Line-Based Multi-Label Energy Optimization for Fisheye Image Rectification and Calibration

Mi Zhang Jian Yao† Menghan Xia Kai Li Yi Zhang Yaping Liu
School of Remote Sensing and Information Engineering, Wuhan University, Hubei, China
†EMail: jian.yao@whu.edu.cn Web: http://cvrs.whu.edu.cn

Abstract

Fisheye image rectification and estimation of intrinsic parameters for real scenes have been addressed in the literature by using line information on the distorted images. In this paper, we propose an easily implemented fisheye image rectification algorithm with line constrains in the undistorted perspective image plane. A novel Multi-Label Energy Optimization (MLEO) method is adopted to merge short circular arcs sharing the same or the approximately same circular parameters and select long circular arcs for camera rectification. Further we propose an efficient method to estimate intrinsic parameters of the fisheye camera by automatically selecting three properly arranged long circular arcs from previously obtained circular arcs in the calibration procedure. Experimental results on a number of real images and simulated data show that the proposed method can achieve good results and outperforms the existing approaches and the commercial software in most cases.

1. Introduction

As a basic step for higher level tasks, such as structure from motion [17], visual navigation and SLAM [16], active research work has been carried out on automatic rectification and calibration for metric information from fisheye images in recent years. Their efforts have led to a remarkable improvements in this field. For example, various open-source omni-directional camera calibration toolboxes have been released[1][2][3] since the launch of the unifying theory for the central panoramic system [15]. These toolboxes are either based on 2D or 3D calibration patterns [1, 20] with prior knowledge or the line features manually selected from the fisheye image [4]. Recently, some automatic rectification and calibration algorithms for the single fisheye image have been proposed [5, 9], which mainly focus on the usage of line information from the distorted image.

It has been a trend that a variety of features on the fisheye image plane are taken into account for rectification and calibration, which introduce an additional challenge for automatic rectification and calibration of the omni-directional camera. A dominant paradigm in rectification and calibration for fisheye images is to use line features extracted from distorted fisheye images [3, 18]. These approaches compute the image of absolute conic from which the intrinsic parameters of the omni-directional camera are recovered. Generally speaking, given at least three conics on the fisheye image, the camera intrinsic parameters, consisting of the focal length, the image center and the aspect ratio, can be recovered from the decomposition of absolute conics. However, despite its geometric success, the rectification and calibration techniques still suffer from the problem of automatic extraction of conics from the fisheye image. In order to address the conic extraction issue, Burchardt and Voss [10] proposed to simplify the conic extraction problem as the circle extraction problem by assuming that the edge segmentations belonging to the same circle have the same distance to the center of the circle. Still a huge number of small arcs that are consistent with the same circle could not be correctly merged.

Some approaches try to make full use of the characteristic of lines on the unit sphere [6, 23]. These methods first detect the connected edge pixels and then project them on the sphere to verify whether they are restrained by the same great circle. Ying and Hu proposed to use the hough transform to detect line images which are then refined by minimizing the orthogonal distances to the conic [23]. However, these approaches suffer from the common limitation as perspective cases such as the expensive computation and importance of parameters sampling [6]. The conic detection algorithm proposed by Bukhari and Dailey [9] avoids the expensive computation and detects the circular arcs from the fisheye image directly. Their proposed algorithm extracts the connected components from the edge image and then finds the maximum pixels belonging to the same circle defined by three randomly selected points on the contour.

[1] http://www.robots.ox.ac.uk/~cmei/Toolbox.html
[2] http://webdiis.unizar.es/~lpuig/DLTOmniCalibration
[3] http://www.isr.uc.pt/~jpbar/CatPack/main.htm

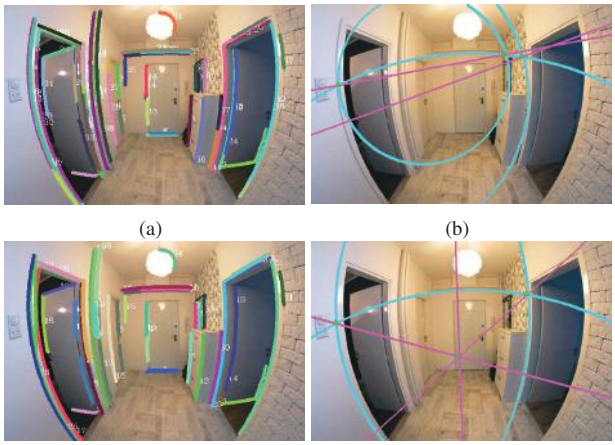


Figure 1. Incorrect estimation of intrinsic parameters due to the local minimum estimation of circular arcs: (a) Circular arcs without clustering; (b) Local minimum estimation of circular arcs; (c) Clustered circular arcs; (d) Correct estimation of circular arcs.

Nevertheless, this approach does not take into consideration the problem of merging the small circles sharing the same or the approximately same parameters. And thus it may lead to a local minimum estimation of intrinsic parameters as shown in Figure 1. Many approaches [21, 22] are proposed recently to solve this problem and obtain state-of-the-art results by using the line constrain between the projection on the viewing sphere of a space point and its catadioptric image.

Recently, one promising direction for automatic correction and calibration of the fisheye image from plumb-lines has emerged with the exploitation of the RANSAC technique [11, 18, 14]. Some of these approaches proposed to extract line images with 2-point RANSAC, which is suitable for different classes of omnidirectional systems. Another promising technique, the RANSAC Uncapacited Facility Location (UFL) method [18], simultaneously detects lines in natural images and estimates the camera parameters. However, this approach dose not take the problem of automatic selection of the conic arcs and the relationship of detected lines in the perspective plane into account. Therefore, it can not work as expected in some situations.

In this paper, we describe an easily implemented approach for fisheye image rectification using the line constrain in the undistorted perspective plane. Inspired by the approaches proposed in [9, 18, 12, 8, 7], we develop an algorithm to automatically merge short circular arcs sharing the same circular parameters and select long circular arcs using the Multi-Label Energy Optimization (MLEO) method. We also describe a framework for automatic calibration of the fisheye camera based on the previous work proposed in [2, 19]. Our approach mainly involves three steps: automatic circular arcs extraction from the fisheye image, the image rectification using clustered circular arcs, and the estimation of camera intrinsic parameters based on the selected conics. The general pipeline is illustrated in Figure 2.

Our approach proposed in this paper utilizes a similar arc detection technique proposed in [9] but in different merging and optimizing manners. Instead of simply detecting the arcs of the same contour without considering the similarity between contours, we formulate the circular arc merging and optimizing process as a multi-label optimization problem. Each detected circular arc is regarded as a label that can be represented by the circular center and radius while the circular points corresponding to the same label are served as input data. Our approach is motivated by [12], which provides a general model for optimization. In our cases, the data term can be represented by the deviations of each input data to its estimated center, which can be also regarded as its corresponding label. The smooth term can be computed from the Euclidean distances between the circular centers and the differences of the circular radii. In addition, we also consider the penalty cost that should be assigned to the label with the assumption that the circular arcs with small arc lengths should be merged.

Beside the initial circular arc extraction process described in [9], which can be regarded as a local circle finding and fitting procedure, we propose to apply the Multi-Label Energy Optimization (MLEO) algorithm as the global merging function to find the long circular arcs from the detected circular points. Therefore, our approach can obtain the more robust estimation of intrinsic parameters of the fisheye image due to the use of more long circular arcs.

Unlike the technique proposed in [18], our work in this paper extends this previous work in two aspects. Firstly, our approach adopts a general energy optimization framework similar to graph cuts to detect the circular arcs on the fisheye image. Secondly, we develop a simplified fisheye image correction algorithm with line constrains on the perspective image plane and present a camera calibration algorithm by automatically selecting the three properly arranged circular arcs. The main difference is that our automatic circular arc selection algorithm not only considers the relationship between lines in the fisheye image plane but also the line relations in the perspective plane. This provides an efficient and robust manner of simultaneously clustering the line contours and estimating the camera intrinsic parameters, which avoids local minimum calibration of the parameters.

The remaining parts of this paper is organized as follows. In Section 2, we describe the circular arcs extraction and selection procedure for the fisheye image rectification and calibration. The fisheye image rectification algorithm based on line constrains is presented in Section 3. The camera calibration algorithm based on the three selected circular

arcs is developed in Section 4. The experimental results and evaluation are provided in Section 5 followed by the conclusion drawn in Section 6.

2. Circular Arcs Extraction and Selection

To automatically rectify the fisheye image and estimate its camera intrinsic parameters, we need to extract and select at least three candidate circular arcs from it in the image plane. It is obvious that the connected components in the edge image are the possible candidates. So we start by applying the edge detector such as the Canny operator on the fisheye image, followed by using the connected components extraction algorithm to obtain possible circular arcs from the edge image. We aim at detecting the circular arc ω belonging to the fisheye image plane Π_F and assigning it a label L represented by a circular center $\mathbf{c} = (c_x, c_y)$ and its corresponding radius r with minimum energy costs in an energy optimization framework. The above step can be regarded as a circular arc extraction process. Further, in order to automatically select three circular arcs from a set of N detected ones represented by $\mathcal{W} = \{\omega_i | \omega_i \in \Pi_F\}_{i=1}^{N}$, we need to select three circular arcs $\{\Omega_s\}_{s=1}^{3}$ from $\mathcal{W}$ on the fisheye image plane Π_F with longer radii $\{r_s\}_{s=1}^{3}$ and their corresponding lines $\{\mathbf{l}_s\}_{s=1}^{3}$ should not totally parallel to each other on the perspective image plane Π_P. The radius r_i is estimated from the detected circular arc $\omega_i \in \Pi_F$ and the parallelism which is denoted by the line slope κ_i is computed from the corresponding line $\mathbf{l}_i$ on the perspective image plane (i.e., the rectified image plane) Π_P. These selected circular arcs $\{\Omega_s\}_{s=1}^{3}$ should support possible labels representing grouped circular arcs. This procedure can be viewed as a circular arc selection process. Following these assumptions, we can model the circular arc clustering and selection problem as an instance of Multi-Label Energy Optimization (MLEO) problem [12, 8, 7].

2.1. Multi-Label Energy Optimization Framework

Given a set of observations $\mathcal{P}$ and a finite set of labels $\mathcal{L}$ correlated to observations, the Multi-Label Energy Optimization (MLEO) problem aims at assigning each observation $p \in \mathcal{P}$ a label $f_p \in \mathcal{L}$ minimizing some function $E(f)$ over the joint labelling f. The mathematical formulation of the MLEO function is represented as:

$$E(f) = \sum_{p \in \mathcal{P}} D_p(f_p) + \sum_{(p,q) \in \mathcal{N}} V_{p,q}(f_p, f_q) + \sum_{l \in \mathcal{L}} h_l \cdot \delta_l(f), \quad (1)$$

where the first term $\sum_{p \in \mathcal{P}} D_p(f_p)$ denotes the data costs over all observations, the second term $\sum_{(p,q) \in \mathcal{N}} V_{p,q}(f_p, f_q)$ represents the smooth costs over all pairs of observations , and the final term $\sum_{l \in \mathcal{L}} h_l \cdot \delta_l(f)$ means the label costs whose indicator function defined on the label set $\mathcal{L}$ as:

$$\delta_l(f) = \begin{cases} 1, & \exists p : f_p \in \mathcal{L}, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

The data term often indicates a standard deviation in the candidate data group and the smooth term is often known as a prior that positively indicates the correlations between observation groups. While the label term gives a penalty to observations meaning that the objective function should use as fewer labels as possible.

Our innovation is inspired by this generalized MLEO method, and we will discuss how this method can be used for circular arcs clustering and selecting in details in the following subsections.

2.2. Circular Arcs Extraction

Let the set of circular arcs $\mathcal{W} = \{\omega_i | \omega_i \in \Pi_F\}_{i=1}^{N}$ be the detected connect components to be further clustered, and $\mathbf{c}_i = (c_x^i, c_y^i)$ and r_i be the corresponding circular center and radius of the i-th circular arc ω_i, respectively. Our objective is to find the minimum number of labels denoted by the corresponding circular parameters (i.e., circular centers and radii) by clustering circular arcs with the same or the approximately same circular parameters that fit the given set of circular arcs on the fisheye image plane. This problem can be regarded as the Multi-Label Energy Optimization (MLEO) problem. Let $s_i^k = \|\mathbf{p}_k - \mathbf{c}_i\|$ be the distance between the k-th circular point $\mathbf{p}_k = (x_k, y_k)$ and the circular arc center $\mathbf{c}_i = (c_x^i, c_y^i)$, and $c_i^k = |s_i^k - r_i|$ be the deviation of s_i^k and r_i. The energy function is denoted as

$$\begin{aligned} E\left(f, \hat{\theta}_c\right) &= E_{data}\left(f, \hat{\theta}_c\right) + E_{smooth}\left(f, \hat{\theta}_c\right) + E_{label}\left(f; \hat{\theta}_c\right) \\ &= \sum_{i=1}^{N(f)} \sum_{k=1}^{M_i} \left(s_i^k - \hat{r}_i\right)^2 + \sum_{i=1}^{N(f)} \sum_{j=1}^{N(f)} |\hat{r}_i - \hat{r}_j| \\ &\quad + \sum_{i=1}^{N(f)} \sum_{j=1}^{N(f)} \|\hat{\mathbf{c}}_i - \hat{\mathbf{c}}_j\|^2 + \sum_{i=1}^{N(f)} \delta_L^i \eta \frac{1}{M_i^2}, \end{aligned} \quad (3)$$

where the optimized parameters consist of the joint labelling f and the parameters $\hat{\theta}_c = \{\hat{r}_i, \hat{\mathbf{c}}_i\}_{i=1}^{N(f)}$ of the set of $N(f)$ clustered circular arcs $\hat{\mathcal{W}} = \{\hat{\omega}_i\}_{i=1}^{N(f)}$, M_i represents the number of points within the i-th clustered circular arc $\hat{\omega}_i$, δ_L^i has the same meaning as in Eq. (2), and η is the coefficient used to augment the penalty cost of the label ($\eta = 100$ was used in this paper). The Eq. (3) corresponds to the minimization of Eq. (1). Specifically, the data term $E_{data}(f, \hat{\theta}_c) = \sum_{i=1}^{N(f)} \sum_{k=1}^{M_i} dc_i^k = \sum_{i=1}^{N(f)} \sum_{k=1}^{M_i} \left(s_i^k - \hat{r}_i\right)^2$ represents the total deviation of the circular points to their corresponding circular centers, the smooth term $E_{smooth}(f, \hat{\theta}_c) = \sum_{i=1}^{N(f)} \sum_{j=1}^{N(f)} sc_i^j =$

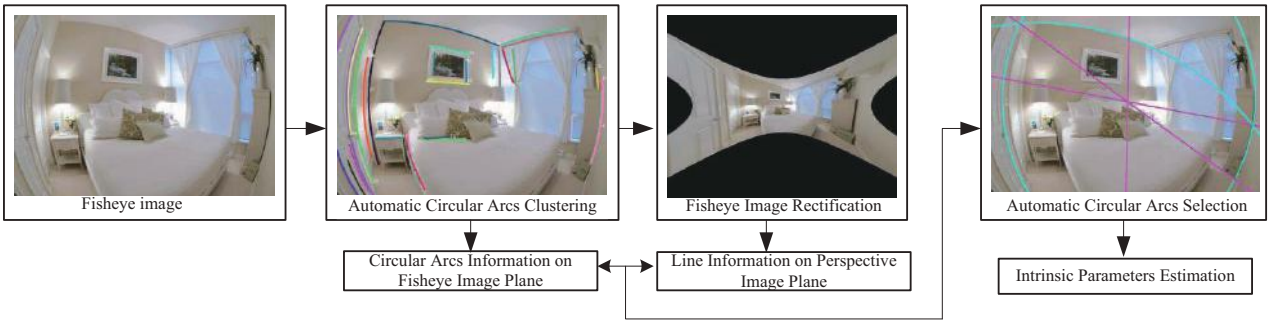


Figure 2. The general framework for our proposed fisheye image rectification and calibration algorithms.

$\sum_{i=1}^{N(f)}\sum_{j=1}^{N(f)}\left(|\hat{r}_i-\hat{r}_j|+\|\hat{\mathbf{c}}_i-\hat{\mathbf{c}}_j\|^2\right)$ denotes the total difference between all pairs of labels (L_i, L_j), which also represents the difference between different circular arcs parametrized by their circular centers and radii. While the label penalty term $E_{label}(f,\hat{\theta}_c) = \sum_{i=1}^{N(f)} lc_i = \sum_{i=1}^{N(f)} \delta_L^i \eta \frac{1}{M_i^2}$ regulates the penalty assigned to each circular arc model, meaning that the candidate circular arcs with short lengths should be merged.

This formulated model can also be depicted by Figures 3(a)-(b). These two figures illustrate the graph cut like process corresponding to Eq. (3). Given a set of circular points $\{\mathbf{p}_k\}$ and a set of circular arcs $\{\hat{\omega}_i \in \Pi_F\}$, each circular point has $n-link$ to its neighbour circular arc points and each circular point also connects to all terminals, namely labels $\{\hat{\omega}_i\}$ (i.e., the circular arcs), with the label cost lc_i. In our case, the data term corresponds to the $t-link$ in the graph and the cost of which is dc_i^k, the smooth term is consistent with the $n-link$ in the graph, which indicates a hidden difference between two circular arcs $\hat{\omega}_i$ and $\hat{\omega}_j$ with the smooth cost sc_i^j. While the label term sums up the label cost lc_i of each potentially clustered circular arc $\hat{\omega}_i$. The minimum cut is reached until these terms obtain its local minimum values respectively. This means that all the circular arc points sharing the same or the approximately same circular parameters are clustered and the candidate long circular arcs used for intrinsic camera calibration are detected. The results with simulated data and real images are illustrated in Figures 3(c)-(d) and Figures 3(e)-(f), respectively. It can be observed that this formulation applied to simulated and real images successfully clusters the circular arc points belonging to the same circle. The shorter circular arcs are correctly detected and clustered and the segments sharing the same label (i.e., the same circular center and radius) are identified and merged.

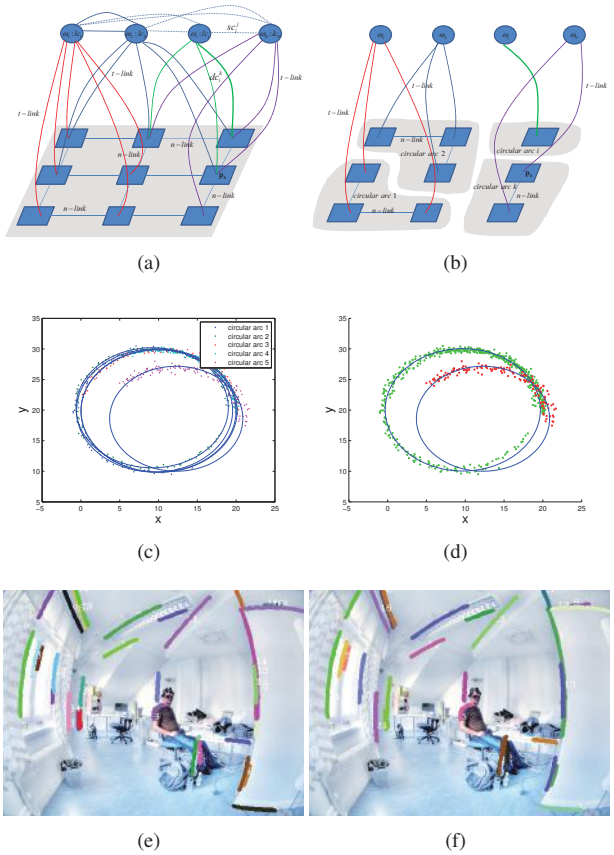


Figure 3. Illustrative graphs and examples for the MLEO energy optimization: (a) Initial graph corresponding to the terms in Eq. (3); (b) Minimum cut of circular arcs after optimization; (c) Initial circular arcs to be clustered for simulated data; (d) Clustered result of simulated data for initial circular arcs; (e) Initially detected circular arcs for the fisheye image before clustering. (f) Clustered result for the fisheye image.

2.3. Circular Arcs Selection

In previous subsection, we propose an algorithm to detect and cluster the circular arcs using the MLEO approach. At the second stage, we need to properly automatically select the three circular arcs which are served to estimate the camera intrinsic parameters. We formulate the energy function in the similar manner but for different goal. Assuming that the circular arcs $\hat{\mathcal{W}} = \{\hat{\omega}_i \in \Pi_F\}_{i=1}^{N(f)}$ are correctly detected and clustered, our goal is to find three circular

arcs $\{\Omega_s \in \hat{\mathcal{W}}\}_{s=1}^{3}$, which can be used for camera calibration. The main baffle is that the selected circular arcs with their corresponding lines in the perspective image plane Π_P should not absolutely parallel to each other, that is to say, for some lines $\{\mathbf{l}_i \in \Pi_P\}$ sharing the similar slopes need to be assigned as the same label. Furthermore, the selected circular arcs should be the ones with longer radii and lengths in the sense that the arcs deviating from the image center should be selected with higher priority. Finally, the distribution of these selected arcs ought to be in different directions on the fisheye image. Based on these assumptions, we formulate the Multi-Label Energy Optimization function for the circular arcs selection as:

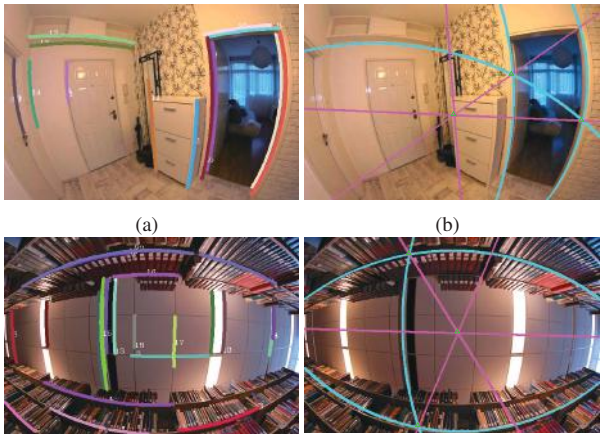
$$E_s\left(f,\hat{\theta}_s\right)=\sum_{c=1}^{4}\sum_{\hat{\omega}_i\in\mathcal{G}_m}\left(\ln\gamma(\hat{r}_i-\bar{r}_c)^2+\ln\beta(\hat{\kappa}_i-\bar{\kappa}_c)^2\right) + \sum_{m=1}^{4}\sum_{n=1}^{4}\lambda|\bar{r}_m-\bar{r}_n|, \tag{4}$$

where the optimized parameters consist of the joint labelling f and the parameters $\hat{\theta}_s = \{\bar{r}_c, \bar{\kappa}_c\}_{c=1}^{4}$, $\bar{r}_c$ and $\bar{\kappa}_c$ represent the optimized radius with respect to the fisheye image plane Π_F and the slope value with respect to the perspective plane Π_P of the c-th clustered group, $\hat{r}_i$ and $\hat{\kappa}_i$ denote the radius and slope of the candidate circular arc $\hat{\omega}_i$ on the planes Π_F and Π_P, respectively, and the coefficients γ, β and λ are used for regulating the weights of three terms in the MLEO method. In Eq. (4), we apply the log like function to the data term in order to augment the difference between each selected group. This equation is solved by using the MLEO method mentioned in the previous section.

Four groups could be clustered using the optimization Eq. (4), which takes the three baffles mentioned above into consideration. Each of these four groups contains a set of candidate circular arcs to be selected. To select three circular arcs from these four groups, we start by sorting each group according to the lengths of circular arcs within this group. Four candidate circular arcs can be chosen from the four groups. Again, we sort them according to their arc lengths and the final three circular arcs $\{\Omega_s\}_{s=1}^{3}$ with longer lengths are selected from them, which can be served as the circular arcs used for estimating the fisheye image intrinsic parameters. Figure 4 illustrates the results of our arc selection algorithm. It is obvious that the arcs that can be properly used for estimating the camera intrinsic parameters have been correctly selected.

3. Line-Based Fisheye Image Rectification

The main goal of the fisheye image rectification is to transform the distorted fisheye image to the so-called perspective image plane which preserves the majority of visual effects as we have usually seen. For example, the straight line on the perspective plane has to be straight [13].



(a) (b) (c) (d)

Figure 4. Circular arc selection results: (a)-(c) Clustered circular arcs; (b)-(d) Selected circular arcs for estimating the camera intrinsic parameters.

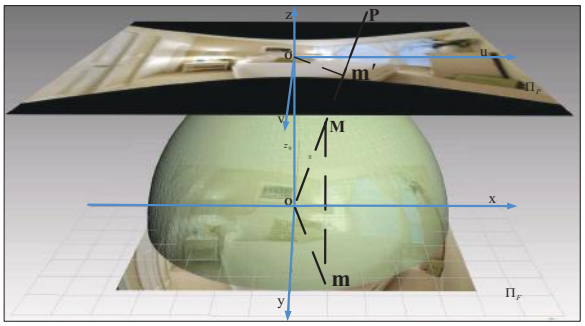


Figure 5. The fisheye image rectification model. The fisheye image point $\mathbf{m} = (x, y)$ on the fisheye image plane Π_F is the orthogonal projection of the point $\mathbf{M}$ on the sphere, which is the intersection of the ray $\mathbf{PO}$ and the sphere. The point $\mathbf{m}' = (u, v)$ is the corresponding point on the perspective image plane Π_P.

We use a simplified spherical projection model with line constrains on the perspective plane to automatically transform the fisheye image to the perspective image plane. This model is depicted in Figure 5. Assume that a point $\mathbf{P} = (X, Y, Z) \in \Re^3$ lies on a ray through the sphere center $\mathbf{O}$ which intersects the sphere surface on the point $\mathbf{M}$. The fisheye image point denoted by $\mathbf{m} = (x, y)$ can be viewed as the orthogonal projection of the point $\mathbf{M}$. The corresponding point $\mathbf{m}' = (u, v)$ on the perspective plane can be regarded as the intersection of the ray through the sphere center and the point $\mathbf{P}$ and the plane is parallel to the fisheye image plane.

The function that maps the distorted point $\mathbf{m}$ on the fisheye image plane Π_F to the corresponding point $\mathbf{m}'$ on the perspective plane Π_P and its inverse map are defined as:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{z_0}{\sqrt{R^2 - x^2 - y^2}} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{5}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{R}{\sqrt{u^2+v^2+z_0^2}} \begin{bmatrix} u \\ v \end{bmatrix}, \tag{6}$$

where R is the radius of the sphere, z_0 is the location of the perspective plane Π_P parallel to the fisheye image plane Π_F, (x, y) denotes a point on the plane Π_F, and (u, v) represents a point on the plane Π_P. This simplified model has the same meaning as the model proposed in [2] in which the camera position denoted by the parameter ξ in the function $\hbar(x)$ moves to infinite along the z axis. Let $\{\omega_i \in \Pi_F\}_{i=1}^N$ be a set of circular arcs and $\{\mathbf{l}_i \in \Pi_P\}_{i=1}^N$ be their corresponding lines in the perspective plane. The transformation relationship between the points lie on the circular arcs and their corresponding points lie on the lines can be expressed by Eqs. (5)-(6).

Given the fact that the straight line in the perspective plane remains to be straight, we use the line constrains on the perspective plane to denote the possible hidden distortion parameters. That is, when the line on the perspective plane is kept as straight after optimization, the distortion on the fisheye image is removed. In fact, the hidden distortion parameters with line constrains in this model are determined by the radius R of the sphere. So the problem of removing distortion is to find the radius R that minimizes the weighted sum of deviations of the lines in the perspective plane defined as:

$$E(d,R) = \sum_{i=1}^{N}\sum_{k=1}^{M_i} w_i \left|d(\mathbf{p}_k, \mathbf{l}_i)\right|^2 = \sum_{i=1}^{N}\sum_{k=1}^{M_i} w_i \left|\frac{a_i u_k + b_i v_k + c_i}{\sqrt{a_i^2+b_i^2}}\right|^2, \tag{7}$$

where $d(\mathbf{p}_k, \mathbf{l}_i)$ is the deviation of a point $\mathbf{p}_k = (u_k, v_k)$ in the circular arc ω_i with respective to the line $\mathbf{l}_i = (a_i, b_i, c_i)$ in the perspective space, and w_i is the weight factor of the line $\mathbf{l}_i$ in proportion to its length.

Now the objective function is fully defined, the parameter R which represents the hidden distortion variable can be estimated as a global minimum:

$$\hat{R} = \operatorname{argmin} E(d, R). \tag{8}$$

The minimization is done using the Levenberg-Marquardt (LM) non-linear least square method. Once the optimized parameter $\hat{R}$ is obtained using this non-linear optimization method, the fisheye image can be rectified.

4. Line-Based Camera Intrinsic Calibration

In order to present a complete rectification and calibration framework, we briefly describe the intrinsic parameters estimation algorithm proposed by [2] from three circular arcs. Assuming that the circular arcs, which can be regarded as conics, used for estimating the camera intrinsic parameters have been clustered and selected from detected contours, our goal is to estimate the camera intrinsic parameters comprised of the focal length, the image center and the aspect ratio.

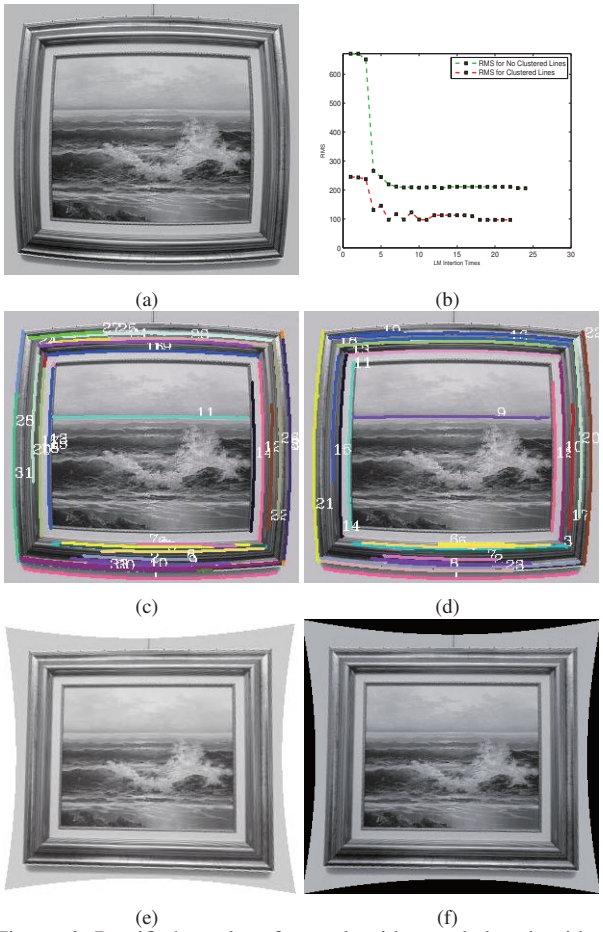


Figure 6. Rectified results of our algorithm and the algorithm in [9]: (a) Source fisheye image to be rectified; (b) Root Mean Square (RMS) of our algorithm and the algorithm proposed in [9]. (c) Extracted circular arcs from the source image using the algorithm proposed in [9]; (d) Extracted and clustered contours from the source image using our algorithm. (e) Corrected fisheye image using the algorithm proposed in [9]; (f) Rectified fisheye image using our algorithm.

The main steps involving in the catadioptric camera calibration from three circular arcs consists of:

1) Determine the conics $\{\Omega_s\}_{s=1}^3$ from the selected circular arcs;

2) Estimate the points $\{\mathbf{P}_s\}_{s=1}^3$ which are the intersections of the polar lines and the conic locus;

3) Estimate the absolute conic $\hat{\Omega}_\infty$ going through the points $\{\mathbf{P}_s\}_{s=1}^3$;

4) Perform the Chollesky decomposition of $\hat{\Omega}_\infty$ to estimate the intrinsic matrix $\mathbf{K}$.

For details reasoning and calculation, please refer to [3].

5. Experimental Results

The proposed fisheye image rectification and calibration algorithm was tested on a number of images, including the

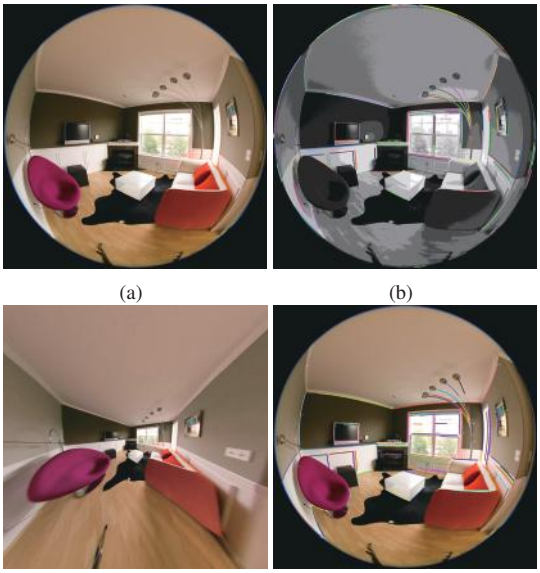


Figure 7. An example which can not be correctly rectified by the method presented in [9]: (a) Source fisheye image to be rectified; (b) Extracted circular arcs using algorithm proposed in [9]; (c) Rectified fisheye image using our method; (d) Clustered circular arcs using our algorithm.

Figure 8. Rectified results of our algorithm and the commercial software DxO. From top to bottom are the source images, the manually rectified results with the commercial software DxO, and our automatically rectified results.

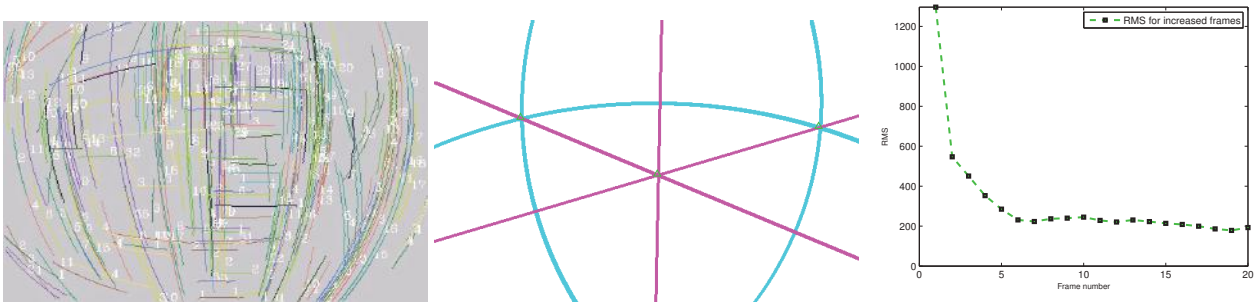
fisheye images in real scenes and the images downloaded from the Internet. To evaluate the performance of our algorithm, we made a comparison between our results and the commercial software DxO [4]. Also we verified the standard deviations of lines in the perspective image plane between our clustering results and the algorithm proposed in [9] when applying the LM optimization algorithm. For the cases where the number of circular arcs extracted from the single image are relatively insufficient to estimate intrinsic parameters, we used a synthetic method to combine the circular arcs from a variety of images captured by the same fisheye camera. We first extracted the circular arcs with our proposed technique for each fisheye image respectively. Then the synthetic circular arcs (see Figure 10(a)) were utilized for the fisheye image rectification. Finally, the three circular arcs selected from synthetic ones are served for estimating the camera intrinsic parameters using the existed algorithm presented in Section 4.

In Figure 6, we show the comparative results of our method and that obtained by the algorithm presented in [9]. It can be observed from Figure 6(d) that the RMS of the clustered circular arcs obtained by our method (denoted by red curve) is much smaller than that of the arcs without clustering obtained by the method in [9] (represented by green curve). In addition, we observed that the LM iteration times of our method is relatively short. Figure 7 shows an example which cannot be correctly rectified by the method in [9] because of the incorrect detection of circular arcs on the out rings edges. However, our method can successfully rectify it based on our clustering results. In Figure 8, we compared our automatically rectified results with manually rectified results of the commercial software DxO. Our line constrained method produced the similar results as the DxO did. Figure 9 presents the results for images downloaded from the Internet. These rectified images produce the visual effects as we have expected. Our method well preserves the line properties in the perspective plane and the straight lines on the rectified image plane remains straight. As the synthetic method mentioned previously, Figure 10 depicts the corresponding results. The circular arcs in different colors represent the circular arcs detected in different fisheye images. From Figure 10, we observed that the RMS of line deviation of our synthetic method becomes more and more steady with the increase of the synthetic frame number. The three circular arcs are correctly selected from the synthetic line image using our proposed algorithm. This makes sense for the situation where the line information is infertile on a single fisheye image and provides a flexible way for estimating the camera intrinsic parameters.

The proposed fisheye image rectification and calibration algorithm was implemented in C++ with the OpenCV library. The source code and more experimental results are publicly available at http://cvrs.whu.edu.cn/projects/FIRC/.

[4] http://www.dxo.com/

Figure 9. Rectified results of the images from the Internet: the source images downloaded from Internet in the first row, the clustered circular arcs using our proposed algorithm in the second row, and the rectified results in the last row.



(a) (b) (c)

Figure 10. Synthetic data for the fisheye image rectification and calibration: (a) Synthetic circular arcs from 20 frames captured by the same fisheye camera; (b) Selected circular arcs used for estimating the camera intrinsic parameters ($\gamma = \frac{1}{100}$, $\beta = 1.0$, $\lambda = 1.0$); (c) Root mean square (RMS) vs the number of frames used..

6. Conclusion

In this paper, we proposed an algorithm for automatic fisheye image rectification and calibration and devised a pipeline based on the pump-line approach. Circular arcs are automatically detected and clustered by using the MLEO approach. In addition, we presented an automatic circular arcs selection algorithm, which considers both the property of lines on the perspective image plane and the characteristic of circular arcs on the fisheye image plane. Our proposed algorithm was tested in various situations, including fisheye images captured in real scenes and images downloaded from Internet. We also made a comparison between our results and the existed approaches. Experimental results demonstrate the robustness of our proposed algorithm. However, our method still has a limitation that a reasonable initial value should be provided for the Levenberg-Marquardt (LM) iteration process. Our future work will focus on this issue and provide a time-saving approach for estimating the camera intrinsic parameters from multiple fisheye images captured by the same catadioptric camera.

Acknowledgement

This work was partially supported by the National Basic Research Program of China (Project No. 2012CB719904), the National Natural Science Foundation of China (Project No. 41271431), and the South Wisdom Valley Innovative Research Team Program.

References

[1] D. Aliaga. Accurate catadioptric calibration for real-time pose estimation in room-size environments. In *ICCV*, 2001. 1

[2] J. P. Barreto. *General Central Projection Systems Modeling, Calibration And Visual Servoing*. PhD thesis, University of Coimbra, 2003. 2, 6

[3] J. P. Barreto and H. Araujo. Paracatadioptric camera calibration using lines. In *ICCV*, 2003. 1, 6

[4] J. P. Barreto and H. Araujo. Geometric properties of central catadioptric line images and their application in calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1327 – 1333, 2005. 1

[5] J. P. Barreto, J. Roquette, P. Sturm, and F. Fonseca. Automatic camera calibration applied to medical endoscopy. In *BMVC*, 2009. 1

[6] J. C. Bazin, I. Kweon, C. Demonceaux, and P. Vasseur. Rectangle extraction in catadioptric images. In *ICCV*, 2007. 1

[7] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124 – 1137, 2004. 2, 3

[8] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001. 2, 3

[9] F. Bukhari and M. N. Dailey. Automatic radial distortion estimation from a single image. *Journal of Mathematical Imaging and Vision*, 45(1):31–45, 2013. 1, 2, 6, 7

[10] C. B. Burchardt and K. Voss. A new algorithm to correct fish-eye- and strong wide-angle-lens-distortion from single images. In *ICIP*, 2001. 1

[11] J. B. Cameo, G. L. Nicolas, and J. J. Guerrero. A unified framework for line extraction in dioptric and catadioptric cameras. In *ACCV*, 2012. 2

[12] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. *International Journal of Computer Vision*, 96(1):1–27, 2012. 2, 3

[13] F. Devernay and O. Faugeras. Straight lines have to be straight – automatic calibration and removal of distortion from scenes of structured environments. *Machine Vision and Applications*, 13(1):14–24, 2001. 5

[14] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 2

[15] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems and practical implications. In *ECCV*, 2000. 1

[16] C. Mei. *Laser-Augmented Omnidirectional Vision for 3D Localisation and Mapping*. PhD thesis, INRIA Sophia Antipolis, Project-team ARobAS, 2007. 1

[17] C. Mei and F. Malis. Fast central catadioptric line extraction, estimation, tracking and structure from motion. In *ICIRS*, 2006. 1

[18] R. Melo, M. Antunes, J. P. Barreto, G. Falcao, and N. Goncalves. Unsupervised intrinsic calibration from a single frame using a "plumb-line" approach. In *ICCV*, 2013. 1, 2

[19] L. P. Morales. *Omnidirectional multi-view systems: calibration, features and 3D information*. PhD thesis, Departamento de Informtica e Ingeniera de Sistemas Escuela de Ingeniera y Arquitectura Universidad de Zaragoza, Spain, 2011. 2

[20] C. Toepfer and T. Ehlgen. A unifying omnidirectional camera model and its applications. In *ICCV*, 2007. 1

[21] F. C. Wu, F. Duan, Z. Y. Hu, and Y. H. Wu. A new linear algorithm for calibrating central catadioptric cameras. *Pattern Recognition*, 41(10):3166–3172, 2008. 2

[22] X. H. Ying and Z. Y. Hu. Catadioptric camera calibration using geometric invariants. In *ICCV*, 2003. 2

[23] X. H. Ying and Z. Y. Hu. Catadioptric line features detection using hough transform. In *ICPR*, 2004. 1