HAL
open science

Multi-View Matching Tensors from Lines for General Camera Models

Simone Gasparini, Peter Sturm

To cite this version:

Simone Gasparini, Peter Sturm. Multi-View Matching Tensors from Lines for General Camera Models. Santiago Aja-Fernandez and Rodrigo de Luis Garcia and Dacheng Tao and Xuelong Li. Tensors in Image Processing and Computer Vision, 33, Springer, pp.197-214, 2009, Advances in Pattern Recognition, 10.1007/978-1-84882-299-3_9 . inria-00384237

HAL Id: inria-00384237

https://inria.hal.science/inria-00384237

Submitted on 14 May 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-View Matching Tensors from Lines for General Camera Models

Simone Gasparini and Peter Sturm

Abstract General camera models relax the constraint on central projection and characterize cameras as mappings between each pixel and the corresponding projection rays. This allows to describe most cameras types, including classical pinhole cameras, cameras with various optical distortions, catadioptric cameras and other acquisition devices. We deal with the structure from motion problem for such general models. We first consider an hierarchy of general cameras first introduced in [28] where the cameras are described according to the number of points and lines that have a non-empty intersection with all the projection rays. Then we propose a study of the multi-view geometry of such cameras and a new formulation of multi-view matching tensors working for projection rays crossing the same 3D line, the counterpart of the fundamental matrices and the multifocal tensors of the standard perspective cameras. We also delineate a method to estimate such tensors and recover the motion between the views.

1 Introduction

Tensors have been widely used in the field of computer vision as they provide frameworks that conveniently represent the multi-view geometry of cameras, help the matching of features across views and, once estimated, they allow to compute the camera motion. Many works dealt with systems of central cameras and studied the geometric and algebraic relations between correspondences of points and lines in an arbitrary number of images, thus finding multifocal tensors [8, 16, 17] useful to

Simone Gasparini
INRIA Grenoble–Rhône-Alpes, 38334 Montbonnot St. Martin, France
e-mail: simone.gasparini@inrialpes.fr

Peter Sturm
INRIA Grenoble–Rhône-Alpes, 38334 Montbonnot St. Martin, France
e-mail: peter.sturm@inrialpes.fr

compute the intrinsic and extrinsic parameters of the cameras. Other efforts have been made on non conventional cameras that can still be modeled as a central projection [1], such as central catadioptric cameras (cameras in front of a curved mirror) [3, 12, 13, 29] or the mixture of them and standard perspective cameras [27].

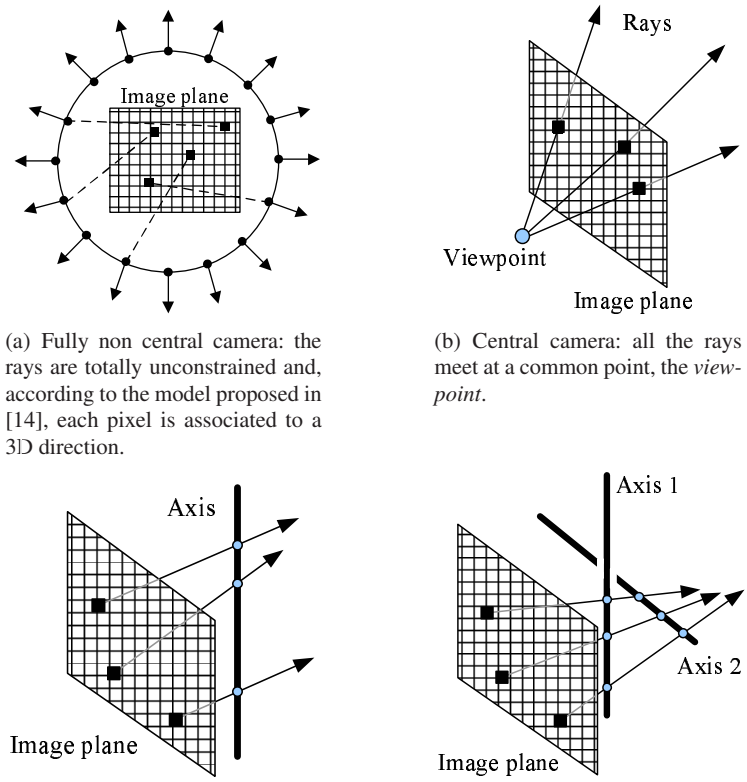
In this work we relax the constraint on central projection and we deal with a more general imaging model inspired by the one proposed by Grossberg and Nayar in [14], where the camera is modeled as a set of pixels that capture the light travelling along rays in 3D. Therefore the camera is fully described by the mapping of each pixel to the corresponding 3D ray expressed in any suitable reference frame. Such a general model can be used to represent cameras with various optical distortions (*e.g.* barrel, pincushion, wide-angle), camera clusters [24, 30], non central catadioptric cameras [2, 9, 18, 22], and other special acquisition devices such as rotating cameras [20, 23, 26] or the so-called *compound cameras* [10, 11] that try to emulate insect eyes.

Some efforts have been done in order to define the multiview geometry of such cameras. Thirthala and Pollefeys [32] proposed a study of the geometry of 1D radial cameras that allows to model some central and non central cameras, such as pinhole cameras, low distortion cameras, fish-eye cameras and catadioptric cameras. They developed a quadrifocal tensor working on uncalibrated images that allows to relate the features seen across four views.

A general approach that applies to a broader set of cameras has been proposed by Sturm in [28]: it introduced a new hierarchy of camera models that allows to describe the camera according to the number of points and lines that have a non-empty intersection with all the projection rays of the camera. Then, assuming known the camera calibration, multi-view matching tensors have been developed by using corresponding projection rays in multiple views of **fully non central** cameras, in which the rays are unconstrained, **axial** cameras, in which all the rays cross a common line, **x-slit** cameras [34], in which all the rays cross two common lines, and **central** ones in which all the rays meet at a common point (see Figure 1).

In this work we use such an hierarchy and we extend the multi-view geometry of general camera models by developing novel multi-view matching tensors for line images, *i.e.* that work on projection rays that cross the same 3D line. By imposing that the projection rays associated to the images of the same 3D line across different views cross a common line in space, we propose the theoretical formulation and a complete characterization of the matching tensors for any kind of cameras described by the model. Using line correspondences can have some advantages: lines can be detected with a better accuracy than points (*e.g.* via fitting and interpolation algorithms) and, above all, they are not affected by (partial) occlusions that may occur between the views. However, it is worth noting that we follow a different approach w.r.t. other classical multifocal formulations using image line correspondences [8, 16]: once the line correspondences are available we work with the projection rays of pixels on line images, that are known by calibration.

The paper is organized as follows. In Section 2 and Section 3 we briefly recall Plücker coordinates and some useful properties of lines in space that will be used throughout the paper; Section 4 introduces the formulation of the multifocal ten-



(a) Fully non central camera: the rays are totally unconstrained and, according to the model proposed in [14], each pixel is associated to a 3D direction.

(b) Central camera: all the rays meet at a common point, the *viewpoint*.

(c) Axial camera: all the rays cross a common line, the *axis* of the camera.

(d) X-Slit camera: all the rays cross two common lines, the *axes* of the camera.

Fig. 1 The hierarchy of cameras introduced by Sturm in [28] that describes the camera according to the number of points and lines that have a non-empty intersection with all the projection rays.

sor for the general camera models which is then derived for other specific camera models in Section 5. Section 6 presents a method to estimate the tensors and some preliminary results obtained with axial cameras. Section 7 concludes the paper.

2 Plücker Coordinates

For our purposes, we represent the rays and lines in 3D space via Plücker coordinates, as it is a convenient representation for lines in space, widely used in many problems involving lines in computer graphics, computational geometry and computer vision.

Given two 3D points $\mathbf{A}$ and $\mathbf{B}$, the line $\mathbf{L}$ joining them can be expressed (up to scale) via the Plücker coordinate vector of length 6

$$\mathbf{L} = \left[(\mathbf{A}-\mathbf{B})^\mathsf{T} \quad (\mathbf{A}\times\mathbf{B})^\mathsf{T}\right]^\mathsf{T} \tag{1}$$

Given two lines $\mathbf{L}_1$ and $\mathbf{L}_2$, their *side* product is defined as

$$side\,(\mathbf{L}_1,\mathbf{L}_2) \triangleq \mathbf{L}_1^\mathsf{T} \begin{bmatrix} \mathtt{0} & \mathtt{I} \\ \mathtt{I} & \mathtt{0} \end{bmatrix} \mathbf{L}_2, \tag{2}$$

that is zero if they intersect or are parallel, and non-zero otherwise. Any 6-vector $\mathbf{L}$ corresponds to a "real" 3D line if and only if it satisfies the constraint $side\,(\mathbf{L},\mathbf{L}) = 0$.

Let us consider a 3D point $\mathbf{Q}$ and a rototranslation $\mathtt{P}$ defined by a rotation $\mathtt{R}$ and a translation $\mathbf{t}$ so that

$$\mathbf{Q}' \rightarrow \mathtt{P}\mathbf{Q} = \begin{bmatrix} \mathtt{R} & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix} \mathbf{Q};$$

the Plücker line representation is then transformed according to the transformation matrix $\hat{\mathtt{P}}$

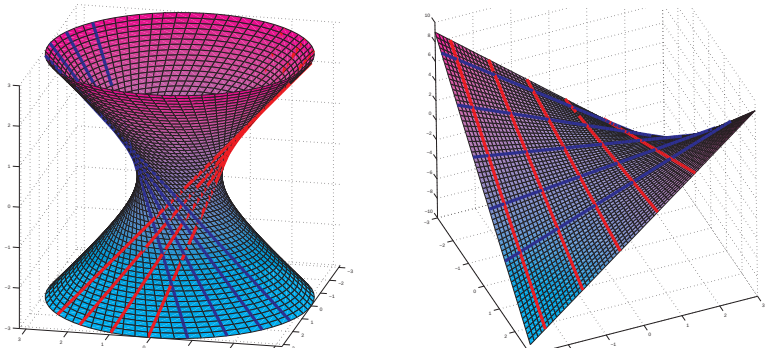
$$\mathbf{L}' \rightarrow \hat{\mathtt{P}}\mathbf{L} = \begin{bmatrix} \mathtt{R} & \mathtt{0} \\ -[\mathbf{t}]_\times \mathtt{R} & \mathtt{R} \end{bmatrix} \mathbf{L}. \tag{3}$$

3 Lines in Space

In [28] the multi-view geometry was defined by using matching rays, *i.e.* the rays associated to the matching points, and imposing the constraint that they meet at a point in space. In this work we do not need to match the projection rays, but we use rays associated to pixels lying on matching line images. It is not necessary that pixels in different images match one another, *i.e.* that they correspond to the same point on the 3D line. We show how to build multifocal tensors by imposing the constraint that they cross a common line in space. To this end, we express the projection rays in Plücker coordinates so that such a constraint can be conveniently expressed using the *side* relation (2): since each given projection ray $\mathbf{L}_n$ crosses the common space line $\mathbf{L}$ (which is, of course, unknown), the relation $side(\mathbf{L}_i,\mathbf{L}) = 0$ must hold for all i, thus leading to the following linear system

$$\begin{bmatrix} \mathbf{L}_1^\mathsf{T} \\ \vdots \\ \mathbf{L}_n^\mathsf{T} \end{bmatrix} \begin{bmatrix} \mathtt{0} & \mathtt{I} \\ \mathtt{I} & \mathtt{0} \end{bmatrix} \mathbf{L} = \mathtt{M}\mathbf{L} = \mathbf{0}. \tag{4}$$

The problem of finding the transversal $\mathbf{L}$ of a set of rays $\mathbf{L}_i$ has been studied in [31] for computer graphics applications and then in [7, 21] to reconstruct lines from single, non central, catadioptric images. Degenerate configurations of rays may occur,



(a) Hyperboloid of one sheet (b) Hyperbolic paraboloid

Fig. 2 Doubly-ruled quadrics are surfaces composed by two families of lines so that each line of one family (*e.g*. the blue ones) crosses all the lines of the other family (*e.g*. the red ones) and the lines of the same family are mutually skew. If the rays lie on one of the family, then there are infinitely many lines crossing them and there is no unique solution to the system (4).

in which there is no unique solution to the system (4). The six rays could lie, *e.g*., on the same plane, so that there will be infinitely (more precisely, ∞^2) many lines crossing all of them; thus the system (4) is of rank 2 because only two of the rays are enough to define the plane and all the others can be seen as a linear combination.

On the other hand, the rays could also lie on a double ruled quadric, a surface composed by two families of lines so that each line of one family crosses all the lines of the other family and the lines of the same family are mutually skew. There are only two double ruled quadrics, the hyperboloid of one sheet and the hyperbolic paraboloid (see Figure 2), while the plane is also a degenerate case of ruled quadric. In this case, there will be infinitely (more precisely, ∞^1) many lines crossing all of them; the system (4) will be of rank 3 because any three skew lines define a ruled quadric [19].

Finally, given four lines in space in general position, there could be 0, 1, 2 or, again, an infinite number of lines intersecting them [5]. All these degenerate configurations may occur according to the camera model considered and the displacement of the line $\mathbf{L}$ in space w.r.t. the camera. As we consider rays coming from different views of general cameras, we can likely assume that the linear system has just one (non-trivial) solution $\mathbf{L}$ defined up to scale by the 1-dimensional null space of $\mathtt{M}$. We will discuss (possible) degenerate configurations that may occur in Section 5 when we derive the multi-view tensors for specific camera models.

4 Multi-Focal Tensors from Lines

In this section we define multi-focal tensors for general cameras. The cameras are supposed to be calibrated, *i.e*. for each pixel of an image the associated 3D ray is

known and expressed in a local metric coordinate frame. Then we apply rotations and translations to put all cameras in a global coordinate system.

The uncalibrated case has already been treated in the case of central perspective cameras, as calibrated and uncalibrated images are linked together by 2D projective transformations. In the case of non central cameras there is not, in general, an analytical relationship among the pairs of pixels and projection rays as they could be completely independent or related by a transformation that depends on the specific camera model.

We first consider the most general model in which the pairs pixel and ray are totally unconstrained and then we derivate the results for other models than the fully non-central one, where the camera rays are constrained to meet a common point (central cameras), a line (axial cameras) or two skew lines (x-slit). As in [28], instead of projection matrices used in perspective cameras, we consider pose matrices – rototranslation matrices – associated to each view i that map space points from some global reference into each camera local reference. Then, we consider a set of 6 rays coming from the views and crossing the space line $\mathbf{L}$: Table 1 reports useful choices of rays from each camera according to the number of views. From now on we call $\mathbf{a}, \mathbf{b}, \ldots \mathbf{f}$ the 6 rays and $\mathtt{A}, \mathtt{B} \ldots \mathtt{F}$ the relevant rototranslation matrices (*cf*. (3)). We also denote with A_i^j the element of the i-th row and j-th column of $\mathtt{A}$ and with $\mathbf{A}_i$ ($\mathbf{A}^j$) its i-th row (j-th column). Thus, for each ray, *e.g*. $\mathbf{a}$, combining (2) and (3) the side relation becomes:

$$\mathbf{a}^\top \begin{bmatrix} \mathtt{0} & \mathtt{I} \\ \mathtt{I} & \mathtt{0} \end{bmatrix} \hat{\mathtt{A}}\,\mathbf{L} = \mathbf{a}^\top \bar{\mathtt{A}}\,\mathbf{L} = 0, \tag{5}$$

where $\bar{\mathtt{A}}$ is the remapped pose matrix. Then the linear system (4) becomes:

$$\begin{bmatrix} \mathbf{a}^\top \bar{\mathtt{A}} \\ \vdots \\ \mathbf{f}^\top \bar{\mathtt{F}} \end{bmatrix} \mathbf{L} = \mathtt{M}\,\mathbf{L} = \mathbf{0}. \tag{6}$$

The 6×6 matrix $\mathtt{M}$ must be rank deficient since it has a null vector, $\mathbf{L}$: we can use the constraint $\det \mathtt{M} = 0$ in order to determine a matching tensor $\mathtt{T}$. In the following we show that this constraint can be written in terms of the rays' Plücker coordinates in the form

$$\sum_{q_1=1}^{6} \sum_{q_2=1}^{6} \cdots \sum_{q_6=1}^{6} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6}\, T_{q_1,\ldots q_6} = 0, \tag{7}$$

where $T_{q_1,\ldots q_6}$ is the element of the $6 \times 6 \times 6 \times 6 \times 6 \times 6$ multifocal tensor. The constraint $\det(\mathtt{M}) = 0$ can be expressed in Einstein's notation as

$$\det(\mathtt{M}) = \varepsilon_{ijklmn}\, M_1^i\, M_2^j\, M_3^k\, M_4^l\, M_5^m\, M_6^n = 0,$$

where, *e.g*., $M_1^i = \mathbf{a}^\top \bar{\mathbf{A}}^1$ is the element in the first row of the i-th column of $\mathtt{M}$, so that

$$\varepsilon_{ijklmn}\ \mathbf{a}^\top \bar{\mathbf{A}}^i\ \mathbf{b}^\top \bar{\mathbf{B}}^j\ \mathbf{c}^\top \bar{\mathbf{C}}^k\ \mathbf{d}^\top \bar{\mathbf{D}}^l\ \mathbf{e}^\top \bar{\mathbf{E}}^m\ \mathbf{f}^\top \bar{\mathbf{F}}^n = 0.$$

#views	Case	Non central	Central	Axial	X-Slit
2	5-1	✓		✓	
	4-2	✓		✓	
	3-3	✓		✓	✓
3	4-1-1	✓			
	3-2-1	✓		✓	✓
	2-2-2	✓	✓	✓	✓
4	3-1-1-1	✓		✓	✓
	2-2-1-1	✓	✓	✓	✓
5	2-1-1-1-1	✓	✓	✓	✓
6	1-1-1-1-1-1	✓	✓	✓	✓

Table 1 For each camera model, each n-tuple in the second column indicates the possible choices of rays from the n views, that lead to a non trivial matching constraint.

Each term of the previous sum is composed in turn of a sum of 6^6 elements; since for each ray it holds $\mathbf{a}^\top \bar{\mathbf{A}}^j = \sum_i a^i \bar{A}^j_i$ and the product of summations is the sum of products, each term can also be written as

$$\sum_{q_1=1}^{6} \dots \sum_{q_6=1}^{6} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6} \bar{A}^i_{q_1} \bar{B}^j_{q_2} \bar{C}^k_{q_3} \bar{D}^l_{q_4} \bar{E}^m_{q_5} \bar{F}^n_{q_6}.$$

It can be noted that any choice of indices $ijklmn$ does not affect the first part of the product involving the Plücker coordinates of the rays and only the term involving the pose matrices $A^i_{q_1} B^j_{q_2} \dots F^n_{q_6}$ changes. So, finally, the constraint can be written as

$$\det \mathtt{M} = \sum_{q_1=1}^{6} \dots \sum_{q_6=1}^{6} a^{q_1} \dots f^{q_6} \left(\varepsilon_{ijklmn} \bar{A}^i_{q_1} \bar{B}^j_{q_2} \dots \bar{F}^n_{q_6} \right) = \dots$$
$$\dots = \sum_{q_1=1}^{6} \dots \sum_{q_6=1}^{6} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6} \; T_{q_1,\dots q_6} = 0,$$

where $\mathtt{T}$ is the $6 \times 6 \times 6 \times 6 \times 6 \times 6$ tensor relating the projection rays. It is worth to note that, for any given choice of indices $q_w \;\; w \in \{1, \dots 6\}$, the sum $\varepsilon_{ijklmn} \bar{A}^i_{q_1} \bar{B}^j_{q_2} \dots \bar{F}^n_{q_6}$ is the determinant of the 6×6 matrix $\mathtt{S}_{q_1,\dots,q_6}$ obtained by assembling together each q_w-th row from the corresponding pose matrix, *i.e.*

$$T_{q_1,\dots q_6} = \det \mathtt{S}_{q_1,\dots,q_6} = \det \begin{bmatrix} \bar{\mathbf{A}}_{q_1} \\ \bar{\mathbf{B}}_{q_2} \\ \vdots \\ \bar{\mathbf{F}}_{q_6} \end{bmatrix}, \tag{8}$$

which is, in general, a sum of $6! = 720$ terms, where $\mathbf{A}_{q_1}$ is the q_1-th row of the matrix $\mathtt{A}$, according to the notation introduced before.

This expression allows to compute the elements of the tensor in the most general case in which each of the 6 rays belongs to a different view. It varies straightforwardly according to the number of views and the chosen case (*cf*. Table 1) where effectively the number of pose matrices involved varies.

The size of the tensor is very large, but some of its elements are zero. Consider the structure of the matrices $\bar{\mathtt{A}}\ldots\bar{\mathtt{F}}$ containing the pose information: if one of the indices $q_w > 3 \;\; w \in \{1,\ldots 5\}$, then the last three elements of the corresponding w-th row in $\mathtt{S}_{q_1,\ldots,q_6}$ are zero (due to the $\mathtt{0}_{3\times 3}$ submatrix in the pose matrices (3)). It follows that $T_{q_1,\ldots,q_6} = \det \mathtt{S}_{q_1,\ldots,q_6} = 0$ if the values of at least 4 of the indices q_w are greater than 3: in such cases, indeed, the last 3 columns of $\mathtt{S}_{q_1,\ldots,q_6}$ are not linearly independent, hence the determinant is zero. Moreover, if there are more than one ray belonging to the same view, the indices q_w that are associated to those rays, must have distinct values, otherwise the corresponding rows of $\mathtt{S}$ are identical, hence the determinant is zero.

Without loss of generality, we can also assume that the global coordinate system coincides with the first camera's local coordinate system, *i.e*. the first camera pose matrix $\mathtt{A}$ is the identity matrix. Hence, the corresponding rows in $\mathtt{S}$ contain just one non-zero element. In order to determine the exact number of non-zero elements of $\mathtt{T}$ for a given case, one should solve a combinatorial problem which takes into account all the combinations of the indices q_w for which at least 4 of the indices q_w are greater than 3 and the indices associated to the same view have distinct values, up to a permutation.

Finally, from the structure of the matrices $\mathtt{A}\ldots\mathtt{F}$ we can observe that they contain 9 zero elements, and only 18 of the remaining elements are unique, since the elements of the rotation matrix $\mathtt{R}$ appear twice. Therefore, different elements of $\mathtt{T}$ may be identical (up to sign), as they can be derived from $\det \mathtt{S}$ with different combinations of the pose matrices' rows.

We collect in Table 2 the number of non-zero elements and the number of unique elements (up to sign) for each useful case according to the number of the views.

5 Constrained Camera Models

We consider now more specific camera models than fully non-central ones where the camera rays are no more unconstrained but are subject to a constraint, such as central, axial, and x-slit cameras. For each case, we briefly report the parameterizations introduced in [28] and then we study the possible cases (*cf*. Table 1) and the relevant properties of the tensor $\mathtt{T}$ (*cf*. Table 2).

Cases (*cf*. Table 1)	**Non Central**		**Central**				**Axial**				**X-Slit**			
			finite		infinite		finite		infinite		2 finite		finite+infinite	
5-1	3240	18	*n.a.*				600	5	240	2	*n.a.*			
4-2	7776	117	*n.a.*				2256	46	1104	22	*n.a.*			
3-3	10152	200	*n.a.*				3276	87	1620	41	576	16	432	12
4-1-1	9072	270	*n.a.*				2712	109	1344	52	*n.a.*			
3-2-1	14796	900	*n.a.*				5244	417	2724	207	1152	96	816	68
2-2-2	18360	1701	216	27	64	8	6912	824	3680	420	1728	216	1216	152
3-1-1-1	17496	2106	*n.a.*				6420	1022	3396	518	1518	253	1056	176
2-2-1-1	21708	4050	324	81	96	24	8460	2011	4564	1037	2268	566	1568	391
2-1-1-1-1	25758	9720	486	243	150	75	10404	4946	5696	2592	2988	1491	2048	1021
1-1-1-1-1-1	30618	27702	729	729	233	233	12825	12185	7120	6480	3942	3942	2688	2678

Table 2 For each camera model, the first column collects the number of non-zero elements in the tensor $\mathtt{T}$ for each possible case, while the second column collects the number of unique elements (up to sign) in the tensor $\mathtt{T}$ (the cases marked with *n.a.* give no useful constraint).

5.1 Central Cameras

Since all rays go through a single point, *i.e*. the optical center (possibly, at an infinite point), we can consider at most two projection rays for each view: any other projection ray in the same view would be a linear combination of the previous ones, hence no useful constraint can be exploited from the matrix $\mathtt{M}$ as it is always rank deficient. Therefore, and since we need 6 rays to build the tensor, two views do not provide useful constraint and do actually not allow a projective reconstruction, as stated in [33]. Table 1 reports the useful cases according to the number of views.

A degenerate configuration occurs when the cameras share a common baseline, *i.e*. the viewpoints are collinear. Then all the projection rays cross the baseline at the relevant viewpoints. Such a configuration can be seen as a particular case of axial cameras (*cf*. Section 5.2).

We distinguish the cases of a finite and infinite optical center.

5.1.1 Finite optical center

Setting the optical center as the origin $\mathbf{O}$ of the local coordinate system, each ray is described by a Plücker vector having the last three elements equal to zero, *i.e*. according to (1) each ray is of the form

$$\mathbf{a} = \begin{bmatrix} a_1 & a_2 & a_3 & 0 & 0 & 0 \end{bmatrix}^{\mathsf{T}}.$$

Hence the base size of the multi-focal tensor $\mathtt{T}$ can be reduced from 6 to 3, and (7) becomes

$$\det \mathtt{M} = \sum_{q_1=1}^{3} \dots \sum_{q_6=1}^{3} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6}\, T_{q_1,\dots,q_6} = 0.$$

so that only the first three rows of each pose matrix $\bar{\mathtt{A}}, \dots, \bar{\mathtt{F}}$ can contribute to $\mathtt{S}$.

5.1.2 Infinite optical center

Choosing a suitable coordinate system where the viewpoint $\mathbf{V}$ has (homogeneous) coordinates $\mathbf{V} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}^\mathsf{T}$, each ray is described by a Plücker vector of the form

$$\mathbf{a} = \begin{bmatrix} 0 & 0 & a_3 & a_4 & a_5 & 0 \end{bmatrix}^\mathsf{T}.$$

Again, the base size of the multi-focal tensor $\mathtt{T}$ can be reduced from 6 to 3 so that (7) becomes

$$\det \mathtt{M} = \sum_{q_1=3}^{5} \dots \sum_{q_6=3}^{5} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6}\, T_{q_1,\dots,q_6} = 0.$$

Since the indices q_w run from 3 to 5, only the rows from the 3rd to the 5th of each matrix $\bar{\mathtt{A}}, \dots, \bar{\mathtt{F}}$ contribute to the matrix $\mathtt{S}$. Looking at the structure of those matrices, we can note that the tensor $\mathtt{T}$ will contain only elements from the rotation matrices $\mathtt{R}_i$ and elements from the last row of matrices $-[\mathbf{t}_i]_\times \mathtt{R}_i$.

5.2 Axial Cameras

In axial cameras, all the projection rays cross a line, the camera axis, but otherwise they can be mutually skew. In such cameras, if the considered 3D line lies in an axial plane, *i.e.* a plane containing the axis, we should consider at most two rays for each view: any other ray in the same view, indeed, would be a linear combination of the previous ones, hence no useful constraint can be exploited from the matrix $\mathtt{M}$, as it would be always rank deficient.

Similarly, if the space line lies in a horizontal plane, *i.e.* a plane perpendicular to the axis, its projection rays are coplanar and, again, only two rays have to be considered from each view. On the other hand, the space line and the axis might be contained in a ruled quadric, hence at most three projection rays should be considered from each view. This is not the case for most axial-symmetric catadioptric cameras, as it has been proved in [6].

Another degenerate configuration occurs when the cameras are coaxial, *e.g.* they differ in terms of a rototranslation along the axis, then there are always two lines (the 3D line and the two coincident axes) that cross the projection rays.

We distinguish axial cameras that have a finite or an infinite axis.

5.2.1 Finite axis

Assume that the camera axis is the Z-axis. Then, all projection rays have Plücker coordinates with $L_6 = 0$. The base size of the multi-focal tensor $\mathtt{T}$ can be reduced from 6 to 5, and the expression (7) becomes

$$\det \mathtt{M} = \sum_{q_1=1}^{5} \dots \sum_{q_6=1}^{5} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6}\, T_{q_1,\dots,q_6} = 0.$$

so that only the last row of each matrix $\bar{\mathtt{A}}, \dots, \bar{\mathtt{F}}$ does not contribute to $\mathtt{S}$.

5.2.2 Infinite axis

We choose a local coordinate system where the axis is the line at infinity with coordinates $\begin{bmatrix}1 & 0 & 0\end{bmatrix}^\top$ (line coordinates on plane at infinity), so that the camera axis' Plücker coordinates are $\alpha = \begin{bmatrix}0 & 0 & 0 & 1 & 0 & 0\end{bmatrix}^\top$ and all the projection rays have the first Plücker coordinate equals to zero, *i.e.* the rays are of the form

$$\mathbf{a} = \begin{bmatrix}0 & a_2 & a_3 & a_4 & a_5 & a_6\end{bmatrix}^\top.$$

Multi-view relations for infinite axial cameras can thus be formulated via tensors of base size 5, and the expression (7) becomes

$$\det \mathtt{M} = \sum_{q_1=2}^{6} \dots \sum_{q_6=2}^{6} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6}\, T_{q_1,\dots,q_6} = 0.$$

so that only the first row of each matrix $\bar{\mathtt{A}}, \dots, \bar{\mathtt{F}}$ does not contribute to $\mathtt{S}$.

5.3 X-Slit Cameras

In x-slit cameras there exist two lines, *i.e.* camera axes, that cut all projection rays. Linear pushbroom cameras [15] is a special case of such cameras. The case of the two axes cutting one another, *i.e.* being coplanar, is not of interest here, so we consider two mutually skew axes. As discussed in the case of axial cameras, if the space line and one of the axes are coplanar then all the projection rays associated to the space line lie on a plane: thus, just in this case, we should consider at most two rays for each view.

Similarly, for any 3D line that is skew w.r.t. the camera axes the projection rays lie on a ruled quadric, since three skew lines completely define a ruled quadric [19]. Therefore, we should consider configurations with at most three rays from each view.

Finally, similarly to axial cameras, if the cameras share a common axis, then there are always two lines crossing all the rays associated to the space line.

Two cases are possible for x-slit cameras: (*i*) both axes are finite lines or (*ii*) one of the two axes is a line at infinity. Since at least one axis is a finite line, we adopt the same reference system used for axial cameras. As for the second axis, we have to distinguish the two cases.

5.3.1 Two finite axes

Having fixed the first axis α, we still have the freedom to rotate about it and translate along it. Since the two axes are skew, we may thus obtain a local coordinate system, where the second axis goes through a point on the Y-axis, and is parallel to the XZ-plane. Hence, it will be defined by two points as follows:

$$Q_1^\mathsf{T} = \begin{bmatrix} 0 & Y & 0 & 1 \end{bmatrix} \qquad Q_2^\mathsf{T} = \begin{bmatrix} X & 0 & Z & 0 \end{bmatrix}.$$

The second axis' Plücker coordinates are thus given by

$$\beta^\mathsf{T} = \begin{bmatrix} X & 0 & Z & -YZ & 0 & YX \end{bmatrix}.$$

Since each projection rays **a** must cut the two axes, by imposing the side constraint with both axes, the rays must be of the form:

$$\mathbf{a}^\mathsf{T} = \begin{bmatrix} a_1 & a_2 & a_3 & (Wa_1 - Ya_3) & a_5 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ W & 0 & -Y & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_5 \end{bmatrix} = \mathtt{G} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_5 \end{bmatrix},$$

where $W = \frac{YZ}{X}$[1] and Y can be seen as the intrinsic parameters of the camera. Then, after some straightforward computations, (5) becomes

$$\begin{bmatrix} a_1 & a_2 & a_3 & a_5 \end{bmatrix} \tilde{\mathtt{A}}\,\mathbf{L} = 0,$$

where

$$\tilde{\mathtt{A}}_{4\times 6} = \mathtt{G}^\mathsf{T}\bar{\mathtt{A}}.$$

The base size of the multi-focal tensor $\mathtt{T}$ is then reduced from 6 to 4 so that the expression (7) becomes

$$\det \mathtt{M} = \sum_{q_1}^{\{1,2,3,5\}} \cdots \sum_{q_6}^{\{1,2,3,5\}} a^{q_1} b^{q_2} c^{q_3} d^{q_4} e^{q_5} f^{q_6}\, T_{q_1,\ldots,q_6} = 0.$$

[1] We divide by X since it can not be zero, otherwise the second axis would be parallel to the first one, and thus coplanar, which is excluded here.

where the indices q_w run on the discrete set $\{1,2,3,5\}$ and $T_{q_1,\ldots,q_6}$ can be computed as in (8) by replacing each matrix $\bar{\mathtt{A}},\ldots,\bar{\mathtt{F}}$ with the corresponding matrix $\tilde{\mathtt{A}},\ldots,\tilde{\mathtt{F}}$.

5.3.2 Finite + infinite axis

Having fixed the first axis, we still have the freedom to rotate about it and translate along it. Translation has no effect on the infinite second axis, but we may rotate about the first axis, such that the second one has coordinates $\begin{bmatrix}0 & \cos\theta & \sin\theta\end{bmatrix}^{\mathsf{T}}$ (homogeneous coordinates of a line at infinity). The second axis' Plücker coordinates are thus

$$\beta^{\mathsf{T}} = \begin{bmatrix}0 & 0 & 0 & 0 & \cos\theta & \sin\theta\end{bmatrix}.$$

Projection rays cut the two axes, so must be of the form

$$\mathbf{a}^{\mathsf{T}} = \begin{bmatrix}a_1 & Wa_3 & a_3 & a_4 & a_5 & 0\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & W & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}a_1\\ a_3\\ a_4\\ a_5\end{bmatrix} = \mathtt{H}\begin{bmatrix}a_1\\ a_3\\ a_4\\ a_5\end{bmatrix},$$

where $W = -\tan\theta$ is the intrinsic parameter of the camera. Straightforwardly, (5) becomes

$$\begin{bmatrix}a_1 & a_3 & a_4 & a_5\end{bmatrix}\tilde{\mathtt{A}}\,\mathbf{L} = 0,$$

where

$$\tilde{\mathtt{A}}_{4\times 6} = \mathtt{H}^{\mathsf{T}}\bar{\mathtt{A}}.$$

The base size of the multi-focal tensor $\mathtt{T}$ is then reduced from 6 to 4 so that the expression (7) becomes

$$\det\mathtt{M} = \sum_{q_1}^{\{1,3,4,5\}}\cdots\sum_{q_6}^{\{1,3,4,5\}} a^{q_1}b^{q_2}c^{q_3}d^{q_4}e^{q_5}f^{q_6}\,T_{q_1,\ldots,q_6} = 0.$$

where the indices q_w run on the discrete set $\{1,3,4,5\}$ and $T_{q_1,\ldots,q_6}$ can be computed as in (8) by replacing each matrix $\bar{\mathtt{A}},\ldots,\bar{\mathtt{F}}$ with the corresponding matrix $\tilde{\mathtt{A}},\ldots,\tilde{\mathtt{F}}$.

6 Experimental Results

In order to estimate the tensor elements and retrieve the camera motion (*i.e.* the rototranslation), we note that (7) can be written as

$$\det\mathtt{M} = \sum_{s=1}^{Z}\alpha_s T_s = 0,$$

where Z is the the number of unique terms of the tensor for the considered case, T_s is the s-th unique element, and α_s is a coefficient that collects together all the contributions $a^{q_1} b^{q_2} \dots f^{q_6}$ from coordinates of rays (known from calibration) associated to the same (unique) element T_s. Moreover, each T_s is a function of the $6(n-1)$ unknowns of the $n-1$ rototranslations relating the views). Since the explicit form of each T_s is known and can be easily computed once from (8), a minimization process on the 6 unknowns can be run in order to get the motion between cameras and the elements of the tensor $\mathtt{T}$ as well.

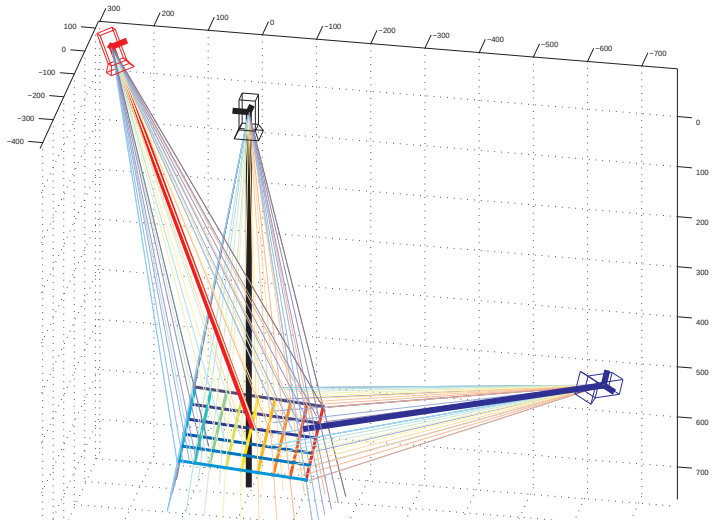
In order to test the effectiveness of the proposed estimation method, we performed some tests with images taken from a standard perspective camera (*cf*. Figure 3) and a central catadioptric one (*cf*. Figure 4). In both cases, we first calibrated the cameras using two image datasets of a calibration grid. The perspective camera has been calibrated with the Camera Calibration Toolbox for perspective cameras [4], while the catadioptric one has been calibrated with the "OcamCalib" Toolbox [25]. Then, for each camera, we chose three images from the dataset and we extracted the lines from the squares of the calibration grid. Since in the case of central camera at most two rays have to be considered, for each line we considered the two projection rays associated to the two extreme points of the line lying on the outer border of the grid. We estimated the tensors with the above method and from them we extracted the motion among the views finding in both cases the same rototranslations obtained by calibration.

The main issue we experienced was that the minimization procedure tends to find the transformations that just align the three cameras: thus all the projection rays cross a common line, the common baseline, satisfying (6). To avoid such degenerate solutions, we added some constraints to the minimization process so that it avoids to align the cameras unless then the rays also cross another common line in space. In general it requires some runnings (with different initial guesses) before converging to the optimal solution, although the initial guess does not have to be close to the real solution.

7 Conclusions

We presented a new formulation of multifocal tensors for general camera models working on projection rays crossing the same 3D line. We extended the theoretical framework proposed in [28] and based on matching rays that meet at a point in space. We also proposed a method to estimate the tensors and retrieve the motion parameters of the camera and performed some preliminary experiments in the special case of axial cameras.

(a) (b) (c)



(d)

Fig. 3 Three views of the calibration grid (a,b,c) taken with a standard perspective camera with the lines used to estimate the matching tensor and (d) the displacement among cameras computed from the tensor.

Acknowledgements

The authors would like to thank Davide Scaramuzza for providing the catadioptric images used in the experiments.

References

1. Baker, S., Nayar, S.K.: A theory of single-viewpoint catadioptric image formation. International Journal of Computer Vision **35**(2), 175–196 (1999). DOI 10.1023/A:1008128724364
2. Bakstein, H., Pajdla, T.: An overview of non-central cameras. In: Computer Vision Winter Workshop (2001)
3. Barreto, J., Araujo, H.: Paracatadioptric camera calibration using lines. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV '03), vol. 2, pp. 1359–1365. IEEE Computer Society, Los Alamitos, CA, USA (2003). DOI 10.1109/ICCV.2003.1238648

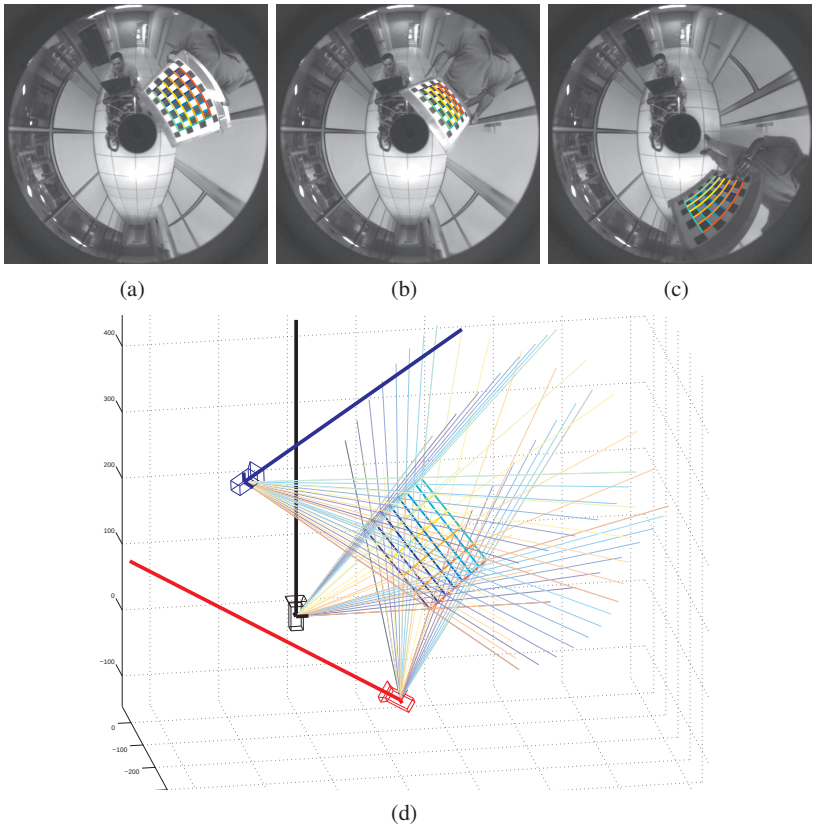


Fig. 4 Three views of the calibration grid (a,b,c) taken with a central catadioptric camera with the lines used to estimate the matching tensor and (d) the displacement among cameras computed from the tensor.

4. Bouguet, J.Y.: Camera calibration toolbox for matlab. http://www.vision.caltech.edu/bouguetj
5. Bronnimann, H., Everett, H., Lazard, S., Sottile, F., Whitesides, S.: Transversals to line segments in three-dimensional space. Discrete Compututational Geometry **34**(3), 381–390 (2005). DOI http://dx.doi.org/10.1007/s00454-005-1183-1
6. Caglioti, V., Gasparini, S.: On the localization of straight lines in 3D space from single 2D images. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '05), vol. 1, pp. 1129–1134. IEEE Computer Society, Los Alamitos, CA, USA (2005). DOI 10.1109/CVPR.2005.257
7. Caglioti, V., Gasparini, S.: "How many planar viewing surfaces are there in noncentral catadioptric cameras?" Towards singe-image localization of space lines. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '06), vol. 1, pp. 1266–1273. IEEE Computer Society, Los Alamitos, CA, USA (2006). DOI 10.1109/CVPR.2006.1
8. Faugeras, O., Mourrain, B.: On the geometry and algebra of the point and line correspondences between n images. In: CVPR95, pp. 951–956 (1995). DOI 10.1109/ICCV.1995.466832

9. Feldman, D., Pajdla, T., Weinshall, D.: On the epipolar geometry of the crossed-slits projection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV '03), vol. 2, pp. 988–995. IEEE Computer Society, Washington, DC, USA (2003)
10. Fermuller, C., Aloimonos, Y., Baker, P., Pless, R., Neumann, J., Stuart, B.: Multi-camera networks: eyes from eyes. In: Proceedings of the 1st Workshop on Omnidirectional Vision (OMNIVIS 2000), pp. 11–18 (2000). DOI 10.1109/OMNVIS.2000.853797
11. Firoozfam, P., Negahdaripour, S.: Multi-camera conical imaging; calibration and robust 3D motion estimation for ROV based mapping and positioning. In: Proceedings of OCEANS MTS/IEEE Conference and Exhibition, vol. 3, pp. 1595–1602 (2002). DOI 10.1109/OCEANS.2002.1191873
12. Geyer, C., Daniilidis, K.: Paracatadioptric camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**(5), 687–695 (2002). DOI 10.1109/34.1000241
13. Geyer, C., Daniilidis, K.: Mirrors in motion: epipolar geometry and motion estimation. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV '03), pp. 766–773. IEEE Computer Society, Los Alamitos, CA, USA (2003)
14. Grossberg, M., Nayar, S.: A general imaging model and a method for finding its parameters. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV '01), pp. 108–115. IEEE Computer Society, Los Alamitos, CA, USA (2001). URL `citeseer.ist.psu.edu/grossberg01general.html`
15. Gupta, R., Hartley, R.I.: Linear pushbroom cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence **19**(9), 963–975 (1997)
16. Hartley, R.I.: Lines and points in three views and the trifocal tensor. IJCV **22**(2), 125–140 (1997). DOI http://dx.doi.org/10.1023/A:1007936012022
17. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, second edn. Cambridge University Press (2004)
18. Hicks, R., Bajcsy, R.: Catadioptric sensors that approximate wide-angle perspective projections. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '00), vol. 1, pp. 545–551. IEEE Computer Society, Los Alamitos, CA, USA (2000). DOI 10.1109/CVPR.2000.855867
19. Hilbert, D., Cohn-Vossen, S.: Geometry and the Imagination. Chelsea Publishing Co., New York: Chelsea (1932)
20. Ishiguro, H., Yamamoto, M., Tsuji, S.: Omni-directional stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(2), 257–262 (1992). DOI 10.1109/34.121792
21. Lanman, D., Wachs, M., Taubin, G., Cukierman, F.: Reconstructing a 3d line from a single catadioptric image. In: Proceedings of the International Symposium on 3D Data Processing, Visualization and Transmission, pp. 89–96 (2006). DOI 10.1109/3DPVT.2006.115
22. Micusik, B., Pajdla, T.: Autocalibration & 3D reconstruction with non-central catadioptric cameras. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '04), vol. 1, pp. 58–65. IEEE Computer Society, Los Alamitos, CA, USA (2004). DOI 10.1109/CVPR.2004.1315014
23. Peleg, S., Ben-Ezra, M.: Stereo panorama with a single camera. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '99), vol. 1. IEEE Computer Society, Los Alamitos, CA, USA (1999). DOI 10.1109/CVPR.1999.786969
24. Pless, R.: Using many cameras as one. In: Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, vol. 2, pp. II–587–93vol.2 (2003). DOI 10.1109/CVPR.2003.1211520
25. Scaramuzza, D., Martinelli, A., Siegwart, R.: A toolbox for easily calibrating omnidirectional cameras. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5695–5701 (2006). DOI 10.1109/IROS.2006.282372
26. Shum, H.Y., Kalai, A., Seitz, S.: Omnivergent stereo. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV '99), vol. 1, pp. 22–29. IEEE Computer Society, Los Alamitos, CA, USA (1999). DOI 10.1109/ICCV.1999.791193
27. Sturm, P.: Mixing catadioptric and perspective cameras. In: Proceedings of the 3rd Workshop on Omnidirectional Vision (OMNIVIS 2002), pp. 37–44. IEEE Computer Society, Los Alamitos, CA, USA (2002). DOI 10.1109/OMNVIS.2002.1044489

28. Sturm, P.: Multi-view geometry for general camera models. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR '05), vol. 1, pp. 206–212. IEEE Computer Society, Los Alamitos, CA, USA (2005). DOI 10.1109/CVPR.2005.237
29. Svoboda, T., Pajdla, T.: Epipolar geometry for central catadioptric cameras. International Journal of Computer Vision **49**(1), 23–37 (2002). DOI 10.1023/A:1019869530073
30. Swaminathan, R., Nayar, S.: Nonmetric calibration of wide-angle lenses and polycameras. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(10), 1172–1178 (2000)
31. Teller, S., Hohmeyer, M.: Determining the lines through four lines. Journal of graphics tools **4**(3), 11–22 (1999)
32. Thirthala, S., Pollefeys, M.: Multi-view geometry of 1d radial cameras and its application to omnidirectional camera calibration. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV '05), vol. 2, pp. 1539–1546 (2005). DOI 10.1109/ICCV.2005.158
33. Weng, J., Huang, T., Ahuja, N.: Motion and structure from line correspondences: Closed-form solution, uniqueness, and optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(3), 318–336 (1992)
34. Zomet, A., Feldman, D., Peleg, S., Weinshall, D.: Mosaicing new views: The crossed-slits projection. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(6), 741–754 (2003)