Efficient Pose Selection for Interactive Camera Calibration

Pavel Rojtberg *
Fraunhofer IGD, Darmstadt
TU Darmstadt

Arjan Kuijper †
Fraunhofer IGD, Darmstadt
TU Darmstadt

ABSTRACT

The choice of poses for camera calibration with planar patterns is only rarely considered — yet the calibration precision heavily depends on it. This work presents a pose selection method that finds a compact and robust set of calibration poses and is suitable for interactive calibration. Consequently, singular poses that would lead to an unreliable solution are avoided explicitly, while poses reducing the uncertainty of the calibration are favoured. For this, we use uncertainty propagation.

Our method takes advantage of a self-identifying calibration pattern to track the camera pose in real-time. This allows to iteratively guide the user to the target poses, until the desired quality level is reached. Therefore, only a sparse set of key-frames is needed for calibration.

The method is evaluated on separate training and testing sets, as well as on synthetic data. Our approach performs better than comparable solutions while requiring 30% less calibration frames.

Index Terms: I.2.10 [ARTIFICIAL INTELLIGENCE]: Vision and Scene Understanding—Modeling and recovery of physical attributes; I.5.5 [PATTERN RECOGNITION]: Implementation—Interactive systems;

1 INTRODUCTION

Camera calibration in the context of 3D computer vision is the process of determining the internal camera geometric and optical characteristics (intrinsic parameters) and optionally the position and orientation of the camera frame in the world coordinate system (extrinsic parameters) [14]. The performance of many 3D computer vision algorithms directly depends on the quality of this calibration. Furthermore, calibration is a recurring task that has to be performed each time the setup is changed. Even if a camera is replaced by an equivalent from the same series, the intrinsic parameters may vary due to build inaccuracies. The prevalent approach to camera calibration [16] is based on acquiring multiple images of a planar pattern of known size.

However, there are degenerate pose configurations [11] that lead to unreliable solutions. Therefore, the task of calibration cannot be performed by inexperienced users — even researchers working in the field often struggle to quantify what constitutes good calibration images.

There has been research on the effect of the angle between image plane and pattern on the estimation error; Triggs [13] related the angular spread to the error in focal length. He found a spread of more than 5° necessary. Sturm and Maybank [11] further differentiated between estimating principal point and focal length. More importantly, they discussed possible singularities when using one and two planes for calibration and related them to the individual pinhole parameters; e.g., if the pattern is parallel to the image plane in every frame, the focal length cannot be determined. These findings were replicated in [16]. However, the effect of poses on the estimation of the distortion parameters or general camera-to-board poses have not been considered so far.

Another aspect is the quality and quantity of calibration data. Sun and Cooperstock [12] evaluated the sensitivity of camera models to noise, training data quantity and the calibration accuracy in respect to model complexity. However, they only measured the residual error on the respective training set, which is subject to over-fitting. To overcome this, Richardson et al. [10] introduce the Max Expected Reprojection Error (Max ERE) metric that instead correlates with the testing error and thus allows a meaningful test for convergence. Furthermore, they automatically compute a "best next pose" and use it for user guidance as an overlaid projection of the pattern. The poses are selected by performing an exhaustive search in a fixed set of about 60 candidate poses. For each pose a hypothetical calibration including this pose is performed and the pose that minimizes the Max ERE is selected. However the candidate poses are uniformly distributed in the field of view and do not explicitly consider the angular spread and degenerate cases [11].

In the general context of user assistance for calibration tasks [9], camera calibration was not yet specifically considered.

We propose to analytically generate optimal pattern poses while explicitly avoiding the degenerate pose configurations. For this we relate poses to constraints on individual parameters, such that the resulting pose sequence constrains all calibration parameters and ensures an accurate calibration. This reduces the computation time from seconds to milliseconds compared to the exhaustive search of [10].

The uncertainty of the calibration parameters is assessed using the covariance of the estimated solution. The pose sequence is then adapted such that more constraints are captured for the most uncertain parameter. The parameters covariance correlates with the testing error and therefore also serves as a convergence criterion.

Based on the above, our key contributions are:

1. Empirical evidence for the need of two distinct pose selection strategies and
2. an efficient pose selection scheme for implementing both of them.

This paper is structured as follows: in Section 2 the used camera model and the uncertainty estimation method are introduced and the choice of a suitable calibration pattern is discussed. Section 3 motivates and describes our novel pose selection method, while Section 4 describes the full calibration pipeline. In Section 5 the method is evaluated on real and synthetic data and compared with OpenCV
2 Preliminaries

We will use the pinhole camera model that, given the camera orientation $R$, position $t$ and the parameter vector $C$, maps a 3D world point $P = [X, Y, Z]$ to a 2D image point $p = [x, y]$

$\pi(P; R, t, C) = K \Delta(\frac{1}{Zc}[R \ t] \ P).$  \hspace{1cm} (1)

Here $[R \ t]$ is a 3x4 affine transformation, $Zc$ denotes the depth of $P$ after affine transformation, and $K$ is the camera calibration matrix containing the focal lengths (and aspect ratio) $[f_x, f_y]$ and the principal point $[cx, cy]$. Zhang [16] also includes a skew parameter $\gamma$ — however, for CCD cameras it is safe to assume $\gamma$ to be zero [12, 6]. $\Delta(\cdot)$ models the commonly used [12] radial (2a) and tangential (2b) lens distortions (following [7]) as

$\Delta(p) = p \left(1 + k_1r^2 + k_2r^4 + k_3r^6 \right) + \frac{2p_1xy + p_2(x^2 + 2y^2)}{p_1(x^2 + 2y^2) + 2p_2xy},$  \hspace{1cm} (2b)

where $r = \sqrt{x^2 + y^2}.$

Therefore $C = [fx, fy, cx, cy, k1, k2, k3, p1, p2].$

2.1 Estimation and error analysis

Given $M$ images each containing $N$ point correspondences, the underlying calibration method [16] minimizes the geometric error

$\epsilon_{res} = \sum_{i} \sum_{j} \lVert p_{ij} - \pi(P_i; R_i, t_j, C) \rVert^2,$  \hspace{1cm} (3)

where $p_{ij}$ is an observed (noisy) 2D point in image $j$ and $P_i$ is the corresponding 3D object point.

Eq. (3) is also referred to as the reprojection error and often used to assess the quality of a calibration. Yet, it only measures the residual error and is subject to over-fitting. Particularly $\epsilon_{res} = 0$ if exactly $N = 10.5$ point correspondences are used [6, §7.1].

The actual objective for calibration however, is the estimation error $\epsilon_{est}$, i.e. the distance between the solution and the (unknown) ground truth. Richardson et al. [10] propose the Max ERE as an alternative metric that correlates with the estimation error and also has a similar value range (pixels). However, it requires sampling and re-projecting the current solution. Yet for user guidance and monitoring of convergence only the relative error of the parameters is needed. Therefore, we directly use the variance $\sigma^2_v$ of the estimated parameters. Particularly, we use the index of dispersion (IOD) $\sigma^2_v/C_i$ to ensure comparability of the parameters among each other.

Given the covariance of the image points $\Sigma_v$ the backward transport of covariance [6, §5.2.3] is used to obtain

$\Sigma_v = \left(J^T \Sigma_p^{-1} J\right)^+$  \hspace{1cm} (4)

where $J$ is the Jacobian matrix, $v = [C, R_1, t_1, \ldots, R_M, t_M]$ is the vector of unknowns and $(\cdot)^+$ denotes the pseudo inverse. For simplicity and because of the lack of prior knowledge we assume a standard deviation of $1px$ in each coordinate direction for the image points thus $\Sigma_p = I$.

The diagonal entries of $\Sigma_v$ contain the variance of the estimated $C$. $J$ is already computed in Levenberg-Marquardt step of [16].

2.2 Calibration pattern

Our approach works with any planar calibration target e.g. the common chessboard and circle grid patterns. However, for interactive user guidance a fast board detection is crucial. Therefore, we use the self-identifying ChArUco [5] pattern as implemented in OpenCV. This saves the time consuming ordering of the detected rectangles to a canonical topology when compared to the classical chessboard. However, one can alternatively use any of the recently developed self-identifying targets [1, 2, 4] here.

The pattern size is set to 9x6 squares resulting in up to 40 measurements at the chessboard joints per captured frame. This allows to successfully complete the initialization even if not all markers are detected as discussed in section 4.3.

3 Pose selection

The core idea of our approach is to explicitly specify individual key-frames which are used for calibration using the method of Zhang [16].

In this section first the relation of intrinsic parameters and board poses is discussed to motivate our split of the parameter vector into pinhole and distortion parameters. For each parameter group we then present our set of rules to generate an optimal pose while explicitly avoiding degenerate configurations.

3.1 Splitting pinhole and distortion parameters

Looking at eq. (1) we see that both $K$ and $\Delta(\cdot)$ are applied at post-projection and thus describe 2D-to-2D mappings. Therefore, one might consider estimating $C$ just from one board pose that uniformly samples the image. However, as both intrinsic and extrinsic parameters are estimated simultaneously by [16], ambiguities arise.

Assuming $R = I$ and the distortion parameters to be zero, by multiplying out (1) we get

$p = \begin{bmatrix} f_x(x + t_x) + c_x \\ f_y(y + t_y) + c_y \\ Z + t_z \end{bmatrix},$  \hspace{1cm} (5)

for all pattern points $P$. In this case there are two ambiguities between

1. the focal length $f$ and the distance to camera $t_z$ and
2. the in-plane translation $[fx, fy]$ and principal point $[cx, cy]$.

These ambiguities can be resolved by requiring the pattern to be tilted towards the image plane such that there is only one $t$ that satisfies eq. (1) for all pattern points.

Considering the distortion parameters of $\Delta(\cdot)$ on the other hand, there are no similar ambiguities due to the non-linearity of the mapping. The parameters are rather determined by the maximal distortion strength evident in the image. Here it is more important to accurately measure the distortion in the corresponding image regions (see Figure 2a).
Therefore, we split the parameter vector $\mathbf{C}$ into $\mathbf{C}_K = [f_x, f_y, c_x, c_y]$ and $\mathbf{C}_\Delta = [k_1, k_2, k_3, p_1, p_2]$ and consider each group separately.

### 3.2 Avoiding pinhole singularities

While optimizing parameters in $\mathbf{C}_K$, singular poses must be avoided. In addition to the case discussed above, we incorporate the cases identified in [11]. Particularly, we restrict the 3D configuration of the calibration pattern as follows:

- The pattern must not be parallel to the image plane.
- The pattern must not be parallel to one of the image axes.
- Given two patterns, the "reflection constraint" must be fulfilled. This means that the vanishing lines of the two planes are not reflections of each other along both a horizontal and a vertical line in the image.

These restrictions ensure that each pose adds information that further constrains the pinhole parameters.

### 3.3 Pose generation

As described in Section 3.1, each parameter group requires a different strategy to generate an optimal calibration pose. For the intrinsic parameters $\mathbf{C}_K$ we follow [13, 16] and aim at maximizing the angular spread between image plane and calibration pattern. Accordingly, poses are generated as follows:

1. We choose a distance such that the whole pattern is visible, maximising the amount of observed 2D points.

2. Depending on the principal axis (e.g. $x$ for $f_y$) the pattern is tilted in the range of $(-70^\circ; 70^\circ)$ around that axis. The actual angle is interpolated using the sequence $[0.25, 0.75, 0.125, 0.375, \ldots]$ which corresponds to the binary subdivision of the $(0; 1)$ range (see Figure 3). This strategy, as desired, maximises the angular spread.

3. The resulting pose would still be parallel to one of the image axes which prevents the estimation of the principal point along that axis [11]. Therefore, the resulting view is rotated by $22.5^\circ$ which implements this requirement while keeping the principal orientation.

4. When determining $[c_x, c_y]$ the view is further shifted along the respective image axis by $5\%$ of the image size. This increases the spread along that axis and leads to faster convergence in our experiments.

For the distortion parameters $\mathbf{C}_\Delta$ the goal is to increase sampling accuracy in image regions exhibiting strong distortions. For this we generate a distortion map based on the current calibration estimate that encodes the displacement for each pixel. Using this map we search for the distorted regions as follows:

1. Threshold the distortion map (Figure 2a) to find the region with the strongest distortion.

2. Given the threshold image, an axis aligned bounding box (AABB) is fitted to the region, corresponding to a parallel view on the pattern. Note that the constraints for $\mathbf{C}_K$ do not apply here.

3. The area covered by the AABB is excluded from subsequent searches (see Figure 3). Effectively, the distorted regions are thereby visited in order of distortion strength.

4. The pattern is aligned with the top-left corner of the AABB and positioned at a depth s.t. its projection covers $33\%$ of the image width.

The angular range and width limits mentioned above were set such that the calibration pattern could be reliably detected using the Logitech C525 camera.

### 3.4 Initialization

The underlying calibration method [16] requires at least two views of the pattern for an initial solution which we select as follows:

- For the parameters $\mathbf{C}_K$ a pose tilted by $45^\circ$ around $x$ is selected (see Section 3.3). This particular angle was suggested by [16] and lies in between the extrema of $0^\circ$ where the focal length cannot be determined and $90^\circ$ where the aspect ratio and principal point cannot be determined.

- Without any prior knowledge we aim at an uniform sampling for estimating $\mathbf{C}_\Delta$. To this end we compute a pose such that the pattern is parallel to the image plane and covers the whole view. While this violates the axis alignment requirements for $\mathbf{C}_K$ poses, it still provides extra information as it is not coplanar to the first pose [16]. Furthermore, the reflection constraint is fulfilled.

To render an accurate overlay for the first pose without prior knowledge of the used camera, we employ a bootstrapping strategy similar to [10]; if the pattern can be detected, we perform a single frame calibration estimating the focal length only — the principal point is fixed at the center and $\mathbf{C}_\Delta$ is set to zero.

### 4 Calibration process

In the following we present the parameter refinement and user guidance parts as well as any employed heuristics. This completes the calibration pipeline as used for the real data experiments.

#### 4.1 Parameter refinement

After obtaining an initial solution using two key-frames, the goal is to minimize the cumulated variance $\sum_i \sigma_i^2 \mid \sigma_i^2 \in \sigma_\mathbf{C}$ of the estimated parameters $\mathbf{C}$. We approach this problem by targeting the variance of a single parameter $C_i \in \mathbf{C}$ at a time. Here we pick the parameter with the highest index of dispersion (MaxIOD) $\sigma_i^2/C_i (\sigma_i^2$ iff $C_i = 0)$. Depending on the parameter group, a pose is then generated as described in Section 3.

For determining convergence, we use a ratio test of the parameter variance $r = \sigma_{i,n+1}^2/\sigma_{i,n}^2$. If the reduction $1 - r$ is below a given threshold, we assume the parameter to be converged and exclude it from further refinement. Here, we only consider parameters from the same group as there is typically only little reduction in the complementary group. The calibration terminates once all parameters $\mathbf{C}$ have converged.
4.2 User Guidance

To guide the user, the targeted camera pose is projected using the current estimate of the intrinsic parameters. This projection is then displayed as an overlay on top of the live video stream (See Figure 1 and the video in the supplemental material).

To verify whether the user is sufficiently close to the target pose we use the Jaccard index $J(A,B)$ (intersection over union) computed from the area covered by the projection of pattern from the target pose $T$ and the area covered by the projection from the current pose estimate $E$. We assume that the user has reached the desired pose if $J(T,E) > 0.8$.

Comparing the projection overlap instead of using the estimated pose directly is more robust since the pose estimate is often unreliable — especially during initialization.

4.3 Heuristics

Throughout the process we enforce the common heuristic [6, 7.2] that the number of constraints should exceed the number of unknowns by a factor of five. The used calibration method [16] not only estimates the intrinsic parameters $C$, but also the relative pose of model plane and image plane i.e. the parameters $R$, a 3D rotation, and $t$, a 3D translation. When using $M$ calibration images we thus have $d = 9 + 6M$ unknowns and each point correspondence provides two constraints. For initialization ($M = 2$) we thus have 21 unknowns, meaning 52.5 point correspondences are needed in total or 27 correspondences per frame. For any subsequent frame only 15 points are required.

To prevent inaccurate measurements due to motion blur and rolling shutter artifacts the pattern should be still. To ensure this we require all points to be re-detected in the consecutive frame and the mean motion of the points to be smaller then 1.5px (determined empirically).

Figure 4: Correlation of pose selection strategies and calibration parameter uncertainty expressed using the standard deviation $\sigma$ (thus the error bars mean “variance of $\sigma$”).

The first two poses are selected according to the initialization method. Poses 2-10 and 11-20 are selected by complementary strategies. Evaluated with synthetic images on 20 camera models sampled around the estimate of the Logitech C525 camera.

Figure 5: (a) Comparing error metrics on synthetic data: both MaxERE and the proposed Max IOD correlate with estimation error $\epsilon_{est}$. (standard deviation over 20 samples) (b) Required number of frames $M$ and $\epsilon_{est}$ in respect to the variance reduction threshold

5 Evaluation

The presented method was evaluated on both synthetic and real data. The synthetic experiments aimed at validating the parameter splitting and pose generation rules presented in Section 3, while the real data was used for comparison with other methods. Furthermore, the compactness of the results with real data was estimated by optimizing directly on the testing set.

5.1 Synthetic data

We performed multiple calibrations, each using 20 synthetic images. The first two camera poses were chosen as described in section 3.4 to allow a rough initial solution. The next 8 poses were chosen to optimize $C_D$ while the last 10 poses were optimizing $C_K$ (and vice versa).

The camera parameters were based on the calibration parameters of a Logitech C525 camera $C_{real}$. However, the actual parameters were sampled around $C_{real}$ using a covariance matrix that allowed 10% deviation for each of the parameters $\Sigma = \text{diag}(0.1 \cdot C_{real})$ as

$$ C \sim \mathcal{N}(C_{real}, \Sigma). \quad (6) $$

Therefore, each synthetic calibration corresponds to using a different camera $C$ with known ground truth parameters. To allow generalization to different camera models, we kept the above generation sequence, but used 20 different cameras $C$.

Figure 4 shows the mean standard deviation $\sigma_C$ of the parameters. Notably there is a significant drop in $\sigma$ iff a pose matching the parameter group is used.

We also evaluated the usage of MaxIOD as an error metric by comparing it to MaxERE [10] and a known estimation error $\epsilon_{est}$. Just as the MaxERE, the MaxIOD correlates with $\epsilon_{est}$ (see Figure 5a). Additionally, as Figure 5b indicates, the IOD reduction is suitable for balancing calibration quality and the number of required calibration frames.

5.2 Real data

For evaluating our method with real images, we recorded a separate testing set consisting of 50 images at various distances and angles covering the whole field of view. All images were captured using a Logitech C525 webcam at a resolution of 1280x720px. The autofocus was fixed throughout the whole evaluation, while exposure was fixed per sequence. Our method was compared to AprilCal [10] and calibrating without any pose restrictions using OpenCV.

We used the pattern described in section 2.2 that provides 40 measurements per frame for OpenCV as well as for our method. With AprilCal, we used the 5x7 AprilTag target that generates approximately the same amount of measurements.
Table 1: Our method compared to AprilCal and OpenCV on real data. Showing the average over five runs. Training on the testing set results in $\epsilon_{est} = 0.479$.

| Method         | mean $\epsilon_{est}$ | frames used | mean $\epsilon_{est}$ |
|----------------|------------------------|-------------|------------------------|
| Pose selection | 0.518                  | 9.4         | 0.470                  |
| OpenCV [3]     | 1.465                  | 10          | 0.345                  |
| AprilCal [10]  | 0.815                  | 13.4        | 1.540                  |
| Compactness test | 0.514               | 7           | 0.476                  |

The convergence threshold was set to 10% for our method and the stopping accuracy parameter of AprilCal was set to 2.0. As the OpenCV method does not provide convergence monitoring, we stopped calibration after 10 frames here.

Table 1 shows the mean results over 5 calibration runs for each method, measuring the required number of frames, $\epsilon_{est}$ and $\epsilon_{res}$. Here our method requires only 70% of the frames required by AprilCal while arriving at a 36% lower $\epsilon_{est}$ (64% compared to OpenCV).

5.3 Analyzing the calibration compactness

The results in the previous section show that our method is able to provide the lowest calibration error $\epsilon_{est}$ while using fewer calibration frames than comparable approaches. However, it is not clear whether the solution is using the minimal amount of frames or whether it is possible to use a subset of frames while arriving at the same calibration error.

Therefore, we further tested the compactness of our calibration result. We used a greedy algorithm that, given a set of frames captured by our method, tries to find a smaller subset. It optimizes for the testing set, directly minimizing the estimation error.

The algorithm is computed as follows; given a set of training images (the calibration sequence)

1. the initialization frames as described in Section 3.4 are added unconditionally;
2. each of the remaining frames is now individually added to the key-frame set and a calibration is computed.
3. For each calibration the estimation error $\epsilon_{est}$ is computed using the testing frames.
4. The frame that minimizes $\epsilon_{est}$ is incorporated into the key-frame set. Continue at step 2.
5. Terminate if $\epsilon_{est}$ cannot be further reduced or all frames have been used.

The greedy optimal solution requires 75% of the frames compared to the proposed method while keeping the same estimation error (see Table 1). This indicates that, while a significant improvement over [10], our method is not yet optimal in the compactness sense. The greedy algorithm requires an a-priori recorded testing set and only finds a minimal subset of an existing calibration sequence, but cannot generate any calibration poses.

5.4 User survey

We performed an informal survey among 5 co-workers to measure the required calibration time when using our method. The tool was used for the first time and the only given instruction was that the overlay should be matched with the calibration pattern. The camera was fixed and the pattern had to be moved. On average the users required 1:33 min for capturing 8.7 frames at $\epsilon_{est} = 0.533$.

6 Conclusion and future work

We have presented a calibration approach to generate a compact set of calibration frames that is suitable for interactive user guidance. Singular pose configurations are avoided such that capturing about 9 key-frames is sufficient for a precise calibration. This is 30% less than comparable solutions. The provided user guidance allows even inexperienced users to perform the calibration in less than 2 minutes. Calibration precision can be weighted against the required calibration time using the convergence threshold. The camera parameter uncertainty is monitored throughout the processes, ensuring that a given confidence level can be reached repeatedly.

Our evaluation shows that the amount of required frames can still be reduced to speed up the process even more. We only use a widespread and simple distortion model, additional distortion coefficients like thin prism [15], rational [8] and tilted sensor are to be considered in future work. Eventually one could incorporate a detection of unused parameters. This would allow to start with the most complex distortion model which could be gradually reduced during calibration.

Furthermore the method needs adaptation to special cases like microscopy where the depth of field limits the possible calibration angles or calibration at large distance where scaling the pattern accordingly is not desirable.

The OpenCV based implementation of the presented algorithm is available open-source at https://github.com/paroj/pose_calib.

References

[1] B. Atcheson, F. Heide, and W. Heidrich. Caltag: High precision fiducial markers for camera calibration. In VMV, volume 10, pages 41–48, 2010.
[2] T. Birdal, I. Dobryden, and S. Ilic. X-tag: A fiducial tag for flexible and accurate bundle adjustment. In 3D Vision (3DV), 2016 Fourth International Conference on, pages 556–564. IEEE, 2016.
[3] G. Bradski, A. Kaehler, and V. Pisarrevsky. Learning-based computer vision with intel’s open source computer vision library. Intel Technology Journal, 9(2), 2005.
[4] M. Fiala and C. Shu. Self-identifying patterns for plane-based camera calibration. Machine Vision and Applications, 19(4):209–216, 2008.
[5] S. Garrido-Jurado. Detection of ChArUco Corners, 2017. URL http://docs.opencv.org/3.2.0/d44/tutorial_charuco_detection.html. [Online; accessed 11-February-2017].
[6] R. Hartley and A. Zisserman. Multiple view geometry in computer vision. Robotics, 23(2):271–271, 2005.
[7] J. Heikkila and O. Silvén. A four-step camera calibration procedure with implicit image correction. In Computer Vision and Pattern Recognition, Proceedings., 1997 IEEE Computer Society Conference on, pages 1106–1112. IEEE, 1997.
[8] L. Ma, Y. Chen, and K. L. Moore. Rational radial distortion models of camera lenses with analytical solution for distortion correction. International Journal of Information Acquisition, 1(02):135–147, 2004.
[9] F. Pankratz and G. Klinker. [poster] ar4ar: Using augmented reality for guidance in augmented reality systems setup. In Mixed and Augmented Reality (ISMAR), 2015 IEEE International Symposium on, pages 140–143. IEEE, 2015.
[10] A. Richardson, J. Strom, and E. Olson. AprilCal: Assisted and repeatable camera calibration. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), November 2013.
[11] P. F. Sturm and S. J. Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 1. IEEE, 1999.

[12] W. Sun and J. R. Cooperstock. Requirements for camera calibration: Must accuracy come with a high price? In *Application of Computer Vision, WACV/MOTIONS’05 Volume I, Seventh IEEE Workshops on*, volume 1, pages 356–361. IEEE, 2005.

[13] B. Triggs. Autocalibration from planar scenes. In *Computer Vision—ECCV’98*, pages 89–105. Springer, 1998.

[14] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344, 1987.

[15] J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis & Machine Intelligence, (10):965–980*, 1992.

[16] Z. Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11):1330–1334, 2000.