A New Camera Calibration Accuracy Standard for Three-Dimensional Image Reconstruction Using Monte Carlo Simulations

by

Nickolas Arthur Stenger

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved June 2012 by the
Graduate Supervisory Committee:

Antonia Papandreou-Suppappola, Chair
Narayan Kovvali
Cihan Tepedelenlioglu

ARIZONA STATE UNIVERSITY

August 2012

ABSTRACT

Camera calibration has applications in the fields of robotic motion, geographic mapping, semiconductor defect characterization, and many more. This thesis considers camera calibration for the purpose of high accuracy three-dimensional reconstruction when characterizing ball grid arrays within the semiconductor industry. Bouguet's calibration method is used following a set of criteria with the purpose of studying the method's performance according to newly proposed standards.

The performance of the camera calibration method is currently measured using standards such as pixel error and computational time. This thesis proposes the use of standard deviation of the intrinsic parameter estimation within a Monte Carlo simulation as a new standard of performance measure. It specifically shows that the standard deviation decreases based on the increased number of images input into the calibration routine. It is also shown that the default thresholds of the non-linear maximum likelihood estimation problem of the calibration method require change in order to improve computational time performance; however, the accuracy lost is negligable even for high accuracy requirements such as ball grid array characterization.

To my father who taught me to strive for excellence

To all those that questioned my intelligence

To my mother who could not be with me today

To my wife for her unconditional love and support

telling me that it will always be okay

ACKNOWLEDGEMENTS

At times it may have felt like it due to my stubborn unwillingness to take a deep breath, but never was I completely alone in my endeavours.

First and foremost, I am deeply thankful to my adviser Dr. Antonia Papandreou-Suppappola. Her willingness to take my dissertation in the last minute has left me indebted. She has constantly provided me with a steady eye and the respect of an equal. I am thankful to have worked with and most importantly to have known her.

I am also grateful to Dr. Narayan Kovvali and Dr. Cihan Tepedelenlioglu for serving on my committee as well as Esther Korner for the thankless help with paperwork and forms throughout the graduate school. Dr. Lina Karam must also be thanked for first giving me the opportunity to work on her research staff that lead me to work with Intel Corporation. If it was not for her early opportunity, I would not have met so many pleasant people and accomplished so much. Of course, I would also like to thank Intel Corporation for giving me the resources to make this happen.

Most importantly, I would like to thank my wonderful wife Hayley. She has been my never-ending support throughout the stressful, work-filled nights. Never would I have accomplished such a feat without gaining so few of gray hairs. I am a better person having lived amongst her smile.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Three-dimensional (3D) image reconstruction is the process of capturing the shape and position of real objects or points represented in three dimensions in the physical world or a simulated space [1]. 3D reconstruction methods can usually be divided among active and passive methods. Active methods can interfere with the object in some physical sense, either by moving light over the object or by using a time of flight laser. Passive methods use only imaging sensors such as those found in a single camera or in multiple cameras for stereo and multi-view reconstruction.

Camera calibration for 3D reconstruction is the process of acquiring the parameters of a camera and lens assembly. In particular, the process describes how an object is captured and projected onto the camera's internal sensor and provides the position of the camera in space when compared to a fixed reference point. In more explicit terms, a camera model is defined to have intrinsic and extrinsic parameters. Intrinsic parameters model how light passes through the camera lens and is projected onto the camera sensor using parameters such as the focal length of the lens and any distortions of the lens that may appear due to its construction. Extrinsic parameters describe the position and direction of the camera system in space.

Both the intrinsic and extrinsic parameters are extremely important depending on the type of application, including 3D reconstruction using passive stereo or active ranging, robot navigation, and any photogrammetric approach for finding metric information from two-dimensional (2D) images. Due to the problems expected with perspective projections, all objects in the world with shape, when imaged, appear to have a different shape on the image due to the orientation of the camera with respect to the world object. An example of a ladder before and after perspective projection is shown in Fig 1.1.

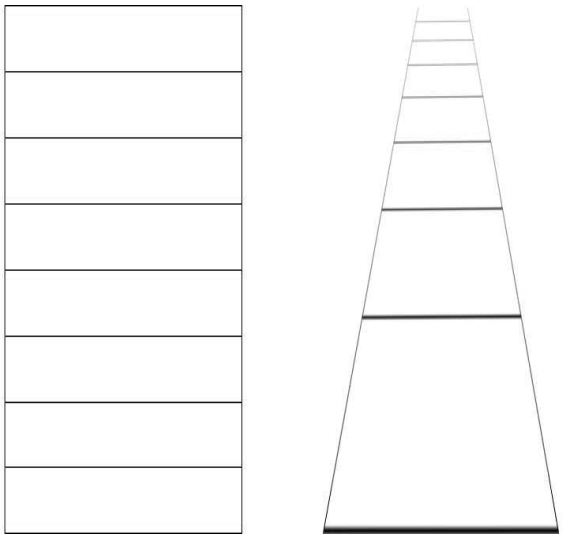

Figure 1.1: Illustration of the perspective problem using the image of a ladder. The left image shows the ladder as typically seen by a viewer perpendicular to the object. On the right is the expected image if the top of the ladder is tilted backwards away from the viewer. Some parallel lines do not stay parallel and begin to converge towards the top of the ladder.

The only exception is when the camera is positioned coplanar to another planar surface, in which case only the object size is changed. In the typical perspective projection problem, all light rays pass through the lens center. This, however, is not entirely complete in a real world model with a lens that actually has size and shape. Because of this, non-linear lens distortions are introduced to the image.

3D image reconstruction is typically accomplished in a three forked approach: camera calibration, feature point selection, and point triangulation. Although other approaches exist that that do not require explicit camera calibration and gain pseudo calibration terms within the triangulation phase, such as with structured lighting [2], this work concentrates on camera calibration. The 3D reconstruction requires both the intrinsic and extrinsic parameters gained from the camera calibration process specifically, when attempting 3D triangulation of two or more cameras, the position and direction of each camera are required in order to take the corresponding points in multiple cameras and compute their individual

depth. Also, without knowing the extrinsic parameters of the cameras, depth can only be expressed in terms of pixels and not real world metrics.

1.1 Application Space

Improving performance measures in camera calibration for image reconstruction has many applications. One application of interest in this work is the 3D characterization of semiconductor packages that have a ball grid array (BGA) pattern via optical stereo imaging [3]. As these BGAs are used for communication between the package and the motherboard processor socket that it sits in, problems occur when the BGAs have incorrect ball height or experience package level warpage [4]. Both of these problems can independently cause shortages and/or open circuits when placed into the processor socket causing mother board failures. The current process tool to inspect the packages does not inspect each individual solder ball and does not output package warpage. In order to individually find the height of each solder ball and output package warpage, a stereo method with high-resolution cameras is chosen to experiment with. The hundreds of solder balls, that constitute a full BGA, range from 60-300 microns in height and require high precision accuracy due to the low tolerances accepted.

This application can be placed onto a manufacturing floor where thousands upon thousands of individual semiconductor units can pass through for inspection every day. It is absolutely critical that the solution implementation be quick just as it is accurate. Thus, there is a need for both quick and extremely precise stereo camera calibration.

1.2 Thesis Contributions

Pixel error has been a long standing measure of camera calibration accuracy. However, it has a disadvantage that it does not correctly characterize the expected results when under a basis of many input images. It has seperately been shown that adding more images or information to the calibration routine should increase the calibra-

tion accuracy. However, only slightly, pixel error trends upwards over increasing number of images well beyond the minimum required. By the use of Monte Carlo simulations, we show that the standard deviation of an estimated camera model parameter can be used as an alternative form of camera calibration accuracy.

Following the requirements of our application space, we also show the default baseline case of the camera calibration method developed by Bouguet [5]. Using Monte Carlo simulations and an analysis of the default case, we show that the default thresholds of the maximum likelihood estimator has room for improvement in our application. For example, the thresholds governing how long the optimizer can run should be reduced in order to not waste computational time. In tandem, the thresholds can be modified to provide insignificant accuracy loss as well.

1.3 Thesis Organization

The rest of this thesis is organized as follows. A concise background into camera calibration is discussed in Chapter 2. All of the coordinate systems and individual models are introduced in order to build up to the final perspective camera model. Chapter 3 discusses the core camera calibration method used for the duration of this thesis. Such details include the image and algorithm setup and the individual methods employeed in order to receive the final camera model parameters that can describe the world to image projection. Chapter 4 details the proposed work in this thesis. The Monte Carlo method is introduced as well as our individual setup and calibration object. We show a new accuracy metric in the form of the variation of an estimated intrinsic camera model parameter. We also make a small improvement to the settings of the camera calibration method in order to reduce computational run time significantly while experiencing an insignificant amount of accuracy lost.

Chapter 2

BACKGROUND IN CAMERA CALIBRATION

2.1 Introduction to Camera Calibration

Camera calibration is an important area of research as it is usually required for 3D image reconstruction. The calibrated parameters expelled from the calibration procedure for a single camera are formed into a full bodied camera model that describes the relationship between a point on an object and its corresponding point in the image. In order to fully capture all needed parameters for stereo and multi-view 3D image reconstruction, camera calibration must be ran for each camera.

2.2 Coordinate Systems and Camera Model

To understand the geometry of the relationship between a point on an object and its corresponding point in the image requires an understanding of the many coordinate systems and the camera model that governs the projection of the object via the lens. This can be described by building the coordinate systems from the object to the image or the image to the object. The procedure involves starting from world coordinates and building the systems to the image coordinates.

The camera 3D coordinate system *(x,y,z)* is a viewer centric system centered at the projection center of the lens. This coordinate system is arbitrarily positioned relative to the world coordinate system *(X,Y,Z)* and referred to as the extrinsic properties of the camera. The coordinate systems do not change size or shape; however, they can change orientation and position. In particular, the camera coordinate vector $[\mathrm{x\ y\ z}]^{\mathrm{T}}$ and the world coordinate vector $[\mathrm{X\ Y\ Z}]^{\mathrm{T}}$ are related by

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{R} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \tau \tag{2.1}$$

where $\mathbf{R}$ is a 3×3 rotation matrix, τ is a 3×1 translation matrix, and $\mathtt{T}$ denotes vector transpose.

The intrinsic camera parameters describe how the image is projected from the camera system through the projection center and onto the image plane $[\mathrm{I\ J}]^{\mathrm{T}}$ centered at the principle point $[I_0\ J_0]^{\mathrm{T}}$. Other intrinsic parameters include the effective focal length f and the scale factor s. This projection of the object via the lens has routinely been expressed by the pin-hole camera model [6]:

$$\begin{bmatrix} I \\ J \end{bmatrix} = \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2.2}$$

The $[\mathrm{I\ J}]^{\mathrm{T}}$ plane is assumed to be coplanar to the $[\mathrm{x\ y}]^{\mathrm{T}}$ plane as they are represented by a linear relationship only defined by f and z. The *(I,J)*, *(x,y,z)*, and *(X,Y,Z)* coordinate systems are depicted in Fig 2.1.

The characterization of an image involves demonstrating the coordinate system of the image array using either a charge-coupled device (CCD) or a complementary metal-oxide semiconductor (CMOS) sensor showing how the sensor is being illuminated by light through the lens [7–9]. This shall be called the pixel coordinate system (I', J'). With today's matured technology, the image sensor is almost always in a square grid format which means the skew factor s equals one. The grid is expressed in rows and columns with the origin typically at the upper-left most corner pixel due to a common image processing ritual. The rows and columns express that the pixel coordinates are in integer format. However, this pixel coordinate system has no real-world length value. Thus, the pixel coordinate system is projected onto the camera sensor and can thus be expressed by the image plane coordinates *(I,J)*.

The image plane is ideally expressed by the image coordinate system using the spacing between adjacent columns and rows on the sensor as well as the principal axis of the lens. The principal axis of the lens dictates the center of the image plane. Although, in order to be ideal, the lens must be manufactured and installed

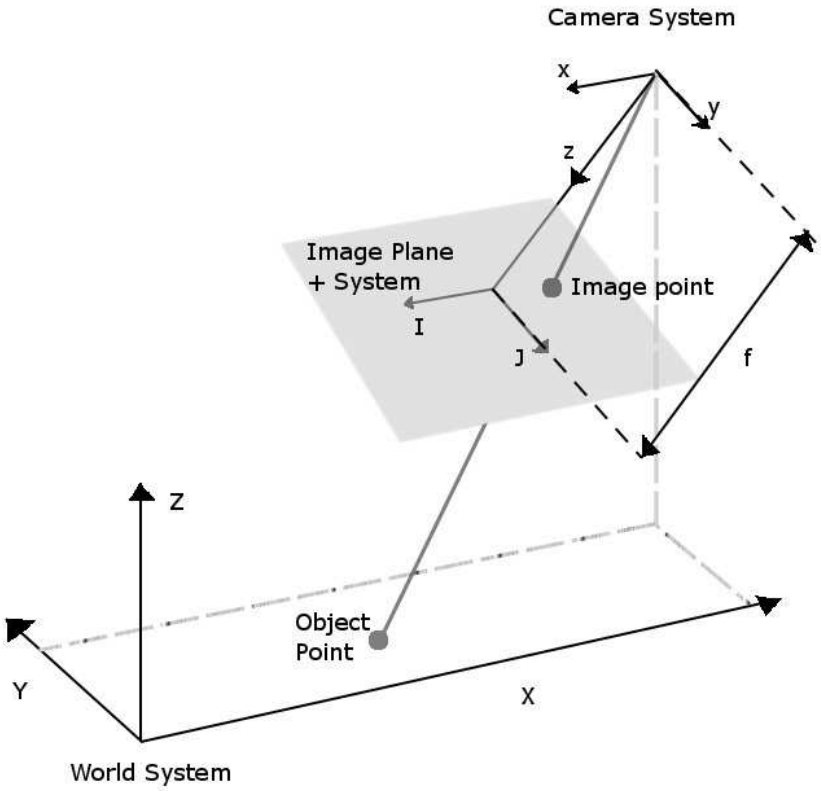

Figure 2.1: Demonstration of the 3D coordinate system of the intrinsic pin-hole camera model arbitrarily placed according to the world coordinate system as the extrinsic model.

to the camera body well. In this case, the principal axis is very close to the center of the camera sensor, which corresponds to the center of the image coordinate system. The spacing between adjacent columns and rows on the sensor is almost always the same, implying a square grid format. This spacing is what dictates the integer spacing between pixels on the image coordinate system and is referred to as ”pixel size” in camera specifications (S_I, S_J). The pixel coordinate system and image plane coordinate system are coplanar to each other; therefore, they can be distinguished by a constant factor for each axis. The skew factor s dictates the ratio multiplier for rectangular grids. However, almost all cameras manufactured today contain grid patterns so s can be idealized to 1. This system is the principle method

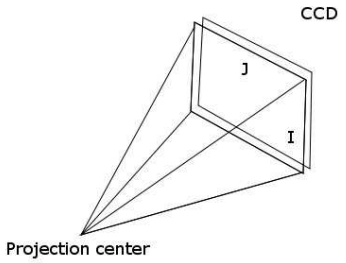

Figure 2.2: Image plane projected through the projection point onto the CCD producing the pixel plane.

to relating pixels on the image coordinate system and real world location on the image plane according to

$$\begin{bmatrix} I' \\ J' \end{bmatrix} = \begin{bmatrix} S_I \, s \, I_D \\ S_J \, J_D \end{bmatrix} + \begin{bmatrix} I_0 \\ J_0 \end{bmatrix} \tag{2.3}$$

The final camera model is used for high accuracy calibration. It uses a much more complete and condensed homogeneous matrix form

$$\lambda \begin{bmatrix} I_D \\ J_D \\ 1 \end{bmatrix} = \begin{bmatrix} s\,f & 0 & I_0 & 0 \\ 0 & f & J_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \tau \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{F} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.4}$$

where λ is a scale factor, $\mathbf{0}$ is a 1×3 row vector of zeros, and $\mathbf{F}$ is the fundamental matrix that describes the complete projection [6]. This camera model is only an approximation of the real camera projection model. It is a simple, linear model. However, it is an ideal model that does not account for systematic distortions required for high accuracy calibration as first noted by D.C. Brown [10]. Distortion terms were added to the camera model [6] according to

$$\begin{bmatrix} I_D \\ J_D \end{bmatrix} = \begin{bmatrix} I \\ J \end{bmatrix} + \begin{bmatrix} I_R + I_T \\ J_R + J_T \end{bmatrix} \tag{2.5}$$

where I_R, J_R are radial distortion axis components, I_T, J_T are tangential distortion axis components, and (I_D, J_D) are the distorted image coordinates.

The first important distortion considered is for radial lens distortion that radially displaces pixels outward or inward. The radial distortion can be approximated by a relation to the image plane coordinates $[I_D\ J_D]^{\mathrm{T}}$ given by

$$\begin{bmatrix} I_R \\ J_R \end{bmatrix} = \begin{bmatrix} I_D(k_1 r^2 + k_2 r^4 + k_3 r^6 + ...) \\ J_D(k_1 r^2 + k_2 r^4 + k_3 r^6 + ...) \end{bmatrix} \tag{2.6}$$

where the infinite series real coefficients k_i, $i = 1, 2, \ldots$ are radial distortion parameters, and $r = \sqrt{I_D^2 + J_D^2}$. As it was noted that the model was sufficiently accurate using two radial distortion parameters [6] the radial distortion model becomes

$$\begin{bmatrix} I_R \\ J_R \end{bmatrix} = \begin{bmatrix} I_D(k_1 r^2 + k_2 r^4) \\ J_D(k_1 r^2 + k_2 r^4) \end{bmatrix} \tag{2.7}$$

Another common distortion often considered is tangential or decentering distortion [11, 12]. This type of distortion is often produced by the decentering of curvatures of lens surfaces with respect to each other and the principle axis, and it can arise from non-ideal manufacturing and design of lens and lens assemblies. The resulting tangential distortion vector is given by

$$\begin{bmatrix} I_T \\ J_T \end{bmatrix} = \begin{bmatrix} 2p_1 I_D J_D + p_2(r^2 + 2I_D^2) \\ p_1(r^2 + 2J_D^2) + 2p_2 I_D J_D) \end{bmatrix} \tag{2.8}$$

where p_1 and p_2 are real tangential distortion coefficients [6].

The intrinsic parameter section of the whole camera model is based on the popular pin-hole model [6], and it uses focal length as a parameter. However, the

true focal length of a lens is very different from the focal length parameter used in the pin-hole model. The pin-hole model assumes the lens is essentially a tiny slit where all rays of light pass through. Its effective focal length is the distance from the slit to where the image is formed on the sensor matrix. This is the case also for a slit camera. In reality, a physical lens is a 3D object with a certain height for rays of light to pass through, not an infinitesimally small slit. The true focal length is the distance from the lens center where all rays of light passing through the entire lens converge into a single point. All things being equal, the longer the true focal length of the lens, the better the model is at saying that the effective focal length of the pin-hole model equals the true focal length of the lens.

2.3 Prior Work in Camera Calibration

One of the first introductions of the need to calibrate cameras was by D.C. Brown [10]. He noticed that straight, parallel lines in the world do not transform to straight, parallel lines in the image when the camera is not at an orthogonal angle to the surface. He introduced the distortion extension of the standard camera model of the day.

However, truly whole, accurate camera calibration techniques that are still in use today did not gain traction until the last two decades of the 20th century. Tsai [13] introducted two main approaches: a procedure using a coplanar set of points (coplanar with the optical axis of the camera) and one using a noncoplanar set of points. Tsai's methodology was catered towards speed, efficiency, and low-cost applications. As camera calibration is often a nonlinear process of solving for a large number of unknown parameters, it is typically very difficult and time consuming. Because of his requirement of speed, Tsai used the previously implemented direct linear transformation (DLT) developed by Abdel-Aziz and Karara [14]. The DLT avoids the large-scale nonlinear search by using a set of linear equations, ignoring the parameter dependency.

Zhang implemented a true non-linear optimization technique [1] using the traditional pin-hole camera model with only radial distortion and planar homographies of at least two images to solve for initial parameters (no distortion was included in the initial parameter search). The parameter solution was iteratively solved to minimize projection error in the least-squares sense using maximum likelihood estimation solved with the Levenberg-Marquardt algorithm to improve accuracy. This method used a coplanar target as previous methods had suggested. Note that this method is the basis for the Caltech Camera Calibration Toolbox as seen in the next chapter [5].

Seamingly parallel with Zhang, Heikkila implemented a similiar approach but instead of finding corners, he introduced a method of finding centroids based on the old coplanar method on a 3D target containing two coplanar planes [6, 15]. His camera model was extended to use both radial and tangential distortions. Heikkila also used the DLT to initialize the Lavenberg-Marguardt non-linear search. Heikkila provides an excellent experiment basis to understand the systematic biases present in a camera calibration scheme such as centroid detection, reverse camera model inaccuracies, illumination changes, and the calibration target and its manufacturing tolerances. The latter is a good observation of how the camera calibration requirements have changed - the inherent inaccuracies of the calibration target are now playing a larger role in system inaccuracies as computational power has risen and methods can use extremely powerful optimization techniques.

2.4 Camera Calibration Application Space

Needing to calibrate one or more cameras and/or other devices with cameras can be placed under the large umbrella of ”machine vision”. The idea of camera calibration has been around for over half a century. During the second World War, there grew an increasingly large need for military aerial reconnaissance and mapping that was the catalyst for developing the first camera calibration techniques [16]. Most of the

more modern camera calibration needs have come from the need for 3D reconstruction just as it did in the second World War. Some of these applications range from traditional geopositioning from aerial video or street-view video [17, 18], semiconductor metrology and manufacturing [3, 19], and hand-eye motion tracking such as the infrared and color cameras seen in the Microsoft Kinect for XBox 360. All of these applications need some form of camera positioning in relation to another reference point and intrinsic properties of the camera such as focal length.

Chapter 3

BOUGUET'S CAMERA CALIBRATION APPROACH

3.1 Introduction to Bouguet's Approach

Camera calibration has matured greatly in the last two decades. With the rise of powerful personal computers, the complicated optimization, often non-linear and computationally expensive calibration procedure approaches have become more of an automated reality. J. Bouguet developed a user-friendly calibration approach and implemented as a toolbox in MATLAB provided as freeware [5]. It was developed with Intel and the California Institute of Technology (Caltech) on a MATLAB platform as a means to transfer over to a C implementation for Intel's Open Source Computer Vision library (OpenCV), freely available online as well. This toolbox was created with a graphical user interface (GUI) that accesses most of the toolbox's assets. The aim was for the end user to be able to implement this toolbox quickly for a variety of applications. We decided to use the Bouguet's approach and toolbox for our application because of its strong GUI, ease of MATLAB, many available assets, and broad acceptance within the field of camera calibration as being reliable.

As previously stated, the main source of inspiration for this implementation was based on the non-linear optimization technique first used for camera calibration by Zhang [1]. In fact, all inspirations of this toolbox have previously been published and this was an aggresive exercise in combining many techniques into a full user package.

The full calibration engine consists of three main parts: initialization of the intrinsic parameters, initialization of the extrinsic parameters, and maximum likelihood estimation of the full camera model parameters. As the maximum likelihood estimator, Zhang chose the Levenberg-Marquardt algorithm first implemented computationally in 1978 [20]. It is a nonlinear algorithm designed to minimize the

algebraic distance between two functions in a least squares sense. This algorithm requires initialization of the intrinsic and extrinsic parameters.

3.2 Image and Algorithm Setup

The procedure for running the toolbox itself has been well documented by Bouguet. However, the proper procedure for image acquisition has been sparsely documented. Although the toolbox may have been created for everyday type applications, for high accuracy applications such as the one considered in this thesis, the documentation is not sufficient.

To start using the toolbox, there needs to be a proper calibration rig and a proper calibration procedure. Typically, the calibration rig is a checkerboard or a map of identically distributed circles either in a 2D plane or 3D cube. The toolbox coded as is accepts only a checkerboard pattern, however we adopted the code with minimal effort to use circles. The object is to have as many feature points on the calibration rig as possible as each captured image with more feature points offers more equations for the nonlinear parameter search. This creates an overdetermined set of equations, which is desired and is explained in more detail in Section 3.5. In a checkerboard, these feature points have traditionally been found as the corners of the squares within the outside of the rig. If the calibration rig is a map of identically distributed circles such as ours, the centroids of each circle are the captured feature points; this is refered to as a ”centroid rig”. However, with both of these calibration rigs, the spacing between points is remarkably important and can hinder the results tremendously if not mapped correctly to their true spacing. If high-accuracy 3D reconstruction is the user's application, the calibration rig must be manufactured under high tolerances. Generally, the lowest tolerance of the system dictates the dependence of the overall accuracy of reconstruction.

The user takes images of the calibration rig all with slightly different orientations. This is so the optimization engine sees different perspectives of exactly the

same calibration rig. Having the same pose institutes copies of the same parameters giving the optimization no new projective information. Zhang's optimization engine used for this toolbox operates on more than two degrees of freedom (DOF) [1]. This is different from other optimization methods such as Tsai's [13] that operate on the principle of "radial allignment constraint" or coplanarity between camera frame and object frame. This means that the camera frame must be perpendicular to the object frame - only translation and rotation in x and y are allowed. This is a simplification of the optimization routine proving to be less accurate. Since this toolbox operates on more than two DOF, rotation, tilt, and vertical displacement are utilized in the calibration procedure from image orientation to the next. Vertical displacement is an important distinction to make from method to method. However, one may quickly experience that this must not be taken lightly. Depth of field of a lens system limits just how far of a displacement is allowed as an image that is not in focus gives very poor feature point detection results. All of these distinctions help the optimization resolve the proper intrinsic and extrinsic parameters except for lens distortions.

The four coefficients for radial and tangential lens distortion describe how the entire image is changed from the ideal pin-hole camera model. In order to properly gather as much accurate information about the lens distortions across the entire lens, the object should provide as many feature points across the entire image as possible. This means that the user, along with the previous procedures, should integrate an operation that includes orienting the object away from the center of the image. This can be accomplished by moving the object to different spots in the image frame. An example of this object movement is shown in Fig. 3.1, where information is missing for values of x between -8 and -4. This is a direct analogy to when the calibration object is primarily imaged in the right side of the image frame only. The distortion model parameters may be incorrectly estimated due to

non-distributed feature points within the image frame. However, a more suitable idea is to have a calibration rig that fills the image as much as possible with feature points. One distinction to make is that the lens' field of view cannot be changed to accomodate this procedure. In fact, the lens and camera system cannot be changed in any way from calibration to object imaging. Instead, the calibration rig must be manufactured or setup with the lens' field of view in mind. These procedures have varying importance depending on how distorted the lens is. An extremely distorted lens, such as a fish-eye or wide-angle lens, depends greatly on this procedure to help the optimization find the correct estimate. However, an expensive lens with distortion correction (called "abberation correction" in industry) may actually not need any distortion model estimation depending on the accuracy requirements. In that situation, the distortion parameters could be set to zero and not be part of the optimization routine.

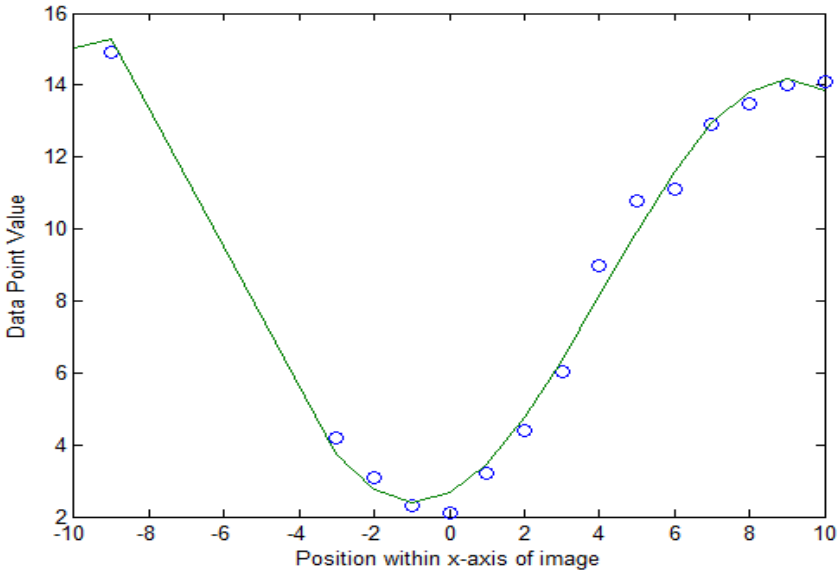

Figure 3.1: An example of a poor polynomial fit due to lack of position data along the x-axis. The example is analogous to missing data on the left side of the image frame required for distortion camera model estimation.

An obvious, but important, distinction is that in a stereo setup using stereo triangulation like in this application, the calibration routine must be ran for both

cameras. Thus, an image set must exist for each camera to accurately provide individual camera parameters. It is important in both experimentation and final design to have the cameras and object be set up to allow for a stable calibration rig and procedure. The cameras must be rigid with respect to the object and each other (the only exception is line-scan cameras, sometimes called ”pushbroom” or ”push-frame”). Also, the calibration rig chosen needs to be stable for imaging purposes yet removable to allow for proper object imaging. For each pose of the calibration rig, each camera must take an image at their respective perspective differences. This enables the cameras to be synchronized to the same calibration procedure. In more detail, the rigid body transformation between the left camera and the calibration rig can be mapped as well as the transformation between the calibration rig and the right camera. It is only once the full extrinsic parameters of each camera are estimated that the full rigid body motion transformation of one camera with respect to another can be found. This information is required for 3D reconstruction using stereo triangulation, among other things. It is also important to ensure that there are no problems with the depth of field or occlusion due to highly oblique angles of the camera with respect to the calibartion rig. Note that, occlusion handling of correspondance points has received a lot of attention in the field of 3D reconstruction [21, 22].

Bouguet has provided a means of feature point extraction for checkerboard patterns. Although this has no baring on the calibration process, a poor feature point extraction algorithm can lead to reconstruction inaccuracies. In order to improve feature point extraction, a segmentation algorithm can be used to segment the circles from its background and find the centroids. From here, each image has a two point coordinate for each feature point that can be used in the subsequent calibration algorithms.

3.3 Intrinsic Parameter Initialization

Both Bouguet and Zhang chose to not explicitly solve for the distortion parameters. Instead, they are set to zero to be initialized [1, 5]. Also, as it was shown that the intrinsic principle point (I_0, J_0) in Equation 2.3 cannot be estimated in a direct manner [13], it is set at the center of the image, following the ideal pin-hole model. Since the skew factor is set to one, the only intrinsic parameter that needs to be initialized for estimation is the focal length f. In Bouguet's implementation, the focal length both in the x and in the y direction is estimated using a method based on the orthogonality of vanishing points, as outlined in [23]. A set of homogeneous linear equations are solved using singular value decomposition (SVD), and the solution is associated with the smallest eigenvalue or the right singular vector of the decomposed vector space. This method is identical to the closed-form solution in [1]. Note that the two parameters differ from each other by the skew factor s, which is initialized to one.

It is important to remember that just because some parameters are set to zero, such as the distortion parameters, that does not mean that they are any less important or unable to converge to a correct estimate. A nonlinear optimization technique always needs an initial guess even if the initial guess is far from the correct estimate. However, the estimator's accuracy may be hindered depending on the algorithm's robustness to outliers as well as the initial guess.

3.4 Extrinsic Parameter Initialization

The rigid body transformation between the camera body point (x, y, z) (with $z = 1$ as the image has no depth information) and the known world points (X, Y, Z), together with the relevant extrinsic parameters, is provided in Equation (2.1). Together with the estimated values of the intrinsic focal length f, principle point (I_0, J_0) and skew parameter s, these provide all the parameters needed in the pin-hole camera model

to back-project any detected feature point into the camera model (x,y). As part of the extrinsic parameter initialization phase, the rigid body transformation between the back projected detected points and the known world points needs to be estimated. This is achieved by finding the 3×3 homography matrix $\mathbf{H}$ between the 3D world points and the expanded homogeneous 2D back-projected points given in

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{3.1}$$

This is accomplished using the SVD method to solve a system of linear equations in a least squares sense. From matrix $\mathbf{H}$, the terms $\mathbf{R}$ and τ in Equation (2.1) can be extracted as the initialized extrinsic parameters of each image.

3.5 Maximum Likelihood Estimation

With the initial guesses of every parameter in our camera model, the non-linear optimization technique based on the maximum likelihood criterion can now be used to obtain the parameter estimates. There are n detected feature point vectors $[u_{ij}\ v_{ij}]$ per calibration object with different poses within m images, $i = 1,\ldots,m$ and $j = 1,\ldots,n$. We can naturally assume that our system is not perfect due to the noise within the camera, the physical limitations in the lens, and the manufactured tolerances of the calibration object. Using the maximum likelihood estimation, the functional p

$$p = \sum_{i=1}^{m} \sum_{j=1}^{n} \left\| \begin{bmatrix} u_{ij} \\ v_{ij} \\ 1 \end{bmatrix} - \hat{F} \begin{bmatrix} X_{ij} \\ Y_{ij} \\ Z_{ij} \end{bmatrix} \right\|^2 \tag{3.2}$$

is minimized where $\|\cdot\|^2$ is the L^2 norm. Here, $\hat{F}$ is the estimated fundamental matrix (from Equation (2.4)) containing all of the estimated intrinsic and extrinsic parameters from their respective initialization phases. Solving this estimation problem involves $m \times n$ equations with 14 unknowns: focal length f, principle point

(I_0, J_0), skew factor s, Euler angles ω, φ, and κ defining the rotation matrix $\mathbf{R}$, and the translation vector $[t_x\ t_y\ t_z]^{\mathrm{T}}$ in Equation (2.1) and the distortion parameters k_1, k_2, p_1, and p_2 from Equations (2.7-2.8). This makes for a highly overdetermined system with many more equations than unknowns. Minimizing the functional p, referred to as pixel error, results in estimating the parameters of the entire camera model as described by $\mathbf{F}$. Bouguet chose to implement this approach following the method that Zhang used - the Levenberg Marquardt algorithm (LMA) for non-linear maximum likelihood estimation [1]. The only minor difference is that Bouguet used the camera model presented by Heikkil and Silven, including two extra distortion coefficients corresponding to tangential distortion [15].

Chapter 4

ANALYSIS OF CAMERA CALIBRATION ALGORITHM PERFORMANCE

Pixel error has been a long standing measure of camera calibration accuracy in the literature. However, it has the disadvantage that it does not always yield the expected results when the number of images increases. It was seperately shown that providing additional images or information to the calibration routine should increase the calibration accuracy [1]. However, albiet slightly, pixel error trends upwards over increasing number of images well beyond the minimum required. In our work, we propose to use a measure of camera calibration accuracy based on the variance (or equivalently, standard deviation) of estimated camera model parameters when Monte Carlo simulations, in the form of multiple input images taken with different perspective poses, are applied.

4.1 Monte Carlo Simulation Trials

In this thesis, we analyze the performance of the Caltech calibration software and its system variables in ways that have not been considered in prior works. Specifically, we consider a new approach to analyze camera model parameter estimation using Monte Carlo simulations. For a single camera, we used 20 images and obtained a calibration set of N images by varying the projection of the calibration object to the camera sensor. The projection was varied by varying the three axis of rotation and the three axis of translation of the calibration object while staying within the depth of field and field of view inherent to the stationary camera setup. A Monte Carlo simulation trial (MCST) consisted of randomly selecting a set of $m < N$ projection images (out of the N possible projections) to be used as calibration input images in Bouguet's calibration toolbox. We performed a total of T=100 MCSTs with each m image subset being a new random permutation of the N image set. We refer to the T trials as a Monte Carlo Simulation Set (MCSS).

There is certainly a distinct chance that a m-image subset can be repeated within T MCSTs. The probability that a m-image subset is perfectly repeated at least once is given by

$$\Pr(m;N,T) = 1 - \frac{M!}{(M-T)!M^T}, \quad M = \frac{N!}{m!(N-m)!} \tag{4.1}$$

where ! is the factorial operator. It is important to note that the image order within the m subset does not matter. The rest of this thesis will use values of $m = 10, 11, \ldots, 19$. Using Equation (4.1) with $N = 20$ images and $T = 100$ MCSTs, the probability that a $m = 10$ image subset repeats at least once is approximately 2.63%. The probability with $m = 15$ is approximately 27.4% and $m = 19$ is 100%. Note that using repeated image sets simply provides identical information and does not affect the calibration performance. In future work, N should be increased until the probability is less than 1% for all values of m.

4.2 Hardware Setup and Calibration Object

All imaging experiments were performed within Intel Corporation as part of a 3D reconstruction project using stereo triangulation. Note that the project itself guided the decision of which camera and optics to use, not this body of work.

The hardware setup used in all subsequent studies is detailed below. The experimental set up was originally for stereo 3D reconstruction applications, that require multiple cameras. Here, two Adimec OPAL 8000 area cameras were configured in a stereo setup. For the camera calibration performance analysis study, we only used one of the cameras. The cameras have an active sensor size of 3296×2472 pixels (~8 Megapixels) with a square pixel size of 5.5×5.5 μm. Both of these cameras use a Schneider Optics purchased Macro-Symmar 5.6/80mm lens. They have a very high modulation transfer function over the 400-700 nm visible spectrum, which entails low aberrations in the specified wavelength range. That should translate to low radial distortion parameters in our camera model. Extension

tubes were needed to fully use the entire camera sensor. Note that Schneider Optics provides the needed extension tubes that will work based upon your field of view (FoV) needs. The depth of field (DoF) was experimentally found to be around 1.5 mm at completely open aperture. No focus ring exists on the lens and no external focus mount was used. Therefore, in order to focus the lens, the working distance (distance from the lens center to the calibration object) needed to be changed. This proved to require other separate hardware outside the camera. Because such high accuracy is required, large vibrations cannot be tolerated so an extremely rigid system is required to hold the camera in place. Large, aluminum, breadboard, bench plates were purchased from Edmund Optics to attach everything in a customizable fashion. TECHSPEC series linear stages from Edmund Optics were purchased to allow the cameras to rotate and translate preciously moving up and down precisely (because of no focus mount). A single similar z-stage was used to move the calibration object up and down. Finally, a goniometer was loosely placed on top of the z-stage for precise angular adjustment of the calibration plate. This is demonstrated in Figure 4.1.

In this setup, a mixture of ambient florescent light as well as directional florescent light were used. The directional light is used to maximize contrast in the image without over-saturating the image due to specular reflections. The calibration target is highly reflective; for future application, a target of low reflectively is optimal as stray specular reflections are common with this target. Because of this, any direct light used must be observed closely. However, the lighting should be as constant as possible throughout the experiments to reduce experimental errors due to lighting. Note that systematic errors from illumination cannot be compensated for yet as no related direct study has been done prior. For the majority of the experiments, the direct florescent lighting was placed directly between the cameras pointing down along the Z-axis. The cameras were placed symmetrically about

Figure 4.1: Optical breadboard setup for calibration with the two 8 MP Adimec cameras with Schneider Optics lenses setup in a stereo application. The Edmunds purchased caliration plate setup in a non-coplanar pose upon a z-state and goniometer for precise movement.

the world Z-axis, so any tilts in the ZX plane may cause specular reflections. Any specular reflections that show up in the image dictate the operator to perform a small deviation of the illumination position just for that image so as to keep the experimental variation as low as possible.

The DoF of macro lenses like this are typically very small. In this setup, the DoF of the lenses proved to be extremely constraining at an experimentally verified 1.5 mm. If too much tilt exists from the camera or calibration target, part of the target's grid will be blurred causing unreliable feature point extraction. Because of this, the amount of tilt was constrained experimentally. In future applications,

higher tilts can be tolerated if the DoF is increased. An image taken exploiting the small DoF is shown in Figure 4.2. This is not only an introduction to what the grid looks like, but also notice the crisp circles near the bottom of the image and very blurry circles near the top.

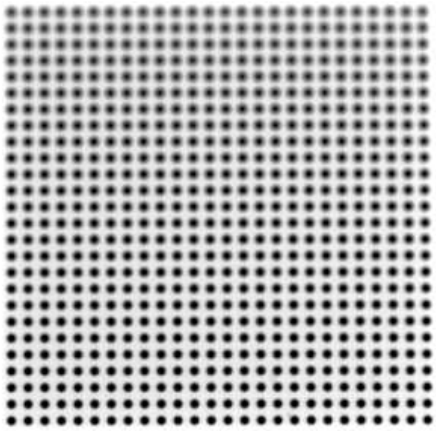

Figure 4.2: Edmunds Optics purchased calibration plate image with poor focusing near the top.

Lastly, the aperture of the lens will be left as open as possible simply to make sure consistency is held throughout the experimentation process. As of now, the literature has not explored compensating for the aperture within the pin-hole camera model. Anything other than an aperture all the way open or closed can prove to give arbitrary results as the f-stop markers that qualify aperture size are not precise. Also, the illumination was not able to be increased to give enough contrast for anything other than the lower end of the f-stop. Future experiments should require higher intensity illumination to exploit a closed aperture causing the DoF to increase.

When the application desires high accuracy results like this, the calibration piece must have very high manufactured accuracy. In fact, Heikkila [6] stated that ”the relative accuracy in the object space should be better than the accuracy aspired to in the image space.” The accuracy of the system is only as good as the worst

Circle Diameter	Center to Center Spacing
0.25 ± 0.0025 mm	0.5 ± 0.0025 mm

Table 4.1: Spacing and accuracies for the supplied calibration plate.

accuracy of any individual piece. If the calibration plate (the object in object space) accuracy is worse than the accuracy desired from the image, then, even with a perfect algorithm and an extremely high resolution camera, the system will surely be constrained by the calibration plate. As previously stated, the calibration piece can be either 2D or 3D such as the ones used by Heikkila [6]. In this application, a single plain calibration plate is used due to availability and size constraints. A non-custom Edmund Optics purchased plate was used called a multi-frequency grid distortion target. It contained printed circles with three different sizes and grid spacing. The inner-most grid was used with a grid circle count of 26×26. The spacing is given as well as the very important manufactured tolerances. These are the tightest tolerances available currently from Edmund Optics without custom ordering. As noted in Table 4.1, the tolerances are 2.5 μm. This means that a 3D reconstruction application such as this cannot expect a better accuracy than 2.5 μm. Lastly, the circles were manufactured to be painted black while the background surface is white. This was to provide for maximum contrast upon the image.

4.3 Intrinsic Parameter Estimation Accuracy

Using Monte Carlo simulation trials, as discussed in Section 4.1 we propose a new measure of camera calibration accuracy. The accuracy measure is the variance, or equivalently, standard deviation, of the estimated intrinsic parameters when multiple images are used for the camera model estimation in the calibration routine.

Investigation of Approach

Typically, when analyzing the accuracy of an estimated camera model, the actual value of the pixel error ε is used as the accuracy measure [1,6,13,15,24]. By actual value of pixel error of the calibration routine, we refer to the mean of n individual

feature point pixel errors over m images. The pixel error ε_{ij} of the ith calibration feature point using the jth image is defined as the average 2D Cartesian Euclidean distance between the projected world feature points (X,Y,Z) and the detected feature points (u,v). A pixel error near zero implies that the camera calibration process has correctly estimated the camera parameters so as to describe the projection of the world feature points into the image.

$$\varepsilon_{ij} = \left\| \begin{bmatrix} u_{ij} \\ v_{ij} \\ 1 \end{bmatrix} - \hat{F} \begin{bmatrix} X_{ij} \\ Y_{ij} \\ Z_{ij} \end{bmatrix} \right\|^2 \tag{4.2}$$

$$\varepsilon = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \varepsilon_{ij} \tag{4.3}$$

However, pixel error (PE) has one distinct disadvantage when analyzing its behavior with respect to the number of images used. Using only a small number of images (up to 5), it [1] was shown in that average PE decreases with increasing number of images. Specifically, the most improvement in PE was noted near the minimum number of images required for the specific projective technique, and continued to converge to a finite number with increasing number of images. Note, however, that when we used many more images (up to 19), we did not observe the same trend in the PE. Instead, as the number of images increased, the PE trended upwards, albeit slowly, as demonstrated in Figure 4.3.

Instead of PE, we propose a different measure of camera calibration accuracy. The measure is related to the accuracy of estimation of the calibration model parameters, and it is expected to change with the number of images, or equivalently, the number of Monte Carlo simulation trials (MCSTs) used in the calibration. As each MCST is used with different input information, an estimated parameter value varies between trials. The variation is due to system noise and variances in cal-

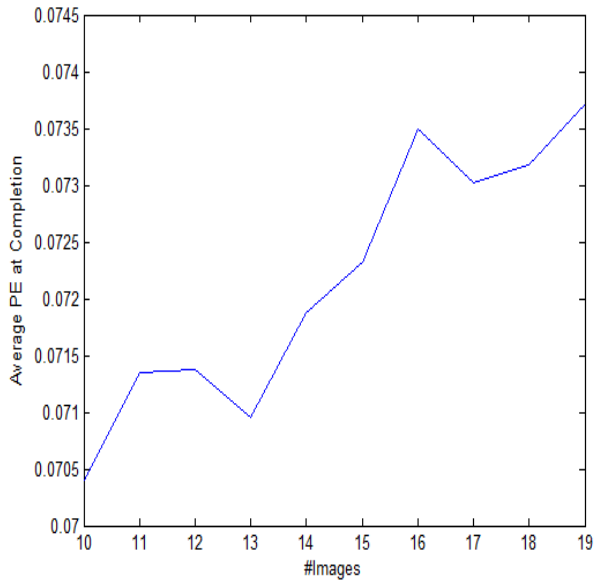

Figure 4.3: Average pixel error as a function of the number of images used as input into the calibration routine

ibration object pose from image to image. Thus, the new metric is the standard deviation of the estimated parameter after T MCSTs.

Since the camera does not change its position while the calibration plate does and the extrinsic parameters are described as the rigid body motion from the calibration plate to the camera, the extrinsic parameters are unique for each image used in a MCST. Between trials, a new subset of images would likely be used, resulting in new, unique extrinsic parameters that, when clustered together with other image sets, are expected to be quite different in value. As a result, it does not make sense to use extrinsic parameters as a performance metric as it would be inconclusive within a MCSS. Instead, certain intrinsic parameter estimates are used to form the metric. We chose to use the focal length f, as it has a direct linear dependence to the camera model. Note that this dependency was also explored by T. Rahman [25] as it related to distortion parameters over a wide range of possible distortions. The variance of the focal length estimated $\hat{f}_i$ is obtained at the ith MSCT and given by

$$\sigma_f^2 = \frac{1}{T}\sum_i^T \hat{f}_i^2 - \left[\frac{1}{T}\sum_i^T \hat{f}_i\right]^2 \tag{4.4}$$

where T is the total number of MCSTs. Using this metric, the lower the variance of the estimated focal length, the more accurate the estimate and thus stronger belief that the focal length has been estimated correctly.

Experimental Results

For our experimental work in analyzing calibration accuracy, we used the same single camera model. Although prior studies using different accuracy metrics did not see any improvement by adding higher order distortion terms to the camera model [6], we wanted to verify this using the new estimate variance metric as well as figure out the best distortion model for our camera model. This experimentation study also demonstrates our method of characterizing parameter estimation accuracy using Monte Carlo simulations.

The distortion camera model given by Equation (2.7) and Equation (2.8) expresses the camera model with four distortion parameters: a first and second order radial k_1 and k_2 and a first and second order tangential p_1 and p_2 distortion parameters. These are the distortion parameters [1, 6] most often used with the common camera model using Taylor series distortion coefficients (see Equation (2.6)). Other distortion models without Taylor series coefficients, such as the one in [26], will not be considered.

We consider four camera models. The first model is in terms of the four distortion parameters, the two highest orders for both radial and tangential and it is given by

$$\begin{bmatrix} I_R + I_T \\ J_R + J_T \end{bmatrix} = \begin{bmatrix} I_D(k_1 r^2 + k_2 r^4) + 2p_1 I_D J_D + p_2(r^2 + 2I_D^2) \\ J_D(k_1 r^2 + k_2 r^4) + p_1(r^2 + 2J_D^2) + 2p_2 I_D J_D) \end{bmatrix} \tag{4.5}$$

The second model only depends on the first order radial distortion parameter and the two tangential distortion parameters.

$$\begin{bmatrix} I_R + I_T \\ J_R + J_T \end{bmatrix} = \begin{bmatrix} I_D(k_1 r^2 + 0) + 2p_1 I_D J_D + p_2(r^2 + 2I_D^2) \\ J_D(k_1 r^2 + 0) + p_1(r^2 + 2J_D^2) + 2p_2 I_D J_D) \end{bmatrix} \tag{4.6}$$

The third model depends only on the first order radial and tangential distortion parameters.

$$\begin{bmatrix} I_R + I_T \\ J_R + J_T \end{bmatrix} = \begin{bmatrix} I_D(k_1 r^2 + 0) + 2p_1 I_D J_D + 0) \\ J_D(k_1 r^2 + 0) + p_1(r^2 + 2J_D^2) + 0) \end{bmatrix} \tag{4.7}$$

And finally, the fourth camera model is an extension of the model in Equation (4.5), but with an added third order radial distortion parameter.

$$\begin{bmatrix} I_R + I_T \\ J_R + J_T \end{bmatrix} = \begin{bmatrix} I_D(k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 I_D J_D + p_2(r^2 + 2I_D^2) \\ J_D(k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2J_D^2) + 2p_2 I_D J_D) \end{bmatrix} \tag{4.8}$$

Using the fourth model, we want to explore whether higher order terms will improve camera model accuracy. Using a single MCSS for each camera model, we considered $T = 100$ MCSTs for calibration using $N = 20$ and $k = 10$. For each trial within a set, the focal length f was recorded to obtain the standard deviation σ_f of the MCSS per camera model.

For consistency in comparing results, for each camera model used, σ_f was normalized by a global average focal length μ_f so as to have a globally normalized standard deviation from the mean. This relative standard deviation (RSTD) metric is given by $\overline{\sigma_f} = \sigma_f / \mu_f$. We used the value μ_f = 21662 pixels throughout this work for this setup as it was found to be the mean estimated focal length of all distortion camera models. For a different setup with a different lens and focal

Camera Model (CM)	First Order Radial	Second Order Radial	First Order Tangential	Second Order Tangential	Third Order Radial	σ_f (pixel)	$\hat{f}$ (%)
CM1	✓	✓	✓	✓	–	527.75	2.44
CM2	✓	–	✓	✓	–	620.75	2.87
CM3	✓	–	✓	–	–	555.45	2.56
CM4	✓	✓	✓	✓	✓	843.50	3.89

Table 4.2: Standard deviation and RSTD metrics for the four different distortion camera models using 100 MCSTs with $k = 10$ calibration input images.

length, it would be difficult to compare results without normalization as the mean is expected to shift. Without normalization, the standard deviation of a smaller focal length would be considerably worse off when compared to a much larger focal length of similiar standard deviation. The results of the RSTD for the four different models using $k = 10$ images are provided in Table 4.2. Note that we use the notation CM1, CM2, CM3, and CM4 for the first, second, third, and fourth camera models, respectively. For visual reference, each distortion parameter is shown as a column and each camera model is shown as a row. The table also further emphasizes which distortion parameters are used by each model. From Table 4.2, the first camera model (CM1) has the lowest estimated focal length RSTD. As a result, we chose camera model 1 in Equation (4.5) as the camera model for the rest of this thesis.

In our first experimental study, we used a constant number of images (k=10) to keep the experimental variables fixed other than the distortion camera model parameters. For the next study, we vary the number of images using the first camera model. There is always discussion of how many images are needed to produce a perfectly determined or overdetermined system of equations. With our calibration plate, we used 676 feature points producing 676 equations per image, so we have a highly overdetermined system of equations. Thus, we want to study how many images would suffice for an acceptable level of estimation accuracy.

Using $T = 100$ MCSTs as before, we varied the number of images from $k = 10$ to $k = 19$ images. For each trial within a set, the focal length f was recorded and the standard deviation as a function of the number of images, k, is shown in Fig. 4.4. This metric can be equated to a real world, by multiplying it by the pixel size S_I or S_J (they are the same in our square pixel camera) of 6.5 microns per pixel. This results in the actual focal length of the lens in our pin-hole camera model. For comparison consistency, the RSTD is shown in Fig 4.5.

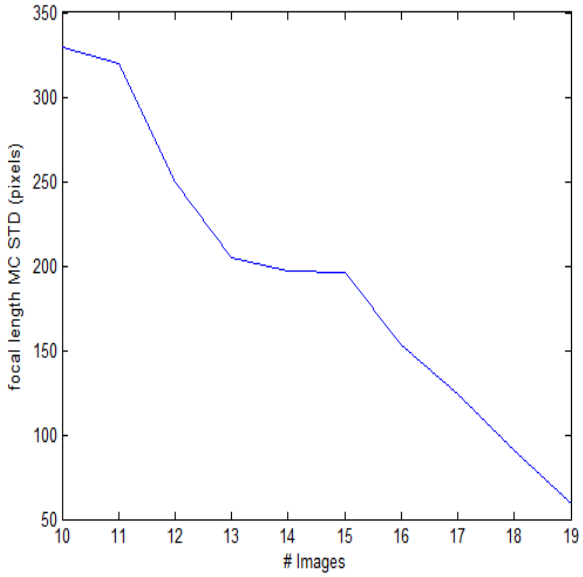

Figure 4.4: Focal Length standard deviation (STD) using 100 MCSTs for different number of images into the calibration routine

Discussion

Table 4.2 shows that, in fact, the model with the two highest order, both radial and tangential distortion parameters, does indeed produce the most statistically consistent terms. It thus indicates the best camera model to produce an accurate estimation of the focal length, which is the most important intrinsic parameter. This result using estimated camera parameters across a Monte Carlo simulation is inline with previous results using pixel error as their metric. If one was to desire a Taylor series representation of distortion parameters as their distortion camera model,

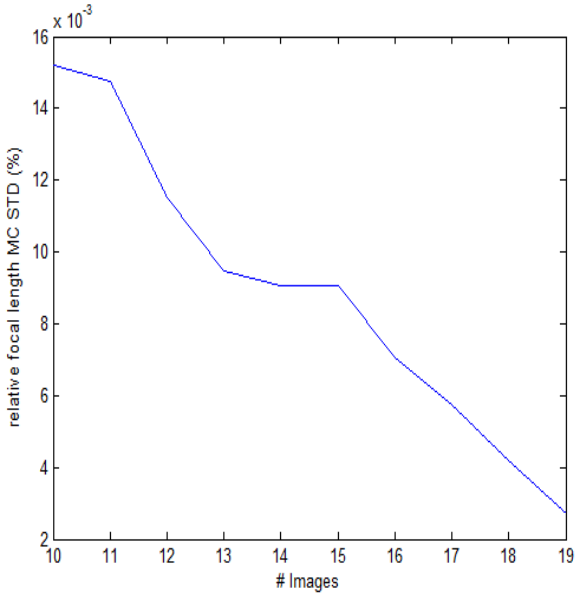

Figure 4.5: Focal length relative standard deviation (RSTD) using 100 MCSTs for different number of images into the calibration routine

the two highest order terms for both radial and tangential would produce the most consistent intrinsic parameter estimation. The first model in Equation (4.5) is the consistently chosen distortion camera model.

From the results in Fig 4.5, the standard deviation of the focal length is approximately 0.015% the value of its mean when $k = 10$ images are used. When $k = 19$ images are used, the standard deviation reduced by an order of magnitude to 0.0015% of its mean. It is clear that adding more images to the calibration routine produces a fairly linear improvement to focal length estimation. We expect that this trend will continue as the number of images increases more than 19. Thus, the more images added to the estimation, the higher the estimated calibration model accuracy.

Note that in order to explore the type of calibration accuracy required for different applications, a similiar approach can be followed. The resulting estimated values will change depending on the calibration object tolerances, number of feature points on an object, and number of images. Also note that this procedure can

be extended to other intrinsic parameters. However, for our application of stereo 3D reconstruction, the focal length was the most relevant intrinsic parameter.

4.4 Reduction in Algorithm Computational Time

Although we have demonstrated in Section 4.3 that the more images used as input to the calibration parameter estimation, the higher the estimated calibration model performance, adding a large number of images can increase computational time. Here, we investigate how the maximum likelihood estimator (MLE) thresholds can be improved for our application to significantly reduce computational time while maintaining high accuracy criteria.

Investigation of Approach

As discussed in Section 3.5, Bouguet's calibration toolbox utilizes a non-linear optimization technique (Levenberg-Marquardt's least squares MLE) to minimize pixel error over the camera model functions in Equation (3.2). Because this is an iterative solution, it requires a stopping criteria. The two stopping criteria used in this toolbox are a lower threshold T_1 on the percentage change δ of the focal length f and principle point (I_0,J_0) compared to the previous iteration's values and an upper threshold T_2 for the number of maximum iterations allowed. The former can be described in more detail as the norm percentage change for each iteration given by

$$\delta = \frac{\sqrt{\left(f^{(new)} - f^{(old)}\right)^2 + \left(I_0^{(new)} - I_0^{(old)}\right)^2 + \left(J_0^{(new)} - J_0^{(old)}\right)^2}}{\sqrt{{f^{(new)}}^2 + {I_0^{(new)}}^2 + {J_0^{(new)}}^2}} \tag{4.9}$$

For example, $f^{(new)}$ refers to the current iteration's focal length estimate while $f^{(old)}$ refers to the previous iteration's estimate. For each iteration, δ is computed and compared with a user defined threshold T_1. The second threshold of maximum iterations per calibration T_2 is also user defined. When the minimization function reaches either of the thresholds, the function ceases to continue and is said to be complete.

Within each iteration, the camera model parameters are approaching their optimal value. They can be said to be approaching a local pixel error minima that is hopefully the global minima of the parameter search space. That, however, is constrained by the abilities of the implemented MLE approach, the Levenberg-Marquardt algorithm (LMA).

Before moving forward, an important note must be said about computational time. All of the studies in this work were done in MATLAB. In order to time how long a section of code takes to run, MATLAB offers two ways of doing so - the *tic toc* approach and a built in Profiler integral to optimizing a piece of code for run time purposes. However, if the computer is running many computationally intensive tasks in the background, the run time will be longer than normal. The absolute best method of correctly identifiying how much computational time spent on a task would be to track the clock cycles. However, MATLAB offers no such built in functionality. Moving forward, to mitigate this, all simulations were run on the same computer with all unneeded processes removed. The *tic toc* method was used as it is extremely simple to implement and has an experimentally measured 99.9957% average accuracy per calibration routine when compared to the MATLAB profiler. Because they are so similar, both approaches are worthy of implementation.

Experimental Results

In this study, we lay a baseline case of Bouguet's implementation to investigate its statistical properties concerning its intrinsic parameter estimation and computational run time. This baseline case is called the "Default Method" (DM). $T = 100$ MCSTs were ran for each number of images k. Within each trial, the focal length, pixel error (PE), and time per iteration were recorded. Because this is a baseline case of how the camera calibration routine runs naturally as designed, the thresholds were set at their default values - $T_1 = 1 \times 10^{-9}$ and $T_2 = 30$. We have included pixel error only because it has been used so extensively in prior works and can be

shown that the same conclusion about computational time can be said for parameter estimation STD as well as pixel error.

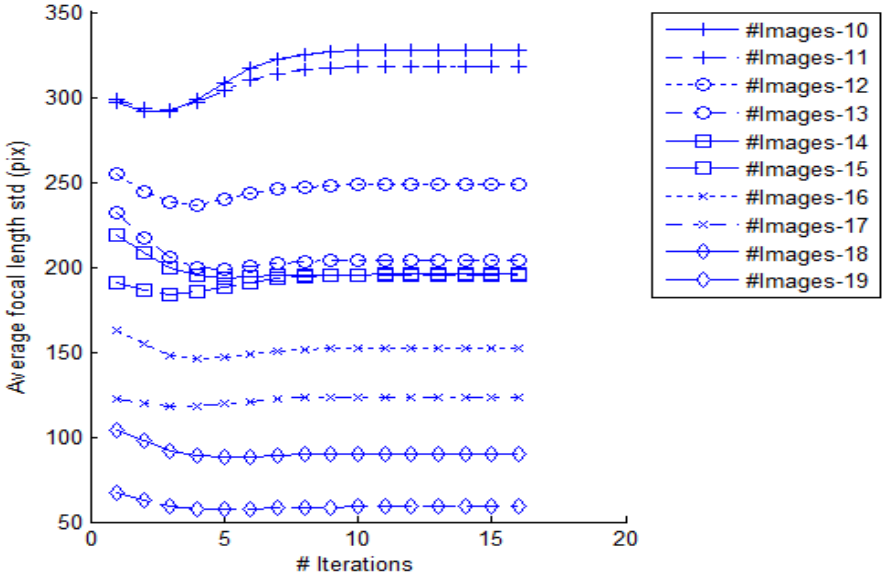

Figure 4.6: The average focal length STD across every iteration of a MCSS versus number of images input into the calibration routine

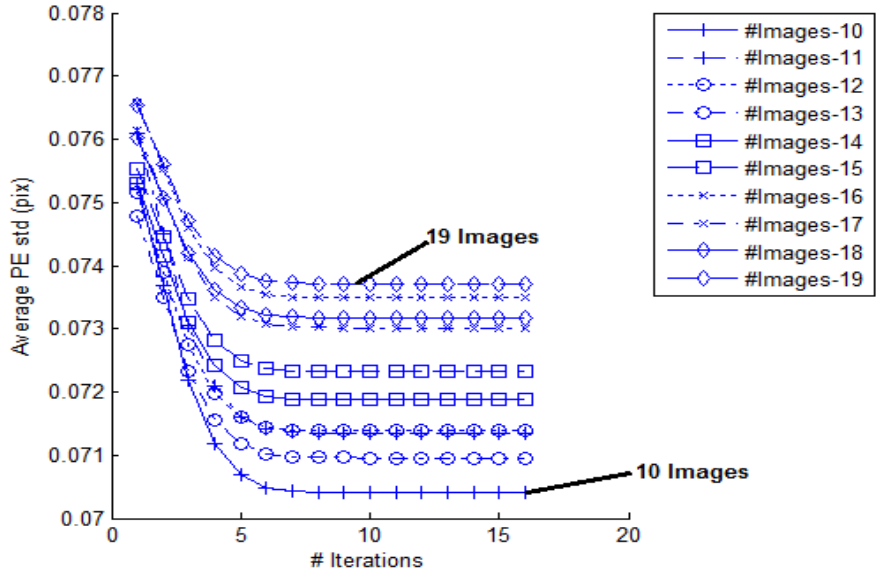

Figure 4.7: The average pixel error across every iteration of a MCSS versus number of images input into the calibration routine

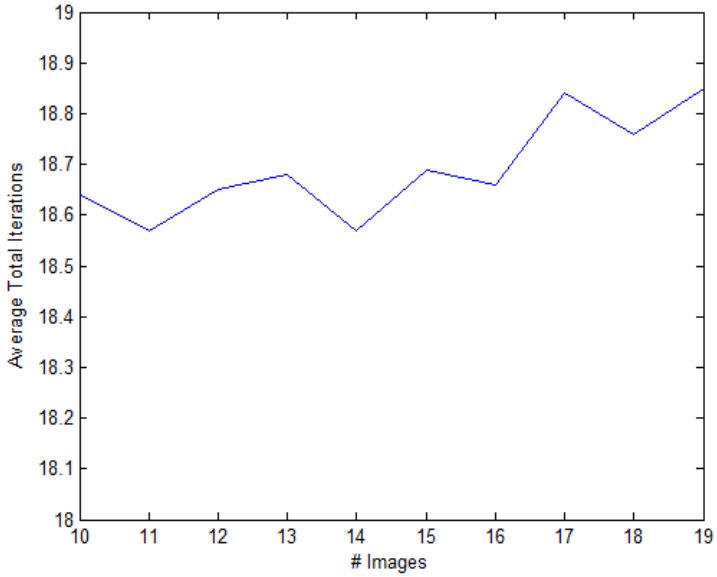

Figure 4.8: The average total iterations needed per Monte Carlo simulation trial (MCST) within a Monte Carlo simulation set (MCSS) versus number of images input into the calibration routine

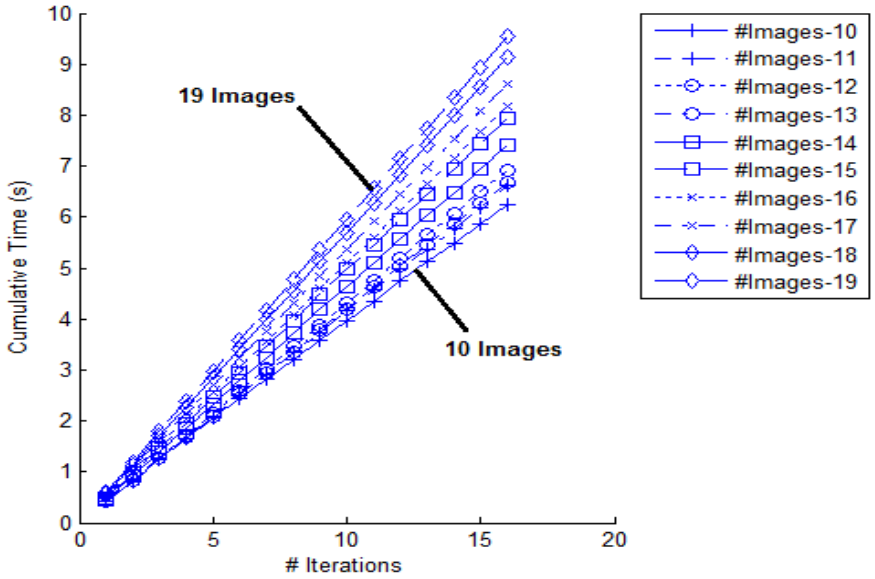

Figure 4.9: The average cumulative time over iterations per Monte Carlo simulation trial (MCST) within a Monte Carlo simulation set (MCSS) versus number of images input into the calibration routine

Fig. 4.6 shows similiar results from Fig. 4.4. However, Fig. 4.6 expands to include the results per iteration. The same could be said for Fig. 4.7 as it is simply an expansion of Fig. 4.3 to include results per iteration. First, we clearly see that the results once again show that the more images are fed to the calibration routine, the lower the focal length STD per MCSS will be. The opposite trend can be said for PE as it rises with increasing number of images. Neither of these conclusions are new, however.

It is important to note that it is very difficult to compare setup to setup using pixel error. The amount of pixel disparity from a projected point to a detected point upon the image from setup to setup is dependent on pixel size, magnification of camera, inherent inaccuracies of the calibration piece, and the intrinsic and extrinsic parameters of the camera (not estimated parameters of the camera model). If one was to use a camera with a larger pixel size with all things being equal, a pixel error of, for example, 0.1 pixels would mean more than from a camera with a smaller pixel size. A camera that is closer to the object will have a higher magnification. This means that the object is projected onto a larger portion of the image sensor. Therefore, higher pixel errors will naturally arise due to the inaccuracies of the calibration plate.

Next, it is noted that the typical cutoff for the optimizer is at 18-19 iterations, shown clearly in Fig. 4.8. This is the trial to trial average within a MCSS of the total iterations needed to finish optimizing under the baseline rules of default thresholds. Because every number of images produced an average between 18 and 19 iterations, we rounded down and called 18 iterations finished. Using 19 iterations as the global average would falsify data as the average iteration count per number of images never even reached that value once.

However, the most important conclusion to draw from both Fig. 4.6 and Fig. 4.7 is that the STD and PE respectively clearly show a converging trend for every plot line. It can be seen that both the STD and PE, due to the estimation of the camera model parameters, converges to their best accuracy extremely quickly at around 8-12 iterations.

There is clearly an opportunity to save time by constraining the maximum number of iterations allowed, T_2. If we take the average cumulative time to reach 10 iterations t_{10}^k per k images and the same average cumulative time to finish optimizing t_f^k per k images under the baseline default thresholds, we can show the average time saved per k images in percentage is given by

$$\hat{t}^k = \frac{t_f^k - t_{10}^k}{t_f^k} \tag{4.10}$$

This is essentially the local normalized time saved. This idea of reducing T_2 is referred to as the "Reduced Default Method" (RDM). This can be shown visually for k=10 in Fig. 4.10. It is incredibely important to note here that though we used a global average to normalize focal length, a local average was used here for time. This was done because focal length can easily change from application to application while time is a human concept that will not change from application to application. Time is already normalized for us based on its definition while focal length is not. Thus, we applied a normalizing agent to move forward.

However, this train of thinking can quickly lead us short of the requirement for high-accuracy 3D reconstruction. Fig. 4.11 shows how much accuracy we are losing by stopping the optimizer short, by taking the STD for the k images at 10 iterations and at final completion and taking their percentage difference for all k values. The methodolgy is exactly the same as with the computational time saved.

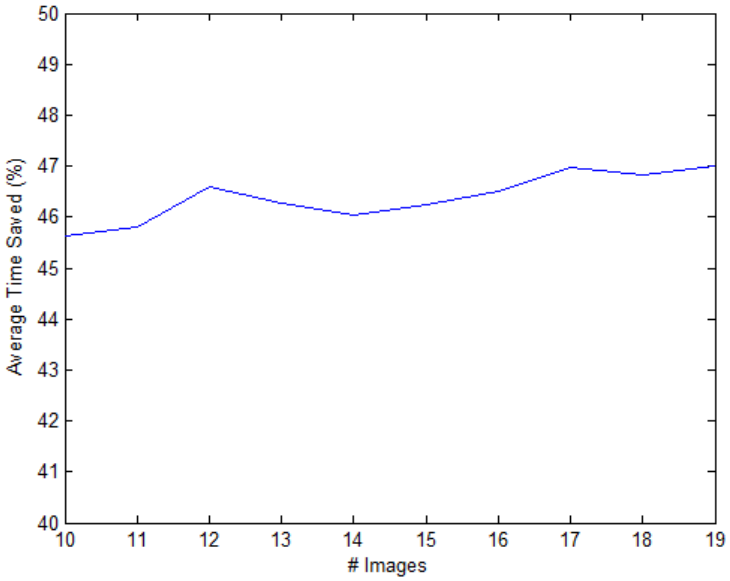

Figure 4.10: RDM - By shortening maximum iterations to 10 (approximate iterations needed for convergence), the mean of the total time saved per Monte Carlo simulation trial (MCST) within a Monte Carlo simulation set (MCSS) versus number of images input into the calibration routine

All of the prior work of this section can be concluded and summarized into a single table shown in Table (4.3).

	DM			RDM			DM to RDM Difference	
k	Iter	STD (pix)	Cumulative Time (s)	Iter	STD (pix)	Cumulative Time (s)	Accuracy Lost (in STD) ($\times 10^{-3}$ %)	Nominal Time Saved (%)
10	18	329.24	7.27	10	328.94	3.95	1.37	45.7
11	18	319.47	7.67	10	319.23	4.16	1.12	45.8
12	18	249.71	8.08	10	249.55	4.31	0.73	46.6
13	18	205.16	7.82	10	205.01	4.20	0.69	46.3
14	18	196.63	8.59	10	196.58	4.64	0.23	46.0
15	18	196.23	9.27	10	196.11	4.98	0.58	46.2
16	18	152.94	9.55	10	152.82	5.11	0.58	46.5
17	18	124.07	10.13	10	123.98	5.37	0.43	47.0
18	18	90.36	10.70	10	90.31	5.69	0.25	46.8
19	18	59.194	11.24	10	59.14	5.96	0.22	47.0

Table 4.3: The summary of the time savings and accuracy lost of improving upon the baseline case by reducing the maximum iterations allowed to $T_2 = 10$.

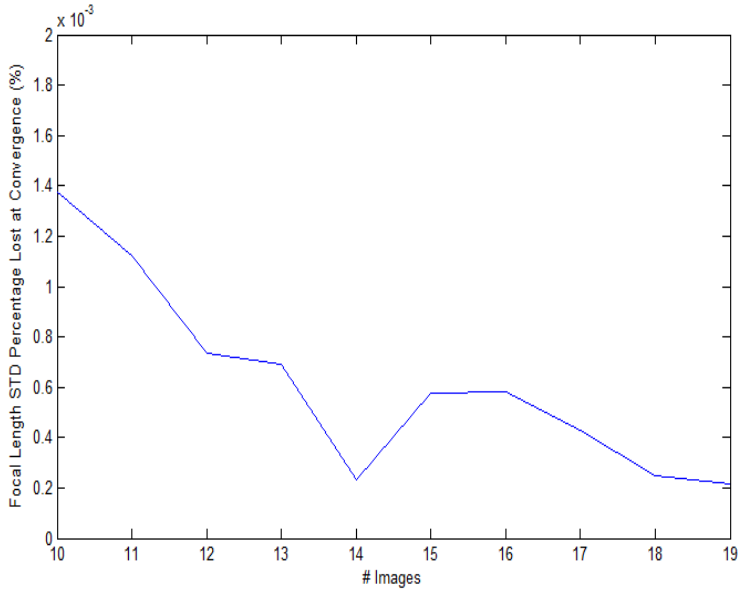

Figure 4.11: By shortening the number of maximum iterations to 10 iterations (iterations needed for convergence), the average percentage lose of the STD per Monte Carlo simulation trial (MCST) within a Monte Carlo simulations set (MCSS) versus number of images input into the calibration routine

Each row of Table 4.3 signifies the results for each number of images k. The difference between DM and RDM is the culmination of Fig. 4.11 and 4.10.

Discussion

Table 4.3 shows that, in fact, we can improve upon the default values present in Bouguet's classic toolbox. Fig. 4.4 and 4.7 both show the typical accuracy metric and our new Monte Carlo parameter estimation accuracy metric as converging to an optimal value much earlier than the maximum iterations T_2 is set to. For example on Fig. 4.7, for 10 images from iteration 12 to iteration 13, on average, only 1.46×10^{-8} pixels of error was corrected for by better camera model parameter estimation. That is approximately only $2.08 \times 10^{-5}\%$ improvement from its current result at 12 iteration. We can conclude that the optimizer is doing very little from iteration to iteration to optimize our camera model after the point of convergence. Because of this, we are allowed to further reduce T_2 to 10 iterations, for example. From all k values, we see an average time savings of about 46.4%. Not only do we see

a significant reduction in computational time, we see a minimal loss in accuracy. From all k values, we see an average accuracy loss of about 6.2×10^{-4} % in terms of the STD.

Table 4.3 also shows a fairly constant time savings no matter how many images are used. This is due to two reasons. First, as already stated, we are using a local normalizing agent from Equation (4.10) that normalizes the cumulative time difference between DM and RDM by the DM time for every k value. The cumulative time at any iteration will indeed be larger, for example at $k = 19$ images, when compared to the cumulative time at the same iteration for a lower k value. Instead, we have chosen a global threshold for maximum iterations and all images will stop at the same iteration count because of this. Coupling that the time per iteration is fairly linear according to Fig. 4.9, and since Equation (4.10) states that the percentage difference is always between the new RDM T_2 value and DM's average total iterations, we should expect a fairly constant time savings irrespective of the value k.

However, we based our decision of choosing $T_2 = 10$ as an early cutoff point based on visual evidence. Moving forward, we constructed a performance trade off analysis. As discussed, we saved computational time by reducing the maximum iterations to 10, however, we lose a certain amount of accuracy. This is a trade-off and changing T_2 will trade benefits from computational time and accuracy, as demonstrated in Fig. 4.12.

This shows us that if we pick $T_2 = 1$, we would save nearly 100% time; however, we would lose almost 0.2% accuracy in terms of STD. That is certainly a very small accuracy loss even for allowing the optimizer only one iteration to work. This is only because of the chosen maximum likelihood estimator and the high density feature point calibration object. This may not be possible for other applications. Nonetheless, for optimal performance trade off, it is clear that choos-

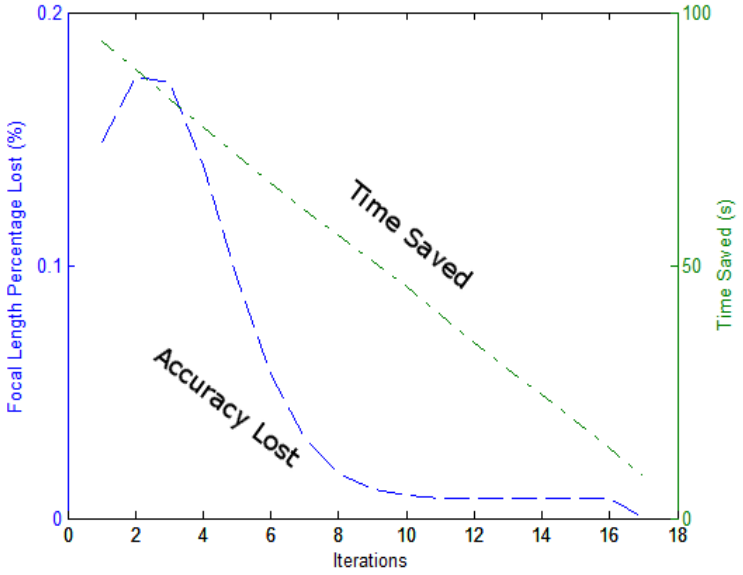

Figure 4.12: Performance trade off - By shortening maximum number of iterations T_2 to a certain value, the average percentage lose of STD and computational time per Monte Carlo simulation trial (MCST) within a Monte Carlo simulation set (MCSS) will change ($k = 10$ images)

ing T_2 between 8 and 10 iterations would maximize computational time saved and minimize accuracy loss.

Chapter 5

CONCLUSIONS AND FUTURE WORK

5.1 Summary

In this thesis, we proposed a new way of characterizing accuracy of a camera calibration method as well as improving upon a well known method for all applications while using Monte Carlo simulations. The standard deviation of an estimated intrinsic camera model parameter within a Monte Carlo set was shown to improve with increasing number of images as well as with increasing iterations within the iterative maximum likelihood estimation (MLE) optimizer. Using this method of experimentation, we have also shown that the MLE optimizer does not require all the iterations it was provided with as thresholds governing its operation. In fact, we reduced the number of iterations to account for over 45% reduction in computational time while only losing 0.001% of the new relative focal length standard deviation within a Monte Carlo set accuracy metric.

5.2 Future Work

Chapter 4 used exclusively either a constant number of images k or a range varying from ten to nineteen. This was able to sufficiently show the required trends. Further analysis should at least include the minimum number of images of two in order to fully understand the limitations. Expanding the maximum number of images to a value in the range of fifety or one hundred would more accurately generalize many of the trends seen here to show whether they are indeed linear as they appear or if they exhibit a higher order response.

It was shown that the Monte Carlo simulation approach can be used to test the random sampling of images in order to show accuracy trending for various parameters within Bouguet's toolbox implementing Zhang's MLE. However, there exists many varied approaches that this Monte Carlo approach could be applied to, such as the approach by Heikkila, Tsai, Weng, and many others [6,13,15,17,27,28].

Lastly, it was shown that focal length estimation accuracy within a Monte Carlo simulation set was reduced with increasing number of images. Focal length was chosen as the key estimated parameter because of its key dependence in obtaining successful 3D reconstruction. In order to explore more possibilities that the Monte Carlo method can be used, further analysis with other intrinsic camera parameters could be implemented including principle point and distortion parameters, especially using a wide-angle lens.

REFERENCES

[1] Z.Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," *Seventh International Conference on Computer Vision*, vol. 1, pp. 666–673, September 1999.

[2] J. Bouguet, "3D photography using shadows in dual-space geometry," *International Journal of Computer Vision*, vol. 35, no. 2, pp. 129–149, 1999.

[3] H.-N. Yen and D.-M. Tsai, "A fast full-field 3D measurement system for BGA coplanarity inspection," *The International Journal of Advanced Manufacturing Technology*, vol. 24, no. 1-2, pp. 132–139, January 2004.

[4] A. Liu, "Characterization of fine-pitch solder bump joint and package warpage for low k high-pin-count flip-chip BGA through shadow moire and micro moire techniques," *Electronic Components and Technology Conference*, pp. 431–440, June 2011.

[5] J.Bouguet, "http://www.vision.caltech.edu/bouguetj/," 2007.

[6] J. Heikkila, "Geometric camera calibration using circular control points," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1066–1077, October 2000.

[7] W. Boyle and G. Smith, "Charge coupled semiconductor devices," *Bell System Technical Journal*, vol. 49, no. 4, pp. 587–593, April 1970.

[8] P. Noble, "Self-scanned silicon image detector arrays," *IEEE Transactions on Electron Devices*, vol. 15, no. 4, pp. 202–209, April 1968.

[9] A. Dickinson, "Standard CMOS active pixel image sensors for multimedia applications," *Conference on Advanced Research in VLSI*, pp. 214–224, March 1995.

[10] D. Brown, "Close-range camera calibration," *Photogrammetric Engineering*, vol. 37, no. 8, pp. 855–866, 1971.

[11] A. Conrady, "Decentering lens systems," *Monthly notices of the Royal Astronomical Society*, vol. 79, pp. 384–390, 1919.

[12] D. Brown, "Decentering distortion of lenses," *Photometric Engineering*, vol. 32, no. 3, pp. 444–462, 1966.

[13] R.Y.Tsai, "A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 1987.

[14] Y.L.Abdel-Aziz and H. Karara, "Direct linear transformation into object space coordinates in close-range photogrammetry," *Proc. of the Symposium on Close-Range Photogrammetry*, pp. 1–18, 1971.

[15] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1106–1112, June 1997.

[16] F.L.Corten, "European point of view on standardising the methods of testing photogrammetric cameras," *Photogrammetric Engineering*, vol. 27, no. 3, pp. 401–405, 1951.

[17] L. Wu, X. Cao, and H. Foroosh, "Camera calibration and geo-location estimation from two shadow trajectories," *Computer Vision and Pattern Recognition*, pp. 585–590, August 2010.

[18] N. Jacobs, N. Roman, and R. Pless, "Toward fully automatic geo-location and geo-orientation of static outdoor cameras," *Applications of Computer Vision, 2008*, pp. 1–6, January 2008.

[19] M. Dong and R. Chung, "Height inspection of wafer bumps without explicit 3-D reconstruction," *IEEE Transactions on Electronics Packaging Manufacturing*, vol. 33, no. 2, pp. 112–121, April 2010.

[20] J. More, "The Levenberg-Marquardt algorithm: Implementation and theory," *Lecture Notes in Mathematics*, vol. 630, pp. 105–116, 1978.

[21] H. Ding, "A novel occlusion planning method for unknown 3D objects automatic reconstruction," *Conference on Artificial Intelligence and Computational Intelligence*, pp. 463–467, October 2010.

[22] W. Fang, "Automatic view planning for 3D reconstruction and occlusion handling based on the integration of active and passive vision," *Symposium on Industrial Electronics*, pp. 1116–1121, May 2012.

[23] J. Bouguet, "Visual methods for three-dimensional modeling," Ph.D. dissertation, California Institute of Technology, 1999.

[24] J. Heikkila and O. Silven, "Calibration procedure for short focal length off-the-shelf CCD cameras," *IEEE Proceedings of the 13th Internation Conference on Pattern Recognition*, vol. 1, pp. 166–170, August 1996.

[25] T. Rahman and N. Krouglicof, "An efficient camera calibration technique offering robustness and accuracy over a wide range of lens distortion," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 626–637, February 2012.

[26] W. Zhu, C. Diao, and J. Huang, "Calibration of radial distortion via QR factorization," *IEEE International Conference on Progress in Informatics and Computing (PIC)*, pp. 728–732, December 2010.

[27] J. Weng, P. Cohen, and M. Herniou, "Camera calibration with distortion models and accuracy evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 10, pp. 965–980, October 1992.

[28] D. Nister, "An efficient solution to the five-point relative pose problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 75–770, June 2004.