A New Chessboard Corner Detection Algorithm with Simple Thresholding

2 authors, including:

Zhang Qi
Huazhong University of Science and Technology

2 PUBLICATIONS   **2** CITATIONS

A New Chessboard Corner Detection Algorithm with Simple Thresholding

Qi Zhang and Caihua Xiong

Institute of Rehabilitation and Medical Robotics, State Key Lab of Digital
Manufacturing Equipment and Technology, Huazhong University of Science and
Technology, Wuhan, Hubei 430074, People's Republic of China
`edisonzhangqi@hust.edu.cn, chxiong@hust.edu.cn`

Abstract. Chessboard detection and corner extraction are imperative
during the camera calibration process, which is a fundamental work in
computer vision. This paper describes a new chessboard corner detection
algorithm using the amplitude spectrum feature of circular sampling at
each point as corner response function. A simple thresholding technique
is that only the points near a chessboard corner will get positive cor-
ner response. However, image noise will bring a lot of false positives.
The distribution of false positive response with respect to image noise is
given, this distribution also has a property like the three-sigma rule in
Gaussian distribution, that can be used as thresholding rule. The only
parameter needed is the standard deviation of image noise, which can
be measured using some patches of the image for once. Experiment is
presented showing the efficiency of the proposed method against noise,
compared with existing algorithm under simulated image.

Keywords: Chessboard Detection, Discrete Fourier Transform, Corner
Response, Difference of Rayleigh Distribution.

1 Introduction

Chessboard pattern is commonly used in camera calibration, which is a neces-
sary step in many computer vision tasks. Since such pattern can provide simple
and clear corners with known geometrical parameters, it is also suitable for 3D
pose estimation and localization in robot vision. Thus extracting chessboard
corners fast, accurately and automatically is still a significant challenge. The
available methods for chessboard corner detection often require hand-tuning of
parameters under different image conditions to filter out false corners. Although
some methods give empirical parameters for corner filtering, they may still fail
due to different noise level or light condition. Therefore, methods need manual
intervention prohibit automated use.

In this paper, we will present a chessboard corner detection method which
only need a priori knowledge of the image noise variance. This can be estimated
using a bright and uniform image patch, e.g. an image of a white paper or
some bright squares of the chessboard, taken from the same camera with fixed

exposure gain. This prerequisite is reasonable, and easy to operate for only one time. Experiment with simulated images and real images shows the efficiency and feasibility of the proposed method.

Following a review of published chessboard corner detection methods in Section 2, in Section 3 we use circular sampling strategy proposed in [1] and rearrange the sampled points into a 1D vector. We discuss the time-frequency characteristic of the sampled vector at different location on the chessboard. Then we present a chessboard corner response function. In Section 4, we reveal that image noise will cause false positive response, and we show the distribution of undesired corner response caused by image noise. Surprisingly, this distribution is similar to Gaussian distribution and has a property like the three-sigma rule, which can be used to choose a threshold for false positive filtering. Experiment is conducted in Section 5, we present both simulated and real-world image to verify the proposed method's efficiency and feasibility. This paper ends with some conclusions and extensions.

2   Related Works

Several methods have been published for chessboard corner detection. These methods usually consist two main steps, locate all corners roughly and select the chessboard corners. General corner detector, such as the Harris [7], SUSAN [12], or FAST [11] features, can be used to locate corners roughly. Harris corner detector uses corner metric threshold to filter out most noise points, and it is especially suitable for L corner. However, the chessboard corner is an X corner, which will looks like two L corners when the image is blurred, thus applying Harris corner detector to a blurred chessboard image may get two corners for each chessboard vertex. FAST use an intensity difference threshold and the number of contiguous disparate pixels to classify a point as or not corner. If the intensity difference threshold is too small, there will be more than one FAST features near a chessboard corner, and if the value is too large, some chessboard corners may be missed due to nonuniform luminance. SUSAN detector only use non-maximal suppression to select corners, but it should be pointed out that SUSAN will treat a chessboard corner without any rotation or distortion as an edge point.

Although general corner detector always need proper threshold to get all chessboards corners, several methods are based on this detector to roughly locate all corners and use different techniques to select chessboard corners. Wang et al. [14] uses Harris detector to select all corners and apply two rotating orthogonal masks to filter out non X corners, then using line intersection to select chessboard internal points. This method will be affected by lens distortion. Kassir et al. [9] use multi-scale Harris detector and improved filtering method. Ha [6]. use Lucas-Kanade feature detector to track corners and use a radial accumulator to select the chessboard corner. These two methods always need a lot of computing resource. Zhu et al. [17] propose an improved SUSAN detector with predefined threshold. He et al. [8] adopt Hessian feature to detect points, and propose an adaptive thresholding technique to extract X corners, then apply a circular

template to eliminate blobs. However, This method also use some predefined threshold.

Besides general corner detector, several local feature based methods have been proposed. Yu et al. [15] insert 5 double-triangle patterns as reference into a chessboard and detect them using rotated templates, then recover chessboard corners using the double-triangle patterns. The performance of their method depends on the success of detecting the double-triangle patterns. Sun et al. [13] rearrange points in a square window centered at a point into several 1D vectors called layers, apply binarization and special morphology operation to select chessboard corners. This method will produce false corners from noise and is rather slow. Zhao et al. [16] divide a window centered at a point into four quadrants, the summed intensity difference of adjacent quadrant is larger than the given threshold may contain a chessboard corner. This method needs hand-tuning threshold and may have a lot of false corners. Geiger et al. [5] use four pairs of convolution kernels to produce a corner response map, after non-maximal suppression for candidates and apply windowed gradient histogram to each candidate, they generate a new template for that chessboard. This method can detect multiple chessboard, but it needs several convolutions, which is quite slow. Donné et al. [3] propose a method based on convolution neural networks, this can be explained as using several weighted convolution kernels to detect chessboard. This may be a generalized method for detecting every single object, but it requires a lot of computing power. Bennett and Lasenby [1] advance a simple, fast and efficient corner response function using only circular points around sampling center. The main drawback is that the corner response function will trigger a lot of false corners, although thresholding will eliminate those false positives, it is nontrivial to give a threshold. Bok et al. [2] also use the features of circular sampled points to discard candidates. This method include fraction coordinates and needs a lot of interpolate intensities which will slow down the computation.

Accurate chessboard corner can also be obtained or selected by line intersection. de la Escalera and Maria [4] proposed a method which adopt Hough line transform twice to obtain edge lines on chessboard. This method also suffers from lens distortion. A complex technique, called ROCHADE proposed by Placht et al. [10], combines both point and edge features. This method is stable but very slow. This paper aims to offer a fast and minimal hand-tuning approach for chessboard corner detection.

3   Chessboard Corner Response Function

Although conventional corner detector like Harris and FAST can be used to extract chessboard corner roughly. It still needs further operation to exclude outliers. Most published filtering methods take advantage of the local characteristic of a chessboard corner, like a circular boundary centered at a corner [1] [2] [13]. The main benefit of circular sampling is rotationally invariant, also it can carry a lot of information to classify corners when samples enough points.

Unlike most methods mentioned above, we use circular sampling to roughly extract points, which have similar local feature to chessboard corner, then filter out false corners. The characteristic of circular sampled data should be discussed first. Sampler in the following paragraphs refers to the circular sampler. Sampling center means the center of circular sampler.

Consider an ideal chessboard image with no noise, good light condition and only three gray levels which appears as a binary image. Apply a circular sampler with resolution one point per degree to the ideal image, three main cases will appear as listed below.

– When sampler moves to a chessboard corner, see Fig. 1a, the black circle indicates the sampling path. Suppose start sampling counter clockwise from the most right point, and put sampled intensities into a 1D vector in the sampling order. The 1D vector, also a 1D signal, will contain two cycles of square wave whose period is half the vector length, see Fig. 1b. If the sampling center is near a chessboard corner, the sampled vector will be slightly different from the one sampled at a corner, it will wrap two square waves with different period.
– As sampler goes to a block edge on the chessboard, see Fig. 1c, if apply the same sampling order mentioned above, we will obtain a wide, one cycle square wave whose period is the vector length, see Fig. 1d. When the sampler is near the block edge, the sampled vector is also square wave with a narrower or wider rectangle.
– While the sampler is on a flat region, e.g. a bright block on chessboard, the sampled vector will appear as straight line.

(a) Sampling on a corner

(b) Sampled vector is a square wave with two cycles

(c) Sampling on a block edge

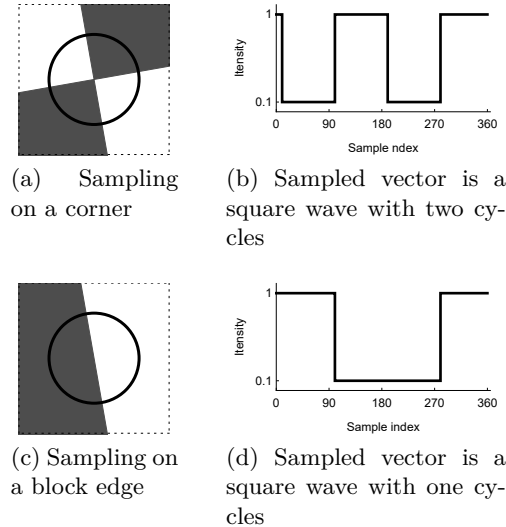(d) Sampled vector is a square wave with one cycles

Fig. 1: Different sampling result

It is obvious that the sampled 1D vector, which behaves as a two cycle square wave, indicate the sampling center is on a chessboard corner. Thus, the chessboard corner detection can be implemented by finding the two cycle square wave in the sampled vectors, which can be solve by template matching. However, the template matching may fail due to the circular shifting of the sampled vector cause by chessboard rotation. So a circular shifting template matching methods is needed, but this method is always time-consuming. Nevertheless, we realize that the Fourier transform of the sampled vector is a reasonable solution to the matching problem. One of the basic properties of Fourier transform is time shifting does not change the magnitude of any frequency component, which means that the magnitude of a frequency component in the amplitude spectrum of the sampled vector will not change under circular shifting cause by chessboard rotation.

For computational efficiency, we adopt the discrete circular sampling strategy proposed by [1]. The sampled 1D vector is denoted as $\mathbf{x} = (x_1, x_2, \ldots, x_M)^T$, where T means transpose, x_i is sampled intensity. The sampling point (u_i, v_i) centered at (u_c, v_c), is defined as

$$u_i = \lfloor r \cos\left(2\pi \tfrac{i}{M}\right) + u_c \rfloor \tag{1}$$

$$v_i = \lfloor r \sin\left(2\pi \tfrac{i}{M}\right) + v_c \rfloor \tag{2}$$

where $i = 1, 2, \ldots, M$. M is the number of samples, r is the radius of the sampler, and $\lfloor x \rfloor$ return the rounded value of x, which ensures generating integer coordinates. For a blurred image, r may need to be larger. In our case, $r = 5$ pixels. The discrete Fourier transform of $\mathbf{x}$ is given as $\mathbf{f} = (f_0, f_1, \ldots, f_{M-1})^T$, where f_k encodes both amplitude and phase of a complex sinusoid which frequency is k cycles per M samples. $|f_k|/M$ is the amplitude of the sinusoid with frequency k, a large $|f_k|/M$ means the source signal mainly depend on the sinusoid with frequency k. This is the key idea of our method.

While dealing with an ideal image mentioned above, see Fig. 2a, apply the circular sampler to each pixel in the ideal image, and perform discrete Fourier transform for each sampled vector, it could be found out that the discrete amplitude spectrum of the sampled vector behaves distinctively at different position. Using only too frequency component, f_1 and f_2, we classify these behaviors into four main cases as listed below.

- When sampling at a chessboard corner, the sampled vector is a two cycles square wave, it is easy to verify that $|f_1| = 0$ and $|f_2|$ is relative big, see the center of Fig. 2b and Fig. 2c. The image center contains a chessboard corner.
- $|f_1| > |f_2|$ when the sampling center is near a block edge, see Fig. 2b bright vertical strip at the center.
- $|f_1| = |f_2| = 0$ when the sampling center is on a flat area, such as bright block, check the for corner of Fig. 2b and Fig. 2c.
- relation between $|f_1|$ and $|f_2|$ is complicated. $0 < |f_1| < |f_2|$ while the sampling center is near a corner, see the points near the center of Fig. 2d. When sampling center moves toward four corners or along a block edge, it can be seen that $|f_2| < |f_1|$

(a) Ideal chessboard image patch, including transition pixels between bright and dark block

(b) $|f_1|$ of each vector sampled at every pixel

(c) $|f_2|$ of each vector sampled at every pixel
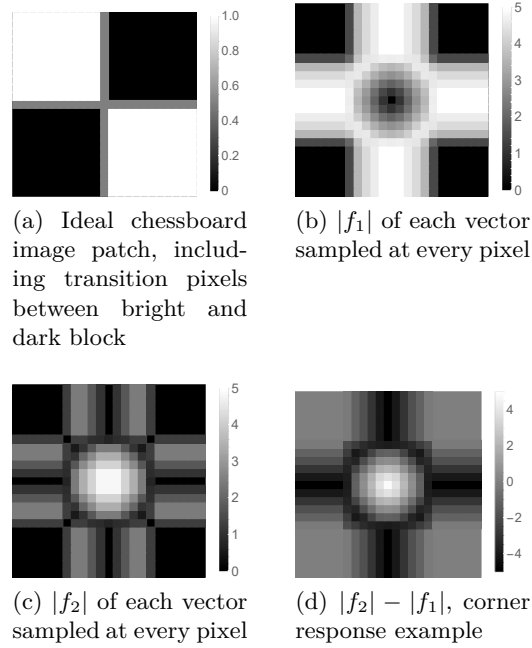

(d) $|f_2| - |f_1|$, corner response example

Fig. 2: Amplitude spectrum analysis of ideal chessboard image

It is obvious that $|f_1| < |f_2|$ only if sampling center is near the chessboard. So we define a chessboard corner response function as

$$R = |f_2| - |f_1| \tag{3}$$

One simple criteria for the point may be a chessboard corner is $R > 0$. And local maximal of R indicate that it is a chessboard corner. However, $R > 0$ is a necessary but not sufficient condition for a point to be a chessboard corner. Consider the real-world image, there always be noise, which can cause $R > 0$ even on a uniform region such as bright block on the chessboard. So it is necessary to eliminate the influence of noise.

4 Corner Response Filtering

A straightforward way to filter out false positives caused by noise is image filtering, such as Gaussian smoothing. But this can not eliminate all noise and will blur the corners, which will affect corner detection mention in Sec. 3. Although image noise may cause false positives, the corner response with respect to noise has a limited range, which can be used as a threshold to filter out false positives. Assume the images have additive Gaussian noise with standard deviation σ, denote a random variable x obeys Gaussian distribution as $x \sim \mathcal{N}(\mu, \sigma^2)$, where

μ is the expectation and σ represents standard deviation. The discrete Fourier transform of the sampled vector $\mathbf{x}$ is defined as

$$f_k = \sum_{m=0}^{M-1} x_m e^{-i2\pi km/M} \tag{4}$$

where M is length of vector $\mathbf{x}$, k is the frequency of sinusoid, substituting Euler's formula in Eq. 4 we get

$$f_k = \sum_{m=0}^{M-1} x_m \cos\left(-2\pi k \frac{m}{M}\right) + i x_m \sin\left(-2\pi k \frac{m}{M}\right) \tag{5}$$

the real and imaginary part of f_k is denoted as

$$R_k = \sum_{m=0}^{M-1} x_m \cos\left(-2\pi k \frac{m}{M}\right) \tag{6}$$

$$I_k = \sum_{m=0}^{M-1} x_m \sin\left(-2\pi k \frac{m}{M}\right) \tag{7}$$

It is worth to note that

$$\sum_{m=0}^{M-1} \cos\left(-2\pi k \frac{m}{M}\right) = \sum_{m=0}^{M-1} \sin\left(-2\pi k \frac{m}{M}\right) = 0 \tag{8}$$

$$\sum_{m=0}^{M-1} \cos^2\left(-2\pi k \frac{m}{M}\right) = \sum_{m=0}^{M-1} \sin^2\left(-2\pi k \frac{m}{M}\right) = \frac{M}{2} \tag{9}$$

Suppose the sampling center is on a uniform region with noise mention above, then $\mathbf{x}$ is a random vector, its elements obey Gaussian distribution, denoted as $x_i \sim \mathcal{N}(\mu, \sigma^2)$, where μ is intensity of the uniform region, σ is the standard deviation of image noise. It can be derived that $R_k \sim \mathcal{N}(0, \sigma^2 M/2)$, $I_k \sim \mathcal{N}(0, \sigma^2 M/2)$. Both R_k and I_k obey Gaussian distribution. The magnitude of frequency k component is given by

$$|f_k| = \sqrt{R_k^2 + I_k^2} \tag{10}$$

Due to the fact that R_k and I_k both obey the same Gaussian distribution, $|f_k|$ obeys Rayleigh distribution, it's probability density function is given as

$$p(z) = \frac{z}{\tau^2} e^{-\frac{z^2}{2\tau^2}}, \quad x \geq 0 \tag{11}$$

where $z = |f_k|$ in our case, $\tau = \sigma^2 M/2$ is a parameter of Rayleigh distribution, it is also the standard deviation of R_k and I_k. Then the corner response function defined above, see Eq. 3, obeys distribution of difference of two variables

that obey Rayleigh distribution. We call this distribution as Rayleigh difference distribution, its probability density function is derived as

$$\gamma(y) = \int_{-\infty}^{+\infty} p(y)p(t-y)\,\mathrm{d}\,t \tag{12}$$

$$= \frac{\mathrm{e}^{-\frac{y^2}{2\tau^2}}}{8\tau^3}\left(2|y|\tau + \mathrm{e}^{\frac{y^2}{4\tau^2}}\sqrt{\pi}\left(2\tau^2 - y^2\right)\mathrm{Erfc}\left(\frac{|y|}{2\tau}\right)\right) \tag{13}$$

where $y = |f_2| - |f_1|$ in this case, while $|f_1|$ and $|f_2|$ are random variables obey a Rayleigh distribution with parameter τ as defined in Eq. 11, $\mathrm{Erfc}(y)$ is the complementary error function defined as

$$\mathrm{Erfc}(y) = \frac{2}{\sqrt{\pi}}\int_y^\infty \mathrm{e}^{-t^2}\,\mathrm{d}\,t \tag{14}$$

This distribution is similar to Gaussian distribution, see Fig. 3, and it also has the three-sigma rule, however the sigma mentioned here is not the standard deviation as the one in Gaussian distribution, it is a parameter of the distribution, which is denoted as τ in our case mentioned above. Thus, we call this property as the three-tau rule. Define the cumulative distribution function of Rayleigh difference distribution as

$$P(y) = \int_{-\infty}^y \gamma(t)\,\mathrm{d}\,t \tag{15}$$

where $\gamma(t)$ is defined as Eq. 13.

Using numerical calculation, we can get that $P(3\tau) = 0.99919456$. This means that if we use 3τ as the threshold to filter out false positives, the ratio of filtered out point to all false positives is about 99.919456%. It need to note that for a VGA image, there are 640×480 points, if we choose the 3τ as threshold, there will be about 247 points that may not be filtered out. However, we can use 5τ as a threshold because of $P(5\tau) = 0.99999988$, which means for VGA resolution, all the false positives will be eliminated.

It is worth noting that when the circular sampler move to a strip like place, such as dark wire on white background, the sampled vector will also behave as the one sampled at chessboard corner. In this case, we adopt the ChESS in [1] to solve this problem.

5 Experiment

In our implementation, we use a circular sampler with radius 5 pixel and 16 sample points as mentioned in Sec. 3. We choose 5τ as the threshold for false positive filtering, where $\tau = 2\sqrt{2}\sigma$, and σ is the standard deviation of image noise. Our method is compared with the ChESS algorithm proposed in [1] under simulated image to demonstrate its feasibility and efficiency.

We generate a simulated grayscale image which is composed of 6 by 6 squares. Each square is 30 pixels wide. The intensities of the bright and dark block are
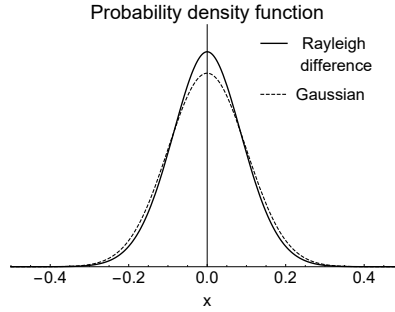
Fig. 3: Rayleigh difference distribution and Gaussian distribution, the standard deviation is 0.1

0.8 and 0.2, while the intensity range is $[0, 1]$. Between bright and dark block we add transition pixels, whose intensity is 0.5. To demonstrate the feasibility of eliminating false positives, we add Gaussian noise to the image while the noise standard deviation σ is 0.05. Then we apply our method and ChESS to this image.

Figure 4 shows the corner response of our method and ChESS. The image is cropped for recognizable, and it only contains 4 true chessboard corners. The image is not smoothed and the corner response maps are not filtered. Notice that the result of ChESS, see Fig. 4c, contains a lot of local maximum response caused by image noise. Although these false positives can be eliminated by setting a proper threshold, it will need hand-tuning, and may miss the true chessboard corner. However, the corner response of our method is quite clean, see Fig. 4b, only points near the corner have positive response as analyzed in Sec. 3. Small local maximum caused by noise are all filter out using the technique proposed in Sec. 4.

(a) Cropped simulate image

(b) Corner response obtained by our method, only points near the real corner have positive response

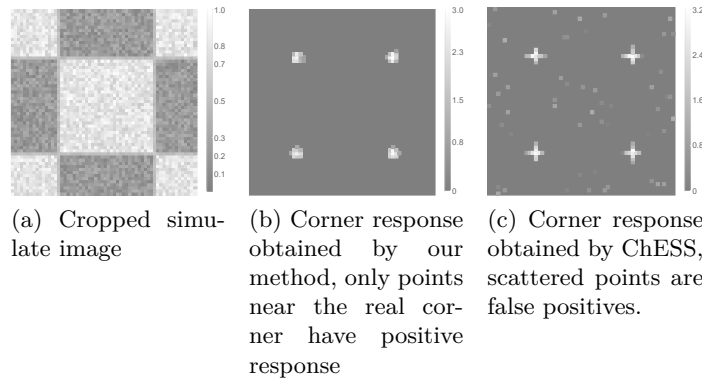(c) Corner response obtained by ChESS, scattered points are false positives.

Fig. 4: Corner response comparison between ChESS and our method, the image is cropped for recognizable

Figure 5 shows the detection result of our method under real-world, poor-lit image. After calculating corner response, we use non-maximum suppression to get the points with the highest local response to represent chessboard corners. Standard deviation σ of image noise is about 0.02, which is measured by choose some black block on the chessboard from this image. It is important to notice that some false positives still exist, see the red circle at the upper left corner of the image. These false positives are not caused by image noise. The main reason leads to this result is that we only use one circular sampler and the first simple threshold strategy proposed in Sec. 3 is a necessary but not sufficient rule. Thus, those points which have same circular sampling feature as the chessboard corner but are even not corners will be detected by our method.
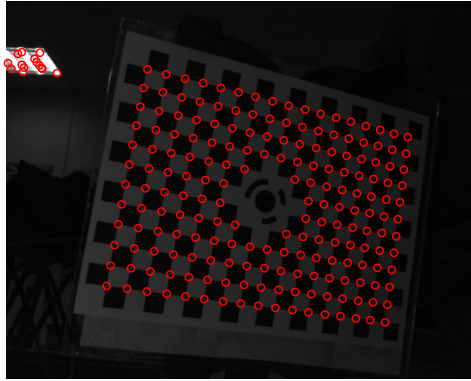

Fig. 5: Test our method using real-world image, red circles indicate chessboard corner, the standard deviation of image noise is about 0.02 which is estimated using dark blocks

6   Conclusion

In this paper, we have proposed a chessboard corner detection algorithm based the magnitude spectrum characteristic of the circular sampled data. Given the standard deviation of image noise, we derived the distribution of the corner response with respect to image noise, which we name it as Rayleigh difference distribution. This distribution is similar to Gaussian distribution, and it has a property like the three-sigma rule in Gaussian distribution, which we call it the three-tau rule. However, we choose five taus to be the threshold due to fact that the probability covered in five taus is nearly 1. Experiment showed our method can efficiently eliminate false positives caused by noise, and is also robust against poor light condition.

Due to the strip image may have the same circular sampling characteristic as the chessboard corner. Our method can be used to extract strip center, such as the center line of a black wire on a bright background.

7   Acknowledgment

References

1. Bennett, S. ,Lasenby, J. :ChESS – Quick and Robust Detection of Chess-Board Features. in: Computer Vision and Image Understanding. pp. 197-210 (2014).
2. Bok, Y. ,Ha, H. ,Kweon, I. S. :Automated Checkerboard Detection and Indexing Using Circular Boundaries. in: Pattern Recognition Letters. pp. 66–72 (2016).
3. Donné, S. et al. :MATE: Machine Learning for Adaptive Calibration Template Detection. in: Sensors. pp. 1858 (2016).
4. de la Escalera, A. ,Armingol, J. M. :Automatic Chessboard Detection for Intrinsic and Extrinsic Camera Parameter Calibration. in: Sensors. pp. 2027–2044 (2010).
5. Geiger, A. et al. :Automatic camera and range sensor calibration using a single shot. in: 2012 IEEE International Conference on Robotics and Automation. pp. 3936–3943 (2012).
6. Ha, J.-E. :Automatic Detection of Chessboard and its Applications. in: Optical Engineering. pp. 067205–067205–8 (2009).
7. Harris, C. ,Stephens, M. :A combined corner and edge detector. in: In Proc. of Fourth Alvey Vision Conference. pp. 147–151 (1988).
8. He, J. et al. :Automatic Corner Detection and Localization for Camera Calibration. in: IEEE 2011 10th International Conference on Electronic Measurement & Instruments. pp. 312–315 (2011).
9. Kassir, A. ,Peynot, T. :Reliable Automatic Camera-Laser Calibration. in: Proceedings of the 2010 Australasian Conference on Robotics & Automation. (2010).
10. Placht, S. et al. :ROCHADE: Robust Checkerboard Advanced Detection for Camera Calibration. in: Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV. pp. 766–779 (2014).
11. Rosten, E. ,Porter, R. ,Drummond, T. :Faster and Better: A Machine Learning Approach to Corner Detection. in: IEEE Transactions on Pattern Analysis and Machine Intelligence. pp. 105-119 (2010).
12. Smith, S. M. ,Brady, J. M. :SUSAN—A New Approach to Low Level Image Processing. in: International Journal of Computer Vision. pp. 45–78 (1997).
13. Sun, W. et al. :Robust Checkerboard Recognition for Efficient Nonplanar Geometry Registration in Projector-camera Systems. in: Proceedings of the 5th ACM/IEEE International Workshop on Projector camera systems. pp. 1–7 (2008).

12   Qi Zhang et al.

14. Wang, Z. et al. :Recognition and Location of the Internal Corners of Planar Checkerboard Calibration Pattern Image. in: Applied Mathematics and Computation. pp. 894–906 (2007).
15. Yu, C. ,Peng, Q. :Robust Recognition of Checkerboard Pattern for Camera Calibration. in: Optical Engineering. pp. 093201–093201–9 (2006).
16. Zhao, F. ,Wang, C. W. J. :An Automated X-corner Detection Algorithm (AXDA). in: Journal of Software. pp. 791–797 (2011).
17. Zhu, W. et al. :A Fast and Accurate Algorithm for Chessboard Corner Detection. in: Image and Signal Processing, 2009. CISP '09. 2nd International Congress on. pp. 1–5 (2009).