Non-parametric Models of Distortion in Imaging Systems

by

Pradeep Ranganathan

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2016

Doctoral Committee:

Associate Professor Edwin Olson, Chair
Assistant Professor Laura K. Balzano
Assistant Professor Honglak Lee
Associate Professor Clayton D. Scott

Acknowledgements

This dissertation is a product of the thoughts and suggestions of many people. I would like to express my sincere gratitude to everyone who made the research in this dissertation possible.

I would like to thank my labmates Johannes Strom, Ryan Morton, Robert Goeddel, Andrew Richardson, John Wang, Pratik Agarwal and Lauren Hinkle for the many useful interactions that helped me learn, both at a personal level and professionally. The experience of working on the MAGIC competition with my labmates was a big jump start to my understanding of perception systems. In particular, my many discussions with Andrew Richardson on projective geometry, cameras and lenses helped shape some key ideas in this dissertation.

It would have been terribly difficult to go through the ups and downs of my graduate studies without the support of my dear friends. I would like to thank Vishnu Prabhu, Vishnu Prakash Mani, Divya Sanghvi and Swathi Maddali for being a constant source of support and happiness; Ananda Narayanan for the many inspiring and useful technical discussions on AI and software; Senthil Kumar Vasudevan and Rajkamal Natarajan for helping me be a better human being. I would like to specifically thank Priyanka Pramanik and Sriram Vaidyanathan for their time and effort in taking care of me during my sickness and hospitalization. This dissertation would have taken a lot longer to complete without their support. This list does not mention many other friends who have influenced my graduate life directly or indirectly in their own special way.

I would like to thank my advisor Prof. Edwin Olson for his continuous support throughout my graduate studies. His advice helped shape many of the research directions in this thesis. His style of research and presentation has had an obvious and big influence on my technical abilities. I would also like to thank my dissertation committee members Prof. Laura Balzano, Prof. Honglak Lee and Prof. Clayton Scott for their support and encouragement.

At a personal level, I would like to thank my mother, Josephine Malar, my father, Ranganathan Vaidyanathan, and my sister, Preethi, for supporting my decision to pursue a Ph.D. It is undoubtedly the environment that they created at our home that made me interested in science and engineering in the first place.

Contents

List of Figures

Abstract

Traditional radial lens distortion models are based on the physical construction of lenses. However, manufacturing defects and physical shock often cause the actual observed distortion to be different from what can be modeled by the physically motivated models.

In this work, we initially propose a Gaussian process radial distortion model as an alternative to the physically motivated models. The non-parametric nature of this model helps implicitly select the right model complexity, whereas for traditional distortion models one must perform explicit model selection to decide the right parametric complexity.

Next, we forego the radial distortion assumption and present a completely non-parametric, mathematically motivated distortion model based on locally-weighted homographies. The separation from an underlying physical model allows this model to capture arbitrary sources of distortion. We then apply this fully non-parametric distortion model to a zoom lens, where the distortion complexity can vary across zoom levels and the lens exhibits noticeable non-radial distortion.

Through our experiments and evaluation, we show that the proposed models are as accurate as the traditional parametric models at characterizing radial distortion while flexibly capturing non-radial distortion if present in the imaging system.

Chapter 1

Introduction

To extract metric information such as object size or depth from images one requires a calibrated camera model. Thus, a calibrated camera model is the corner stone of algorithms used for image registration and image-based spatial reasoning. At a high-level, the camera calibration procedure optimizes a parametric camera model to best explain a set of observed correspondences between world and image points.

For applications in computer vision and photogrammetry, it is common to use the projective pinhole camera model to model the camera as a system of rays. This simple pinhole model, however, does not account for the distortion caused by the camera lens. Lens distortion is predominantly radial and hence camera models are augmented with a radial lens distortion model to compensate for lens effects. The calibration procedure for the augmented camera is then modified to estimate the parameters of the lens distortion along with the other camera parameters.

Usually, lens distortion is assumed to be well modeled by a parametric family of functions. Examples of parametric distortion models are described in [9] and include polynomial, field-of-view, rational function, and division models. The particular choice of parametric family used is based on the type of lens being modeled. In reality, lenses may not conform to any one of the proposed parametric distortion models. In this situation, a rigorous approach for choosing the best parametric model is to perform model selection on multiple calibrated lens distortion models as done, for example, by Melegy and Farag in [13].

In some cases, even for lenses with the usual axis aligned construction, the radial model is insufficient. Manufacturing errors can cause deviations from the intended distortion. Physical shock can disrupt the alignment of lens components and produce distortions that are not radial, as explained by Brown in [6]. Thus, a radial distortion model that is motivated by the physical construction of lenses

cannot capture non-radial sources of distortion that often arise in practice.

In this work, we propose non-parametric models of lens-distortion as alternatives to parametric distortion models. Unlike parametric models, non-parametric models are flexible because they can automatically expand or simplify model complexity to explain the observed distortion.

The first non-parametric model that we describe is a Gaussian process (GP) model of radial distortion. The traditional camera calibration procedure is a non-linear least-squares model fitting procedure and uses a parametric description of lens distortion. It is not immediately obvious as to how one can incorporate a GP, which has no explicit parameters, into a framework that optimizes parametric models. To achieve this, we show how the non-linear least-squares model fitting can be viewed as maximum-likelihood inference in a factor graph with Gaussian factor potentials and then incorporate a GP model into this factor graph inference framework. This method adds great flexibility to the factor graph optimization formulation, while preserving much of its computational efficiency.

This non-parametric GP radial distortion model can automatically choose the most appropriate radial distortion model; but it is still limited by the physically motivated assumption of radial distortion. Our next model removes this radial distortion assumption and produces a truly flexible lens distortion model.

Our second non-parametric model of lens distortion is based on locally-weighted homographies. A locally-weighted homography is a novel, non-parametric, non-linear homography formulation that can map lines on the source plane to arbitrary smooth curves on the target plane. We adapt this non-linear homography to build a model of lens distortion by characterizing lens distortion as deviation from the homography at the center of the image. We use this estimate of lens distortion to bootstrap camera calibration and show that it improves the stability of the classic camera calibration.

In our experiments, we also propose a rigorous evaluation strategy that evaluates the performance of camera models on a set of test images. The use of separate training and testing datasets might seem obvious to readers with a machine learning background. Nonetheless, this has not been a standard practice in camera modeling tasks[1].

We structure the remainder of this work as follows: In the next chapter, we review other related work before presenting relevant technical background in Chap. 3.

---

[1]However, literature in the related field of optics and optical modeling report testing errors when evaluating camera models. For example see [27].

Next, we present our Gaussian process radial distortion model in Chap. 4 and the locally-weighted homography based lens distortion model in Chap. 5. We then apply the locally-weighted homography distortion model to characterize the distortion across zoom levels of a zoom lens in Chap. 6 and conclude with a discussion of our contributions in Chap. 7.

Chapter 2

Related work

Camera calibration techniques generally fall into two broad categories: *photogrammetric calibration* and *self-calibration*. Photogrammetric calibration uses a calibration target with a precisely known 3D structure like in the work of Heikkilä [18]. Frahm and Koch describe an alternative procedure that uses a planar target undergoing controlled motion in [14]. In contrast, self-calibration methods, such as the work of Maybank et al., [20], use multiple views of the same scene and exploit the rigidity of the scene to calibrate the camera parameters.

Photogrammetric and self-calibration methods use the same lens distortion models; only the calibration procedures are different. In our work, we choose to implement photogrammetric calibration since it is the more mature technique. A classic reference for the reader interested in the history of camera calibration methods is the work by Clarke and Fryer [8].

Zhang, in his landmark paper [32], was the first to present a technique that used multiple views of a single planar calibration target to calibrate a camera. This technique is very popular because it requires only a planar calibration target that is easily manufactured. Many popular open-source calibration toolkits, such as the OpenCV calibration toolkit [5] and Bouguet's Matlab calibration toolkit [19], are based on this technique.

Calibrating a lens distortion model in a photogrammetric or self-calibration setting falls under the technique of *metric lens calibration*. Alternatively, one can exploit the fact that straight lines in the scene must project to straight lines, in order to correct lens distortion in a *non-metric* fashion, as done in [12, 26]. In our work, we use metric lens calibration, since it is the natural choice for photogrammetric calibration methods.

Non-parametric lens distortion models: Prior to our work, other non-parametric models have been proposed in the literature. Hartley and Kang use a locally averaged estimate of observed distortion to build a non-parametric distortion model in [16]. In their work [26], Ricolfe-Viala and Sanchez-Salmeron explore the use of model-free distortion estimation in a non-metric lens calibration setting. Our work differs from these approaches in that we use a GP as the non-parametric distortion model. This gives our method the advantage that the smoothness of the resulting distortion function is well determined. It also gives our model the ability to capture the uncertainty in the distortion function estimate.

Non-linear homography models: The technique of a *lifted homography* has been used to model non-linearity in cameras by Claus and Fitzgibbon in [10], Barreto et al., in [2], and Gasparini et al., in [15]. This formulation projects the input coordinates into a higher-dimensional feature space (*lifting*) and then constructs a homography in this higher-dimensional space, thus producing a non-linear mapping. By construction, lifted homographies are parametric with respect to the size of the higher-dimensional space, and they are geometrically unintuitive.

In contrast, the non-linear locally-weighted homography proposed in this work is non-parametric. The technique retains geometric intuition because its mapping action can be interpreted in terms of a local linear homography. This formulation of a non-parametric non-linear homography is a novel idea to the best of the author's knowledge. Its ability is not limited by a choice of parametric model for lens distortion, and it has the capability to model arbitrary sources of distortion. More interestingly, it produces an independent estimate of the distortion without relying on any prior knowledge of the intrinsics or extrinsics of the camera used to acquire the image.

In essence, the non-metric *straight lines are straight* rectification procedure as described by Devernay and Faugeras in [12] serves a similar purpose to our method. However, our method operates in terms of homographies; therefore, it implicitly encodes projective constraints.

Distortion models for zoom lenses: In Chap. 6 of this work, we extend the locally-weighted homography base lens distortion model to characterize distortion in a zoom lens. Compared to the use of fixed focal-length lenses, zoom lenses are seldom mentioned in the research literature. This is because zoom lenses offer no inherent advantage for most traditional vision tasks, where prior knowledge of

the task allows one to select a single appropriate focal-length. However, they are indispensable in applications such as depth from defocus and surveillance.

Agapito, Hayman and Reid provide a solution to the problem of calibrating a camera with varying intrinsics in [1]. Their method extends the self-calibration method of Hartley [17] to zoom cameras. However, they do not model any lens-distortion in their work.

The straightforward approach of calibrating a zoom camera at a few fixed settings and then using interpolation to predict camera parameters at intermediate settings is first mentioned and evaluated in [30] by Wilson. Later work in zoom camera calibration adapts this technique to pan-tilt-zoom camera calibration, like the work of Sinha and Pollefeys [28], or explores the use of a different radial distortion model to capture lens distortion across zoom levels, as in the work of Wu and Radke [31].

Our work is motivated primarily by the requirements of a calibration procedure for zoom lenses used on a robot, where the assumption of radially symmetrical distortion is seldom true. As mentioned previously, lenses used on robots are subject to physical shock that introduces non-radial distortion. The model for zoom-lens distortion proposed in this paper is flexible enough to capture both the radial and non-radial components of lens distortion.

Chapter 3

Preliminaries

In this chapter we introduce the technical background required to understand the material in subsequent chapters. We briefly review the perspective camera model, lens distortion models and the camera calibration procedure. We also describe the non-parametric regression techniques used in this work while comparing them to traditional parametric regression.

3.1   The Perspective Pin-hole Camera Model

Consider an object imaged by a pin-hole camera as shown in Figure 3.1.1. The rays from the object pass through the pin-hole and form an image on the sensor. All rays imaged by the camera must pass through this pin-hole and hence the pin-hole is the **center of projection**. The sensor is the **imaging plane**, the plane on which the points from the object are projected.

It is useful to note that the line through the pinhole and perpendicular to the sensor plane is the **principal axis** of the camera. By convention, the principal axis of the camera coincides with the z-axis of the system. The point at which the principal axis meets the imaging plane is the **principal point** or **image center**. Once again, by convention, the imaging plane coincides with the xy-plane and the principal point coincides with the origin of the camera coordinate system. The perpendicular distance between the imaging plane and the pin-hole is the **focal length** f.

It is useful to note that in practice a separate 2D pixel coordinate system (not shown in Fig. 3.1.1) is assigned to the imaging plane, where the origin, pixel location $(0, 0)$, is at the top-left of the imaging plane; the positive x-axis is oriented left to right, and the positive y-axis is oriented top to bottom. The process of calibrating
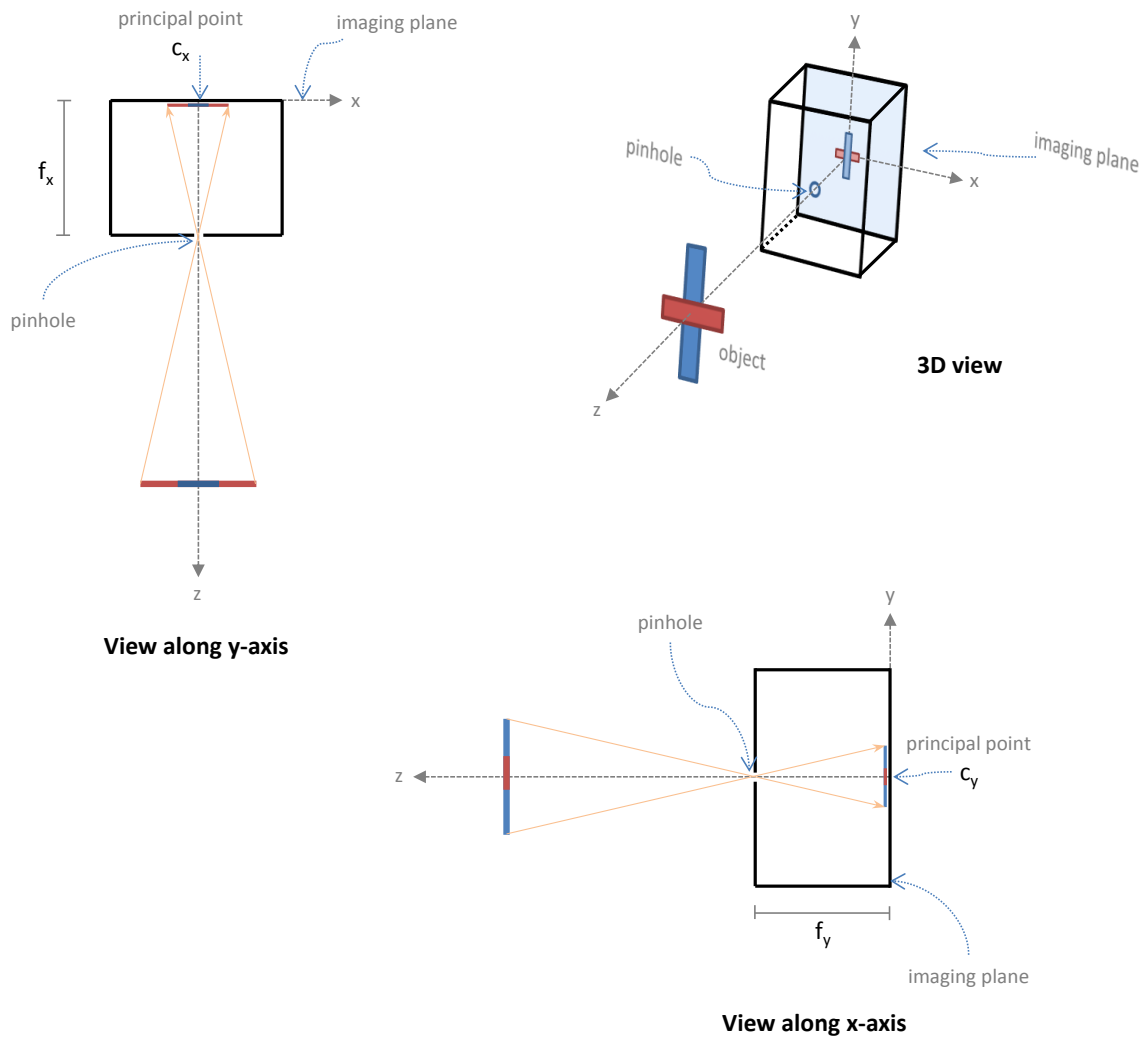
Figure 3.1.1: *The perspective pin-hole camera model. This figure shows an object being imaged by a pin-hole camera. Also shown, are the plan (top-down) and elevation (sideways) views of the imaging setup. The camera is placed with its **principal point** or **image center,** (c_x, c_y)**,** at the origin of the 3D space. The **principal axis** of the camera lies along the z-axis and the **imaging plane** lies along the xy-plane. The directional focal-lengths f_x and f_y are labeled in the projected 2D views.*
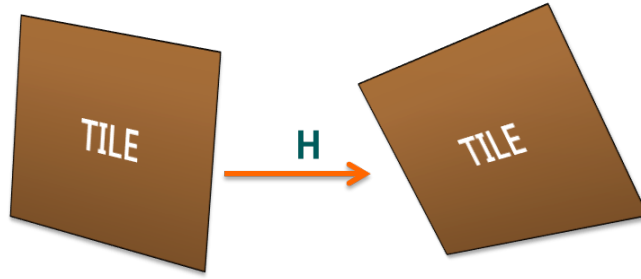
Figure 3.2.1: Planar perspective homography. The two perspective views of the tile are related by a homography. Note how the homography mapping preserves the straightness of lines: the straight-line alphabets in one view are mapped onto straight-line alphabets in the other view.

a camera model defines a mapping between 3D world coordinates and the 2D pixel coordinates of the image plane.

3.2   Planar Perspective Homography

The term homography means *similar drawing* and it describes the transformation between two perspective views of the same scene. Mathematically, it is an invertible linear transformation between two perspective planes. If a point $\mathbf{p}$ on one perspective plane is mapped by a homography $\mathbf{H}$ onto a point $\mathbf{q}$ on another perspective plane, we write

$$\mathbf{q} \sim \mathbf{Hp} \tag{3.2.1}$$

where $\mathbf{p}$ and $\mathbf{q}$ are homogeneous points and $\mathbf{H}$ is a matrix. In homogeneous 3D space

$$\mathbf{m} \triangleq \begin{bmatrix} m_x \\ m_y \\ 1 \end{bmatrix} \sim \begin{bmatrix} \lambda m_x \\ \lambda m_y \\ \lambda \end{bmatrix}, \lambda \neq 0$$

That is, points in homogeneous 3D space are determined up to scale: a point $\mathbf{m}$ represents the same point as $\lambda \mathbf{m}$. Thus, in Eq. 3.2.1 the point $\mathbf{q}$ is similar, and not equal, to $\mathbf{Hp}$.
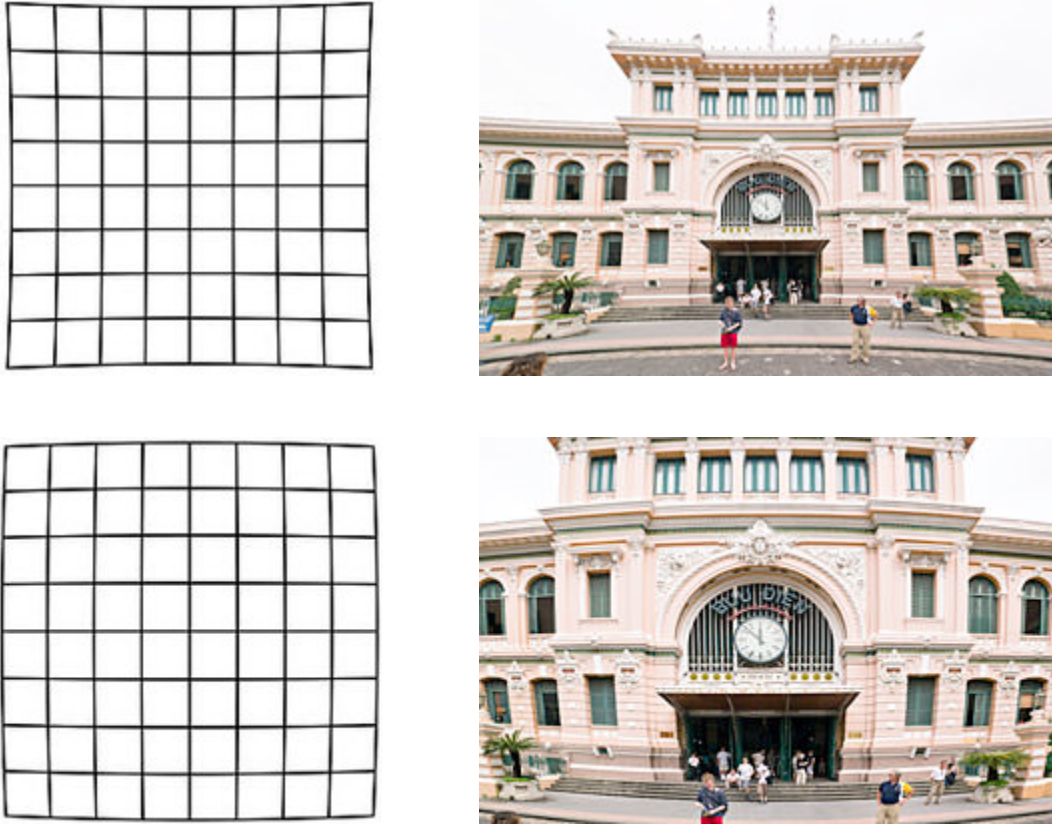
Figure 3.3.1: Examples of lens distortion. The use of a lens system in a camera produces distortions where straight-lines are rendered as curves. By construction, the distortion produced by the lenses are dominated by a radially symmetric component .

3.3 Lens distortion

Unlike a pinhole camera, a practical camera must use a large enough aperture to capture a sufficient amount of light within a specified amount of time. This large aperture requires the use of a lens system to focus incoming light rays onto the imaging plane. However, the use of a lens system distorts the straight-line path of incoming light-rays and produces a distorted image, as show in Fig. 3.3.1.

These lens systems consist of one or more lenses aligned along the principal axis of the camera. This alignment causes the distortion produced by the lens system to be radially-symmetric about the image center, with the amount of distortion increasing in magnitude as we move away from the image center. Thus, the distortion produced by lenses is predominantly *radial*. However, out-of-plane misalignments of the lens components can produce a small but significant distortion component that does not exhibit radial symmetry. This component is called *tangential distortion* and is described by Brown in [6].

10

3.4 Camera Calibration

In many tasks such as registering images and visual surveillance, we are interested in the relation between an object's size in pixels and its size in the real world. The process of camera calibration quantifies this relationship between pixel size and object size by finding optimal values for the parameters of the perspective pin-hole camera model. Once we have a calibrated model for a camera, and given an image of an object obtained with that camera, we can use the model to reason about how the object was placed relative to the camera.

Camera calibration is a model-fitting problem, where we have a set of ground-truth observations and we require model parameter settings that best explain these observations. At a high-level, the camera calibration procedure is as follows:

1. First, we obtain images of a planar printed target with known dimensions. These images will serve as our set of ground truth observations.

2. The points on the planar target, with coordinates expressed in meters, are related to the corresponding camera image points by a planar perspective homography. Hence, for each image we can estimate a homography that best relates image points to world points.

3. The homography between each image and the world can be interpreted as the composition of rigid body transformation and a projective transformation. The matrices representing this rigid body transform and projection transform are called the *extrinsic matrix* and *intrinsic matrix* respectively. This interpretation is physically motivated – we take the projective pin-hole camera model and apply a rigid body transform to account for the object's position relative to the camera in 3D world space.

4. From the last three steps, we have: a set of ground-truth observations relating world points to image points, a homography for each image that models observations for that image and a decomposition for each of these homographies into intrinsic and extrinsic matrices. Also, each of these intrinsic matrices must be the same because the same camera was used to obtain all object images. We then setup a non-linear least-squares optimization to estimate the rigid body transformations and camera matrix that explain the observed world-image correspondences.
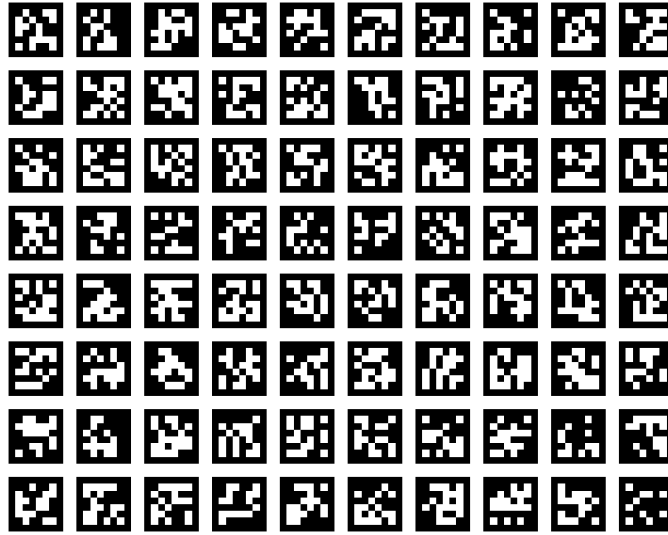
Figure 3.4.1: *The planar calibration target. This calibration target uses a mosaic of visual fiducials, where each fiducial marker occupies an area of one sq.inch. The visual fiducials are described in [23].*

In the following sub-sections we explain each of the above steps with mathematical detail.

3.4.1 Ground-truth observations

Ground-truth observations for camera calibration are obtained by acquiring multiple images of a printed planar target with the camera. The planar target is printed on a flat surface and has easily detected features with known positions and dimensions. In our experiments we use the planar target shown in Fig. 3.4.1.

Calibration target images that are pure translations or pure rotations of each other produce redundant constraints for model estimation. Hence, the image aquisition process should be such that one image is not a pure translation or rotation of another image.

3.4.2 Homography estimation using the direct linear transform

Given a set of correspondences between 3D world points of a planar target and points on its 2D image, a homography can be computed using the *Direct Linear Transform* procedure, as follows. Let $\mathbf{p} = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \end{bmatrix}^\top$ be a 3D homogeneous world point that corresponds to an image point $\mathbf{q} = \begin{bmatrix} u & v & w \end{bmatrix}^\top$; let $\mathbf{p}$ and $\mathbf{q}$ be

related by a homography $\mathbf{H} \in \mathbb{R}^{3 \times 4}$. Plugging these into Eq. 3.2.1 gives

$$\underbrace{\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}}_{\mathbf{q}} = \frac{1}{w} \begin{bmatrix} x_2 \\ y_2 \\ w \end{bmatrix} = \underbrace{\begin{bmatrix} h_{1a} & h_{1b} & h_{1c} & h_{1d} \\ h_{2a} & h_{2b} & h_{2c} & h_{2d} \\ h_{3a} & h_{3b} & h_{3c} & h_{3d} \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix}}_{\mathbf{p}}$$

Without loss of generality, we can assume that the 3D position of the planar target is along the xy-plane, and hence the z-coordinate of all points on the plane are zero. Setting $z_1 = 0$ gives us:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{w} \begin{bmatrix} x_2 \\ y_2 \\ w \end{bmatrix} = \begin{bmatrix} h_{1a} & h_{1b} & h_{1d} \\ h_{2a} & h_{2b} & h_{2d} \\ h_{3a} & h_{3b} & h_{3d} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \tag{3.4.1}$$

We then obtain expressions for u and v by multiplying out the matrix entries:

$$u = \frac{x_2}{w} = \frac{h_{1a}x_1 + h_{1b}y_1 + h_{1d}}{h_{3a}x_1 + h_{3b}y_1 + h_{3d}}$$
$$v = \frac{y_2}{w} = \frac{h_{2a}x_1 + h_{2b}y_1 + h_{2d}}{h_{3a}x_1 + h_{3b}y_1 + h_{3d}}$$

After rearranging terms, we obtain two linear equations:

$$h_{3a}\,ux_1 + h_{3b}\,uy_1 + h_{3d}\,u - h_{1a}\,x_1 - h_{1b}\,y_1 - h_{1d} = 0$$
$$h_{3a}\,vx_1 + h_{3b}\,vy_1 + h_{3d}\,v - h_{2a}\,x_1 - h_{2b}\,y_1 - h_{2d} = 0$$

The above equations can be written as

$$\begin{bmatrix} \mathbf{c}_1^\top \\ \mathbf{c}_2^\top \end{bmatrix} \mathbf{h} = \mathbf{0}$$

where

$$\begin{aligned} \mathbf{h} &= \begin{bmatrix} h_{1a} & h_{1b} & h_{1d} & h_{2a} & h_{2b} & h_{2d} & h_{3a} & h_{3b} & h_{3d} \end{bmatrix}^\top \\ \mathbf{c}_1 &= \begin{bmatrix} -x_1 & -y_1 & -1 & 0 & 0 & 0 & ux_1 & uy_1 & u \end{bmatrix}^\top \\ \mathbf{c}_2 &= \begin{bmatrix} 0 & 0 & 0 & -x_1 & -y_1 & -1 & vx_1 & vy_1 & v \end{bmatrix}^\top \end{aligned}$$

Thus, each correspondence between a world point $\mathbf{p}$ and an image point $\mathbf{q}$ gives

13

us two constraints on the elements of $\mathbf{H}$. With multiple correspondences, we can collect the constrain coefficients into a matrix $\mathbf{A}$ and write

$$\mathbf{A}\mathbf{h} = \mathbf{0} \tag{3.4.2}$$

and then solve for $\mathbf{h}$. The solution for $\mathbf{h}$ is the right-most singular vector of $\mathbf{A}$, which is found by computing the singular-value decomposition of $\mathbf{A}$.

3.4.3 Homography Decomposition

The homography transformation of a homogeneous point can be decomposed as the action of a rigid body transform, $\mathbf{E} = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{t} \end{bmatrix} \in \mathbb{R}^{3\times4}$, followed by a projection or camera matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$. The camera matrix $\mathbf{K}$ has the general form:

$$\mathbf{K} = \begin{bmatrix} f_x & \kappa & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

where f_x, f_y are the focal-lengths in the x and y directions respectively, c_x, c_y are the coordinates of the principal point and κ is the aspect ratio or skew-factor of the camera. For most modern cameras the skew-factor $\kappa = 0$, and the values of f_x, f_y are very close, though they might not be exactly equal. Since we assume that the source points $\mathbf{p}$ lie on the plane $z = 0$, we get $\mathbf{E} = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix}$. Then the relation between the source point $\mathbf{p}$ and target point $\mathbf{q}$ is given by

$$\mathbf{q} = \lambda \mathbf{K} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} \mathbf{p} \tag{3.4.3}$$

By comparing Eq. 3.4.3 and Eq. 3.4.1, we write

$$\begin{bmatrix} \mathbf{h}_a & \mathbf{h}_b & \mathbf{h}_d \end{bmatrix} = \lambda \mathbf{K} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix}$$

Now, using the knowledge that $\mathbf{r}_1$ and $\mathbf{r}_2$ are orthonormal we get

$$\mathbf{h}_a^\top \mathbf{\Omega} \, \mathbf{h}_b = 0 \tag{3.4.4}$$

$$\mathbf{h}_a^\top \mathbf{\Omega} \, \mathbf{h}_a = \mathbf{h}_b^\top \mathbf{\Omega} \, \mathbf{h}_b \tag{3.4.5}$$

where $\mathbf{\Omega} = \mathbf{K}^{-\top}\mathbf{K}^{-1}$. Using the above constraints, and the fact that we have multiple homographies in our ground-truth observations, we find a least-squares es-

timate of the elements of Ω, and obtain $\mathbf{K}$ through the Cholesky decomposition of Ω^{-1}. Zhang provides alternative analytical formulas for recovering $\mathbf{K}$ from Ω in [32]. Once we have $\mathbf{K}$, we compute $\mathbf{E}$ as follows:

$$\mathbf{E} = \left[\begin{array}{ccc} \mathbf{r}'_1 & \mathbf{r}'_2 & \mathbf{t}' \end{array} \right] = \mathbf{K}^{-1} \left[\begin{array}{ccc} \mathbf{h}_a & \mathbf{h}_b & \mathbf{h}_d \end{array} \right]$$

The resulting $\mathbf{r}'_1$, $\mathbf{r}'_2$ are not unit-norm; we force this by scaling the matrix $\mathbf{E}$ by $\lambda = \left(\sqrt{\|\mathbf{r}'_1\| \, \|\mathbf{r}'_2\|} \right)^{-1}$. Finally, $\mathbf{r}'_3$ is simply the cross-product: $\mathbf{r}'_1 \times \mathbf{r}'_2$. Because of noise in the observations, the matrix $\mathbf{R}' = \left[\begin{array}{ccc} \mathbf{r}'_1 & \mathbf{r}'_2 & \mathbf{r}'_3 \end{array} \right]$ does not in general conform to the properties of a rotation matrix. Hence, we find the closest rotation matrix to $\mathbf{R}'$ through a polar decomposition as follows. Let $\mathbf{USV}^\top$ be the singular-value decomposition of $\mathbf{R}'$. Then the closest rotation matrix to $\mathbf{R}'$ is

$$\mathbf{R} = \mathbf{UV}^\top$$

3.4.4 Non-linear refinement

The estimates of $\mathbf{E}$ and $\mathbf{K}$ obtained previously were obtained by minimizing constraint errors that are not directly related to re-projection error. In practice we require estimates that minimize re-projection error. Hence, we directly optimize re-projection error via non-linear optimization using our initial values of $\mathbf{E}$ and $\mathbf{K}$ as starting points for the optimization.

Furthermore, the use of non-linear optimization allows us to incorporate a model for lens distortion. For example, the dominant radial component of distortion is often modeled as a symmetric polynomial function of pixel radius

$$d\left(\left[\begin{array}{c} u \\ v \end{array} \right] \right) = \left[\begin{array}{c} r'(u, v) \, \cos\left(\theta(u, v) \right) \\ r'(u, v) \, \sin\left(\theta(u, v) \right) \end{array} \right] \tag{3.4.6}$$

where

$$r'(u, v) = r(u, v) + \Delta_r(u, v)$$

$$\theta(u, v) = \arctan\left(v - c_y, u - c_x\right)$$

$$r(u, v) = \left\|\begin{bmatrix} u \\ v \end{bmatrix} - \begin{bmatrix} c_x \\ c_y \end{bmatrix}\right\|$$

$$\Delta_r(u, v) = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix}^\top \begin{bmatrix} r^2(u, v) \\ r^4(u, v) \\ \vdots \\ r^{2n}(u, v) \end{bmatrix}$$

The function $d(\cdot)$ distorts an image point $\begin{bmatrix} u & v \end{bmatrix}^\top$ by $\Delta_r(u, v)$ along the radius. Thus, the function $\Delta_r(u, v)$ is the underlying symmetric polynomial distortion function that acts along the radius of the image. The projection of the source point $\mathbf{p}$ onto a target point $\mathbf{q}$ is now given by the equation

$$\mathbf{q} = \lambda\, d\left(\mathbf{K}\, \mathbf{E}\, \mathbf{p}\right) \tag{3.4.7}$$

It is useful to compare Eq. 3.4.3 and Eq. 3.4.7 in order to understand how distortion is incorporated into the projective pin-hole camera model. Optimal estimates of $\mathbf{K}$, $\mathbf{E}$ and $\mathbf{m} = \begin{bmatrix} m_1 & m_2 & \cdots & m_n \end{bmatrix}^\top$ are obtained by non-linear least-squares optimization of Eq. 3.4.7. The method of choice for solving the non-linear optimization problem is the *Levenberg-Marquadt* algorithm [21, 22].

3.4.4.1   Adding a tangential distortion model

When the planes of individual lens components are misaligned with respect to each other (decentering), a radial distortion model cannot capture the distortion completely. In such cases, the recommended solution is to augment the radial distortion model with a tangential distortion model. The mathematical form of the tangential distortion model is

$$\tau\left(\begin{bmatrix} u \\ v \end{bmatrix}\right) = \rho\left(1 + p_3 r^2(u, v) + p_4 r^4(u, v) + \cdots + p_n r^{2(n-2)}(u, v)\right)$$

$$\text{where } \rho = \begin{bmatrix} 2uv & r^2(u, v) + 2u^2 \\ r^2(u, v) + 2v^2 & 2uv \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$$

The distortion in a system with both radial and tangential distortion is then

$$d\left(\begin{bmatrix} u \\ v \end{bmatrix}\right) = \begin{bmatrix} r'(u,v) \cos(\theta(u,v)) \\ r'(u,v) \sin(\theta(u,v)) \end{bmatrix} + \tau\left(\begin{bmatrix} u \\ v \end{bmatrix}\right)$$

with the functions d, r' and θ as defined in Sec. 3.4.4. When using a tangential distortion model, the non-linear refinement procedure of Sec. 3.4.4 is modified to jointly optimize $\mathbf{p} = [\begin{array}{cccc} p_1 & p_2 & \cdots & p_n \end{array}]^\top$ along with $\mathbf{K}$, $\mathbf{E}$ and $\mathbf{m}$.

3.5 Statistical Models

A statistical model describes the relationships among a set of random variables. Unlike a deterministic mathematical model, a statistical model encodes stochastic relationships between variables. An example of a stochastic model that is often encountered in practice is the standard linear regression model:

$$y = w_1 x + w_0 + \epsilon, \tag{3.5.1}$$

where the random variable $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is the additive error term. This stochastic model describes a relationship between the independent variable x and the dependent variable y. The random variable ϵ makes the relationship between y and x stochastic. In the absence of explicit information about the noise distribution, it is common to invoke the central limit theorem and assume that errors are normally distributed.

3.5.1 Parametric Statistical Models

The statistical model described by Eq. 3.5.1 is a parametric statistical model with parameters w_1, w_0 and noise variance σ^2. When modeling a set of observations with this model, the modeling task reduces to the problem of determining optimal values for these parameters.

This parametric modeling scheme is better explained with the use of an example. Consider the set of observations for Eq. 3.5.1 shown in Fig. 3.5.1.

It can be shown that, under a Gaussian additive noise assumption, the most likely settings of $\theta = \{w_1, w_0, \sigma\}$ are those that maximize the log-likelihood of observations

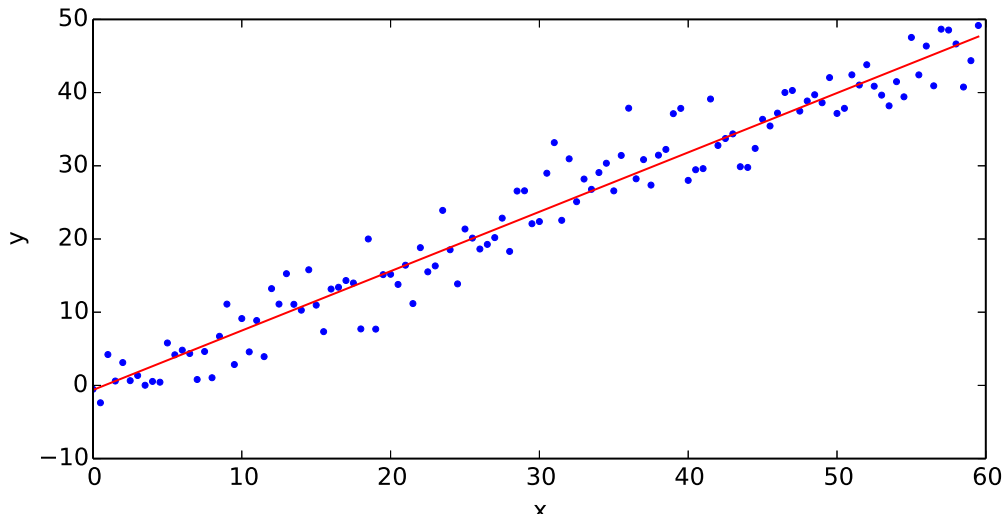$$\mathcal{L}(y|\theta) = \sum_i (y_i - (w_1 x_i + w_0))^2 \tag{3.5.2}$$

Figure 3.5.1: *Linear regression on noisy data. The y-coordinate of each point is linearly related to the x-coordinate, with the actual observation of the y-coordinate corrupted by additive Gaussian noise.*

The optimal values of the parameter set θ are estimated by gradient-descent on the negative log-likelihood function $-\mathcal{L}(y|\theta)$[1].

Sometimes we might conclude (based on intuition or prior knowledge), that a quadratic relationship is a better model of the observed data. We must then expand our parametric model to account for the additional model complexity: we now set

$$\theta = \{\, w_2,\, w_1,\, w_{0,}\, \sigma\,\}$$
$$y = w_2 x^2 + w_1 x + w_0 + \epsilon \tag{3.5.3}$$
$$\mathcal{L}(y|\theta) = \sum_i \left(y_i - (w_2 x^2 + w_1 x_i + w_0) \right)^2 \tag{3.5.4}$$

Once again, the optimal parameter set θ is obtained by gradient-descent on the negative log-likelihood $-\mathcal{L}(y|\theta)$.

3.5.2 Model misspecification

In the general case (and especially for high-dimensional data), it is hard to make valid prior assumptions about the complexity of the statistical model. This causes the risk of model misspecification: using a model that is too simple or too complex

---

[1]For the particular case of least-squares linear regression, the optimal values of the parameter set θ can be determined in closed-form (see [4]). However, we choose to mention gradient-descent here, as it is applicable to a wider variety of models.

to model the observed data. A model that is too simple might not fit the observations correctly; a model that is too complex might over-fit to noise in the observations. It is also possible that the functional form of the model itself is misspecified, resulting in inconsistent and biased predictions.

The parametric modeling paradigm offers a variety of tools to address this problem of determining an appropriate complexity for the model. One can assume a very complex model and avoid over-fitting by regularizing the model. Alternatively, one can perform model selection among a set of possible models using *cross-validation,* or by using information criterion such as the *Akaike Information Criterion* (AIC) or *Bayesian Information Criterion* (BIC).

3.5.3 Non-parametric models

In contrast to parametric statistical models, non-parametric statistical models automatically adjust the number of model parameters to the complexity of the observed data. Because of their name, it is tempting to think of non-parametric methods as those that involve no parameters; on the contrary, non-parametric models have parameters but choose the number of modeling parameters automatically. Thus, non-parametric models perform automatic model selection.

It is also important to note that parametric and non-parametric modeling are not diametrically opposite approaches – both methods involve some modeling assumptions and some flavor of model selection. Parametric approaches assume a functional form for the data being modeled, while non-parametric approaches make more global assumptions, like model *smoothness*. As we will show in this work, the kind of the assumptions made by non-parametric modeling techniques are better suited for modeling distortion in imaging systems.

In the following sub-sections we review locally-weighted regression and Gaussian process regression, two non-parametric approaches that we use in our work.
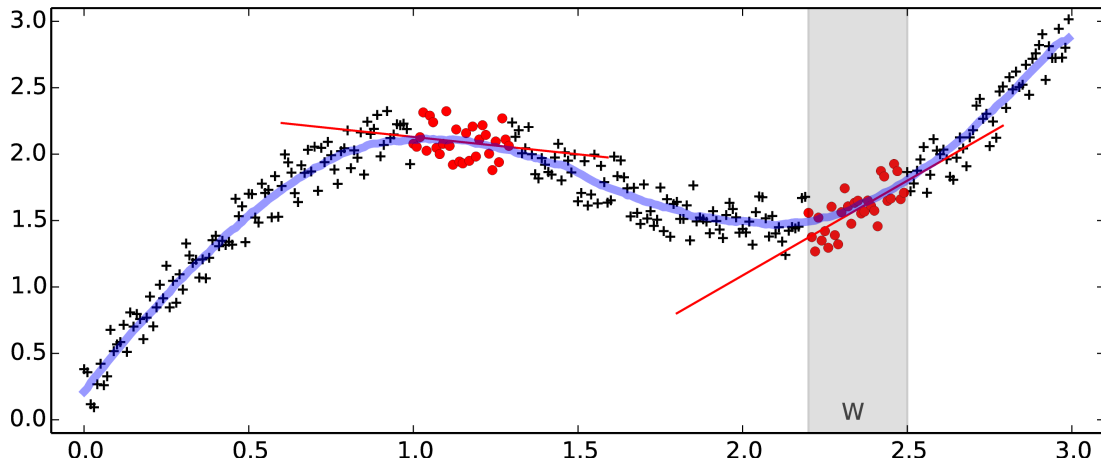
3.5.3.1 Locally-weighted regression

Figure 3.5.2: *Local regression on a non-linear data-set*

Consider the plot in Fig. 3.5.2. The data in this figure is an obvious candidate for non-linear regression. However, in a sufficiently small window, shaded gray, a linear model provides a good fit as shown by the red line.

By systematically sliding this small window *W* over the entire data set, we can compute local linear estimates at each window location. For example, we get another locally-valid linear model for a different window position at the other set of red points. Combining the prediction of these local linear models gives us the non-linear regressor shown in blue.

To enforce smoothness in the resulting non-linear regressor, it is typical to weight the regression problem such that points near the prediction location are weighted higher than those further away. This technique is called *locally-weighted linear regression*[2] and was first described by Cleveland in [11]. Locally-weighted regression is a non-parametric technique that automatically adapts to the amount of non-linearity present in the data being modeled. It has *hyper-parameters* that correspond to the parameters of the weighting function, but the total number of hyper-parameters remain the same for any amount of non-linearity in the data. For effective regression, optimal values for the hyper-parameters are found by minimizing error on a held-out validation subset.

---

[2]This process is similar to the computation of a convolution or a moving average.

3.5.3.2 Gaussian process regression

The use of Gaussian processes (GP) for regression is an extensive topic; we present a practical definition of the technique here in the context of non-parametric regression, and refer the reader to [25] for a thorough mathematical treatment of the subject.

To predict using a GP on a set of observations $\mathbf{t} \in \mathbb{R}^{N \times 1}$, observed at a set of input locations $\mathbf{x} \in \mathbb{R}^{N \times 1}$, we assume that the data was obtained from a GP prior with covariance function $k(x_n, x_m)$. Let $\mathbf{C}_N \in \mathbb{R}^{N \times N}$ be the covariance matrix with elements $\mathbf{C}_{nm} = k(x_n, x_m)$. When we require a prediction for a new input x_{N+1}, we first construct the covariance matrix $\mathbf{C}_{N+1}$, which has the form

$$\mathbf{C}_{N+1} = \begin{bmatrix} \mathbf{C}_N & \mathbf{k} \\ \mathbf{k}^\top & c \end{bmatrix}$$

in other words

$$c = k(x_{N+1}, x_{N+1})$$
$$\mathbf{k}^\top = \begin{bmatrix} k(x_1, x_{N+1}) & k(x_2, x_{N+1}) & \cdots & k(x_N, x_{N+1}) \end{bmatrix}$$

Then the mean and variance of a prediction at the new location x_{N+1} are given by

$$m(x_{N+1}) \; = \; \mathbf{k}^\top \mathbf{C}_N^{-1} \mathbf{t} \tag{3.5.5}$$
$$\sigma^2(x_{N+1}) \; = \; c - \mathbf{k}^\top \mathbf{C}_N^{-1} \mathbf{k} \tag{3.5.6}$$

When the matrix product in equation (3.5.5) is expanded, it has the form

$$m(x_{N+1}) = \sum_{i=0}^{N} w_i t$$

This shows us that GP regression (non-parametrically) predicts new outputs as a weighted sum of training target values. However, the weighting scheme is not simple, as in the case of locally-weighted regression, and may not have a closed form expression. The relationship between a kernel function and its equivalent weighting function is explored by Sollich in [29].

A common choice for the covariance function is the squared exponential (SE)

kernel

$$k(x_m, x_n) = \theta_1^2 \exp\left\{-0.5\frac{(x_m - x_n)^2}{\theta_0^2}\right\} + \beta^{-1}\delta_{ij} \qquad (3.5.7)$$

Here β is the input noise precision and δ is the delta function

$$\delta_{ij} = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$$

This covariance function captures the correlation between GP outputs at different input locations. It produces regression curves that are very smooth. Informally, one can think of the SE covariance function parameters, which are the *hyper-parameters* of the GP, as controlling the smoothness of the resulting regression curve. It is reasonable to assume that lens distortions are smooth differentiable functions and hence it is appropriate to model such distortions using the SE covariance function.

For effective regression, one has to determine appropriate settings for the hyper-parameters β, θ_0 and θ_1. Optimal values of the hyper-parameters are usually found by optimizing the marginal likelihood of observed data given the hyper-parameters. In this work however, we also use cross-validation on data-folds to find good settings for the hyper-parameters. Rasmussen and Williams provide a detailed description of hyper-parameter selection for GPs in [25].

Chapter 4

Gaussian process radial distortion models

The traditional approach for calibrating a camera with noticeable lens distortion is the non-linear least squares procedure described in Sec. 3.4. However, there is no simple way to decide the degree of polynomial required to model distortion in the camera model. A principled approach is to try multiple settings for the degree of polynomial and choose a setting that minimizes validation error: a set of observations is set aside as a validation set, and optimized models with different polynomial degree settings are used to predict distortion on the held-out validation set. The model that produces the least amount of error on the validation set is chosen as the most appropriate.

In this chapter we present a Gaussian process (GP) model of radial distortion. Instead of training multiple parametric models and then choosing the most appropriate model, we directly model lens distortion using a non-parametric GP model. To achieve this, we first describe how the camera calibration procedure can be viewed as performing maximum-likelihood inference on a factor graph. We then show how a non-parametric GP radial distortion model can be made a part of this inference.

4.1 Camera calibration as maximum likelihood factor graph inference

Consider a world point $\mathbf{p}$ projected onto an image point $\mathbf{q}$ by a camera with intrinsic parameters $\mathbf{K}$, extrinsic parameters $\mathbf{E}$ and lens distortion function $d(\cdot)$ with
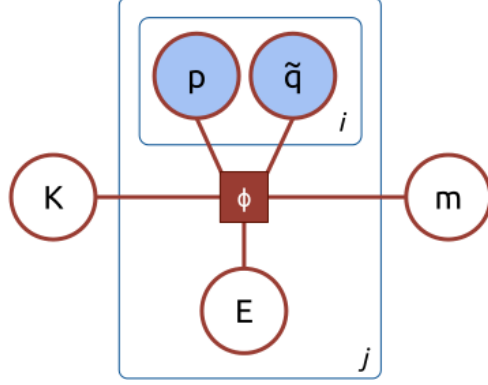
Figure 4.1.1: *Factor graph representation of camera calibration. This figure shows a factor graph representation of the constraint induced by an observation between a world point $\mathbf{p}$ and image point $\tilde{\mathbf{q}}$.*

parameters $\mathbf{m}$. The model for this projection is given by Eq. 3.4.7, repeated below for convenience

$$\mathbf{q} = \lambda\, d\, (\mathbf{K}\, \mathbf{E}\, \mathbf{p})$$

Now, if we observe a correspondence between a world point $\mathbf{p}$ and an image point $\tilde{\mathbf{q}}$, we would like to set $\mathbf{K}$, $\mathbf{E}$ and $\mathbf{m}$ to minimize $(\mathbf{q} - \tilde{\mathbf{q}})^2$. In other words, we want our prediction $\mathbf{q}$ to be as close to the observed point $\tilde{\mathbf{q}}$ as possible. Thus, the possible settings for $\mathbf{K}$, $\mathbf{E}$, $\mathbf{m}$, $\mathbf{p}$ and $\mathbf{q}$ are constrained by the factor $\phi = (\mathbf{q} - \tilde{\mathbf{q}})^2$. We express this relationship as a factor graph, as show in Fig. 4.1.1. The nodes representing the observed variables $\mathbf{p}$ and $\tilde{\mathbf{q}}$ are shaded, implying that their values are fixed to their observed values.

Since we obtain ground-truth observations at different poses $\mathbf{E}_j$, we have multiple corresponding $\mathbf{p}_{ij}$ and $\tilde{\mathbf{q}}_{ij}$ at each $\mathbf{E}_j$ as described in Sec. 3.4.1. This is represented using the plate notation shown in Fig. 4.1.1, where there are N_j pose plates and each pose plate contains N_i nested observation plates. The values of $\mathbf{K}$ and $\mathbf{m}$ are common for all observations and hence appear outside of all plates. A factor ϕ_j, which is the ϕ corresponding to pose $\mathbf{E}_j$, connects $\mathbf{K}$ and $\mathbf{m}$ with the corresponding $\mathbf{E}_j$, $\mathbf{p}_{ij}$ and $\tilde{\mathbf{q}}_{ij}$ for all i.

The most likely values of $\mathbf{K}$, $\mathbf{m}$ and $\mathbf{E}_j$ are computed using iterated least-squares as follows. Each factor constraint ϕ_j is linearized to obtain a linear constraint at the current values of $\mathbf{K}$, $\mathbf{E}_j$, $\mathbf{m}$, $\mathbf{p}_{ij}$ and $\mathbf{q}_{ij}$. The least-squares solution to this set of linear constraints gives us the new values for $\mathbf{K}$, $\mathbf{m}$ and $\mathbf{E}_j$. This process is iterated until convergence. As mentioned in Sec. 3.4.4, it is preferable to use a damped Levenberg-Marquadt update instead of the standard least-squares update.

This description of camera calibration is just another view of the optimization process. It not fundamentally different from the description of the calibration procedure in Sec. 3.4.4.

4.2   Gaussian process factors

It is not immediately obvious as to how one can incorporate an infinite dimensional object like a GP into a factor graph. To achieve this, we make use of the fact that a finite set of outputs of the GP, at input locations $\mathbf{g}$, have the following exploitable structure. By definition, for a zero mean GP with covariance function $k(x_m, x_n)$, the values $f(\mathbf{g})$ at a finite set of input locations $\mathbf{g}$ have a multi-variate Gaussian distribution

$$f(\mathbf{g}) \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$$

where the covariance matrix entries are $\mathbf{C}_{ij} \triangleq k(x_i, x_j)$.

Thus, to create a Gaussian process factor, we assume a finite set of input locations $\mathbf{g} \in \mathbb{R}^{N \times 1}$ and constrain the function values $f(\mathbf{g}) \in \mathbb{R}^{N \times 1}$ to conform to the mean and covariance structure of a GP. The factor potential for a particular setting of $f(\mathbf{g})$ is given by

$$\phi = f(\mathbf{g})^\top \mathbf{C}^{-1} f(\mathbf{g})$$

This factor potential constrains the values $f(\mathbf{g})$ to vary smoothly according to the parameters of the kernel function k (see Sec. 3.5.3.2), and the resulting Gaussian process factor node captures a non-parametric representation of a smooth function. To evaluate this function at a new input location, we make use of Eq. 3.5.5, by setting $\mathbf{t} = f(\mathbf{g})$.

In comparison with the camera calibration optimization described in the Sec. 3.4.4 and in the previous section, the only change here is the use of a non-parametric radial distortion function. The distortion function is evaluated similar to Eq. 3.4.6, but uses an alternate definition of r':

$$r'(u, v) = \mathbf{k}^\top \mathbf{C}^{-1} f(\mathbf{g})$$

where $\mathbf{k}^\top = \left[\begin{array}{cccc} k(g_1, g_{N+1}) & k(g_2, g_{N+1}) & \cdots & k(g_N, g_{N+1}) \end{array} \right]$, $g_{N+1} = r(u, v)$. The above is basically an evaluation of the GP mean prediction at $g_{N+1} = r(u, v)$, given the values $f(\mathbf{g})$ of the GP at input locations $\mathbf{g}$.

In other words, we try to infer the distortion function values at a set of pre-
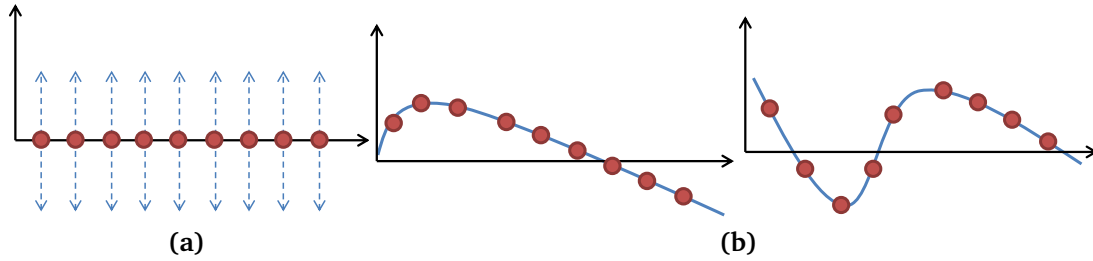
(a)   **(b)**

Figure 4.2.1: *Control point analogy for Gaussian process models. (a) The set of function evaluations $f(\mathbf{g})$ at input locations $\mathbf{g}$ in a GP factor node act as control point that induce a function based on their position. (b) Inference on the GP node can then be visualized as positioning these points so that they induce the required function. This is analogous to the use of control points for manipulating Bézier and spline curves in graphic modeling software.*

specified pixel radii. Intuitively, these distortion function values *induce* a mean function that is used to predict distortion values for new input pixel radii. One way of visualizing this technique is to see these distortion function values as *control points* that control a smooth curve, as explained in Fig. 4.2.1. Thus, with the GP distortion model, the factor graph inference procedure infers a function indirectly by inferring the position of these control points.

4.2.1 Hyper-parameter optimization

In a fully Bayesian setting, a rigorous way of optimizing hyper-parameters is to optimize the marginal likelihood of the data given the hyper-parameters. However in our case, the GP is embedded inside a factor graph and calculating the marginal likelihood of the observed data given the hyper-parameters would involve integrating out the matrices $\mathbf{K}$ and $\mathbf{E}$.

Instead, we set the hyper-parameters using a method based on cross-validation. First, we set aside a subset of the observations as a validation set. Then, we perform a grid-search over the domain of hyper-parameter values and infer a camera model at each of these hyper-parameter settings. Finally, we choose the hyper-parameter setting that produced the least error on the validation set.

4.3 Experiments

When integrating a GP model into the factor graph, one has to decide the number of input locations used in the GP node. A large number of samples makes the opti-

mization process computationally expensive. However, since the distortion function being modeled is smooth, a relatively small number of samples should suffice. In our implementation, we perform inference on 25 equally spaced function values.

For the calibration procedure, we used a training set of 14 images. An average of 40 world-image correspondences were extracted from each image. In order to find hyper-parameter settings, we chose to further divide the training set into a development set consisting of 11 images and a validation set of 3 images. By random shuffling, 30 different development/validation partitions were created from the same training set.

As mentioned in Sec. 4.2.1, We perform a grid search over the space of hyper-parameter settings and evaluate the validation error of each hyper-parameter setting on the 30 different training set partitions. We then choose the hyper-parameter settings that gave the lowest validation error as the best estimate of the hyper-parameters.

We evaluate prediction error on a testing/validation image as follows:

1. We first undistort the test image using the lens distortion model that was estimated. World-image point correspondences are then obtained from this undistorted image.

2. Then, an initial estimate of the calibration target extrinsics is obtained using the DLT method. This estimate is further refined by non-linear optimization of the reprojection error, with the camera model and lens distortion held constant.

3. The reprojection error after convergence of the non-linear optimization is reported as the prediction error.

We evaluated the performance of our method on two wide field-of-view camera lenses: a Tamron lens with a focal length of 2.2 mm and another Tamron lens with a focal length of 2.8 mm; the 2.2 mm lens produces more distortion than the 2.8 mm lens. We report test errors as a rigorous assessment of model predictive performance.

For both lenses, we use an independent calibration sequence of 10 images as the test set. We present the results of this evaluation in Fig. 4.3.1. From our plots, we conclude that the GP distortion model is capable of capturing a suitable distortion function. Also, the performance of the GP distortion model is comparable to the performance of the best polynomial lens distortion models.
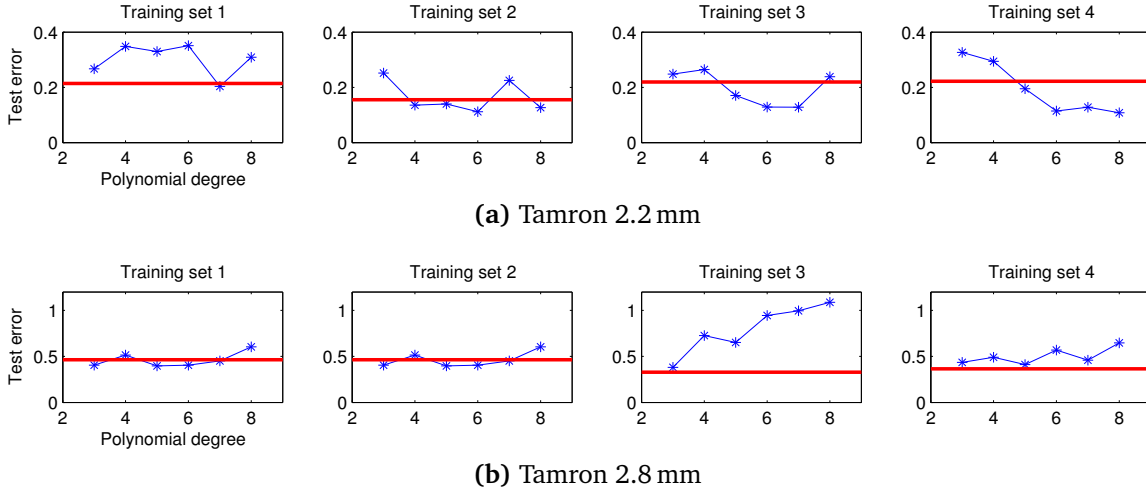
(a) Tamron 2.2 mm

(b) Tamron 2.8 mm

Figure 4.3.1: *Performance comparison of distortion models for two Tamron lenses. Here, we plot the test pixel reprojection errors obtained when using the GP model (red bar) and polynomial models of orders 3,4,··· 8 (blue segments). We observe that the GP model performs on par with the best polynomial distortion models.*

We also study the variation in test-error with training set size and present results in Fig. 4.3.2. We use the results from this study to decide that the optimal size of training set is 11 images.

The lens distortion model estimated for the lenses and images rectified using the distortion model are shown in Fig. 4.3.3. The estimated models suggest a barrel distortion for both lenses. Visual inspection of the acquired camera images confirms the presence of predominant barrel distortion and thus confirms the validity of the estimated model.

4.4 Summary

In this chapter, we proposed the use of a GP model for lens distortion. This model of lens distortion is an alternative to performing explicit model selection amongst multiple parametric models.

To model lens distortion non-parametrically using a GP, we first expressed the camera calibration procedure as maximum likelihood inference on a factor graph and then showed how Gaussian process nodes can be embedded into this factor graph.

Finally, we presented results confirming the suitability this GP lens distortion model for modeling the distortion of real world lenses. Our results confirmed that the GP distortion model performed on par with polynomial lens distortion models.
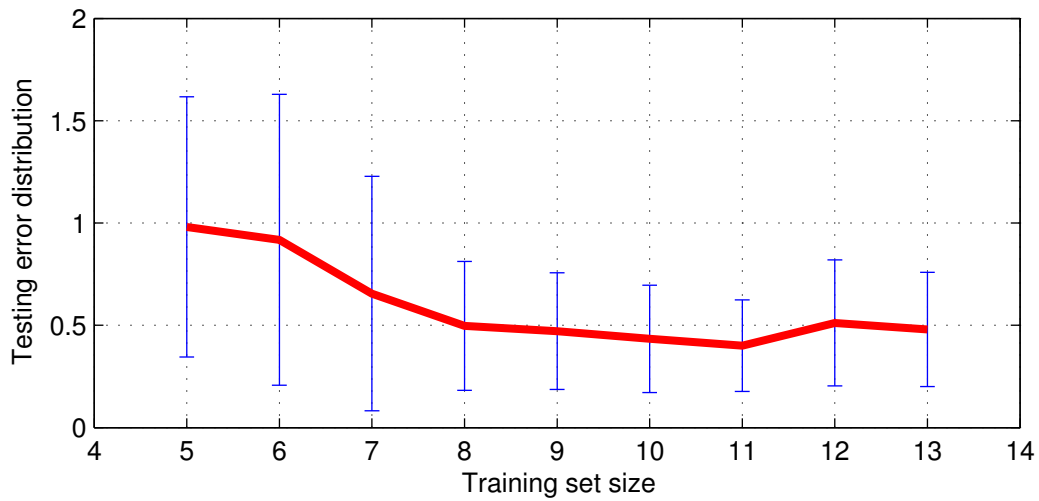
Figure 4.3.2: *Variation in test error as a function of training set size. This plot shows that the that variation in test error reduces with increasing training set size. However, the test error stabilizes around a training set size of 12. Hence, for our experiments, we choose a training (development when performing validation) set size of 11 images.*

(a) Estimated GP distortion functions
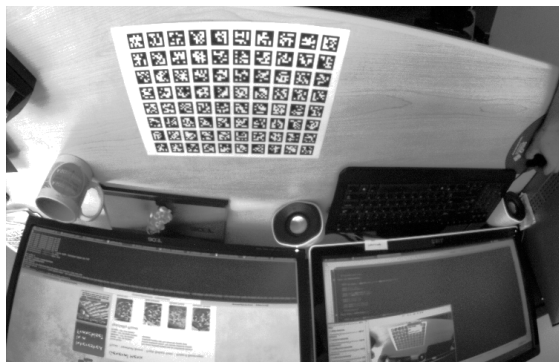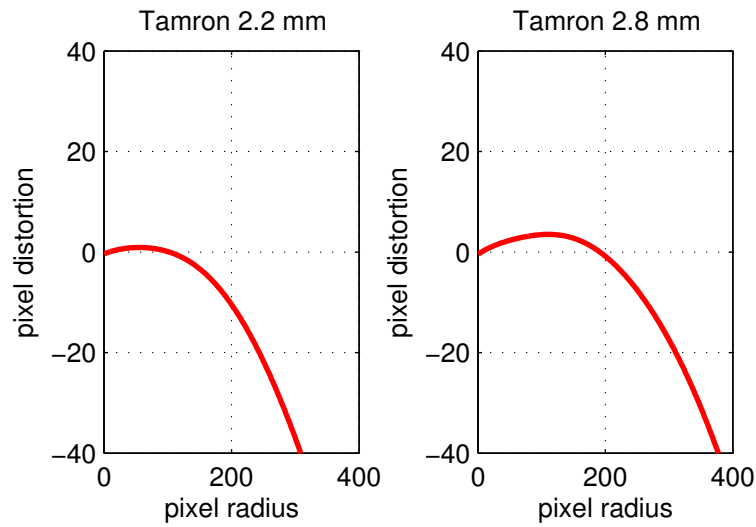
(b) Tamron 2.2 mm distorted

(c) Tamron 2.2 mm rectified
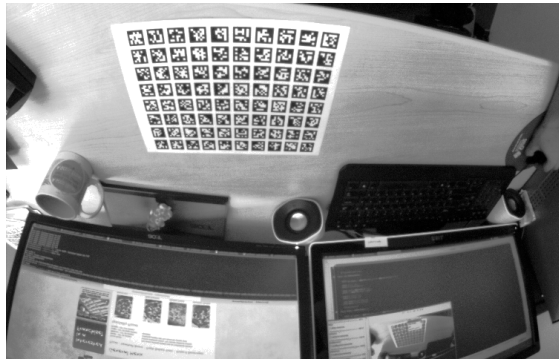
(d) Tamron 2.8 mm distorted

(e) Tamron 2.8 mm rectified

Figure 4.3.3: *Images rectified with the GP distortion model. The models shown in (a) were estimated using the factor graph inference method of Sec. 4.1. The estimated distorted models were then used to rectify the acquired images (b) and (d). The lenses have a predominant barrel type distortion. The 2.2 mm focal length lens has a wider field-of-view and hence produces more distortion than the 2.8 mm lens.*

Chapter 5

Locally-weighted Homographies

In the previous chapter, we presented a non-parametric model of lens distortion. This model simplified the process of choosing the right model complexity for describing lens distortion. However, the model is still limited by its physical assumptions: it assumes that the imaging components are optically aligned. Hence, it can only model distortions that are radially symmetric.

Here, we will describe an alternative non-parametric description of lens distortion that provides a pure mathematical description of lens distortion. Since it is not based on an underlying physical model, it is capable of modeling arbitrary sources of lens distortion.

In the following sections, we first introduce the concept of a locally-weighted homography, which is a type of a non-parametric non-linear homography. We then show how lens distortion can be modeled using this locally-weighted homography. We then evaluate this method on real lenses and also show how this method can be used to rectify arbitrary sources of image distortion.

5.1 Non-linear homographies

The idea of locally-weighted linear models, described in Sec. 3.5.3.1, can be extended to the problem of homography estimation by considering the weighted version of the regression problem Eq. 3.4.2. This gives us

$$\left(\mathbf{A}^\top \mathbf{W} \mathbf{A} \right) \mathbf{h} = \mathbf{0}$$

where $\mathbf{W}$ is a diagonal matrix of weights. The weights themselves are computed using the squared exponential weighting function
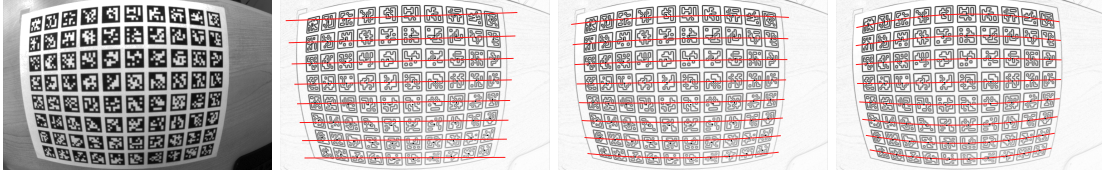
Figure 5.1.1: *Locally-weighted homography estimation. Left-most image is the camera image of the planar target. The following figures show the mapping of lines through row-centers on the target onto the image plane, for weighting bandwidth $\tau = 1, 0.005$ and 0.0005, respectively. Note how the line mappings are progressively more curved. At $\tau = 0.0005$, the mapping is accurate to within 0.1 pixels.*

$$w(\mathbf{q}, \mathbf{x}_i) = \nu^2 \exp\left\{-\frac{\|\mathbf{q} - \mathbf{x}_i\|^2}{2\tau^2}\right\} + \lambda^2 \qquad (5.1.1)$$

where $\mathbf{x}_i$ are the data points on the source plane and $\mathbf{q}$ is the query location on the source plane where a local homography is being estimated. The weighting function is such that the $\mathbf{x}_i$ closer to the query point $\mathbf{q}$ are weighted higher than the $\mathbf{x}_i$ that are far away. The parameter τ controls the spatial extent of the weighting function: larger values of τ correspond to computing local estimates over larger neighborhoods, thereby producing smoother regressors. The parameters ν and λ control the magnitude of the weights and thus regularize the underlying regression problem.

When the weighting parameters ν, τ and λ are set to their optimal values, the locally-weighted homographies computed over the domain of the planar target can map straight lines on the target onto curves in its image as shown in Fig. 5.1.1. This is in contrast to a conventional planar homography that can only map straight lines onto straight lines.

5.2 Distortion observations and interpolation

A locally-weighted homography, by itself, is just a tool for producing a non-linear mapping between source and target planes. In this section we show how a locally-weighted homography can be used to estimate lens distortion in an image.

Let us denote the center of an image by $\mathbf{c}_0$ and consider the local homography at $\mathbf{c}_0$. Because of distortion in the image, this homography is accurate only over a small region near $\mathbf{c}_0$. If there was no distortion, this homography would accurately match world-image correspondences everywhere on the image. This observation leads us to an alternative definition of distortion in an image: distortion is the deviation

from the homography at the center of the image. Given a set of correspondences between world points and image points, $\mathbf{x}_i \rightleftharpoons \mathbf{u}_i$, we estimate lens distortion as follows:

1. Estimate a locally-weighted homography $\mathbf{H}_0$ from world to image coordinates at the center of the image.

2. Project the world points $\mathbf{x}_i$ through the homography $\mathbf{H}_0$ to obtain the points $\mathbf{u}'_i$.

3. The distortion estimate at the image point $\mathbf{u}_i$ is then $\mathbf{d}_i = (\mathbf{u}'_i - \mathbf{u}_i)$.

4. As a result, the undistortion needed to rectify the point $\mathbf{u}_i$ is then $\bar{\mathbf{d}}_i = -\mathbf{d}_i = -(\mathbf{u}'_i - \mathbf{u}_i)$.

This alternative definition of distortion provides a very flexible way of modeling lens distortion without making assumptions about symmetry or parametric form. The above-mentioned procedure provides us with distortion/undistortion observations at the image points $\mathbf{u}_i$. To predict the undistortion at any other point we use interpolation.

To continue with our goal of not making any radial or tangential assumptions, we interpolate undistortion as a direct 2D function of pixel coordinates. A reasonable choice is to use a 2D polynomial mapping to model the undistortion. Instead, we chose to model the undistortion using a 2D GP. This implicitly takes care of regularization and model selection with the additional advantage of making the distortion function non-parametric.

The 2D GP consists of two independent GPs, one $\bar{d}_x$ modeling the undistortion in the x-direction and the other $\bar{d}_y$ modeling undistortion in the y-direction. The undistortion components can be modeled separately because, given the pixel coordinates s and t, $\bar{d}_x(s,t)$ is conditionally independent of $\bar{d}_y(s,t)$. This GP interpolation model uses a 2D squared exponential covariance function

$$k(\mathbf{x}_m, \mathbf{x}_n) = \theta_1^2 \exp\left\{-\frac{1}{2}\mathbf{x}_m^T \mathbf{\Sigma}^{-1} \mathbf{x}_n\right\} + \delta_{mn}\beta^{-1}$$

which has the hyper-parameters θ_1, β and the matrix $\mathbf{\Sigma} \in \mathbb{R}^{2 \times 2}$. The 2D function squared exponential function has six hyper-parameters, while the 1D function, shown in Eq. 3.5.7, has only three. Distortion observations and an interpolating undistortion model for the image in Fig. 5.1.1 are show in Fig. 5.2.1.
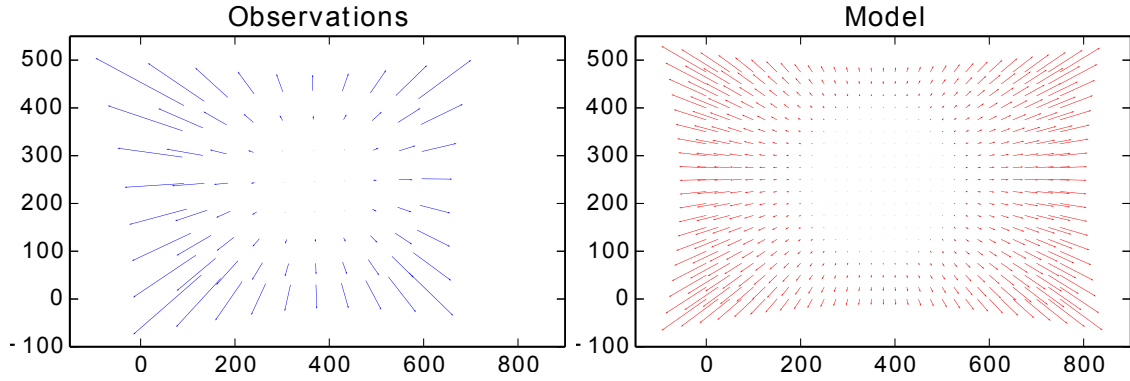
Figure 5.2.1: Distortion observations and interpolated distortion model. The quiver plot on the left shows the distortion observations obtained for the image in Fig. 5.1.1. On the right is a quiver plot of the GP undistortion model inferred from these observations.

The completely non-parametric nature of this lens distortion model allows us to rectify arbitrary sources of image distortion as show in Fig. 5.2.2.

5.2.1 Estimating the local homography at the center

We have previously glossed over details about tuning the parameters of the weighting function used to compute the local homography estimates. We explain this process in detail here. To obtain an optimal local homography at the center $\mathbf{H}_0$, we tune the parameters τ, ν and λ of the weighting function, Eq. 5.1.1, as follows:

1. We choose nine correspondences closest to the center of the image.

2. We construct nine folds of this data by leaving out one correspondence at a time. Each left-out correspondence is the validation correspondence for its respective fold.

3. For a given τ, ν and λ, we construct locally-weighted homographies for each fold and compute prediction error on the validation sample for that fold. The average prediction error across folds gives us the error for the given value of τ, ν and λ.

4. We find the optimal values $\hat{\tau}$, $\hat{\nu}$ and $\hat{\lambda}$ by minimizing the error mentioned in the previous step.

In other words, we restrict our attention to a dataset consisting of the nine correspondences closest to the image center. We find the optimal parameters of the
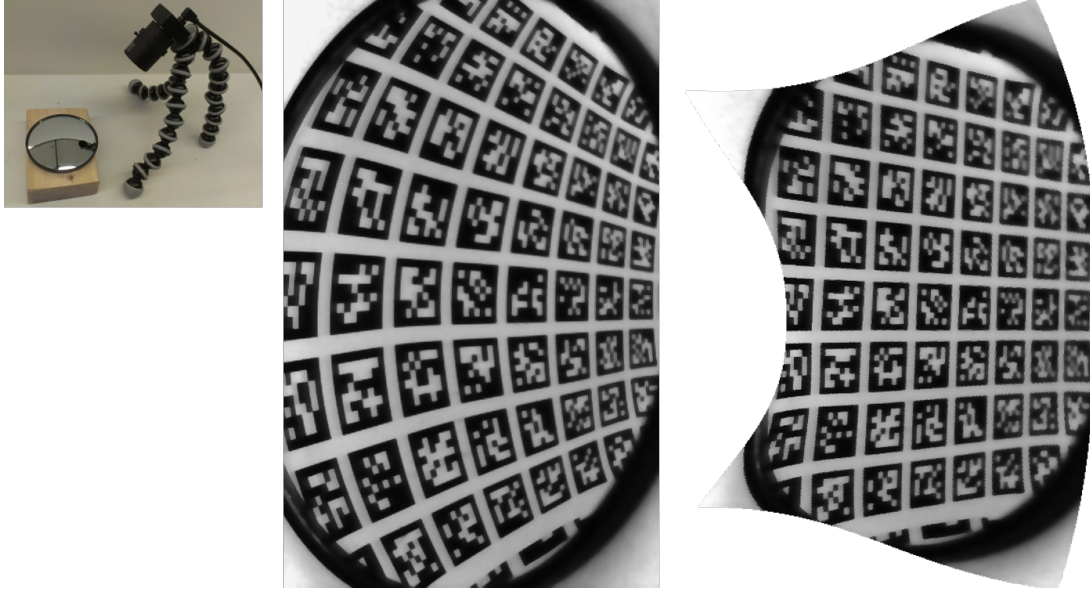
34

Figure 5.2.2: *Rectifying arbitrary sources of distortion. Original distorted image and undistorted image from an ad hoc setup involving a camera and an automotive blind-spot mirror. The proposed distortion model can non-parametrically estimate and correct arbitrary distortion from just a single image of a planar target. Single image calibration is described later in Sec. 5.3.1.*

locally-weighted homography for these nine correspondences by minimizing the *leave-out-one cross-validation* (LOO-CV) error across folds of the dataset. Since we have not analyzed the differentiability of the LOO-CV error objective used here, we use Powell's method, described in [24], to perform direct minimization without computing derivatives.

5.2.2   Accuracy of the distortion model

As an evaluation of the local homography-based undistortion technique, we undistort correspondences from three real-world lenses and report pixel deviation from straightness in Table 5.1. Overall, we observe deviations of less than one pixel. However, for the set of lenses used in our evaluation, there is no significant difference in the performance of the polynomial (up to fifth order) and GP models. Fig. 5.2.3 has some examples of images undistorted using a non-parametric GP model.

Pix Err→		Dataset 1		Dataset 2		Dataset 3		Dataset 4		Dataset 5	
		AVG	MAX	AVG	MAX	AVG	MAX	AVG	MAX	AVG	MAX
Tamron 2.2	POLY	0.09	0.37	0.11	0.54	0.09	0.39	0.05	0.17	0.09	0.41
	GP	0.09	0.36	0.12	0.51	0.08	0.44	0.04	0.16	0.09	0.38
Tamron 2.8	POLY	0.07	0.24	0.11	0.39	0.11	0.56	0.16	0.65	0.13	0.76
	GP	0.07	0.24	0.11	0.39	0.10	0.37	0.15	0.64	0.13	0.78
Tokina 3.3	POLY	0.04	0.21	0.05	0.32	0.11	0.65	0.06	0.29	0.08	0.43
	GP	0.04	0.32	0.05	0.31	0.12	0.67	0.06	0.33	0.08	0.42

Table 5.1: *Evaluation of straightness after undistortion. We report mean absolute deviation from straightness after undistorting test datasets using the estimated lens distortion model. Overall, we observe that the deviations (mean and max) are within 1 pixel. For the lenses used in this evaluation, there is no significant difference between the polynomial and GP undistortion models.*
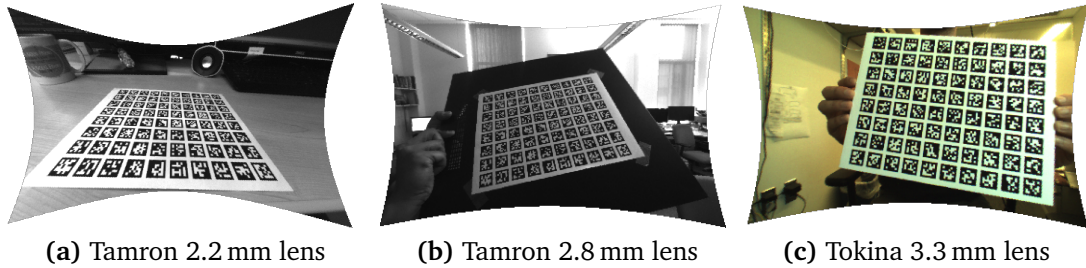
(a) Tamron 2.2 mm lens   **(b)** Tamron 2.8 mm lens   **(c)** Tokina 3.3 mm lens

Figure 5.2.3: *Images undistorted using a GP model estimated from a single image.*

5.3 Extensions and Implementation

In this section, we describe two extensions to the method of estimating lens distortion with locally-weighted homographies. First, we describe how an initial camera calibration along with distortion can be obtained from a single image of a planar target. Next, we show how this initial estimate can be used to bootstrap a classic camera calibration.

5.3.1 Single image calibration

Combining the technique of Sec. 5.2 with camera intrinsics estimation using orthogonal vanishing points, as described by Beardsley in [3] or Cippola in [7], one can obtain useful estimates of both the camera intrinsics and lens distortion simultaneously. This can be done with just one calibration image as follows:

1. Obtain an image of the planar target, such that the target covers most of the image plane while producing two vanishing points. A target that covers the image plane helps in a confident distortion estimate and at least two vanishing points are required for recovering the camera matrix **K**.

2. Estimate the undistortion as described in Sec. 5.2, and undistort the image.

3. Estimate vanishing points from the undistorted image and use the technique of Beardsley [3] to estimate the camera matrix. Note that with only two vanishing points, we must assume that the principal point (c_x, c_y) is at the center of the image.

This estimate of camera intrinsics and distortion from a single image can be further refined by non-linear optimization.

5.3.2 Augmented camera calibration

The lens distortion estimate obtained using the single image calibration method of Sec. 5.3.1 can be used to simplify classic camera calibration. This is done by first undistorting the calibration images and then optimizing just the camera intrinsics and extrinsics, resulting in an optimization that optimizes a smaller set of parameters.

However, we must note that by undistorting calibration images, we are committing to a single point-estimate of the undistortion and ignoring any uncertainty. As an empirical approximation, we can integrate out the uncertainty parameters from the optimization by sampling from the GP lens distortion estimate and then using these samples to expand the calibration image set. We list out the steps in this *augmented camera calibration* algorithm below:

1. Obtain a model of undistortion using the technique listed in section 5.2.

2. Obtain a set of calibration images.

3. For each image in the calibration set, sample the GP undistortion model multiple times, and undistort the image using the obtained samples. This results in an expanded calibration set that empirically accounts for the uncertainty in the distortion estimate.

4. Perform non-linear least-squares optimization of the camera intrinsic and extrinsic parameters on the undistorted images, as in the classic calibration method.

5.4 Experiments

In Fig. 5.4.1, we compare the performance of the classic and augmented calibration methods and present histograms of errors obtained on multiple testing datasets. We report both the RMSE and max pixel errors and find that the classic calibration method has significantly more outliers.

In Fig. 5.4.2, we compare the convergence of classic non-linear least-squares-based camera calibration with augmented camera calibration. The augmented calibration results in an optimization problem that has a smaller number of parameters and hence has more consistent convergence.
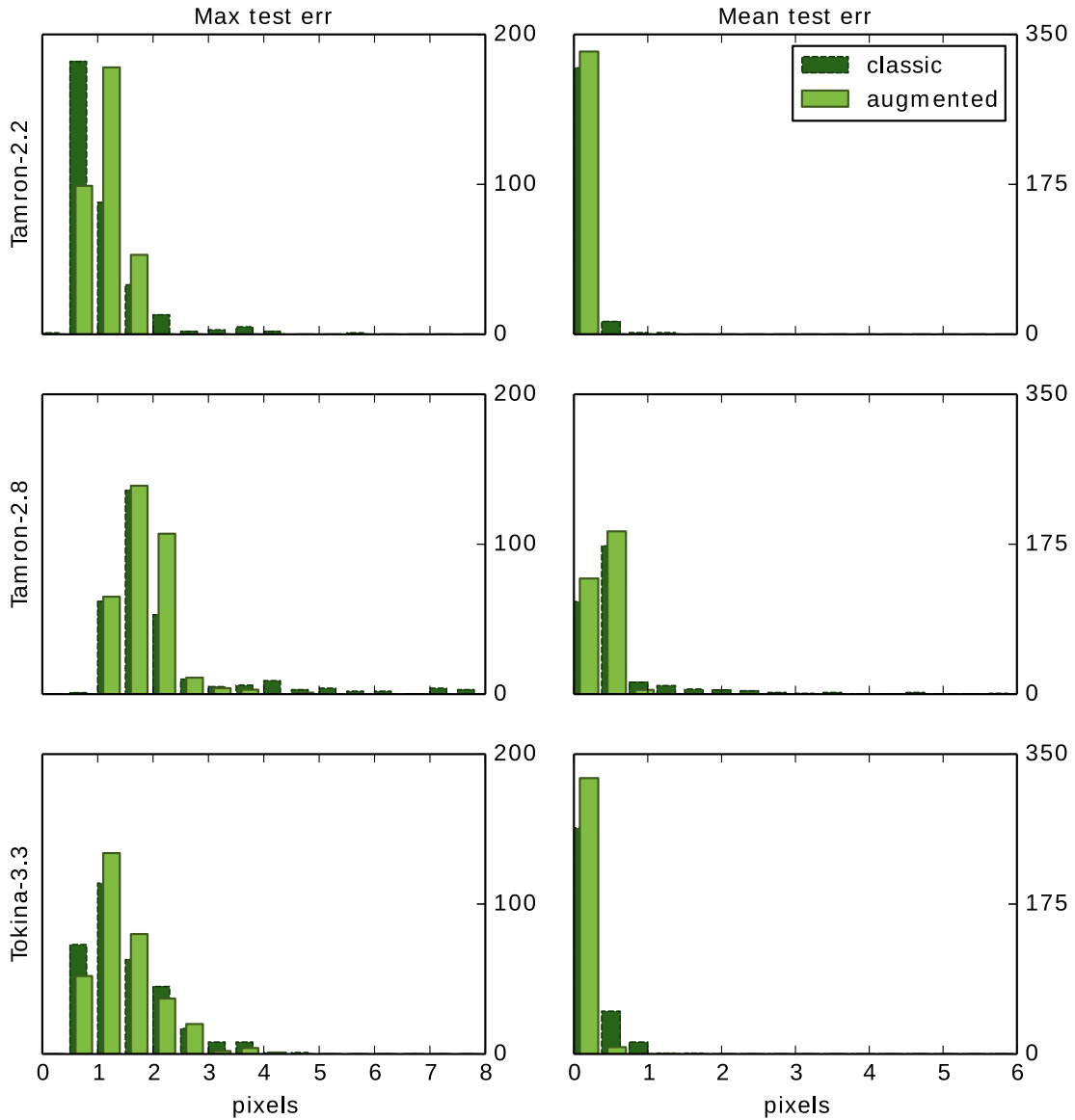
Figure 5.4.1: *Distribution of testing errors for the classic and augmented calibration methods. We observe that both the classic and augmented methods have a very similar distribution of errors except for a significant tail of outliers for the classic method. This tail of outliers suggests that the classic method is prone to over-fitting.*
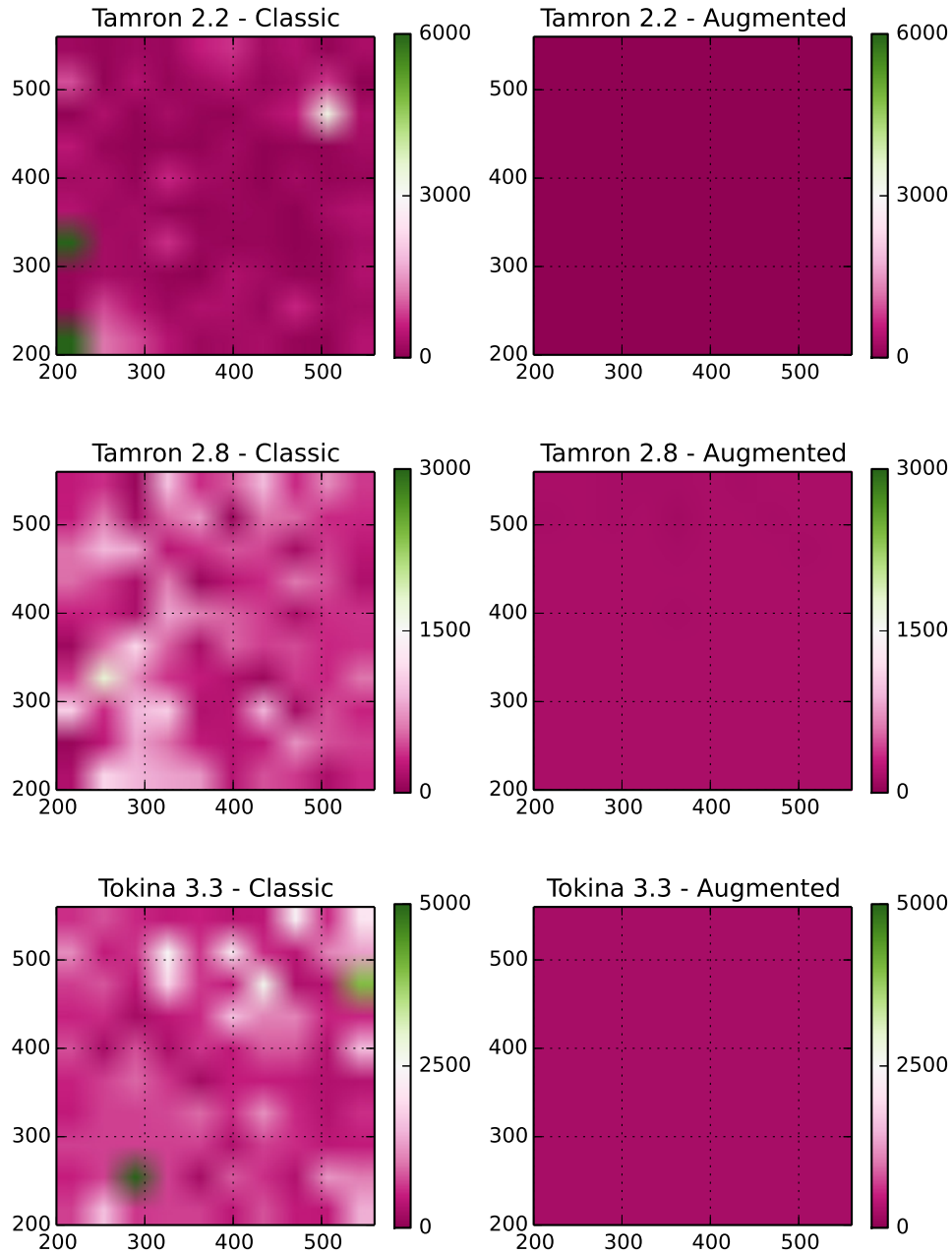
Figure 5.4.2: *Convergence of classic camera calibration vs. augmented camera calibration. Results are reported for three different lenses. The augmented calibration method uses the non-parametric undistortion model estimate to undistort the image before calibrating camera intrinsics and extrinsics. The x-axis and y-axis are the values of the focal lengths f_x and f_y used to initialize the optimization. The principal point (c_x, c_y) was initialized to the pixel image center. The color-mapped value at each point corresponds to the Euclidean distance of the final calibration from a nominal reference calibration. We find that the augmented camera calibration is very flat showing that it is more consistent. For reference, the range of values for the augmented calibration plots are [2.0, 6.9], [3.3, 44.8] and [2.12, 12.9], top to bottom. This improvement in convergence suggests that the augmented method results in a more stable optimization.*

5.5 Summary

In this chapter, we described a novel non-parametric non-linear homography technique. Unlike a planar homography, it is capable of mapping lines from a source plane onto arbitrary smooth curves in the target plane. We then adapted this non-linear homography technique to estimate lens distortion by computing observed deviation from the homography at the image center.

Classic camera calibration involves simultaneous estimation of lens distortion and camera intrinsic parameters. We show that this classic calibration technique can be augmented with our independent lens distortion estimation technique to improve the stability of camera calibration.

Our technique can be visualized as building a continuous mosaic of homographies from the source plane to the target plane. As an extension, we can decompose the homographies on the mosaic and interpret the resulting image as the result of multiple appropriately placed pinhole cameras. This lets us interpret this technique as building a non-parametric, non-linear mapping from 2D image points to rays in 3D.

Chapter 6

Distortion models for zoom-lenses

Zoom lenses are widely used in modern surveillance systems because they allow the user to quickly switch between surveying an entire scene and examining an object in detail. When tracking or inspecting an object, it is faster to zoom-in than to move closer. This convenience afforded by zooming instead of moving makes zoom lenses an important tool in robotic surveillance. Zoom cameras are a standard component in military surveillance drones as well as commercially available aerial vehicles used for photography.

The availability of an accurate calibration for a zoom lens enables good photogrammetric registration of video frames from a zoom camera, since such video is often acquired with continuously changing zoom. Zoom lenses have peculiar distortion characteristics that make classic radial distortion models insufficient. In this chapter, we study the peculiar distortion characteristics of zoom lenses and adapt the non-parametric distortion model of Chap. 5 to model the distortion in zoom lenses.

6.1   The model selection problem for zoom-lenses

A straightforward extension of classic camera calibration to zoom camera calibration is possible by simply calibrating a different model at a discrete set of zoom levels, and then interpolating the model parameters to intermediate zoom levels. The problem with this approach arises from the fact that classic camera calibration assumes a fixed complexity polynomial model, while the complexity of distortion in a zoom lens varies across its zoom levels.

A more pressing issue is the presence of non-radial distortion in zoom lenses.

42

The distortion of most conventional lenses is predominantly radial: dominated by a component that is along the radius of the image and symmetric about the optical center. Hence, it is common to assume a radial model for lens-distortion. However, zoom lenses have multiple optical elements and any misalignment in these elements manifests in distortions that are orthogonal to the radial component. This is a real concern with lenses used in robotic applications because they are often subject to physical shock.

From a calibration perspective, it is useful to model, or at least detect, these non-radial distortions before deeming a lens suitable for use. Fig. 6.1.1 depicts the typical deficiencies of radial polynomial distortion models when used to model the distortion of zoom lenses.

The conventional solution to deal with these non-radial distortions is to complement the radial distortion model with a *tangential distortion model,* as described in [6]. However, in the case of zoom lens-distortion calibration, this additional distortion model just exacerbates our primary problem: we now have to choose models of appropriate complexity for both radial and tangential distortion at different zoom levels.

As an alternative, in this chapter we adapt the method of Sec. 5.2 to estimate a non-parametric model of distortion in a zoom lens. In our experiments, we also show that simple linear interpolation of this non-parametric distortion across zoom levels produces an accurate characterization of lens distortion in a zoom lens.
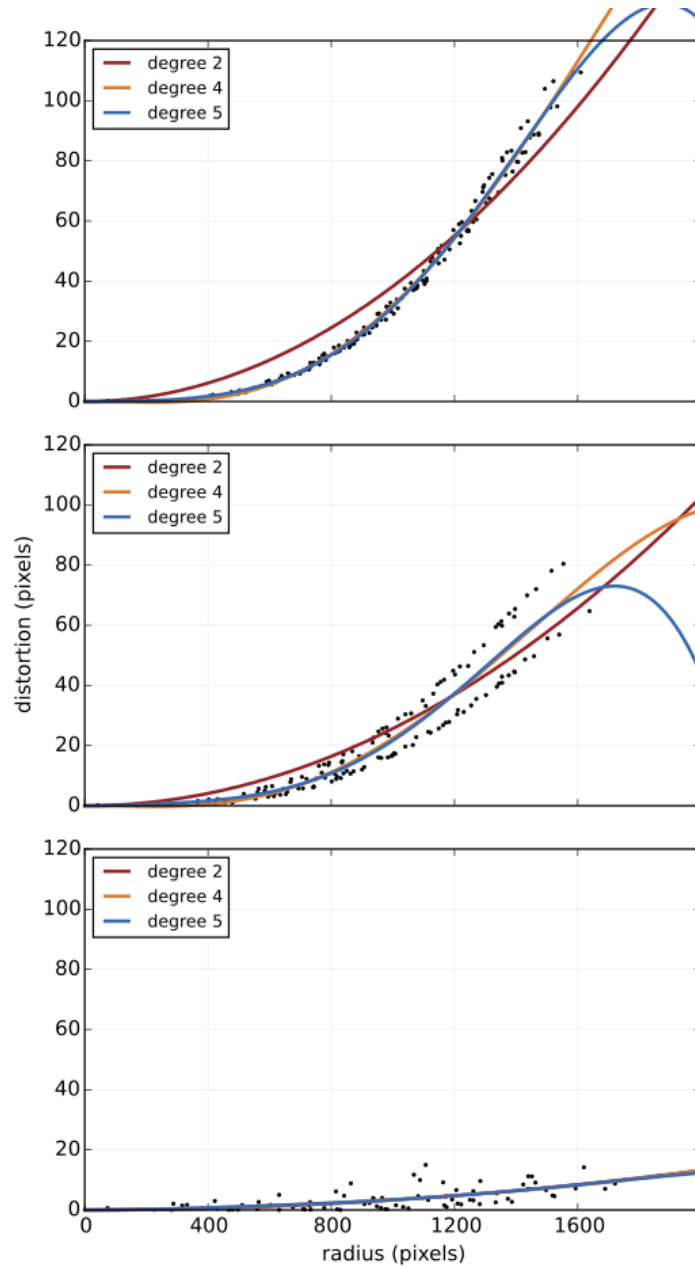
Figure 6.1.1: *Deficiencies of radial polynomial distortion models. The three plots show distortion observations (black points) from three different zoom levels in a Nikon 18-200 mm zoom lens. Camera models with three different degrees of radial polynomial distortion were calibrated at each zoom level. In the first two plots, the degree-2 polynomial model produces biased results, but shows good performance in the last plot. In the second plot, the degree-5 polynomial has low bias, but extrapolates badly in areas with sparse data. Thus, a different complexity of polynomial is appropriate at each zoom level. Also, the second plot exhibits correlations that group observations at higher radii into two distinct clusters. This non-Gaussian error profile suggests that some aspect of the underlying distortion is not completely captured by the radial distortion model.*

44

6.2 Modeling distortion across zoom levels

The initial step in our method for building a zoom lens distortion model is to capture a model of distortion at a fixed set of zoom levels. We use technique of Sec. 5.2 for capturing lens distortion at a fixed zoom level. The important piece of information captured, at each zoom level, by this model is $\mathbf{H}_0$ the homography at the center of the image.

Once we have an estimate of $\mathbf{H}_0$ at each zoom level for the camera, we can obtain camera intrinsics as well as distortion for each zoom level as explained in Sec. 5.3.1. However, we can obtain a better estimate if we perform a joint calibration using multiple images, as explained below.

6.2.1 Non-linear refinement

To perform a joint calibration, we begin by acquiring multiple planar target images at each zoom level, while following to the guidelines of Sec. 3.4.1. Given the local homography at the center for each image, $\mathbf{H}_0$, the camera intrinsics matrix $\mathbf{K}$ and the extrinsics matrix $\mathbf{E}$ can be approximately recovered using the homography decomposition described in Sec. 3.4.3. Let

$$\mathbf{H}' = \mathbf{KE}$$

Ideally, $\mathbf{H}_0 = \mathbf{H}'$. But this is generally not the case because the homography decomposition is approximate. As a consequence, the re-projection error of world points through $\mathbf{K}$ and $\mathbf{E}$ has significant errors. To remove these errors, we minimize the re-projection error using the *Levenberg-Marquadt* optimization [22]. This optimization is structured such that the set of images acquired at one zoom level have the same intrinsics but different extrinsics corresponding to different poses of the calibration target.

In this non-linear refinement step, we essentially perform classic calibration at each zoom level with the following modifications:

1. We use the local homography, $\mathbf{H}_0$, to get initial estimates of intrinsics and extrinsics. In classic camera calibration, the initial estimates are obtained from a homography that is fit to all correspondences in the image.

2. We ignore the role of distortion. Instead, we weight re-projection errors according to the weighting function of the locally-weighted homography $\mathbf{H}_0$.

The use of weighted re-projection errors implicitly compensates for distortion. This is because, by construction, the weights of $\mathbf{H}_0$ were optimized to produce a locally optimal model while compensating for distortion in the neighboring correspondences.

6.2.2 Interpolating distortion across zoom levels

At each zoom level, we obtain distortion observations and build a predictive Gaussian Process model as explained in Section 5.2. The Gaussian Process model allows us to predict distortion within a calibrated zoom level. To predict the distortion at an intermediate zoom level, we linearly interpolate between the GP predictions of the two bracketing calibrated zoom levels.

As a concrete example, let $d_z(x, y)$ be the predictive model of distortion at pixel (x, y) for zoom level z. Suppose we require the distortion at zoom level $24\,mm$, but we only have calibrated models at $20\,mm$ and $26\,mm$. We obtain a distortion prediction for $24\,mm$ by linearly interpolating between $d_{20}(x, y)$ and $d_{26}(x, y)$

$$d_{24}(x, y) = \alpha\, d_{20}(x, y) + (1 - \alpha)\, d_{26}(x, y)$$

where $\alpha = \frac{(26-24)}{(26-20)} = \frac{1}{3}$ is the interpolation ratio

6.3 Experiments

We use a Nikon 18-200 mm auto-focus zoom lens mounted on a Nikon D7100 camera for our experiments. We calibrate the zoom lens at thirteen different zoom levels between the nominal focal-lengths of 18 mm and 70 mm.

The nominal focal-length used while acquiring an image is available from the EXIF meta-data for that image. This focal-length, expressed in millimeters, corresponds to a zoom level setting and varies with the focal-length in pixels. However, the exact relationship between zoom level and focal-length in pixels needs to be estimated as part of the calibration.

6.3.1 Intrinsics model

With the chosen lens, we proceed with our method as described in the previous section. This gives us point-estimates of the focal-lengths f_x and f_y, and the principal point (c_x, c_y) at each zoom level, shown in Fig. 6.3.1.

For the zoom camera, we are interested in a model of how the intrinsic parameters vary over a continuous range of zoom levels. We have point-estimates of the parameters, but information about the distribution of these parameters will help us make an informed decision about the quality of calibration and choose a good interpolating model. To obtain the distribution of these intrinsic parameters, we perform *bootstrap sampling* by estimating multiple models using subsets of the calibration images. The resulting distribution for the intrinsics parameters is shown in Fig. 6.3.1.

Also shown in Fig. 6.3.1 are the interpolation models for the intrinsics parameters. Since we have a large number of samples of the intrinsic parameters, we can potentially fit high-degree polynomials to this data. But we observed that modeling accuracy does not improve beyond a fourth-degree polynomial for f_x and f_y, and a sixth-degree polynomial for c_x and c_y.

6.3.2 Distortion model

Our method also gives us a Gaussian process model of the lens-distortion at each zoom level. There are two aspects of this distortion model that need to be evaluated: how good the distortion model is at a calibrated zoom level, and how good the distortion model is at interpolating distortion to intermediate zoom levels.

First, to present a qualitative evaluation of the distortion model at a calibrated zoom level, we plot the model residuals at three different zoom levels in Fig. 6.3.2. We observe that the residuals conform to an uncorrelated 2D Gaussian distribution, suggesting good modeling performance. In contrast, the residuals of the radial polynomial model in Fig. 6.1.1 present a clear non-Gaussian profile, where they cluster into two groups on either size of the regressor. The residual distribution at other zoom levels is similarly Gaussian, though we do not show results for every level due to space constraints.

For a quantitative evaluation of the lens-distortion model, we make use of the fact that a good distortion model should rectify an image such that known straight lines map to straight lines in the image. Any image of the planar calibration target contains a number of straight lines corresponding to the rows and columns of fiducials, allowing us to evaluate the straight-line rectification performance. As a quantitative measure of rectification, we report mean residual errors for the lines fit to the rectified row and column centers and compare rectification performance of the proposed model with the rectification performance of a sixth degree radial

polynomial model.

First, to evaluate the distortion rectification at a calibrated zoom level, we obtain a separate set of test images at the calibrated zoom levels. We then use the distortion model to rectify the image and plot residual error statistics from the line fit in Fig. 6.3.4 (a).

Next, to evaluate the distortion rectification at intermediate zoom levels, we obtain test images at a set of intermediate zoom levels. We then use the interpolated model described in Section 6.2.2 to rectify the test image. We plot residual error statistics from the line fit in Fig. 6.3.4 (b). In both cases, we observe less than one pixel of error, confirming good performance for the distortion model. Also, the proposed distortion model produces residual errors that are significantly lower than the residual error of the radial polynomial model.

6.3.3 Discussion

From the data in Fig. 6.3.1, we see that the variation in focal length is well behaved and exhibits a smooth trend across zoom levels. However, the same cannot be said for the variation in the principal point: the trend is mostly smooth till a zoom level of 35, after which the trend exhibits a significant change in smoothness. It is hard to reason about this behavior without explicit knowledge about the internal optics of the lens used in our experiments. Nevertheless, our non-parametric distortion model shows good modeling performance and is able to rectify distortions across the calibrated zoom range.

The Nikon 18-200 mm lens used in this evaluation is primarily targeted for use in photography. It exhibits minimal distortion near the nominal focal length of 30 mm, which is taken as the *normal* focal length in photography, where the field-of-view is closest to the field-of-view of the human eye.

This is depicted in Fig. 6.3.3, where we plot the mean and maximum distortion magnitudes across zoom levels, as estimated using our distortion model.

6.4   Summary

In this chapter, we adapted the non-parametric distortion model of Chap. 5 to model distortion in a zoom lens. This model is suitable for a zoom lens because it captures both the radial and non-radial components of distortion. We applied the proposed method to a Nikon 18-200 mm auto-focus zoom lens and evaluated its accuracy by

quantifying its straight-line rectification performance. Our results showed that the non-parametric model can accurately rectify distortion across the calibrated zoom range.
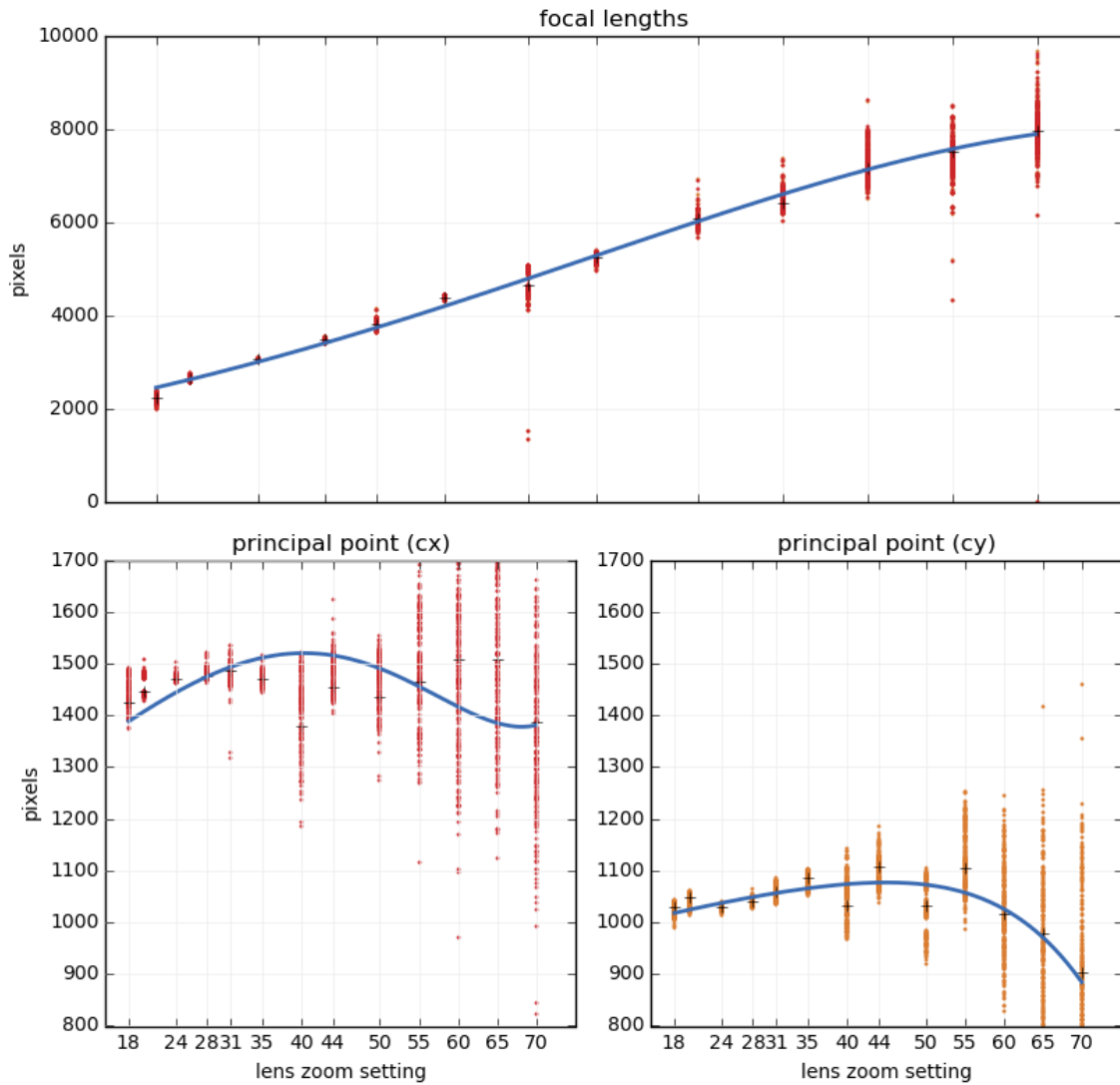
***Figure 6.3.1:** Variation in camera parameters across zoom levels. Point estimates of parameters obtained after non-linear refinement are shown with a + marker. Bootstrap samples showing the distribution of parameter values are shown as points along the y-axis. The interpolating polynomials are drawn in blue. The top plot shows the variation in the focal-length parameters. Values of f_x and f_y are almost equal and are indistinguishable in the plot. The bottom plot shows the variation in the principal point. In general, we observe that the parameter estimates are less confident with increasing zoom.*
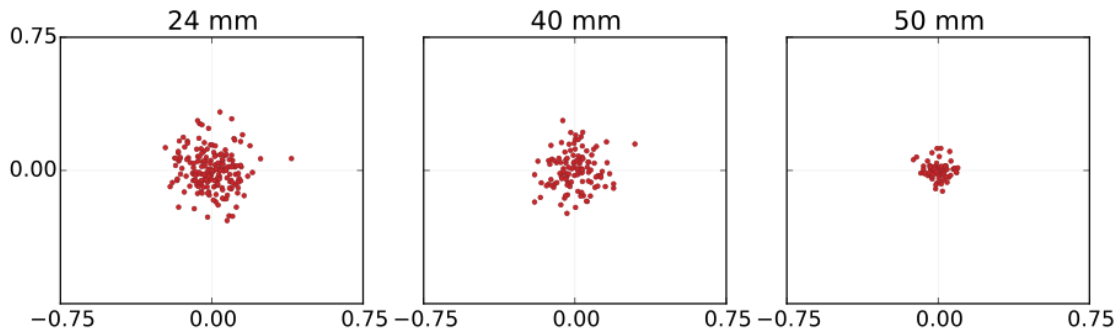
Figure 6.3.2: *Distortion model residuals at different zoom levels. In this figure we show distortion model residual distributions at zoom levels of 20 mm, 40 mm and 55 mm. In contrast to the residuals in Fig. 6.1.1, the residuals conform to uncorrelated Gaussians. Samples are sparser with increasing zoom level because we observe fewer correspondences with increasing zoom.*
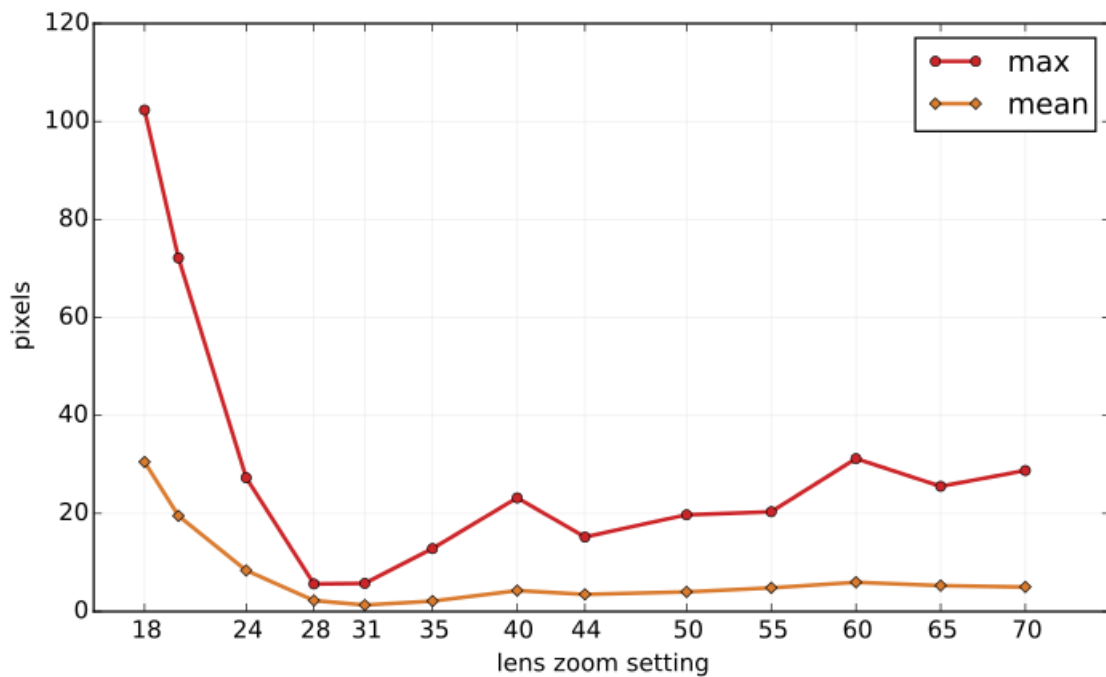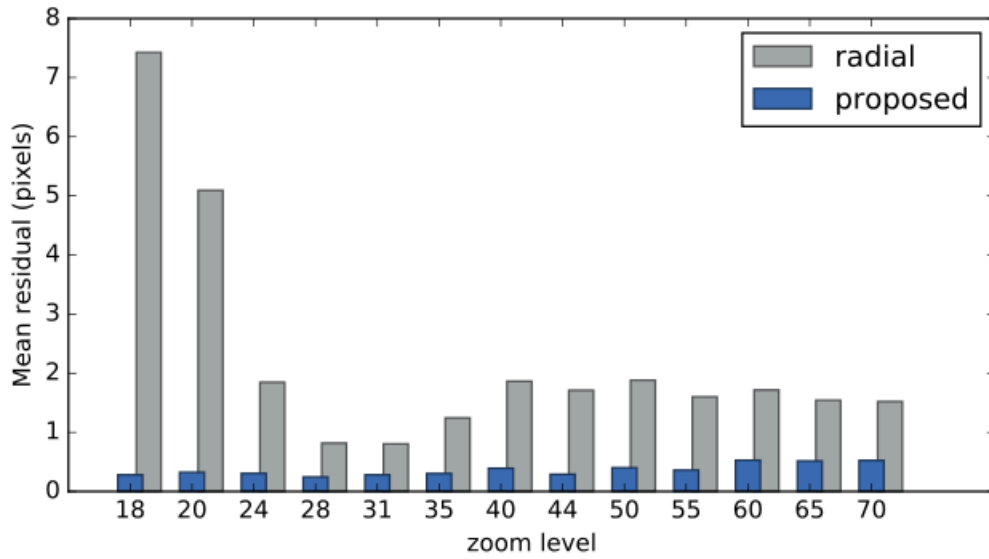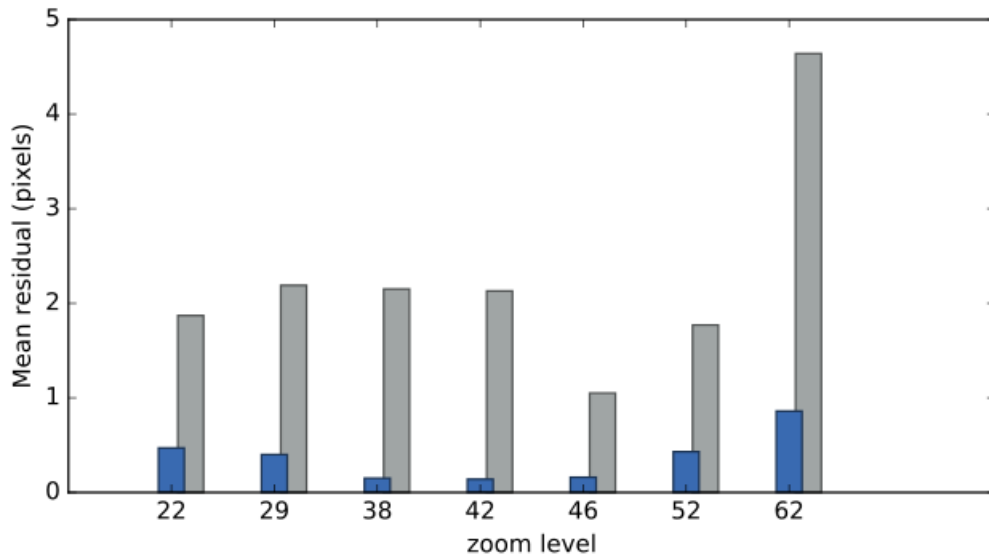

Figure 6.3.3: *Variation in distortion magnitude with zoom for a Nikon 18-200 mm zoom lens. The lens distortion is minimal near the nominal focal length of 30 mm.*

(a) Straight line rectification error for calibrated zoom levels

(b) Straight line rectification error for interpolated zoom levels

Figure 6.3.4: Straight line rectification residual errors. First, test images containing known straight lines were rectified using both the proposed distortion model and a sixth degree radial polynomial model. Next, we fit straight lines to the points that must lie on a straight line after rectification. The residual error statistics from the line fit are plotted here. Plot (a) shows error statistics at calibrated zoom levels. Plot (b) shows error statistics for interpolated zoom levels. In both cases we observe that the residual error of the proposed model is significantly lower than the residual error of the radial polynomial model.

Chapter 7

Conclusion

Cameras are commonly used to capture pictures of objects and scenes. However, once the transformation from image pixels to object sizes is identified using a process termed camera calibration, a camera can also be used to measure distances and sizes. When a calibrated camera is used for surveying or surveillance, the accuracy of its calibration greatly affects the quality of image registration. Often, cameras used for surveying and surveillance capture a wide field-of-view and employ lenses with significant distortion. Accurate calibration of these cameras also requires accurate calibration of lens distortion.

Traditional models of lens distortion are motivated by the physical construction of lenses. It is common to use a radially-symmetric polynomial model of lens distortion when calibrating a camera because the distortion of lenses is predominantly radial. However compound lens systems might exhibit lens distortions that are significantly non-radial or exhibit symmetry about a point that is not the center of the image. In these cases, the practitioner seeking to calibrate the lens distortion must try out different augmentations to the standard model.

In this dissertation, we proposed alternative non-parametric models of lens distortion. Unlike conventional parametric descriptions of lens distortion, these non-parametric methods flexibly and automatically capture the various sources of distortion in a lens system.

We began in Chap. 4 by proposing a Gaussian Process model of radial distortion that automatically adjusted the complexity of the lens distortion model. First, we showed how the non-linear least squares camera calibration problem can be viewed as maximum likelihood inference in a factor graph. Next, we extended this formulation by showing how a Gaussian process node and factor can be embedded in this factor graph to non-parametrically describe distortion in the camera lens. We

then compared the performance of classic polynomial models and Gaussian process distortion models on camera calibration tasks by reporting a principled test error metric. Our evaluation showed that the Gaussian process distortion models performed on par with the best polynomial models while automatically performing model selection.

Next, in Chap. 5, we introduced the concept of a locally-weighted homography. Then, we formulated a non-parametric definition of lens-distortion by analyzing the difference between correspondence predictions by the local homography at the center of the image and the actual observed point correspondences. We showed that this non-parametric definition of lens-distortion can correct arbitrary sources of distortion in an imaging system. Furthermore, we showed that augmenting classic camera calibration using this non-parametric model of distortion produced better convergence of the optimization and reduced the overall probability of errors.

Finally, in Chap. 6, we used the locally-weighted homography based distortion model on a zoom lens. We showed that this model is suitable for a zoom lens because it captures both the radial and non-radial components of lens distortion. We applied the proposed method to a Nikon 18-200 mm auto-focus zoom lens and showed that the non-parametric model accurately rectifies distortion across the calibrated zoom range.

Software for the different models described in this dissertation has been made publicly available at https://github.com/memorydump85/zoomcalib. As an extension to the work in this dissertation, the method of local homographies described here may be extended to analyze the distortion in a catadioptric system. Catadioptric systems use both reflective and refractive elements to focus light rays on an imaging plane. The distortion in a catadioptric system is best described by a camera that undergoes motion to capture the image. The method of locally-weighted homographies describes the distortion as a continuous mosaic of homographies. Therefore, it is well suited for explaining distortion as motion in the imaging system.

Bibliography

[1] L. De Agapito, E. Hayman, and I. Reid. Self-calibration of a rotating camera with varying intrinsic parameters. In *In Proc 9th British Machine Vision Conf, Southampton*, pages 105–114, 1998.

[2] Joao Barreto, Jose Roquette, Peter Sturm, and Fernando Fonseca. Automatic Camera Calibration Applied to Medical Endoscopy. In *BMVC 2009 - 20th British Machine Vision Conference*, September 2009.

[3] P. A. Beardsley and D. W. Murray. Camera calibration using vanishing points. In *Proceedings of the British Machine Vision Conference*, pages 43.1–43.10. BMVA Press, 1992. doi:10.5244/C.6.43.

[4] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[5] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[6] D. C. Brown. Decentering Distortion of Lenses. *Photometric Engineering*, 32(3):444–462, 1966.

[7] R. Cipolla, T. Drummond, and D. Robertson. Camera calibration from vanishing points in image of architectural scenes. In *Proceedings of the British Machine Vision Conference*, pages 38.1–38.10. BMVA Press, 1999. doi:10.5244/C.13.38.

[8] T. A. Clarke and J. G. Fryer. The development of camera calibration methods and models. *The Photogrammetric Record*, 16(91):51–66, 1998.

[9] D. Claus and A.W. Fitzgibbon. A rational function lens distortion model for general cameras. In *Computer Vision and Pattern Recognition, 2005. CVPR*

2005. IEEE Computer Society Conference on, volume 1, pages 213 – 219 vol. 1, june 2005.

[10] David Claus and Andrew W. Fitzgibbon. A rational function lens distortion model for general cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 213–219, June 2005.

[11] W. S. Cleveland. Robust Locally Weighted Regression and Smoothing Scatterplots. *Journal of the American Statistical Association*, 74:829–836, 1979.

[12] Frédréric Devernay and Olivier Faugeras. Straight lines have to be straight: Automatic calibration and removal of distortion from scenes of structured enviroments. *Mach. Vision Appl.*, 13(1):14–24, August 2001.

[13] M.T. El-Melegy and A.A. Farag. Nonmetric lens distortion calibration: Closed-form solutions, robust estimation and model selection. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 554–559. IEEE, 2003.

[14] J.M. Frahm and R. Koch. Camera calibration with known rotation. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1418 –1425 vol.2, oct. 2003.

[15] S. Gasparini, P. Sturm, and J.P. Barreto. Plane-based calibration of central catadioptric cameras. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1195–1202, Sept 2009.

[16] R. Hartley and Sing Bing Kang. Parameter-free radial distortion correction with center of distortion estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(8):1309 –1321, aug. 2007.

[17] Richard I. Hartley. Self-calibration from multiple views with a rotating camera. pages 471–478. Springer-Verlag, 1994.

[18] J. Heikkilä. Geometric camera calibration using circular control points. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1066 – 1077, oct 2000.

[19] J.Bouguet. Camera calibration toolbox for matlab, 2010.

[20] Stephen J. Maybank and Olivier D. Faugeras. A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, 8:123–151, 1992.

[21] Jorge Moré. The levenberg-marquardt algorithm: Implementation and theory. In G. Watson, editor, *Numerical Analysis,* volume 630 of *Lecture Notes in Mathematics,* pages 105–116. 1978.

[22] J. Nocedal and S. Wright. *Numerical Optimization.* Springer Series in Operations Research and Financial Engineering. Springer New York, 2006.

[23] 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.

[24] Michael JD Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The computer journal,* 7(2):155–162, 1964.

[25] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning).* The MIT Press, 2005.

[26] Carlos Ricolfe-Viala and Antonio-José Sánchez-Salmerón. Correcting nonlinear lens distortion in cameras without using a model. *Optics and Laser Technology,* 42(4):628 – 639, 2010.

[27] Carlos Ricolfe-Viala and Antonio-Jose Sanchez-Salmeron. Using the camera pin-hole model restrictions to calibrate the lens distortion model. *Optics and Laser Technology,* 43(6):996 – 1005, 2011.

[28] Sudipta N. Sinha and Marc Pollefeys. Pan-tilt-zoom camera calibration and high-resolution mosaic generation. *Comput. Vis. Image Underst.,* 103(3):170–183, September 2006.

[29] Peter Sollich and Christopher K. I. Williams. Understanding gaussian process regression using the equivalent kernel. In *Deterministic and Statistical Methods in Machine Learning,* pages 211–228, 2004.

[30] Reg G. Willson. Modeling and calibration of automated zoom lenses. Technical report, 1994.

[31] Ziyan Wu and R.J. Radke. Keeping a pan-tilt-zoom camera calibrated. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 35(8):1994–2007, August 2013.

[32] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, November 2000.