

# Edge Segmentation in Images of a Focused Plenoptic Camera

Niclas Zeller

Institute of Applied Research  
(IAF)  
University of Applied Sciences  
Karlsruhe, Germany  
niclas.zeller@hs-karlsruhe.de

Franz Quint, Clemens Zangl

Faculty of Electrical Engineering and  
Information Technology  
University of Applied Sciences  
Karlsruhe, Germany  
franz.quint@hs-karlsruhe.de

Uwe Stilla

Department of Photogrammetry  
and Remote Sensing  
Technische Universität München  
Munich, Germany  
stilla@tum.de

**Abstract**—In this article we present a method for 3D segmentation which is optimized for the recordings of a focused plenoptic camera. Since, caused by the imaging concept of a plenoptic camera, depth information is only received for regions of high contrast our approach is based on retrieving 3D edge segments out of the RGB-D data acquired from a plenoptic camera. In our method the depth data is filtered by a nonlinear filter. Afterwards, a 2D edge segmentation is performed to the RGB image. The 2D segments are projected to the 3D space by applying a weighted least squares estimate to the filtered depth data.

**Keywords**—3D Edge Segmentation, Focused Plenoptic Camera, RGB-D Camera

## I. INTRODUCTION

The concept of a plenoptic camera already has been developed more than hundred years ago [1][2]. Nevertheless, only for the last few years the existing graphic processor units (GPUs) are capable to evaluate the recordings of a plenoptic camera with acceptable frame rates ( $> 30$  fps).

Today, there exist mainly two concepts of a plenoptic camera which are based on a micro lens array (MLA). One concept is the “unfocused” plenoptic camera, which was developed by Ng in 2006 [3]. The strength of this camera is the image synthesis, like refocusing an image after recording. The second concept is the focused plenoptic camera (plenoptic camera 2.0), which was described the first time by Lunsdaine and Georgiev [4]. The advantage of the focused plenoptic camera is the high resolution of the synthesized image. Perwaß and Wietzke developed this concept further and optimized it for depth estimation [5].

The advantage of the Raytrix camera [5], compared to other depth cameras is especially its small size. Even though its depth accuracy strongly decays for far distances, the camera presents a promising alternative in close range photogrammetry. The camera seems to be ideal in navigation and collision avoidance applications where depth ranges up to 10 m and centimeter accuracy is sufficient.

Navigation applications usually consider man-made scenarios, which can be described quite well by geometric primitives. In this article we present a segmentation method

which is optimized for the recordings of a plenoptic camera. Due to the characteristics of the depth map acquired by a plenoptic camera we perform 3D edge segmentation. The received edges later can be combined to surfaces by common triangulation methods.

This article is organized as follows. Section II presents the concept of a focused plenoptic camera. In Section III we briefly analyze the characteristics of the depth map acquired from a plenoptic camera and Sections IV presents our segmentation algorithm. Section V presents some results of the algorithm.

## II. THE FOCUSED PLENOPTIC CAMERA

Different from traditional cameras, which only record the intensity of incident light on the image sensor, a plenoptic camera records the light-field inside the camera as a four dimensional (4D) function. By retracing the path of rays through the main lens the light-field outside the camera can be calculated. In [6] it is shown that in free space it is sufficient to define the light-field as a 4D function. Since the intensity along a ray does not change in free space, a ray can be defined by two position and two angle coordinates. From the recorded 4D light-field a depth map of the scene can be calculated or images focused on different object distances can be synthesized after recording.

As already mentioned in Section I there exist different concepts of a plenoptic camera. Since this article describes segmentation methods based on the recordings of a Raytrix camera, only the concept of this camera is presented here. Nevertheless, the existence of other concepts has to be mentioned [3][4].

Figure 1 shows the projection of an object which is in the distance  $a_L$  in front of a thin lens to the focused image in a distance  $b_L$  behind the lens. The relationship between the object distance  $a_L$  and the image distance  $b_L$  is defined by the thin lens equation as given in (1).

$$\frac{1}{f_L} = \frac{1}{a_L} + \frac{1}{b_L} \quad (1)$$





Figure 3: Recordings of a Raytrix camera. Left: Synthesized total focused image. Center: Raw depth image. Right: Filled depth image.

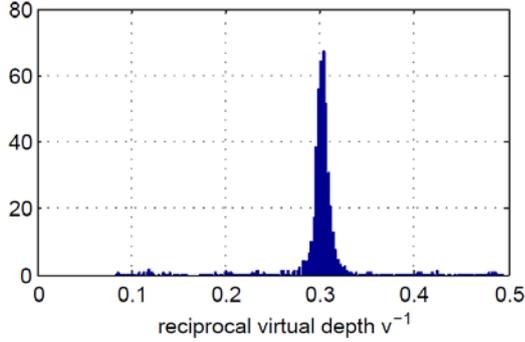


Figure 4: Histogram of the reciprocal virtual depth values  $v^{-1}$  of a planar object.

depth image. Subsection IV.B presents our line segmentation approach, which is applied to the total focused RGB image. Subsection IV.C describes how the line segments are fitted to the depth data.

#### A. Depth Preprocessing

To get rid of outliers, the depth map is filtered by a nonlinear filter. Therefore a quadratic filter mask of size  $N \times N$ , where  $N$  is an odd number, as well as a percentage value  $p$  is defined. Since, as mentioned in Section III, the reciprocal values of the virtual depth  $v^{-1}$  has a Gaussian distribution, the filtering is performed on these reciprocal values.

For filtering, the mask is centered to each pixel in the inverse depth map which contains a valid depth value. Out of all valid values under the mask the median  $\tilde{x}$  is calculated as given in (3).

$$\tilde{x} = \frac{1}{2} \cdot x_{\lfloor \frac{n+1}{2} \rfloor} + \frac{1}{2} \cdot x_{\lceil \frac{n+1}{2} \rceil} \quad (3)$$

In (3)  $x_i$  represents the valid depth values under the filter mask sorted in ascending order, where  $i \in \{1, 2, \dots, n\}$ . Afterwards, out of the  $p$  percentage of valid values which are centered around the median  $\tilde{x}$  the arithmetical mean  $\bar{x}$  is calculated. The definition of  $\bar{x}$  is given in (4). This mean is the new reciprocal depth value for the centered pixel.

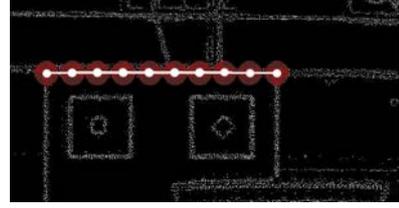


Figure 5: Calculation of the average depth values for certain points  $p_l$  on the estimated straight line.

$$\bar{x} = \frac{1}{\lfloor p \cdot n / 100 \rfloor} \cdot \sum_{i=\lceil (n+1-p/100 \cdot n) \cdot 0.5 \rceil}^{\lfloor (n+1+p/100 \cdot n) \cdot 0.5 \rfloor} x_i \quad (4)$$

If the number of valid values under the mask  $n$  is less than 5 % of the mask size  $N^2$ , the centered pixel is considered as outlier and is erased from the depth map. After filtering the values are converted back to virtual depths  $v$ .

#### B. 2D Image Contour Segmentation

The first step of the presented segmentation approach is a contour line segmentation which is applied to the 2D image without regarding the depth data. The contour segmentation is performed, because as shown in Section III depth information is only acquired at these contours (regions of high contrast).

For the contour segmentation firstly from the RGB image a gray scale image is calculated. From this gray scale image edge pixels are detected by using a Canny edge detector [7]. After detecting the edge pixels a line segmentation method is applied to these pixels. Therefore basically any common method can be used. For the results presented in Section V the functions implemented in OpenCV [8] were used. Here, a contour detection based on [9] is performed and the contour pixels are segmented into straight lines using the Ramer-Douglas-Peucker algorithm [10].

For further developments it is also considered fitting other shapes (e.g. ellipses) or already 2D geometric primitives to the edge image.

#### C. 3D Contour Fitting

After the contours in the 2D image are found and segmented into straight lines, these segments have to be projected to the 3D space using the depth data. Therefore, the transformation from 2D image coordinated to Cartesian 3D coordinates has to be determined. The virtual depth  $v$  is transformed into metric object distances based on the calibration method presented in [11] or [12]. Besides, as shown in [11] the camera matrix can be estimated based on any common camera calibration method (e.g. [13]).

Since the depth information acquired by the plenoptic camera is very noisy, it is not sufficient to calculate a 3D point cloud from the 2D depth map and estimate the 3D straight lines by linear regression. Instead, since we already have the 2D projection of the straight line which we want to estimate in 3D space, we define a number of points on this 2D line  $p_l$ .



Figure 6: Scene recorded by a Raytrix R5. Left: Total focused RGB image. Right: Edges detected by Canny edge detector.

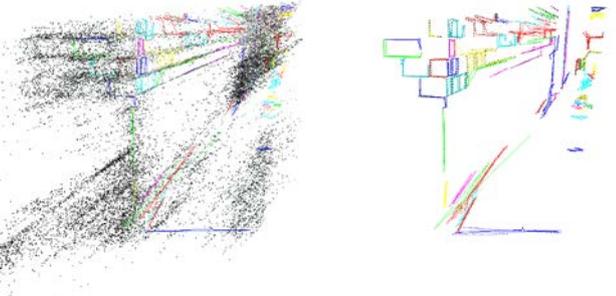


Figure 7: Segmentation results. Left: 3D raw data point cloud overlaid by segmentation results. Right: 3D line segments.

These points are aligned uniformly distributed over the whole line segment with regular spacing in between, as show in Figure 5. For each of the points  $p_l$  we collect all depth values within a radius  $r$  (red circles in Figure 5) and calculate the average value out of it. Besides, the number of valid depth values  $N_p$  within the given radius  $r$  is counted.  $N_p$  gives a measure of reliability for each point  $p_l$ .

After the average depth value for each point  $p_l$  is calculated, the points  $p_l$  are projected to the 3D space, resulting in the points  $p_w$ . Since the points  $p_l$  are all located on a straight line, the 3D points  $p_w$  will be located on a plane in 3D space. On this plane again we can define a 2D Cartesian coordinate system. The points in the new coordinate system we denote by  $p_w'$ . Based on the points  $p_w'$  a straight line is estimated. The straight line is estimated based on a weighted least squares estimation, where each point  $p_w'$  is weighted by its measure of reliability  $N_p$ .

## V. RESULTS

In this section we present the results of our segmentation method. Figure 6 (left) shows the RGB image of a scene. Figure 6 (right) shows the edge image resulted from the Canny edge detector. For the edge detector the threshold was set very high such that not all significant edges were detected. Otherwise the visualization of the 3D line segments, shown in Figure 7 would have been overloaded. For a real application the threshold of the edge detector has to be adjusted to detect all significant edges in the RGB image. Figure 7 (left) shows the 3D point cloud calculated from the raw depth data overlaid

by the 3D line segments. Figure 7 (right) shows only the 3D line segments. As one can see from Figure 7 our segmentation method supplies quite good results. Especially the accuracy of long straight edges is highly improved compared to the raw depth data.

## VI. CONCLUSION

In this article we presented a novel method for segmenting the RGB-D data acquired by a plenoptic camera. The segmentation algorithm is capable to handle the high number of outliers which occur in the depth image received from a plenoptic camera and improves the accuracy especially in areas of planar objects.

In further developments triangulation methods have to be applied to fill the gaps between the acquired lines segments to receive a complete 3D model of the recorded scene.

## VII. ACKNOWLEDGEMENT

This research is funded by the Federal Ministry of Education and Research of Germany in its program “IKT 2020 – Research for Innovation”.

## REFERENCES

- [1] F.E. Ives, “Parallax stereogram and process of making same,” 1903.
- [2] G. Lippmann, “Epreuves reversibles. Photographies integrales,” in *Comptes Rendus De l'Academie Des Sciences De Paris*, vol. 146, 1908, pp. 446-451.
- [3] R. Ng, “Digital light field photography,” PhD thesis, Stanford University, 2006.
- [4] A. Lumsdaine, and T. Georgiev, “Full resolution lightfield rendering,” Technical report, Adobe Systems, Inc, 2008.
- [5] C. Perwaß, and L. Wietzke, “Single lens 3d-camera with extended depth-of-field,” in *Human Vision and Electronic Imaging*, vol. XVII, 2012.
- [6] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen, “The lumigraph,” in *Proc. 23rd annual conference on computer graphics and interactive techniques, SIGGRAPH*, 1996, pp. 43-54.
- [7] J. Canny, “A Computational Approach to Edge Detection,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8, 1986, pp. 679-698.
- [8] G. Bradski, “The OpenCV Library,” *Dr. Dobb's Journal of Software Tools*, 2000.
- [9] S. Suzuki, and K. Abe, “Topological Structural Analysis of Digitized Binary Images by Border Following,” in *Computer Vision, Graphics, and Image Processing, CVGIP*, vol. 30 no. 1, 1985, pp. 32-46.
- [10] U. Ramer, “An iterative procedure for the polygonal approximation of plane curves,” in *Computer Graphics and Image Processing*, vol. 1, no. 3, 1972, pp. 244-256.
- [11] N. Zeller, F. Quint, and U. Stilla, “Calibration and accuracy analysis of a focused plenoptic camera,” in *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-3, 2014, pp. 205-212.
- [12] N. Zeller, F. Quint, and U. Stilla, “Kalibrierung und Genauigkeitsuntersuchung einer fokussierten plenoptischen Kamera,” in *DGPF Tagungsband*, vol. 23, 2014.
- [13] Z. Zhang, „Flexible camera calibration by viewing a plane from unknown orientations,” in *Proc. 7th IEEE International Conference on Computer Vision*, 1999, pp. 666-673.