

Accurate Camera Calibration using Iterative Refinement of Control Points

Ankur Datta

Jun-Sik Kim

Takeo Kanade

Robotics Institute

Carnegie Mellon University

{ankurd, kimjs, tk}@cs.cmu.edu

Abstract

We describe a novel camera calibration algorithm for square, circle, and ring planar calibration patterns. An iterative refinement approach is proposed that utilizes the parameters obtained from traditional calibration algorithms as initialization to perform undistortion and unprojection of calibration images to a canonical fronto-parallel plane. This canonical plane is then used to localize the calibration pattern control points and recompute the camera parameters in an iterative refinement until convergence. Undistorting and unprojecting the calibration pattern to the canonical plane increases the accuracy of control point localization and consequently of camera calibration. We have conducted an extensive set of experiments with real and synthetic images for the square, circle and ring pattern, and the pixel reprojection errors obtained by our method are about 50% lower than those of the OpenCV Camera Calibration Toolbox. Increased accuracy of camera calibration directly leads to improvements in other applications; we demonstrate recovery of fine object structure for visual hull reconstruction, and recovery of precise epipolar geometry for stereo camera calibration.

1. Introduction

Camera calibration is an important problem in which even small improvements are beneficial for tasks such as 3D reconstruction, robot navigation, etc. Algorithms for calibrating a pinhole camera can be primarily classified into two categories; those that require objects with known 3D geometry [5], and those that use self-calibration, including the use of planar calibration patterns [18, 22, 15, 17]. Due to their ease of use, calibration algorithms that use planar patterns have gained widespread acceptance. In addition to the square planar pattern, circle and ring patterns have also been used [16, 7, 4, 3, 24, 21, 11, 9]. The calibration procedure typically consists of either localizing the calibration pattern control points (square corners, circle or ring centers) [18, 22, 7, 9] and then solving for the camera parameters, or using some geometric property of the pattern itself to solve for the camera parameters directly [4, 3, 24, 21, 11].

A major source of error that affects both camera calibration approaches; of either localizing the control point or using geometric properties of the pattern directly, is that the input camera calibration images are non-fronto parallel images that suffer from nonlinear distortion due to camera optics. Therefore, precise localization of control points or accurate determination of geometric properties under such conditions is a very difficult task, where even small errors may lead to imprecise camera calibration. The difficulty of localizing square control points in distorted non-fronto parallel images was noted by Zhang in [23], however, no steps were presented to rectify it. Heikkila noticed a variant of this problem where the center of the projected circle is mistaken for the projected center of the circle and proposed a refinement approach for control points that is restricted to *only* the circle pattern [7, 8] and does not extend to other planar patterns such as the square and the ring pattern.

In this paper, we advocate an iterative refinement approach for accurate localization of calibration pattern control points that is applicable to *all* planar patterns: square, circle and ring. We propose to undistort and unproject the input pattern images to canonical fronto-parallel images with no distortion; pattern control points are then localized in these canonical images. We can localize the control points with high accuracy in the canonical images because they are fronto-parallel and do not suffer from distortion effects. Once the control points have been localized, they are then used to recompute the camera calibration parameters. This process is then repeated until convergence. This iterative refinement approach can be bootstrapped using standard calibration routine like OpenCV [1, 22], which provide initial estimates for radial distortion and camera parameters. We have conducted an extensive set of experiments with real and synthetic images of square, circle, and ring calibration patterns and our results demonstrate recovery of calibration parameters with accuracy far exceeding the traditional approach as employed by OpenCV [1] (see Section 5). In addition, we also present results on two applications: visual hull reconstruction, where we show that fine object structure can be recovered from accurate calibration using the proposed approach, and stereo camera calibration, where we show that the proposed approach results in precise recovery of epipolar geometry.

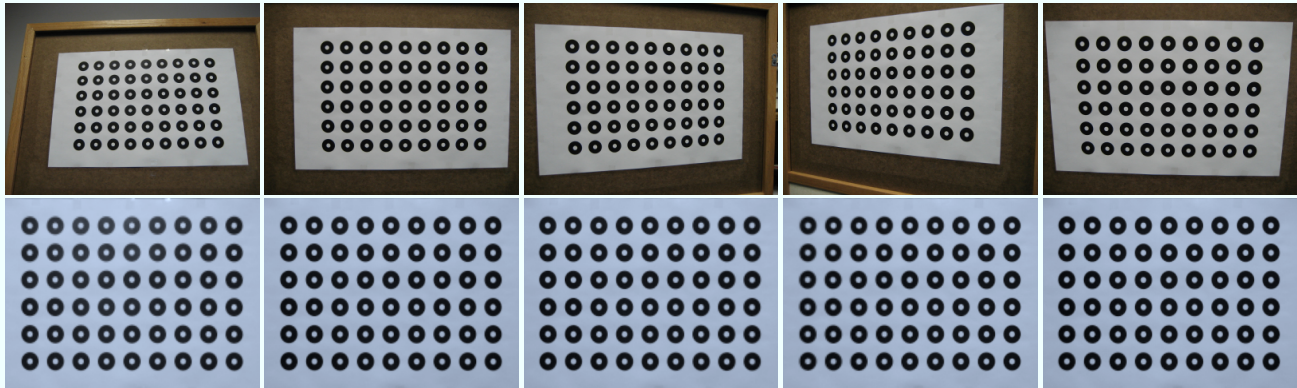


Figure 1: Top row: Input images of the ring calibration pattern. Bottom row: Input images have been undistorted and unprojected to canonical fronto-parallel images. Control points can be precisely localized in the canonical images as compared to the input images.

2. Related Work

Square control points were first used in their earliest form by Brown [2] and were made popular by the subsequent work on camera calibration by Tsai [18]. Sturm *et al.* in [15] use imaged absolute conic to recover camera intrinsic parameters using linear calibration equations. Zhang introduced an algorithm that required only a few images of a planar checkerboard pattern to compute the calibration parameters [22]. The approach consisted of an initial closed-form solution of the camera parameters, followed by a non-linear refinement using Levenberg-Marquardt [14]. Zhang’s approach to camera calibration has inspired OpenCV [1] and serves as our benchmark to compare progress in camera calibration accuracy, consequently we elaborate briefly on the details of OpenCV in section 3.

Circular control points were introduced as an alternative to the square control points for camera calibration [16, 7, 4, 3, 24, 21, 19]. Heikkila in [7] performed a minimization over the weighted sum of squared differences between the observation and the camera model using Levenberg-Marquardt [14]. Chen *et al.* in [3] estimated the extrinsic camera parameters and the focal length of the camera from a single image of two coplanar circles with arbitrary radius. Colombo *et al.* in [4] presented a system to compute camera parameters from an image of at least two circles arranged coaxially. Coaxial arrangement occurs naturally in situations under which rotational symmetry exists. Meng *et al.* in [13] proposed a calibration method similar in spirit to [22] but with a calibration pattern consisting of a circle with straight lines passing through its center. Wu *et al.* in [19] exploited the quasi-affine invariance property of two parallel circles to recover intrinsic camera parameters.

Researchers in [11, 9, 12] have employed the ring pattern for camera calibration. Kim *et al.* in [11, 12] developed al-

gebraic and geometric constraints of projected concentric circle and used these properties to compute camera parameters. A single, large concentric circle image may, however, suffer from nonlinear distortion leading to inaccurate determination of geometric properties of the concentric circle, and therefore, inaccurate camera calibration. A grid of concentric circles, on the other hand, is less susceptible to distortion effects. Jiang *et al.* in [9] presented an automatic method that constructs a sequence of points, strictly convergent to the image of the circle center from an arbitrary point on the ring pattern. These center points were then used for camera calibration using Zhang’s algorithm [22].

The primary difficulty in obtaining accurate calibration stems from the problem of working directly with the non-fronto parallel distorted input images in which precise localization of control points or accurate determination of geometric properties of the calibration pattern is a difficult task. Previous work has neglected to address this problem directly [1, 3, 9, 11]. Square control points in OpenCV [1] are localized in distorted non-fronto parallel images, which results in their imprecise localization in images. Methods that rely on the properties of the circular and ring features [3, 9, 11] can not handle the non-linear distortion by itself because it is not possible to distinguish if the difference in the feature parameters comes from the distortion or from projective effects. The goal of our work is accurate camera calibration by addressing the problem of precise localization of control points in input images. We propose an iterative refinement approach for localizing the control points. We undistort and unproject the input images to a canonical fronto-parallel image, which is then used to precisely localize the control points and re-estimate the camera parameters. This procedure is performed in an iterative manner till convergence, resulting in an accurate recovery of camera parameters (see results in Section 5).

3. Traditional Camera Calibration

Planar patterns based on square control points are a popular choice for calibrating a camera. There are widely used standard Matlab toolbox and OpenCV routines for camera calibration [1]. At least two different images of the pattern are required for calibration, although usually five to six images are used for higher accuracy and numerical stability [22]. The calibration process includes two steps: corner detection and calibration parameter optimization using Levenberg-Marquardt [14]. Corners are detected using the Harris interest point detector and then refined using a gradient-based search. In the optimization step, the initial calibration parameters are first estimated by a closed-form solution and then optimized by Levenberg-Marquardt to minimize the reprojection error [22].

3.1. OpenCV Square Control Point Refinement

Traditional calibration approach as employed by OpenCV [1] performs initial corner detection using the Harris interest point operator. These control points are then further refined using an iterative subpixel localization with a gradient-based search,

$$\sum_{i \in N} \Delta_{p_i} \times (q - p_i) = 0, \quad (1)$$

where q is the control point, p_i is a location in the neighborhood N of the control point q and Δ_{p_i} is the image gradient at location p_i . The above equation can be solved for q ,

$$q = \left(\sum_{i \in N} \Delta_{p_i}^T \Delta_{p_i} \right)^{-1} \sum_{i \in N} \Delta_{p_i}^T \Delta_{p_i} p_i. \quad (2)$$

When the calibration pattern is distorted and not in the fronto-parallel pose, then the orthogonal relationship between the gradient Δ_{p_i} and the position vector p_i does not hold, and consequently, the gradient based search process fails to perform sub-pixel localization of control points. Figure 2 (a & b) show the input image and the gradient vectors around a corner respectively. It can be observed that the gradient is not orthogonal to edges, contradicting the assumption made by the OpenCV algorithm [1].

4. Iterative Control Point Refinement

Figure 2 (c & d) show the canonically transformed image and gradient vectors around a corner respectively. In this canonical view, we can observe that the gradient is indeed orthogonal to the image edge and, therefore, using a canonical image for sub-pixel localization will result in precise control point localization. The canonical fronto-parallel image can be obtained by following an iterative approach towards the refinement of calibration pattern control points.

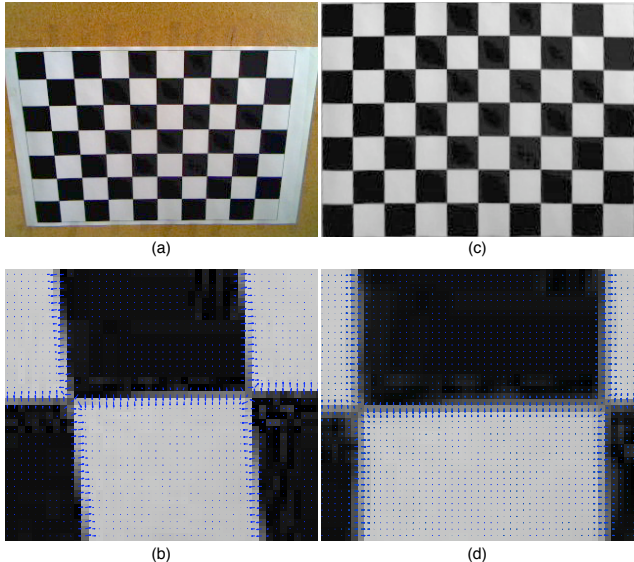


Figure 2: (a) Calibration pattern input image. (b) OpenCV [1] requires orthogonality of gradient to the image edges, which is not satisfied in the distorted non fronto-parallel input images. (c) Canonically transformed view proposed in this paper. (d) We can observe that the gradient, to a large degree, is indeed orthogonal to the image edges in the canonical view.

In the first iteration, we use OpenCV to obtain an initial estimate of camera parameters [1, 22]. This initial estimate is then used to undistort the input images for radial distortion and then unproject them onto a canonical fronto-parallel plane in the world coordinate system (see Figure 1 for an example). Since the canonical images have been both undistorted and unprojected, control points can be detected easily and localized precisely as compared to the original input images. After localizing the control points of a pattern, we then reproject the control points using the estimated calibration parameters into the camera coordinate system. These projected control point locations are then used to recompute the camera parameters and the process repeats until convergence (see Figure 4 for the complete algorithm). An important aspect of our iterative refinement approach for pattern control points is that it is pattern agnostic. We have applied this approach to refine the control points of square (checkerboard), circle and ring pattern. Different patterns, however, do need different control point detection algorithms.

Square Control Point: In case of the square pattern (checkerboard), we have used two different control point localization approaches. In the “OpenCV Iterative” approach, we localize the control points using the approach outlined in OpenCV (see Section 3.1). In the “Square Iterative” approach, we have used a checkerboard template to localize the square corners using normalized cross-correlation.

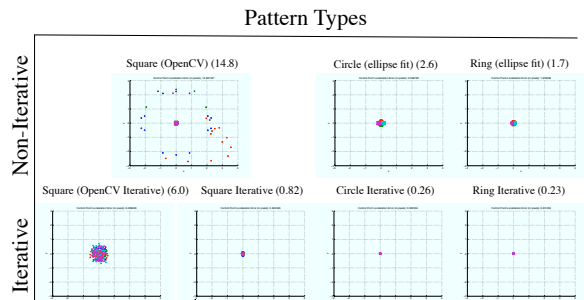


Figure 3: Localization error of the control points against the known ground-truth control point locations in synthetic images. Number in the bracket is the RMS localization error in pixels. In terms of the patterns, Ring is the most accurate pattern to localize. Iterative refinement of control points leads to improvement in the localization accuracy of control points for all patterns. Figure best seen zoomed in.

Circle Control Point: In case of the circle pattern, we use the circle center as a control point and localize it in the input images by using the center of the ellipse fitted to the circle edge [6], we call this approach “Circle (ellipse fit)”. The detected center of the ellipse is not the projected center of the circle because of perspective distortion introduced by the non-frontal orientation of the calibration grid [8]. However, since we perform calibration in an iterative fashion, the center of the ellipse can be used as a starting point and subsequently refined. In the “Circle Iterative” approach, we transform the circle pattern to a canonical fronto-parallel plane, the calibration pattern images are then correlated with a circle filter (Gaussian blurred circle template) to localize the peak of normalized cross-correlation. We perform quadratic fitting in the neighborhood of the peaks to obtain sub-pixel accurate circle control points.

Ring Control Point: We follow the same approach for detecting the center of rings as outlined earlier for the circle pattern except that we use the average of the inner and the outer ellipse centers as our ring control point for the non-iterative approach, which we call “Ring (ellipse fit)”. During the iterative refinement stage (“Ring Iterative” approach), we use a Laplacian as our cross-correlation filter to localize the control points of the ring pattern in the canonical fronto-parallel images. This estimate of the ring control point is then further refined using quadratic fitting to obtain sub-pixel accuracy.

Figure 3 shows the control point localization error for a set of radially-distorted synthetic images (details on the synthetic image generation are given in Section 5.1). We can observe that by using the ring pattern for calibration instead of the square pattern, we can obtain a reduction in the Root-Mean-Square (RMS) localization error from ~ 15 pixels to ~ 1.7 pixels. Incorporation of the proposed iterative refine-

Objective

Given N images of the planar calibration grid, estimate the camera parameters.

Algorithm

1. **Detect control points:** Detect calibration pattern control points (corners, circle or ring centers) in the input images.
2. **Parameter Fitting:** Use the detected control points to estimate camera parameters using Levenberg-Marquardt [1].

Do until convergence

1. **Undistort and Unproject:** Use the camera parameters to undistort and unproject input images to a canonical pattern.
2. **Localize control points:** Localize calibration pattern control points in the canonical pattern.
3. **Reproject:** Project the control points using the estimated camera parameters.
4. **Parameter Fitting:** Use the projected control points to refine the camera parameters using Levenberg-Marquardt [1].

Figure 4: Camera calibration using iterative refinement of control points.

ment stage for the control points leads to a further one-order magnitude improvement from ~ 1.7 pixels to ~ 0.23 pixels. Our results show that improvements in the accuracy of control point localization directly leads to improvements in the accuracy of camera calibration.

5. Evaluation

We have conducted detailed evaluation of our proposed approach of iterative refinement of control points for camera calibration on both synthetic and real images. In addition, we have also evaluated the proposed approach on different calibration patterns with different control point localization algorithms. We show the benefits of increased camera calibration accuracy, due to the proposed approach, towards the task of visual hull reconstruction and stereo camera calibration.

5.1. Evaluation on Synthetic Images

We generated five synthetic images with radial distortion for each calibration pattern (square, circle, and ring) that consisted of a fronto-parallel image of the grid along with four images with $+/- 45^\circ$ rotation along the x and y axes respectively.

Table 1 compares the estimated camera parameters using the OpenCV approach [1] versus the proposed approach. The “Ground Truth” column presents the camera parameters that were used to generate the synthetic images, F_x, F_y are the focal length, U_0, V_0 is the principal point and K_1, K_2

Table 1: Calibration result for synthetic images. Compared to the traditional approach of OpenCV [1], the proposed iterative refinement approach leads to a dramatic improvement in calibration accuracy. In addition, the ring planar pattern produces accurate results as compared to the circle planar pattern, which in turn is better than the square planar pattern.

Parameters	Ground Truth	OpenCV [1]		OpenCV Iterative		Square Iterative		Circle (ellipse fit)		Circle Iterative		Ring (ellipse fit)		Ring Iterative	
		Values	Final	σ	Final	σ	Final	σ	Final	σ	Final	σ	Final	σ	Final
F_x	800	819.2	8.1	799.7	1.8	804.2	0.5	800.6	0.8	800.2	0.2	800.7	0.8	800.1	0.18
F_y	800	817.8	7.9	799.4	1.7	803.8	0.5	800.7	0.7	800.1	0.2	800.8	0.7	800.1	0.18
U_0	319.5	304.4	12.8	321.1	3.8	316.6	1.1	321.0	1.7	320.6	0.4	320.8	1.7	320.6	0.4
V_0	239.5	248.8	12.2	239.9	2.6	242.5	0.8	240.5	1.1	240.5	0.3	240.4	1.1	240.5	0.3
K_1	-0.30	-0.18	0.1	-0.31	0.02	-0.28	0.01	-0.31	0.01	-0.30	0.002	-0.31	0.01	-0.30	0.003
K_2	-0.30	-0.44	1.21	-0.19	0.27	-0.29	0.01	-0.17	0.12	-0.28	0.03	-0.18	0.12	-0.28	0.03
RMS	0	0.4855		0.1263		0.0319		0.0517		0.0125		0.0515		0.0119	

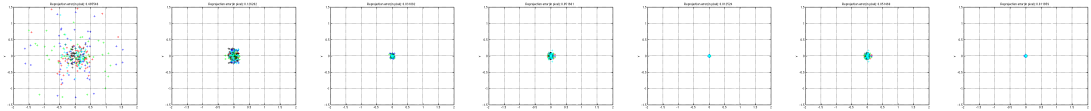


Figure 5: Scatter plots for the Root Mean Square (RMS) error between the detected control points and the re-projected control points using the estimated calibration parameters. The figures from the left to the right correspond to the columns of the table under which they are positioned respectively. It can be observed that the proposed approach of iterative refinement of the control points leads to accurate and stable recovery of calibration parameters across the patterns. Figure best viewed when zoomed in.

are the radial distortion parameters. The column denoted by “OpenCV” presents the calibration parameter estimates as obtained by OpenCV using the square corner control points. We next apply our iterative approach to detect and localize square corners in the canonical fronto-parallel images and we can see a dramatic improvement in the accuracy as noted in the “OpenCV Iterative” column. The RMS error drops from 0.48 pixels to 0.126 pixels, a reduction of 75%. In the “Square Iterative” column, we have combined our iterative approach with the checker-board template that uses normalized cross-correlation to localize the square corners, providing more accuracy and stability, resulting in a further drop in the RMS error to 0.032 pixels. Changing from the square pattern to the circle pattern, the “Circle” and the “Circle Iterative” columns present our results without and with iterative refinement of the circular control points (circle-centers) respectively. Note that not only is the circle pattern more accurate than the square pattern even without the iterative refinement, but also better compared to the “Square Iterative” approach after the iterative refinement. Finally, we present results on the ring pattern without and with iterative refinement in the columns “Ring” and “Ring Iterative” respectively. We can see that the ring pattern is more accurate as compared to the circle pattern and the RMS error drops to 0.0119 pixels. In summary, it is evident that the proposed approach of iterative refinement of the control points improves the accuracy of camera calibration.

Figure 5 shows the scatter plot for the RMS error between the ground truth control point location and the re-projected control point locations from the resulting camera

calibration using the different approaches. It can be observed that not only does the proposed approach recover the calibration parameters accurately but it is also stable as the scatter plot is symmetric around the origin reflecting the symmetry of the synthetic images around the origin. In all of our experiments, we use only 2 iterations for the refinement of the control points. Figure 6 (a) shows the RMS calibration error comparing the proposed approach of iterative refinement of control points and the non-iterative approach as the number of images are increased. Figure 6 (b) shows the RMS calibration error as additive Gaussian noise corrupted ground-truth control point locations are used as the localized control points for calibration. We can observe that in both these scenarios, the proposed approach of iterative control point refinement recovers the camera parameters with accuracy and stability.

5.2. Evaluation on Real Images

We collected five images from a camera for each pattern to evaluate our proposed approach. Table 2 compares the traditional approach of OpenCV [1] versus the proposed approach. The different columns of the table have the same description as before (from the previous sub-section) and present results for camera calibration using different patterns, and with and without iterative refinement. We can see by looking at the RMS error of the traditional versus the proposed approach that the iterative refinement of control points in the canonical images leads to large improvements in the accuracy of camera calibration. In particular, the proposed approach of iterative refinement of control points

Table 2: Camera calibration result for images taken from a camera. We can notice that the proposed iterative refinement helps to reduce the error of the traditional approach using OpenCV [1] and the square planar pattern from 0.198 to 0.059 using the iterative refinement along with the ring planar pattern.

Parameters	OpenCV [1]		OpenCV Iterative		Square Iterative		Circle (ellipse fit)		Circle Iterative		Ring (ellipse fit)		Ring Iterative	
	Final	σ	Final	σ	Final	σ	Final	σ	Final	σ	Final	σ	Final	σ
F_x	337.85	8.77	334.51	4.05	335.58	2.40	330.39	3.84	330.39	1.95	328.28	3.18	327.17	1.64
F_y	338.90	8.71	335.38	4.02	336.57	2.39	332.15	3.80	332.11	1.94	329.91	3.15	328.64	1.62
U_0	164.23	3.12	163.81	1.48	163.65	0.87	164.84	1.53	165.08	0.77	164.80	1.33	164.71	0.68
V_0	130.11	3.45	130.89	1.62	130.37	0.96	131.47	1.57	131.08	0.79	131.16	1.41	130.80	0.72
K_1	-0.11	0.06	-0.14	0.03	-0.13	0.02	-0.12	0.02	-0.14	0.015	-0.17	0.02	-0.17	0.001
K_2	0.11	0.42	0.27	0.19	0.22	0.12	0.08	0.17	0.16	0.09	0.24	0.12	0.25	0.06
RMS	0.198		0.139		0.081		0.135		0.068		0.115		0.059	

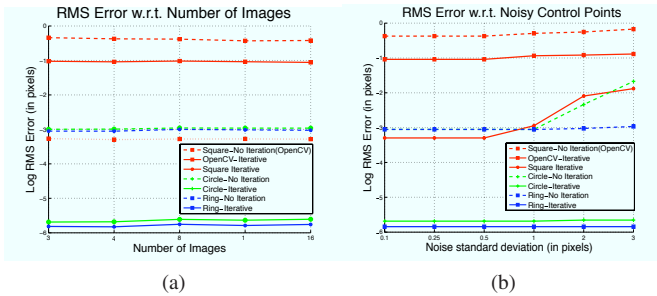


Figure 6: (a) Root-Mean-Square (RMS) error remains stable as the number of images are increased. (b) In the case of additive Gaussian noise corrupted ground-truth control point locations being used as the localized control points, we can see that the proposed iterative approach consistently performs better than the non-iterative approach. Figure best seen in color.

leads to $\sim 60\%$, $\sim 50\%$, and $\sim 50\%$ improvement in the RMS error over the non-iterative approach for the square, circle and the ring calibration patterns respectively.

Table 3 shows a comparison of the proposed approach of camera calibration for the circle pattern against OpenCV and the approach presented by Kannala *et al.* [10] for camera calibration using real images. In comparison to OpenCV [1], the proposed approach obtains roughly 50% RMS error reduction at a moderate computational cost. In comparison to Kannala [10], the proposed iterative refinement approach not only obtains a 35% improvement in the RMS error, but also is more than 60 times faster.

5.3. Application: Visual Hull Reconstruction

Visual hull reconstruction refers to the task of constructing the 3D convex hull from multiple silhouette images. Since, the process relies on the intersection of the silhouettes, it therefore requires accurate camera calibration. This requirement becomes all the more demanding for applica-

Table 3: Calibration comparison for circle pattern using real images. The proposed iterative approach achieves the lowest RMS error at a moderate computational cost.

Parameter	OpenCV [1]	Kannala [10]	Circle Iterative (Proposed)
RMS (pixels)	0.135	0.111	0.068
Time (seconds)	~ 4	> 600	~ 10

tions that involve the reconstruction of fine details of an object. In this experiment, we generated a set of 60 cameras located around an object at 6° interval. Gaussian noise was then added to the camera parameters in accordance with the noise levels obtained from each calibration approach in Table 1. We then use the visual hull reconstruction algorithm of Yamazaki *et al.* [20] to reconstruct polyhedral visual hull for the two synthetic objects: seaweed and ant. Figure 7 shows the result of visual hull reconstruction for the *seaweed* and *ant* objects respectively. We can observe that the fine details of the objects, such as the thin branches of the seaweed or the ant tentacles, are completely lost after calibration is done using OpenCV [1]. The proposed approach of iterative refinement of control point, on the other hand, is able to recover the fine object structures due to accurate calibration. Note that the ring iterative approach is more accurate compared to the circle iterative approach.

5.4. Application: Stereo Camera Calibration

In this experiment, we calibrate two cameras using the different calibration algorithms and quantitatively check the accuracy of the resulting calibration using epipolar lines. Specifically, a pair of epipolar lines must pass through the same point in the left and the right image. We intrinsically calibrate the two cameras and then solve for the extrinsic parameters to recover metric stereo camera calibration. The fundamental matrix is then recovered from the intrinsic and the extrinsic camera parameters. Figure 8 shows two sets of objects that were used to test the accuracy of the cali-

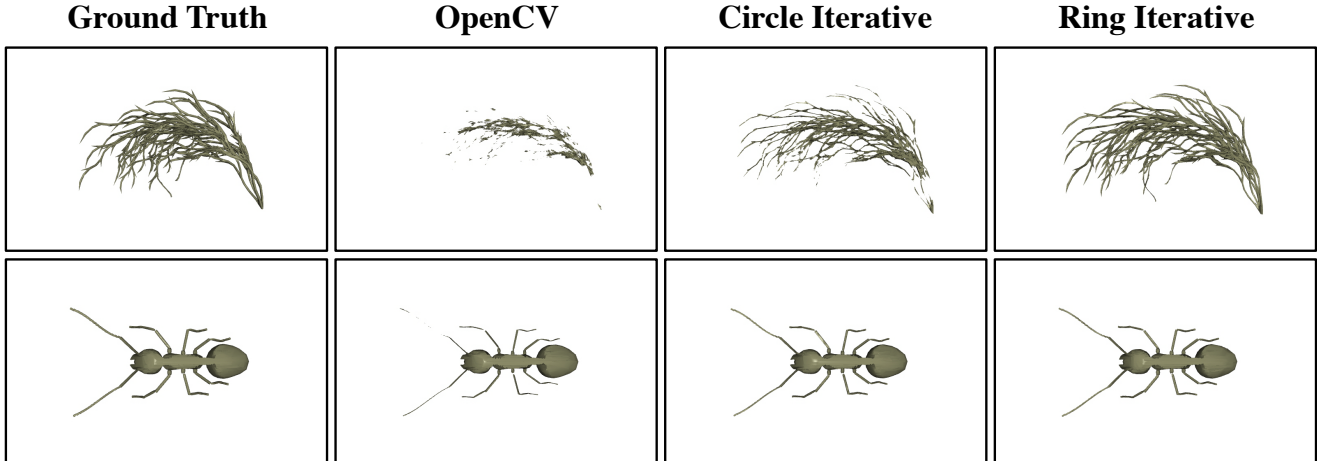


Figure 7: Visual hull reconstruction requires accurate camera calibration. *Ground Truth*: Synthetic model of a seaweed from which 60 images were taken at an interval of 6° around the object. *OpenCV*: Visual hull reconstruction after calibration using the OpenCV camera calibration toolbox. *Circle & Ring Iterative*: Visual hull reconstruction after calibration using the proposed circle and ring iterative approaches respectively. Notice the difference in the reconstruction of thin branches of the seaweed and the tentacles of the ant.

Table 4: The proposed iterative refinement approach leads to an accurate recovery of the epipolar geometry across the different planar calibration patterns.

Epipole error (pixels)	OpenCV [1]	Square Iterative	Circle Iterative	Ring Iterative
Flower	10.02	4.64	3.22	0.45
Sculpture	9.32	4.58	2.58	0.17

bration. Figure 8 (first and fourth row) are the corresponding left and right images for the the two objects used in the experiment and the lines of the same color must pass through the same point (denoted by a ‘+’ sign) in the both the images. Table 4 presents the epipole error that is defined to be the distance between the epipole line and the image point. The “OpenCV” column shows that the calibration using OpenCV contains large errors (~ 9 pixels). The other columns in the table show improvement in the stereo calibration as we move from the “Square Iterative” column to the “Ring Iterative” column, which is the most accurate of the calibration approaches. Note that the presence of a large non-uniform overhead lighting during calibration pattern image capture made the task of calibration difficult. However, this non-uniform lighting provides a good testing scenario for comparison between the non-iterative and the proposed iterative refinement approach for calibration.

6. Conclusions

We have proposed an approach for accurate camera calibration using iterative refinement of control points. Using an initial calibration estimate, we propose to undistort and

unproject the input calibration images onto a fronto-parallel plane for precise localization of the control points. The localized control points are then used to recompute the camera calibration parameters and this iterative process continues until convergence. We have presented results on synthetic and real images outlining the increased camera calibration accuracy obtained by the proposed iterative refinement approach for the square, circle and ring pattern. In addition, we show the benefits of increased camera calibration accuracy for visual hull reconstruction and stereo camera calibration.

References

- [1] G. Bradski, A. Kaehler, and V. Pisarevsky. Learning-based computer vision with intel’s open source computer vision library. *INTEL-TECH-J*, 9(2), may 2005.
- [2] D. C. Brown. Decentering distortion of lenses. *Photogrammetric Engineering*, 32(3), 1966.
- [3] Q. Chen, H. Wu, and T. Wada. Camera calibration with two arbitrary coplanar circles. In *ECCV*, 2004.
- [4] C. Colombo, D. Comanducci, and A. del Bimbo. Camera calibration with two arbitrary coaxial circles. In *ECCV*, 2006.
- [5] O. Faugeras and G. Toscani. The calibration problem for stereo. In *CVPR*, 1986.
- [6] A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *PAMI*, 21(5), 1999.
- [7] J. Heikkila. Geometric camera calibration using circular control points. *PAMI*, 22(10), 2000.
- [8] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *CVPR*, 1997.
- [9] G. Jiang and L. Quan. Detection of concentric circles for camera calibration. In *ICCV*, 2005.
- [10] J. Kannala and S. S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *PAMI*, 2006.

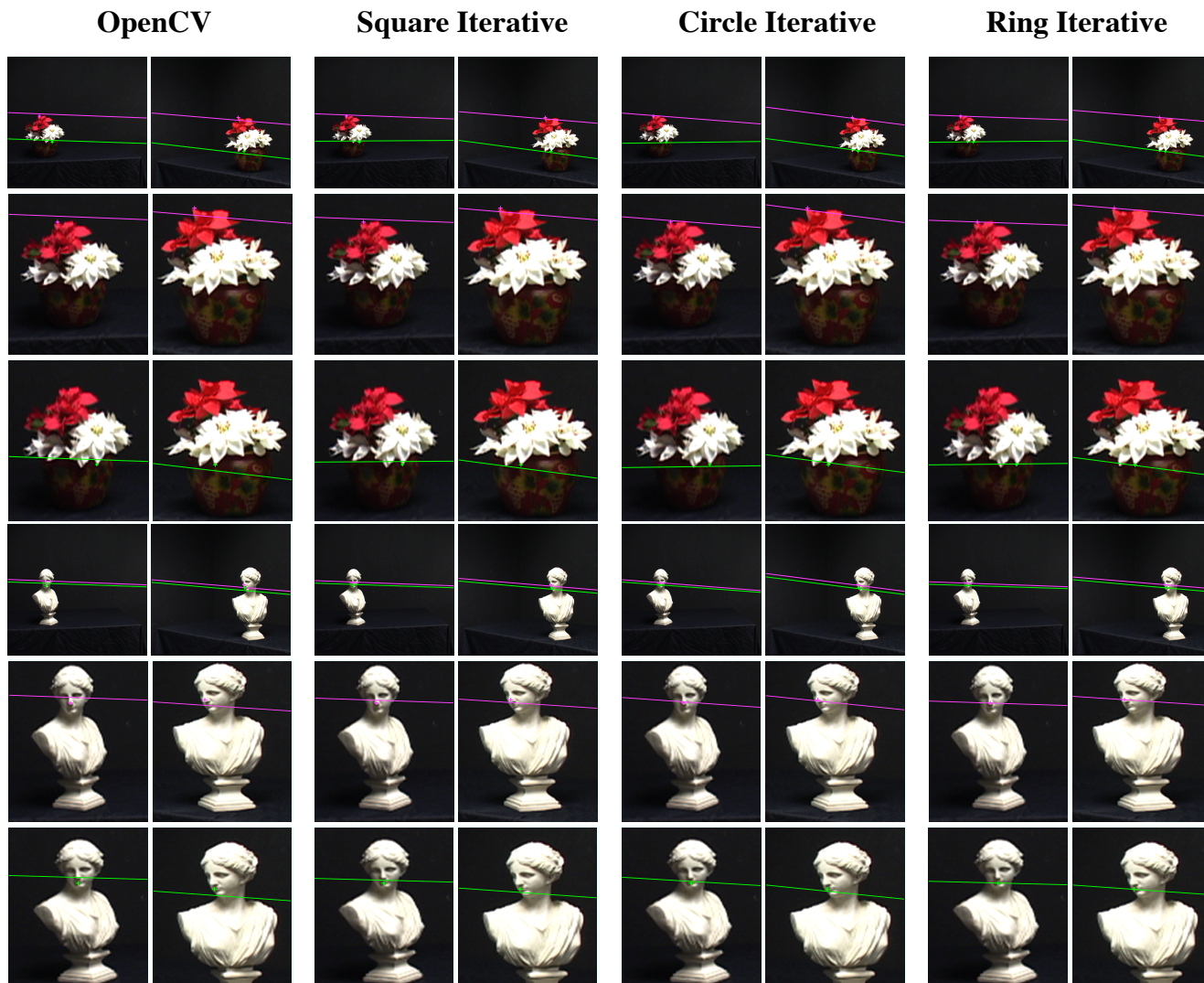


Figure 8: Stereo camera calibration: A pair of corresponding epipolar lines (same color) must pass through the same point, denoted by the ‘+’ sign. The “Ring Iterative” approach recovers the most accurate epipolar lines. “Circle Iterative” does better than “Square Iterative”, which is better than OpenCV [1]. Figure best viewed in color and zoomed in.

- [11] J.-S. Kim, P. Gurdjos, and I.-S. Kweon. Geometric and algebraic constraints of projected concentric circles and their applications to camera calibration. *PAMI*, 27(4), 2005.
- [12] J.-S. Kim, H.-W. Kim, and I.-S. Kweon. A camera calibration method using concentric circles for vision applications. In *ACCV*, 2002.
- [13] X. Meng and Z. Hu. A new easy camera calibration technique based on circular points. *Pattern Recognition*, 36(5), 2003.
- [14] J. J. More. *The Levenberg-Marquardt Algorithm: Implementation and Theory*. Springer, Berlin, 1977.
- [15] P. Sturm and S. Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *CVPR*, 1999.
- [16] J.-P. Tarel and A. Gagaloicz. Calibration de caméra à base d’ellipses. *Traitement du Signal*, 12(2), 1995.
- [17] B. Triggs. Autocalibration from planar scenes. In *ECCV*, 1998.
- [18] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE JRA*, 3(4), 1987.
- [19] Y. Wu, H. Zhu, Z. Hu, and F. Wu. Camera calibration from the quasi-affine invariance of two parallel circles. In *ECCV*, 2004.
- [20] S. Yamazaki, S. G. Narasimhan, S. Baker, and T. Kanade. Coplanar shadowgrams for acquiring visual hulls of intricate objects. In *ICCV*, 2007.
- [21] C. Yang, F. Sun, and Z. Hu. Planar conic based camera calibration. In *ICPR*, 2000.
- [22] Z. Zhang. A flexible new technique for camera calibration. *PAMI*, 22(11), 2000.
- [23] Z. Y. Zhang. A flexible new technique for camera calibration. In *Microsoft Research*, 1998.
- [24] H. Zhong, F. Mai, and Y. Hung. Camera calibration using circle and right angles. In *ICPR*, 2006.