Ray Space Features for Plenoptic Structure-from-Motion

Yingliang Zhang[1] Peihong Yu[1] Wei Yang[2,3] Yuanxi Ma[1] Jingyi Yu[1,3]

[1]ShanghaiTech University
{zhangyl,yuph,mayx}@shanghaitech.edu.cn

[2]University of Delaware
wyangcs@udel.edu

[3]Plex-VR Inc.
jingyi.yu@plex-vr.com

Abstract

Traditional Structure-from-Motion (SfM) uses images captured by cameras as inputs. In this paper, we explore using light fields captured by plenoptic cameras or camera arrays as inputs. We call this solution plenoptic SfM or P-SfM solution. We first present a comprehensive theory on ray geometry transforms under light field pose variations. We derive the transforms of three typical ray manifolds: rays passing through a point or point-ray manifold, rays passing through a 3D line or ray-line manifold, and rays lying on a common 3D plane or ray-plane manifold. We show that by matching these manifolds across LFs, we can recover light field poses and conduct bundle adjustment in ray space. We validate our theory and framework on synthetic and real data on light fields of different scales: small scale LFs acquired using a LF camera and large scale LFs by a camera array. We show that our P-SfM technique can significantly improve the accuracy and reliability over regular SfM and PnP especially on traditionally challenging scenes where reliable feature point correspondences are difficult to obtain but line or plane correspondences are readily accessible.

1. Introduction

Structure from motion (SfM) estimates three-dimensional structures from two-dimensional image sequences by simultaneously recovering camera parameters (intrinsics, extrinsics, and poses) and 3D geometry of feature points. The problem is critical for computer vision and brings important insights on human visual perception. Tradition SfM is composed of three steps: extracting features and matching them across images, using the inlier features for camera intrinsic/extrinsic estimation, and bundle adjustment. Tremendous efforts have been made in the past decade [9, 16, 29, 38] on recovering indoor and outdoor scenes [2, 8, 15] with stunning performance in speed and quality [7] [30] etc.

In this paper, we investigate changing the input of the SfM problem: instead of using images captured by cameras, we use light fields captured by plenoptic cameras. Commodity plenoptic systems such as the Lytro light field camera and portable camera arrays can now record in a snapshot the radiance of nearly all rays emitting from every location and along every direction from the scene. We show that the 4D light fields provide a number of extremely useful ray geometric attributes that are not accessible in 2D images. Further we show that the use of these ray geometry attributes enable feature matching beyond 3D points as well as improve pose estimation and bundle adjustment. We call this technique plenoptic SfM or P-SfM.

A brute-force approach to P-SfM is to modify the regular SfM to handle the light field input data.

P-SfM vs. SfM. The most straightforward approach is to treat the recorded light fields as a set of (perspective) camera views or subaperture images[1] and directly apply regular SfM on subaperture views as shown in Fig.6. Such an approach, however, fails to use rich geometric constraints embedded in the ray space. For example, the subaperture cameras are confined to a plane and are regularly sampled on the grid. However, traditional SfM treats the relatively poses of subaperture images as unknown and hence does not effectively utilize this important constraint. Second, light fields provide some unique geometry properties over a 2D image, e.g., rays passing through a 3D line lie on a bilinear subspace (Fig.2) and one can derive pose variations by analyzing how this subspace transforms.

P-SfM vs. PnP. Another alternative is to first estimate the 3D position of feature points using, for example, one light field, and then apply the perspective-n-point algorithm [19] by matching them to feature pixels in the second light field.

[1]We adopt the subaperture image notation to be consistent with the light field camera terminologies.

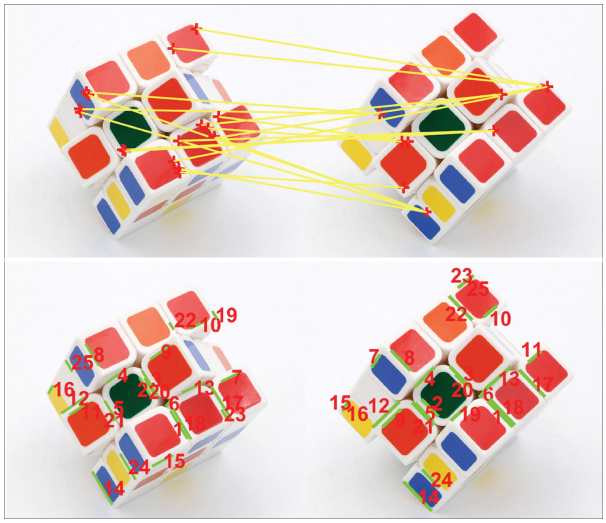

Figure 1. Plenoptic SfM (P-SfM) using point vs. line features. Top: Regular SIFT point feature matching produces insufficient inliers, Bottom: Our technique produces sufficient line matching for reliable P-SfM with 16 inliers out of 25 matches.

A strong assumption there is that the scene contains abundant feature points lying at different depths to ensure robustness in PnP estimation. When only a sparse set of reliable feature points are available or the estimated depths at scene points are less accuracy (which occurs rather often in a light field due to the ultra-small microlenslet baseline), the PnP-based approaches can introduce large errors, as shown in Fig.8.

In contrast, our P-SfM directly exploits how ray geometry transforms under light field pose variations. We first derive the transforms of typical ray manifolds. Specifically we consider three types of ray manifolds: rays passing through a point or point-ray manifold, rays passing through a 3D line or ray-line manifold, and rays lying on a common 3D plane or ray-plane manifold. We show if we are able to match these manifolds across LFs, we can conduct robust pose estimation and bundle adjustment in ray space in terms of ray manifold transforms. To validate our theory and framework, we experiment on synthetic and real data on light fields of different scales: small scale ones acquired using a light field camera and large scale ones by a camera array (Fig.5). We show that our technique exhibits some unique advantages over regular SfM and PnP, and when combined with point fusion, it provides a new state-of-the-art passive 3D scanning technique. For example, we show the line-ray manifold transforms enable high fidelity reconstruction on traditionally challenging scenes such as unfolliaged trees where reliable feature pixel correspondences are difficult to obtain but line constraints are readily available.

2. Related Work

Our work combines recent advances on light field stereo matching with SfM. For the scope of our work, we only discuss most relevant works.

SfM is one of the most studied techniques in computer vision. The very early root of SfM can be traced back to 1980s when Higgins [22] introduced a relative orientation estimation technique. A SfM pipeline includes feature detection and matching [23], camera pose estimation [26], triangulation and bundle adjustment [12]. Modern SfM has shown great success in obtaining extremely realistic models [5, 25]. With immerse computational powers, SfM can now be used to recover very large scale 3D models [13, 33, 35], e.g., by using community photo collections shared on the internet [32].

Reliable feature correspondences is the basis for the success of SfM. Traditional SfM technique uses the local maximums of the scale space (SIFT [23], Harris [11]) for robust matching. However for scenes with textureless surfaces(Walls, Indoor environment), the results are often less satisfactory since only a small number of reliable feature correspondences can be established across views and thereby be reconstructed, as shown in Fig.1. Instead of focusing on point features, we show that ray geometry can enable uses of other types of geometric features, e.g., lines and planes that exist independent of texture [3, 14, 39]. Although they are also used in Manhattan World, to our knowledge, they have not yet been thoroughly explored in ray space. Zhang et al. [42] proposed to estimate the poses of LFs from ray correspondences instead of fully more informative ray space features. A notable exception is the work by Yu et al. [41] that employed the bilinear constraint 3D lines in a LF for stereo matching. We, in contrast, show that such ray manifolds also help with pose estimation and bundle adjustment.

Our work is also enabled by the availability of commodity plenotpic cameras. In contrast to perspective cameras, which capture a centric bundle of rays, a plenoptic camera can record nearly all rays emitting from the scene [1]. More recently, there have been significant advances on light field stereo matching by exploiting specific attributes of the light field [18, 37]. Tao et al. estimated dense depth from defocus and correspondence cues of EPI [34]. Chen et al. explored the ray statistics and used a bilateral consistency metric for reliable stereo matching [6]. Most related to our approach is the work by Johannsen et al. [17] that derived the ray-point structure under the Plucker ray coordinates for image registration. However, their technique first recovers the 3D point position and then use the ray constraints to estimate pose. In other words, it resembles the PnP approach discussed above. In this paper, we apply ray geometry analysis to a broader class of primitives and directly conduct pose estimation in the ray space.

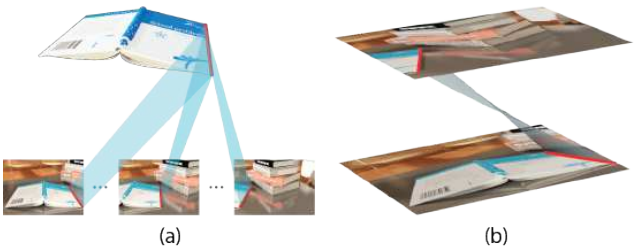

Figure 2. Line-ray manifold. A 3D line projects to 2D lines in individual images (a) but maps to a bilinear surface in a 3D LF (b).

3. Ray-Manifold Transformation

Before proceeding, we first clarify our notions. We use superscripts, such as p^x, p^y and p^z to represent the x, y and z coordinates of a point or a vector. We use the classical two-plane-parameterization (2PP) [20] to describe a ray in 3D space, where each ray is parameterized by its intersection with two parallel planes - $[s,t]$ with the first plane Π_{st} and $[u,v]$ with the second plane Π_{uv}. We further use $[\sigma,\tau,1]$ to represent the direction of a ray, where $\sigma = s-u$ and $\tau = t-v$. To simplify our derivation, we first consider two LFs, namely the reference LF L and the target LF L', we assume L is aligned with the world coordinates. We use $\mathbf{R} \in SO(3)$ and $\mathbf{T} \in \mathbb{R}^3$ to represent the transformation from L to L':

$$\mathbf{R} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \mathbf{T} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{1}$$

3.1. Point-Ray Manifold

We firstly reiterate the point-ray manifold in ray space: given a 3D point $\dot{P}[x,y,z]$ in the light field L and a ray $r=[u,v,s,t]$ passing though $\dot{P}$, we have:

$$\begin{cases} x = u + \lambda(s-u) \\ y = v + \lambda(t-v) \\ z = \lambda \end{cases} \tag{2}$$

This implies all rays passing through $\dot{P}$ lie on a 2D affine linear manifold L_P:

$$L_p : \begin{cases} \mathsf{A}_p u + \mathsf{B}_p s + \mathsf{C}_p &= 0 \\ \mathsf{A}_p v + \mathsf{B}_p t + \mathsf{D}_p &= 0 \end{cases} \tag{3}$$

where $\mathsf{A}_p = 1 - p^z, \mathsf{B}_p = p^z, \mathsf{C}_p = -p^x, \mathsf{D}_p = -p^y$ represent the coefficients of the point-ray manifold.

Now consider the same manifold in the target LF. The manifold should remain as a linear manifold but its coefficients $[\mathsf{A}'_p, \mathsf{B}'_p, \mathsf{C}'_p, \mathsf{D}'_p]$ are transformed from $[\mathsf{A}_p, \mathsf{B}_p, \mathsf{C}_p, \mathsf{D}_p]$ as:

$$\begin{bmatrix} \mathsf{A}'_p \\ \mathsf{B}'_p \\ \mathsf{C}'_p \\ \mathsf{D}'_p \end{bmatrix} = \mathbf{M}_p[\mathbf{R},\mathbf{T}]\mathbf{M}_p^{-1} \begin{bmatrix} \mathsf{A}_p \\ \mathsf{B}_p \\ \mathsf{C}_p \\ \mathsf{D}_p \end{bmatrix} \tag{4}$$

where

$$\mathbf{M}_p = \begin{bmatrix} 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \tag{5}$$

We call this transform point-ray manifold transform. Using this transform, Johannsen et al. [17] derived how to conduct SfM on light fields.

3.2. Line-Ray Manifold

Next, we consider a different type of ray manifold: all rays passing through a common 3D line. [27] [40][41] have previously explored this manifold but not its transform. To briefly reiterate, consider a line r_0 that is not parallel to Π_{uv} and Π_{st}. r_0 hence will intersect with the 2PP and hence can be directly parameterized as if it were a ray $[u_0,v_0,s_0,t_0]$. Any ray $r=[u,v,s,t]$ intersect with r_0 should satisfy that:

$$\begin{cases} u_0 + \lambda_1(s_0-u_0) = u + \lambda_2(s-u) \\ v_0 + \lambda_1(t_0-v_0) = v + \lambda_2(t-v) \\ \lambda_1 = \lambda_2 \end{cases} \tag{6}$$

Eliminating λ_1 and λ_2 we obtain a bi-linear manifold $\mathcal{B}_l(r,r_0)$:

$$\mathcal{B}_l(r,r_0) : \frac{s-s_0}{t-t_0} = \frac{u-u_0}{v-v_0} \tag{7}$$

Alternatively, we can write $\mathcal{B}_l(r,r_0)$ as a conic function F_l:

$$\mathsf{A}_l u + \mathsf{B}_l v + \mathsf{C}_l s + \mathsf{D}_l t + ut - vs + \mathsf{E}_l = 0 \tag{8}$$

where $\mathsf{E}_l = \mathsf{A}_l\mathsf{D}_l - \mathsf{B}_l\mathsf{C}_l$ and

$$\begin{bmatrix} \mathsf{A}_l \\ \mathsf{B}_l \\ \mathsf{C}_l \\ \mathsf{D}_l \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{M}_l} \begin{bmatrix} u_0 \\ v_0 \\ s_0 \\ t_0 \end{bmatrix} \tag{9}$$

A sample line-ray manifold is shown in Fig.2.

Next, we study how the coefficients transform under light field pose variations. Since E_l is directed computed from $\mathsf{A}_l, \mathsf{B}_l, \mathsf{C}_l, \mathsf{D}_l$, we only need to derive their transform to $[\mathsf{A}'_l, \mathsf{B}'_l, \mathsf{C}'_l, \mathsf{D}'_l]$. Similar to the the point-ray manifold

transform, we can derive the line-ray manifold transform as:

$$\begin{bmatrix} \mathsf{A}_l' \\ \mathsf{B}_l' \\ \mathsf{C}_l' \\ \mathsf{D}_l' \end{bmatrix} = \mathbf{M}_l \Gamma(\mathbf{R}, \mathbf{T}, \mathbf{M}_l^{-1} \begin{bmatrix} \mathsf{A}_l \\ \mathsf{B}_l \\ \mathsf{C}_l \\ \mathsf{D}_l \end{bmatrix}) \tag{10}$$

where Γ is the corresponding transform that maps the 3D line $r[\sigma, \tau, u, v]$ parameterized under reference to $r^*[\sigma, \tau, u, v]$ under the target as:

$$\begin{aligned} \sigma^* &= \frac{a_{11}\sigma + a_{12}\tau + a_{13}}{a_{31}\sigma + a_{32}\tau + a_{33}} \\ \tau^* &= \frac{a_{21}\sigma + a_{22}\tau + a_{23}}{a_{31}\sigma + a_{32}\tau + a_{33}} \\ u^* &= a_{11}u + a_{12}v + b_1 - (a_{31}u + a_{32}v + b_3)\sigma^* \\ v^* &= a_{21}u + a_{22}v + b_2 - (a_{31}u + a_{32}v + b_3)\tau^* \end{aligned} \tag{11}$$

The transform above reveals how line-ray manifold transforms between the reference and target manifold $\Gamma(\mathbf{R}, \mathbf{T})$. This is an important transform in plenoptic SfM : for scenes where it is difficult to establish point-ray manifold matching across the light fields, we can still use line-ray manifolds for reliable extrinsic estimation.

3.3. Plane-Ray Manifold

The third class of ray manifold is plane-ray manifold, i.e., all rays lying on a plane. Assume a 3D plane has normal $[n^x, n^y, n^z]$ and is d distance away from the origin has form $\frac{n^x}{n^z}x + \frac{n^y}{n^z}y + z + \frac{d}{n^z} = 0$ (we assume $n_z \neq 0$ so that the plane is not parallel to the 2PP). We use $[A_\pi, B_\pi, 1]$ to represent the normal, where $\mathsf{A}_\pi = \frac{n^x}{n^z}, \mathsf{B}_\pi = \frac{n^y}{n^z}$ and $\mathsf{C}_\pi = \frac{d}{n^z}$. $\mathsf{A}_\pi, \mathsf{B}_\pi, \mathsf{C}_\pi$ correspond to the coefficients of the plane-ray manifold. Any ray $[\sigma, \tau, u, v]$ lying on the plane should satisfy two linear constraints:

$$\begin{cases} \mathsf{A}_\pi u + \mathsf{B}_\pi v + \mathsf{C}_\pi = 0 \\ \mathsf{A}_\pi s + \mathsf{B}_\pi t + \mathsf{C}_\pi + 1 = 0 \end{cases} \tag{12}$$

The first indicates the origin of the ray lies on the plane and second indicates the direction of the ray is perpendicular to the normal.

Consider how this manifold transforms under $\mathbf{R}$ and $\mathbf{T}$ from $[\mathsf{A}_\pi, \mathsf{B}_\pi, \mathsf{C}_\pi]$ of light field L, we have:

$$\lambda \begin{bmatrix} \mathsf{A}_\pi' \\ \mathsf{B}_\pi' \\ 1 \end{bmatrix} = \mathbf{R} \begin{bmatrix} \mathsf{A}_\pi \\ \mathsf{B}_\pi \\ 1 \end{bmatrix}$$

$$\mathsf{C}_\pi' = < (\mathbf{R}\mathsf{C}_\pi \frac{[\mathsf{A}_\pi, \mathsf{B}_\pi, 1]^T}{\|[\mathsf{A}_\pi, \mathsf{B}_\pi, 1]^T\|} + \mathbf{T}), \frac{[\mathsf{A}_\pi', \mathsf{B}_\pi', 1]^T}{\|[\mathsf{A}_\pi', \mathsf{B}_\pi', 1]^T\|} > \tag{13}$$

where λ is a scalar and $< \dot{a}, \dot{b} >$ is the dot product of $\dot{a}$ and $\dot{b}$. The ray-plane transform allows us to derive the planar homography across light fields as follows.

3.4. Light Field Planar Homography

Planar homography matrix between 2D perspective cameras is well-known [12] [24]: given two cameras c and c' where c is aligned to the world coordinates and a plane with unit normal vector $N \in \mathbb{R}^3$ and $d > 0$ corresponding to the distance from the plane to the optical center of c, the point coordinates $\mathbf{X}, \mathbf{X}'$ of same 3D point $\dot{P}$ on the plane under c and c' 's coordinate systems satisfy a homography constraint:

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T} = (\mathbf{R} + \frac{1}{d}\mathbf{T}N^T)\mathbf{X} \tag{14}$$

where $\mathbf{H} = \mathbf{R} + \frac{1}{d}\mathbf{T}N^T$ as the homography matrix. Further, the corresponding two images x, x' of $\dot{P}$ also satisfy:

$$x' \sim \mathbf{H}x \tag{15}$$

where $\sim$ refers to equality up to a scale. If a 3D line on the plane is projected to c as l, and to c' as l', l and l' satisfy:

$$l \sim \mathbf{H}^T l' \tag{16}$$

Now that we can substitute the 2D images with two LFs, and explore the planar homography constraint between LFs.

We assume all light field sample views (subaperture views if a plenoptic camera) are pinhole camera with identical intrinsic parameters. The transform between views within the same LF is merely a translation without rotation. We first model the homography constraint between the center views S_0 and S_0' in respective LFs. The transform clearly follows Eq.15 and Eq.16. Let S_i' be any view within L', and the translation between $S_0{}'$ and $S_i{}'$ is $t_{S_i'}$. Now assume a 3D point $\dot{P}$ on the plane whose coordinates are $\mathbf{X}_{S_0}$, $\mathbf{X}_{S_0'}$, and $\mathbf{X}_{S_i'}$ with respect to S_0, S_0', and $S_i{}'$. For $\mathbf{X}_{S_i'}$ and $\mathbf{X}_{S_0}$, they satisfy:

$$\mathbf{X}_{S_i'} = \mathbf{R}\mathbf{X}_{S_0} + \mathbf{T} + t_{S_i'} = \underbrace{\left[\mathbf{R} + \frac{1}{d}(\mathbf{T} + t_{S_i'})N^T\right]}_{\mathbf{H_i}} \mathbf{X}_{S_0} \tag{17}$$

where N is the unit normal of the plane and d is the distance from the optical center of S_0 to the plane within L. Recall that d and N can be directly mapped to the ray-plane manifold $[\mathsf{A}_\pi, \mathsf{B}_\pi, \mathsf{C}_\pi]$ described in Sec.3.3. We can write the transform function above as:

$$\mathbf{X}_{S_i'} = \mathbf{R}\mathbf{X}_{S_0} + \mathbf{T} + t_{S_i'} = \underbrace{\left(\mathbf{R} + \frac{1}{\mathsf{C}_\pi}(\mathbf{T} + t_{S_i'}) \begin{bmatrix} \mathsf{A}_\pi \\ \mathsf{B}_\pi \\ 1 \end{bmatrix}^T\right)}_{\mathbf{H_i}} \mathbf{X}_{S_0} \tag{18}$$

Notice that we can obtain $t_{S_i'}$ via calibration. Therefore, any matched lines l_{S_0} in S_0 and $l_{S_i'}$ in S_i' that corresponds to a 3D line on the plane should satisfy:

$$l_{S_0} \sim \mathbf{H}_i^T l_{S_i'} \tag{19}$$

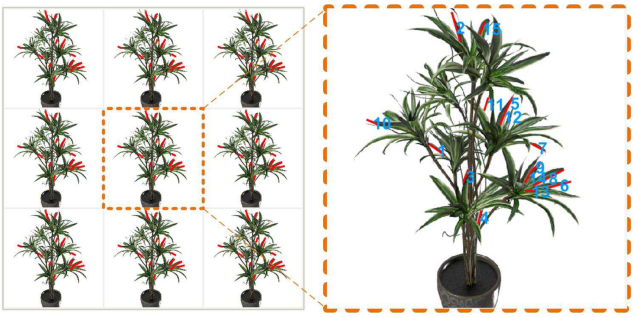

Figure 3. Our line matching results on a 3×3 LF.

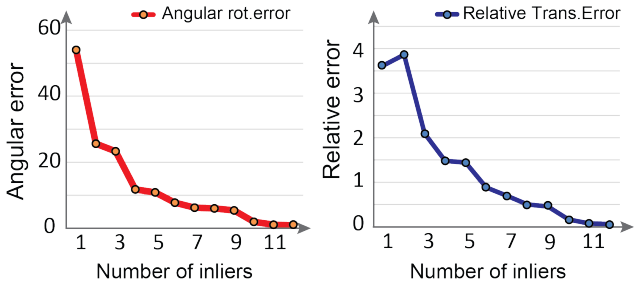

Figure 4. The charts shows error on the estimated rotation (left) and translation (right) vs. the number of inliers.

We call this constraint light field planar homography and we show in the following sections how to use them for P-SfM.

4. Validation and Experiments

To conduct P-SfM, we combine point-ray manifold transform, line-ray manifold transform and light field planar homography. Specifically, to utilize the line-ray manifold transform, we first need to solve for the coefficients $[\mathsf{A}_l, \mathsf{B}_l, \mathsf{C}_l, \mathsf{D}_l]$ for a line in respective LFs.

4.1. Ray Manifold Parameterization within Light Field

The Lytro light field camera uses a regularly arranged microlenslet array and can record the scene as a grid subaperture image array in one single shot. Consequently, the disparities of a 3D point between vertical or horizontal pair of subaperture images are identical.

To recover the line-ray manifold of a 3D line in a LF, we need to first conduct 2D line segment maching across all subaperture images within the LF. For simplicity, we use a row of subaperture images $\{S_1, S_2, ..., S_n\}$ as an example (n is the number of subaperture images in a row) although it can easily extended to 2D arrays as for all our experiments.

We start with the LSD algorithm [10] to detect all available line segments in each subaperture image. Next, we compute the disparity between two line segments in adjacent subaperture images. We locate two scanlines passing though endpoints of the line segment in the first image. We further extend the line segment in the second image to insect with the two scanlines. The pixel difference between the two intersection points on the same scanline should correspond to the disparity. Therefore, we can obtain two disparities d^1, d^2 for each pair line segments, one for each endpoint.

Next, we use a predetermined disparity range to conduct initial line matching between pairwise adjacent subaperture images (S_1, S_2), (S_2, S_3), etc. We gather all potential matches as tracks across the row subaperture images, such as one track $[l^1_{t_i}, l^2_{t_i}, l^3_{t_i}, ..., l^n_{t_i}]$ where $l^j_{t_i}, j = 1...n$ refer to the matched line number of subaperture image S_j in track t_i. We use a cost function to determine whether the reliability of a track:

$$E_m = \sum_{i=1}^{n-1} (||d^1(i,i+1) - d^1_h|| + ||d^2(i,i+1) - d^2_h||) \quad (20)$$

where d^1_h, d^2_h are two hypothesized disparities calculated from a pair line segments in a track, and $d^1(i,i+1), d^2(i,i+1)$ are the computed disparities from rest pairs. If we find the right track matches, E_m should be very small due to LF regular sampling. We hence choose tracks that yield to minimum cost as our line segment matching results. We further apply the same matching process along the vertical direction. Finally, we merge the line matching results from rows and columns to obtain final line matching across all subaperture images. Fig.3 shows our matching result in a 3×3 LF. Our technique is able to effectively remove the outliers.

Recall that each track is a group of 2D projected lines that correspond to a 3D line l. Each endpoint $[x, y]$ of the 2D line maps to a ray $[u, v, s, t]$ that passes though l using the calibration intrinsic parameters as $K^{-1}[x, y, 1]^T$. Therefore, we can use Eq.8 in Sec.3.2 to compute the line-ray manifold $[\mathsf{A}_l, \mathsf{B}_l, \mathsf{C}_l, \mathsf{D}_l]$ that minimizes F_l.

The similar process can be applied to compute the line-ray manifolds on the target LF and directly seek out to find the optimize line-ray transform. To further improve robustness, we use the SMSLD algorithm [36] to match the 2D lines on the center subaperture images of LFs. The matched 2D lines pairs help match line-ray manifolds. Finally, we combine the transform error measures from line-ray manifold and corresponding 2D lines (using light field homography) computed from Eq.10, 13,19 to solve for the optimal $\mathbf{R}, \mathbf{T}$.

4.2. Light Field Bundle Adjustment

Once we estimate the pose of the light fields, we further employ a LF bundle adjustment step to refine LF pose estimation and 3D line localization. Similar to classical bundle

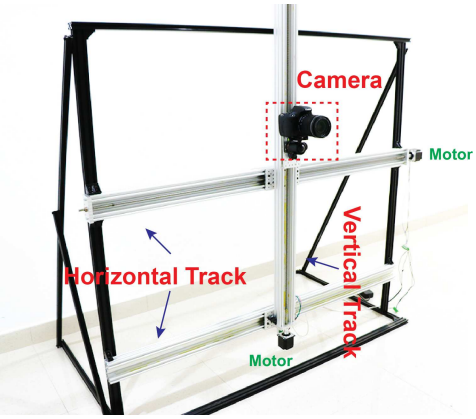

Figure 5. Our camera array that can capture 7×7 large scale LFs. We mount a pre-calibrated Canon 760D camera on an electrically controlled rig.

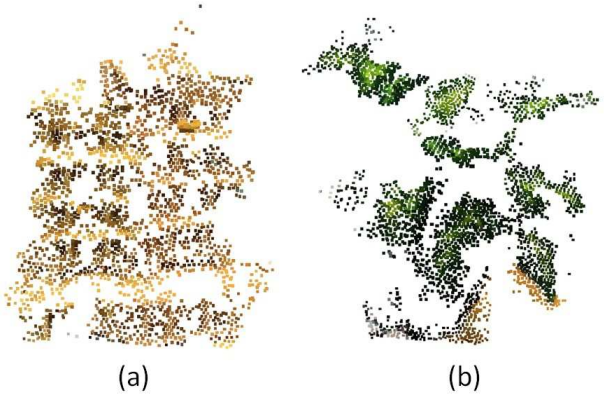

Figure 6. VisualSFM results from our light field data of real scenes. The results show that VisualSFM can't generate dense point clouds from light field inputs.

adjustment that minimizes the 3D point reprojection error, we set out to use reprojection error of estimated 3D lines. We assume each light field has s subaperture images and we obtain m 3D lines seen in n light fields. Let $r_{l,i,k}$ be the projection of kth line in the ith subaperture $S_{l,i}$ of lth light field, the objective function for bundle adjustment is:

$$E_{ba} = \sum_{l=1}^{n} \sum_{i=1}^{s} \sum_{k=1}^{m} w_{l,i,k} d(P(S_{l,i}, R_k), r_{l,i,k})^2 \qquad (21)$$

where $w_{l,i,k}$ is the visibility of a 3D line R_k in $S_{l,i}$. Function d measures the Euclidean distance. $P(S_{l,i}, R_k)$ denotes the projection from the 3D ray R_k to $S_{l,i}$. We use Levenberg-Marquardt to find the optimal solution.

Table.1 compares our P-SfM results with Iterative Closest Point (ICP) [43], PnP [19], point-ray manifold method (LF-SfM) [17] on simulate data. Our algorithm is able to handle highly challenging scenes where point-ray manifold transforms are difficult to obtain. Fig.4 shows how our estimation error changes with respect to the number of inliers. With only a small number of inliers (10), our technique can already produce reliable pose estimations.

We further generate two synthetic light fields of a book scene and a plant scene using 3ds Max. The book scene contains piles of books that lack point features but exhibit strong line features. The plant scene exhibit heavy occlusions and similar features that are difficult to separate using point-based approaches. Fig.7 shows the recovered point clouds using our technique.

4.3. Real Scenes

For real scenes, we validate our framework on two setups, a small scale LF captured by a Lytro light field camera and a large scale one captured by a light field camera array. For the camera array, we designed an electrically controlled rig so that images can be captured at a regularly sampled grid at a high accuracy.

We calibrate the Lytro camera's intrinsic using the geometric calibration [4] toolkit. After our calibration, each pixel $[i, j]$ in a subaperture image $[k, l]$ can be mapped to a ray $[u, v, s, t]$ using 2PP parameterization. The resolution of each subaperture image is 552×383 and we obtain a 5×5 LF extracted from the Lytro toolkit. We select three highly complex scenes to test our P-SfM, a tower and two flower scenes with heavy occlusions. We use the calibrated Lytro to capture multiple LF images facing towards the target object.

Then We use our P-SfM to estimate LF poses and apply the focal stack symmetry based depth algorithm [21] to generate depth maps from respective LFs. Finally, we fuse the results. We compare our method with the E-PnP algorithm, Colmap [31] and commercial SfM software RealityCapture [28]. Fig.8 shows that our P-SfM algorithm is able to handle very complex scenes where state-of-the-art solutions fail.

We further validate our P-SfM method on a large scale scene. We mount a pre-calibrated Canon 760D camera on an electrically controlled rig. Then we use this device to capture the LFs of a room scene consists of layers of chairs. We capture 3 LFs at different positions and then use our P-SfM to estimate their poses. We use the computed poses to register the 3 LFs. Fig.9 shows that our P-SfM method is still robust for large scale LFs.

5. Conclusions and Future Work

We have presented a new P-SfM framework for multi-view light field reconstruction. Our approach is based on a new ray manifold transform theory that studies how ray manifolds of points, lines, and planes transform under pose variations. We have further developed robust algorithms

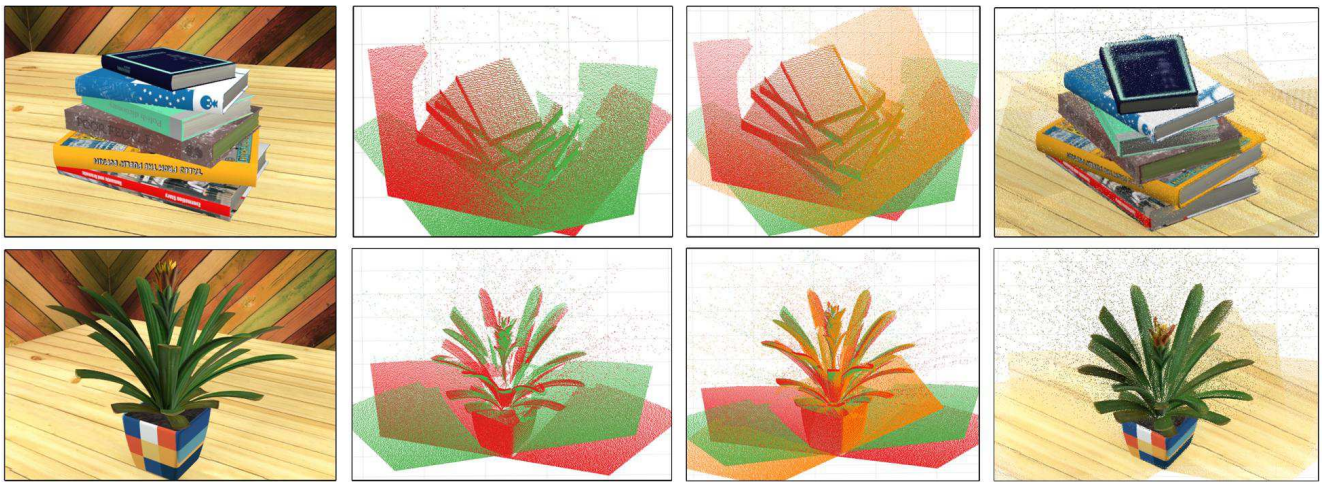

Figure 7. Our P-SfM results on synthetic 5×5 LFs rendered using 3dx Max. From left to right: the synthetic scene, our depth fusion results using 2, 3, and 5 LFs.

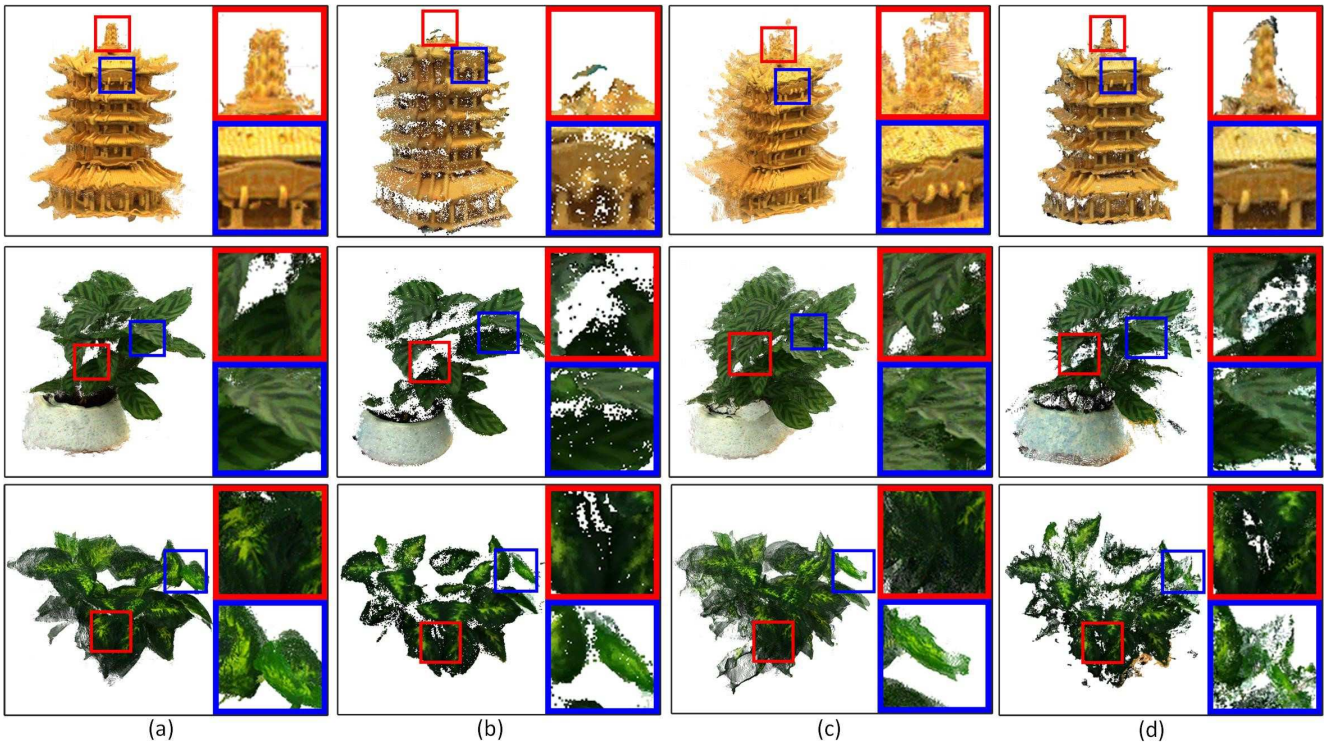

Figure 8. Comparison on real scenes captured by a Lytro camera. (a) shows our reconstruction results from 5 LFs. (b) shows the results of the commercial software RealityCapture. (c) shows the EPnP results. (d) shows the Colmap results.

Figure 9. Our P-SfM results on large scale scenes. We captured LFs using a camera array at 3 different positions, we estimate poses of the LFs, and show the fused stereo results. From left to right: single point cloud, fused two point clouds and fused three point clouds.

	GT	R:14, T:33mm				R:19, T:60mm				R:22, T:87mm			
	Noise	0.2	0.5	1.0	2.0	0.2	0.5	1.0	2.0	0.2	0.5	1.0	2.0
Rot. Err [deg]	ICP	0.809	1.164	0.693	4.694	0.628	1.275	2.808	4.499	0.994	2.546	1.486	5.797
	PnP	0.333	1.208	1.826	1.760	1.540	2.353	3.348	1.242	1.296	0.988	2.378	2.769
	LF-SfM	0.372	0.347	0.644	**0.174**	0.861	1.019	**1.512**	0.450	1.053	0.648	0.439	1.258
	Ours	**0.326**	**0.314**	**0.520**	0.808	**0.126**	**0.971**	1.094	**0.428**	**0.189**	**0.214**	**0.675**	**0.614**
Tran. Err	ICP	0.098	0.140	1.232	0.663	0.028	**0.065**	0.143	0.204	0.052	0.095	0.062	0.216
	PnP	0.043	0.141	0.198	0.170	0.118	0.125	0.210	0.048	0.062	0.040	0.093	0.102
	LF-SfM	0.047	0.049	0.096	0.087	0.077	0.083	**0.114**	**0.041**	0.057	0.026	**0.017**	0.068
	Ours	**0.023**	**0.015**	**0.076**	**0.034**	**0.021**	0.096	0.037	0.166	**0.006**	**0.018**	0.006	**0.025**
	GT	R:26, T:112mm				R:31, T:146mm				R:37, T:163mm			
	Noise	0.2	0.5	1.0	2.0	0.2	0.5	1.0	2.0	0.2	0.5	1.0	2.0
Rot. Err [deg]	ICP	0.368	5.231	1.232	20.758	0.521	1.633	1.700	17.323	2.782	3.042	6.521	27.965
	PnP	0.904	1.467	3.779	7.856	3.539	1.968	0.898	6.504	2.307	3.926	2.672	9.692
	LF-SfM	0.232	0.668	0.916	3.512	1.947	1.006	0.705	2.086	1.515	1.728	6.427	1.373
	Ours	**0.131**	**0.307**	**0.103**	**1.268**	**0.090**	**0.438**	**0.347**	**0.771**	**0.352**	**0.624**	**0.298**	**0.763**
Tran. Err	ICP	0.013	0.157	0.029	0.370	0.015	0.037	0.047	0.609	0.050	0.085	0.117	0.160
	PnP	0.046	0.087	0.137	0.248	0.108	0.049	**0.012**	0.334	0.046	0.198	0.087	0.220
	LF-SfM	0.020	0.051	0.032	0.135	0.052	**0.022**	0.015	0.030	0.032	0.077	0.120	0.037
	Ours	**0.011**	**0.006**	**0.009**	**0.072**	**0.008**	0.025	0.014	**0.024**	**0.007**	**0.054**	**0.011**	**0.020**

Table 1. Comparisons on accuracy using our technique vs. the state-of-the-art methods. Rotation errors are computed using the difference between the measured and the ground truth angle (in degrees) and translation errors are measured as relative distance.

that use the transform for recovering LF poses as well as a companion LF bundle adjustment step to refine the estimation. Experiments on small and large scale LFs show that our technique can handle very complex scenes where reliable point feature correspondences are difficult to obtain but line features are readily available.

Although our work is largely theoretical, we have demonstrated practical uses on specific types of scenes. Clearly, it would be ideal to automatically decide when to use the point-ray manifolds and when to use the line-ray manifolds, a problem closely related to scene understanding. Therefore, we plan to investigate integrating machine learning approaches with our framework to improve the accuracy and reliability. Further, by estimating LF poses, we can potentially fuse multiple LFs into a bigger LF. This can benefit applications such as virtual navigation of real environments, e.g., on a VR headset. In the future, we plan to integrate our work with LF fusion techniques for achieving this goal.

References

[1] E. H. Adelson and J. R. Bergen. The plenoptic function and the elements of early vision. In *Computational Models of Visual Processing*, pages 3–20. MIT Press, 1991.

[2] S. Agarwal, Y. Furukawa, N. Snavely, B. Curless, S. M. Seitz, and R. Szeliski. Reconstructing rome. *Computer*, 43(6):0040–47, 2010.

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

[4] Y. Bok, H.-G. Jeon, and I. S. Kweon. Geometric calibration of micro-lens-based light-field cameras using line features. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014.

[5] E. Bylow, J. Sturm, C. Kerl, F. Kahl, and D. Cremers. Real-time camera tracking and 3d reconstruction using signed distance functions. In *Robotics: Science and Systems*, 2013.

[6] C. Chen, H. Lin, Z. Yu, S. Bing Kang, and J. Yu. Light field stereo matching using bilateral statistics of surface cameras. In *CVPR*, 2014.

[7] A. Cohen, T. Sattler, and M. Pollefeys. Merging the unmatchable: Stitching visually disconnected sfm models. In *ICCV*, 2015.

[8] A. Cohen, J. L. Schönberger, P. Speciale, T. Sattler, J.-M. Frahm, and M. Pollefeys. *Indoor-Outdoor 3D Reconstruction Alignment*, pages 285–300. Springer International Publishing, Cham, 2016.

[9] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher. SfM with MRFs: Discrete-continuous optimization for large-scale structure from motion. *IEEE TPAMI*, 2013.

[10] R. Grompone von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. LSD: a Line Segment Detector. *Image Processing On Line*, 2:35–55, 2012.

[11] C. Harris and M. Stephens. A combined corner and edge detector. In *In Proc. of Fourth Alvey Vision Conference*, pages 147–151, 1988.

[12] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[13] J. Heinly, J. L. Schönberger, E. Dunn, and J.-M. Frahm. Reconstructing the World* in Six Days *(As Captured by the Yahoo 100 Million Image Dataset). In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[14] M. Hofer, M. Donoser, and H. Bischof. Semi-global 3d line modeling for incremental structure-from-motion. In *British Machine Vision Conference*, number 25, 2014.

[15] S. Ikehata, H. Yang, and Y. Furukawa. Structured indoor modeling. In *ICCV*, pages 1323–1331. IEEE Computer Society, 2015.

[16] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. ACM Symposium on User Interface Software and Technology, October 2011.

[17] O. Johannsen, A. Sulc, and B. Goldluecke. On linear structure from motion for light field cameras. In *ICCV*, 2015.

[18] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross. Scene reconstruction from high spatio-angular resolution light fields. *ACM Trans. Graph.*, 32(4):73:1–73:12, July 2013.

[19] V. Lepetit, F.Moreno-Noguer, and P.Fua. Epnp: An accurate o(n) solution to the pnp problem. *International Journal Computer Vision*, 81(2), 2009.

[20] M. Levoy and P. Hanrahan. Light field rendering. SIGGRAPH '96. ACM, 1996.

[21] H. Lin, C. Chen, S. B. Kang, and J. Yu. Depth recovery from light field using focal stack symmetry. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 3451–3459, Dec 2015.

[22] H. C. Longuet-Higgins. Readings in computer vision: Issues, problems, principles, and paradigms. chapter A Computer Algorithm for Reconstructing a Scene from Two Projections, pages 61–62. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

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

[24] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003.

[25] D. Martinec and T. Pajdla. 3d reconstruction by fitting low-rank matrices with missing data. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 198–205 vol. 1, June 2005.

[26] D. Martinec and T. Pajdla. Robust rotation and translation estimation in multiview reconstruction. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[27] J. Ponce. What is a camera? In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1526–1533, June 2009.

[28] RealityCapture. Realitycapture. `https://www.capturingreality.com/`.

[29] J. L. Schönberger, F. Radenović, O. Chum, and J.-M. Frahm. From single image query to detailed 3d reconstruction. In *CVPR*, 2015.

[30] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys. *Pixelwise View Selection for Unstructured Multi-View Stereo*, pages 501–518. Springer International Publishing, Cham, 2016.

[31] J. L. Schnberger and J. M. Frahm. Structure-from-motion revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, June 2016.

[32] N. Snavely, I. Simon, M. Goesele, R. Szeliski, and S. M.Seitz. Scene reconstruction and visualization from community photo collections. *Proceedings of the IEEE*, 2010.

[33] F. Steinbruecker, C. Kerl, J. Sturm, and D. Cremers. Large-scale multi-resolution surface reconstruction from rgb-d sequences. In *IEEE International Conference on Computer Vision (ICCV)*, Sydney, Australia, 2013.

[34] M. Tao, S. Hadap, J. Malik, and R. Ramamoorthi. Depth from combining defocus and correspondence using light-field cameras. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2013.

[35] V. Usenko, J. Engel, J. Stueckler, and D. Cremers. Reconstructing street-scenes in real-time from a driving car. In *Proc. of the Int. Conference on 3D Vision (3DV)*, Oct. 2015.

[36] B. Verhagen, R. Timofte, and L. V. Gool. Scale-invariant line descriptors for wide baseline matching. In *IEEE Winter Conference on Applications of Computer Vision*, pages 493–500, March 2014.

[37] T.-C. Wang, A. Efros, and R. Ramamoorthi. Occlusion-aware depth estimation using light-field cameras. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[38] J. Xiao and Y. Furukawa. Reconstructing the world's museums. *International Journal of Computer Vision*, 110(3):243–258, 2014.

[39] W. Yang, Y. Ji, J. Ye, S. S. Young, and J. Yu. Coplanar common points in non-centric cameras. In *European Conference on Computer Vision*, pages 220–233. Springer, 2014.

[40] J. Yu and L. McMillan. General linear cameras. In *ECCV*, 2004.

[41] Z. Yu, X. Guo, H. Ling, A. Lumsdaine, and J. Yu. Line assisted light field triangulation and stereo matching. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2792–2799. IEEE, 2013.

[42] Y. Zhang, Z. Li, W. Yang, P. Yu, H. Lin, and J. Yu. The light field 3d scanner. In *Computational Photography (ICCP), 2017 IEEE International Conference on*, pages 1–9. IEEE, 2017.

[43] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *Int. J. Comput. Vision*, 13(2):119–152, Oct. 1994.