Automatic corner matching in highly distorted images of Zhang's calibration pattern.

Miguel Alemán-Flores[1,3], Luis Alvarez[1,3], Luis Gomez[2,3], and Daniel Santana-Cedrés[1,3]

[1] Departamento de Informática y Sistemas.
[2] Departamento de Ingeniería Electrónica y Automática.
[3] CTIM : Centro de Tecnologías de la Imagen,
Universidad de Las Palmas de Gran Canaria, Spain
{maleman,lalvarez,lgomez,dsantana}@ctim.es
http://www.ctim.es

Abstract. Zhang's method is a widely used technique for camera calibration from different views of a planar calibration pattern. This pattern contains a set of squares arranged in a certain configuration. In order to calibrate the camera, the corners of the squares in the images must be matched with those in the reference model. When the images show a strong lens distortion, the usual methods to compute the corner matching fail because the corners are shifted from their expected positions. We propose a new method which automatically estimates such corner matching taking into account the lens distortion. The method is based on an automatic algorithm for lens distortion correction which allows estimating the distorted lines passing through the edges of the squares. We present some experiments to illustrate the performance of the proposed method, as well as a comparison with the usual technique proposed in a Matlab toolbox.

Keywords: Zhang's method, camera calibration, lens distortion.

1 Introduction

Camera calibration is one of the most relevant issues in Computer Vision and addresses the problem of determining the set of parameters which describe the mapping between 3D world coordinates and 2D image coordinates. Some calibration techniques do not require the use of a calibration pattern but the standard approach requires such a pattern with known metric (photogrammetric calibration). Two perspective views of a single flat pattern, or a single view of a pattern consisting of two or three planes orthogonal to each other, are needed to extract the instrinsic and extrinsic camera parameters (see [7] for a review of camera calibration methods). For low-cost or wide-angle cameras, distortion due to lenses must be incorporated into the camera model.

Zhang's method, published in the seminal paper [6], is a widely used technique for camera calibration. It makes use of a planar calibration pattern containing a set of squares arranged in a certain configuration. In this paper, we deal with the problem of automatic corner matching of Zhang's calibration pattern when images are significantly affected by lens distortion.

Radial distortion is considered as the most common type of distortion and causes barrel distortion at short focal lengths as well as pincushion distortion at long focal lengths. Radial distortion is mainly due to the imperfection of the lens and the misalignment of the optical system, and is embedded in the well-known pinhole camera model [5] by means of a distortion model.

To calibrate the camera from images of Zhang's calibration pattern, the first step consists in extracting a set of features (corners) to be matched with the calibration pattern through a homography. This step is usually performed in an automatic manner for images without significant distortion. However, when distortion is high, the usual (automatic) technique to estimate such corner matching, which is implemented in MatLab [1] tools, fails. This is due to the fact that the lens distortion produces a displacement of the expected location of the square corners in the image and, therefore, human intervention is required, which makes the calibration a tedious and time-consuming task (a typical calibration uses around 20 images, which implies hundreds of corners).

In order to cope with this problem, we propose to use automatic algorithms for the estimation of the lens distortion model, for which no human intervention is required. In [2], an efficient method to detect lines in images showing significant lens distortion is proposed. The method uses an extension of the Hough transform, which includes the radial distortion parameters, in order to detect straight lines. Once a set of straight lines have been detected, an algebraic lens distortion model with two distortion parameters [3] is applied to estimate and correct the distortion.

To summarize, the proposed technique can be divided into the following steps :

1. Extraction of the distorted lines in the image.
2. Detection of the red, green, and blue squares we use to automatically set a reference system in the image.
3. Estimation of the matching between the distorted lines in the image and the lines of the reference pattern.
4. Initial approximation of square corner matching using the intersections of the distorted lines.
5. Computation of the Harris corner detector and refinement of the location of the corners using the initial approximation obtained in the previous step .

In the next section, we present our automatic method to extract corners in highly distorted images. The results are compared with the corner matching method provided in Matlab to illustrate that our method is better suited for such cases. Additionally, the proposed method can also be applied to improve camera calibration in cases with low distortion, since including the correction of the distortion provides a larger number of corners and, therefore, a better calibration.

This paper is organized as follows: in Section 2, we present the details of the proposed method. In Section 3, we show some experiments to illustrate the performance of the method. Finally, in Section 4, we summarize our main conclusions.

2 Automatic Corner Matching

In this section, we present the different steps of the proposed method for corner matching. Basically, our method consists in extracting the distorted lines which delimit the squares in the image, associate them with those lines in the reference pattern, estimate the corners through their intersections, and refine their positions by means of Harris corner detector.

In order to estimate the distorted lines corresponding to the edges of the squares, we use the automatic method proposed in [2]. The distortion of these lines is corrected by means of the distortion model extracted using this method.

Once we have a collection of lines, they must be matched with the lines in the reference pattern. With the aim of automatically identifying them, we introduce three colored squares (red, green and blue) located in the center of the calibration pattern and arranged in an L-shaped configuration. To find their positions in the image, we transform the RGB channels in the following way:

$$\begin{cases} R_{new} = max\{\ 0\ ,\ R - max\{G, B\}\ \} \\ G_{new} = max\{\ 0\ ,\ G - max\{R, B\}\ \} \\ B_{new} = max\{\ 0\ ,\ B - max\{R, G\}\ \} \end{cases}$$

Next, we convolve the new channels with a Gaussian function to remove noise and, finally, we compute the maximum of each channel, which correspond to points located in the red, green and blue squares. In Figure 1, we illustrate this procedure.

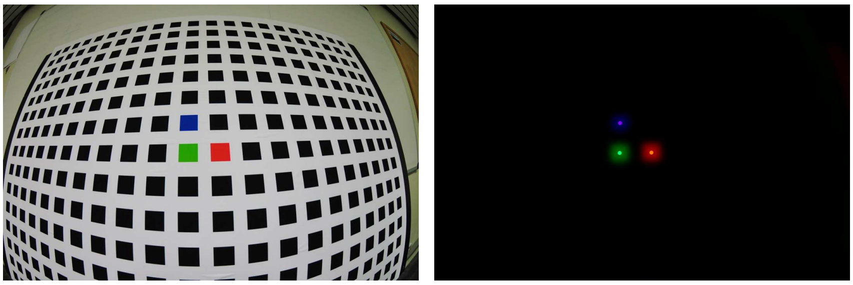


Fig. 1. Calibration pattern with the colored squares (left) and processed channels $R_{new}G_{new}B_{new}$ where the maximun of each channel is extracted (right).

Once the colored squares have been located, we determine, by proximity, which 8 lines in the image correspond to those passing through the edges of the squares. Using these line correspondences we compute a homography to project all the lines in the reference model onto the image, and associate them with the nearest line in the image, when it is feasible. As a results, we will obtain a set of line pairs (reference and image lines) which have been satisfactorily associated.

Since we need to estimate the location of the corners, we must calculate the intersections of certain pairs of lines, but first their distortion must be corrected. After correcting them and calculating the intersections, we use the inverse of the lens distortion model to extract an initial estimation of the corner position in the image. Since we know the matching between the lines in the image and those in the reference model, we can deduce the matching of the associated corner points.

To improve the initial estimation of the corner location obtained in the previous step, we use the corners obtained by the Harris method. For each corner extracted in the previous step, we find the nearest corner provided by the Harris corner detector. The final outcome of this step is a collection of Harris corners and their corresponding matching in the reference model. This information is the starting point of Zhang's calibration method.

3 Results

In this section we compare the proposed method with the results obtained using the freely available camera calibration toolbox for Matlab to show that, by using our line detection algorithm, the detection of corners is significantly improved. First, we give some details about this Matlab toolbox, which is commonly used for camera calibration.

3.1 Matlab toolbox for camera calibration

A Matlab toolbox is a downloadable specific software coded in Matlab devoted to solve a specific problem, which generally includes a user-friendly graphical user interface. In this case, the issue is camera calibration, the toolbox can be downloaded from [4], and its author is Jean-Yves Bouguet. This toolbox allows calibrating a camera using a particular implementation which is similar to Zhang's method and provides the same results.

The first step to use this toolbox, as in most methods, consists in extracting the corners for a given calibration pattern. This pattern can be any image with known metric, but usually a planar checkerboard is used (see examples in [4]). Extracting the corners with a high accuracy is a crucial task in order to get a satisfactory camera calibration. That is why this stage is extremely important and all our efforts in this proposal are focused on it. Additionally, to get an accurate camera calibration, more than one image is required, being usually necessary to handle around 20 images to cover some camera perspectives and tackle lens distortion. Because of that, corner extraction must be performed in a fast and automatic way.

For the matching process, some information about the metric of the pattern is required. In this case, the user must provide the sizes of each square along the

$X - Y$ directions to facilitate the identification of the corners, although some default values can be used. Furthermore, the user must select a region to extract the desired information (corners). It is recommended that this area be as large as possible to obtain an accurate calibration. The size of the selected area is also limited by the distortion because, for cases showing a severe distortion, the corner extraction is not well performed and the extraction must be reduced to a small selected area. An example for an image with strong radial distortion is depicted in Fig. 2. When the selected area is small (as in this case), user intervention is not usually required. However, for images with a strong radial distortion, the user's intervention is strongly required, even if the area is small, as appreciated in Fig. 2 (top image). Therefore, the user must either select a reduced area, which would provide a poor camera calibration solution, or help the program improve the corner location by providing an estimation of a single distortion parameter, which does not guarantee a solution. Moreover, this is a tedious method, even for a single image, and quite hard to apply for a set of images. Correcting the distortion by guessing a suitable distortion parameter when distortion is strong or the center of distortion is not located at the center of the image can be an extremely difficult task.

In Fig. 2, two examples are shown. The red crosses correspond to the final location of the corners provided by this method. In the first case (top), using the fully automatic matching, the corners are not properly identified (the red crosses do not match the corners of the squares). In the second case (bottom), a user-defined value for the radial distortion parameter has been applied, but the collection of corners is still unsatisfactory (although the matching has notably been improved, some corners are not correctly matched). It is important to remark that, using these matchings for camera calibration, an inaccurate result is expected.

3.2 Results using the proposed method

In Fig. 3, we show the results obtained by applying the proposed method. As observed, we are dealing with a highly distorted image. The detected lines are drawn using different colors. Note that all visible lines are correctly detected. The corners estimated by mapping the corners in the pattern using the homography and refining their positions with the Harris corner detector are depicted with crosses. Harris corners with no matching in the reference model are drawn using a red 45 degree oriented cross.

In view of these results, our proposal is able to detect most of the corners within the image (note extreme positions at edges with severe radial distortion) without demanding human intervention.

Due to the fact that the image in Fig. 3 shows a severe distortion, when using the MatLab toolbox, a small area for corner extraction was selected (the maximum number of corners to detect is $4 \times 6 \times 8 = 192$). In larger areas, we observed that the MatLab toolbox did not work properly. However, by applying our proposal, the whole image is used for the corner extraction and, therefore, the number of corners extracted is much larger (1016). This outperformance of

Method	Number of Corners	Computational Cost
MatLab toolbox	169	User dependent
Our approach	1016	16.82 secs.

Table 1. Number of detected corners properly matched and computational cost using the MatLab toolbox (fig. 2) and our approach (fig. 3).

the proposed method can be evaluated through the number of corners. In table 1 we show some comparison results.

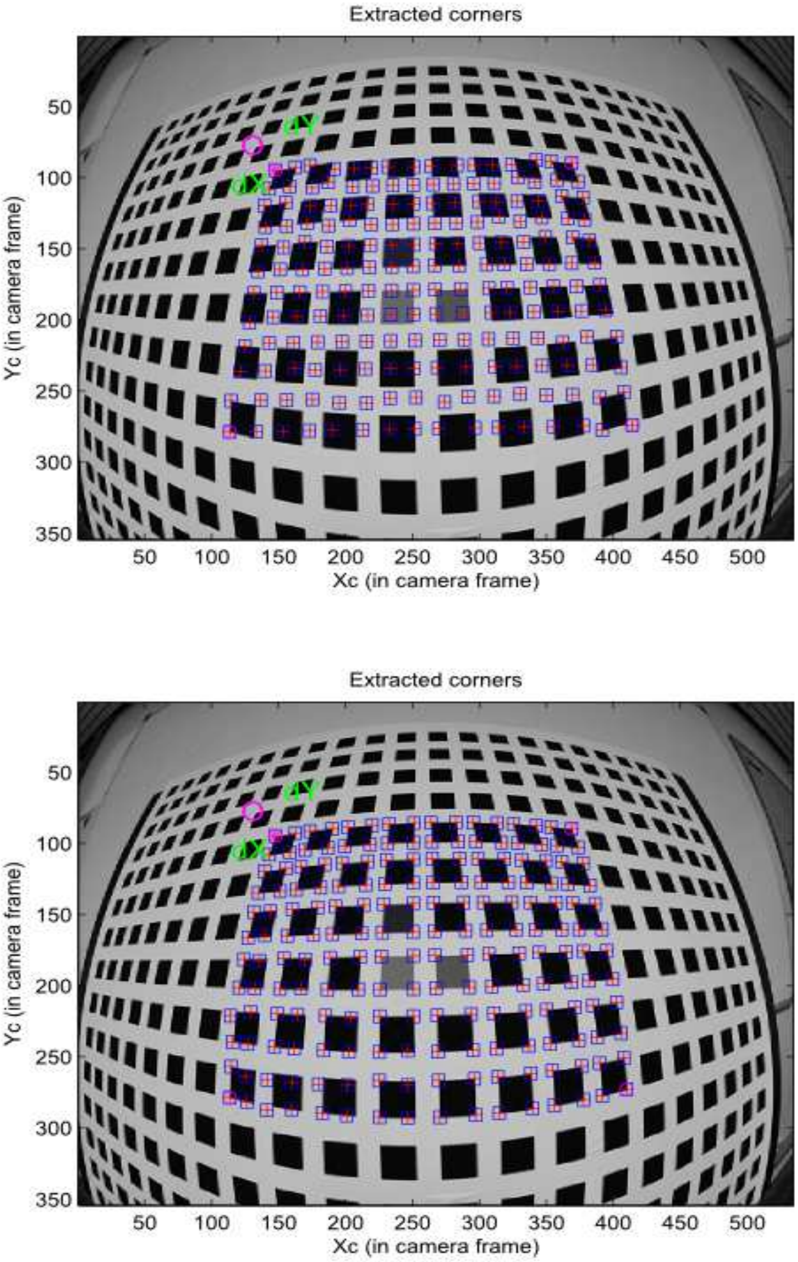
4 Conclusion

Automatic corner matching between views of Zhang's calibration pattern and the reference model is a difficult task when the images show significant lens distortion. In particular, standard methods to match square corners fail and tedious user intervention is required to fix such matching. In this paper, we propose a fully automatic corner matching procedure with no user intervention. The method is based on an automatic algorithm to correct lens distortion, which allows us to estimate the distorted lines passing through the square edges. An initial estimation of the square corners is obtained from the intersection of such lines. Afterward, the corner location is improved using the Harris corner detector. We present some experiments to illustrate the performance of the proposed method. When the image is highly distorted, our method automatically detects a quite larger number of corner matchings than the method proposed in the MatLab toolbox (even after the user's interaction). This results in a much more complete and accurate information for camera calibration.

Acknowledgement
This research has partially been supported by the MICINN project reference MTM2010-17615 (Ministerio de Ciencia e Innovación. Spain).

References

1. www.mathworks.com/products/matlab
2. Alemán-Flores, M., Alvarez, L., Gomez, L., Santana-Cedrés, D.: Line detection in images showing significant lens distortion and application to distortion correction. Pattern Recognition Letters 36, 261–271 (2014)
3. Alvarez, L., Gomez, L., Sendra, R.: An algebraic approach to lens distortion by line rectification. Journal of Mathematical Imaging and Vision 39(1), 36–50 (2008)
4. Bouguet, J.: www.vision.caltech.edu/bouguetj/
5. Faugeras, O.: Three-dimensional computer vision. MIT Press (1993)
6. Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(11), 1330–1334 (2000)
7. Zhang, Z.: Camera Calibration. Emerging Topics in Computer Vision, Prentice Hall Professional Technical Reference (2004)



Extracted corners

Extracted corners

Fig. 2. Corners provided by the Matlab toolbox, without user intervention (top), and with user intervention to fix a distortion parameter (k= -0.4) (bottom).

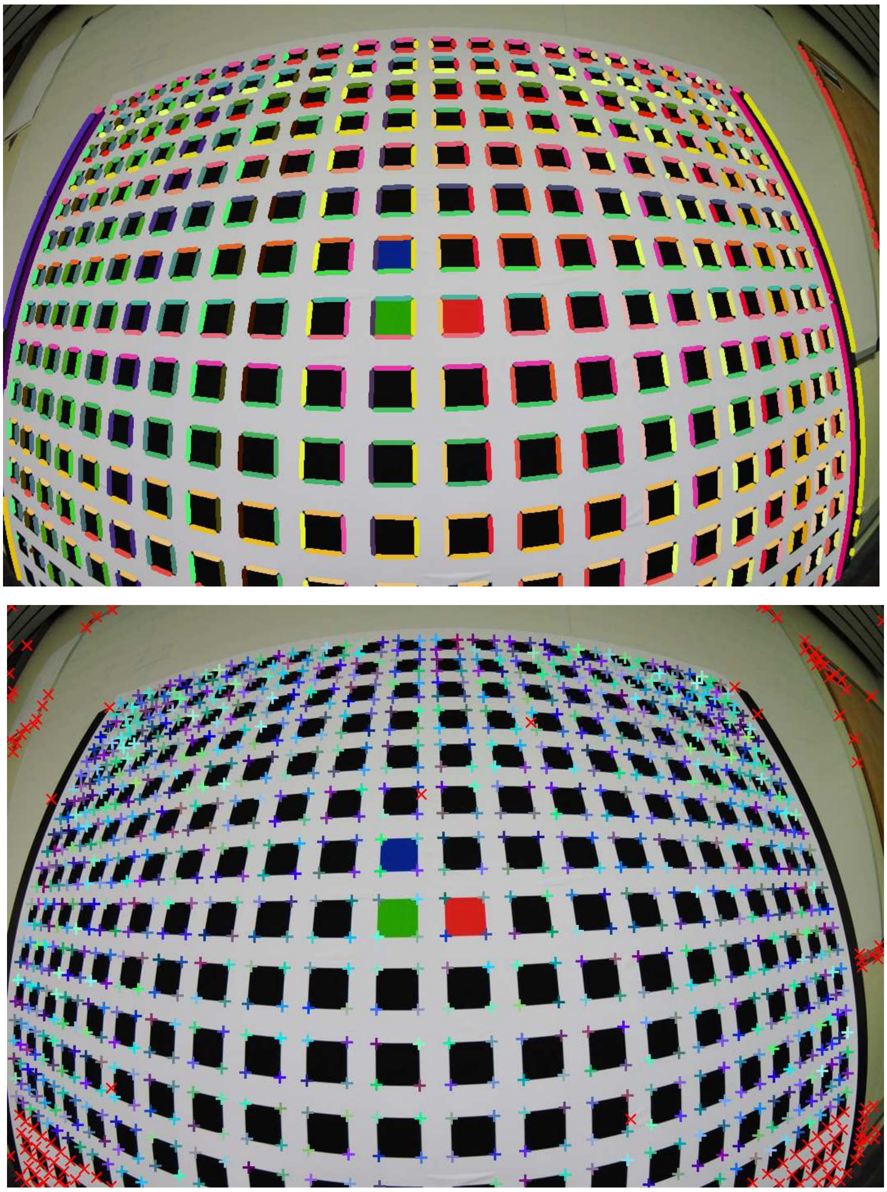


Fig. 3. Corner detection using the proposed method: automatic distorted line detection (top), and final Harris corner detection (bottom). Corners with no matching in the reference model are drawn in red using a 45 degree oriented cross.