Calibration Method for Fisheye Camera Based on Multi-checkerboard Detection

Zhou Su[1,2,a], Zhu Xiaofeng[1,b,*], Lu Yanda[1,c]

[1]School of Automotive Studies, Tongji University, Shanghai, China
[2]Sino-German College, Tongji University, Shanghai, China
[a]zhousu@tongji.edu.cn, [b]zhuxiaofeng@tongji.edu.cn, [c]luyanda@tongji.edu.cn
[*]Corresponding author

Keywords: Fisheye camera, Calibration, Inverse perspective transformation, Corner detection

Abstract: This paper proposes a fisheye camera calibration method based on multiple chessboard detection. To address the complexity and multiple-frame requirement of traditional calibration methods, this method detects multiple chessboard corner points using libcbdetect algorithm and obtains pixel coordinates. Then, it uses a depth-first search algorithm to obtain the world coordinates of the chessboard corners and calculates the homography matrix based on the world-pixel coordinates pair through RANSAC algorithm. Finally, the undistorted image is transformed into a bird's eye view (BEV) using the obtained homography matrix. This method is simple and effective, and can improve the accuracy of lane keeping functions, which is of practical significance in autonomous driving applications.

1. Introduction

In autonomous driving technology, lane-keeping function is a crucial part. Lane-keeping function requires accurate detection of parameters such as lane curvature radius and lane spacing, and therefore, inverse perspective transformation (IPM) is needed to obtain a bird's-eye view of the lane lines from the camera's perspective [1]. Fisheye camera is often used as an important sensor for environment perception in autonomous driving, and calibrating the fisheye camera is necessary for IPM. This involves obtaining the intrinsic and extrinsic parameters of the camera.

Traditional camera calibration methods often rely on chessboard patterns [2] for corner detection. However, popular calibration tools often have a complicated process. Mainstream calibration toolboxes require specifying the size of the chessboard in advance and only support single-frame detection of a single chessboard, making it necessary to capture multiple frames to achieve full coverage of the chessboard on the image. Therefore, proposing a convenient and robust fisheye camera calibration method is of great significance to the development of autonomous driving technology. Traditional calibration methods for cameras and depth sensors require the use of multiple sets of data for calibration, professional equipment, and a long time to complete, and are difficult to implement in practical applications. In contrast, this method only requires one image to calibrate the fisheye camera.

2. Basic Principle

A fisheye camera calibration method based on multi-chessboard detection is proposed for the fisheye camera calibration process. Firstly, the camera's intrinsic parameters are calibrated according to the fisheye camera model, and distortion-corrected images are acquired. Then, multiple chessboard corner points are detected and pixel coordinates are obtained based on the libcbdetect algorithm. Next, the chessboard corner points' world coordinates are obtained based on the depth-first search algorithm. The homography matrix is calculated by the RANSAC algorithm based on the world-pixel coordinate pairs. Finally, the distortion-corrected image is transformed using the obtained homography matrix to produce a bird's-eye view (BEV) image.

2.1. Fisheye model and Intrinsic calibration

The calibration process of a fisheye camera is essentially finding the mapping relationship between 3D points in the world coordinate system and pixel points in the pixel coordinate system during the imaging process. Calibration is generally divided into two steps: intrinsic calibration and extrinsic calibration. Intrinsic calibration is to find the relationship between 2D pixel points on the image and 3D points in the camera coordinate system. Extrinsic calibration is to find the correspondence between 3D points in the camera coordinate system and 3D points in the world coordinate system, that is, to determine the pose of the camera in the world coordinate system. To perform intrinsic calibration, a fisheye camera model needs to be constructed, which establishes the relationship between 2D image coordinates and 3D vectors based on a non-linear imaging geometry model. Compared with pinhole cameras, the complex refraction relationships of fisheye cameras result in their projection model being divided into two main types: the Kannala-Brandt model [3] and the OCam model[4]. The former proposes a unit-sphere projection model, which divides the fisheye camera projection into two steps. The specific imaging process is shown in Figure 1. In Figure 1, θ is the angle between the incident light and the Zc axis, r is the distance from the image point to the distortion center, Oxyz is the pixel coordinate system, and OcXcYcZc is the camera coordinate system. In the first step, the point P from the 3D world coordinate system is projected onto a virtual unit sphere that coincides with the camera's own coordinate system and the sphere's center. In the second step, the point on the sphere is projected onto the pixel coordinate system to form the image point p, and this projection relationship is non-linear.
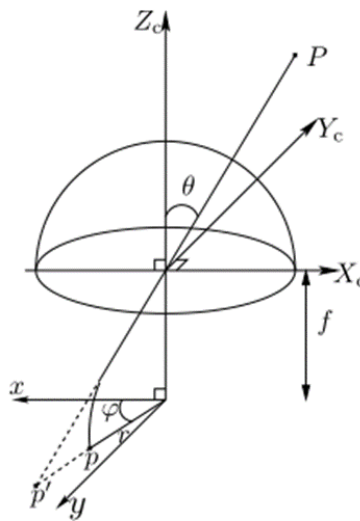

Figure 1: Fisheye model.

Kannala classified the different projection relationships between the camera coordinate system and the pixel coordinate system of a fisheye camera into four types: equidistant projection model, equisolid angle projection model, orthographic projection model, and stereographic projection model. Due to the inability to accurately manufacture lenses according to a specific projection model in practical lens production, a generalized projection model was proposed based on the four projection models. The odd function of θ, r (θ), can be obtained from the first four projection models, and r(θ) obtained by Taylor expansion can be represented by odd-degree polynomials of θ. The projection function is represented as:

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + k_4\theta^7 + k_5\theta^9 + \cdots \tag{1}$$

In OpenCV, taking the first five terms is sufficient to meet the accuracy requirements, and the calibration process of the fisheye camera intrinsics is to solve the coefficients of the projection function's first five terms.
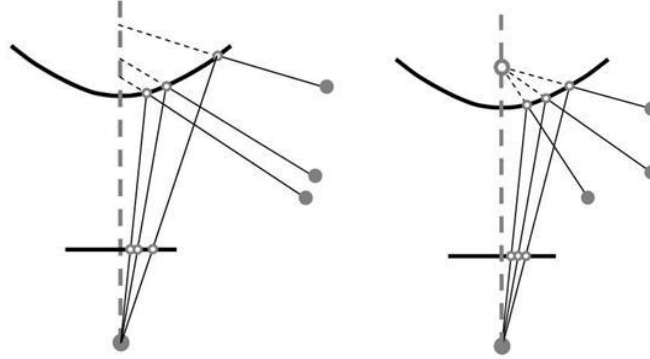

Figure 2: a) Non-central camera model b) Central camera model

MATLAB uses the OCam fisheye camera model, which represents the fisheye camera's perspective relationship as a combination of a perspective camera and a curved mirror, as shown in Figure 2. The mirror model is represented by a rough curve, and the imaging plane is represented by a rough straight line. Depending on whether the light emitted from the perspective camera extends to a point on the mirror, it is divided into non-central camera models and central camera models. Among them, the central camera model has the light rays intersecting at a single effective viewpoint.

In the OCam model, the following assumptions are made: 1. the mirror camera model is central, 2. the camera optical axis is well aligned with the mirror axis, 3. the mirror is symmetric along the mirror axis, and 4. lens distortion is integrated into the projection function. Based on these assumptions, a camera model is constructed.

Let p be the image point, (u, v) be the pixel coordinates of p, and vector P be the vector from the single effective viewpoint to point p. It can be assumed that there is a relationship between vector P and pixel coordinates u, v as follows:

$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} u \\ v \\ f(u,v) \end{bmatrix} \tag{2}$$

From assumption 2, it can be deduced that x and y are proportional to u and v, respectively:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \alpha \cdot u \\ \alpha \cdot v \end{bmatrix}, \alpha > 0 \tag{3}$$

From assumption 3, it can be deduced that the function f (u, v) depends only on the distance from the point to the image center. Let $\rho = \sqrt{u^2 + v^2}$, and equation (2) can be simplified as:

$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} u \\ v \\ f(\rho) \end{bmatrix} \tag{4}$$

The function f (ρ) can be described using a polynomial:

$$f(\rho) = a_0 + a_1\rho + a_2\rho^2 + a_3\rho^3 + a_4\rho^4 + \cdots \tag{5}$$

2.2. Multi-chessboard corner detection algorithm

Two commonly used corner detection algorithms are the Harris corner algorithm [5] and the Shi-Tomasi corner algorithm [6]. Geiger et al. [7] proposed a more robust corner detection method for dealing with image noise and improving localization accuracy. Based on the literature, a multi-chessboard detection algorithm was implemented, which consists of three steps: 1) locating the chessboard corners, 2) sub-pixel level corner detection and refinement of direction, and 3) optimizing the energy function and growing the chessboard.

Two corner prototypes with four filters each are shown in Figure 3 for corner detection methods of various significant distortion.
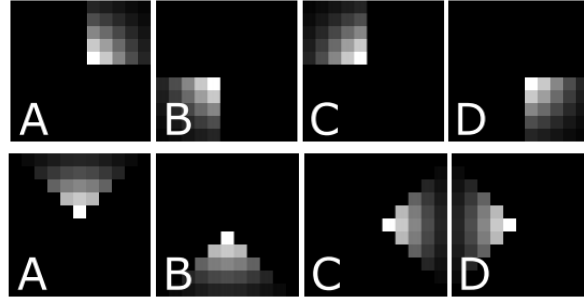

Figure 3: Corner prototype 1 and Corner prototype 2.

To convolve the image with the filter kernel, and calculate the similarity (corner likelihood) between each pixel and the corner. The calculation method is as follows:

$$c = \max(S_1^1, S_2^1, S_1^2, S_2^2) \tag{6}$$

$$S_1^i = \min\left(min\left(f_A^i, f_B^i\right) - \mu, \mu - min\left(f_C^i, f_D^i\right)\right) \tag{7}$$

$$S_2^i = \min\left(\mu - min\left(f_A^i, f_B^i\right), min\left(f_C^i, f_D^i\right) - \mu\right) \tag{8}$$

$$\mu = 0.25\left(f_A^i + f_B^i + f_C^i + f_D^i\right) \tag{9}$$

f_A^i represents the convolution response of kernel A and prototype i (i=1,2) at a certain pixel. S_1^i and S_2^i represent the possibility of different prototype flipping.

For each pixel, the corner similarity is calculated to obtain a corner similarity map. Non-maximum suppression (NMS) [8] is used to obtain candidate points. Weighted directional histograms are calculated using Sobel filtering [9] for the candidate points, and mean shift algorithm [10] is used to calculate the two main modes α_1 and α_2. A template T is constructed based on edge directions and the expected gradient magnitude $\|\nabla I\|_2$. The product of $T * \|\nabla I\|_2$ and the similarity score from Equation (6) is used as the corner score, and the final corner list is obtained by thresholding the score, where the "*" denotes a normalized cross-correlation operator.

In subpixel-level corner localization, the following fact is utilized: at the ideal corner position $c \in \mathbb{R}^2$, the image gradient $g_p \in \mathbb{R}^2$ of its neighboring pixel $p \in \mathbb{R}^2$ should be approximately orthogonal to p-c, leading to the optimization problem:

$$\mathbf{c} = \arg\min_{\mathbf{c}'} \sum_{\mathbf{p}\in\mathcal{N}_{\mathbf{I}}(\mathbf{c}')} \left(\mathbf{g}_{\mathbf{p}}^{T}(\mathbf{p}-\mathbf{c}')\right)^{2} \tag{10}$$

$\mathcal{N}_{\mathbf{I}}$ is a local neighborhood of 11x11 pixels. Note that pixels are automatically weighted according to their gradient magnitudes. By taking the derivative of equation (10) with respect to c' and setting it to zero, we obtain:

$$\mathbf{c} = \left(\sum_{\mathbf{p}\in\mathcal{N}_{\mathbf{I}}} \mathbf{g}_{\mathbf{p}}\mathbf{g}_{\mathbf{p}}^{T}\right)^{-1} \sum_{\mathbf{p}\in\mathcal{N}_{\mathbf{I}}} \left(\mathbf{g}_{\mathbf{p}}\mathbf{g}_{\mathbf{p}}^{T}\right)\mathbf{p} \tag{11}$$

Here, $g_{\mathbf{p}}^{i}$ represents the i-th element of g_{p}.

The extracted corner points are grown into a checkerboard grid using an energy function that is minimized to recover the checkerboard structure. Here, the energy function for the checkerboard grid is defined as:

$$E(\mathcal{X},\mathcal{Y}) = E_{\text{corners}}(\mathcal{Y}) + E_{\text{struct}}(\mathcal{X},\mathcal{Y}) \tag{12}$$

The energy function for the chessboard is defined as follows, where $\mathcal{X} = \{\mathbf{c}_{1}, \dots, \mathbf{c}_{N}\}$ is the set of candidate corners, and $\mathcal{Y} = \{\mathbf{y}_{1}, \dots, \mathbf{y}_{N}\}$ is the corresponding labels, where $\mathbf{y} \in \{\mathcal{O}\} \cup \mathbb{N}^{2}$ represents outliers $(\mathcal{O})$ or rows and columns in the chessboard $(\mathbb{N}^{2})$. The first term E_{corners} represents the negative of the total number of corners in the current chessboard. The second term E_{struct} describes the matching degree of predicting the third corner using two adjacent corners. For every triplet of adjacent corners in each row and column of the chessboard, the structural energy is calculated. The maximum value of the structural energy of each triplet is taken as the structural energy of the chessboard. It should be noted that since the structural energy constraint uses local linear constraints, the above chessboard growing method is also suitable for high-distortion images captured by fisheye lenses.

3. Calibration experiment

To facilitate the calibration process for different vehicle models and ensure comprehensive coverage of the camera image, calibration stickers with chessboard patterns were designed as shown in the figure 4.
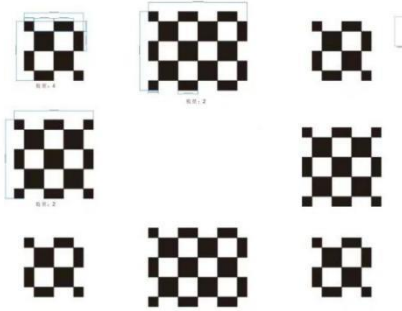

Figure 4: Checkerboard sticker.

The experimental vehicle was equipped with a fisheye camera. First, the camera was calibrated for intrinsic parameters based on the Kannala-Brandt model, and the undistorted images were obtained. Corner detection was performed on the undistorted images, and the detected corners are shown in the figure 5.
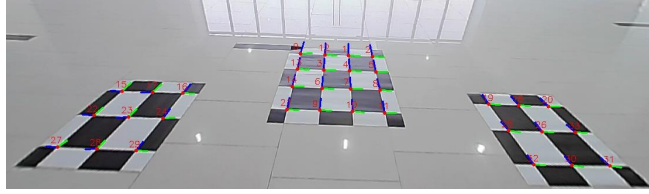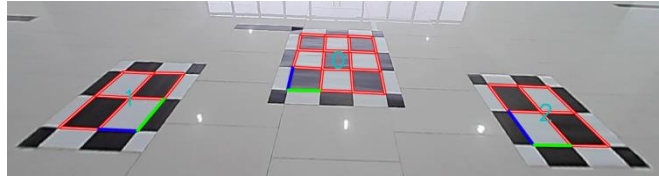
Figure 5: Corner detection.

Figure 6: Checkerboard growth.

The detected corner points were used to fit the checkerboard grid, and the fitting result is shown in Figure 6. The coordinates of the grid's internal corners in the pixel coordinate system were generated, and a depth-first search algorithm [11] was used to traverse the grid's corner points to obtain their world coordinates. Using the obtained coordinate pairs, the RANSAC algorithm [12] was applied to calculate the perspective transformation matrix between the front view and BEV (bird's eye view) images. The front view image was then transformed into the BEV using the inverse perspective transformation.

4. IPM Result

The bird's-eye view image obtained through the internal and external parameter transformation after calibration using this method is shown in Figure 7.
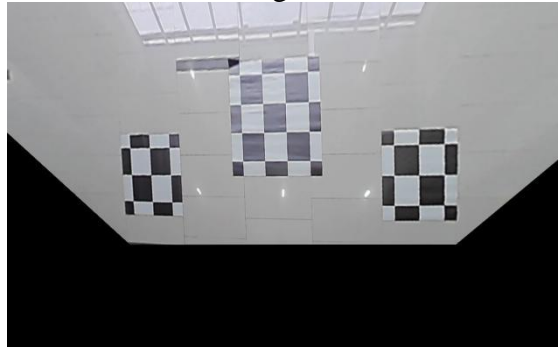

Figure 7: Bev image based on external parameter calibration.

5. Conclusion

This method is based on the adaptive recognition of multiple chessboard schemes, which does not require prior data and can complete camera extrinsic calibration with only one image, greatly improving the calibration efficiency. Figure 8 shows the comparison between this method and the OpenCV method [13] under different lighting conditions and scenes. It can be seen that the corner detection effect of this calibration method is more robust under complex lighting conditions.
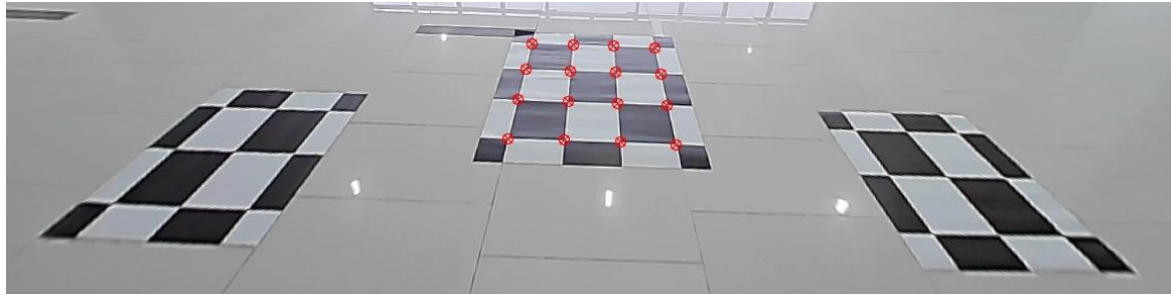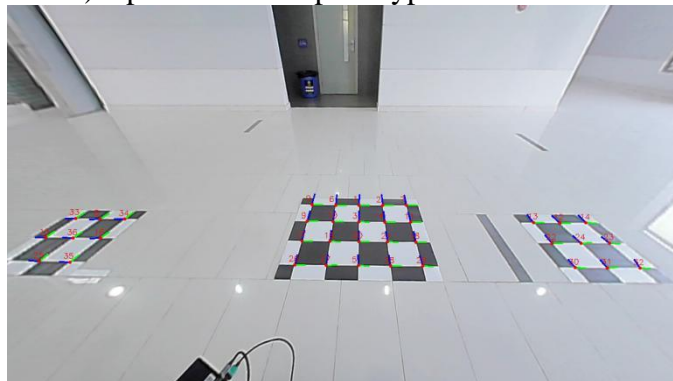
a) OpenCV Corner prototype in scenarios 1
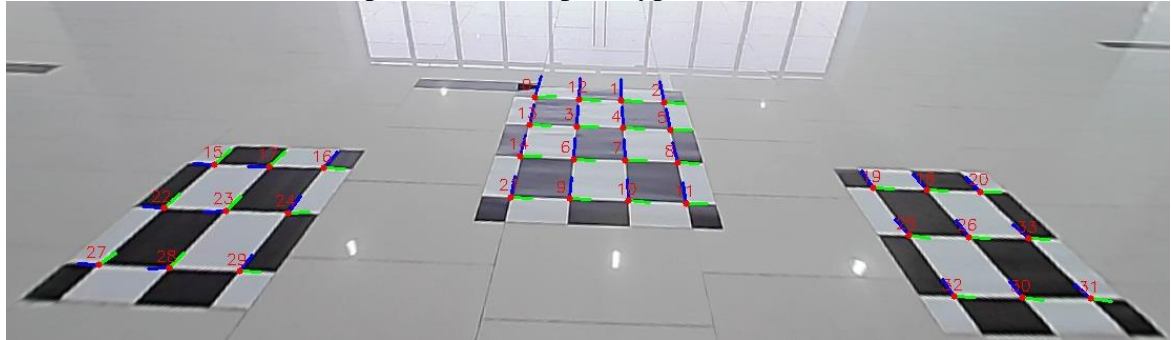
b) Our method in scenarios 1

c) OpenCV corner prototype in scenarios 2

d) Our method in scenarios 2

Figure 8: Our method is compared with OpenCV corner detection in two scenarios

References

[1] Li H., Yang M. (2008). Camera Distortion Parameter Calibration Based on Nonlinear Inverse Perspective Transformation. Journal of Shanghai Jiaotong University, (10), 1736-1739.
[2] Ha JE. Automatic detection of chessboard and its applications [J]. Optical Engineering, 2009, 48(6):067205-067205-

8.

[3] Kannal J, Brandt S S. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence-PAMI, 2006, 28(8):1335-1340.

[4] Scaramuzza Davide, Agostino Martinelli, and Roland Siegwart. "A flexible technique for accurate omnidirectional camera calibration and structure from motion." Fourth IEEE International Conference on Computer Vision Systems (ICVS'06). IEEE, 2006.

[5] Harris Chris, and Mike Stephens. "A combined corner and edge detector." Alvey vision conference. Vol. 15. No. 50. 1988.

[6] Shi Jianbo. "Good features to track." 1994 Proceedings of IEEE conference on computer vision and pattern recognition. IEEE, 1994.

[7] Geiger Andreas, et al. "Automatic camera and range sensor calibration using a single shot." 2012 IEEE international conference on robotics and automation. IEEE, 2012.

[8] Devernay Frál éric. A non-maxima suppression method for edge detection with sub-pixel accuracy. Diss. INRIA, 1995.

[9] Vincent O. Rebecca, and Olusegun Folorunso. "A descriptive algorithm for sobel image edge detection." Proceedings of informing science & IT education conference (InSITE). Vol. 40. 2009.

[10] Carreira-Perpinan, Miguel A. "Gaussian mean-shift is an EM algorithm." IEEE Transactions on Pattern Analysis and Machine Intelligence 29.5 (2007): 767-776.

[11] Tarjan Robert. "Depth-first search and linear graph algorithms." SIAM journal on computing 1.2 (1972): 146-160.

[12] Derpanis Konstantinos G. "Overview of the RANSAC Algorithm." Image Rochester NY 4.1 (2010): 2-3.

[13] Liu Yu, et al. "Automatic chessboard corner detection method." IET Image Processing 10.1 (2016): 16-23.