International Journal of Computer Vision (2021) 129:3006–3026
https://doi.org/10.1007/s11263-021-01516-1

3D Scene Reconstruction with an Un-calibrated Light Field Camera

Qi Zhang[1] · Hongdong Li[2] · Xue Wang[1] · Qing Wang[1]

Received: 7 October 2020 / Accepted: 9 August 2021 / Published online: 27 August 2021

Abstract

This paper is concerned with the problem of multi-view 3D reconstruction with an un-calibrated micro-lens array based light field camera. To acquire 3D Euclidean reconstruction, existing approaches commonly apply the calibration with a checkerboard and motion estimation from static scenes in two steps. Self-calibration is the process of simultaneously estimating intrinsic and extrinsic parameters directly from un-calibrated light fields without the help of a checkerboard. While the self-calibration technique for conventional (pinhole) camera is well understood, how to extend it to light field camera remains a challenging task. This is primarily due to the ultra-small baseline of the light field camera. We propose an effective self-calibration method for a light field camera for automatic metric reconstruction without a laborious pre-calibration process. In contrast to conventional self-calibration, we show how such a self-calibration method can be made numerically stable, by exploiting the regularity and measurement redundancies unique for the light field camera. The proposed method is built upon the derivation of a novel ray-space homography constraint (RSHC) using Plücker parameterization as well as a ray-space infinity homography (RSIH). We also propose a new concept of "rays of the absolute conic (RAC)" defined as a special quadric in 5D projective space $\mathbb{P}^5$. A set of new equations are established and solved for self-calibration and 3D metric reconstruction specifically designed for a light field camera . We validate the efficacy of the proposed method on both synthetic and real light fields, and have obtained superior results in both accuracy and robustness.

Keywords Light field · Self-calibration · 3D reconstruction · Rays of the absolute conic (RAC)

1 Introduction

Light fields are commonly represented as spatial and angular discrete sampling of rays. Since the first hand-held light field camera (LFC) (Ng et al. 2005) is put forwarded in the last decade, significant efforts have been spent to develop compact and handy LFCs. Due to the regular and abundant angular information of light fields, commercial micro-lens array based LFCs such as Lytro (2011) and Raytrix (2013) gain increasing popularity and have been applied to many computer vision tasks, e.g. structure-from-motion (SfM) (Johannsen et al. 2015; Zhang et al. 2017b; Nousias et al. 2019), 3D reconstruction (Johannsen et al. 2016; Zhang et al. 2017a; Vianello et al. 2018), light field stitching (Birklbauer and Bimber 2014; Guo et al. 2016; Ren et al. 2017) and robust SLAM (Dansereau et al. 2011; Dong et al. 2013; Li et al. 2019). In order to perform these 3D metric related tasks, having an LFC metrically calibrated is essential.

Traditionally, existing multi-view LFC based applications conduct LFC calibration as a pre-processing separately and achieve their method with calibrated light fields. However, most LFC calibration methods are often conducted in an ad hoc way, depending on a direct adaption of the calibration method designed for a single pinhole camera. Even if it only requires a single printed checkerboard, the pre-calibration process is still time-consuming and laborious, consider-

Communicated by Adrien Bartoli.

The work was supported by NSFC under Grant 61531014, 61801396, 62031023.

✉ Xue Wang
xwang@nwpu.edu.cn

✉ Qing Wang
qwang@nwpu.edu.cn

Qi Zhang
nwpuqzhang@gmail.com

Hongdong Li
hongdong.li@anu.edu.au

[1] School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China

[2] ANU and ACRV, Australian National University, Canberra, ACT 2600, Australia

ing need of the additional captured light fields. It is much desirable to have a "self-calibration" (aka auto-calibration, or on-the-fly) method that can automatically determine the parameters of an LFC without the aid of any specific calibration target, but simply from observing the static scene. Once this is done, it is possible to compute a 3D metric reconstruction from multiple un-calibrated light fields directly.

Although an underlying LFC can be equivalent to an array of pinhole cameras, for which one could use the traditional self-calibration technique developed for one single pinhole camera. Taking a large number of sub-aperture images extracted from the light field into consideration, self-calibrating the sub-aperture images independently is still an onerous task. Moreover, treating each sub-aperture as a tiny pinhole camera overlooks the regularity among sub-apertures resulting in unstable estimation (Zhang et al. 2019c). Due to regularly arranged view points of an LFC on a plane, it is necessary to process the light field as a whole. It is our view that the abundant and regular rays captured by the LFC provide rich and redundant constraints that one needs to utilize for better numerical stability of any LFC based algorithms.

Besides, the constant baseline during self-calibration is another advantage of LFCs compared with traditional pinhole cameras. Since the traditional self-calibration methods cannot recover the scale directly, the translation between the first pair of cameras has a unit vector. However, the light field is equivalent to numerous sub-aperture images regularly arranged on a plane, and the baseline indicates the translation between neighboring sub-aperture images (Bok et al. 2017; Zhang et al. 2019c). Consequently, the constant intrinsic parameters, especially the baseline, make it easy to constrain the translation up to a uniform scale for 3D metric reconstruction (aka similar reconstruction) directly. It is also worth noting that since the Euclidean distance of static scenes is not provided in advance, the LFC self-calibration method recovers metric structure instead of isometric structure.

To the best of our knowledge, there is no dedicated LFC self-calibration algorithm available in the literature. This work is proposed to fill in this gap by providing the first self-calibration algorithm specifically designed for an LFC. The overview of the proposed algorithm is illustrated in Fig. 1. Specifically, by exploiting ray-space infinity homography and its conjugate rotation, a novel "rays of the absolute conic" (RAC) quadric equation is derived for an LFC. Solving these RAC equations gives rise to accurate and robust LFC self-calibration, and at the same time, 3D metric reconstruction can be computed via ray–ray correspondences. The ray–ray correspondence is two rays emitted from the same 3D point but sampled by two light fields. Also note that similar to some traditional self-calibration algorithms, the effect of radial distortion is neglected to simplify the proposed self-calibration method. Moreover, a common belief in these traditional self-calibration algorithms is numerical instability because of the insufficient features, and it still requires further research. Contrary to this, we show in this paper, by extensive experiments, the proposed LFC self-calibration method is numerically stable, and the estimated results (both camera parameters and reconstructed 3D structures) are accurate and robust to noise and outliers.

In summary, the main contributions are:

1. we develop a ray-space homography constraint based on ray–ray correspondences with Plücker parameterization.
2. we explore the ray-space infinity homography and present a new concept of "rays of the absolute conic" for LFC self-calibration.
3. we design a self-calibration algorithm of which the effectiveness is verified by 3D metric reconstruction with an un-calibrated LFC.

2 Related Work

2.1 Light Field Camera Motion Estimation

Researchers focus on recovering accurate LFC positions since the hand-held LFC is first introduced by Ng (2005). Light field essentially records the rays in space. Plücker coordinate also provides a reliable mathematical mechanism to uniformly parameterize rays and describe ray-to-ray transformation. For this reason, motion estimation for generalized cameras is a natural application for light field imaging. Pless (2003) utilizes rays to represent image pixels with Plücker coordinates. The generalized epipolar constraint between ray–ray correspondences is first proposed for motion estimation for generalized cameras. A linear framework requiring 17 ray–ray correspondences is proposed and solved via the Singular Value Decomposition (SVD). Based on Plücker coordinates, Bartoli and Sturm (2001, 2004) present the 3D line motion matrix for projective transformation and then specialize it to the affine, similar and Euclidean transformations. Different algebraic distances are defined to estimate motion from 3D line-line correspondences. Bartoli and Sturm (2005) also derive Plücker constraints for traditional camera motion estimation and 3D reconstruction from 3D line-line correspondences, including initialization, triangulation and bundle adjustment. Sturm (2005) further unifies the theory of multi-view geometry for generalized cameras. This can also be applied to LFCs by considering an LFC as an array of pinhole cameras. Li et al. (2008) analyze the degeneracy of the generalized epipolar constraint on three typical generalized cameras, including non-overlapping, axial and non-overlapping-axial multi-cameras. For these types of cameras, a linear method, where the essential matrix is first recovered and the rotation matrix is then decomposed, is proposed to avoid the ambiguity. It is worth noting that an LFC

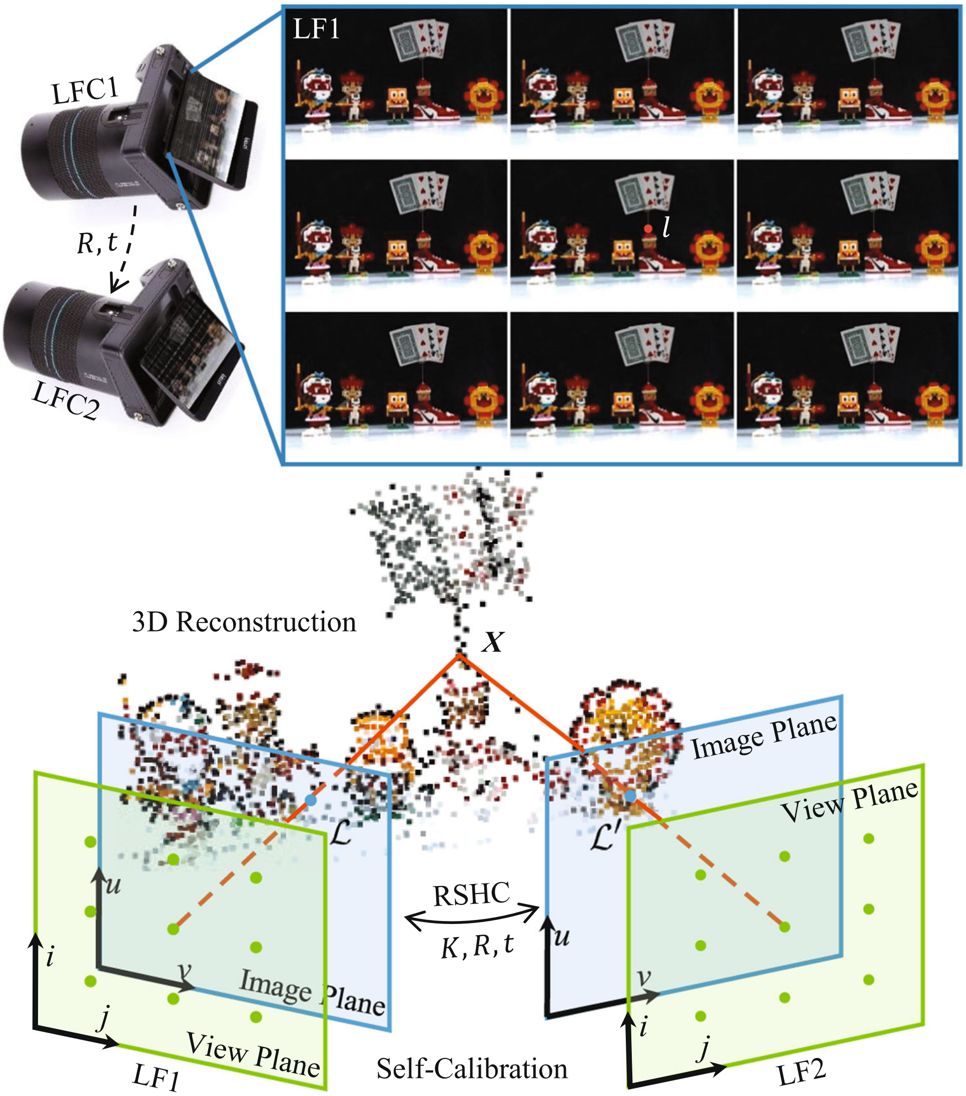


Fig. 1 An overview of LFC self-calibration and 3D metric reconstruction. We decode sub-aperture images and extract ray–ray correspondences from two LFs with Plücker parameterization. Ray–ray correspondence is two rays from the same 3D point but sampled by two light fields. Given these ray–ray correspondences, we generate the ray-space homography constraint (RSHC). The LFC parameters including LFC intrinsic matrix $\boldsymbol{K}$ and relative pose $\boldsymbol{R}$, $\boldsymbol{t}$ are then solved. We also implement 3D metric reconstruction with an un-calibrated LFC

does not cause these degeneracies due to the overlapping and planar views. Moreover, Kneip et al. (2014) combine the generalized epipolar constraint with eigenvalue minimization and develop an iterative solution to estimate motion of multi-cameras from at least 7 correspondences. This algorithm is numerical instability and sensitive to initialization although less correspondences are utilized.

Johannsen et al. (2015) first introduce Plücker parameterization to represent rays captured by an LFC. They derive a linear ray-point constraint extended from the method of Li et al. (2008) to estimate LFC's pose. This constraint defines the relationship between the 3D point and the rays intersected at that point, but it is difficult to recover the 3D point accurately with rays in a single light field, taken the ultra-small baseline of an LFC into consideration. Instead of ray-point constraint associated with 3D points, the ray constraints using line and plane are developed respectively by Zhang et al. (2017b). They explore the transformations of ray-line and ray-plane with the changing pose. Similarly, either ray-line constraint or ray-plane constraint is sensitive to small noises, due to ultra-small baseline of the LFC. More recently, Nousias et al. (2019) summarize a complete structure-from-motion pipeline for a calibrated LFC. They also demonstrate that ray–ray constraint of Li et al. (2008) is stable compared with ray-point constraint of Johannsen et al. (2015) for the initialization of LFC motion estimation.

In the following, the term *metric structure* implies that the structure is defined up to a similarity according to Hartley and Zisserman (2003). The metric structure is an isometric structure composed with an isotropic scaling. In summary, the above methods enable 3D metric reconstruction to recover from pre-calibrated LFCs. The reconstruction that is directly recovered from multiple light fields without a prior calibration may result in a projective reconstruction, as also demonstrated in Zhang et al. (2019c), so having a metrically calibrated LFC is important.

2.2 Light Field Camera Calibration

Many research groups have extensively explored various LFC calibration algorithms to further perform multi-view light field based tasks. The first LFC calibration algorithm is proposed by Dansereau et al. (2013), which derives a 4D decoding matrix containing 12 free parameters. This matrix transforms recorded pixels into rays outside the LFC. However, they utilize a traditional calibration for initialization, which is still a time-consuming process for LFC, as verified in Zhang et al. (2019c). Besides, their intrinsic parameters are redundant and dependent, which leads to irregular rays for post-processing (e.g. SfM and 3D reconstruction, as also demonstrated in Birklbauer and Bimber 2014). Contrary to this, Bok et al. (2014; 2017) propose a geometric projection with 6 intrinsic parameters that are complex but have clear physical meaning. They generate line features extracted from sub-images of raw data for calibration. However, the low-resolution of sub-image brings challenges for accurate line feature extraction, so does the unfocused capturing status. It also demonstrates that raw data with low-resolution sub-images (e.g. 14 × 14 for Lytro Illum) can not perform LFC based algorithms stably. Consequently, for most LFC based applications, the first step is usually seeking another light field representation, such as sub-aperture images.

Zhang et al. (2019c) also present a 6-parameter multi-projection-center (MPC) model. It is applicable to both traditional LFC and focused LFC designs. A 3D linear point-ray constraint is defined as the relationship between geometric structure and sampling rays, which also shows the importance of intrinsic parameters for 3D reconstruction. An efficient LFC calibration pipeline is developed for generic LFC. The projections of an LFC onto planes and conics are also explored under the MPC model by Zhang and Wang (2018) and Zhang et al. (2019b) respectively. Given that a light field essentially represents the collection of rays as a whole, a ray-space projection model is extended from the MPC model with the introduction of Plücker parameterization by Zhang et al. (2019a). They also develop a simple 6×6 intrinsic matrix which encapsulates all the six intrinsic parameters. A linear constraint and a ray–ray cost function are established for linear initial solution and non-linear optimization respectively. Based on ray-space projection model, Zhang et al. (2020) propose the ray-space epipolar geometry to intrinsically describe the relation between two light fields.

Existing LFC calibration algorithms are conducted with the help of special calibration targets. Even only a single printed checkerboard is needed, the prior calibration is still time-consuming, let alone to capture the additional light fields for calibration before applying multi-view light field tasks. A self-calibration algorithm specifically designed for an LFC is important to reduce the workload of laborious calibration.

2.3 Self-Calibration

Traditional self-calibration has gained increasing attentions since the seminal work presented by Maybank *et al.* (1992). Existing approaches can be roughly divided into three categories: (1) direct method for estimating dual image of the absolute conic (DIAC) based on the Kruppa's equation (Luong and Faugeras 1997; Seo et al. 2001; Paudel and Van Gool 2018); (2) stratified method for estimating the plane at infinity based on Modulus constraint followed by solving DIAC (Hartley et al. 1999; Pollefeys and Van Gool 1999; Nistér 2004; Chandraker et al. 2007b; Gherardi and Fusiello 2010); (3) joint estimation of both the plane at infinity and DIAC in the form of dual image of the absolute quadric (DIAQ) (Triggs 1997; Chandraker et al. 2007a; Habed et al. 2014). In order to increase numerical stability and find the minimal case for self-calibration, algebraic polynomial methods are involved (Paudel and Van Gool 2018; Larsson et al. 2018). Gurdjos et al. (2009) try to add spectral constraints in self-calibration algorithm to avoid ambiguous motion sequences and increase the numerical stability.

A seemingly straightforward choice for self-calibrating an LFC is to consider it as a direct extension of that for a single pinhole camera, given that an LFC is equivalent to a collection of pinhole cameras. Dealing with sub-aperture images independently, however, incurs burdensome processes. Moreover, an LFC is also a special design whose principle points are regularly arranged on a plane and this arrangement remains constant among light fields, which provides geometric constraint for feature extraction and self-calibration. For efficient use of abundant and regular rays captured by an LFC, it is necessary to uniformly process a light field as a whole. Inspired by the traditional self-calibration, a ray-space homography is established to decompose ray-space infinity homography and specifically derive rays of the absolute conic for an LFC.

3 Ray-Space Homography

A traditional 2D camera is a mapping between the 3D scene point and a 2D image point, as described by Hartley and Zisserman (2003), while compact LFCs are innovated from the traditional 2D camera with a similar but different way to record 3D scene. An LFC attaches to a micro-lens array in front of the sensor enables 3D reconstruction of the scene from a single photographic exposure, because it collects rays emanating from the scene point at different directions. To simplify the discussion of geometric analysis in the following section, a pixel captured by an LFC is generalized to a 4D ray from a 3D point (Ng 2006) in contrast with a 2D image point for a traditional camera. With the angular sampling of the light field, the ray captured by an LFC is usually represented

Table 1 Definitions of notations used in the paper

Notation	Definition
$\boldsymbol{l} = (i, j, u, v)^\top$	Indexed pixel of the light field in light field coordinate frame
$\boldsymbol{r} = (s, t, x, u)^\top$	Generalized ray in camera coordinate frame
$\mathcal{L} = (\boldsymbol{n}^\top, \boldsymbol{p}^\top)^\top$	Plücker coordinates of the ray $\boldsymbol{l}$ captured by an LFC, where $\boldsymbol{n}$ and $\boldsymbol{p}$ are moment and direction vectors
$\mathcal{R} = (\boldsymbol{m}^\top, \boldsymbol{q}^\top)^\top$	Plücker coordinates of the ray $\boldsymbol{r}$ in the space, where $\boldsymbol{m}$ and $\boldsymbol{q}$ are moment and direction vectors
$\boldsymbol{K} = diag(\boldsymbol{K}_{ij}, \boldsymbol{K}_{uv})$	6 × 6 Ray-space intrinsic matrix for an LFC, which is decomposed as block matrices $\boldsymbol{K}_{ij}$ and $\boldsymbol{K}_{uv}$
$\boldsymbol{R} = (\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3)$	Rotation matrix
$\boldsymbol{t}$	Translation vector
$\boldsymbol{P}$	6 × 6 ray-space projection matrix
$\boldsymbol{H}$	6 × 6 ray-space homography matrix
$\boldsymbol{H}_\infty$	6 × 6 ray-space infinity homography matrix
$\boldsymbol{\omega} = \boldsymbol{K}^\top \boldsymbol{K}$	Rays of the absolute conic (RAC)

by $\boldsymbol{l} = (i, j, u, v)^\top \in \mathbb{R}^4$ in a two-parallel-plane parameterization (Levoy and Hanrahan 1996), where $(u, v)^\top$ refers to *relative* image points of a pinhole camera with the projection center at $(i, j, 0)^\top$, as shown in Fig. 3. Each independent view point $(i, j)^\top$ corresponds to a sub-aperture image. Subsequently, with the help of shifted view points, an LFC can also be considered as an array of pinhole cameras regularly arranged on the view plane. Inversely, the traditional camera is the specialization of the LFC to the case of only one projection center $(0, 0, 0)^\top$ on the view plane (Zhang et al. 2019c).

In this section, a light field is described as a collection of rays, which is represented via Plücker parameterization. A ray–ray correspondence indicates two rays from the same 3D point but sampled by two light fields. Based on the ray-space projection matrix, ray-space homography is derived to describe the transformation of Plücker ray between different light field coordinate frames. Ray-space infinity homography is then decomposed from ray-space homography. Table 1 gives the definitions of notations used in the following sections.

3.1 Plücker Parameterization of Ray

A rigid body in 3D projective space is well-known to have six degrees of freedom (three for rotation and three for translation), while a ray (line) has only four degrees of freedom (Hartley and Zisserman 2003). It is hard to linearly formulate transformations of rays, such as rotation and translation.

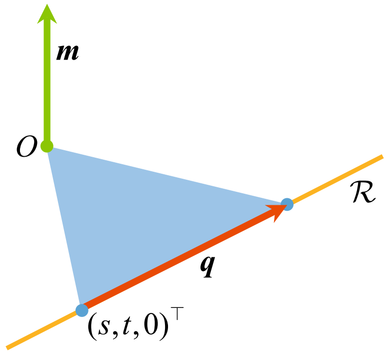


Fig. 2 Plücker parameterization of the ray. A ray can be represented by its direction $\boldsymbol{q}$ and a point $(s, t, 0)^\top$ on the ray. The Plücker coordinates of the ray is defined as $\mathcal{R} = (\boldsymbol{m}^\top, \boldsymbol{q}^\top)^\top$, where $\boldsymbol{m} = (s, t, 0)^\top \times \boldsymbol{q}$ indicates the moment vector and be perpendicular to $\boldsymbol{q}$, i.e. $\boldsymbol{m}^\top \boldsymbol{q} = 0$

Considering that a light field typically represents a collection of rays in 3D projective space, we need a new mechanism to parameterize arbitrary rays. Consequently, the approach we will take is to represent a ray in free space by Plücker parameterization. It provides mathematically elegant and linear equations for transformations of rays. In addition, Plücker coordinates are a homogeneous parameterization to unambiguously represent a ray in 3D projective geometry. We will briefly introduce Plücker parameterization of rays as follows.

Suppose a ray $\boldsymbol{r}$ in 3D projective space denotes $(s, t, x, y)^\top$ in a two-parallel-plane parameterization, where $(s, t)^\top$ and $(x, y)^\top$ indicate the view point (positional information of $\boldsymbol{r}$) and relative image point (directional information of $\boldsymbol{r}$) respectively. A ray $\boldsymbol{r}$ can be represented by its direction $\boldsymbol{q} = (x, y, 1)^\top$ and a point $(s, t, 0)^\top$ that it passes through. In Plücker parameterization, the ray is defined as a pair of vectors, namely a moment vector $\boldsymbol{m} \in \mathbb{R}^3 \backslash \{\boldsymbol{0}\}$ and a direction vector $\boldsymbol{q} \in \mathbb{R}^3$, as shown in Fig. 2. Note that, the moment vector is the cross product between the direction of the ray and arbitrary point on the ray. In other words, the moment vector $\boldsymbol{m}$ is perpendicular to the plane containing the ray and the origin, that is $\boldsymbol{m}^\top \boldsymbol{q} = 0$, as shown in Fig. 2. In summary, the moment vector and the direction vector of arbitrary ray are defined as (Pottmann and Wallner 2009),

$$\begin{cases} \boldsymbol{m} = (s, t, 0)^\top \times (x, y, 1)^\top = (t, -s, sy - tx)^\top \\ \boldsymbol{q} = (x, y, 1)^\top \end{cases}, \tag{1}$$

where $\mathcal{R} = (\boldsymbol{m}^\top, \boldsymbol{q}^\top)^\top$ is Plücker coordinate. The Plücker coordinates are homogeneous coordinates. In upcoming equations, the calligraphic symbol, such as $\mathcal{R}$ for the ray $\boldsymbol{r}$, indicates the Plücker coordinates obtained by stacking the moment vector on top of the direction vector.

For the sake of discussions in the following section, we list the relevant equations. The point represented by a homogeneous vector $(\boldsymbol{X}^\top, X_4)^\top$ lies on the ray $\mathcal{R}$ iff

$$-X_4\boldsymbol{m} + \boldsymbol{X} \times \boldsymbol{q} = \boldsymbol{0}, \tag{2}$$

The matrix form is defined as,

$$\left(-X_4\boldsymbol{I}\ [\boldsymbol{X}]_\times\right)\mathcal{R} = \boldsymbol{0}, \tag{3}$$

where $[\cdot]_\times$ denotes the vector cross product (Hartley and Zisserman 2003).

Two rays $\mathcal{R}_1$ and $\mathcal{R}_2$ given in the same coordinate frame intersect iff

$$\boldsymbol{q_2}^\top\boldsymbol{m}_1 + \boldsymbol{m_2}^\top\boldsymbol{q}_1 = 0. \tag{4}$$

3.2 Ray–Ray Correspondences

For traditional pinhole cameras, a 3D scene point is projected to a pixel on the 2D image plane. The relations between two images taken from different view points is established by finding pixel-pixel correspondence relating to the same 3D scene point. In contrast, a light field is represented as a collection of rays. The ray with Plücker coordinates is a basic algebraic entity of a light field. With the help of shifted views in a light field, multiple rays emanating from the same 3D point are sampled. Consequently, the ray–ray correspondences between two light fields are defined as two sets of rays with Plücker coordinates,

$$\{\mathcal{L}_i\}_{i=1,\ldots,n} \longleftrightarrow \left\{\mathcal{L}'_j\right\}_{j=1,\ldots,m}, \tag{5}$$

where $\mathcal{L}_i$ and $\mathcal{L}'_j$ are from the same 3D point but are recorded in different light fields. $\mathcal{L} = (\boldsymbol{n}^\top, \boldsymbol{p}^\top)^\top$ is the Plücker coordinates of the ray $\boldsymbol{l} = (i, j, u, v)^\top$ captured by an LFC. As mentioned in Sect. 3.1, $\boldsymbol{n} = (i, j, 0)^\top \times (u, v, 1)^\top = (j, -i, iv - ju)^\top$ and $\boldsymbol{p} = (u, v, 1)^\top$ indicate the moment and direction vectors of the Plücker ray $\mathcal{L}$.

3.3 Ray-Space Projection Matrix

According to Zhang et al. (2019a), with the introduction of Plücker parameterization, the ray-space projection (RSP) matrix $\boldsymbol{P} \in \mathbb{R}^{6\times6}$ is proposed for an LFC to describe the ray-ray transformation from the ray $\mathcal{L} = (\boldsymbol{n}^\top, \boldsymbol{p}^\top)^\top$ captured by an LFC to the generalized ray $\mathcal{R} = (\boldsymbol{m}^\top, \boldsymbol{q}^\top)^\top$ in 3D space,

$$\mathcal{R} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{E} \\ \boldsymbol{O}_{3\times3} & \boldsymbol{R} \end{bmatrix} \boldsymbol{K}\mathcal{L}, \tag{6}$$

where $\boldsymbol{E} = [\boldsymbol{t}]_\times\boldsymbol{R}$ is the essential matrix. $\boldsymbol{R} \in SO(3)$ and $\boldsymbol{t} \in \mathbb{R}^3$ refer to the rotation and translation of an LFC respectively.

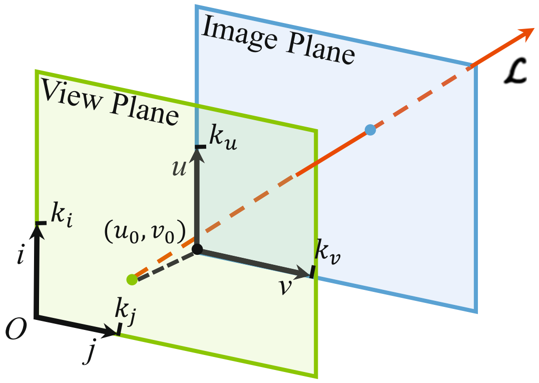


Fig. 3 LFC intrinsic parameters. A ray is represented as its intersections on the view plane (green dot) and image plane (blue dot). $(i, j)^\top$ on view plane indicates the view point. $(u, v)^\top$, which is relative to the intersection (black dot) of optical axis placed at $(i, j, 0)$, denotes the direction of the ray. (k_i, k_j) are the scale factors on view plane (i.e. baseline of an LFC), and (k_u, k_v) for the image plane (i.e. focal length of an LFC). $(-\frac{u_0}{k_u}, -\frac{v_0}{k_v})$ is the principle point of sub-aperture image, which implies the offset between view plane and image plane (Color figure online)

In addition, $\boldsymbol{K} \in \mathbb{R}^{6\times6}$ is defined as the ray-space intrinsic matrix with the following format,

$$\boldsymbol{K} = \begin{bmatrix} k_j & 0 & 0 & 0 & 0 & 0 \\ 0 & k_i & 0 & 0 & 0 & 0 \\ -k_j u_0 & -k_i v_0 & k_i k_v & 0 & 0 & 0 \\ 0 & 0 & 0 & k_u & 0 & u_0 \\ 0 & 0 & 0 & 0 & k_v & v_0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_{ij} & \boldsymbol{O}_{3\times3} \\ \boldsymbol{O}_{3\times3} & \boldsymbol{K}_{uv} \end{bmatrix}, \tag{7}$$

which contains all six LFC intrinsic parameters $(k_i, k_j, k_u, k_v, u_0, v_0)$, as shown in Fig. 3. (k_i, k_j) are the scale factors on view plane, namely baseline of an LFC. (k_u, k_v) are the scale factors on image plane. $(-\frac{u_0}{k_u}, -\frac{v_0}{k_v})$ can be considered as the principle point of a sub-aperture image. It also implies the offset between the two-parallel-plane. This matrix describes the ray sampling of an LFC between light field coordinate frame and camera coordinate frame.

Moreover, $\boldsymbol{K}$ can be decomposed as a lower triangular block matrix $\boldsymbol{K}_{ij} \in \mathbb{R}^{3\times3}$ and a upper triangular block matrix $\boldsymbol{K}_{uv} \in \mathbb{R}^{3\times3}$. An important property of $\boldsymbol{K}$ can be observed for the equation derivation in Sect. 4,

Property 1 Block intrinsic matrices $\boldsymbol{K}_{ij}$ and $\boldsymbol{K}_{uv}$ are orthogonal, i.e. $\boldsymbol{K}_{ij}\boldsymbol{K}_{uv}^\top = \boldsymbol{K}_{uv}\boldsymbol{K}_{ij}^\top = k_i k_v \boldsymbol{I}$.

3.4 Ray-Space Homography Constraint

Instead of treating an LFC as a collection of perspective cameras independently for self-calibration, we develop a unified framework that considers all rays captured by an LFC as a whole and propose *ray-space homography constraint*.

Corollary 1 *Consider rays $\mathcal{L}$ and $\mathcal{L}'$ sampled by two light fields intersecting at a single point, the ray-space homography constraint (RSHC) is,*

$$\boldsymbol{p}^\top \boldsymbol{K}_{ij}^{-1} \boldsymbol{R} \boldsymbol{K}_{ij} \boldsymbol{n}' + \boldsymbol{p}^\top \boldsymbol{K}_{ij}^{-1} \boldsymbol{E} \boldsymbol{K}_{uv} \boldsymbol{p}' + \boldsymbol{n}^\top \boldsymbol{K}_{uv}^{-1} \boldsymbol{R} \boldsymbol{K}_{uv} \boldsymbol{p}' = 0 \tag{8}$$

where $\mathcal{L} = (\boldsymbol{n}^\top, \boldsymbol{p}^\top)^\top$ and $\mathcal{L}' = (\boldsymbol{n}'^\top, \boldsymbol{p}'^\top)^\top$.

Proof Given RSP matrices for two light field, the light field coordinate frame of the first light field is assumed as the reference,

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix} \boldsymbol{K}, \qquad \boldsymbol{P}' = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{E} \\ \boldsymbol{O} & \boldsymbol{R} \end{bmatrix} \boldsymbol{K}. \tag{9}$$

We first back-project a ray $\mathcal{L}_2'$ in the second light field coordinate frame, and ray-trace to a ray $\mathcal{L}_2$ in the first light field coordinate frame according to Eq. (9), named as *ray-space homography* $\boldsymbol{H} = \boldsymbol{P}^{-1}\boldsymbol{P}'$,

$$\mathcal{L}_2 = \underbrace{\boldsymbol{K}^{-1} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{E} \\ \boldsymbol{O} & \boldsymbol{R} \end{bmatrix} \boldsymbol{K}}_{=:\boldsymbol{H}} \mathcal{L}_2', \tag{10}$$

which describes the ray–ray transformation in 3D projective space $\mathbb{P}^3$, as shown in Fig. 4. In order to provide connivence for the self-calibration algorithm to calculate relative poses, we partition Eq. (10) into 2 × 2 block matrices,

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{H}_{11} & \boldsymbol{H}_{12} \\ \boldsymbol{O}_{3\times3} & \boldsymbol{H}_{22} \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_{ij}^{-1} \boldsymbol{R} \boldsymbol{K}_{ij} & \boldsymbol{K}_{ij}^{-1} \boldsymbol{E} \boldsymbol{K}_{uv} \\ \boldsymbol{O}_{3\times3} & \boldsymbol{K}_{uv}^{-1} \boldsymbol{R} \boldsymbol{K}_{uv} \end{bmatrix}, \tag{11}$$

where $\boldsymbol{H}_{ij}$ is a 3 × 3 block matrix.

Then, let the re-traced ray $\mathcal{L}_2$ intersect $\mathcal{L}_1$ at 3D point $\boldsymbol{X}$. Substituting Eq. (10) into Eq. (4), we obtain RSHC and rewrite it in a block matrix form,

$$\begin{bmatrix} \boldsymbol{p}^\top & \boldsymbol{n}^\top \end{bmatrix} \boldsymbol{K}^{-1} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{E} \\ \boldsymbol{O} & \boldsymbol{R} \end{bmatrix} \boldsymbol{K} \begin{bmatrix} \boldsymbol{n}' \\ \boldsymbol{p}' \end{bmatrix} = 0, \tag{12}$$

which needs to be satisfied by every ray–ray correspondence $\mathcal{L} \leftrightarrow \mathcal{L}'$ intersecting at a 3D point. □

In geometry, the ray with Plücker coordinates satisfies self-constraint of Klein quadric in $\mathbb{P}^5$. It can also be considered as a point on Klein quadric in $\mathbb{P}^5$. $\boldsymbol{H}$ describes the ray–ray projective transformation between different coordinate frames in $\mathbb{P}^3$. Geometrically, $\boldsymbol{H}$ also represents the point-point hyper-transformation on Klein quadric in $\mathbb{P}^5$. Moreover, Bartoli and Sturm (2001, 2004) derive the 3D line motion matrix based on Plücker coordinates to estimate the relative motion of the calibrated camera. They also analyze the linear representation of 3D lines under different transformations (e.g. Euclidean, similar, affine and homography), which is similar to ray-space homography of rays. According to the definition proposed by Bartoli and Sturm (2001, 2004) and *Property* 1, ray-space intrinsic matrix could be considered as a specialization of the 3D line homography matrix. Note that Plücker transformation related to extrinsic parameters is Euclidean transformation, the geometric interpretation of ray-space homography is also defined as the 3D line homography matrix of the Plücker ray between different coordinate frames.

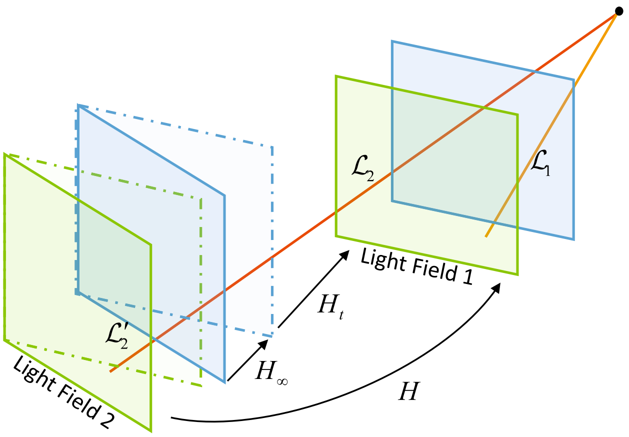


Fig. 4 Ray-space homography. Ray-space homography represents the transformation of a Plücker ray between different light field coordinate frames. The second light field (on the left) may be rotated and corrected to simulate a pure translation. The ray-space homography $\boldsymbol{H}$ can be divided into ray-space infinity homography $\boldsymbol{H}_\infty$ (first) and ray-space translation homography $\boldsymbol{H}_t$(second)

3.5 Ray-Space Infinity Homography

According to Eq. (10), the ray-space homography $\boldsymbol{H}$ only depends on the intrinsic parameters of an LFC and relative pose. Given two arbitrary light fields captured by an LFC, we may rotate the LFC used for the second light field so that it is aligned with the first light field. This rotation may be simulated by applying a homography to the second light field. Then, we utilize the homography of pure translation to transform the two light fields in a uniform coordinate frame, as shown in Fig. 4. Consequently, according to the pure rotation and pure translation, $\boldsymbol{H}$ can be divided into two parts,

$$\mathcal{L} = \underbrace{\boldsymbol{K}^{-1} \begin{bmatrix} \boldsymbol{I} & [\boldsymbol{t}]_\times \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix} \boldsymbol{K}}_{=:\boldsymbol{H}_t} \ \underbrace{\boldsymbol{K}^{-1} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{R} \end{bmatrix} \boldsymbol{K}}_{=:\boldsymbol{H}_\infty} \mathcal{L}', \tag{13}$$

where one can see the decomposition into translation homography $\boldsymbol{H}_t$ (partitioned first) and rotation homography $\boldsymbol{H}_\infty$ (partitioned second), as shown in Fig. 4. Note that $\boldsymbol{H}_t$ indicates the homography generated by pure translation and does not influence the direction of rays.

We then focus on the analysis of rotation homography $\boldsymbol{H}_\infty$. Consider $(\boldsymbol{X}^\top, 0)^\top$ is the homogeneous coordinates of point on the plane at infinity in the first light field coordinate frame, $\mathcal{L}'$ is the ray in the second light field coordinate frame which passes through this point. Substituting Eq. (13) into Eq. (3), we have,

$$\left[\,\boldsymbol{O}\;[\boldsymbol{X}]_\times\,\right]\boldsymbol{H}_t\boldsymbol{H}_\infty\mathcal{L}' = \left[\,\boldsymbol{O}\;[\boldsymbol{X}]_\times\,\right]\boldsymbol{H}_\infty\mathcal{L}' = \boldsymbol{0}, \tag{14}$$

where $\boldsymbol{H}_t$ does not influence the direction vectors of rays from the plane at infinity, therefore it could be eliminated.

Remarks we can conclude that, for any ray–ray correspondences intersecting at points on the plane at infinity, RSHC does not rely on the translation, only on the rotation and LFC intrinsic parameters. Alternatively, $\boldsymbol{H}_\infty$ is obtained if the translation $\boldsymbol{t}$ is $\boldsymbol{0}$, which corresponds to a rotation about the LFC. Thus $\boldsymbol{H}_\infty$ is ray-space homography that relates arbitrary rays if the LFC's motion is a pure rotation.

In summary, the plane at infinity is a particularly important plane for self-calibration and metric reconstruction. The ray-space infinity homography $\boldsymbol{H}_\infty$ represents the ray–ray transformation on the plane at infinity. In addition, according to the definition of $\boldsymbol{H}_\infty$, i.e. Eq. (14), it is interesting to note that $\boldsymbol{H}_\infty$ is a *conjugate rotation* which is important for self-calibration.

3.6 Rays of the Absolute Conic

The ray-space infinity homography gives additional insight into the LFC self-calibration. The absolute conic is on the plane at infinity such that a novel concept of "rays of the absolute conic" (RAC) is defined, just as what has been developed for the traditional camera, namely,

Definition 1 The rays of the absolute conic (RAC) is the quadric $\boldsymbol{\omega} = \boldsymbol{K}^\top\boldsymbol{K}$ in $\mathbb{P}^5$.

The RAC $\boldsymbol{\omega}$ depends only on the intrinsic parameters $\boldsymbol{K}$ of an LFC, and it does not depend on the LFC orientation or position. Since $\boldsymbol{\omega}$ is the rays of the absolute conic, it may be thought of as a convenient algebraic entity, and will be used in computations on LFC self-calibration.

According to the orthogonal rotation, we subsequently derive an important corollary of RAC $\boldsymbol{\omega}$,

Corollary 2 *The projection of RAC $\boldsymbol{\omega}$ under RSIH $\boldsymbol{H}_\infty$ is equal to the RAC,*

$$\boldsymbol{H}_\infty^\top\boldsymbol{\omega}\boldsymbol{H}_\infty = \boldsymbol{\omega}. \tag{15}$$

Remarks It is well known that the image of the absolute conic (IAC) is the key idea for the traditional self-calibration method of a pinhole camera, whereas for an LFC this role is played by the rays of the absolute conic (RAC). Eq. (15) could be utilized to calculate intrinsic parameters of an LFC once $\boldsymbol{H}_\infty$ is estimated. In addition, the presences of $\boldsymbol{H}_\infty$ may be expressed by saying that a RAC transforms invariantly. It is also worth noting that IAC for the traditional camera is a special case of RAC for the LFC when there is only one projection center on the view plane according to Eq. (15).

4 Self-Calibration Algorithm for LFC

For the traditional camera, infinity homography is the key to solve the self-calibration problem. It is estimated based on Modulus constraint which is a quartic polynomial or the special imaging condition (e.g. pure rotation). In contrast, considering that abundant and regular rays recorded by the LFC, we linearly establish RSHC and decompose RSIH for robust self-calibration. Also, RAC is retrieved from the RSIH. We finally design an effective self-calibration algorithm specifically for an LFC, including linear initialization and non-linear optimization.

4.1 Ray-Space Homography Estimation

Given a ray–ray correspondence $(\boldsymbol{n}^\top, \boldsymbol{q}^\top)^\top \leftrightarrow (\boldsymbol{n}'^\top, \boldsymbol{q}'^\top)^\top$, using Kronecker product operator $\otimes$, we re-state Eq. (8) as,

$$\left(\boldsymbol{p}^\top \otimes \boldsymbol{n}'^\top, \boldsymbol{p}^\top \otimes \boldsymbol{p}'^\top, \boldsymbol{n}^\top \otimes \boldsymbol{p}'^\top\right)\vec{\boldsymbol{H}} = 0, \tag{16}$$

where $\vec{\boldsymbol{H}}$ refers to a 27-vector made up of the non-zero elements of $\boldsymbol{H}$ in row-major order respectively. Considering N sets of $n \times m$ ray–ray correspondences ($N \times n \times m \geq 26$) in the form of Eq. (5), Eq. (16) is stacked as a homogeneous set of $(N \times n \times m) \times 27$ linear equations, i.e., $\boldsymbol{A}\vec{\boldsymbol{H}} = 0$. Ray-space homography estimation is numerical stability with sufficient ray–ray correspondences. Then $\vec{\boldsymbol{H}}$ can only be determined up to a scale factor via standard SVD (Hartley and Zisserman 2003). Once $\boldsymbol{H}$ is computed from considerable RSHCs, we can directly decompose RSIH $\boldsymbol{H}_\infty$ according to Eq. (13).

Degeneracy It is worth noting that previous work (Li et al. 2008) on motion estimation from pre-calibrated generalized cameras proposes some degenerate cases, i.e. non-overlap multi-cameras and axial cameras, which can not be solved by standard SVD directly. For this reason, they propose a numerical method where first the essential matrix is recovered, from which one obtains the rotation using a decomposition step. However, the special design of the LFC permits the scene to be captured by different view points during a single shoot. Sub-aperture images share overlapping field-of-view and regularly arrange on a plane. For this reason, RSHCs generated by the ray–ray correspondences can not arise these degeneracies.

In addition, RSHC also contains intrinsic matrix $\boldsymbol{K}$ compared with the generalized epipolar constraint, according to Eq. (8). It also decides the ambiguity of the solution will not happen in RSHCs. Consequently, a unique solution of Eq. (16) is readily solvable via the standard SVD method.

Normalization Considering the huge difference between angular resolution and spatial resolution of a light field, ray normalization specifically designed for an LFC is introduced to improve the numerical stability of the ray-space homography estimation. In a traditional camera, image normalization is utilized to improve accuracy and tackle less well-conditioned problems, like the linear estimation of the homography or the fundamental matrix. Similar to traditional normalization, two similarity transformations $\boldsymbol{T}$ and $\boldsymbol{T}'$ on Plücker coordinates, consisting of scale factors on view plane and image plane and a translation on image plane, are computed according to Eq. (7). Note that, the scale factors on view plane or image plane are identical. It is convenient to establish Eq. (7) which needs to satisfy the condition $k_u/k_v = k_i/k_j$. The ray normalization is partly summarized in Alg. 1 and described in detail as follows,

- *NormalizeRays*. Transform the rays $\mathcal{L}$ to a new sets of rays $\widetilde{\mathcal{L}}$, so do rays $\mathcal{L}'$, namely $\widetilde{\mathcal{L}} = \boldsymbol{T}\mathcal{L}$ and $\widetilde{\mathcal{L}}' = \boldsymbol{T}'\mathcal{L}'$.
- *EstimateRSH*. Apply Eq. (16) to linearly estimate $\widetilde{\boldsymbol{H}}$ via SVD according to ray–ray correspondences $\{\widetilde{\mathcal{L}}\} \leftrightarrow \{\widetilde{\mathcal{L}}'\}$.
- *DenormalizeRSH*. Set the ray-space homography $\boldsymbol{H} = \frac{1}{t'_{22}t'_{44}}\boldsymbol{T}'^{\top}\widetilde{\boldsymbol{H}}\boldsymbol{T}$, where t'_{ij} is i-th row and j-th column element of $\boldsymbol{T}'$.

Ray normalization is an essential step in ray-space homography estimation. It must not be considered optional. In addition, this ray normalization can be applied to various linear estimation methods of light field imaging.

Ray-Space Infinity Homography Estimation After the computation of $\boldsymbol{H}$ from ray–ray correspondences, let us revisit the estimated RSIH $\hat{\boldsymbol{H}}_\infty$. As mentioned above, it has been linearly solved and decomposed up to a scale factor by Eq. (16). We first eliminate the scale factor based on conjugate condition. Specifically, a rotation matrix $\boldsymbol{R}$ has eigenvalues $(1, e^{i\theta}, e^{-i\theta})$. θ refers to the angle of rotation about a rotation axis $\boldsymbol{v}$, which is satisfied by $\boldsymbol{Rv} = \boldsymbol{v}$ (Hartley and Zisserman 2003). The rotation can also be calculated from θ and $\boldsymbol{v}$. Given that $\boldsymbol{H}_\infty$ is also a conjugate rotation (i.e., a similar matrix of rotation) according to Eq. (13), its eigenvalues are preserved under a conjugate relationship so the eigenvalues of $\hat{\boldsymbol{H}}_\infty$ are also $(1, e^{i\theta}, e^{-i\theta}, 1, e^{i\theta}, e^{-i\theta})$ up to a common scale. Subsequently, the scale factor is computed from the average of real eigenvalues of $\hat{\boldsymbol{H}}_\infty$. Overall, the accurate $\boldsymbol{H}_\infty$ is solved without scale by the conjugate relationship, so does $\boldsymbol{H}$. Note that, the complex eigenvalues also determine the angle θ through which the LFC rotates. It means the rotation can be directly solved with $\boldsymbol{H}_\infty$.

Special Imaging Conditions In this part, we begin the consideration of self-calibrating an LFC under special imaging conditions. The situation first considers here is the one in which the LFC rotates about its center but does not translate ($\boldsymbol{t} = \boldsymbol{0}$), i.e. pure rotation. This situation occurs frequently. According to Eqs. (11) and (13), the ray-space homography $\boldsymbol{H}$ is equivalent to RSIH $\boldsymbol{H}_\infty$, namely $\boldsymbol{H} = \boldsymbol{H}_\infty$. Consequently, we could directly estimate RSIH without scale based on Eq. (16) and conjugate rotation. In addition, pure rotation is a convenient motion to simplify the formulation of ray-space homography and further self-calibrate an LFC. In practice, when the LFC is not completely rotated about its center, the translation compared to the distance of scene points is small and then could be neglected. The constrained nature of the motion makes self-calibration of the LFC simpler. However, compared with complex infinity homography estimation for traditional self-calibration, the simplification for LFC's self-calibration algorithm is not significant. RSIH can be easily extracted from ray-space homography due to the special design of the LFC.

Secondly, a case of some practical importance is that of an LFC translating without rotation. Suppose the motion of an LFC is a pure translation, substituting $\boldsymbol{R} = \boldsymbol{I}$ into Eqs. (11) and (13), $\boldsymbol{H}$ can be simplified as ray-space translation homography $\boldsymbol{H}_{\boldsymbol{t}}$, that is $\boldsymbol{H} = \boldsymbol{H}_{\boldsymbol{t}}$. However, according to Eq. (14), $\boldsymbol{H}_{\boldsymbol{t}}$ does not affect the direction vectors of rays from the plane at infinity. It means that we can not use ray–ray correspondences under pure translation to compute RSIH, which is all that is needed to further estimate RAC and self-calibrate an LFC. In practice, slightly rotating LFC to record light fields is necessary for self-calibration.

4.2 Closed-Form Initialization

It is easy to verify that Eq. (15) is not influenced by k_i and k_j according to *Property* 1. In general, the RAC $\boldsymbol{\omega}$ is linearly solved in form of Kronecker product $\otimes$. However, RAC $\boldsymbol{\omega}$ is a symmetric matrix so that we revise Eq. (15) in the form of $\boldsymbol{Ab} = \boldsymbol{0}$ for simplicity and robustness,

$$\begin{bmatrix} h_{11}^2-1 & h_{12}^2 & 2h_{11}h_{13} & 2h_{12}h_{13} & h_{13}^2 \\ h_{21}^2 & h_{22}^2-1 & 2h_{21}h_{23} & 2h_{22}h_{23} & h_{23}^2 \\ h_{11}h_{31} & h_{12}h_{32} & h_{11}h_{33}+h_{31}h_{13}-1 & h_{12}h_{33}+h_{32}h_{13} & h_{13}h_{33} \\ h_{21}h_{31} & h_{22} & h_{21}h_{33}+h_{31}h_{23} & h_{22}h_{33}+h_{32}h_{23}-1 & h_{23}h_{33} \\ h_{31}^2 & h_{32}^2 & 2h_{31}h_{33} & 2h_{32}h_{33} & h_{33}^2-1 \\ h_{11}h_{21} & h_{12}h_{22} & h_{11}h_{23}+h_{21}h_{13} & h_{12}h_{23}+h_{22}h_{13} & h_{13}h_{23} \\ h_{41}^2-1 & h_{51}^2 & 2h_{41}h_{61} & 2h_{51}h_{61} & h_{61}^2 \\ h_{42}^2 & h_{52}^2-1 & 2h_{42}h_{62} & 2h_{52}h_{62} & h_{62}^2 \\ h_{41}h_{43} & h_{51}h_{53} & h_{41}h_{63}+h_{43}h_{61}-1 & h_{51}h_{63}+h_{53}h_{61} & h_{61}h_{63} \\ h_{42}h_{43} & h_{52} & h_{42}h_{63}+h_{43}h_{62} & h_{52}h_{63}+h_{53}h_{62}-1 & h_{62}h_{63} \\ h_{43}^2 & h_{53}^2 & 2h_{43}h_{63} & 2h_{53}h_{63} & h_{63}^2-1 \\ h_{41}h_{42} & h_{51}h_{52} & h_{41}h_{62}+h_{42}h_{61} & h_{51}h_{62}+h_{52}h_{61} & h_{61}h_{62} \end{bmatrix} \begin{bmatrix} k_u^2 \\ k_v^2 \\ k_u u_0 \\ k_v v_0 \\ 1+u_0^2+v_0^2 \end{bmatrix} = \mathbf{0}, \tag{17}$$

where h_{ij} denotes the i-th row and j-th column element of $\boldsymbol{H}$. $\boldsymbol{A}$ is a 12×5 matrix whose rank is sufficient for solving $\boldsymbol{b}$ with an unknown scaling. It is interesting to note that a non-zero solution $\boldsymbol{b}$ can be obtained by at least one RSIH (estimated from a pair of light fields). Note that $\boldsymbol{b}$ includes 5 distinct non-zero elements of a symmetric matrix $\boldsymbol{K}_{uv}^{\top}\boldsymbol{K}_{uv}$. $\hat{\boldsymbol{K}}_{uv}$ is linearly solved from $\boldsymbol{K}_{uv}^{\top}\boldsymbol{K}_{uv}$ by Cholesky factorization (Hartley and Zisserman 2003).

As discussed in Sect. 3, ray-based RAC and RSIH could be regarded as the generalization of point-based IAC and infinity homography in the higher dimension. The traditional self-calibration algorithm is therefore the special case of the proposed algorithm when an LFC has only one view (i.e., traditional camera) according to Eq. (17). In the traditional self-calibration algorithm, 3×3 infinity homography matrix is estimated from point-point correspondences and used to linearly calculate IAC and intrinsic parameters, as mentioned by Hartley and Zisserman (2003). It is common sense that traditional self-calibration is numerical instability, but the proposed self-calibration algorithm can provide a stable linear solution based on two reasons. Firstly, as discussed in Sect. 4.1, sufficient ray–ray correspondences are used to stably and accurately compute ray-space homography. The more accurate $\boldsymbol{H}$, the more stable RAC and intrinsic parameters can be obtained according to Eq. (17). Secondly, considering P denotes the number of RSIH, $(12 \times P) \times 5$ linear equations are solved to compute RAC, while the size of linear equations for traditional self-calibration is $(6 \times P) \times 5$. The higher dimension means that RAC estimation is a more stable solution compared with IAC estimation.

Based on the estimated intrinsic parameters $\hat{\boldsymbol{K}}_{uv}$ and ray-space homography $\boldsymbol{H}$, rotation $\boldsymbol{R}$ and translation $\boldsymbol{t}$ are further computed,

$$\boldsymbol{R} = \frac{1}{2}\left(\hat{\boldsymbol{K}}_{uv}^{-\top}\boldsymbol{H}_{11}\hat{\boldsymbol{K}}_{uv}^{\top} + \hat{\boldsymbol{K}}_{uv}\boldsymbol{H}_{22}\hat{\boldsymbol{K}}_{uv}^{-1}\right), \tag{18}$$

$$[\boldsymbol{t}]_{\times} = \hat{\boldsymbol{K}}_{ij}\boldsymbol{H}_{12}\hat{\boldsymbol{K}}_{uv}^{-1}\boldsymbol{R}^{\top} = \lambda\hat{\boldsymbol{K}}_{uv}^{-\top}\boldsymbol{H}_{12}\hat{\boldsymbol{K}}_{uv}^{-1}\boldsymbol{R}^{\top}, \tag{19}$$

where $\boldsymbol{H}_{ij}$ indicates the i-th row and j-th column $3{\times}3$ block matrix of $\boldsymbol{H}$, as shown in Eq. (11). According to *Property* 1, $\lambda = k_i k_v$. Considering $\boldsymbol{H}_{\infty}$ and $\boldsymbol{K}_{uv}$ are accurate without scaling, each part of Eq. (18) is already an orthogonal rotation matrix with determinant unit. Therefore, there is no need to orthogonalize the estimated rotation. The rotation averaging in Eq. (18) can be computed according to the method proposed by Hartley et al. (2013).

Remarks λ is an empirical parameter without physical meaning that helps us to uniform the relative translation between each pair of light fields to global scaling. Since the traditional self-calibration method cannot recover a uniform scaling directly, the translation between the first and second cameras sets to a unit vector. However, due to special design of an LFC, the baseline (i.e., intrinsic parameters k_i and k_j) could be considered as the translation between neighboring sub-aperture images (Bok et al. 2017; Zhang et al. 2019c). More importantly, it keeps constant during the self-calibration of an LFC, so λ is constant. Consequently, compared with the traditional self-calibration, the proposed LFC self-calibration is easy to estimate the translation up to a uniform scaling for 3D metric reconstruction. In addition, since the Euclidean distance of static scenes is not provided in advance, k_i and k_j as the translation between sub-aperture images cannot be recovered, as shown in Eq. (17). According to the observation about LFC calibration results, we conclude that $\frac{k_u}{k_i} = \frac{k_v}{k_j} = r_m$, where r_m denotes the radius of the micro-lens in pixels. Consequently, we empirically set λ to $\frac{k_u k_v}{r_m}$.

4.3 Non-linear Optimization

The initial solution is then refined via non-linear optimization. Considering RSHC constrains the intrinsic and extrinsic parameters well, we minimize the geometrically more meaningful Sampson error based on RSHC and $\{\mathcal{L}_i\}_n \leftrightarrow \{\mathcal{L}'_j\}_m$ of N 3D points,

$$\sum^{N}\sum_{i}^{m}\sum_{j}^{n}\frac{\left|\left(\boldsymbol{p}_i^\top, \boldsymbol{n}_i^\top\right)\boldsymbol{H}\left(\boldsymbol{n}'^\top_j, \boldsymbol{p}'^\top_j\right)^\top\right|}{\left\|\boldsymbol{H}^\top\left(\boldsymbol{p}_i^\top, \boldsymbol{n}_i^\top\right)^\top\right\| + \left\|\boldsymbol{H}\left(\boldsymbol{n}'^\top_j, \boldsymbol{p}'^\top_j\right)^\top\right\|}, \tag{20}$$

where $\|\cdot\|$ denotes L_2-norm, and $\boldsymbol{H}$ is formulated by $\boldsymbol{K}$, $\boldsymbol{R}$ and $\boldsymbol{t}$ according to Eq. (10). Compared with the re-projection error (Zhang et al. 2019c) or ray-projection error (Dansereau et al. 2013), Sampson error does not depend on the unstable reconstructed 3D points within a light field. Also, compared with algebraic distance RSHC, the Sampson distance is the first-order approximation of the ray–ray geometric distance. According to the geometry definition of Plücker coordinates, Eq. (20) which describes the ray–ray Sampson distance in $\mathbb{P}^3$ is also geometrically equivalent to the point-point Sampson distance in $\mathbb{P}^5$. Moreover, the ray–ray Sampson distance in Eq. (20) for light fields is also applied to outliers detection within a RANSAC framework (Fischler and Bolles 1981). We linearly estimated the ray-space homography matrix from random ray–ray correspondences. The Sampson distances of all ray–ray correspondences are then calculated, and the outliers are distinguished and discarded with a given threshold. The radial distortion caused by main lens is neglected in the proposed self-calibration method. Consequently, the non-linear optimization excludes the distortion coefficients.

Considering the assumption of constant intrinsic parameters during self-calibration, multiple light fields are recorded to improve the effectiveness and robustness of the proposed self-calibration method. In order to extend Eq. (20) to multiple light fields, we calculate ray–ray Sampson distances from random pairs of light fields to select arbitrary light field in the pair with minimum distances as the reference light field. In practice, we capture $P + 1$ light fields using a same LFC. For each pair of reference light field and p-th light field, $1 \le p \le P$, the RSIH is first estimated. With the increasing RSIHs, Eq. (17) is then stacked as $12 \times P$ linear equations to obtain robust intrinsic parameters. The relative pose of p-th light field, consisting of $\boldsymbol{R}_p$ and $\boldsymbol{t}_p$, is subsequently estimated. For each pair of light fields, a non-linear cost function is established according to Eq. (20). Finally, P cost functions are accumulated together to optimize intrinsic and extrinsic parameters. To minimize the above non-linear functions, we parameterize rotation $\boldsymbol{R}$ with its Rodrigues form (Faugeras 1993). Then we utilize Levenberg-Marquardt optimization solver lsqnonlin in Matlab.

Once the LFC intrinsic and extrinsic parameters are determined, the ray-space projection matrices can be rebuilt according to Eq. (7). Subsequently, the ray–ray correspondences of a 3D point are transformed in a common camera coordinate frame via Eq. (10). For a certain 3D point observed by generalized rays, according to Eq. (3), it can be reconstructed,

$$\left([\boldsymbol{q}]_\times \ \boldsymbol{m}\right)\begin{bmatrix}\boldsymbol{X}\\ 1\end{bmatrix} = 0. \tag{21}$$

Although we have removed most mismatched ray–ray correspondences, considering the small baseline of an LFC, the triangulation is adopted from all ray–ray correspondences according to Eq. (21) within a RANSAC framework based on midpoint method (Hartley and Sturm 1997). Similarly, we employ the Levenberg-Marquardt algorithm to refine the triangulated 3D points via minimizing the re-projection error. The self-calibration and 3D metric reconstruction algorithm for an LFC is summarized in Alg. 1.

Algorithm 1 LFC Self-Calibration and 3D Reconstruction.

```
Require: Ray–ray correspondences $\{\mathcal{L}'\} \leftrightarrow \{\mathcal{L}\}$ of $P + 1$ light fields
Ensure: Intrinsic parameters $(k_i, k_j, k_u, k_v, u_0, v_0)$,
         Extrinsic parameters $\boldsymbol{R}_p, \boldsymbol{t}_p, (1 \le p \le P)$,
         3D metric reconstruction $\boldsymbol{X}$.
for $p = 1$ to $P$ do
    $\widetilde{\mathcal{L}} = NormalizeRays(\mathcal{L}, \boldsymbol{T}_p)$
    $\widetilde{\mathcal{L}}' = NormalizeRays(\mathcal{L}', \boldsymbol{T}'_p)$
    $\widetilde{\boldsymbol{H}}_p = EstimateRSH(\widetilde{\mathcal{L}}, \widetilde{\mathcal{L}}')$                ▷ Eq. (16)
    $\boldsymbol{H}_p = DenormilizeRSH(\widetilde{\boldsymbol{H}}_p, \boldsymbol{T}_p, \boldsymbol{T}'_p)$
    $\boldsymbol{H}_{\infty,p} = DecomposeRSIH(\boldsymbol{H}_p)$                ▷ Eq. (13)
end for
$\boldsymbol{\omega} = FormulateRAC(\bigcup_{p=1}^{P} \boldsymbol{H}_{\infty,p})$                ▷ Eq. (15)
$\boldsymbol{K}_{uv} = CalculateIntrin(\boldsymbol{\omega})$                ▷ Eq. (17)
for $p = 1$ to $P$ do
    $(\boldsymbol{R}_p, \boldsymbol{t}_p) = CalculateExtrin(\boldsymbol{H}_p, \boldsymbol{K}_{uv})$                ▷ Eqs. (18, 19)
end for
$OptimizeSampsonError(\boldsymbol{K}, \bigcup_{p=1}^{P-1}(\boldsymbol{R}_p, \boldsymbol{t}_p))$                ▷ Eq. (20)
for $p = 1$ to $P$ do
    $\boldsymbol{P}_p = FormulateRSH(\boldsymbol{K}, \boldsymbol{R}_p, \boldsymbol{t}_p)$                ▷Eq. (7)
end for
$\boldsymbol{X} = Triangulate(\boldsymbol{K}\mathcal{L}, \bigcup_{p=1}^{P} \boldsymbol{P}_p\mathcal{L}')$                ▷Eq. (21)
$OptimizeReProjectionError(\boldsymbol{X})$.
```

5 Experiments

In this section, we experimentally evaluate the proposed self-calibration algorithm on both synthetic LFCs and commercial LFCs. To evaluate the performance of the proposed method, similar to Li et al. (2008), direction errors of rotation and translation are defined when ground truth data is available,

$$e_{\boldsymbol{R}} = \arccos\left(\mathrm{trace}(\boldsymbol{R}^\top \boldsymbol{R}_{GT}) - 1\right)/2, \tag{22}$$

$$e_{\boldsymbol{t}} = \arccos(\boldsymbol{t}^\top \boldsymbol{t}_{GT}/(\|\boldsymbol{t}\|\|\boldsymbol{t}_{GT}\|)). \tag{23}$$

5.1 Simulated Data

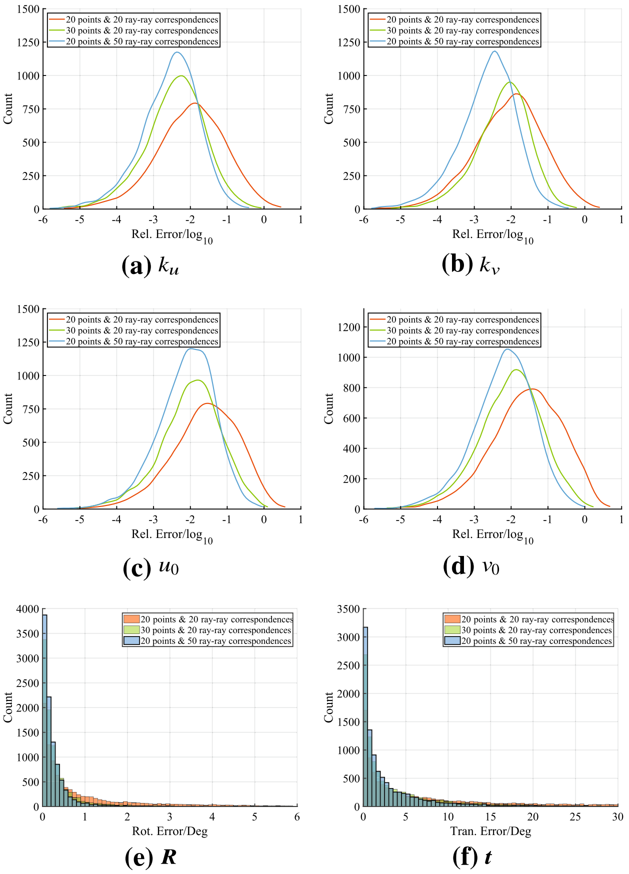
In order to evaluate the performance of the proposed method, we simulate a realistic LFC, close to a Lytro Illum, with 11×11 views, a baseline of $0.36mm$, a focal length of 500, and the sub-aperture image resolution of 540×360. The rotation angles between a couple of light fields are randomly generated from $-30°$ to $30°$, while the translation $\boldsymbol{t}$ is randomly chosen in the cube $[-0.1, 0.1]^3$. Besides, the depth of 3D points is set to a range between $0.2m$ and $0.8m$. We randomly generate ray–ray correspondences from these 3D points based on camera parameters so that we obtain plausible input close to real-world scenarios.

Numerical Stability.

In this experiment, we evaluate the numerical stability of the proposed closed-form initialization. We generate random but feasible noise (0.1-1.0 pixels) simulated problem instances. Geometrically realistic ray–ray correspondences between two light fields are involved in the linear solution. Figure 5 presents the distribution of the $\log_{10}$ relative errors of intrinsic parameters and the histogram of rotation and translation direction errors for $10,000$ instances. As shown in Fig. 5, three combinations of 3D points and ray–ray correspondences of each 3D point are performed. We can see that the numerical stability and accuracy are obviously influenced by the number of ray–ray correspondences (i.e. the number of 3D points multiple the number of ray–ray correspondences of each 3D point). We use $\log_{10}$ to evaluate the relative error of intrinsic parameters. As shown in Fig. 5, the peaks of error distributions are increased with the number of ray–ray correspondences. Moreover, as shown in Fig. 5, the more ray–ray correspondences, the higher distribution of small direction errors. All results demonstrate the numerical stability of the proposed linear initialization. The mean errors of each parameter are also listed in Table 2. The relative errors of intrinsic parameters and direction errors of extrinsic parameters are decreased with the number of ray–ray correspondences, which verifies the accuracy of the proposed linear initialization. Besides, Table 2 also shows the mean execution time of the linear initialization. According to Eqs. (16) and (17), the main time complexity is spent on the solution of $\boldsymbol{H}$ from different ray–ray correspondences. The execution time of the linear solution increases with the ray–ray correspondences when the number of light field pairs is constant. In summary, these linear solutions show that the proposed linear algorithm does not suffer from numerical instability and is a good enough starting guess for the non-linear optimization Eq. (20).

Table 2 Mean relative errors of intrinsic parameters, mean direction errors of extrinsic parameters and mean execution time of linear initialization under different number of 3D points and ray–ray correspondences of each 3D point

		20 points 20 rays	30 points 20 rays	20 points 50 rays
	k_u	6.8567	1.8197	1.1679
Intrinsic	k_v	5.7910	1.9498	0.8622
Unit: %	u_0	12.6815	4.4960	3.0303
	v_0	15.9066	4.8713	2.7890
Extrinsic	$\boldsymbol{R}$	1.2806	0.4184	0.2692
Unit: deg	$\boldsymbol{t}$	10.2459	4.5236	3.1055
Time (Unit: s)		0.0561	0.1853	0.3055



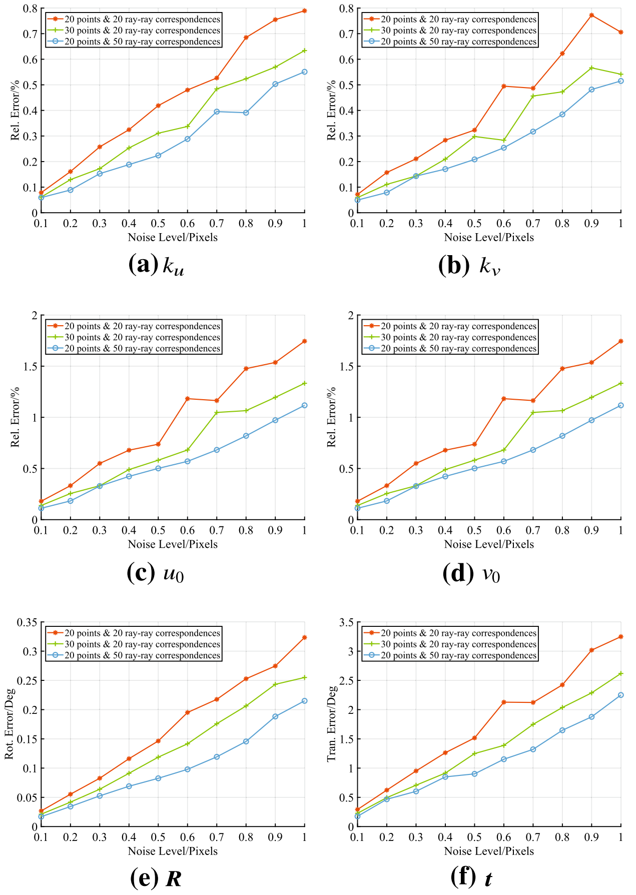
(a) k_u **(b)** k_v

(c) u_0 **(d)** v_0

(e) $\boldsymbol{R}$ **(f)** $\boldsymbol{t}$

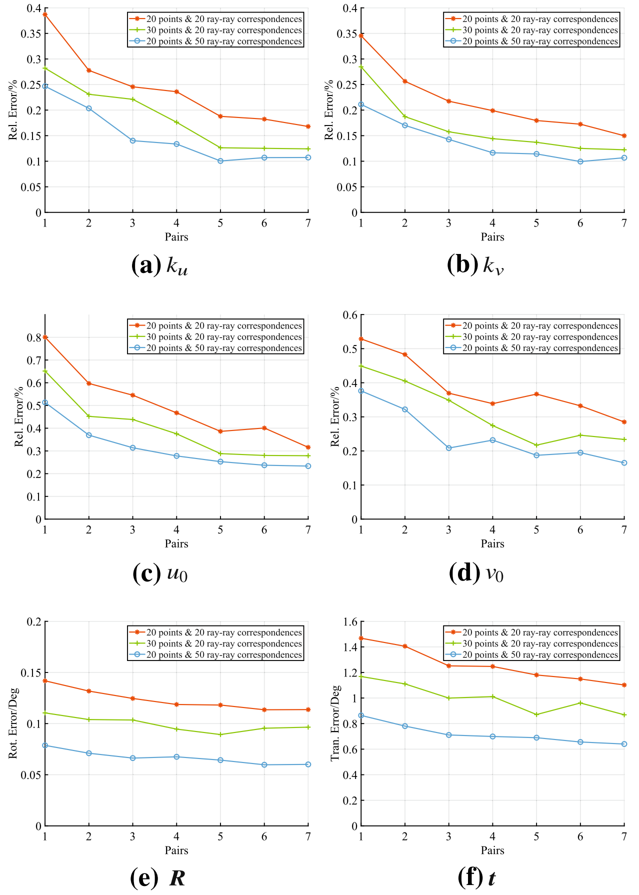
Fig. 5 Distribution of relative errors of intrinsic parameters and direction errors of rotation and translation for $10,000$ random simulated linear solution instances

Noise Resilience. In this experiment, we generate one pair of light fields to examine the noise resilience on the proposed method. Depending on the experiment, different levels of white Gaussian noises varying from 0.1 to 1.0 pixels with a 0.1 pixels step are then added to the direction of projected rays. For each noise level, we carry out 150 trials, each of which includes three combinations of 3D points and ray–ray correspondences, as shown in Fig. 6. Figure 6 summarizes the non-linear optimized results compared with ground truth, including mean relative errors of intrinsic parameters and mean direction errors of rotation and translation. It certi-



(a) k_u (b) k_v (c) u_0 (d) v_0 (e) $\boldsymbol{R}$ (f) $\boldsymbol{t}$

Fig. 6 Performance evaluation of intrinsic and extrinsic parameters on the simulated data with different levels of noise σ



(a) k_u (b) k_v (c) u_0 (d) v_0 (e) $\boldsymbol{R}$ (f) $\boldsymbol{t}$

Fig. 7 Performance evaluation of intrinsic and extrinsic parameters on the simulated data with different number of light field pairs

fies that the errors increase almost linearly with the noise level. Compared with linear solutions, the mean errors of each parameter descend obviously, which verifies the effectiveness of non-linear optimization. Meanwhile, the errors reduce with the number of RSHCs (i.e. points×ray–ray correspondences). Specifically, for $\sigma = 0.5$ pixels which is larger than normal noise in commercial LFCs, the relative error of (k_u, k_v) and (u_0, v_0) and direction errors of $\boldsymbol{R}$ and $\boldsymbol{t}$ are less than 0.45%, 0.8%, 0.15° and 1.5° respectively. It also verifies that the proposed non-linear optimization remains robust at high noise.

Number of Constraints In this experiment, we investigate the influence of the number of constraints on the accuracy of the proposed method. To this end, we vary the number of light field pairs from 1 to 7. Similarly, three combinations of the number of 3D points and the ray–ray correspondences from each point are involved. We execute 100 independent trials, each of which added with 0.5 pixels Gaussian noise. The mean relative errors of intrinsic parameters and mean direction errors of rotation and translation with increasing light field pairs are shown in Fig. 7. We can see that the errors of intrinsic parameters decrease significantly, while the direction errors are roughly constant. The reason why the performance of intrinsic parameters is better is related to increasing equations result stable linear solutions according to Eq. (17) and hence convergence into better minima. While the better intrinsic parameters lead to the small improvement of rotation and translation as shown in Fig. 7. With the increasing constraints, the proposed method presents better solutions. When the number of light field pairs is more than 4, the errors are descending slower. In summary, all results demonstrate the effectiveness of the proposed method with increasing constraints.

5.2 Real Scene

To further substantiate the proposed self-calibration algorithm that experiments on real scene light fields are performed. Using a Lytro Illum, two real scene light field datasets, named as "Board-1" and "Board-2", are collected from different scenes with checkerboard shown in Fig. 9a. Given that the proposed algorithm is the first attempt to self-calibrate an LFC, the checkerboard can help to quantitatively compare the effectiveness of the proposed method

Table 3 Inliers proportion (unit: %) on the collected datasets

Dataset	# light fields	Inliers proportion
Board-1	10	49.15
Board-2	9	51.60
Toy-1	8	53.04
Toy-2	13	52.93
Toy-3	10	53.13
Teemo-1	13	50.81
Teemo-2	18	51.58
Desk	18	51.56

on intrinsic and extrinsic estimation with the state-of-the-art LFC calibration method (Zhang et al. 2019c). Considering the checkerboard may cause concerns on self-calibration, another six datasets "Toy-1", "Toy-2", "Toy-3", "Teemo-1", "Teemo-2" and "Desk" are also captured to demonstrate the estimated results (both camera parameters and reconstructed 3D structures), as shown in Fig. 10a.

The number of light fields on each dataset is listed in the second column of Table 3. Taken the small baseline of an LFC into consideration, the depth of real scenes from an LFC ranges from $0.3m$ to $0.8m$. In order to demonstrate the performance of the proposed algorithm, LFC configurations, such as focal length and zoom factor, are different between different datasets. Note that, the LFC configurations are constant in a dataset to facilitate self-calibration.

5.2.1 Pre-processing

The pre-processing before LFC self-calibration includes the raw light field decoding, extraction of ray–ray correspondence and inliers detection.

The raw micro-lens image recorded by an LFC can be decoded using an open-source toolbox (Dansereau et al. 2013) or Lytro Power Tool (Lytro 2011). However, compared with the open-source toolbox, Lytro Power Tool also applies rectification, reducing the appearance of lens distortion. As mentioned in Sect. 4, the main lens distortion has been neglected in the proposed method. Consequently, we utilize Lytro Power Tool to decode raw data to rectified light field. The raw data is first de-bayered and pixel-aligned to an orthogonal grid of sub-images. It is easy to load and interpret as a 2D array of 2D sub-images. There are 14 × 14 pixels per sub-image and about 541 × 376 sub-images. In addition to de-bayering and aligning, the Lytro Power Tool also applies rectification to reduce the appearance of lens distortion. Considering that marginal sub-aperture images suffer from severe vignetting and aberrations, the middle 11 × 11 sub-aperture images are used.

To self-calibrate an LFC, sets of rays which correspond to the same 3D point among light fields are first extracted. The sub-images of raw data and sub-aperture images of view points are different visualizations of light field recorded by an LFC and easy to be generated to describe the rays, as shown in Ng (2006). However, it is difficult to extract accurate rays in small sub-images. Therefore, similar to methods in Johannsen et al. (2016) and Nousias et al. (2019), we extract sparse features from every sub-aperture image in a light field via Difference of Gaussians (DoG). Different from the feature extraction for traditional images, light field is equivalent to a collection of sub-aperture image. The feature (u, v) extracted on a sub-aperture image of view (i, j) refers to a ray (i, j, u, v). The features of the central sub-aperture image are subsequently matched with those of other sub-aperture images via SIFT (Lowe 2004). Given that the view points of an LFC are regularly arranged on a plane, we filter the rays according to the invariant depth and generate sets of rays in a light field. After extracting the sets of rays in light fields, the ray–ray correspondences are then matched between different light fields via SIFT. For efficiency, only the features of the central sub-aperture images are matched to associate sets of rays between light fields.

As mentioned in Sect. 4, the Sampson distance of RSHC can be used to discard the outliers based on a RANSAC framework. The proportions of inlier ray–ray correspondences are summarized in Table 3. The proposed RANSAC framework with Sampson distance produces the reliable ray–ray correspondences for self-calibration. Fig. 8 also partially illustrates the result of inliers detection on datasets "Toy-2". Fig. 8 randomly marks 40 ray–ray correspondences on arbitrary sub-aperture images of a pair of light fields. It can be seen that the rays within a light field preserve the depth invariant (i.e. same disparity), and the ray–ray correspondences between light fields lie in the similar pixel of sub-aperture images without outliers. Table 3 and Fig. 8 quantitatively and qualitatively verify the performance of the proposed RSHC and its Sampson distance on inliers detection, respectively. In order to intuitively express ray–ray correspondences emanating from the same 3D scene point, we also randomly draw 5 sets of rays with 3D reconstruction in line form. As shown in Fig. 8, all rays corresponding to a 3D scene point in both light fields are utilized for 3D reconstruction. Moreover, datasets "Board-1" and "Board-2" combined with a checkerboard may cause concerns about the ray–ray correspondences. It provides sufficient ray–ray correspondences however introduces more outliers on the checkerboard due to the similar local structures. It is also demonstrated in Table 3, from which we can see the proportions of inliers on datasets "Board-1" and "Board-2" are less than those on datasets "Toy-1", "Toy-2" and "Toy-3" with similar scenes.

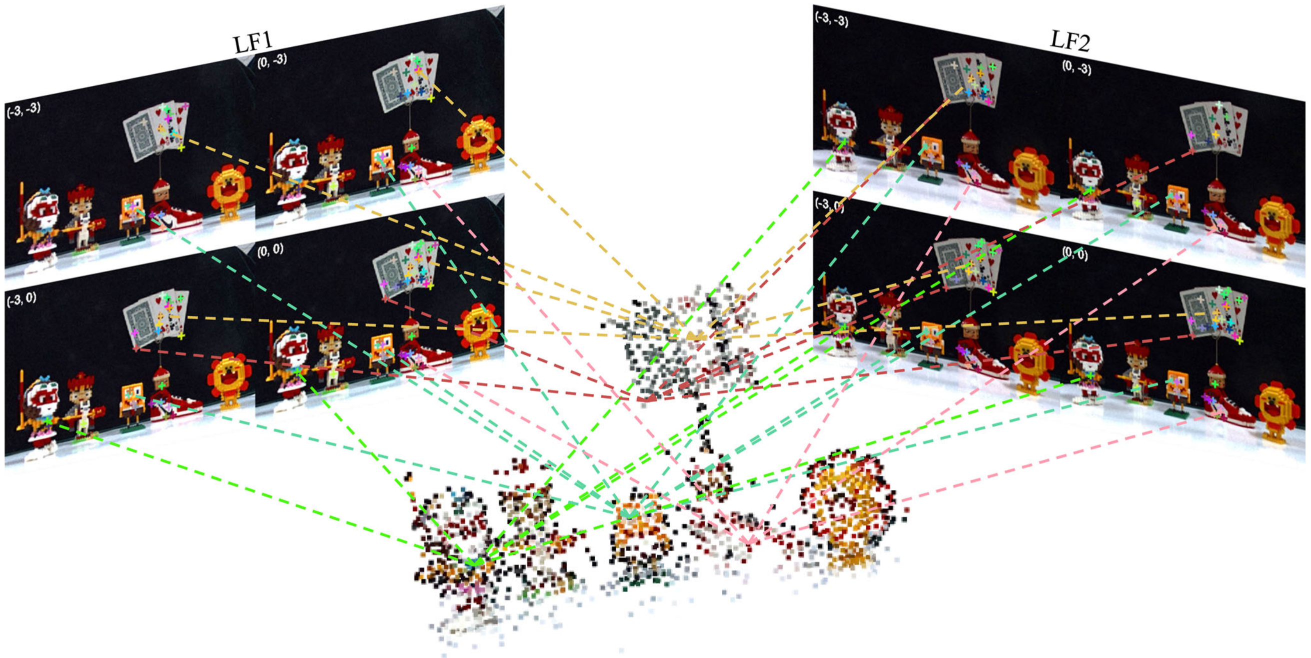


Fig. 8 Inliers detection on "Toy-2" based on ray-space homography estimation and its Sampson distance. 40 random inliers are marked on arbitrary sub-aperture images with colors, wherein 5 sets of ray–ray correspondences connected to the corresponding 3D reconstructed points are drawn in dashed line form

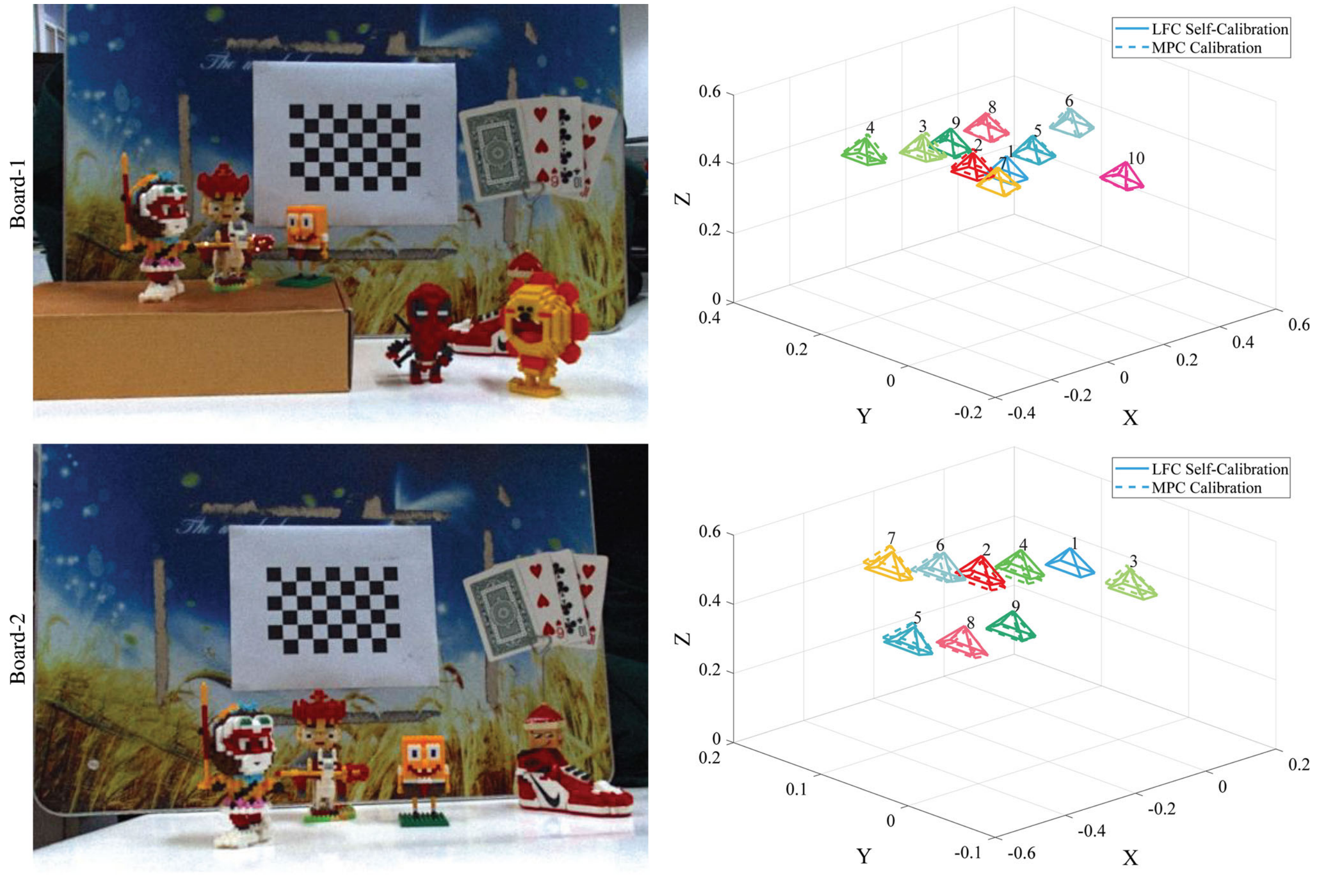
5.2.2 Self-Calibration

The checkerboards on datasets "Board-1" and "Board-2" make it possible to compare the proposed self-calibration meth-od with state-of-the-art calibration method represented by MPC (Zhang et al. 2019c), treating the latter as the gold standard. Note that, compared with the *isometric* parameters obtained by MPC, the proposed LFC self-calibration algorithm estimates *metric* parameters due to static scenes without a known physical size. Neither accurate ray–ray correspondences of corners nor Euclidean distance could be provided by the checkerboard for self-calibration. Consequently, baselines k_i and k_j which decide the magnitude of reconstruction are not compared. In addition to MPC, baseline methods proposed by Dansereau et al. (2013) and Bok et al. (2017) are other common LFC calibration methods using checkerboard. The reasons why the proposed method is not compared with other baseline methods have three. Firstly, the MPC outperforms compared with other baseline methods, as demonstrated by Zhang et al. (2019c). Secondly, the 12-free-parameter model provided by Dansereau et al. (2013) has redundancy and dependency. Intrinsic parameters estimated by Dansereau et al. (2013) can not be compared with the results of the proposed self-calibration. Thirdly, the baseline method proposed by Bok et al. (2017) uses line features on sub-images to calibrate an LFC. Although it can provide similar 6 intrinsic parameters compared with the proposed method, the checkerboard should be captured under an unfocused status to make the line feature detectable.

Table 4 Differences of intrinsic and extrinsic parameters obtained by the proposed self-calibration method compared with state-of-the-art calibration (Zhang et al. 2019c) on datasets with a checkerboard

	Intrinsic Unit: %				Extrinsic Unit: deg	
	k_u	k_v	u_0	v_0	$\boldsymbol{R}$	$\boldsymbol{t}$
Board-1	0.8422	0.8567	1.9509	2.0301	0.7866	5.6119
Board-2	1.3603	1.3736	1.5087	1.5556	0.5113	4.3980

Table 4 presents the differences of intrinsic parameters and relative poses, measured with relative errors (%) and direction errors (deg), respectively. Overall, the errors in Table 4 are closed to the gold standard except the direction error of translation. As shown in Eq. (18), the translation is decomposed from the estimated essential matrix, which includes the rotation. It results in that the estimated rotation is more accurate than translation, which is also demonstrated in simulated data. Besides, taken the noise of ray–ray correspondences into consideration, the results are acceptable. Moreover, the relative comparisons in Table 4 also demonstrate the performance of RSHCs to constrain the intrinsic and extrinsic parameters. In addition, Table 5 illustrates intrinsic parameters estimated by the proposed method and MPC calibration method on datasets "Board-1" and "Board-2".



(a) Central Sub-Aperture Image

(b) LFC Pose Comparison

Fig. 9 The central sub-aperture image of the reference light field. The comparison of pose estimation of the proposed self-calibration (colored solid lines) with that of state-of-the-art MPC calibration (colored dotted lines) (Color figure online)

Table 5 The comparison of intrinsic parameters of the proposed self-calibration with that of state-of-the-art calibration

	Board-1		Board-2	
	Ours	MPC	Ours	MPC
k_u	1.7861e–03	1.8013e–03	1.5585e–03	1.5376e–03
k_v	1.8320e–03	1.8164e–03	1.5150e–03	1.5364e–03
u_0	–0.4707	–0.4801	–0.4408	–0.4476
v_0	–0.3406	–0.3477	–0.3159	–0.3209

As mentioned in Zhang et al. (2019c), $(-k_i i, -k_j j, 0)^\top$ also indicates the translation between sub-aperture images within a light field which remains constant during self-calibration. For this reason, it is easy to constrain the translation between light fields up to a uniform scaling according to Eq. (19). In order to qualitatively visualize the differences of relative poses on datasets "Board-1" and "Board-2", we respectively illustrate the poses calculated by the proposed self-calibration and MPC calibration, as shown in Fig. 9b. Since the proposed method is metrically estimated, the translation vectors of MPC are scaled so that the translation vectors of two methods have the same norm. Clearly, the poses of the proposed self-calibration method and MPC are very similar. Although the direction errors of translation are larger than that of rotation in Table 4, we can see in Fig. 9b that the direction errors of translation have a smaller effect on pose visualization and 3D reconstruction compared with rotation.

5.2.3 3D Reconstruction

The checkerboard on previous datasets is convenient to compare the proposed method with calibration method, but it may cause concern that corners on the checkerboard may provide sufficient and accurate ray–ray correspondences for self-calibration. Consequently, we further validate the self-calibration on the datasets without any specific calibration targets to reconstruct 3D scenes. Considering that there is little self-calibration method designed for an LFC, we first use the Sampson errors of RSHCs to evaluate the effectiveness of the proposed self-calibration. Table 6 summarizes average

Table 6 Average ray–ray correspondences of each light field pair and mean Sampson errors (unit: 10^{-3}) for the estimation of LFC parameters on the collected datasets

	Toy-1	Toy-2	Toy-3	Teemo-1	Teemo-2	Desk
Ray–ray correspondences	9588.5	8134.7	8393.0	9677.2	6775.6	11320.8
Sampson error	2.2913	0.6507	1.6583	1.3778	1.2081	1.5469

Table 7 Intrinsic parameters estimated by the proposed method

	Toy-1	Toy-2	Toy-3	Teemo-1	Teemo-2	Desk
k_i	3.6795e–04	3.2618e–04	3.1994e–04	3.9684e–04	3.9098e–04	3.9820e–04
k_j	3.6303e–04	3.2963e–04	3.2811e–04	4.0820e–04	4.1113e–04	4.1058e–04
k_u	1.8397e–03	1.6309e–03	1.5996e–03	1.9842e–03	1.9549e–03	2.0618e–03
k_v	1.8152e–03	1.6481e–03	1.6405e–03	2.0410e–03	2.0557e–03	2.1204e–03
u_0	–0.5100	–0.4528	–0.4327	–0f.5608	–0.4557	–0.5537
v_0	–0.3232	–0.3014	–0.3084	–0.3235	–0.3606	–0.3848

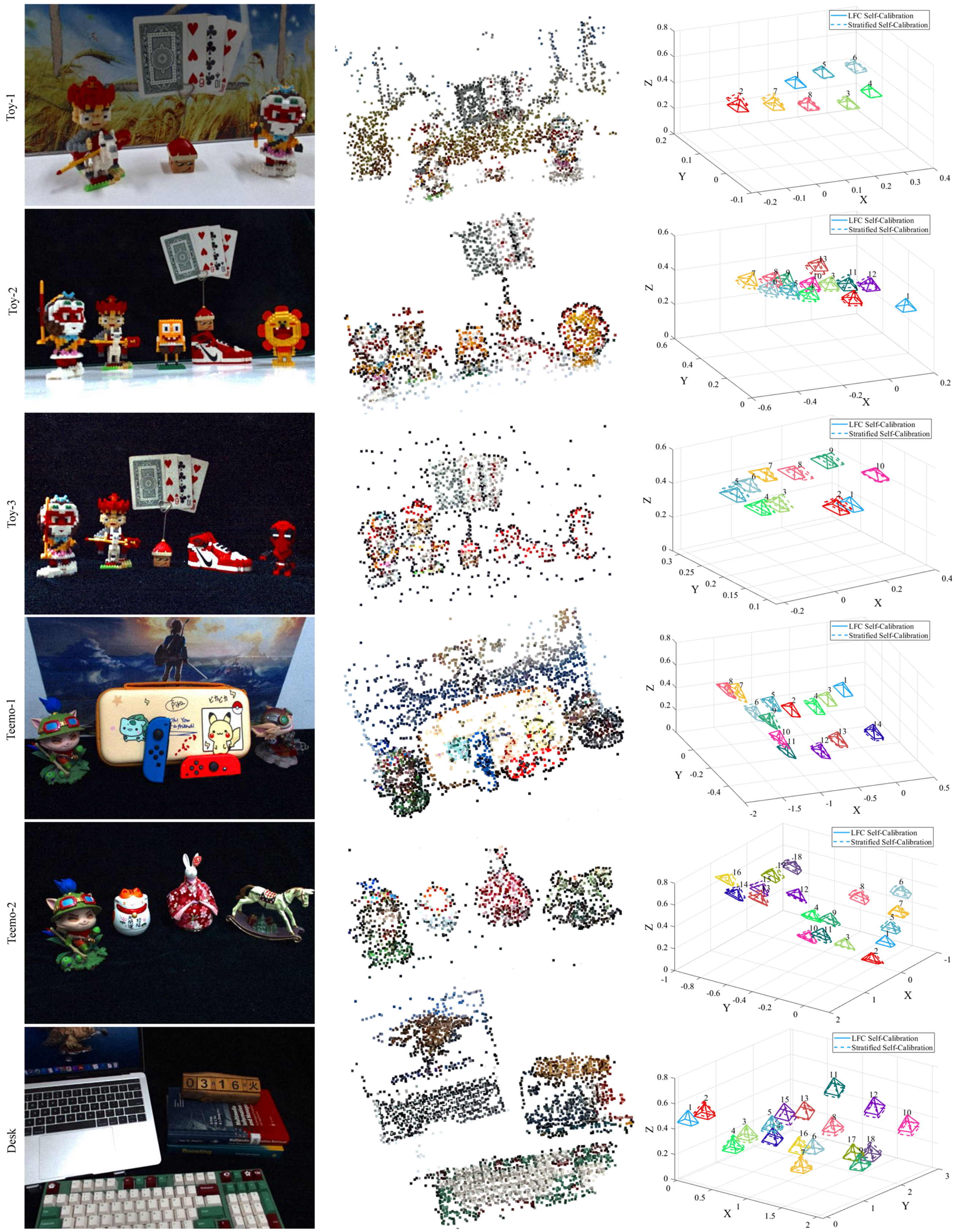
ray–ray correspondences of each pair of light fields and the mean Sampson errors on the collected datasets. The Sampson distance is a close approximation to the geometric distance between the ray and its corresponding ray after the transformation established by LFC parameters. Compared with the algebraic error, it is reasonable to utilize Sampson error with geometric meaning to represent the distance of ray–ray correspondences for optimization. As shown in Table 6, the ultra-small Sampson errors on collected datasets can verify the effectiveness of the proposed method for the estimation of LFC parameters. Specifically, we can see that the Sampson errors of each datasets have fluctuations with the number of ray–ray correspondences. In addition, Table 7 illustrates the results of intrinsic parameters estimation, where k_i and k_j are also listed. As discussed in Sect. 4.2, since the Euclidean distance of static scenes is not provided in advance, k_i and k_j as the translation between sub-aperture images cannot be estimated but help to constrain the relative translation between each pair of light fields up to a uniform scaling, so the LFC self-calibration recovers metric structure. Here, we empirically set the uniform scale to $\frac{k_u}{k_i} = \frac{k_v}{k_j} = 5$, which is the radius of view points.

Table 8 Differences of intrinsic and extrinsic parameters obtained by the proposed self-calibration method compared with the stratified self-calibration (Pollefeys et al. 1999) on the collected datasets

	Intrinsic Unit: %				Extrinsic Unit: deg	
	k_u	k_v	u_0	v_0	$\boldsymbol{R}$	$\boldsymbol{t}$
Toy-1	0.9851	0.9755	2.3192	2.1223	0.8743	4.4863
Toy-2	0.6777	1.7403	1.6828	1.7414	0.7939	6.4364
Toy-3	1.1606	1.2582	2.1606	2.2582	1.2947	4.8382
Teemo-1	0.9148	0.8722	2.0050	1.9277	0.9654	4.5930
Teemo-2	1.0562	1.0343	3.5861	2.1506	1.1100	6.1437
Desk	1.4010	1.3102	2.1171	2.2039	1.1184	6.4707

In order to further quantify the performance of intrinsic and extrinsic parameters estimation, we compared the output of the proposed self-calibration algorithm with the traditional stratified self-calibration method provided by Pollefeys and Van Gool (1999). Considering an LFC can be considered as a collection of traditional cameras, the stratified self-calibration treats each sub-aperture image independently, not accounting for the special design of the LFC. We note that a medium collected dataset of 10 light fields contains 1210 images, which is too large for traditional self-calibration. Consequently, we only use 3 × 3 sub-aperture images regularly arranged in the light field to perform stratified self-calibration. Table 8 summarizes the differences of intrinsic and extrinsic parameters, measured with relative errors (%) and direction errors (deg) respectively. Since the stratified self-calibration cannot recover a uniform scaling of translations directly, we scale the translation between center sub-aperture images of first and second light fields to the same norm with that of the LFC self-calibration method. To fair comparison with the stratified self-calibration, we perform the LFC self-calibration from the light fields with the same view sampling in the experiments of Table 8. Besides, Fig. 10c qualitatively presents the LFC pose comparison between the LFC self-calibration and the stratified self-calibration. All results demonstrate the performance of the proposed self-calibration method.

Once the LFC intrinsic and extrinsic parameters are obtained, the 3D metric reconstruction can be computed. Figure 10 qualitatively exhibits 3D reconstruction results and the estimated poses on the collected datasets. The median point of the reconstructed scene is set as the origin of the coordinate frame. The rotation and translation of the reference light field are set to the identity matrix and zero vector, respectively. Even if the physical magnitude of the scene is unknown, the relative poses and 3D scene reconstruction are still estimated with a common scaling. In addition,



(a) Central Sub-Aperture Image **(b)** 3D Metric Reconstruction **(c)** LFC Pose Comparison

Fig. 10 The central sub-aperture image of the reference light field, 3D metric reconstruction and LFC pose comparison on collected datasets. The proposed LFC self-calibration (colored solid lines) is compared with stratified self-calibration (colored dotted lines) (Color figure online)

Table 9 3D reconstructed points and mean re-projection errors (unit: pixels) of the reconstructed structures on the collected datasets

	Toy-1	Toy-2	Toy-3	Teemo-1	Teemo-2	Desk
3D points	924	669	632	2426	1402	3585
Re-projection	0.5984	0.8095	0.9975	1.0488	0.9663	0.7881

Table 9 summaries the number of 3D reconstructed points and the mean re-projection errors according to the reconstructed points. We can see that the re-projection errors is deduced with the increasing 3D points. Specifically, according to Table 6, the background on datasets “Toy-1” and “Teemo-1” could provide more ray–ray correspondences to reconstruct 3D points. More 3D points for optimization will help to reduce the re-projection errors. The light fields of large-scale scenes on dataset “Desk” also extract sufficient ray–ray correspondences to increase accuracy of reconstruction, as shown in Tables 6 and 9. Moreover, we can also see from Fig. 10a that the central sub-aperture image on dataset “Toy-3” has more noise than the other two datasets “Toy-1” and “Toy-2” with similar scenes. It is another reason why the re-projection error on dataset “Toy-3” is larger than datasets “Toy-1” and “Toy-2”. In summary, all results have verified the effectiveness of the proposed self-calibration algorithm.

5.3 Limitations

In this part, we analyze the limitations of the proposed self-calibration algorithm to better understand the utility in practice. The main limitation of our algorithm is that the depth range of the scenes is limited due to the ultra-small baseline of the LFC. The disparity (pixel difference between neighboring sub-aperture images) which is defined by the depth of the scene point is also limited. Suppose the scene points lie at distance larger than $3m$ from an LFC so that their disparities are less than 0.1 pixels. This may cause inaccurate detection of rays in a light field, which is a common failure mode in methods that use ray–ray correspondences of light fields to estimate LFC pose. Therefore, when we reconstruct 3D scenes or estimate relative pose using an LFC in practice, the scene should not be too far from the LFC.

As discussed in Sect. 4.1, if there is no rotation between two light fields then the ray-space infinity homography cannot be estimated. This can be seen from Eq. (13), in the case of pure translation, the ray-space homography decomposes the ray-space translation homography and ignores the ray-space infinity homography. Consequently, when we capture light fields for self-calibration in practice, it is necessary to have the rotation between two light fields.

6 Closing Remarks

Light field cameras have gained increasing popularity and have been applied to a wide range of computer vision tasks, including 3D reconstruction from multiple views. Although using one single light field from an LFC, one is already able to compute a disparity map (despite suffering from very narrow baselines), recent researches have shown that taking multiple light fields significantly improves the 3D reconstruction accuracy. For multi-view LFCs, an easy-to-use and accurate self-calibration algorithm specifically designed for an LFC will be proven handy in practice. We have proposed in this paper a novel, compact, accurate and stable LFC self-calibration method, which is to the best of our knowledge the first of the kind in the literature. More importantly, while it is a commonly held opinion that self-calibration algorithm is usually fragile numerically no matter how elegant the theory is, in this paper we have demonstrated that this is not the case for the light-field camera, because of the rich redundancies and regularities presented in the ray-space of LFCs. For the future work, we will explore the correction of lens distortion induced by the main lens, whose effect has been neglected in the present work.

Acknowledgements The work was supported by NSFC under Grant 61531014, 61801396, 62031023. We thank the editors and reviewers for valuable suggestions on contents and experiments. We also thank Ying Feng for helpful supports on data collection. Qi Zhang was also supported by Innovation Foundation for Doctor Dissertation of Northwestern Polytechnical University under CX201919 and China Scholarship Council (CSC).

References

Bartoli, A., & Sturm, P. (2001). The 3d line motion matrix and alignment of line reconstructions. In *IEEE conference on computer vision and pattern recognition (CVPR)* (Vol. I, pp. I–I). IEEE.

Bartoli, A., & Sturm, P. (2004). The 3d line motion matrix and alignment of line reconstructions. *International Journal of Computer Vision, 57*, 159–178.

Bartoli, A., & Sturm, P. (2005). Structure-from-motion using lines: Representation, triangulation, and bundle adjustment. *Computer Vision and Image Understanding, 100*(3), 416–441.

Birklbauer, C., & Bimber, O. (2014). Panorama light-field imaging. *Computer Graphics Forum, 33*(2), 43–52.

Bok, Y., Jeon, H. G., & Kweon, I. S. (2014). Geometric calibration of micro-lens-based light-field cameras using line features. In *European conference on computer vision (ECCV)* (pp 47–61).

Bok, Y., Jeon, H. G., & Kweon, I. S. (2017). Geometric calibration of micro-lens-based light field cameras using line features. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), 39*(2), 287–300

Chandraker, M., Agarwal, S., Kahl, F., Nistér, D., & Kriegman, D. (2007a). Autocalibration via rank-constrained estimation of the absolute quadric. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1–8).

Chandraker, M., Agarwal, S., Kriegman, D., & Belongie, S. (2007b). Globally optimal affine and metric upgrades in stratified autocalibration. In *IEEE international conference on computer vision (ICCV)* (pp. 1–8).

Dansereau, D. G., Mahon, I., Pizarro, O., & Williams, S. B. (2011). Plenoptic flow: Closed-form visual odometry for light field cameras. In *2011 IEEE/RSJ international conference on intelligent robots and systems* (pp. 4455–4462).

Dansereau, D. G., Pizarro, O., & Williams, S. B. (2013). Decoding, calibration and rectification for lenselet-based plenoptic cameras. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1027–1034).

Dong, F., Ieng, S. H., Savatier, X., Etienne-Cummings, R., & Benosman, R. (2013). Plenoptic cameras in real-time robotics. *The International Journal of Robotics Research, 32*(2), 206–217.

Faugeras, O. (1993). *Three-dimensional computer vision: a geometric viewpoint*. MIT Press.

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM, 24*(6), 381–395.

Gherardi, R., & Fusiello, A. (2010). Practical autocalibration. In *European conference on computer vision (ECCV)* (pp. 790–801).

Guo, X., Yu, Z., Kang, S. B., Lin, H., & Yu, J. (2016). Enhancing light fields through ray-space stitching. *IEEE Transactions on Visualization and Computer Graphics, 22*(7), 1852–1861.

Gurdjos, P., Bartoli, A., Sturm, P. (2009). Is dual linear self-calibration artificially ambiguous? In *International conference on computer vision (ICCV)* (pp. 88–95). IEEE.

Habed, A., Pani Paudel, D., Demonceaux, C., & Fofi, D. (2014). Efficient pruning LMI conditions for branch-and-prune rank and chirality-constrained estimation of the dual absolute quadric. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 493–500).

Hartley, R., & Zisserman, A. (2003). *Multiple view geometry in computer vision*. Cambridge University Press.

Hartley, R., Trumpf, J., Dai, Y., & Li, H. (2013). Rotation averaging. *International Journal of Computer Vision, 103*(3), 267–305.

Hartley, R. I., & Sturm, P. (1997). Triangulation. *Computer vision and image understanding, 68*(2), 146–157.

Hartley, R. I., Hayman, E., de Agapito, L., & Reid, I. (1999). Camera calibration and the search for infinity. In *IEEE international conference on computer vision (ICCV)* (pp. 510–517).

Johannsen, O., Sulc, A., & Goldluecke, B. (2015). On linear structure from motion for light field cameras. In *IEEE international conference on computer vision (ICCV)* (pp. 720–728).

Johannsen, O., Sulc, A., Marniok, N., & Goldluecke, B. (2016). Layered scene reconstruction from multiple light field camera views. In *Asian conference on computer vision (ACCV)* (pp. 3–18)

Kneip, L., & Li, H. (2014). Efficient computation of relative pose for multi-camera systems. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 446–453).

Larsson, V., Kukelova, Z., & Zheng, Y. (2018). Camera pose estimation with unknown principal point. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2984–2992).

Levoy, M., & Hanrahan, P. (1996). Light field rendering. In *Proceedings of the 23rd annual conference on computer graphics and interactive techniques* (pp. 31–42).

Li, H., Hartley, R., & Kim, J. H. (2008). A linear approach to motion estimation using generalized camera models. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp 1–8).

Li, Y., Zhang, Q., Wang, X., & Wang, Q. (2019). Light field slam based on ray-space projection model. In *Optoelectronic imaging and multimedia technology VI* (Vol. 11187, p 1118706).

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision, 60*(2), 91–110.

Luong, Q. T., & Faugeras, O. D. (1997). Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of Computer Vision, 22*(3), 261–289.

Lytro. (2011). Lytro redefines photography with light field cameras. http://www.lytro.com.

Maybank, S. J., & Faugeras, O. D. (1992). A theory of self-calibration of a moving camera. *International Journal of Computer Vision, 8*(2), 123–151.

Ng, R. (2006). Digital light field photography. PhD thesis, Stanford University.

Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., Hanrahan, P., et al. (2005). Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report, 2*(11), 1–11.

Nistér, D. (2004). Untwisting a projective reconstruction. *International Journal of Computer Vision, 60*(2), 165–183.

Nousias, S., Lourakis, M., & Bergeles, C. (2019). Large-scale, metric structure from motion for unordered light fields. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp 3292–3301).

Paudel, D. P., Van Gool, L. (2018). Sampling algebraic varieties for robust camera autocalibration. In *European conference on computer vision (ECCV)* (pp. 275–292)

Pless, R. (2003). Using many cameras as one. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2*, 587–593.

Pollefeys, M., & Van Gool, L. (1999). Stratified self-calibration with the modulus constraint. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), 21*(8), 707–724

Pollefeys, M., Koch, R., & Van Gool, L. (1999). Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *International Journal of Computer Vision, 32*(1), 7–25.

Pottmann, H., & Wallner, J. (2009). *Computational line geometry*. Springer Science & Business Media.

Raytrix. (2013). 3d light field camera technology. http://www.raytrix.de.

Ren, Z., Zhang, Q., Zhu, H., & Wang, Q. (2017). Extending the FOV from disparity and color consistencies in multiview light fields. In *IEEE international conference on image processing (ICIP)* (pp. 1157–1161).

Seo, Y., Heyden, A., & Cipolla, R. (2001). A linear iterative method for auto-calibration using the dac equation. In *IEEE conference on computer vision and pattern recognition (CVPR)*.

Sturm, P. (2005). Multi-view geometry for general camera models. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1*, 206–212.

Triggs, B. (1997). Autocalibration and the absolute quadric. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 609—-614).

Vianello, A., Ackermann, J., Diebold, M., & Jähne, B. (2018). Robust hough transform based 3d reconstruction from circular light fields. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 7327–7335).

Zhang, Q., & Wang, Q. (2018). Common self-polar triangle of concentric conics for light field camera calibration. In *Asian conference on computer vision (ACCV)* (pp. 18–33).

Zhang, Q., Ling, J., Wang, Q., & Yu, J. (2019a). Ray-space projection model for light field camera. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 10121–10129).

Zhang, Q., Wang, X., & Wang, Q. (2019b). Light field planar homography and its application. In *Optoelectronic imaging and multimedia technology VI* (Vol. 11187, p. 111870S).

Zhang, Q., Zhang, C., Ling, J., Wang, Q., & Yu, J. (2019c). A generic multi-projection-center model and calibration method for light field cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), 41*(11), 2539–2552.

Zhang, Q., Wang, Q., Li, H., & Yu, J. (2020). Ray-space epipolar geometry for light field cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 1. https://doi.org/10.1109/TPAMI.2020.3025949.

Zhang, Y., Li, Z., Yang, W., Yu, P., Lin, H., & Yu, J. (2017a). The light field 3d scanner. In *IEEE international conference on computational photography (ICCP)* (pp. 1–9).

Zhang, Y., Yu, P., Yang, W., Ma, Y., & Yu, J. (2017b). Ray space features for plenoptic structure-from-motion. In *IEEE international Conference on computer vision (ICCV)* (pp. 4631–4639).