UNIVERSITÀ DEGLI STUDI DI PARMA

Dottorato di Ricerca in Tecnologie dell'Informazione

XXIX Ciclo

Boosting camera calibration performances

Coordinatore:

Chiar.mo Prof. Marco Locatelli

Tutor:

Prof. Pietro Cerri

Dottorando: *Marco Patander*

Dicembre 2016

Contents

Introduction **1**

1 Preface: current trends in computer vision . . . 1

2 Historical point of view . . . 2

3 Available methods . . . 3

4 Main contribution . . . 6

5 Thesis outline . . . 7

1 Notes on camera calibration procedures **9**

1.1 Introductory clarifications . . . 9

1.2 Marker type . . . 10

1.2.1 Choice of the marker . . . 10

1.2.2 Marker accuracy evaluation . . . 11

1.2.3 Marker detector foundations . . . 13

1.2.4 Removing marker perspective bias . . . 16

1.3 Camera models . . . 18

1.3.1 Pinhole projection and world-to-image mapping pipeline . . . 18

1.3.2 Brown-Conrady distortion . . . 19

1.3.3 Division and rational models . . . 20

1.3.4 Classical fisheye mapping functions . . . 21

1.3.5 Kannala-Brandt model . . . 23

1.3.6 Catadioptric-fisheye unified model . . . 24

1.4 Calibration pattern distance . . . 27

2 Planar pattern pose suggestion **29**

2.1 Motivation . . . 29

2.2 Pose goodness evaluation . . . 30

2.2.1 Sampled expected reprojection error maximization . . . 32

2.2.2 Efficient estimation optimality criteria . . . 33

2.2.3 Parameters uncertainty minimization . . . 34

2.2.4 Maximum predicted residuals variance minimization . . . 34

2.2.5 Maximum likelyhood reprojection error minimization . . . 35

2.3 Results . . . 36

3 Conclusions and future work **43**

3.1 What has been done . . . 43

3.2 What may be done . . . 44

Bibliography **54**

Introduction

1 Preface: current trends in computer vision

Recent years see computer vision growing as never before, with great deal of attention from both research and industrial communities. On one hand, the development of deep learning techniques in artificial intelligence has brought great advances into scene understanding. On the other hand, a lot of efforts have been put into geometric techniques (projective geometry, computational geometry, photogrammetry) to recover accurate measurement of scene-related quantities, especially in relation to model rendering and virtual reality. Low cost availability of easy-to-use, high-resolution camera systems and off-the-shelf devices with remarkable computational power lead to ubiquitous, interconnected systems, able to tackle difficult problems, perceiving the surrounding environment and automatically generating and transforming representations to, e.g., automatically identify people in photos, generate a text description from a photo and vice-versa, recognizing hand-written text, converting text to speech and vice-versa, doing automatical translation between different languages, inspect industrial manifacturing, drive actuation of autonomous vehicles.

A wide range of computer vision algorithms involve mapping functions and quantities depending upon the specific pair camera-lens in use, especially those regarding geometric measurements and relations between scene and image spaces. Manufacturers provide estimates of some of these quantities, but cameras and lenses are often sold separated, so better results can be obtained with a calibration procedure. Some parameters, such as camera positioning with respect to a fixed frame (extrinsic parameters), are often not even available to the manifacturer, depending on single customers setups and conventions. An accurate calibration procedure provides highly valuable information for quantitative evaluations in other algorithms (e.g. visual odometry and 3D dense reconstruction) and photogrammetric considerations. Examples of applications include virtual reality, medical imaging, product quality assessment, vehicles and robot navigation.

2 Historical point of view

This thesis concerns the problem of geometric camera calibration (or *resectioning*). The task consists in finding a set of parameters to describe the camera according to a predefined reference model[1]. This is one of the most studied computer vision problems of all times. Indeed, writing a survey about camera calibration is a difficult task, needing to gather up a huge base of references, even when the attention is constrained to specific aspects [2]. The following paragraph is meant to give the reader a short overview of the leading aspects of the topic through the years, without the presumption of being exahustive.

Even before the birth of computer vision itself cameras were employed to measure the focal length and distortion of lenses. As an example, a quick search on the topic brings us back to 1937, when precision cameras were adopted to test airplane lenses [3]. Predating existence of computer vision (ca. 1960), measurements on the image were done by hand. We have to wait until the 1970s, when computer vision community investigated the scene-to-image projection and, exploiting projective geometry, developed the pinhole camera model, together with methods for estimating its parameters [4]. During the 1980s calibration reached a good accuracy, and the now-called “classical” calibration algorithms see the light – Tsai [5] being the most famous among others [6, 7, 8]. At the beginning of the 1990s computer vision focused on shape understanding: calibration markers position could be accurately estimated, boosting the precision [9, 10, 11]. Later years see the study of lenses

[1] Regarding the so-called model-free techniques (e.g. [1]), the task consists in finding the distortion field induced by displacement of each pixel – we can consider that as the reference model.

nonlinearities [12, 13, 14] and multistep techniques [15, 16]. Zhang gives birth to one of the most used calibration techniques [17]. Challenges in the new millenium involve inter-calibration between cameras and other sensors [18, 19, 20, 21, 22], such as IMU, LIDARs, GPSs, and great attention is given to camera calibration for virtual reality applications [23, 24, 25]. Most recent works deal mostly with self-calibration [26, 27, 28, 29, 30], although achievement of high-accuracy camera calibration is still of some interest [31].

Until now, photogrammetric (marker-based) camera calibration outperforms self-calibration techniques in terms of reprojection error minimization. There are some practical, nebulous aspects that lead to difficult procedures: accurate camera calibration remains a topic for expert, trained users. Even then, common procedures rely on acquisition of more images than necessary and check both model error and parameters uncertainty to be low; images are replaced or added under unsatisfactory results. To enhance the reliability of this process the community the use of a GUI to provide assistence has been introduced [32].

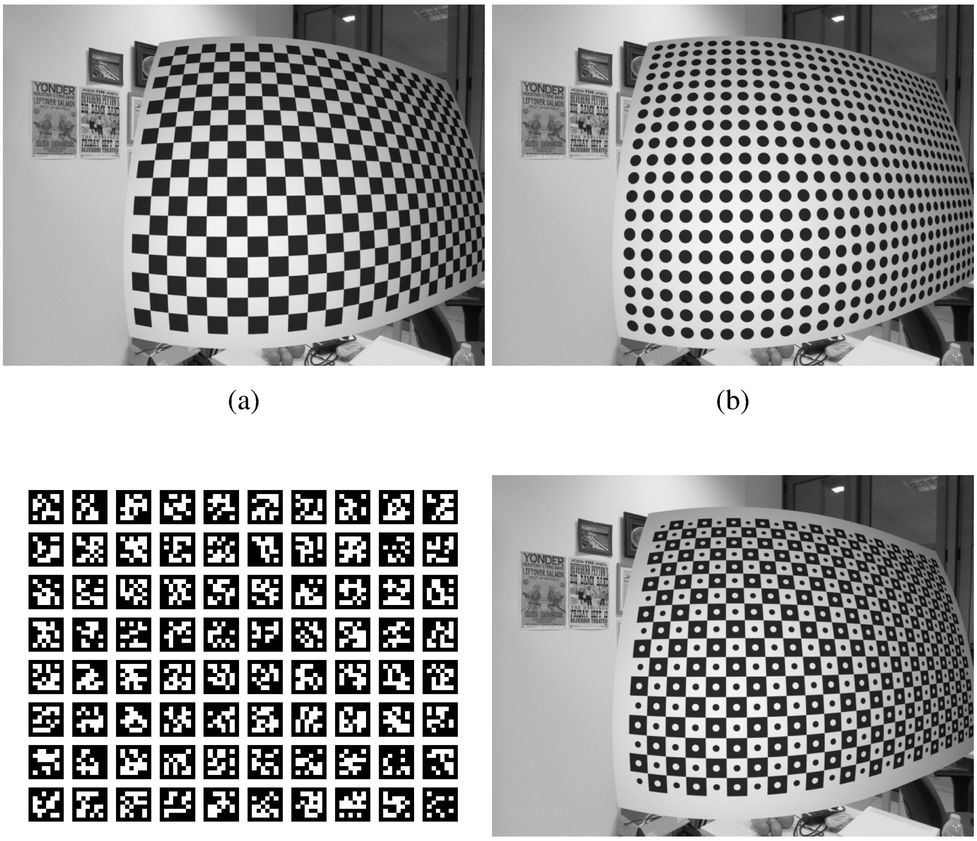
3 Available methods

In this section we want to give an overview of the different aspects involved in the camera calibration procedure, again without any presumption of being exahustive. We will limit ourself to photogrammetric (marker-based) calibration of cameras of the consumer/industrial type, with pinhole or fisheye optics – even if probably the same or at least similar techniques could be applied to different camera systems. The procedure consists in showing a known pattern to the camera. Intrinsic parameters calibration require multiple point of views of the pattern, so the relative position between the camera and the pattern must be changed, while for extrinsic parameters a single view of a fixed pattern may suffice. Common intrinsic parameters calibration techniques involve the estimation of the pose of the pattern, so we will refer to those in the dissertation, treating extrinsic calibration as a special case with world reference-positioned pattern(s). This kind of algorithms represents a preprocessing step for computer vision tasks, aimed to provide an accurate estimation of what can be elsewise considered prior knowledge. They can be run offline once, so no strict timing requirements are involved, even if sometimes collected data grows significantly and complexity must be constrained.

One of the main aspects to be considered is the error function for results evaluation. Literature papers divide themself in the ones using the vector of model parameters and the ones evaluating the reprojection error, i.e. the distance between a detected keypoint and its position after pattern(s) pose estimation, camera parameter optimization and marker projection on the image. Whilst the former has the advantage of working with non-keypoint features, it is unclear how comparison should be made (unless all the parameters converge simultaneously) and ground-truth information is required, moreover comparisons between results with different camera models are not possible in the general case. The latter is a single number that does not suffer from these problems but its minimization may not, in general, lead to better results (i.e. is not a measure of the model trueness).

Regarding the marker type, most works use keypoint-based ones, usually in the form of checkerboard [33] or circles [9]; with fiducial markers [23, 34] or with those that Kim et al. [35] call new calibration techniques [36, 37, 38] perspective information coded in the marker projection can be exploited; in some cases the whole marker shape can contribute to the optimization [39, 40]. Examples of marker types are available in Fig. 1. The aspects involving the reconstruction of marker grids (if any) given the list of detected markers, taking into account potential obstructions, receive lower interest with respect to the other topics, despite requiring topological ad-hoc [41, 42, 43] or graph-based [44] non-trivial techniques to be accomplished. Taking into account perspective bias on pattern projections [45] or reducing the effect of noise on pattern fitting [46, 47] represent another huge field of study.

Fisheye lenses introduce nonlinearities into the projection equation, opening possibilities to work on different camera models: several studies look for a simple, high accuracy model that can fit most of the fisheye projection mapping functions. Examples can include radial distortion, model-free calibrations [49, 50] and catadioptric-adapted models [51, 52] of different complexities [53]. Some of these models reach enough accuracy to be used for stereo cameras mounting fisheye lenses [54]. Compar-



(a) (b)

(c) (d)

Figure 1: Different marker types

(a-b) checkerboard and circle (blob) markers, the traditional and most used ones

(c) AprilTags [48] mosaic, a type of fiducial marker

(d) weird marker type, obtained mixing checkerboards and circles isons between pinhole and other models were made [55], besides studies to improve model convergence [56].

Final parameter estimations use different approximations [5, 57, 58] (often obtained by linearization) to initialize Newton or Levenberg-Marquardt nonlinear optimization. As proposed by Zhang [57], the function to be minimized is usually expressed in terms of the reprojection error, summing up the contribution of all the observed markers to obtain the maximum accuracy. Tang et al. [59] propose the use of a calibration harp for better take into account lens distortion, at the expense of requiring two different calibration patterns.

4 Main contribution

You might wonder if camera calibration can be considered a closed topic. Answering this question is not as easy as it seems: from one side, the forementioned global nonlinear optimization approach is optimal in a image-space error measurement sense. From another side, the effect of image noise cannot be compensated completely and practical aspects (limited number of markers, difficult to capture nonlinear lens distortion, bias in the distribution of the marker poses), when combined, prevents you from reaching high-accuracy results. Furthermore it makes sense having the best model fit not depending on what it will be used for, but when it comes down to apply that generic model solution in a specific case we would like to reach accuracies as if we had performed an ad-hoc optimization.

In this thesis we will:

- outline some practical considerations concerning camera calibration
- propose a method for suggesting marker grid poses

5 Thesis outline

In **Chapter 1** we outline practical aspects involved in camera calibration procedures that may be useful to readers approaching the subject to improve the results, especially in relation to marker types and camera models.

Chapter 2 introduces the pose suggestion topic, providing a new efficient algorithm for generating pose suggestions together with some test results.

Chapter 3 draws some conclusions on the work and tries to foresee what may be in the forthcoming developements.

Chapter 1

Notes on camera calibration procedures

1.1 Introductory clarifications

The aim of this chapter is to provide an insight of practical considerations arising in developing a camera calibration procedure, studying the state of the art more in-depth. Instead of focusing on a constrained problem version, we look for an high-accuracy, easy-to-use, general procedure. With “general” we mean it should not be restricted to specific camera and/or optics types (in the range of the consumer/industrial type). With “easy-to-use” we mean that, ideally, even untrained users are able to perform the procedure. Working in the automotive field, we require a calibration accuracy that allows to keep reasonable results even at several meters of distance – as an example, calibrating cameras for stereo matching we target no more than $0.1\,\text{px}$ reprojection error. In many cases practical aspects are difficult to model, sometimes leading to mathematical fields beyond our current skill level, making a rigorous theoretical treatise infeasible. We provide some empirical evidences of our intuitions, but we have not implemented the data gathering and comparison for all the different possibilities. A deep and complete study of the calibration markers would have required a study of many different marker types available in literature, an implementation of the detection algorithms and a thorough analysis of the error sources and data reliability: this is out of this thesis' scope. As a consequence claims are not accompanied by desirable scientific proofs, but by the study of the state of the art and by the opinion of professionals in the field.

1.2 Marker type

All of the highest-performing calibration techniques share a common fact: they require the camera to acquire a known pattern. This enables the estimation of the model parameters, given the marker model (in the 3D world) and its observations (in the 2D image). The choice of the marker types and methods for detection represents a deeply studied topic.

1.2.1 Choice of the marker

Most frequently used marker types in literature are: checkerboard, circles, concentric circles and fiducial markers. Single markers are often replicated into *marker grids* for the simultaneous acquisition of multiple data and for exploiting planarity information. Our research group had already developed a circles-based marker detector based on the OpenCV library [60]. With pinhole, 1280×960 images it obtains a reprojection error as low as $0.05\,\text{px}$, but its use on newly taken images from a recent 4K camera leads to poor results. This could be attributed to perspective bias [45], so we directed towards bias-free markers. We excluded fiducial ones, because the localization errors for the highest-accuracy ones available in literature [61] were more than one order of magnitude higher than the ones we obtained with circles. Even if we think that optimal[1] results could be obtained with checkerboard lines under the assumption of certain kinds of distortion and field-of-view ranges (see also par. 1.3.6), we thought that the use of keypoints in the form of a grid of concentric circles markers could give us results good enough for our aims. The number of concentric circles on the same marker in the grid represent a trade-off between the marker compression (occupied space) and the precision at which you can estimate the point; we thought two edges would be enough to provide enough precision without significant effects due to the distance from the linearization point. The resulting marker can be seen in Figure 1.1. We will use the term *ring marker* (or simply *ring*) to refer to this kind of marker from now on.

1.2.2 Marker accuracy evaluation

The keypoint accuracy performance has been evaluated on a synthetic image sequence with 20 different poses of the calibration pattern. Each image, similar to Figure 1.1(a) has a resolution of 1280×960, 8-bit color depth and a fixed, i.e. not distorted according to the camera model background to stress a bit the marker detector. The pattern is oversampled for better color accuracy (24x), image noise is simulated with gaussian blurring ($\sigma = 0.8$) and additive noise according to a uniform distribution $\mathcal{U}(-4,+4)$. The points available as an output of the marker detector have been compared with the ground truth set of points, choosing always the nearest match and discarding everything with more than 2 px of error norm. The average of the absolute values on each error component is taken as marker location error (Equation 1.1).

$$INLN = \left\{ (|\hat{x}-x^*|, |\hat{y}-y^*|) \;\middle|\; (x^*,y^*) = p^* = \operatorname*{argmin}_p \|\hat{p}-p\|_2,\ \|p^*\|_2 \leq 2 \right\}$$
$$\overline{\varepsilon} = \frac{\sum_{m \in INLN} m}{\#INLN} \tag{1.1}$$

Tipically we want the location error on the two components to be independent and identically distributed. In that case, the covariance matrix is a multiple of the identity and the worst-case maximum value $\|\overline{\varepsilon}\|_\infty = \max(\overline{\varepsilon}_x, \overline{\varepsilon}_y)$ can be computed picking any of the two vector coordinates.

[1] in the sense of reprojected model matching

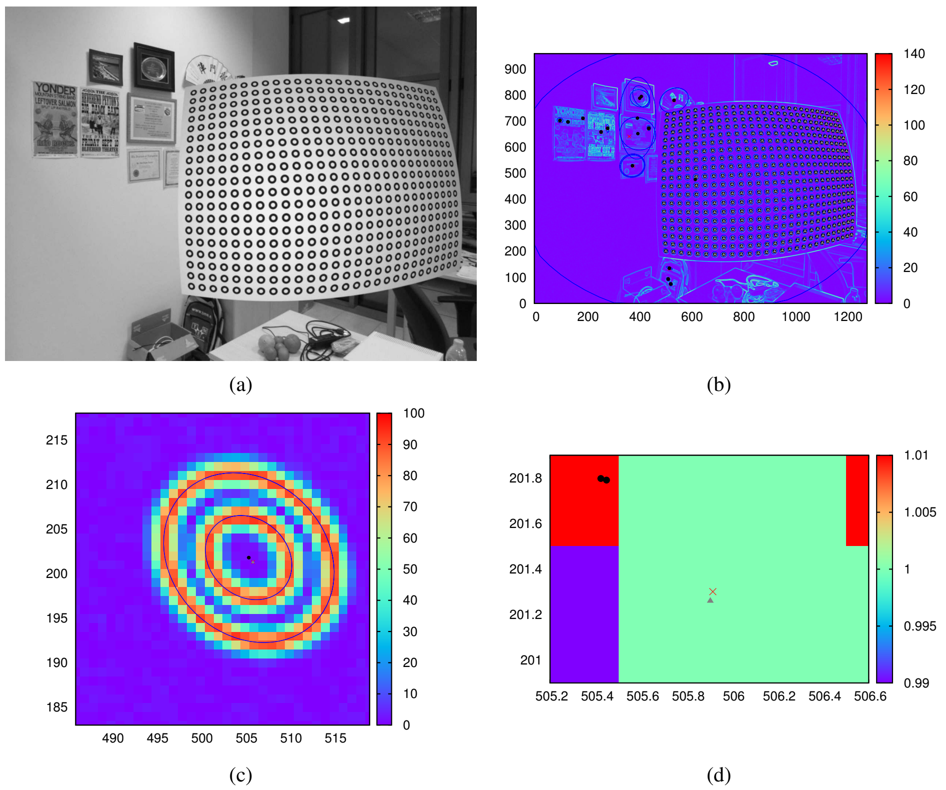


Figure 1.1: Planar marker made up of several rings (simulator rendering). (a) is the source image; (b) shows the color-encoded gradient magnitude, together with an intermediate result (after partial filtering) of the marker detector (blue ellipses); (c) and (d) display zoomed views of the bottom-left marker of the grid in (b), to highlight the position difference of the marker center localization. The black dots represent the ellipses centers, the red cross is the proposed marker location, the gray triangle is the ground-truth marker location. The marker location error norm in this example is approximately 0.0425 px. Best viewed in colors, zooming on the screen.

Using the keypoints reprojection error minimization, the calibration can be found following Alg. 1, having I_i the set of extracted keypoints in the i-th image.

Algorithm 1 Keypoint calibration using reprojection error

1: **function** CALIBRATE($\{I_i\}$, μ_0) ▷ returns the camera calibration $\hat{\mu}$
$\{I_i\}$ is the set of images keypoints
μ_0 is an initial guess for the calibration
2: **for all** $I_i \in \{I_i\}$ **do**
3: $\hat{p}_i \leftarrow$ `poseFromNPoints`(I_i)
4: $J_i \leftarrow$ `projectModelAtPose`$(\hat{p}_i, \mu_0)$
5: **end for**
6: $\hat{\mu} \leftarrow \underset{\mu}{\operatorname{argmin}}$ `reprojection_error`$(\{I_i\}, \{J_i\})$
▷ use μ_0 as nonlinear optimization starting point
7: **return** $\hat{\mu}$
8: **end function**

1.2.3 Marker detector foundations

For testing purposes we implemented a simple ring marker detector, clustering the image gradient magnitude and fitting an ellipse on each connected component. Most of the generated false positives, i.e. ellipses that do not belong to the pattern, can be then discarded keeping only concentric ellipses. This is a strong feature to be found in the image: the more the marker has concentric circles, the more is unlikely the same number of concentric circles will be found elsewhere in the image. More advanced methods could be employed, but in our simple tests we managed to remove all of them by appropriate thresholding and parameter tuning.

We were concerned about the error introduced by the lens distortion. Surely the geometry of pinhole camera maps the circles of the pattern into ellipses, but it is not so when different mapping functions are involved (as it will be shown in par. 1.3.4). Anyway even in this case the forementioned algorithm was able to find out the ellipses, with a slight loss in precision. Marker location error tests with different camera models are summarized in Table 1.3.

Given the points in a connected component, multiple techniques are available for least squares ellipse fitting. A proper, geometrical fitting of the ellipse requires a nonlinear least squares fitting. To avoid nonlinearity one could approximate the distance function using the Sampson one, as proposed by Szpak et al. [62]. Another simpler approach relies the algebraic distance; methods in this category mainly differ from each other for the normalization constraint. Given the implicit equation

$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

we want to find the set of parameters a, b, c, d, e, f that minimize

$$\frac{1}{2}\sum_i w_i \left(ax_i^2 + bx_iy_i + cy_i^2 + dx_i + ey_i + f\right)^2$$

for a given dataset (x_i, y_i) and corresponding weighting w_i. Using the notation of perspective geometry and homogeneous coordinates, notation simplifies to the minimization of $\frac{1}{2}\sum_i w_i p_i^\intercal C p_i = \frac{1}{2}\chi^\intercal D^\intercal D\chi$, where

$$C = \begin{bmatrix} a & \frac{b}{2} & \frac{d}{2} \\ \frac{b}{2} & c & \frac{e}{2} \\ \frac{d}{2} & \frac{e}{2} & f \end{bmatrix}, \quad \chi = \begin{bmatrix} a & b & c & d & e & f \end{bmatrix}^\intercal$$

and $D^i = w_i \begin{bmatrix} x_i^2 & x_iy_i & y_i^2 & x_i & y_i & 1 \end{bmatrix}$, the last one being the i-th row of the design matrix D.
The minimization can be solved finding the kernel of the gradiend, $D^\intercal D\chi = 0$. Obviously there is a multiplicative factor ambiguity in the solution (as one can see also from the implicit equation), so a normalization constraint has to be chosen. Most common constraints are $\|c\| = 1$, its dual[2] $f = 1$, or $4ac - b^2 = 1$; the last one constraining the conic to be an ellipse, as originally done by Fitzgibbon et al. [63].

We managed to obtain good results using the improved version proposed by Halíř [64], paying attention to data preconditioning before fitting. Splitting the conic

[2]Each of the two constraints represent the other under a duality transformation of the conic

parameters column vector $a = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$ into two blocks, the fitting problem can be stated as:

$$\begin{aligned} M\chi_1 &= \lambda\chi_1 \\ \chi_1^\mathsf{T} C_1 \chi_1 &= 1 \\ \chi_2 &= -S_3^{-1} S_2^\mathsf{T} \chi_1 \end{aligned} \tag{1.2}$$

having $M = C_1^{-1}\left(S_1 - S_2 S_3^{-1} S_2^\mathsf{T}\right)$ the 3×3 reduced scatter matrix, $S = \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} = D^\mathsf{T} D$ the full scatter matrix and $C_1 = \begin{bmatrix} 0 & 0 & 2 \\ 0 & -1 & 0 \\ 2 & 0 & 0 \end{bmatrix}$ the top-left anti-diagonal part of the elsewhere-zero normalization matrix. This formulation, in addition to being computationally more efficient, has been proven being numerically more robust [64]. Unfortunately, unlike stated by Halíř and Flusser [64, end of pag. 4], reduced scatter matrix M may not have three real eigenvalues: in many practical situations it ends up having two complex conjugates eigenvalues and one real eigenvalue. But complex eigenvalues correspond to complex eigenvectors, i.e. complex conics, so in this case we are interested in the eigenvector corresponding to the only one real eigenvalue.

Despite all the precautions this algorithm leads to numerically unstable results when provided non-preconditioned data that already represent a good ellipse estimate. Paying attention to the premultiplication by C_1^{-1} in the computation of M (it must be done swapping the rows of the right-hand operand, exploiting C_1 structure as stated in the reference paper) is important but not enough. Preconditioning the scatter matrix is a necessary step to obtain good ellipses estimates.

We could not get improvements with dual ellipses fitting techniques, as proposed by Tabatabai [65], maybe because we chose a poor gradient estimate (8-bit per coordinate, 3×3 sobel filter) to have a faster detector. We would have expected to gain some accuracy by contour points refinement, as proposed by Safaee-Rad et al. [66] and improved by Heikkila [67], but we did not – we are not sure of the reasons for this improvement lacking.

1.2.4 Removing marker perspective bias

A simple approach for extracting the marker keypoint location may be getting the centroid of the border points or of the filled circle (getting rid of the ellipse fitting), or taking the center of the ellipse(s). Unfortunately, as known from projective geometry, the projection of the center of a circle does not coincide with the center of the projection of the same circle, i.e. the center of the ellipse we are fitting. As shown in Table 1.1, our test reflect this fact: while it does not improve very much the ellipses center keypoint in terms of marker location absolute mean error, it significantly reduces the bias (signed mean error).

Keypoint	$\overline{\varepsilon_x}$	$\overline{\varepsilon_y}$	$\overline{\lvert\varepsilon_x\rvert}$	$\overline{\lvert\varepsilon_y\rvert}$
Centroid of the two ellipses	-0.063	-0.059	0.110	0.104
Average of the two ellipse centers	-0.019	0.022	0.045	0.044
Perspective bias corrected	0.001	0.001	0.049	0.039

Table 1.1: Marker location error components with different keypoint extraction methods (under the same pinhole, Brown-Conrady camera model)

The first who exploited this fact in the topic of camera calibration were Kim and Kweon [68, 69]. The idea was picked up by others [70, 71, 72], but were again the original authors to provide stronger foundations to their method into linear algebra [35]. Later Minh et al. [73] extended the idea to multiple concentric circles for increased accuracy.

Essentially the idea exploits the cross-ratio projective invariant to find out the projection of the concentric circles center. The method consists in the selection of four points (we can take Figure 1.2 as a reference) and building up a relation invariant to the projective transformation.

Even if not very popular in computer vision literature, given four collinear points A, B, C, D the cross-ratio:

$$(A, B; C, D) = \frac{\overline{AC} \cdot \overline{BD}}{\overline{AD} \cdot \overline{BC}} \tag{1.3}$$

is indeed a projective invariant value ([74, Theorem II-1]). It is worth noting (corol-

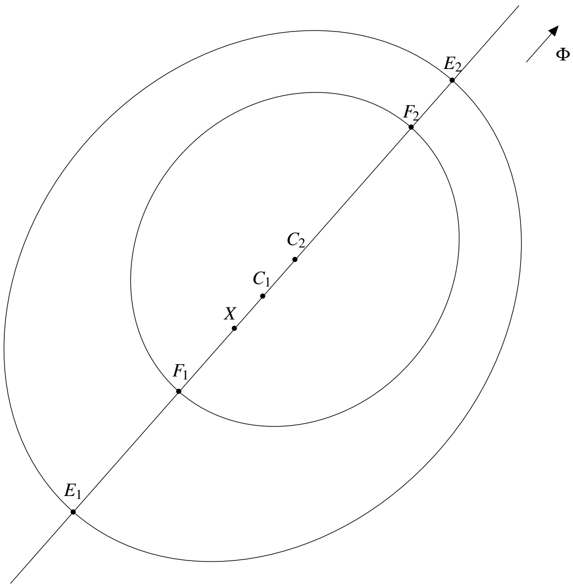


Figure 1.2: Cross-ratio projected center estimation – reference drawing

lary) that relation holds even involving the point at infinity on the projective line, for that being the case of *harmonic conjugates* pairs. Different versions of the method for finding the center projection X could be developed, differring from each other for the four points selection.

Original version by Kim and Kweon set up the relations:

$$\begin{aligned}(E_1, E_2; X, \Phi) = 2 \\ (F_1, F_2; X, \Phi) = 2\end{aligned} \tag{1.4}$$

on a projective one-dimensional reference system on the line through the two centers of projections C_1C_2. They solve the system of two equations in two unknowns, recovering the projected center. Omitting the horizon line information, the system could be also be reduced to the form:

$$(E_1, F_1; X, F_2) = (E_2, F_2; X, F_1) \tag{1.5}$$

Expansion of the equation results in a quadratic equation with two real distinct solutions, only one of which belongs to the segment F_1F_2.

As they point out in their seminal paper [68, par 2.2a], all the centers of projected concentric circles should lie on the same line. However noise in the ellipses estimation affects the results, so one may gain some improvement with an iterative solution as presented in [71].

1.3 Camera models

Dealing with noise is hard, coping with noise and nonlinearities is harder. Here we will present some the models that have been developed in the years to handle camera lens distortions.

1.3.1 Pinhole projection and world-to-image mapping pipeline

A lot of different models have been developed to take into account lens distortion, pinhole camera being the simplest one. Under this model points undergo a linear projective transformation. Unhomogenizing the coordinates, the projection transformation from the camera reference system to the normalized image plane can be written as:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{{}^{C}Z} \begin{pmatrix} {}^{C}X \\ {}^{C}Y \end{pmatrix} \tag{1.6}$$

To take into account lenses imperfection the Brown-Conrady polynomial model of distortion [75, 76] is the most used – simply a truncated Maclaurin series expansion of a nonlinear function from an "undistorted" radius to a "distorted" one.

Since calibration of these parameters often assume the introduced distortion to be small, better modelling of fisheye lenses is performed through nonlinear mapping functions. A complete image projection mapping pipeline is shown in Fig. 1.3, having f_p the non-linear mapping as explained in sec. 1.3.4 and K the intrinsics homogeneous affinity with matrix:

$$K = \begin{bmatrix} k_u & s & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.7}$$

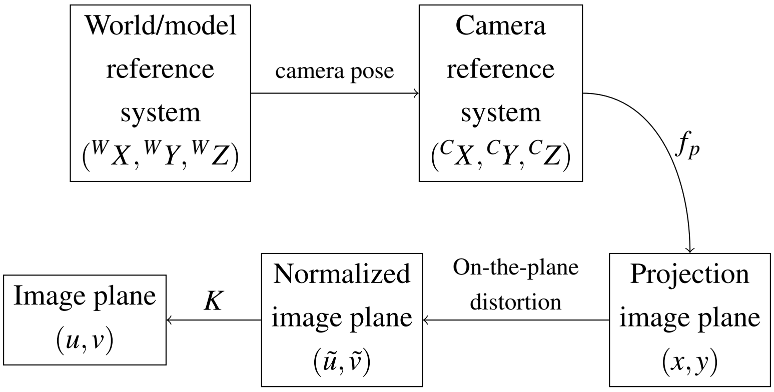


Figure 1.3: Complete image projection mapping pipeline.
On-the-plane distortion is usually omitted when
projection function f_p is different from pinhole

where k_u, k_v are the focal lengths, s is the skew parameter and (u_0, v_0) the principal point. Surely the boundary between f_p and on-the-plane (Brown-Conrady) distortion is ill-posed, they are drawn separately in the diagram only referencing the pinhole case, where f_p is linear in a projective space. Indeed some of the functions f_p may be seen as non-linear distortions on a projective plane through the change of variable $\theta = \arctan(r)$, as it will be clear in sec. 1.3.4, but this kind of modeling looses validity when nonlinear projections can handle fields of view greater than 180°. So, even if a single nonlinear projection $\mathbb{R}^3 \to \mathbb{R}^2$ would suffice, for treatise simplicity and historical reasons makes sense to separate it into two components.

1.3.2 Brown-Conrady distortion

The Brown-Conrady model of lens distortion composes different distortion effects as polynomial terms of the projection image point. A full model comprises radial distortion, decentering distortion and thin-prism distortion.

All of them express the point with respect to the transformation fixed point, the so-called *center of distortion* (x_d, y_d). A common approximation have this point coincident with the principal point $(u_d, v_d) = (u_0, v_0)$, i.e. $(\tilde{u}_d, \tilde{v}_d) = (x_d, y_d) = (0, 0)$.

Having $(\tilde{x}, \tilde{y}) = (x, y) - (x_d, y_d)$ and $r = \|(\tilde{x}, \tilde{y})\|$, the contributions δ_r (radial), δ_d (decentering) and δ_p (thin-prism) can be written as, respectively (Eq. 1.8–10):

$$\delta_r = \begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} \left(1 + \sum_{i=1}^{n_r} k_i r^{2i} \right) \tag{1.8}$$

$$\delta_d = \begin{pmatrix} p_1(r^2 + 2\tilde{x}^2) + 2p_2\tilde{x}\tilde{y} \\ p_2(r^2 + 2\tilde{y}^2) + 2p_1\tilde{x}\tilde{y} \end{pmatrix} \left(1 + \sum_{\substack{i=3 \\ j=i-2}}^{n_d} p_i r^{2j} \right) \tag{1.9}$$

$$\delta_p = \sum_{i=1}^{n_p} \left(\begin{pmatrix} s_{2i-1} \\ s_{2i} \end{pmatrix} r^{2i} \right) \tag{1.10}$$

and the final distorted point on the normalized image plane as:

$$(\tilde{u}, \tilde{v}) = (x, y) + \delta_r + \delta_d + \delta_p \tag{1.11}$$

1.3.3 Division and rational models

Claus and Fitzgibbon propose the rational model [77] for generic cameras, predated by the division model [78] by Fitzgibbon himself. The two models deal with radial distortion, acting on the radii with a multiplicative factor expressed by Eq. 1.12 and Eq. 1.13, respectively:

$$\frac{1 + \sum_{i=1}^{n_n} a_i r^{2i}}{1 + \sum_{i=1}^{n_d} b_i r^{2i}} \tag{1.12}$$

$$\frac{1}{1 + \lambda r} \tag{1.13}$$

having λ, $a_1, \dots, a_{n_n}, b_1, \dots, b_{n_d}$ scalar parameters. Despite being convenient to treat mathematically there is more risk of overfitting, caused by the doubled number of parameters with respect to a same-order Maclaurin polynomial.

1.3.4 Classical fisheye mapping functions

Fisheye lenses are divided in different categories, depending on the mapping function the lens maker takes as a reference – these are called projection functions. Each function maps the angle with the optical axis, θ, to the radius of the point on the normalized image plane, r, as depicted in Fig. 1.4. In this context Brown-Conrady distortion is usually omitted, so the projection image plane (x,y) and the normalized image plane $(\tilde{u},\tilde{v})$ coincide.

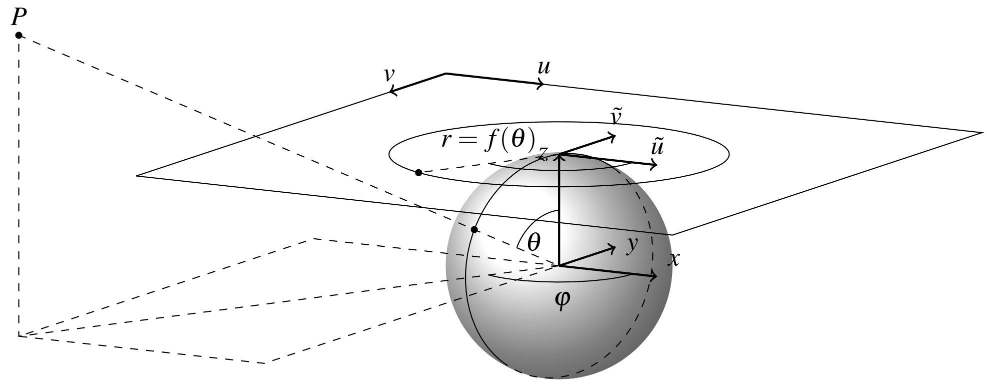


Figure 1.4: Lens distortion model representation

In this context pinhole projection is referred to as *gnomonical*. Most common mappings are listed in Table 1.2 and plotted in Figure 1.5.

Usually these equations are equivalently indicated multiplied by the focal parameter: we chose to separate this information, incorporating the focal into matrix K. Being $f_r : \theta \mapsto r$ one of these radial mappings, we can write the projection function f_p as per Eq. 1.14:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} {}^{c}X \\ {}^{c}Y \end{pmatrix} \frac{f_r\left(\operatorname{atan2}\left(\left\|\begin{smallmatrix} {}^{c}X \\ {}^{c}Y \end{smallmatrix}\right\|, {}^{c}Z\right)\right)}{\left\|\begin{smallmatrix} {}^{c}X \\ {}^{c}Y \\ {}^{c}Z \end{smallmatrix}\right\|} \tag{1.14}$$

Special care may be needed in the computation of the right-hand-side radii ratio to avoid numerical instability.

Projection type	Mapping function
gnomonical	$r = \tan(\theta)$
equidistant	$r = \theta$
stereographic	$r = 2\tan\left(\frac{1}{2}\theta\right)$
equisolid	$r = 2\sin\left(\frac{1}{2}\theta\right)$
ortographic	$r = \sin(\theta)$

Table 1.2: Mapping functions under different projection types

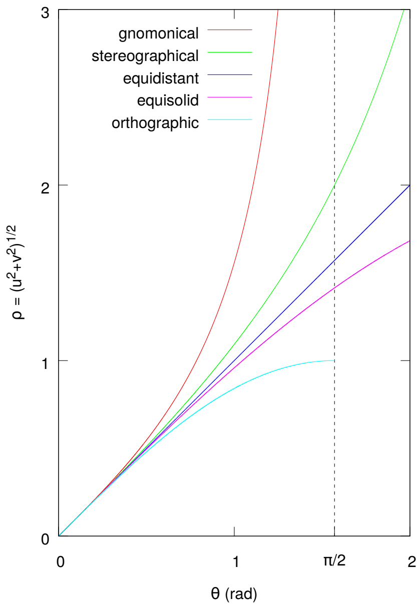


Figure 1.5: Mapping functions plotted in different colors

While gnomonical projection, as well as stereographic and ortographic, maps circles into ellipses, this is not true for the other ones: circles become quartic curves under equisolid projection, while for equidistant the result is even non-polynomial [79]. Table 1.3 shows the rings marker detector performance under different camera mappings. The test outlines how marker location accuracy does not degrade very much with stronger nonlinearities. The removal of image distortion and noise leaves a mean absolute error of $(0.029, 0.023)$ on the marker location that we attribute to ellipses border quantization.

These models often do not take into account lenses imperfections, as done by the Brown-Conrady model. An exception is made in the model by Kanatani [39], expressed in Eq. 1.15, introducing a polynomial expansion around the already-exisiting

Camera model	$\overline{\lvert\varepsilon_x\rvert}$	$\overline{\lvert\varepsilon_y\rvert}$
Pinhole (without distortion nor noise)	0.029	0.023
Pinhole (Brown-Conrady)	0.049	0.039
Fisheye (equidistant)	0.057	0.062
Fisheye (equisolid)	0.061	0.067

Table 1.3: Marker location error components with different camera models (using aforementioned method)

nonlinearity f_r taken from Table 1.2.

$$\sum_i k_i \left(r/\gamma_0\right)^{2i+1} = \frac{1}{\gamma_0} f_r(\theta) \tag{1.15}$$

In our field most common mappings act from the three-dimensional world space to the image space, so in a similar fashion, omitting the normalizer γ_0 and with different meaning and values for parameters k_i, we could write Eq. 1.16 to replace polynomial root finding with evaluation and gain some computation speedup.

$$r = \sum_i k_i \left(f_r(\theta)\right)^{2i+1} \tag{1.16}$$

1.3.5 Kannala-Brandt model

Attempts have been tried to provide a unique model, alternative to the mapping function f_p, for both the pinhole and one or more fisheye projections. The proposal of Kannala and Brandt [80] is based on the expression of the function by expansion in Maclaurin series:

$$r(\theta) = \sum_{h=1}^{n_{rf}} k_h \theta^{2h-1} \tag{1.17}$$

This is similar to the radial distortion treatise by Brown-Conrady seen in 1.3.2, but applied to a function of θ instead of the undistorted radius r (viz $\tan(\theta)$ in the pinhole case). To take into account nonradial lens distortions they propose, instead of modelling all physical phenomena, to add two distortion terms δ_r and δ_t, respectively in the radial and tangential directions, separable functions of θ and φ.

$$\delta_r(\theta,\varphi) = \left(\sum_{h=1}^{n_{r\theta}} l_h \theta^{2h-1}\right)\left(\sum_{h=1}^{n_{r\varphi}} i_{2h-1}\cos(h\varphi) + i_{2h}\sin(h\varphi)\right) \tag{1.18}$$

$$\delta_t(\theta,\varphi) = \left(\sum_{h=1}^{n_{t\theta}} m_h \theta^{2h-1}\right)\left(\sum_{h=1}^{n_{t\varphi}} j_{2h-1}\cos(h\varphi) + j_{2h}\sin(h\varphi)\right) \tag{1.19}$$

Having u_r and u_t the radial and tangential components of (x,y) around the distortion center (x_d, y_d), the distorted point $(\tilde{u},\tilde{v})$ can be written as:

$$(\tilde{u},\tilde{v}) = (r(\theta) + \delta_r(\theta,\varphi))\, u_r + \delta_t(\theta,\varphi) u_t \tag{1.20}$$

1.3.6 Catadioptric-fisheye unified model

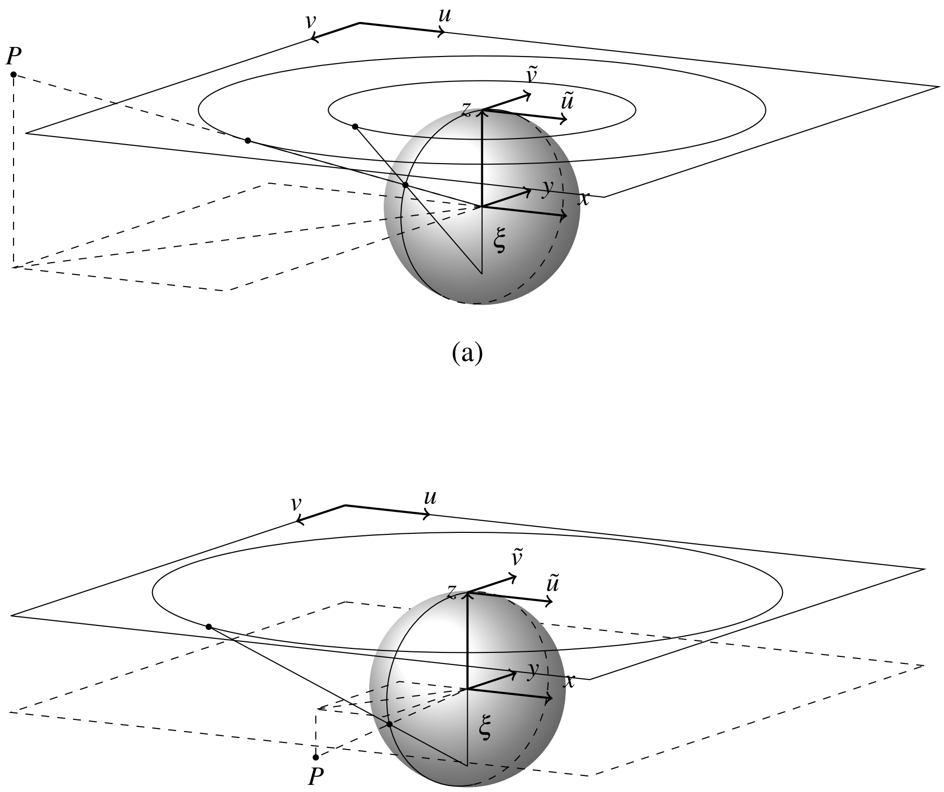
In the same aim of trying to provide a unique model, Ying and Hu [81] proposed to use the models of catadioptric cameras. Mei provided a calibration procedure for this kind of model [51]. The model is similar to the undistorted pinhole one, with the addition of a nonlinear translation term $\xi \|P\| e_z$ before the perspective projection, having $P = \begin{pmatrix} {}^C X \\ {}^C Y \\ {}^C Z \end{pmatrix}$ the point in camera coordinates, ξ a parameter and e_z the third canonical basis versor. In other words the mapping equation 1.6 becomes:

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \end{pmatrix} = \frac{1}{{}^C Z + \xi \left\| \begin{matrix} {}^C X \\ {}^C Y \\ {}^C Z \end{matrix} \right\|} \begin{pmatrix} {}^C X \\ {}^C Y \end{pmatrix} \tag{1.21}$$

The effect of this transformation is depicted in Fig. 1.6.

Besides the trivial case $\xi = 0$ corresponding to a pinhole mapping, one can show that Eq. 1.21 can express exactly the stereographic ($\xi = 1$) as well as orthographic ($\xi = +\infty$) projections.
While the latter does not allow $\theta \geq \frac{\pi}{2}$ (field of view greater than 180°), the former does, but the proof of equivalence involves the trigonometric identity

$$\tan\left(\frac{\theta}{2}\right) = \frac{\tan(\theta)}{1 + \sqrt{1 + \tan^2(\theta)}} \tag{1.22}$$



(b)

Figure 1.6: Catadioptric/fisheye model.
(a) From the outer to the inner one, the circles on the image plane correspond to the pinhole case ($\xi = 0$) and fisheye case with $0 < \xi < 1$, respectively.
(b) The model can handle points behind the camera ($\theta > \frac{\pi}{2}$)

which is valid only in $\left(-\frac{\pi}{2},\frac{\pi}{2}\right)$. Outside the interval the model does not take into account the change of sign of $\cos(\theta)$ and nothing is but an approximation of the mapping function.

Proof of the equivalence of the models. In the stereographic case it is enough to write the expression of $\tan(\theta)$:

$$\tan(\theta) = \frac{\|{}^{C}_{C}{}^{X}_{Y}\|}{{}^{C}Z} \tag{1.23}$$

put it into Eq. 1.22 and factor up $\frac{1}{{}^{C}Z}$:

$$r_{\text{stereographic}} = 2\tan\left(\frac{\theta}{2}\right) = 2\frac{\frac{1}{{}^{C}Z}\left\|\begin{matrix}{}^{C}X\\{}^{C}Y\end{matrix}\right\|}{1+\sqrt{\frac{1}{{}^{C}Z^2}\left({}^{C}Z^2+\left\|\begin{matrix}{}^{C}X\\{}^{C}Y\end{matrix}\right\|^2\right)}}$$

$$\overset{{}^{C}Z\geq 0}{=} 2\frac{\frac{1}{{}^{C}Z}\left\|\begin{matrix}{}^{C}X\\{}^{C}Y\end{matrix}\right\|}{\frac{1}{{}^{C}Z}\left({}^{C}Z+1\cdot\left\|\begin{matrix}{}^{C}X\\{}^{C}Y\\{}^{C}Z\end{matrix}\right\|\right)} \propto \left\|\begin{matrix}x\\y\end{matrix}\right\|_{\xi=1}$$

where the proportionality factor can be absorbed in the focal lengths.

As for the orthographic case, we shall start from the catadioptric-fisheye model and work in homogeneous coordinates to project the point

$$\begin{pmatrix}{}^{C}X_P & {}^{C}Y_P & {}^{C}Z_P+\xi\,\|{}^{C}P\|\end{pmatrix} \tag{1.24}$$

on the plane ${}^{C}Z = \xi + 1$, so that we obtain the 2D point coordinates:

$$\begin{pmatrix}x & y\end{pmatrix}_{\xi} = \begin{pmatrix}{}^{C}X_P\frac{\xi+1}{{}^{C}Z_P+\xi\|P\|} & {}^{C}Y_P\frac{\xi+1}{{}^{C}Z_P+\xi\|P\|}\end{pmatrix} \xrightarrow{\xi\to+\infty} \begin{pmatrix}{}^{C}X & {}^{C}Y\end{pmatrix}/\|P\| \tag{1.25}$$

that is the same of an ortographic projection, being

$$\left\|x \quad y\right\|_{\xi=+\infty} = \left\|{}^{C}X \quad {}^{C}Y\right\|/\|P\| = \sin(\theta)$$

□

1.4 Calibration pattern distance

Another question arising in camera calibration is: what should be the optimal distance of the target from the camera? It is well known [17] that multiple grid orientations are needed for a correct calibration. Apart from that, there is no easy answer, especially if we abstract away the camera model. Common practice is to take multiple images, at different orientations and different distances. While it is difficult to properly handle noise, something can be said, at least intuitively, on why taking grids at multiples distances may improve calibration accuracy.

We can safely assume the accuracy of the printed grid does not change with distance. Taking as a reference the pinhole projection for simplicity, it is easy to see that the farther the grid, the lesser the uncertainty effect on the image plane. Unfortunately we also have quantization effects in the image plane, preventing arbitrary increase in accuracy. Furthermore some parameters, like translation component of relative extrinsics in pinhole stereo cameras, show opposite behaviour, increasing the accuracy as distance decreases: we attribute this to the reduction of quantization effects over disparity values.

Chapter 2

Planar pattern pose suggestion

2.1 Motivation

Estabilished a calibration procedure with a certain planar pattern, under zero-mean σ-covariance gaussian marker location noise, we expect convergence (asymptotic, excluded overfitting effects) of the model parameters values towards the real ones, of the average (signed) error towards zero and of the absolute error towards a half-gaussian distribution, having mean $\sigma\sqrt{\frac{2}{\pi}}$.

Common practice consists in the use of more images than needed for the calibration, together with a test set for results validation. Taking a test set can be difficult and not exempt from bias. Moreover, after some thousands of marker observations, optimization running times slow down. You can either select the frames manually, but this will limit their number and requires a user who must be trained to recognize bad frames from better ones, or take a whole sequence of images, letting the detector discard frames. With a significant number of points contributing to the objective function optimization can require days.

In general, this is not a problem: accurate intrinsics parameters calibration is done only once, offline, and extrinsics parameters calibration can be done after that with a reduced number of points. Unfortunately, things may go wrong: the detector may fail and produce misplaced points, illumination may not be the best one (producing artifacts on the grid), motion of the grid may cause image blurring, the image poses may bias the result towards certain values. In all this cases you don't want to wait for days to discover that you have to start again from the beginning.

Some help could be found with active markers, i.e. markers that are not printed but instead generated on a display shown to the camera. This does not solve all the problems: computing the accuracy of the marker to provide second order moment information may be trickier and you may still need to change the camera-display relative position.

Some years ago another idea came up to help users in the calibration procedure. Instead of acquiring a lot of images from different poses you want to take only a few of them, with the requirement they have to be *good* ones, i.e. they must provide a reliable and accurate camera calibration. An augmented reality display shows where to put the calibration grid to the user, making the entrire calibration approach interactive. This is the idea developed some years ago by Richardson et al. with the creation of the AprilCal library [32]: a Java piece of software for assisted camera calibration based on their own fiducial markers (AprilTags [48]).

This chapter has been heavily influenced by the aforementioned work. In the next section we will outline the general idea and identify a possible improvement.

2.2 Pose goodness evaluation

Finding out a pose to be suggested to the user can be modeled essentially as a function optimization problem. Given a utility function $f_s : M \times \Omega \to \mathbb{R}$ we want to provide as a suggestion the pose $p^* \in \Omega \subset SE(3)$ with maximum utility, given current camera model parameters estimate $\hat{\mu} \in M$:

$$p^* = \operatorname*{argmax}_{p \in \Omega} f_s(\hat{\mu}, p) \tag{2.1}$$

The camera calibration algorithm can thus be expressed iteratively (Alg. 2).

Algorithm 2 Assisted camera calibration algorithm

$I_1, \ldots, I_k \leftarrow$ *take a few images for algorithm initialization, extract keypoints*
$\hat{\mu} \leftarrow \texttt{calibrate}(\{I_1, \ldots, I_k\}, \mu_0)$
for i **from** $k+1$ **to** n **do**
    $\overline{\varepsilon} \leftarrow \texttt{average\_reprojection\_error}\,(\hat{\mu}, \{I_1, \ldots, I_{i-1}\})$
    if $\|\overline{\varepsilon}\| \leq$ threshold **then break**; **end if**
    $p^* \leftarrow \underset{p \in \Omega}{\operatorname{argmax}}\, f_s(\hat{\mu}, p)$
    $I_i \leftarrow$ *take image near* p^**, extract keypoints*
    $\hat{\mu} \leftarrow \texttt{calibrate}(\{I_1, \ldots, I_i\}, \hat{\mu})$
end for

Even if calibration (`calibrate` as from Alg. 1) is required at each iteration, the choice of the pose will enable the reaching of the required accuracy with a small number of images. Furthermore the nonlinear optimization can start from the previous parameters guess, allowing faster convergence. The difficulty lies in choosing the function f_s, or the update of p^*, to be representative of poses that can reliably provide accurate camera calibrations.

Fiducial markers represent a unique 3D orientation. However we see how there is no need to disambiguate pattern symmetries: traditional calibration patterns such as checkerboards or circles grids do not need such information during calibration, neither does this approach. Replacing fiducial markers with other marker types, such as the ring markers introduced in sec. 1.2, allows to start from lower marker location errors.

We are then going to show some ideas for defining such function, together with some test results.

2.2.1 Sampled expected reprojection error maximization

In developing AprilCal, Richardson et al. point out that usage of the mean reprojection error or mean squared error as indicators of calibration quality is problematic: even when low, calibration quality can result poor. Their proposal consists in minimizing the maximum expected reprojection error, computed by sampling from the posterior distribution over the model parameters.

The update of the best pose, performed over a quantized version of the domain $\Omega \subset SE(3)$, proceeds as follows. For each candidate pose we want to find a value, i.e. define the function f_s. This is done with a Monte Carlo method: for each candidate pose, samples are generated by perturbation of the target extrinsics. For each sample calibration is performed and reprojection error is computed by averaging the target points mean errors over perturbations of the newly-found camera parameters. The estimate of the marginal posterior covariance of the model parameters $P(\mu|I_0,\dots,I_n)$ is computed by marginalization of the target extrinsics poses $p_0,\dots,p_n$ (Eq. 2.2).

$$P(\mu|I_0,\dots,I_n) = \int \cdots \int_{p_0,\dots,p_n \in \Omega^n} P(\mu, p_0,\dots,p_n|I_0,\dots,I_n)\, dp_0 \cdots dp_n \tag{2.2}$$

The best pose is the one with the maximum expected reprojection error. In order to initialize properly the distribution estimates a bootstrapping of 3 images is taken: the first one frontal to the camera and the other two suggested through the calibration of a reduced model.

Calibration is done with an iterative gradient descendt:

$$J^{\intercal}\Sigma_z^{-1} J \Delta x = J^{\intercal}\Sigma_z^{-1} r \tag{2.3}$$

$$x_{i+1} = x_i + \Delta x \tag{2.4}$$

The higher the iterations number is the harder the algorithm will try to reach the required accuracy of the solution at the expense of computational time. Moreover high computational costs limit the discretization of the suggestions search space: the cost of a single residual computation has to be multiplied by the number of points per grid times the number of grids in the search space (bruteforce search). In the original paper a 5×5 grid is used, with a grid of approximatively 1000 grids in the pose space Ω. While 1000 poses may seem a lot to sample from, having the pose space 6 degrees of freedom only few possibilities per degree of freedom are available, thus greatly limiting the selection and making the grid “coarse”, as outlined by AprilCal authors themselves.

We investigated whether some speed could be gained by changing the utility function, trying to keep the accuracy and speed and to refine the grid in the pose space; equivalently, one could keep the same pose space size and have shorter computational times.

2.2.2 Efficient estimation optimality criteria

The evaluation of the calibration suggestion performance, as for camera calibration itself, is done with statistical methods. Maximizing the uncertainty of the observations indeed gives the grid with the maximal information content. Another approach could be looking for the grid that allows us to be as sure as possible of the to-be-found camera parameters, i.e. minimizing the uncertainty of the parameters. Surely we would like the camera parameters cross-correlation matrix Σ_μ to be small, but its coefficients may not decrease with the same speed: it is not easy to find a solution. One option could be minimize a functional of its elements, e.g. one of its element-wise norms, like the Frobenius one. Instead of taking arbitrary functions, we tried to exploit a statistical interpretation: in statistics the minimization of a covariance matrix in order to maximize the Fisher information of the estimator is a well-known problem. Traditionally a solution was searched among functionals of the eigenvalues of the information matrix (or, equivalently, of the covariance matrix Σ_μ). However, few of the common criteria suited our context. Variables in the parameters vector μ have different dimensionalities and so have their variances: this removes physical meaning from certain operations. As an example, we liked *E-optimality* (maximization of minimum eigenvalue, i.e. minimization of inverse's spectral radius) as a worst-case optimization, but the difference in dimensionalities prevents the comparison. The same argumentation was applied for *A-optimality* (maximization of covariance matrix trace $\mathrm{tr}(\Sigma_\mu^{-1})$) and, equivalently, *T-optimality* (minimization of information matrix trace). *C-optimality* minimizes the variance using a predefined linear combination of the eigenvalues. This surely works with an appropriate weighting, but we considered its definition too much application-specific: it depends on which parameters you want to optimize more aggressively, while we are looking for criteria that do not depend on downstream processing.

2.2.3 Parameters uncertainty minimization

It is possible to optimize the eigenvalues of Σ_μ as a whole minimizing their product. This is known as the *D-optimality*, being the product of the eigenvalues the determinant of the matrix, and it is a popular solving criterion for statistic information maximization.

The covariance matrix can be computed from $\Sigma_\mu^{-1} = J^\top \Sigma_z^{-1} J$. To avoid the matrix inversion, one can exploit the fact that eigenvalues of the inverse are the eigenvalues inverses:

$$\begin{aligned} \min \det \Sigma_\mu &= \max \left(\det \Sigma_\mu\right)^{-1} = \max \det \Sigma_\mu^{-1} \\ &= \max \det J^\top \Sigma_z^{-1} J \approx \max \det J^\top J \end{aligned} \tag{2.5}$$

having in the last step the approximation of the inverse covariance matrix with the design matrix $J^\top J$.

2.2.4 Maximum predicted residuals variance minimization

A popular statistical criterion is *G-optimality*: it consists in minimizing the maximum entry in the diagonal of the system's *projection matrix*. It has the effect of minimizing the maximum variance of the predicted values. Given a linear system observation affected by additive gaussian noise ν:

$$z = Hx + \nu \tag{2.6}$$

the projection matrix (sometimes called *influence* matrix or *hat* matrix) P allows to map response values z into fitted values $\hat{z}$:

$$Pz = \hat{z} = H\hat{x} \tag{2.7}$$

When the observation weights are identical and the errors uncorrelated we have Eq. 2.8 and we can trivially find an expression of P from H, as per Eq. 2.9.

$$H^{\intercal} H \hat{x} = H^{\intercal} z \tag{2.8}$$

$$P = H \left(H^{\intercal} H\right)^{-1} H^{\intercal} \tag{2.9}$$

In our specific case the roles of the observation vector z and of the observation matrix H are played respectively by the residuals vector r and the jacobian matrix J.

2.2.5 Maximum likelyhood reprojection error minimization

Sticking to an idea more similar to the one of AprilCal paper, we can compute a maximum likelyhood estimate of the reprojection error through $P(z_0, \ldots, z_n | \mu)$, thus avoiding the perturbation of the pose parameters and the expensive sample-wise recalibration. As for the maximum a posteriori, both the mean or the maximum of the single reprojection errors could be taken: we have, respectively, the *maximum likelyhood mean reprojection error minimization* and *maximum likelyhood maximum reprojection error minimization* approaches.

The statistics (mean and variance) of the distribution of perturbed intrinsic parameters have been obtained with the Unscented Transform instead of the Monte Carlo approach, reducing in this way the number of samples. Furthermore the sigma-points have been precomputed for additional speedup.

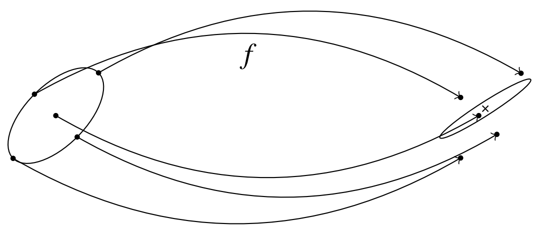


Figure 2.1: Unscented transform
Individual points (sigma-points) are transformed according to the nonlinearity f and used to reconstruct the statistics of the transformed distribution

2.3 Results

We performed some qualitative tests, based on plotting the parameters trend with increasing number of poses, on the convergence of the algorithm towards the model parameters values under different suggestion optimizations. The algorithm was run with the camera parameters from Table 2.1, where two sets of camera parameters are available:

- "Real" camera parameters were used only as a ground truth and for generating the noiseless keypoints. To ensure plausibility they were obtained calibrating a real camera with a preexisting algorithm.

- "Starting point" camera parameters were used to initialize the nonlinear optimization algorithm: a manual setup was created to avoid a good result by the linear optimizer that is usually employed for a first guess, we expect a performance increase when having a starting point obtained from linear optimization.

Real camera parameters	Starting point camera parameters
Intrinsics:	Intrinsics:
• $k_u = k_v = 535.17539043$	• $k_u = k_v = 1000$
• $u_0 = 635.87852568$	• $u_0 = 640$
• $v_0 = 488.40054881$	• $v_0 = 480$
Brown-Conrady:	Brown-Conrady:
• $k_1 = -0.23554278$	• $k_1 = 0$
• $k_2 = 0.05994505$	• $k_2 = 0$
• $k_3 = -0.00973610$	• $k_3 = 0$
• $k_4 = 0.00090471$	• $k_4 = 0$
• $k_5 = -0.00004364$	• $k_5 = 0$
• $k_6 = 0.00000084$	• $k_6 = 0$

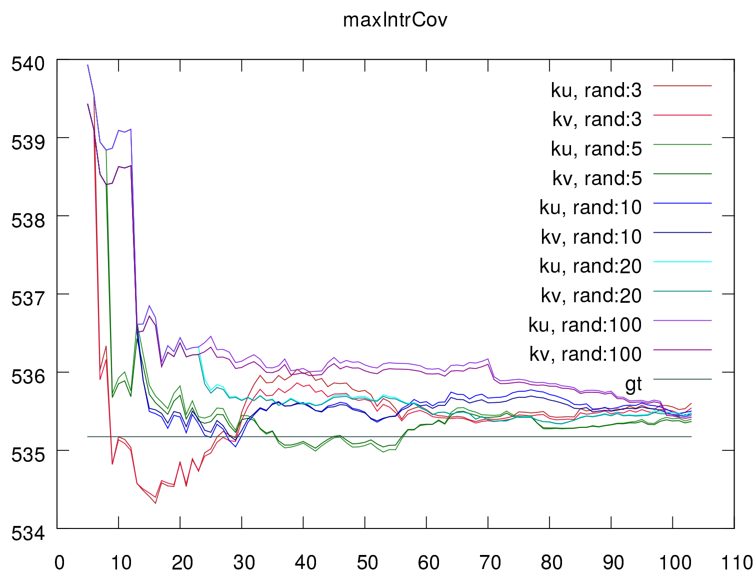
Table 2.1: Assisted camera calibration algorithm test parameters, having a pinhole camera with radial Brown-Conrady distortion as reference model

No ground truth ("real") parameters would have been available in a real use-case, so the experiments were performed by simulation, generating synthetic data resembling the real one through the addition of noise. A marker location noise variance of 0.05 px was considered.

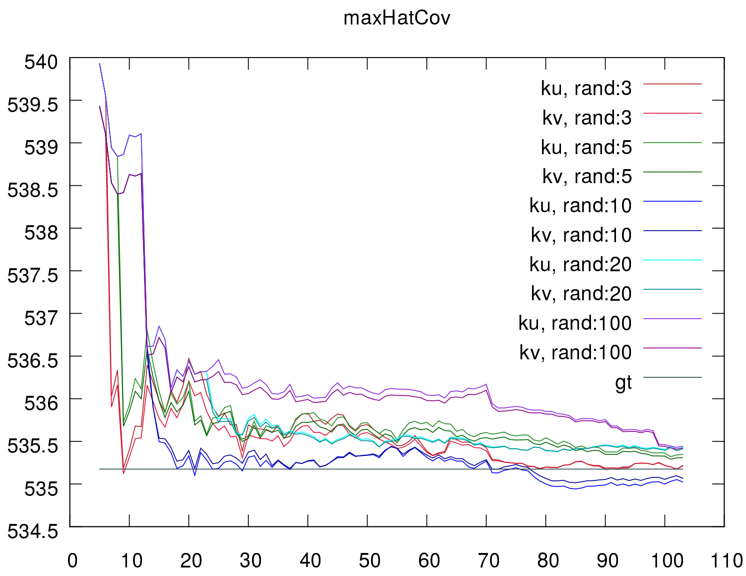
A pinhole camera model with Brown-Conrady radial distortion having a degree 11 polynomial was taken as a reference. Levenberg-Marquardt algorithm iterations limit was set to 100 and function value tolerance (relative minimal cost step) to 10^{-12}, so that the most common reason of algorithm termination in experimental evidence results the reach of parameter accuracy tolerance value (fixed at 10^{-8}). The simulated image sequence has been generated to depict grids consisting in 10×8 ring markers, with inter-marker step of 0.1 cm, both horizontally and vertically. The plotted data display one hundred of added poses, even if fewer are needed and would be added in a real system, to outline the estimated parameters trend. With more poses the solution may present an improvement due to the information increase, as well as a worsening due to the noise.

With the criteria proposed in paragraphs 2.2.3–2.2.5 we were able to conduct tests in the order of few minutes on an Intel® Core™ i7-4790 processor, even if we brought the number of poses in the search space Ω from approximatively 1000 to 196000 and no special care was taken in optimizing the code. The camera-model relative pose space has been defined from the Tait-Bryan rotations and cartesian translations. Each cartesian translational component t_x, t_y, t_z has been quantized in 10 equidistributed steps, with $t_x \in [-0.7, 0.7]$, $t_y \in [-0.7, 0.7]$ and $t_z \in [1.5, 3.5]$; both yaw and pitch angles have been quantized in 14 equidistributed steps in $[-30°, 30°]$ while the roll angle was held constant and null. The pose space parameters were chosen such that no calibration grid point could be placed behind the camera: such condition is necessary to ensure convergence of the optimizer in the pinhole case and will always be satisfied in a real scenario.

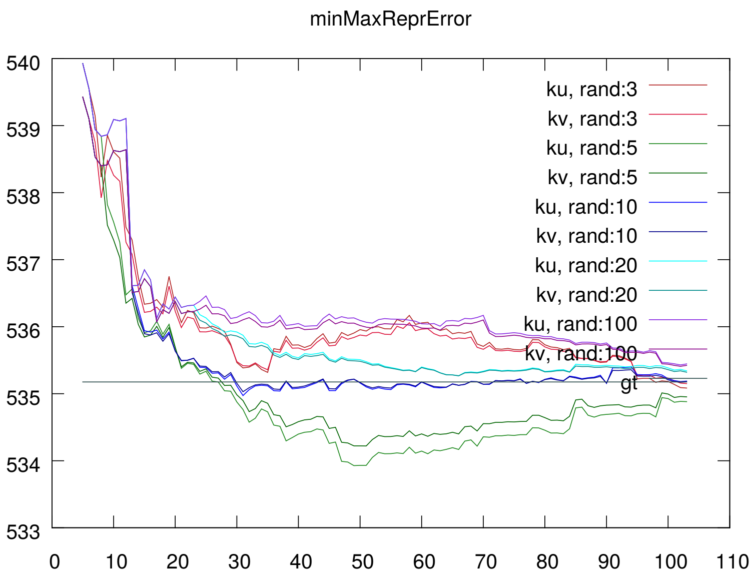
Some results are plotted in Fig. 2.2, depicting focal lengths and absolute errors as functions of number of added grids, with variable number of starting random grids for initialization. The results for the first five poses are omitted from the graphs to avoid out-of-scale numbers. All the simulations stopped after the insertion of 100 grids, whose poses were chosen either randomly or suggested by the proposed criteria. The plot trends show how the criteria are roughly equivalent and generally better than a full-random selection (purple lines in the figure). The small difference between the obtained error at convergence and the reference value, being the former slightly lower than the latter, has been attributed to model overfitting.



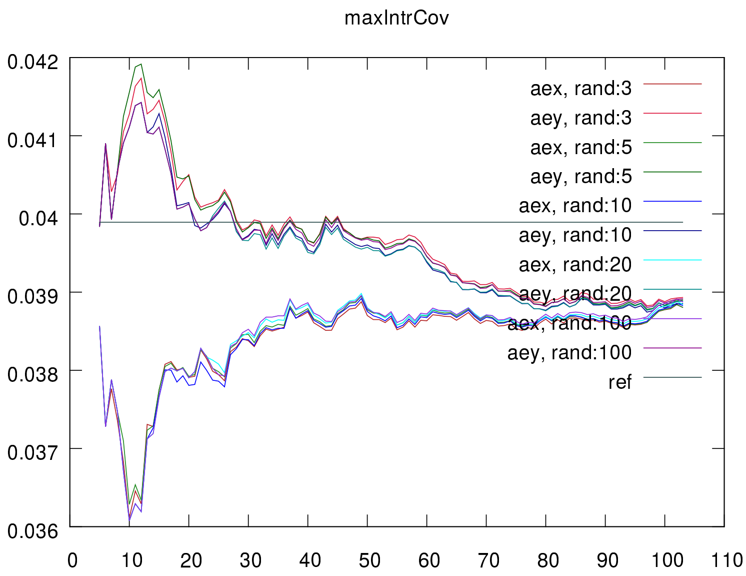
(a)


maxHatCov
ku, rand:3
kv, rand:3
ku, rand:5
kv, rand:5
ku, rand:10
kv, rand:10
ku, rand:20
kv, rand:20
ku, rand:100
kv, rand:100
gt
540
539.5
539
538.5
538
537.5
537
536.5
536
535.5
535
534.5
0
10
20
30
40
50
60
70
80
90
100
110

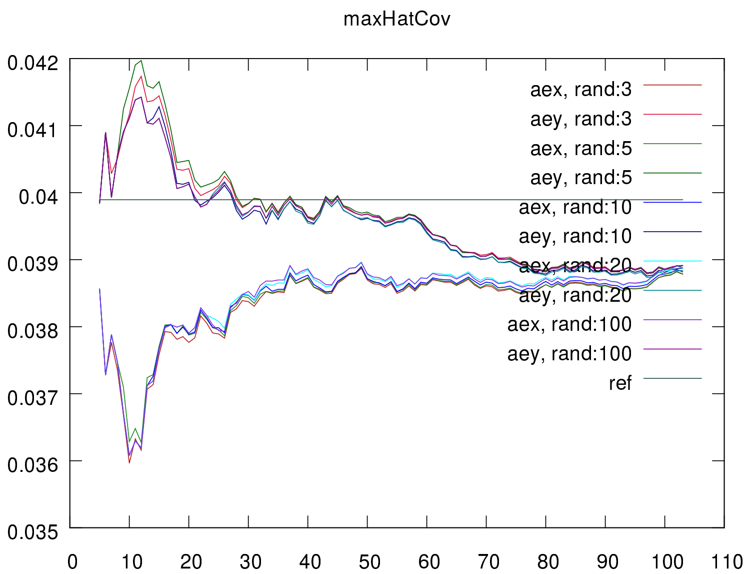
(b)


minMaxReprError
ku, rand:3
kv, rand:3
ku, rand:5
kv, rand:5
ku, rand:10
kv, rand:10
ku, rand:20
kv, rand:20
ku, rand:100
kv, rand:100
gt
540
539
538
537
536
535
534
533
0
10
20
30
40
50
60
70
80
90
100
110

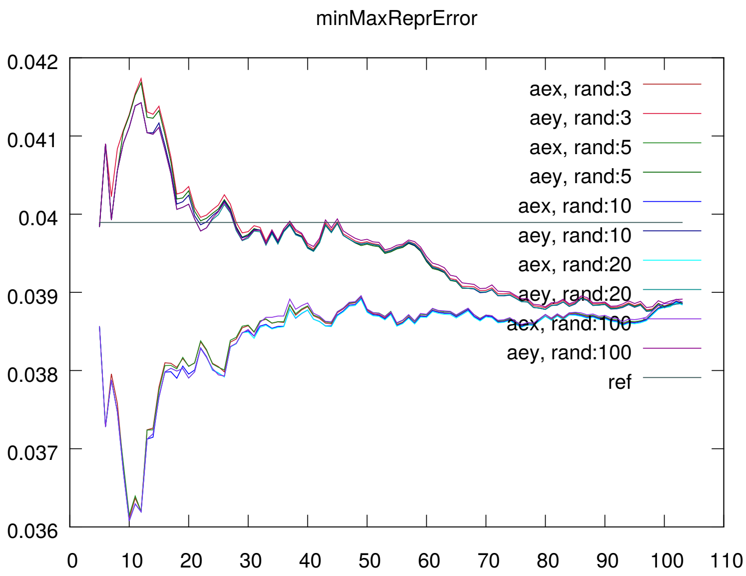
(c)


maxIntrCov
0.042
0.041
0.04
0.039
0.038
0.037
0.036
0
10
20
30
40
50
60
70
80
90
100
110
aex, rand:3
aey, rand:3
aex, rand:5
aey, rand:5
aex, rand:10
aey, rand:10
aex, rand:20
aey, rand:20
aex, rand:100
aey, rand:100
ref

(d)


maxHatCov
0.042
0.041
0.04
0.039
0.038
0.037
0.036
0.035
0
10
20
30
40
50
60
70
80
90
100
110
aex, rand:3
aey, rand:3
aex, rand:5
aey, rand:5
aex, rand:10
aey, rand:10
aex, rand:20
aey, rand:20
aex, rand:100
aey, rand:100
ref

(e)



(f)

Figure 2.2: Grid pose suggestion performance.
Each pair of lines represents a test having a different number of initialization random poses (3, 5, 10, 20 and 100 initialization poses were used).
(a-c) represent the values of the optimized focals, “gt” is the ground-truth value.
(d-f) show the reprojection errors, where “ref” represents the mean of the half-gaussian noise distribution $\sigma\sqrt{2/\pi}$
maxIntrCov: maximization of camera intrinsics parameters covariance inverse matrix.
maxHatCov: minimization of predicted residuals maximal variance.
maxReprErrAvg: minimization of maximum likelyhood maximum reprojection error

Chapter 3

Conclusions and future work

Finally, we would like to sum up this thesis work and draw some conclusions. We will also exploit the gained knowledge to try to forsee what the next role of camera calibration and advances in the topic will be.

3.1 What has been done

The camera resectioning problem has been analyzed. It has been introduced as a problem that can be considered solved for most applications, but for the higher-demanding ones (big scale factors) and in the presence of strong nonlinear distortions more accuracy would be desirable.

Literature solutions have been explored, paying specific attention to the highest accuracy ones. The problem of incongruity between accuracy and trueness has been outlined: achieving accurate values might not be a clue of physical soundness of the model. The principal ideas behind nonlinear lens distortion modelling have been extensively described, going through polynomial distortions, well-known mapping projection functions and the mixed catadioptric-fisheye model. General camera calibration issues bound to the procedure itself have been pointed out, such as the distance of the calibration pattern from the camera and the number of images to be taken.

We spotlighted assisted calibration pose suggestion and looked for faster alternatives to the algorithms available in literature. Estimators optimality criteria and matrix operations properties have been exploited to obtain computationally cheap ways to evaluate poses, allowing to have a denser pose space. In detail, the determinant of the design matrix has been maximized to obtain minimal variance parameters; the maximum predicted residuals variance has been minimized through the computation of the influence matrix of the system; finally, sigma-points maximum likelyhood reprojection error has been maximized. With every method you should be able to obtain computational times similar to those of the a-posteriori reprojection error maximization original approach, even having a pose space of increased size. As a future work, being the approach driven by experimental evidence, surely more testing needs to be done to strengthen the obtained results.

3.2 What may be done

The major cause of accuracy loss in marker detection has been attributed to image spatial quantization. Although the results will surely benefit from the currently increasing camera resolution trend, it would be interesting to know if we can push the accuracy further. Even if difficult, it may be worth spending some time in inserting the quantization error into the model. Trying to provide a ground truth for the camera-pattern relative poses (if possible), so that only the camera parameters would remain as problem variables, could help in distinguishing the errors sources. We expect the research community to meld the aspects of machine-learned detection and world scene from image measuring, with mixed techniques that take the best from both worlds, making camera-based computer vision systems more accurate and autonomous.

Bibliography

[1] G. Montone, J. K. O'Regan, and A. V. Terekhov. Unsupervised model-free camera calibration algorithm for robotic applications. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 3058–3063, Sept 2015. doi:10.1109/IROS.2015.7353799.

[2] E. E. Hemayed. A survey of camera self-calibration. In *Advanced Video and Signal Based Surveillance, 2003. Proceedings. IEEE Conference on*, pages 351–357, July 2003. doi:10.1109/AVSS.2003.1217942.

[3] Irvine C. Gardner and Frank A. Case. Precision camera for testing lenses. *Journal of Research, National Bureau of Standards*, 18:449–460, Apr 1937. URL: http://nvlpubs.nist.gov/nistpubs/jres/18/jresv18n4p449_A1b.pdf.

[4] Irwin Sobel. On calibrating computer controlled cameras for perceiving 3-d scenes. In *Proceedings of the 3rd International Joint Conference on Artificial Intelligence*, IJCAI'73, pages 646–657, San Francisco, CA, USA, 1973. Morgan Kaufmann Publishers Inc. URL: http://dl.acm.org/citation.cfm?id=1624775.1624853.

[5] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, August 1987. doi:10.1109/JRA.1987.1087109.

[6] J. K. Kearney, Xiaoli Yang, and Shenzhi Zhang. Camera calibration using geometric constraints. In *Computer Vision and Pattern Recognition, 1989. Proceedings CVPR '89., IEEE Computer Society Conference on*, pages 672–679, Jun 1989. doi:10.1109/CVPR.1989.37918.

[7] R. Lenz and R. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3d machine vision metrology. In *Proceedings. 1987 IEEE International*

Conference on Robotics and Automation, volume 4, pages 68–75, Mar 1987. doi:10.1109/ROBOT.1987.1088012.

[8] K. D. Gremban, C. E. Thorpe, and T. Kanade. Geometric camera calibration using systems of linear equations. In *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*, pages 562–567 vol.1, Apr 1988. doi:10.1109/ROBOT.1988.12111.

[9] R. Safaee-Rad, K. C. Smith, B. Benhabib, and I. Tchoukanov. An analytical method for the 3d-location estimation of circular features for an active-vision system. In *Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on*, pages 215–220, Nov 1990. doi:10.1109/ICSMC.1990.142095.

[10] G. P. Stein. Accurate internal camera calibration using rotation, with analysis of sources of error. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 230–236, Jun 1995. doi:10.1109/ICCV.1995.466781.

[11] K. Kanatani and T. Maruyama. Optimal focal length calibration system. In *Intelligent Robots and Systems '93, IROS '93. Proceedings of the 1993 IEEE/RSJ International Conference on*, volume 3, pages 1816–1821 vol.3, Jul 1993. doi:10.1109/IROS.1993.583882.

[12] Sheng-Wen Shih, Yi-Ping Hung, and Wei-Song Lin. When should we consider lens distortion in camera calibration. *Pattern Recognition*, 28(3):447 – 461, 1995. doi:http://dx.doi.org/10.1016/0031-3203(94)00107-W.

[13] G. P. Stein. Lens distortion calibration using point correspondences. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 602–608, Jun 1997. doi:10.1109/CVPR.1997.609387.

[14] C. Geyer and K. Daniilidis. Catadioptric camera calibration. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 398–404 vol.1, 1999. doi:10.1109/ICCV.1999.791248.

[15] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 1106–1112, Jun 1997. doi:10.1109/CVPR.1997.609468.

[16] J. Batista, H. Araujo, and A. T. de Almeida. Iterative multistep explicit camera calibration. *IEEE Transactions on Robotics and Automation*, 15(5):897–917, Oct 1999. doi:10.1109/70.795794.

[17] Zhengyou Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 666–673 vol.1, 1999. `doi:10.1109/ICCV.1999.791289`.

[18] D. Guo, L. Wu, J. Wang, X. Zheng, and Q. Li. Use the gps/imu new technology for photogrammetric application. In *2006 IEEE International Symposium on Geoscience and Remote Sensing*, pages 1107–1110, July 2006. `doi:10.1109/IGARSS.2006.286`.

[19] L. Zhang, Q. Li, D. Guo, Y. He, and Y. Yan. Evaluation of gps/ imu supported aerial photogrammetry. In *2006 IEEE International Symposium on Geoscience and Remote Sensing*, pages 1512–1514, July 2006. `doi:10.1109/IGARSS.2006.390`.

[20] M. Fleps, E. Mair, O. Ruepp, M. Suppa, and D. Burschka. Optimization based imu camera calibration. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3297–3304, Sept 2011. `doi:10.1109/IROS.2011.6095067`.

[21] G. G. Scandaroli, P. Morin, and G. Silveira. A nonlinear observer approach for concurrent estimation of pose, imu bias and camera-to-imu rotation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3335–3341, Sept 2011. `doi:10.1109/IROS.2011.6094702`.

[22] Qilong Zhang and Robert Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2301–2306. IEEE, 2004.

[23] D. Claus and A. W. Fitzgibbon. Reliable automatic calibration of a marker-based position tracking system. In *Application of Computer Vision, 2005. WACV/MOTIONS '05 Volume 1. Seventh IEEE Workshops on*, volume 1, pages 300–305, Jan 2005. `doi:10.1109/ACVMOT.2005.101`.

[24] Hyon Lim and Young Sam Lee. Real-time single camera slam using fiducial markers. In *ICCAS-SICE, 2009*, pages 177–182, Aug 2009.

[25] J. Jun, Q. Yue, and Z. Qing. An extended marker-based tracking system for augmented reality. In *Modeling, Simulation and Visualization Methods (WMSVM), 2010 Second International Conference on*, pages 94–97, May 2010. `doi:10.1109/WMSVM.2010.52`.

[26] Z. Zhang and Q. Tang. Camera self-calibration based on multiple view images. In *2016 Nicograph International (NicoInt)*, pages 88–91, July 2016. `doi:10.1109/NicoInt.2016.16`.

[27] A. E. Abderrahmani and K. Satori. Camera self-calibration with varying intrinsic parameters and arc of the circle. In *2016 SAI Computing Conference (SAI)*, pages 1309–1314, July 2016. `doi:10.1109/SAI.2016.7556149`.

[28] I. El Batteoui, K. Satori, and A. Saaidi. A stereo planar self-calibration method of a camera with variable intrinsic parameters. In *2016 13th International Conference on Computer Graphics, Imaging and Visualization (CGiV)*, pages 6–11, March 2016. `doi:10.1109/CGiV.2016.11`.

[29] Jianping Xu and Fang Deng. A camera self-calibration method based on ios-pso. In *Chinese Automation Congress (CAC), 2015*, pages 489–494, Nov 2015. `doi:10.1109/CAC.2015.7382550`.

[30] G. Fuhr and C. Jung. Camera self-calibration based on non-linear optimization and applications in surveillance systems. *IEEE Transactions on Circuits and Systems for Video Technology*, PP(99):1–1, 2015. `doi:10.1109/TCSVT.2015.2511812`.

[31] V. Rudakova and P. Monasse. Camera matrix calibration using circular control points and separate correction of the geometric distortion field. In *2014 Canadian Conference on Computer and Robot Vision*, pages 195–202, May 2014. `doi:10.1109/CRV.2014.34`.

[32] Andrew Richardson, Johannes Strom, and Edwin Olson. AprilCal: Assisted and repeatable camera calibration. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1814–1821, November 2013. `doi:10.1109/IROS.2013.6696595`.

[33] C. Chen and Y. F. Zheng. A new robotic hand/eye calibration method by active viewing of a checkerboard pattern. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 770–775 vol.2, May 1993. `doi:10.1109/ROBOT.1993.291941`.

[34] Mark Fiala and Chang Shu. Self-identifying patterns for plane-based camera calibration. *Machine Vision and Applications*, 19(4):209–216, 2008. `doi:10.1007/s00138-007-0093-z`.

[35] Jun-Sik Kim, P. Gurdjos, and In-So Kweon. Geometric and algebraic constraints of projected concentric circles and their applications to camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):637–642, April 2005. doi:10.1109/TPAMI.2005.80.

[36] Yihong Wu, Haijiang Zhu, Zhanyi Hu, and Fuchao Wu. Camera calibration from the quasi-affine invariance of two parallel circles. In Tomás Pajdla and Jiří Matas, editors, *Computer Vision - ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part I*, pages 190–202. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. doi:10.1007/978-3-540-24670-1_15.

[37] Xiaoqiao Meng and Zhanyi Hu. A new easy camera calibration technique based on circular points. *Pattern Recognition*, 36(5):1155 – 1164, 2003. doi:10.1016/S0031-3203(02)00225-X.

[38] Fengmei Sun. Planar conic based camera calibration. In *Proceedings of the International Conference on Pattern Recognition - Volume 1*, ICPR '00, pages 1555–, Washington, DC, USA, 2000. IEEE Computer Society. URL: http://dl.acm.org/citation.cfm?id=876866.877409.

[39] K. Kanatani. Calibration of ultrawide fisheye lens cameras by eigenvalue minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):813–822, April 2013. doi:10.1109/TPAMI.2012.146.

[40] Mi Zhang, Jian Yao, Menghan Xia, Kai Li, Yi Zhang, and Yaping Liu. Line-based multi-label energy optimization for fisheye image rectification and calibration. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4137–4145, June 2015. doi:10.1109/CVPR.2015.7299041.

[41] L. Alvarez, A. Salgado, and J. Sánchez. Robust detection and ordering of ellipses on a calibration pattern. *Pattern Recognition and Image Analysis*, 17(4):508–522, 2007. doi:10.1134/S1054661807040098.

[42] Chang Shu, Alan Brunton, and Mark Fiala. A topological approach to finding grids in calibration patterns. *Machine Vision and Applications*, 21(6):949–957, 2010. doi:10.1007/s00138-009-0202-2.

[43] A. Bevilacqua, A. Gherardi, and L. Carozza. Automatic perspective camera calibration based on an incomplete set of chessboard markers. In *Computer Vision, Graphics Image*

Processing, 2008. ICVGIP '08. Sixth Indian Conference on, pages 126–133, Dec 2008. doi:10.1109/ICVGIP.2008.10.

[44] Jiewei Sun, Yisong Chen, and Guoping Wang. Computable relation graph based calibration pattern extraction algorithm on multiple calibration boards. In *Proceedings of the 27th Conference on Image and Vision Computing New Zealand*, IVCNZ '12, pages 227–231, New York, NY, USA, 2012. ACM. doi:10.1145/2425836.2425883.

[45] John Mallon and Paul F. Whelan. Which pattern? biasing aspects of planar calibration patterns and detection methods. *Pattern Recogn. Lett.*, 28(8):921–930, June 2007. doi:10.1016/j.patrec.2006.12.008.

[46] Yves Nievergelt. Perturbation analysis for circles, spheres, and generalized hyperspheres fitted to data by geometric total least-squares. *Mathematics of Computation*, 73(245):169–180, 2004. URL: http://www.jstor.org/stable/4099863.

[47] Yves Nievergelt. Fitting conics of specific types to data. *Linear Algebra and its Applications*, 378:1 – 30, 2004. doi:10.1016/j.laa.2003.08.022.

[48] Edwin Olson. AprilTag: A robust and flexible visual fiducial system. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3400–3407. IEEE, May 2011. doi:10.1109/ICRA.2011.5979561.

[49] Carlos Ricolfe-Viala and Antonio-José Sánchez-Salmerón. Correcting non-linear lens distortion in cameras without using a model. *Optics & Laser Technology*, 42(4):628 – 639, 2010. doi:10.1016/j.optlastec.2009.11.002.

[50] R. I. Hartley and Sing Bing Kang. Parameter-free radial distortion correction with centre of distortion estimation. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1834–1841 Vol. 2, Oct 2005. doi:10.1109/ICCV.2005.184.

[51] Christopher Mei and Patrick Rives. Single View Point Omnidirectional Camera Calibration from Planar Grids. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3945–3950, Rome, Italy, April 2007. IEEE. URL: https://hal.inria.fr/hal-00767674, doi:10.1109/ROBOT.2007.364084.

[52] Luis Puig, J. Bermúdez, Peter Sturm, and J.J. Guerrero. Calibration of omnidirectional cameras in practice: A comparison of methods. *Computer Vision and Image Understanding*, 116(1):120 – 137, 2012. Virtual Representations and Modeling of Large-scale Environments (VRML). doi:10.1016/j.cviu.2011.08.003.

[53] B. Khomutenko, G. Garcia, and P. Martinet. An enhanced unified camera model. *IEEE Robotics and Automation Letters*, 1(1):137–144, Jan 2016. `doi:10.1109/LRA.2015.2502921`.

[54] Weijia Feng, Juha Röning, Juho Kannala, Xiaoning Zong, and Baofeng Zhang. A general model and calibration method for spherical stereoscopic vision. In *Proc. SPIE 8301, Intelligent Robots and Computer Vision XXIX: Algorithms and Techniques*, volume 8301, pages 830107–830107–8, 2012. `doi:10.1117/12.907071`.

[55] K. Wang, L. Zhao, and R. Li. Fisheye omnidirectional camera calibration – pinhole or spherical model? In *Robotics and Biomimetics (ROBIO), 2014 IEEE International Conference on*, pages 873–877, Dec 2014. `doi:10.1109/ROBIO.2014.7090442`.

[56] Ying, Xianghua and Mei, Xiang and Yang, Sen and Wang, Ganwen and Rong, Jiangpeng and Zha, Hongbin. Imposing differential constraints on radial distortion correction. In Daniel Cremers, Ian Reid, Hideo Saito, and Ming-Hsuan Yang, editors, *Computer Vision – ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part I*, pages 384–398. Springer International Publishing, Cham, 2015. `doi:10.1007/978-3-319-16865-4_25`.

[57] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, Nov 2000. `doi:10.1109/34.888718`.

[58] Fuyu Huang, Yongzhong Wang, Xueju Shen, Chao Lin, and Yudan Chen. Method for calibrating the fisheye distortion center. *Appl. Opt.*, 51(34):8169–8176, Dec 2012. URL: `http://ao.osa.org/abstract.cfm?URI=ao-51-34-8169`, `doi:10.1364/AO.51.008169`.

[59] R. G. von Gioi, P. Monasse, J. M. Morel, and Z. Tang. Lens distortion correction with a calibration harp. In *2011 18th IEEE International Conference on Image Processing*, pages 617–620, Sept 2011. `doi:10.1109/ICIP.2011.6116626`.

[60] G. Bradski. OpenCV – Open Computer Vision Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[61] Bradley Atcheson, Felix Heide, and Wolfgang Heidrich. CALTag: High precision fiducial markers for camera calibration. In 15th *International Workshop on Vision, Modeling and Visualization*, Siegen, Germany, November 2010.

[62] Zygmunt L Szpak, Wojciech Chojnacki, and Anton Van Den Hengel. Guaranteed ellipse fitting with the sampson distance. In *European Conference on Computer Vision*, pages 87–100. Springer, 2012.

[63] Andrew Fitzgibbon, Maurizio Pilu, and Robert B. Fisher. Direct least square fitting of ellipses. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(5):476–480, May 1999. `doi:10.1109/34.765658`.

[64] Radim Halíř and Jan Flusser. Numerically stable direct least squares fitting of ellipses. In *Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization. WSCG*, volume 98, pages 125–132. Citeseer, 1998.

[65] A. J. Tabatabai and O. R. Mitchell. Edge location to subpixel values in digital imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(2):188–201, March 1984. `doi:10.1109/TPAMI.1984.4767502`.

[66] R. Safaee-Rad, I. Tchoukanov, B. Benhabib, and K.C. Smith. Accurate parameter estimation of quadratic curves from grey-level images. *CVGIP: Image Understanding*, 54(2):259 – 274, 1991. `doi:10.1016/1049-9660(91)90067-Y`.

[67] Janne Heikkila. Moment and curvature preserving technique for accurate ellipse boundary detection. In *Proceedings of the 14th International Conference on Pattern Recognition-Volume 1 - Volume 1*, ICPR '98, pages 734–, Washington, DC, USA, 1998. IEEE Computer Society. URL: `http://dl.acm.org/citation.cfm?id=839289.842275`.

[68] Jun-Sik Kim and In-So Kweon. A new camera calibration method for robotic applications. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 2, pages 778–783 vol.2, 2001. `doi:10.1109/IROS.2001.976263`.

[69] Jun-Sik Kim, Ho-Won Kim, and In So Kweon. A camera calibration method using concentric circles for vision applications. *ACCV2002, Melbourne, Australia*, 2002.

[70] Francisco Abad, Emilio Camahort, and Roberto Vivó. Camera calibration using two concentric circles. In Aurélio Campilho and Mohamed Kamel, editors, *Image Analysis and Recognition: International Conference ICIAR 2004, Porto, Portugal, September 29- October 1, 2004, Proceedings, Part I*, pages 688–696. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. `doi:10.1007/978-3-540-30125-7_85`.

[71] Guang Jiang and Long Quan. Detection of concentric circles for camera calibration. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 333–340 Vol. 1, Oct 2005. `doi:10.1109/ICCV.2005.73`.

[72] Ashutosh Morde, Mourad Bouzit, and Lawrence R. Rabiner. A novel approach to planar camera calibration. In Alpesh Ranchordas, Helder Araújo, and Bruno Encarnação, editors, *VISAPP 2006: Proceedings of the First International Conference on Computer Vision Theory and Applications, 2 Volumes, Setúbal, Portugal, February 25-28, 2006*, pages 87–92. INSTICC - Institute for Systems and Technologies of Information, Control and Communication, 2006.

[73] Minh Vo, Zhaoyang Wang, Long Luu, and Jun Ma. Advanced geometric camera calibration for machine vision. *Optical Engineering*, 50(11):110503–110503–3, 2011. `doi:10.1117/1.3647521`.

[74] John Greenlees Semple and Geoffrey Thomas Kneebone. *Algebraic projective geometry*. Oxford science publications. Clarendon, cop. 1952, Oxford, 1952 [printed in 1979, reprinted in 1998].

[75] Duane C. Brown. Decentering distortion and the definitive calibration of metric cameras. In *Photogrammetric Engineering*, pages 444–462. The American Society of Photogrammetry Convention, may 1965. URL: `https://eserv.asprs.org/PERS/1966journal/may/1966_may_444-462.pdf`.

[76] A. Conrady. Decentered lens systems. In *Royal Astronomical Society Monthly Notices*, volume 79, pages 384–390, 1919.

[77] A. W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–125–I–132 vol.1, 2001. `doi:10.1109/CVPR.2001.990465`.

[78] D. Claus and A. W. Fitzgibbon. A rational function lens distortion model for general cameras. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 213–219 vol. 1, June 2005. `doi:10.1109/CVPR.2005.43`.

[79] J. Bermudez-Cameo, G. Lopez-Nicolas, and J. J. Guerrero. Automatic line extraction in uncalibrated omnidirectional cameras with revolution symmetry. *International Journal of Computer Vision*, 114(1):16–37, 2015. `doi:10.1007/s11263-014-0792-7`.

[80] J. Kannala and S. S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8):1335–1340, Aug 2006. `doi:10.1109/TPAMI.2006.153`.

[81] Xianghua Ying and Zhanyi Hu. Can we consider central catadioptric cameras and fish-eye cameras within a unified imaging model. In Tomás Pajdla and Jiri Matas, editors, *ECCV (1)*, volume 3021 of *Lecture Notes in Computer Science*, pages 442–455. Springer, 2004. `doi:10.1007/978-3-540-24670-1_34`.