

Robust Hough Transform Based 3D Reconstruction from Circular Light Fields

Alessandro Vianello, Jens Ackermann

Robert Bosch GmbH

Robert Bosch Campus 1, Renningen, Germany

alessandro.vianello3@gmail.com

Maximilian Diebold, Bernd Jähne

Heidelberg Collaboratory for Image Processing

Berliner Str. 43, Heidelberg, Germany

maximilian.diebold@iwr.uni-heidelberg.de

Abstract

Light-field imaging is based on images taken on a regular grid. Thus, high-quality 3D reconstructions are obtainable by analyzing orientations in epipolar plane images (EPIs). Unfortunately, such data only allows to evaluate one side of the object. Moreover, a constant intensity along each orientation is mandatory for most of the approaches. This paper presents a novel method which allows to reconstruct depth information from data acquired with a circular camera motion, termed circular light fields. With this approach it is possible to determine the full 360° view of target objects. Additionally, circular light fields allow retrieving depth from datasets acquired with telecentric lenses, which is not possible with linear light fields. The proposed method finds trajectories of 3D points in the EPIs by means of a modified Hough transform. For this purpose, binary EPI-edge images are used, which not only allow to obtain reliable depth information, but also overcome the limitation of constant intensity along trajectories. Experimental results on synthetic and real datasets demonstrate the quality of the proposed algorithm.

1. Introduction

Three dimensional geometry reconstruction is one of the most important tasks in computer vision and image processing. Depth data plays a crucial role in industrial applications (e.g., automatic optical inspection), the game and movie industry, as well as common consumer products. Active systems, such as structured light, laser scanners, or time-of-flight cameras are often costly and/or time consuming. Differently, passive systems like *multi-view stereo* [21, 10] or *structure from motion* are more attractive, considering the simple hardware required and the possibility to achieve high-quality 3D reconstructions.

Passive systems based on *light fields* have been widely developed. A light field is a densely sampled image sequence, where an object is acquired from different views.



Figure 1. The proposed algorithm processes data generated from a circular camera motion, retrieving the depth from sinusoidal trajectories of 3D points in the EPIs. The resulting depth maps can be used to generate a point cloud and a mesh of the target scene.

When all images are stacked on top of each other, forming an *image volume*, one slice through this volume is called epipolar plane image. EPIs give information about the motion parallax of the image points, which are moving with specific trajectories, depending on the camera motion, see Figure 1 for the circular case. In contrast to light fields, classic multi-view stereo algorithms do not make use of the redundancy contained in a densely sampled image sequence. In fact, these algorithms often have a view selection [11] which leads to discarding images due to the small baseline. Furthermore, *optical flow* algorithms normally use two or just few images to compute the depth [15, 3], hence they are less robust than light fields. One of the most popular types of light fields are the so called *linear light fields*, which are a collection of images captured along a linear path. With this type of data, scene points trace straight lines on the EPIs, whose slopes are inversely proportional to the distance of the points. The main disadvantage of linear light fields is that they are restricted to linear camera movements. In this way only one side of the scene can be reconstructed. To have the complete 3D shape, the target object has to be recorded from four different sides, and then the results have to be merged. This constraint makes the acquisition procedure long and tedious. Moreover, most of the light field algorithms strongly rely on the Lambertian hypothesis, which means that an EPI-line should have constant intensity.

Linear light field algorithms are generally developed for data acquired with standard *perspective lenses*. However,

for certain applications, *e.g.*, precise measurement tasks in optical inspection, *telecentric lenses* are better suited. This particular type of lens allows to obtain an *orthographic projection*. Therefore, two identical objects will look the same even if one is closer to the camera than the other. Thus, a linear light field acquired with a telecentric lens would lead to EPIs where all the lines have the same slope, making it impossible retrieving any depth information.

To overcome all these issues, we propose a new approach to extract 3D information from *circular light fields*. A circular light field acquires the scene by rotating the object in front of the camera (or vice versa). In this way it is possible to reconstruct the full 360° shape with just one continuous acquisition. With this setup, every captured scene point corresponds to a curved trajectory in the EPI. Variations of the depth lead to sine shaped curves with different amplitudes and phase offsets, as will be explained in Section 3. It will be shown that circular light fields can be used to retrieve depth information even from datasets acquired with a telecentric lens. The proposed algorithm uses a coarse EPI-slope map, generated with the local *structure tensor*, together with a binary edge map of the EPI, to extract trajectories by using an adapted version of the *Hough transform*. The result, is a set of highly accurate depth maps of the target scene from all sides. Since the Hough transform uses binarized EPIs to retrieve trajectories, it is possible to get rid of the Lambertian hypothesis and process datasets with strong intensity changes along the EPI-curves. In fact, even if a trajectory is only partially visible or its intensity saturates because of a specular reflection, the Hough transform can still recover the full curve. In order to apply circular light fields to both perspective and telecentric lenses, two slightly different versions of the algorithm are proposed.

Our method is based on [22, 23], where we introduced a new approach for linear light fields which retrieves EPI-lines with a combined structure tensor and Hough transform.

2. Related Work

Although the term light field was already introduced in 1936 by Gershun [12] as the multi-dimensional function (later called *plenoptic function* [1]) describing all the information available from the scene’s reflected light, light fields were introduced into computer graphics only in 1996 by both Gortler *et al.* [14] and Levoy *et al.* [19] for an image base rendering application. In order to acquire light fields, the plenoptic function is simplified to a 4D subspace, termed the *Lumigraph* [14]. With this representation, the ray space of the light field can be parametrized by the two points of intersection of a ray with two parallel planes, so that the light field can be considered as a collection of views, where the focal points of the cameras lie in a 2D plane. The first attempt to extract depth from EPIs was the work of

Bolles [2], where salient lines were derived by finding zero crossings and then by merging collinear segments. Criminisi *et al.* [5] proposed to extract EPI-regions by using photo-consistency. More recently, Wanner [24, 25], and later on Diebold [7], used the structure tensor to estimate the local slope of each pixel in the EPI, obtaining a coarse depth map which is then refined by means of a global optimization. Unfortunately, structure tensor methods provide only a local evaluation of EPIs’ orientations. This can be a problem especially in noisy datasets, where using all the available information, *i.e.*, the full EPI-line, helps to increase the quality of the final reconstruction. Additionally, global optimization tends to smooth depth discontinuities by averaging between foreground and background disparities.

An approach which takes advantage of the whole trajectory was proposed by Feldmann *et al.* [8, 9], who used the intensity constancy as a measure to determine valid paths. In their work the 3D space is discretized into voxels and then, for each hypothetical 3D point, the algorithm seeks in the image volume if the corresponding path exists. This method was also adapted to the case of a camera which rotates around the target scene. Crispell *et al.* [6], and later on Lanman *et al.* [18], retrieve EPI-trajectories in circular light fields only on depth discontinuities instead of texture edges. They use a contour tracking approach which is not robust to specularities and cannot deal with EPIs having many close trajectories. Similarly to [8], Kim *et al.* [17] proposed a method for linear light fields which computes depth estimates around the scene’s highly textured areas by testing all the possible disparity hypotheses and choosing the one that leads to the best color constancy along the EPI-line. This approach was later applied to circular camera motion by Yücer *et al.* [27], to segment the target object and compute the visual hull. Then, they extended their approach to estimate the depth also in concave areas by analyzing the local gradient directions in the image volume [26]. Unfortunately, the main limitation of these methods is the restriction to Lambertian surfaces.

3. Circular Light Fields

EPI analysis was extended to the case of circular camera movements by Feldmann *et al.* [8]. The acquisition setup is composed of a fixed camera and an object rotating around a point M aligned with the camera’s optical center C , as shown in the left side of Figure 2. In this section, the image formation of circular light fields is explained, for both orthographic and perspective camera projection models.

3.1. Orthographic Camera

The simplest camera projection is the orthographic projection, which can be obtained through a telecentric lens. Let $\mathbf{P} = [X, Y, Z]^T$ be an arbitrary 3D point, assuming a

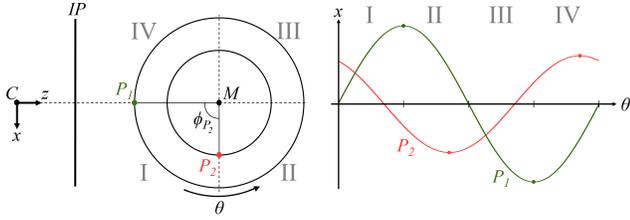


Figure 2. Orthographic camera: left, xz -plane showing the projection in the image plane IP of the points P_1 and P_2 rotating with a phase θ around M ; right, trajectories of the two points in the EPI ($x\theta$ -plane), the dots indicate the points of maximum amplitude.

sensor with square pixels, its projection into image coordinates (x, y) is expressed as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} m & 0 & 0 & x_c \\ 0 & m & 0 & y_c \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \quad (1)$$

where m is the telecentric lens *magnification* divided by the pixel pitch σ , and (x_c, y_c) denotes the sensor's principal point.

The left side of Figure 2 shows the xz image plane of an orthographic camera and two points P_1 and P_2 rotating with two different radii R_{P_1} and R_{P_2} around the rotation center M , with a phase θ . The points have, respectively, a phase offset ϕ_{P_1} and ϕ_{P_2} . From Figure 2 it is possible to define the X, Z components of the generic point \mathbf{P} in polar coordinates as

$$X = R \cdot \sin(\theta + \phi) \quad (2a)$$

$$Z = R_M - R \cos(\theta + \phi), \quad (2b)$$

where R is the point's radius, $\theta \in [0, 2\pi]$ is the rotation's phase, and R_M is the distance between the center of rotation M and the camera optical center C . On the right side of Figure 2 the corresponding trajectories of the points P_1 and P_2 projected onto the image plane are shown. The trajectory of a point \mathbf{P} can be derived from Equations 1 and 2 as

$$x = A \cdot \sin(\theta + \phi) + x_c \quad (3a)$$

$$y = H + y_c, \quad (3b)$$

where $A = m \cdot R$ is the trajectory's amplitude in pixel, and $H = m \cdot Y$ is the point's height in pixel. It is important to note that y only depends on the height Y of the 3D point (due to the depth independence of the orthographic projection). Consequently, in the orthographic case, the full trajectory of a rotating 3D point is imaged in one EPI. An example of such a circular light field is shown in Figure 3.

From Equations 3 it can be seen that any scene point is simply defined by its radius R and its phase offset ϕ . With

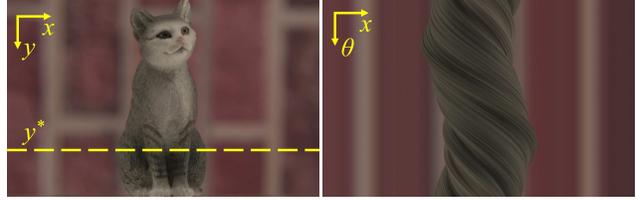


Figure 3. Example of circular light field acquired with a telecentric lens: left, first image; right, EPI corresponding to the coordinate y^* highlighted by the dashed line.

this parametrization, Feldmann *et al.* [8] defined two occlusion rules:

1. All the points in the quadrants I and IV will occlude those in the quadrants II and III if their projection rays are equal;
2. In the quadrants I and IV all the points with a larger radius will occlude those with a smaller one if their projection rays are equal. Vice versa, for the quadrants II and III, points with a smaller radius will occlude those with a larger one.

Points moving in the quadrants I and IV correspond to curves with positive slope ($\frac{\delta x}{\delta \theta} > 0$), whereas points moving in the quadrants II and III lead to curves with negative slope ($\frac{\delta x}{\delta \theta} < 0$).

3.2. Perspective Camera

With a standard lens, a *perspective projection* is obtained. The pinhole camera model defines the projection of the 3D point $\mathbf{P} = [X, Y, Z]^T$ into image coordinates (x, y) as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & x_c & 0 \\ 0 & f & y_c & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \quad (4)$$

In this case, due to the projection's depth dependency, the sinusoidal trajectories of Section 3.1 are slightly distorted. If the generic 3D point \mathbf{P} is again considered, its trajectory can be derived from Equations 2 and 4 as

$$x = f \cdot \frac{R \sin(\theta + \phi)}{R_M - R \cos(\theta + \phi)} + x_c \quad (5a)$$

$$y = f \cdot \frac{Y}{R_M - R \cos(\theta + \phi)} + y_c, \quad (5b)$$

where f is the focal length. Figure 4 shows the trajectories' x components for different rotation radii, and the y components for different rotation radii and Y coordinates, as a function of the rotation phase θ . For the perspective case, the trajectory of a point does not completely lie in the xz -plane, as it was in the orthographic projection, but is also moving in the y -direction during the rotation.

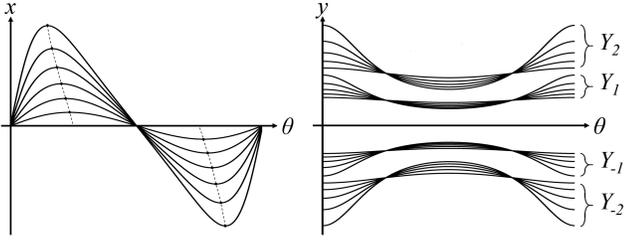


Figure 4. Perspective camera: left, trajectories with increasing radius in the $x\theta$ -plane, the dots indicate the points of maximum amplitude; right, trajectories with increasing radius and different height Y in the $y\theta$ -plane.

4. Hough Transform for Orthographic Camera

The Hough transform is an elegant method for estimating parametrized line segments (*e.g.*, lines, circles, parabolas, and ellipses) in an image. This approach has the advantage of being robust against noise and unaffected by occlusions or gaps in the trajectory. In practice, Hough transform finds curves in a binary image, which we obtain with a Canny edge detector. For a circular light field, the orthographic projection case of Equations 3 can be correctly solved by analyzing a single EPI. On the contrary, the perspective projection case of Equations 5 can have only an approximated solution by analyzing EPI like slices, due to the fact that a point changes its y -coordinate during the rotation. In the following, we describe the general Hough transform algorithm for orthographic circular light fields. The perspective circular approximation will be explained in the next section.

4.1. Hough Space Generation

With the parametrization of Equation 3a each EPI-trajectory t (*i.e.*, a 3D point) can be associated with a pair (A_t, ϕ_t) . The (A, ϕ) plane is termed *Hough space* (H). The occlusion ordering rule 1 imposes that two Hough spaces have to be computed in order to identify the trajectories: a Hough space H_1 for trajectories in the quadrants (I, IV), and a Hough space H_2 for the ones in quadrants (II, III). The two Hough spaces are discretized into *cells* and initially populated with zeros. This discretization depends on the chosen sensor and the acquired images. Specifically, a sensor with resolution $N_x \times N_y$ pixel yields to amplitudes $A \in [0, 1, \dots, N_x/2]$. On the other hand, the phase offsets are determined by the number of images N , and are $\phi \in [0, 2\pi/N, \dots, 2\pi]$. In fact, each image corresponds to a rotation angle of $2\pi/N$, which defines the phase resolution. Now that the Hough spaces are defined, each non-zero point i of the EPI binary image has to *vote*. This means it increments by 1 the cell having coordinates (A_i, ϕ_i) in the correct Hough space. In order to determine if an edge point is related to H_1 or H_2 , *i.e.*, the trajectory point is in the (I, IV) or (II, III) quadrants, the local slope of the EPI-image

is computed with the structure tensor: points with positive slope belong to the quadrants (I, IV), whereas points with negative slope belong to the quadrants (II, III).

In the voting procedure, for each edge point in the EPI binary image, its coordinates (x_i, θ_i) identify a rotation phase θ_i and a point coordinate x_i , *i.e.*, the amplitude of the possible trajectory at θ_i . From these two values it is possible to invert Equation 3a and derive the trajectory's phase offset ϕ with

$$\phi = \begin{cases} \arcsin\left(\frac{x_i - x_c}{A_i}\right) - \theta_i & \text{if } \frac{\delta x}{\delta \theta} > 0 \\ \arccos\left(\frac{x_i - x_c}{A_i}\right) - \theta_i + \frac{\pi}{2} & \text{otherwise.} \end{cases} \quad (6)$$

This equation has to be solved for all the possible trajectory's amplitudes $A_i \in [1, \dots, x_i]$ (with $x_i \leq N_x/2$), and each resulting pair (A_i, ϕ_i) determines the cell in the Hough space which has to be incremented. Once all the edge points have been processed, cells whose values are local maxima or peaks define the parameters for the trajectories in the EPI.

4.2. Trajectories Determination

Since the 3D points can have different radii, the corresponding EPI-trajectories will also have different amplitudes. This leads, in the EPI binary image, to a set of sinusoidal curves with a different number of points: less points for small amplitude curves, more points for large amplitude ones. Consequently, in the Hough spaces, the local maxima corresponding to larger amplitudes A will have a higher cell value than local maxima corresponding to smaller amplitudes. Moreover, EPI-points with large amplitude can be fitted to more curves than points with small amplitude. Therefore, the noise in the Hough space increases with large A , as can be seen in Figure 5 left. In order to correctly detect all these local maxima, each one of the two Hough spaces has to be post-processed. The first step consists of removing the low frequencies by subtracting from H its low-pass filtered version. Then, the result is rescaled by multiplying it with a *weighting matrix* W which gives more weight to small amplitudes and reduces high amplitudes. All the columns of W have the same weighting vector: an exponential function $e^{-0.001 * [1, 2, \dots, A_{max}]^T}$ was chosen. Eventually, thanks to the post-processing, it is possible to apply a global threshold (Otsu's method) to identify all the local maxima. The Hough space H_1 for the EPI of Figure 3 and the corresponding post-processed space are shown in Figure 5.

4.3. Trajectories Propagation

Each one of the identified local maxima is a pair (A, ϕ) which defines a trajectory that will be propagated in the EPI. For this task, the occlusion ordering rules of Section 3.1

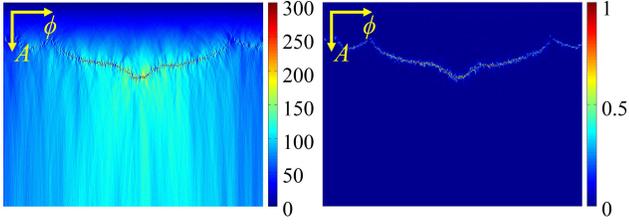


Figure 5. Orthographic Hough transform: left, the Hough space H_1 for the EPI of Figure 3; right, the corresponding post-processed space (rescaled between 0 and 1) for the local maxima detection.

are fundamental. To ensure the visibility of the foreground points, all the pairs (A, ϕ) from H_1 are sorted in descending order of amplitude A , whereas the pairs from H_2 are sorted in ascending order of amplitude (see rule 2). For each trajectory, defined by a pair (A_i, ϕ_i) , its x -coordinates are computed with Equation 3a. Then, an *occlusion visibility range*, based on the rules defined in Section 3.1, is used to determine the phase locations θ , *i.e.*, the EPI vertical coordinates, where the trajectory is visible. This range differs from H_1 to H_2 , and is defined as

$$\begin{aligned} \frac{\pi}{2} < \theta + \phi_i < \frac{3}{2}\pi & \quad \text{for } H_1 \\ 0 < \theta + \phi_i < \frac{\pi}{2} \wedge \frac{3}{2}\pi < \theta + \phi_i < 2\pi & \quad \text{for } H_2. \end{aligned} \quad (7)$$

For example, the point P_1 in Figure 2 has $\phi_{P_1} = 0$ and a visibility range equal to $[\frac{\pi}{2}, \frac{3}{2}\pi]$, *i.e.*, the quadrants II and III.

In order to take into account already propagated trajectories, and avoid new ones to overwrite them, an *EPI binary mask* is introduced. Moreover, to prevent propagation in wrong areas, trajectories from H_1 (H_2) are only propagated where the EPI-slopes are positive (negative). Eventually, the remaining portion of the trajectory can be written in the EPI, and in parallel into the EPI-mask. These steps are repeated for all the pairs (A, ϕ) . The first propagated trajectories are the ones related to points belonging to the quadrants (I, IV), *i.e.*, H_1 . Then, also the trajectories from H_2 are propagated (see rule 1). The propagation procedure is summarized in Algorithm 1.

```

foreach  $(A_i, \phi_i) \in H$  do
  compute the trajectory coordinates  $(x_i, \theta_i)$ ;
  remove occluded coordinates;
  remove masked coordinates;
  remove wrong slope coordinates;
  propagate the remaining trajectory portion;
end

```

Algorithm 1: Trajectory propagation for the space H .

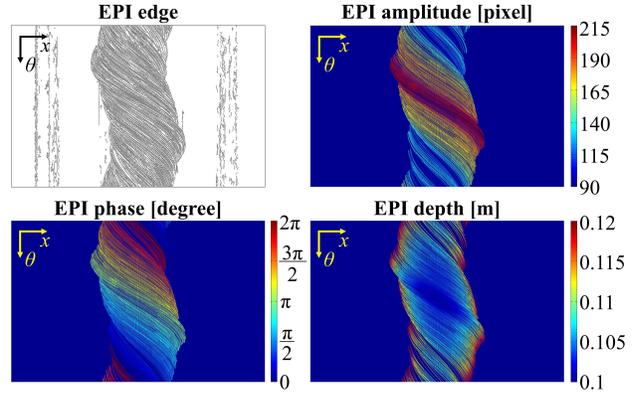


Figure 6. The input EPI-edge image (Canny) and the results of the orthographic Hough transform: EPI-amplitude, EPI-phase, and EPI-depth image. Note that the depth of a trajectory changes along the trajectory itself, since the 3D point is moving in space, whereas amplitude and phase are constant for each trajectory.

4.4. EPI-Depth Generation

Once all the trajectories have been propagated, it is straightforward to compute the depth map. In fact, any 3D point \mathbf{P} , which corresponds to a trajectory with parameters (A, ϕ) , has a depth Z , with respect to the origin C , defined by Equation 2b, with $R = m \cdot A$. The results of the orthographic Hough transform for the EPI of Figure 3 right are shown in Figure 6.

By applying the described steps to all the EPIs of the image volume, the final output is a set of sparse depth images of the target scene, one for each rotation angle θ .

5. Hough Transform for Perspective Camera

When a standard perspective camera is considered, Equations 5 show that a trajectory is not confined in a single EPI, but also moves in the $y\theta$ -plane, leading to a curve through the whole 3D image volume. The y -shift increases with the distance between a 3D point and the horizontal plane through the camera's optical center, where there is no shift. Therefore, in order to find a trajectory, a 3D search in the full image volume should be performed, as described by Feldmann *et al.* [8]. In this section, the Hough transform approach is adapted to the perspective projection case. In order to continue using EPI like slices, we propose an approximate solution which neglects the y -shift. Even though the full trajectory is not available, a portion of it is always visible in the EPI. One of the advantages of the Hough transform is that even a portion of a curve can be retrieved if it has enough support. Therefore, it is possible to reconstruct the EPI-trajectories and achieve good results even with this approximation.

5.1. Hough Space Generation

As in the orthographic case, the discretization of the Hough spaces depends on the chosen sensor and the acquired images. In this case, the relation between amplitude A in pixel and radius R in meters is defined in [9] as

$$R = 2 \cdot \frac{R_M \tan(1/(2FOV)) \cdot A}{\sqrt{4(\tan(1/(2FOV)))^2 \cdot A^2 + N_x^2}}, \quad (8)$$

where $FOV = 2 \arctan((\sigma * N_x)/2f)$ is the field of view. With this formula it is possible to associate the correct radius value to each trajectory’s amplitude $A \in [0, 1, \dots, N_x/2]$.

In the perspective case, the behavior of a trajectory is determined by Equation 5a. Therefore, in the voting procedure this equation has to be solved in order to find the trajectory’s phase offset ϕ from the EPI binary image point (x_i, θ_i) . We chose to invert the equation by means of a look-up table. The *trajectories determination* via local maxima detection follows the same procedure used for the orthographic case in Section 4.2.

5.2. Trajectories Propagation

Once a trajectory is determined, its amplitude A (radius R) and phase offset ϕ are used to compute its coordinates through the EPI. The propagation procedure is similar to the one described in Section 4.3, with two differences:

1. The x -coordinates are computed with Equation 5a;
2. The trajectories are no longer perfect sines. As can be seen in Figure 4, three-dimensional points with larger radius R deviate more from the ideal sinusoidal curve. The phase of the maximum amplitude, which determines the *occlusion visibility range* is

$$\phi_{max} = \arccos(R/R_M). \quad (9)$$

From this peak it is possible to determine the segments where the trajectory is visible and can be propagated.

The remaining steps are the same as in Algorithm 1. Eventually, the depth maps are computed by projecting every 3D point into each camera’s image plane, taking into account the perspective projection.

6. Experiments and Results

To evaluate the quality of the reconstruction, tests with both synthetic and real datasets were performed. The result of the proposed algorithm are a set of depth images, one for each rotation angle θ , which can be converted into a point cloud and eventually into a mesh. To this end, we used the *Poisson Surface Reconstruction* of [16]. The

same meshing procedure was employed to generate meshes from the point cloud obtained with two publicly available multi-view algorithms. The first is the patch based method *Clustering Views for Multi-view Stereo* (CMVS) from Furukawa and Ponce [11], which has an optimized view selection that discards some images due to the small baseline. The second is the *Multi-View Environment* (MVE) from Goesele *et al.* [13], which computes per-image depth maps, later merged in 3D space.

6.1. Synthetic Datasets

In order to evaluate the robustness with respect to specular surfaces, synthetic circular light fields of a Buddha’s head were generated with *Blender* [4]. For both telecentric and perspective lenses two datasets were generated by setting the surface to Lambertian and specular. All the datasets are composed of 720 images (*i.e.*, one image each 0.5°) with a resolution of 1001×1001 pixel, and pixel pitch $\sigma = 6 \mu\text{m}$. The focal length for the perspective camera is $f = 18$ [mm], whereas the telecentric camera has a magnification of 0.1. A visual comparison of the obtained meshes is presented in Figure 7. In the Lambertian case, our method yields to the best meshes, which are less noisy than MVE and have more details than CMVS. In the specular datasets, CMVS and MVE have more problems, due to the non-Lambertian surface. On the contrary, we are still able to obtain very good reconstructions, with much less noise, especially in the Buddha’s face. This is due to the way the voting procedure of the Hough transform uses all the available images to find the EPI-trajectories, which can be retrieved even if they are only partially visible. Note that in the perspective case our method is using an approximated model (we are ignoring the y -shift), whereas MVE and CMVS assume the right pinhole model. Table 1 shows the root-mean-square error (RMSE) [20] of the meshes in percentage, normalized by the extent of the bounding box.

Dataset	Ours	CMVS	MVE
Telecentric Lambertian	0.49	0.59	0.56
Telecentric Specular	0.45	0.91	0.72
Perspective Lambertian	0.54	0.62	0.59
Perspective Specular	0.56	0.91	0.81

Table 1. BUDDHA synthetic datasets: RMSE of the meshes in percentage. Our method outperforms the others and is not affected by specular surfaces.

6.2. Real Datasets

Real datasets were acquired both with a telecentric lens (*Zeiss Visionmes 105/11*) for the orthographic case, and a standard lens (*Zeiss Makro-planar 2/100 ZF.2*) for the perspective one. Calibration was performed for both the lenses, in order to remove distortion and determine the correct rota-

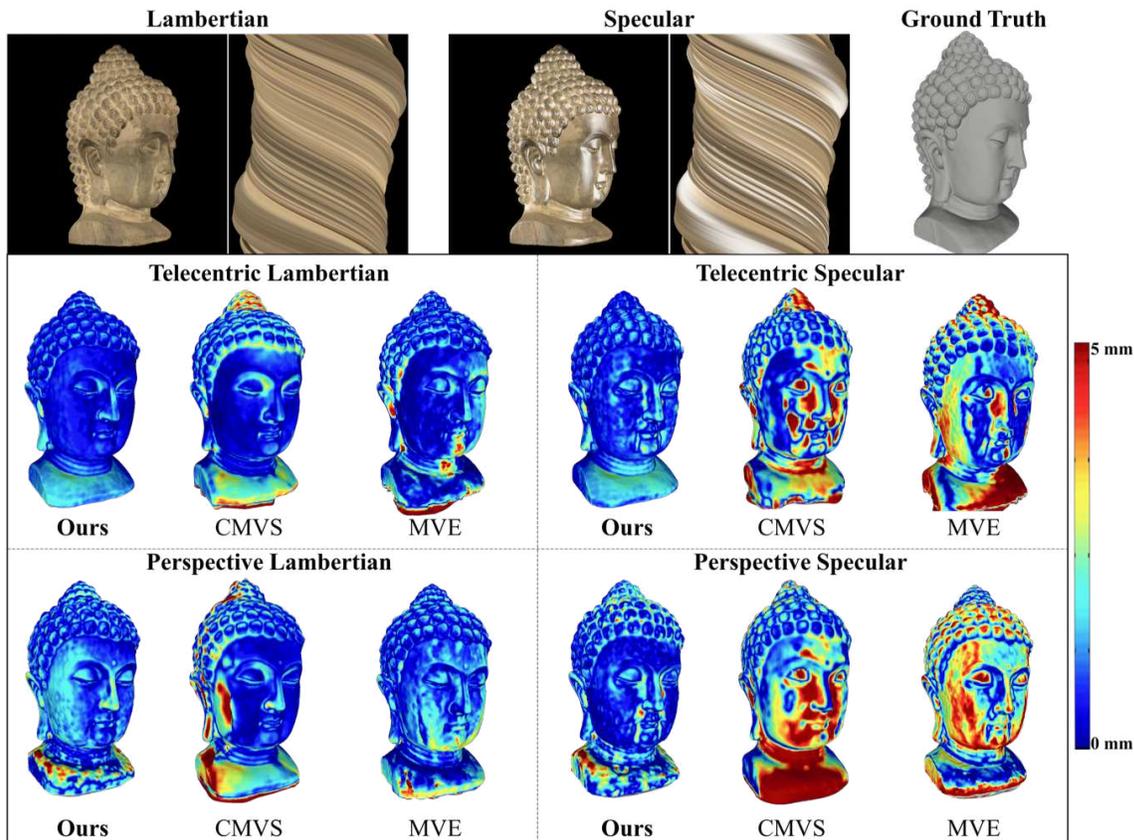


Figure 7. BUDDHA synthetic datasets: mesh comparison of our method with CMVS [11] and MVE [13] (the colors encode the error in millimeters, with respect to the ground truth mesh). Our results are always less noisy and more precise. Note the strong intensity variations in the EPI-trajectories for the specular case.

tion center. We used a *pco.edge 5.5* camera with a resolution of 2560×2160 pixel and a pixel pitch of $\sigma = 6.5 \mu\text{m}$. Test objects were placed on a high precision rotation stage, and light fields composed of $N = 720$ images were acquired. Figure 8 presents the results for the CAT dataset in the orthographic and perspective case. In the latter, the MVE reconstruction is not available since the algorithm failed with this dataset. However, it can be seen that our approach produces a result comparable to CMVS. In the dataset acquired with the telecentric lens we achieve the best reconstruction. In the same way, for the SEAHORSE dataset our approach outperforms the others in the orthographic case presented at the top of Figure 9. The reconstructions from data acquired with the standard lens, presented in the bottom of Figure 9, show again three similar results, meaning that our approximation is comparable to the other two algorithms. Eventually, we tested the robustness of our algorithm against non-Lambertian surfaces by reconstructing a drill bit, acquired with the telecentric lens. This is a highly specular and challenging metallic part. Nevertheless, we can precisely reconstruct the object and correctly retrieve the EPI-

trajectories even with strong intensity variations, as can be seen in Figure 10. In order to further appreciate the robustness to specular surfaces of the proposed algorithm, Figure 11 shows a frame of the circular light field, as well as an EPI. In this image the non-Lambertian effects of the metallic surface are visible, with two clear specular peaks due to the two main lights used to illuminate the scene. This type of data cannot be correctly resolved by classical multi-view algorithms which try to find correspondences between the views, assuming color constancy. However, our approach is able to determine these correspondences in the form of trajectories. Some of these are highlighted in the EPI, and the correspondent intensity values are plotted in Figure 11 (c). For simplicity, we report the intensity of the grayscale EPI, rescaled between 0 and 255. Also from this plot the two specular peaks are clearly distinguishable.

7. Conclusion

We introduced a novel method to recover depth information from circular light fields even in presence of specular reflections. Two variants were presented: one for images

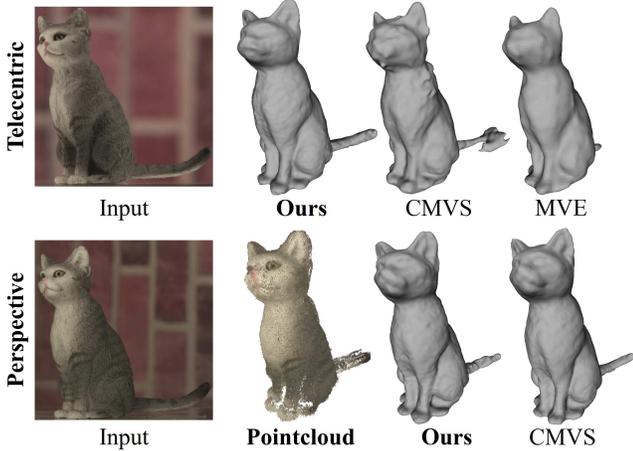


Figure 8. CAT datasets acquired with telecentric and perspective lenses: mesh comparison of our method with CMVS [11] and MVE [13]. Note: for the perspective case also the point cloud of our approach is presented, and only the CMVS mesh is available.

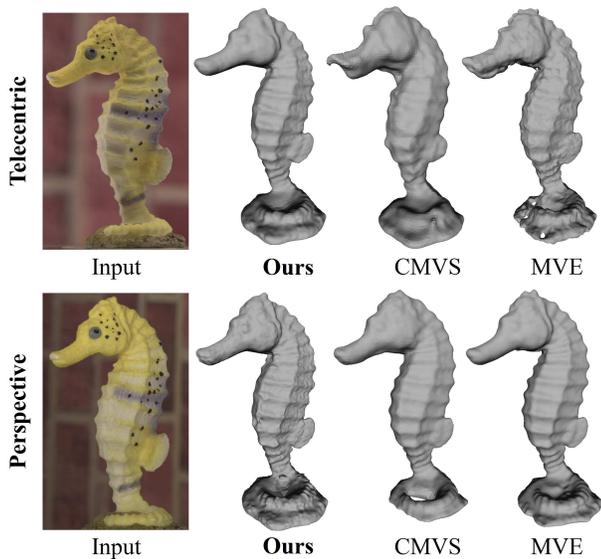


Figure 9. SEAHORSE datasets (telecentric and perspective): mesh comparison of our method with CMVS [11] and MVE [13].

acquired with telecentric cameras and the other for standard perspective lenses. Differently from classic linear light fields, with circular light fields it is possible to reconstruct the full 360° view of the target scene with just one continuous acquisition. Additionally, they allow retrieving depth even from image sequences acquired with telecentric lenses, a task which is not possible with simple linear motion. In this way, also setups that require telecentric optics can be used to make 3D reconstruction from images without having to change the lens or placing an additional perspective camera. Our method also overcomes the limitation to Lam-

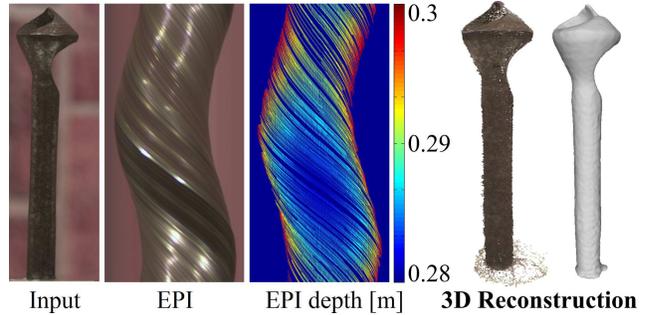


Figure 10. DRILL BIT dataset acquired with a telecentric lens: with our approach we are able to correctly identify EPI-trajectories even in presence of specular reflections, and consequently obtain a precise 3D reconstruction.

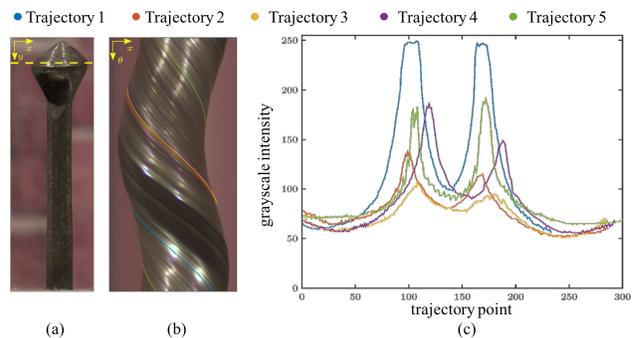


Figure 11. Trajectories analysis for the DRILL BIT telecentric dataset: first light field frame (a), and the EPI corresponding to the y -coordinate highlighted by the yellow dashed line (b). The correspondent grayscale intensity values of five trajectories are plotted in (c). These trajectories have similar intensity behaviours, with two specular peaks, due to the two main illumination units.

bertian surfaces of state-of-the-art algorithms, by using the Hough transform of binarized EPIs. This leads to a very robust estimation of the EPI-trajectories, which can be found in presence of specular reflections, noise, or even wiggles due to some imprecisions in the calibration or in the rotation mechanism. The robustness against specular surfaces makes our approach suitable to many tasks in industrial optical inspection. Besides 3D reconstruction, an example of application is material classification based on BRDF estimation. In fact, the intensity variation along each trajectory can be used to associate the material properties to a specific BRDF.

Acknowledgements

The authors would like to thank *Robert Bosch GmbH* and the *Heidelberg Collaboratory for Image Processing* for funding and supporting this research.

References

- [1] E. Adelson and J. Bergen. The plenoptic function and the elements of early vision. *Computational models of visual processing*, 1, 1991.
- [2] H. H. Baker and R. C. Bolles. Generalizing epipolar-plane image analysis on the spatiotemporal surface. In *In IJCV*, 1989.
- [3] J. Barron, D. Fleet, and S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.
- [4] Blender Foundation. Blender. <http://www.blender.org/>, 2016.
- [5] A. Criminisi, S. Kang, R. Swaminathan, R. Szeliski, and P. Anandan. Extracting layers and analyzing their specular properties using epipolar-plane-image analysis. *Computer vision and image understanding*, 97(1):51–85, 2005.
- [6] D. Crispell, D. Lanman, P. G. Sibley, Y. Zhao, and G. Taubin. Beyond silhouettes: Surface reconstruction using multi-flash photography. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pages 405–412, June 2006.
- [7] M. Diebold, B. Jähne, and A. Gatto. Heterogeneous Light Fields. In *Proc. International Conference on Computer Vision and Pattern Recognition*, June 2016.
- [8] I. Feldmann, P. Eisert, and P. Kauff. Extension of Epipolar Image Analysis to Circular Camera Movements. In *Int. Conf. on Image Processing*, pages 697–700, 2003.
- [9] I. Feldmann, P. Kauff, and P. Eisert. Optimized Space Sampling for Circular Image Cube Trajectory Analysis. In *Int. Conf. on Image Processing*, pages 1947–1950, October 2004.
- [10] Y. Furukawa and C. Hernández. Multi-View Stereo: A Tutorial. In *Foundations and Trends in Computer Graphics and Vision*, volume 9, pages 1–148, 2013.
- [11] Y. Furukawa and J. Ponce. Accurate, dense and robust multi-view stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(1), 2008.
- [12] A. Gershun. The Light Field. *J. Math. and Physics*, 18:51–151, 1936.
- [13] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz. Multi-View Stereo for Community Photo Collections. In *Proc. International Conference on Computer Vision*, pages 265–270, 2007.
- [14] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The Lumigraph. In *Proc. SIGGRAPH*, pages 43–54, 1996.
- [15] B. K. Horn and B. G. Schunck. Determining optical flow. Technical report, Cambridge, USA, 1980.
- [16] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, pages 61–70, 2006.
- [17] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross. Scene reconstruction from high spatio-angular resolution light fields. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2013.
- [18] D. Lanman, D. C. Hauage, and G. Taubin. Shape from depth discontinuities under orthographic projection. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 1550–1557, Sept 2009.
- [19] M. Levoy and P. Hanrahan. Light field rendering. In *Proc. SIGGRAPH*, pages 31–42, 1996.
- [20] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47:7–42, 2002.
- [21] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 519–528, 2006.
- [22] A. Vianello, G. Manfredi, M. Diebold, and B. Jähne. 3D Reconstruction by a Combined Structure Tensor and Hough Transform Light-Field Approach. In *Forum Bildverarbeitung 2016*, page 161. KIT Scientific Publishing, 2016.
- [23] A. Vianello, G. Manfredi, M. Diebold, and B. Jähne. 3D Reconstruction by a Combined Structure Tensor and Hough Transform Light-Field Approach. *tm - Technisches Messen*, 84(7-8):460–478, April 2017.
- [24] S. Wanner, J. Fehr, and B. Jähne. Generating EPI representations of 4D Light fields with a single lens focused plenoptic camera. *Advances in Visual Computing*, pages 90–101, 2011.
- [25] S. Wanner and B. Goldluecke. Globally consistent depth labeling of 4D light fields. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 41–48, 2012.
- [26] K. Yücer, C. Kim, A. Sorkine-Hornung, and O. Sorkine-Hornung. Depth from Gradients in Dense Light Fields for Object Reconstruction. In *Proceedings of International Conference on 3D Vision (3DV)*, 2016.
- [27] K. Yücer, A. Sorkine-Hornung, O. Wang, and O. Sorkine-Hornung. Efficient 3D Object Segmentation from Densely Sampled Light Fields with Applications to 3D Reconstruction. *ACM Transactions on Graphics*, pages 22:1–22:15, March 2016.