays, $\mathbf{d}_{p,m}(\lambda)$ (from $\mathbf{p}(\lambda)$ to $\mathbf{m}(\lambda)$) and $\mathbf{d}_{m,c}(\lambda)$ (from $\mathbf{m}(\lambda)$ to $\mathbf{c}$) respectively, in Fig. 2(b), must satisfy the Snell's law of reflection.*

We start with the plane constraint at Definition 1. Using the fact that $\mathbf{k}(\lambda) = \mathbf{m}(\lambda) + \mu\nabla\Omega(\mathbf{m})$, for $\mu \in \mathbb{R}$, is in the 3D plane and intersects the mirror's axis of symmetry at $\mu = -1$, we write

$$\mathbf{k}(\lambda) = [0 \quad 0 \quad (1 - A)z - B/2]^T. \tag{5}$$

Now, stacking $\mathbf{m}(\lambda)$, $\mathbf{k}(\lambda)$, $\mathbf{c}$, and $\mathbf{p}(\lambda)$, such that

$$\mathbf{M}^T = \begin{bmatrix} \mathbf{m}(\lambda) & \mathbf{p}(\lambda) & \mathbf{k}(\lambda) & \mathbf{c} \\ 1 & 1 & 1 & 1 \end{bmatrix}, \tag{6}$$

since all these points are on the same plane, a constraint is obtained; the determinant of $\mathbf{M}$ must be zero, $\mathcal{C}_1(\mathbf{m}, \lambda) = \det(\mathbf{M}) = 0$. By expanding the determinant of $\mathbf{M}$, we describe the following Lemma:

**Lemma 1** (Plane reflection constraint)**.** *A 3D point on the line, $\mathbf{p}(\lambda)$, verifying Definition 1 provides a constraint $\mathcal{C}_1(\mathbf{m},\lambda)=0$, such that*

$$\mathcal{C}_1(\mathbf{m},\lambda) \doteq \kappa_1^1[x,y]\lambda z + \kappa_2^1[x,y]z + \kappa_3^1[x,y]\lambda + \kappa_4^1[x,y]. \tag{7}$$

Let us now focus on Definition 2. Taking the Snell reflection law and the fact that $\mathbf{d}_{p,m}(\lambda)\times\mathbf{d}_{p,m}(\lambda)=\mathbf{0}$, after some simplifications, we obtain

$$\begin{aligned}\langle \mathbf{n}(\mathbf{m}),\mathbf{n}(\mathbf{m})\rangle\ [\mathbf{d}_{p,m}(\lambda)]_{\times}\ \mathbf{d}_{m,c}(\lambda) - \\ 2\langle \mathbf{d}_{m,c}(\lambda),\mathbf{n}(\mathbf{m})\rangle\ [\mathbf{d}_{p,m}(\lambda)]_{\times}\ \mathbf{n}(\mathbf{m}) = \mathbf{0},\end{aligned} \tag{8}$$

in which $[\mathbf{a}]_{\times}$ is a skew-symmetric matrix that linearizes the cross product, *i.e.*, $\mathbf{a}\times\mathbf{b}=[\mathbf{a}]_{\times}\mathbf{b}=\mathbf{0}$. In addition, by definition, we have

$$\mathbf{d}_{m,c}(\lambda)=\mathbf{m}(\lambda)-\mathbf{c}, \text{ and } \mathbf{d}_{p,m}(\lambda)=\mathbf{p}(\lambda)-\mathbf{m}(\lambda). \tag{9}$$

Substituting (9) in (8), we get three algebraic constraints[3] and define the following Lemma:

**Lemma 2** (Reflection law constraints)**.** *Definition 2 generates the following three algebraic constraints:*

$$\mathcal{C}_{i+1}(\mathbf{m},\lambda)=\kappa_{3+2i}^3[x,y,z]\lambda+\kappa_{4+2i}^4[x,y,z], \quad i=1,2,3. \tag{10}$$

Next, we define the line reflection constraint, which will use for modeling the 3D line projection.

### 4.2. Line-Reflection Constraint

Line projection does not depend on the depth of the point on the line, *i.e.*, $\lambda$. To define the line-reflection constraint, we use Lemmas 1 and 2 to get a constraint as a function of only $x$, $y$ and $z$. We start by taking $\mathcal{C}_2(\mathbf{m},\lambda)=0$, from Lemma 1, and solve it as a function of $\lambda$:

$$\lambda = -\frac{\kappa_2^1[x,y]z+\kappa_4^1[x,y]}{\kappa_1^1[x,y]z+\kappa_3^1[x,y]}. \tag{11}$$

Now, taking Lemma 2, we substitute the $\lambda$ from (11) at (10), and pre-multiply the resulting equations by the denominator of (11). We obtain $\widetilde{\mathcal{C}}_{i+1}(\mathbf{m})=0$ such that

$$\widetilde{\mathcal{C}}_{i+1}(\mathbf{m})=\kappa_{11}^2[x,y,z]\kappa_{11+i}^4[x,y,z], \quad i=1,2,3. \tag{12}$$

We note that (12) contain two polynomial factors of degrees 2 and 4, where the former depends only on the mirror and camera's parameters. These lower degree polynomial factors are thus considered a system singularity and disregarded, leaving $\widetilde{\mathcal{C}}_{i+1}(\mathbf{m})$ as a 4 degree polynomial. The degree of $\widetilde{\mathcal{C}}_{i+1}(\mathbf{m})$ can be further decreased by replacing $z^2$ by $C-x^2-y^2-Bz$. Doing this turns all 3 constraints at (12) linear dependent. Taking one, we describe the following theorem:

[3] From (8), only two are linearly independent.

**Theorem 1** (Line-reflection constraint)**.** *The Line-Reflection constraint is given by the points in the mirror with coordinates $x$, $y$ and $z$ verifying the algebraic constraints $\mathcal{C}_{lr}(\mathbf{m})=0$, such that*

$$\mathcal{C}_{lr}(\mathbf{m})=\kappa_{15}^2[x,y]z+\kappa_{16}^3[x,y]. \tag{13}$$

Although not needed to define the 3D line projection into the image, Thm. 1 can be used to represent the 3D line reflection curve in the mirror:

**Remark 1** (3D reflection curve on the mirror)**.** *Taking the point on the mirror constraint $\Omega(x,y,z)=0$, the line reflection constraint $\mathcal{C}_{lr}(\mathbf{m})=0$ in Thm. 1, and $z\mathcal{C}_{lr}(\mathbf{m})=0$, we define*

$$\underbrace{\begin{bmatrix} A & B & x^2+y^2-C \\ 0 & \kappa_{15}^2[x,y] & \kappa_{16}^3[x,y] \\ \kappa_{15}^2[x,y] & \kappa_{16}^3[x,y] & 0 \end{bmatrix}}_{\mathbf{N}} \begin{bmatrix} z^2 \\ z \\ 1 \end{bmatrix} = \mathbf{0}. \tag{14}$$

*With this, we define a constraint $\mathcal{R}(x,y)=det(\mathbf{N})=0$, which is a 6-degree polynomial equation, parametrizing the reflection curve as a function of $x$ and $y$. To get the respective $z$ for modeling the 3D curve on the mirror, we solve (13), for a known $\{x,y\}$.*

Combining our line-reflection constraint, the mirror equation, and the perspective projection model, we obtain bi-variable polynomials that implicitly parameterize the projection curve of the 3D line on the image plane. These derivations are given in the following subsections.

### 4.3. Modeling Line Projection

Here, we use the mirror's equation, (1), the line-reflections constraint (13), and the projection into the normalized plane:

$$\begin{bmatrix}\widetilde{u}\\ \widetilde{v}\\ 1\end{bmatrix} \times (\mathbf{m}-\mathbf{c}) = \mathbf{0}, \tag{15}$$

which is obtained from (3) without the last collineation step; *i.e.*, without the application of the transformation $\mathbf{H}$ (see Fig. 1). Notice that $\xi$ was removed from the camera equation (see (3)) by considering the cross product of $[\widetilde{u}\ \widetilde{v}\ 1]^T$ on both sides. This operation provides two linearly independent equations that can be use to write $x$ and $y$ as a function of $\widetilde{u}$, $\widetilde{v}$ and $z$:

$$x=\frac{\kappa_{19}^2[\widetilde{u},\widetilde{v},z]}{\kappa_{18}^1[\widetilde{u},\widetilde{v}]}, \text{ and } y=\frac{\kappa_{21}^2[\widetilde{u},\widetilde{v},z]}{\kappa_{20}^1[\widetilde{u},\widetilde{v}]}. \tag{16}$$

Plugging (16) into $\Omega(\mathbf{m})$ and $\mathcal{C}_{lr}(\mathbf{m})$ gives

$$\widetilde{\Omega}[\widetilde{u},\widetilde{v},z]=\kappa_{22}^2[\widetilde{u},\widetilde{v}]z^2+\kappa_{23}^2[\widetilde{u},\widetilde{v}]z+\kappa_{24}^2[\widetilde{u},\widetilde{v}], \tag{17}$$

$$\widetilde{\mathcal{C}}_{lr}[\widetilde{u},\widetilde{v},z]=\kappa_{25}^5[\widetilde{u},\widetilde{v}]z+\kappa_{26}^5[\widetilde{u},\widetilde{v}], \tag{18}$$

where in $\widetilde{\mathcal{C}}_{lr}[\widetilde{u},\widetilde{v},z]$ we have replaced $z^2$ by $-(\kappa^2_{23}[\widetilde{u},\widetilde{v}]z+\kappa^2_{24}[\widetilde{u},\widetilde{v}])/\kappa^2_{22}[\widetilde{u},\widetilde{v}]$.

Now, to get the line in the image space, we want to remove $z$ from the constraints. Taking (17), (18), and $z\widetilde{\mathcal{C}}_{lr}[\widetilde{u},\widetilde{v},z]=0$, we setup the following algebraic equation:

$$\underbrace{\begin{bmatrix}\kappa^2_{22}[\widetilde{u},\widetilde{v}] & \kappa^2_{23}[\widetilde{u},\widetilde{v}] & \kappa^2_{24}[\widetilde{u},\widetilde{v}]\\ 0 & \kappa^5_{25}[\widetilde{u},\widetilde{v}] & \kappa^5_{26}[\widetilde{u},\widetilde{v}]\\ \kappa^5_{25}[\widetilde{u},\widetilde{v}] & \kappa^5_{26}[\widetilde{u},\widetilde{v}] & 0\end{bmatrix}}_{\mathbf{N}\in\mathbb{R}^{3\times3}}\begin{bmatrix}z^2\\ z\\ 1\end{bmatrix}=\mathbf{0}. \quad (19)$$

For (19) to be true, the determinant of $\mathbf{N}$ must be zero. Then, by computing $\det(\mathbf{N})$, after some simplifications, we get a polynomial of maximum degree 6, which we use to describe the following Theorem:

**Theorem 2** (Projection curve on the normalized plane)**.** *The projection curve on the normalized plane of a 3D line is given by*

$$\widetilde{\mathcal{I}}(\widetilde{u},\widetilde{v})=0, \quad (20)$$

*where $\widetilde{\mathcal{I}}(\widetilde{u},\widetilde{v})$ is a polynomial of maximum degree 6.*

With respect to the degree of $\widetilde{\mathcal{I}}(\widetilde{u},\widetilde{v})$, we define the following remark:

**Remark 2.** *The degree of $\widetilde{\mathcal{I}}(\widetilde{u},\widetilde{v})$ is lower for some specific system configurations. For instance, for spherical and conical systems the degree of $\widetilde{\mathcal{I}}(\widetilde{u},\widetilde{v})$ is 4. The same happens to the central cases, getting a 2 degree curve. These reductions are obtained by disregarding factors in $\widetilde{\mathcal{I}}(\widetilde{u},\widetilde{v})$, which are only system dependent. Some specific configurations and corresponding polynomial degrees are summarized in Tab. 1. They validate previous results.*

MATLAB scripts with all our derivations are public at https://github.com/pmiraldo/line-projection-catadioptric.

### 4.4. On the Application of the Collineation

As shown in Fig. 1, collineation between the normalized and the image plane is expressed as

$$\zeta\begin{bmatrix}u\\ v\\ 1\end{bmatrix}=\underbrace{\mathbf{KR}}_{\mathbf{H}}\begin{bmatrix}\widetilde{u}\\ \widetilde{v}\\ 1\end{bmatrix}. \quad (21)$$

Assuming $\mathbf{K}$ as an upper triangular matrix, from the third equation of (21) we have that

$$\zeta=r_{31}\widetilde{u}+r_{32}\widetilde{v}+r_{33}, \quad (22)$$

and replacing $\zeta$ in the first two equations of (21), a relation between $(\widetilde{u},\widetilde{v})$ and $(u,v)$ is obtained as $\widetilde{u}=\frac{\kappa^1_{30}[u,v]}{\kappa^1_{29}[u,v]}$ and $\widetilde{v}=\frac{\kappa^1_{31}[u,v]}{\kappa^1_{29}[u,v]}$, where the denominators of the two fractions are identical. Variables $\widetilde{u}$ and $\widetilde{v}$ are replaced in (20) and, after multiplication with $(\kappa^1_{29}[u,v])^6$, we get the following result:

| System | Parameters | | | | |
|---|---|---|---|---|---|
| | $A$ | $B$ | $C$ | $c_2$ | $c_3$ |
| Elliptic | 0 | 10 | 0 | 2 | 35 |
| Parabolic | 0.5 | 0 | 80 | 2 | 40 |
| General | -1.2 | -1.4 | -23.2 | 10 | 30 |
| Hyperboloidal Central | -0.4 | 14 | 35 | 0 | 35 |
| Ellipsoidal Central | 0.14 | -4.9 | 7.0 | 0 | 35 |
| Axial Cone | -1 | 0 | 0 | 0 | 25 |

Table 2. **Simulated Systems:** Mirror parameters $A$, $B$, and $C$, and the camera positions with respect to the mirror $c_2$ and $c_3$, used in to generate the synthetic data (see results of Fig. 3 and Fig. 4). The top three rows show three examples of systems that cannot be modeled by previous models.

**Theorem 3** (Projection curve on the image)**.** *The projection of a 3D line defined by a point $\mathbf{q}$ and direction $\mathbf{s}$, a perspective camera centered at $\mathbf{c}$ with rotation $\mathbf{R}$, intrinsic calibration matrix $\mathbf{K}$, and a quadratic mirror $\Omega(\mathbf{m})$ is defined by*

$$\mathcal{I}(u,v)=0, \quad (23)$$

*where $\mathcal{I}(u,v)$ is a polynomial of maximum degree 6.*

## 5. Experiments

We present the first unified model for line projection in catadioptric cameras. Therefore, we start these experimental results by giving some line projections for general configurations, mirror, and camera positions settings, validating the theoretical contributions of the paper. In addition, we consider cases where small misalignments of specific configurations occur to show the advantages of using our unified model over previous specific ones. The reader can test other catadioptric systems with the `Matlab` scripts in the link indicated above. Section 5.2 presents some results with real data. Section 5.3 shows an application of our methods in absolute pose estimation and AR problems.

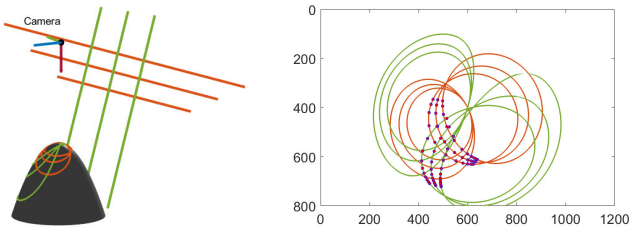
### 5.1. Line Projection

We start with synthetic experiments for validation and comparisons with prior modelings. Using `Matlab`, we simulate systems with parameters in Tab. 2, and

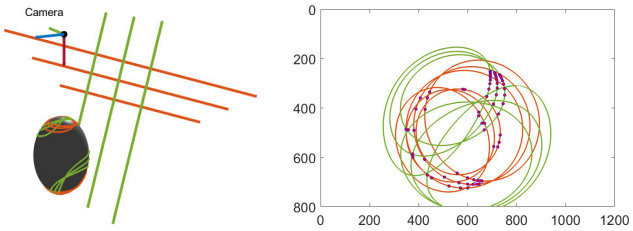
$$\mathbf{K}=\begin{bmatrix}750 & 0 & 600\\ 0 & 750 & 400\\ 0 & 0 & 1\end{bmatrix} \text{ and } \mathbf{R}=\text{diag}(1,-1,-1), \quad (24)$$

with images size $1200\times800$. Then, we apply results from Thm. 3 and Rmk. 1, to obtain the projection curves in the image and mirror, respectively.

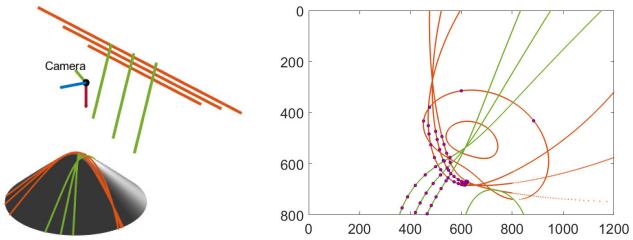
**Validation:** We take the first three systems of Tab. 2, which are examples of systems that previous techniques cannot




(a) Elliptical System

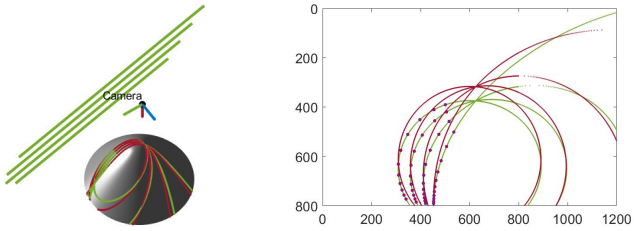



(b) Parabolic System

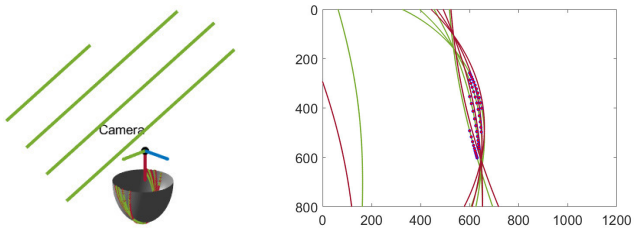



(c) General mirror and camera position

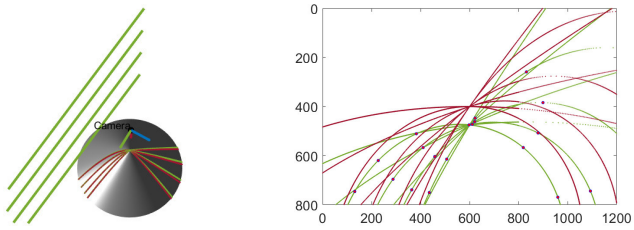
Figure 3. **Validation:** We show the application of the proposed unified line projection modeling in the images and mirrors for three catadioptric systems. On the left, we show the camera system, the 3D lines, and the reflection curve in the mirror (Rmk. 1). At the right, we show the projection of the 3D lines into the images (Thm. 3). We consider two sets of three 3D parallel lines and their respective projections for each system in orange and green curves. We show that the curve matches the projection of 3D points from [2], identified with purple dots in the images. The 3D graphs only show one sheet of the mirror and their respective projection curves. We do not have this constraint in the image and, therefore, get up to two curves per 3D line. We note that these are three examples of systems that previous techniques cannot model concerning the mirror and camera position parameters.




(a) Hyperboloidal Central




(b) Ellipsoidal Central




(c) Axial cone

Figure 4. **Exact vs. approximation models:** Prior models consider rigorous camera placements with respect to the mirror. In practice, we cannot guarantee that this will happen. This figure evaluates the effects of considering our unified model against the previous specific models when computing the line projection with small misalignments. We use the three cases listed in the three last raws of Tab. 2, and a misalignment of 5%; *i.e.* corresponding to systems with settings $c_2 = 0.05c_3$. Green curves represent the proposed model, and the red ones represent the approximations using previous methods. As we can see, the red curves are, in some cases, significantly far from the correct point projection (computed using [2]) that lies on top of our unified model.

model, and use our unified model. Results are shown in Fig. 3. Two sets of three parallel 3D lines are used in each system. To validate our model, we sample a small number of points from the 3D lines and project them into the image using [2]. We see that the images of these 3D points lie on the projection curves from Thm. 3, validating our results.

**Comparisons with previous modelings:** In addition to the validation experiments, we show how our model can be used to represent systems that the current techniques can only approximate. As described in the introduction, it is impossible to have a perfectly aligned camera with the axis of symmetry or have it at a specific distance. To test the importance of these deviations and the advantages of the use of the proposed unified model, we run a test using three of the four systems[4] in which the specific models were derived in the literature. Namely central cases [4, 18] and axial cone [7, 8]). The settings are listed in the three last rows of Tab. 2. Then, we consider small misalignment. Specifically, we consider the cases in which the perspective camera deviates from the perfect system requirement position.

Figure 4 shows the results with misalignments in the $y$–axis of only 5% with respect to the distance of the camera to the mirror (instead of $c_2 = 0$, we have $c_2 = 0.05c_3$). In addition, we run an experiment that generates 300 points in the

[4] In the spherical case we can always align the camera to get $c_2 = 0$.

| System | Misalignment | | | |
|---|---|---|---|---|
| | 1% | 5% | 10% | 15% |
| Hyperboloidal Central | 2.6 | 12.5 | 26.4 | 41.3 |
| Ellipsoidal Central | 3.6 | 16.7 | 36.2 | 64.9 |
| Axial Cone | 11.2 | 58.0 | 119.1 | 181.7 |

Table 3. **Misalignments:** Average of the distance in pixels from the projection of the 3D points and the approximated projection, by considering camera misalignment and previous modelings. A total of 300 points equally distributed in the environment were considered and their projection lines were considered.

world incident with a set of distinct 3D lines. We vary the misalignment from 1% of to 15% and compute the average error in pixels corresponding to the distance between the 3D projection of the points (exact projection) and the respective line projection approximated by the previous specific models[5], *i.e.,* setting $c_2 = 0$ and using perfect systems and previous models. The errors in pixels are shown in Tab. 3. The table concludes that the approximation of slightly misaligned systems needed for using previous models can significantly deteriorate the results. On the other hand, our unified model always gives the perfect fit.

### 5.2. Results with Real Data

We use a non-central catadioptric camera with a commercially available hyperbolical mirror and a FLIR Flea3, model FL3-U3-13E4C-C, with a resolution of 1280x1024. The perspective camera was previously calibrated, and the mirror's manufacturer gave its parameters. We use [45] to get transformation between the mirror and camera.

Four images are used to evaluate our unified model. First, we run an edge detection to extract candidate pixels to line projections. We compute a 3D line using a straightforward RANSAC cycle, in which, for each iteration:

1. We sample a set of four pixels that are candidates for images of a line;
2. Compute the four inverse projection rays corresponding to each of the four pixels obtained in item 1;
3. Using [54][6], we compute the 3D line that passes through four lines inverse projection rays of item 2. This 3D line will be the hypothesis for the 3D line coordinates obtained from the four points in item 1;
4. Using the 3D line estimated by item 3, we compute its projection $\mathcal{I}(u, v) = 0$, using Thm. 3;
5. We do inlier counting using $\mathcal{I}(u, v) = 0$ and computing a distance to the remaining pixels in the image listed as potential line images. The ones with a distance smaller than a defined threshold are considered inliers. This distance is calculated as in Sec. 5.1, paragraph "Comparisons with previous modelings."

We repeat the process a certain number of iterations, and the final 3D line estimate is given by the hypothesis that obtained the largest number of inliers. Then, we apply our 3D projection line modeling again to the resulting 3D line model. By running the previously defined RANSAC cycle multiple times (and removing inliers from previous runs), we can extract multiple 3D lines. Figure 5(a) shows the results of fitting multiple lines with $\mathcal{I}(u, v) = 0$.

The same technique was applied to new images of a chessboard in different positions. In this case, we apply corner detection, select the corners within the chessboard, define the chessboard's rows and columns individually, and run the pipeline described in the previous paragraph. The results are shown in Fig. 5(b).

This section shows that our unified imaging model obtains the correct projection of a 3D line and that, in combination with [54], our model can be used to fit multiple line images. We ran additional experiments with the RANSAC-based line fitting, namely synthetic data, evaluating different noise levels, specific camera systems, and RANSAC 2D vs. 3D. Due to space limitations, we send these experiments in supplementary materials. Note that these results could be refined using nonlinear optimization using the detected inliers, which we leave for future work.

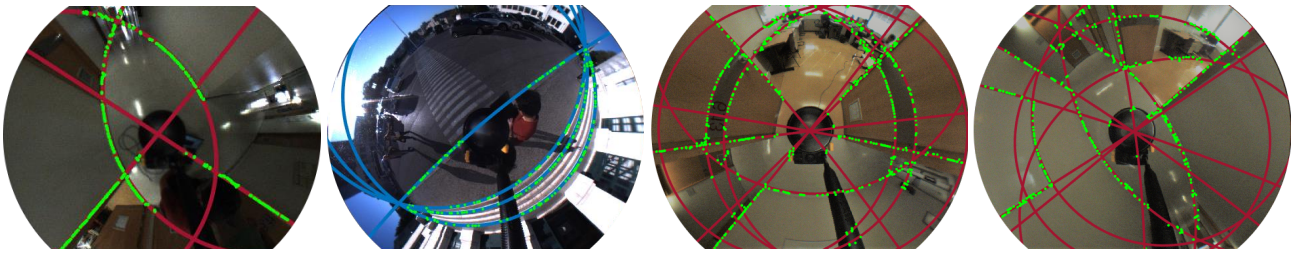
### 5.3. Applications to Pose Estimation and AR

We propose a simple application for a camera localization problem and Augmented Reality (AR) to evaluate our methods further. We use a non-central catadioptric camera with a spherical mirror, calibrated as described in the previous subsection. Four green 3D straight lines are placed on for floor, and we consider the following procedure:

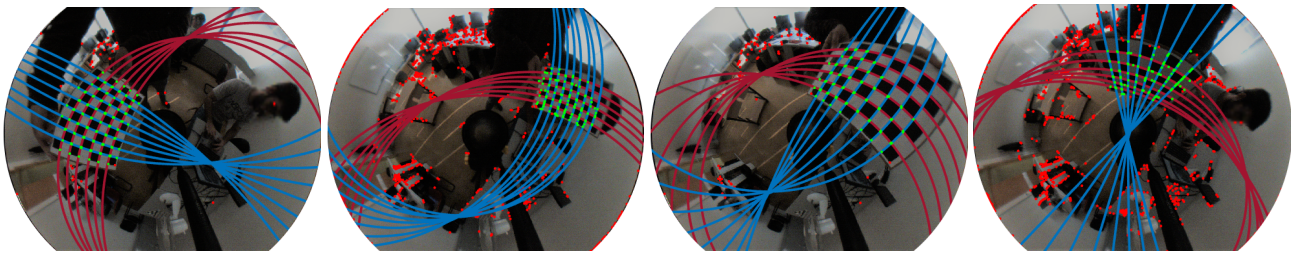
1. We run a color filter to get green pixels in the image;
2. Four projection curves are estimated using our RANSAC-based fitting method described in the previous section. Figure 6(a) shows the fitted curves;
3. We compute the intersecting points of the projection curves;
4. Using three of the four intersecting points and their respective coordinates in the world, we compute the camera pose using minimal data [13, 30, 39, 42], getting up to four solutions for the camera pose; and
5. We get the correct transformation by selecting the pose that minimizes the fourth point's re-projection error.

[5]For the computation of this distance, we find the closest image point on the line image projection (discretized) using `pdist2` from `Matlab`.

[6]In future work, we will explore the possibility of using our implicit parameterization to represent the surface corresponding to the inverse projection of the line image.

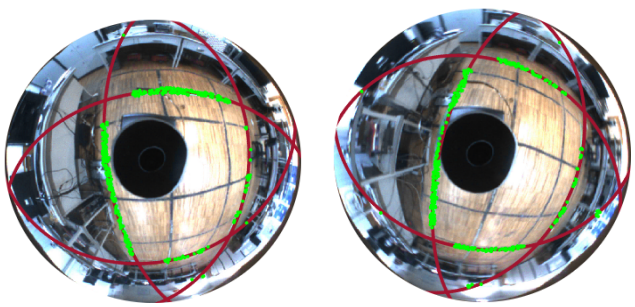


(a) Results for potential images of 3D straight lines given by edge detection.

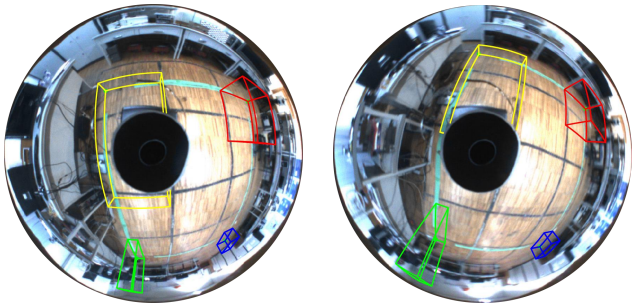


(b) Results for images of 3D straight lines given corner detection in a chessboard.

Figure 5. Line projection and fitting techniques using a non-central hyperbolic catadioptric camera. Images were acquired outside and inside. Red or blue curves represent the projection curves, while green points are the inliers considered to fit each curve.



(a) We show curves representing the 3D projection lines in spherical catadioptric cameras. Green points are the image point candidates for line images.



(b) We estimate the 3D line using the RANSAC-based method in Sec. 5.2, compute their intersections, and the camera's pose. Using the estimated pose, we offer a simple augmented reality application.

Figure 6. (a) shows two images acquired with a spherical catadioptric camera, four green 3D straight lines on the floor, and the respective four-line images. (b) shows the augmented reality application. Four parallelepiped objects are correctly projected into the image.

To evaluate our results, we define four rectangular parallelepipeds in the world and project their edges into the image using our model. The results are shown in Fig. 6(b). We conclude that the parallelepipeds are correctly projected.

## 6. Discussion

This paper proposes the first unified model for 3D line projection in catadioptric cameras. We start by describing the line-reflection constraint. Then, derive the reflection curve on the mirror and get a projection curve in the image. We ran several experiments with both synthetic & real data and different types of catadioptric camera systems.

For future work, we plan to i) derive a unified method for catadioptric camera calibration using line projections and ii) add geometrical constraints for improving the RANSAC-based line fitting for a better 3D and 2D estimation.

## Acknowledgements

This work was supported by the LARSyS – FCT Project UIDB/50009/2020, by the European Regional Development Fund, under the Smart Growth Operational Programme, project POIR.01.01.01-00-0102/20, with the title "Development of an innovative, autonomous mobile robot for retail stores", by the Swedish Research Council (grant 2018-05375), the Swedish Foundation for Strategic Research (Semantic Mapping and Visual Navigation for Smart Robots), and the Wallenberg AI, Autonomous Systems and Software Program (WASP). We thank the reviewers and area chairs for valuable feedback.

## References


[1] Amit Agrawal, Yuichi Taguchi, and Srikumar Ramalingam. Analytical forward projection for axial non-central dioptric and catadioptric cameras. In *European Conf. Computer Vision (ECCV)*, pages 129–143, 2010. 1, 2, 3

[2] Amit Agrawal, Yuichi Taguchi, and Srikumar Ramalingam. Beyond Alhazen's problem: Analytical projection model for non-central catadioptric cameras with quadric mirrors. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 2993–3000, 2011. 1, 2, 3, 6

[3] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. *Int'l J. Computer Vision (IJCV)*, 35(2):175–196, 1999. 1, 2

[4] Joao P. Barreto and Helder Araujo. Issues on the geometry of central catadioptric image formation. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 422–427, 2001. 1, 2, 3, 6

[5] Joao P. Barreto and Kostas Daniilidis. Epipolar geometry of central projection systems using veronese maps. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 1258–1265, 2006. 1

[6] Jean Charles Bazin, Cedric Demonceaux, and Pascal Vasseur. Fast central catadioptric line extraction. In *Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA)*, pages 25–32, 2007. 2

[7] J. Bermudez-Cameo, G. Lopez-Nicolas, and J.J. Guerrero. Line-images in cone mirror catadioptric systems. In *IEEE Int'l Conf. Pattern Recognition (ICPR)*, pages 2083–2088, 2014. 1, 2, 6

[8] Jesus Bermudez-Cameo, Gonzalo Lopez-Nicolas, and Jose J. Guerrero. Fitting line projections in non-central catadioptric cameras with revolution symmetry. *Computer Vision and Image Understanding (CVIU)*, 167:134–152, 2018. 1, 2, 6

[9] J. Bermudez-Cameo, Luis Puig, and J.J. Guerrero. Hypercatadioptric line images for 3d orientation and image rectification. *Robotics and Autonomous Systems (RAS)*, 60(6):755–768, 2012. 2

[10] Roland Bunschoten and Ben Krose. Robust scene reconstruction from an omnidirectional vision system. *IEEE Trans. Robotics and Automation (T-RA)*, 19(2):351–357, 2002. 1

[11] Vincenzo Caglioti and Simone Gasparini. On the localization of straight lines in 3d space from single 2d images. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 1129–1134, 2005. 1

[12] David Caruso, Jakob Engel, and Daniel Cremers. Large-scale direct slam for omnidirectional cameras. In *IEEE/RSJ Int'l Conf. Intelligent Robots and Systems (IROS)*, pages 141–148, 2015. 1

[13] Chu-Song Chen and Wen-Yan Chang. On pose recovery for generalized visual sensors. *IEEE Trans. Pattern Analysis and Machine Intelligence (T-PAMI)*, 26(7):848–861, 2004. 7

[14] Yuanyuan Ding, Jingyi Yu, and Peter Sturm. Recovering specular surfaces using curved line images. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 2326–2333, 2009. 2

[15] O. D. Faugeras, Q.-T. Luong, and S. J. Maybank. Camera self-calibration: Theory and experiments. In *European Conf. Computer Vision (ECCV)*, pages 321–334, 1992. 1

[16] Andrew W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–I, 2001. 2

[17] Simone Gasparini and Vincenzo Caglioti. Line localization from single catadioptric images. *Int'l J. Computer Vision (IJCV)*, 94(3):361–374, 2011. 1

[18] Christopher Geyer and Kostas Daniilidis. A unifying theory for central panoramic systems and practical implications. In *European Conf. Computer Vision (ECCV)*, pages 445–461, 2000. 1, 2, 6

[19] Raja N. Ghanem, Charulatha Ramanathan, Ping Jia, and Yoram Rudy. Heart-surface reconstruction and ecg electrodes localization using fluoroscopy, epipolar geometry and stereovision: Application to noninvasive imaging of cardiac electrical activity. *IEEE Trans. Medical Imaging (T-MI)*, 22(10):1307–1318, 2003. 1

[20] Joshua Gluckman, Shree K. Nayar, and Keith J. Thoresz. Real-time omnidirectional and panoramic stereo. In *DARPA Image Understanding Workshop*, pages 299–303, 1998. 1

[21] Nuno Goncalves. On the reflection point where light reflects to a known destination on quadratic surfaces. *OSA Opt. Lett.*, 35(2):100–102, 2010. 2, 3

[22] Nuno Goncalves and Ana Catarina Nogueira. Projection through quadric mirrors made faster. In *IEEE Int'l Conf. Computer Vision Workshops*, pages 2141–2148, 2009. 2

[23] Nuno Goncalves, Ana Catarina Nogueira, and Andre Lages Miguel. Forward projection model of non-central catadioptric cameras with spherical mirrors. *Robotica*, 35(6):1378–1396, 2017. 2

[24] Michael D. Grossberg and Shree K. Nayar. A general imaging model and a method for finding its parameters. In *IEEE Int'l Conf. Computer Vision (ICCV)*, volume 2, pages 108–115, 2001. 1

[25] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, 2004. 1, 2, 3

[26] Richard I. Hartley. In defense of the eight-point algorithm. *IEEE Trans. Pattern Analysis and Machine Intelligence (T-PAMI)*, 19(6):580–593, 1997. 1

[27] Fay Huang. Epipolar geometry in concentric panoramas. Technical report, CMP, Czech Technical University – CTU-CMP-2000-07, 2000. 2

[28] Sing Bing Kang. Catadioptric self-calibration. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 201–207, 2000. 1

[29] Douglas Lanman, Megan Wachs, Gabriel Taubin, and Fernando Cukierman. Reconstructing a 3d line from a single catadioptric image. In *Int'l Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, pages 89–96, 2006. 1

[30] Gim Hee Lee, Bo Li, Marc Pollefeys, and Friedrich Fraundorfer. Minimal solutions for the multi-camera pose estimation problem. *The International Journal of Robotics Research (IJRR)*, 34(7):837–848, 2015. 7

[31] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An Invitation to 3-D Vision*. Springer-Verlag, 2004. 1, 2

[32] Christopher Mei and Ezio Malis. Fast central catadioptric line extraction, estimation, tracking and structure from motion. In *IEEE/RSJ Int'l Conf. Intelligent Robots and Systems (IROS)*, pages 4774–4779, 2006. 2

[33] Christopher Mei and Patrick Rives. Single view point omnidirectional camera calibration from planar grids. In *IEEE Int'l Conf. Robotics and Automation (ICRA)*, pages 3945–3950, 2007. 2

[34] Marc Menem and Tomas Pajdla. Constraints on perspective images and circular panoramas. In *British Machine Vision Conference (BMVC)*, 2004. 2

[35] Branislav Micusik and Tomas Pajdla. Autocalibration & 3d reconstruction with non-central catadioptric cameras. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 58–65, 2004. 1

[36] Branislav Micusik and Tomas Pajdla. Para-catadioptric camera auto-calibration from epipolar geometry. In *Asian Conf. Computer Vision (ACCV)*, volume 2, pages 748–753, 2004. 1

[37] Branislav Micusik and Tomas Pajdla. Structure from motion with wide circular field of view cameras. *IEEE Trans. Pattern Analysis and Machine Intelligence (T-PAMI)*, 28(7):1135–1149, 2006. 1

[38] Ondrej Miksik, Yousef Amar, Vibhav Vineet, Patrick Perez, and Philip H. S. Torr. Incremental dense multi-modal 3d scene reconstruction. In *IEEE/RSJ Int'l Conf. Intelligent Robots and Systems (IROS)*, pages 908–915, 2015. 1

[39] Pedro Miraldo and Helder Araujo. A simple and robust solution to the minimal general pose estimation. In *IEEE Int'l Conf. Robotics and Automation (ICRA)*, pages 2119–2125, 2014. 7

[40] Pedro Miraldo, Helder Araujo, and Joao Queiro. Point-based calibration using a parametric representation of the general imaging model. In *IEEE Int'l Conf. Computer Vision (ICCV)*, pages 2304–2311, 2011. 1

[41] Shree K. Nayar. Catadioptric omnidirectional camera. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 482–488, 1997. 1

[42] David Nister. A minimal solution to the generalised 3-point pose problem. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 560–567, 2004. 7

[43] David Nister, Oleg Naroditsky, and James Bergen. Visual odometry. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 652–659, 2004. 1

[44] Tomas Pajdla. Stereo with oblique cameras. *Int'l J. Computer Vision (IJCV)*, 47(1):161–170, 2002. 2

[45] Luis Perdigoto and Helder Araujo. Calibration of mirror position and extrinsic parameters in axial non-central catadioptric systems. *Computer Vision and Image Understanding (CVIU)*, 117(8):909–921, 2013. 7

[46] Marc Pollefeys, Reinhard Koch, and Luc Van Goo. Self-calibration and metric reconstruction in spite of varying and unknown intrinsic camera parameters. *Int'l J. Computer Vision (IJCV)*, 32(1):7–25, 1999. 1

[47] Marc Pollefeys, Reinhard Koch, and Luc Van Goo. A simple and efficient rectification method for general motion. In *IEEE Int'l Conf. Computer Vision (ICCV)*, volume 1, pages 496–501, 1999. 1

[48] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A toolbox for easily calibrating omnidirectional cameras. In *IEEE/RSJ Int'l Conf. Intelligent Robots and Systems (IROS)*, pages 5695–5701, 2006. 2

[49] Peter Sturm, , and Srikumar Ramalingam. A generic concept for camera calibration. In *European Conf. Computer Vision (ECCV)*, pages 1–13, 2004. 1

[50] Peter Sturm and Joao P. Barreto. General imaging geometry for central catadioptric cameras. In *European Conf. Computer Vision (ECCV)*, pages 609–622, 2008. 1

[51] Peter Sturm, Srikumar Ramalingam, Jean-Philippe Tardif, Simone Gasparini, and Joao Barreto. Camera models and fundamental concepts used in geometric computer vision. *Foundations and Trends in Computer Graphics and Vision*, 6(1/2):1–183, 2011. 1

[52] Tomas Svoboda and Tomas Pajdla. Epipolar geometry for central catadioptric cameras. *Int'l J. Computer Vision (IJCV)*, 49(1):23–37, 2002. 1

[53] Rahul Swaminathan, Michael D. Grossberg, and Shree K. Nayar. Non-single viewpoint catadioptric cameras: Geometry and analysis. *Int'l J. Computer Vision (IJCV)*, 66(3):211–229, 2006. 1, 2

[54] Seth Teller and Michael Hohmeyer. Determining the lines through four lines. *J. Graphics Tools*, 4(3):11–22, 1999. 1, 7

[55] SriRam Thirthala and Marc Pollefeys. The radial trifocal tensor: a tool for calibrating the radial distortion of wide-angle cameras. In *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 321–328, 2005. 2

[56] Zhiyu Xiang, Xing Dai, Yanbing Zhou, and Xiaojin Gong. Self-calibration for a non-central catadioptric camera with approximate epipolar geometry. *Meas. Sci. Technol.*, 25:085005, 2014. 1

[57] Wei Yang, Yu Ji, Jinwei Ye, S. Susan Young, and Jingyi Yu. Coplanar common points in non-centric cameras. In *European Conf. Computer Vision (ECCV)*, pages 220–233, 2014. 2

[58] Jinwei Ye, Yu Ji, and Jingyi Yu. A rotational stereo model based on xslit imaging. In *IEEE Int'l Conf. Computer Vision (ICCV)*, pages 489–496, 2013. 2

[59] Jinwei Ye and Jingyi Yu. Ray geometry in non-pinhole cameras: a survey. *The Visual Computer (Vis Comput)*, 30(1):93–112, 2013. 1