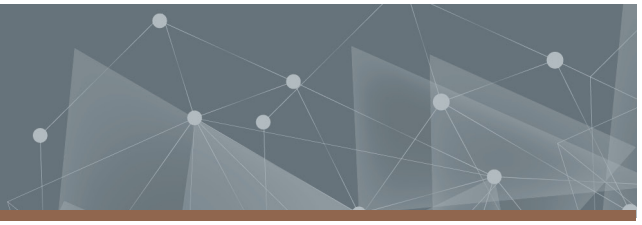
CHALMERS
UNIVERSITY OF TECHNOLOGY



Optimization of estimating intrinsic parameters of a camera using a robot arm

Master's thesis in Systems, control and mechatronics

SHILPA SUDHIR

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2025
www.chalmers.se

Master's thesis 2025

Optimization of estimating intrinsic parameters of a camera using a robot arm

SHILPA SUDHIR

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2025

Optimization of estimating intrinsic parameters of a camera using a robot arm
SHILPA SUDHIR

Supervisors: Anders Ericsson and Jörgen Thaung, Smart Eye AB
Examiner: Yasemin Bekiroglu, Department of Electrical Engineering, Chalmers University of Technology

Master's Thesis 2025
Department of Electrical engineering
Division of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2025

Optimization of estimating intrinsic parameters of a camera using a robot arm
SHILPA SUDHIR
Department of Electrical Engineering
Chalmers University of Technology

Abstract

Eye-tracking technology is applied across industries such as automotive, aerospace, and human-computer interaction, where precise gaze estimation relies on the accurate calibration of eye-tracking cameras. In this work, a robot arm is used to perform the intrinsic calibration of eye tracking cameras. The motivation behind using a robotic arm for the intrinsic calibration of an eye-tracking camera instead of a manual procedure stems from several key advantages related to precision, repeatability, and efficiency. Control over the camera poses in front of the checkerboard leads to a more accurate and efficient calibration process. This eliminates the challenges and inconsistencies associated with manual camera movements and opens the door for automating and scaling the calibration process with minimal human involvement. The robotic arm ensures a level of consistency and reliability that would be hard to achieve manually, making it a compelling choice for eye-tracking camera calibration, especially in high-precision applications.

Keywords: camera calibration, camera matrix, distortions, camera intrinsics, extrinsics, pinhole camera model, optimization, robot arm.

Acknowledgements

I would like to express my sincere gratitude to my supervisors Anders Eriksson and Jörgen Thaung, for their generous support, guidance and encouragement throughout this project. I am also grateful to Torsten Wilhelm for his ideas and encouragement during challenging situations.
My heartfelt thanks go to my academic supervisor and examiner, Yasemin Bekiroglu, for her thoughtful feedback and guidance in academic writing.
I am grateful to Niklas Baerveldt for his time and care in providing peer reviews.
I also thank Mattias Torstensson for his helpful advice and support with Blender, and my office colleagues for making my time at Smart Eye AB enjoyable. Finally, I am thankful to Smart Eye AB for providing the opportunity and environment to carry out this project.

Shilpa Sudhir, Gothenburg, November 2025

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AIS	Smart Eye's driver monitoring cameras
A-FOV	Angular Field Of View
BCS	Blender Coordinate System
DOF	Degree Of Freedom
DMS	Driver Monitoring System
GUI	Graphical User Interface
HDR	High Dynamic Range
LED	Light Emitting Diode
LUT	Look Up Table
NIR	Near Infrared
ROS	Robot Operating System
RCS	Robot Coordinate System
RMS	Root Mean Square
WCS	World Coordinate System

Nomenclature

Intrinsic and lens distortion parameters

$\mathbf{K}$	Intrinsic matrix
f_x, f_y	Focal length in both axes
c_x, c_y	Principal point
γ	Skew coefficient
k_1, k_2, k_3	Radial distortion coefficients
p_1, p_2	Tangential distortion coefficients

Frame conventions

P_c	3D point expressed in the camera frame
P_w	3D point expressed in the world frame
$\mathbf{R}$	3D Rotation matrix
$\mathbf{t}$	Translation vector
x_i, y_i	Ideal image coordinates
x_d, y_d	Actual image coordinates
u, v	Pixel coordinates

Other symbols

C	Detection coverage
N	Number of images
W	Image width
H	Image height
N_{covered}	Number of cells that contain detections

N_{total}	Total number of cells in the grid
$\mathbf{M}$	A point on the model plane
$\mathbf{m}$	Image corresponding to $\mathbf{M}$
$\mathbf{H}$	Homography
SE	Polynomial distortion model with two radial distortion coefficients and fixed principal point in Smart Eye's intrinsic calibrator
$SEPP$	Polynomial distortion model with two radial distortion coefficients and free principal point in Smart Eye's intrinsic calibrator
$TSFP$	Polynomial distortion model with two radial distortion coefficients and fixed focal length and principal point in Smart Eye's intrinsic calibrator

Contents

List of Figures

List of Tables

1

Introduction

1.1 Background

This project is done in collaboration with Smart Eye AB. Smart Eye's primary products are head, eye and gaze tracking systems for building advanced Driver Monitoring and Interior Sensing systems in the automotive industry, investigative tools in neuroscience and psychology, and research, development and education in the aerospace industry.

Camera calibration is a fundamental process in computer vision, necessary for determining the *intrinsic* and *extrinsic* parameters that are vital for accurate 3D reconstruction and image rectification [1]. Intrinsic parameters, such as the focal length and principal point, describe the internal characteristics of the camera, while extrinsic parameters describe its pose relative to the world coordinate system. In addition to camera parameters, the distortion coefficients that represent localized changes in magnification, can also be estimated in intrinsic calibration.

Precise gaze estimation relies on accurate calibration of the eye-tracking cameras. The goal of this project is to investigate if the intrinsic calibration process can be made faster with a reduced number of images, while maintaining the robustness. Smart Eye uses a robot arm to capture images for calibrating the intrinsic parameters of their cameras. A subset of their large internal dataset containing images recorded over years was used to study the relationship between the intrinsic parameters and the number of images in this project.

Traditional calibration methods, such as checkerboard pattern calibration [2], Direct Linear Transform [3], and the widely used Zhang's method based on projection error minimization [4], are widely accepted techniques for establishing camera parameters. These methods typically involve capturing multiple images of a known calibration pattern (e.g., a checkerboard) from different orientations. The images are then used to compute both intrinsic and extrinsic parameters by minimizing projection errors [4].

Many current calibration techniques present challenges. These approaches often require multiple images of a calibration object, such as a checkerboard [5] [6] [7], dot grid [8], or spherical object [8]. Alternatively, some methods depend on identify-

ing specific scene features, like straight lines or vanishing points in geometrically structured scenes [9] [10] [11]. Among these techniques, the checkerboard based calibration [5] [6] [7] is the most commonly used, which involves capturing multiple views of the checkerboard. It is robust and suitable for controlled environments.

There are also approaches that consider using different calibration targets, like a moving Light Emitting Diode (LED) for wide-area multiple-camera environment [12] and a custom calibration cube [13].

Other targets include state-of-the-art visual fiducials [1] such as ARTag, AprilTag, ArUco, and STag [14] [15] [16].

Another technique uses GEOCAL, a geometric camera calibration device that uses a beam expanded laser in combination with a diffractive optical element to project patterns directly on the camera sensor [17].

Target based calibration is still the accuracy gold standard, when used to capture many views with strong pose variety and corners pushed to the image periphery, paired with guided capture to avoid bad pose sets [18]. ChArUco, a modern fiducial that improves corner/localization robustness, performs well in mediocre lighting and partial occlusion, and has been validated in recent calibration/pose datasets [19] [20] [21].

Typically, the calibration process is manual. An expert user takes many images of an object of known geometry with the camera, typically a checkerboard with known cell dimensions and spacing, from different points of view. Then, using traditional algorithms, the relations between the detected target in the captured images, and the known structure of the target are found. The value of the calibration parameters can be estimated from these relations. For example, in case of a Charuco board, the algorithm would aim at detecting the location of the corners of each cell in the image, and relate them to the previously stored geometry of the target. An example of images used for manual calibration is demonstrated in Figure 1.1.

Manual calibration technique is still popularly used and is supported by many frameworks such as OpenCV and Matlab, however the performance of these algorithms greatly depend on the selection of poses and results in low quality if the user is not an expert in the field. The motivation behind using a robotic arm for intrinsic calibration as opposed to a manual procedure stems from several key advantages related to precision, repeatability and efficiency. By utilizing a robotic arm in the intrinsic calibration process, we achieve higher precision, repeatability, and control over the camera poses in front of the checkerboard, which leads to a more accurate and efficient calibration process. This eliminates the challenges and inconsistencies associated with manual camera movements and opens the door for automating and scaling the calibration process with minimal human involvement [22]. Furthermore, once we have determined the robot positions, we can adjust the external lighting

[1] Visual fiducials are artificial landmarks designed to be easy to recognize and distinguish from one another.

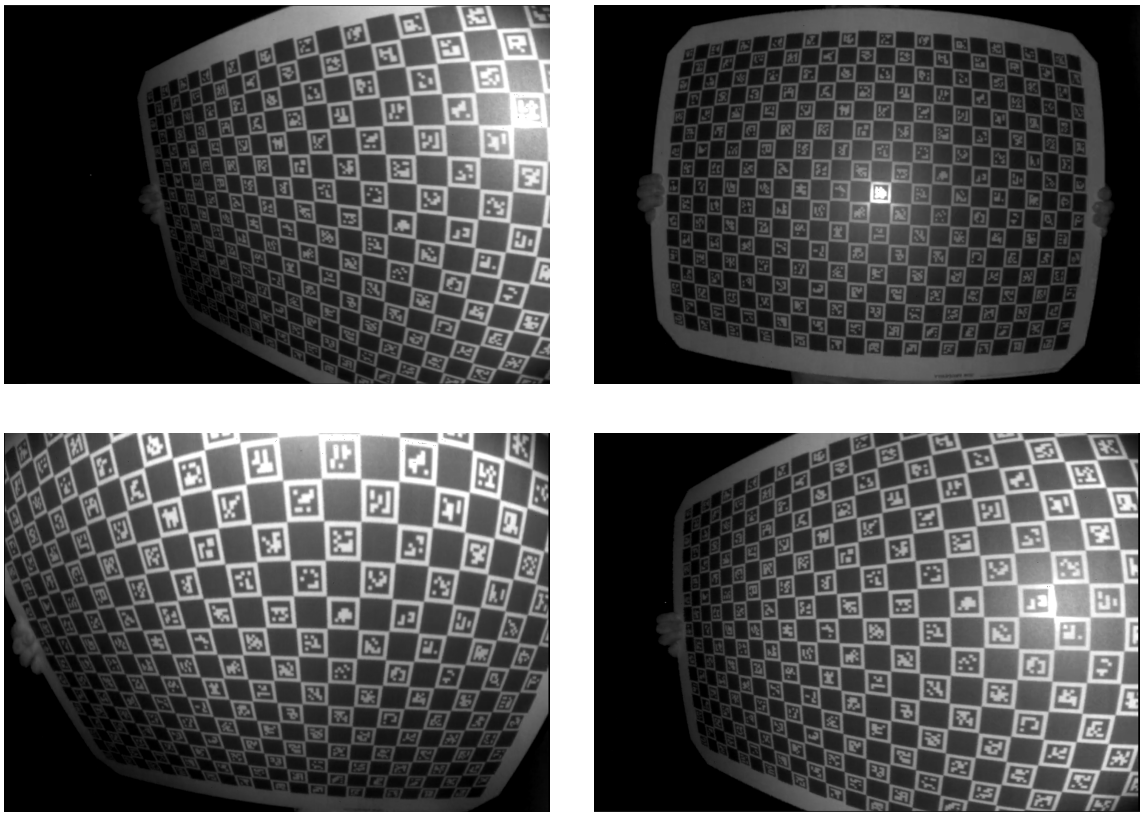


Figure 1.1: Sample dataset used to calibrate a camera using a Charuco board

to minimize the risk of strong reflections and thus saturated pixel regions with loss of information. The robotic arm ensures a level of consistency and reliability that would be hard to achieve manually, making it a compelling choice for eye-tracking camera calibration, especially in high-precision applications.

Though direct studies on using robotic arms for eye-tracking calibration are limited, the principles from camera calibration and robotic automation in related fields provide a solid foundation for this approach [22]. It's highly beneficial for ensuring the accuracy and reliability of intrinsic camera parameters, which are crucial for high-performance remote eye-tracking systems.

1.2 Purpose / Aim of the Study

The aim of this thesis is to perform intrinsic calibration on multiple sets of real and synthetic images as input and analyze the calibration results to determine the optimal number of images for calibration. The real images are obtained with the camera positioned on the robot arm.

Smart Eye's internal intrinsic calibration software is used to perform the intrinsic calibration. The software is based on OpenCV's intrinsic calibration package which uses different models to optimize the intrinsic parameters and distortion coefficients. In standard intrinsic calibration, the size of the image is used to initialize the intrinsic parameters, which is continuously optimized while reading the images. This proce-

dure will be referred to as Standard calibration in the rest of this report. Intrinsic calibration using a different initialization procedure where the intrinsic parameters are optimized from an initial set of images and the distortion coefficients are optimized from the remaining set of images is also evaluated. The intrinsic calibration with this initialization procedure will be referred to as Two-step calibration in the rest of this report.

1.3 Research Question(s)

The main research question that this thesis aims to address is:

How does the number of images relate to the performance of the intrinsic camera calibration for a standard lens with relatively low distortion?

As described in section 1.1, images are captured for intrinsic calibration using a robot arm for precision, repeatability and efficiency. To further optimize the process, the number of images beyond which the calibration results do not improve significantly is estimated by performing intrinsic calibration on both synthetic and real images. The synthetic images have known intrinsics and are considered as the baseline to evaluate the models used in calibration.

1.4 Methodology Overview

This study investigates the performance of the intrinsic calibration software with standard calibration and two-step calibration to determine the optimal number of images required for calibration. This is achieved by performing the steps defined below.

Step 1: Data collection for both real and synthetic images.

Step 2: Intrinsic calibration with standard and two-step calibration.

Step 3: Result analysis to determine optimal N.

1.5 Limitations

The robot used to capture the images is a small tabletop robot with a reach[2] of 490mm, and not an industrial robot with higher reach and payload[3] resulting in a smaller boundary of the robot's workspace. The robot also has a payload of 300g, and is very sensitive while changing poses during calibration.

[2]The reach of a robot refers to the outer boundary of the robot's workspace, how far it can reach outward, upward or downward.

[3]The payload of a robot arm is the maximum weight it can safely carry at its end effector.

1.6 Key Contributions

The key contributions of this work are as follows:

1. A script to automate sampling images, performing intrinsic calibration and plotting the results.
2. Results of standard and two-step calibration.
3. A function to transform between coordinate systems.

1.7 Thesis Structure

The report is organized as follows:

Chapter 2 gives relevant background on camera intrinsics and calibration. It concludes by providing an overview of the entire system used in this thesis.

Chapter 3 explains the theory behind the algorithms used in this project. It includes the necessary equations and optimization followed in the intrinsic calibration.

Chapter 4 gives information on the different components of the robotic setup, hardware, software and software libraries used during the study.

Chapter 5 describes the experiments carried out, both using synthetic and real images.

Chapter 6 consolidates the experimental results with relevant plots.

Chapter 7 follows with the closing arguments and ideas for future work.

2

Theoretical Background

The theory chapter provides an outline of relevant theory including camera models, intrinsic matrix, calibration and concludes with an overview of the system considered in this thesis.

2.1 Pinhole Camera Model

The camera model is a function that maps the 3D world points onto the 2D image plane and is designed to closely model the physical camera. The selection of camera model depends on the target application but essentially they can be classified according to their ability to capture perspective or not. The cameras studied in this thesis are Driver Monitoring System (DMS) cameras with standard lenses and are best represented with the pinhole camera model.

The pinhole camera model is the simplest and most widely used perspective camera model [23]. It consists of a tiny hole, or aperture, through which light rays from a 3D scene pass and form an image on the opposite side of the hole. Pinhole cameras have no lenses, which eliminates the complexities associated with lens-induced distortions. This simplicity makes the pinhole model especially useful for theoretical studies in computer vision, as it strictly adheres to the principles of projective geometry. By avoiding lens-induced distortions, this model allows for straightforward mathematical calculations without the need for complex correction equations.

The simplicity of the pinhole camera model lies in its ability to create sharp images without the use of lenses. By allowing a narrow beam of light through the tiny aperture, the model eliminates most aberrations and produces a clear image. However, this simplicity limits light intake, resulting in darker images and requiring longer exposure times for adequate brightness.

The view of a scene is obtained by projecting a scene's 3D point P_w into the image plane using a perspective transformation which forms the corresponding pixel p. The distortion-free projective transformation given by a pinhole camera model is described in equation 2.1.

$$s\,p = \mathbf{K}\left[\mathbf{R} \mid \mathbf{t}\right] P_w \tag{2.1}$$

where

P_w	3D point expressed with respect to the World Coordinate System (WCS)
p	2D pixel in the image plane
$\mathbf{K}\left[\mathbf{R} \mid \mathbf{t}\right]$	camera projection matrix
$\mathbf{K}$	intrinsic matrix
$\left[\mathbf{R} \mid \mathbf{t}\right]$	extrinsic matrix consisting of 3x3 rotation matrix $\mathbf{R}$, and 3x1 translation vector $\mathbf{t}$ that positions the camera in WCS
s	the projective transformation's arbitrary scaling and not a part of the camera model

The projection of a point P onto the image plane using the pinhole camera model is illustrated in Figure 2.1.

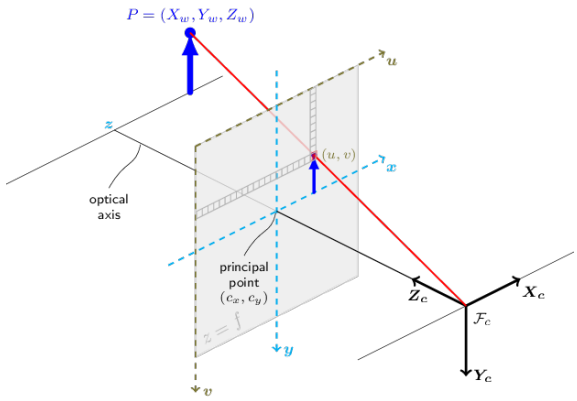


Figure 2.1: Projection of a point P onto the image plane using the pinhole camera model

The image formed contains the projections of all the points in 3D world whose straight line with the camera centre intersects the image plane. In the pinhole camera model, there are five parameters defining the characteristics of the camera, called *intrinsic* parameters:

- focal length in both axes (f_x and f_y)
- 2D principal point (c_x and c_y), and
- skew coefficient (γ)

These parameters are gathered in the intrinsic or projection matrix, $\mathbf{K}$ as described by equation 2.2.

$$\mathbf{K} = \begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

The focal length is the distance between the pinhole and the image plane, measured in pixels. In a true pinhole camera, both f_x and f_y have the same value, which is illustrated as f in Figure 2.2.

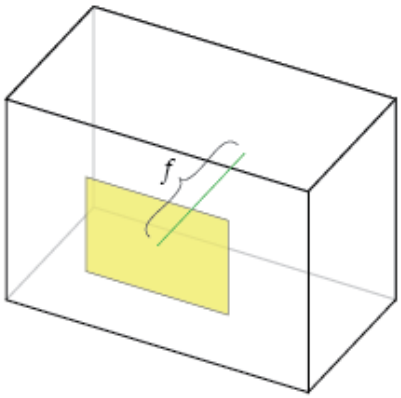


Figure 2.2: Focal length in a pinhole camera

In practice, f_x and f_y can differ for a number of reasons such as,

- Flaws in the digital camera sensor.
- The image has been non-uniformly scaled in post-processing.
- The camera's lens induces unintentional distortion.
- The camera uses an anamorphic format, where the lens compresses a widescreen into a standard-sized sensor.
- Errors in camera calibration.

In all of these cases, the resulting image has non-square pixels.

The camera's principal axis is the line perpendicular to the image plane that passes through the pinhole. Principal point is the intersection of the principal axis on the image plane, with the location relative to the image plane's origin, as illustrated in Figure 2.3. The principal point is also measured in pixels.

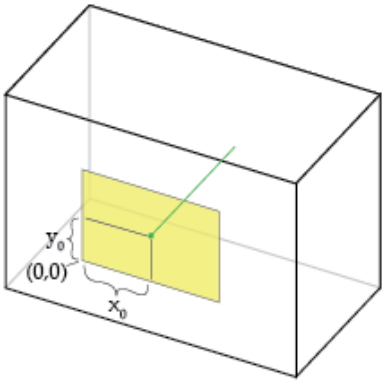


Figure 2.3: Principal point in a pinhole camera

The transformation from 3D world coordinates(X_w, Y_w, Z_w) to 2D image coordinates (u, v) involves several steps:

1. World to Camera Coordinates

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \mathbf{R} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + \mathbf{t} \tag{2.3}$$

   where $\mathbf{R}$ is the 3x3 rotation matrix, and $\mathbf{t}$ is the 3x1 translation vector that positions the camera in the world coordinate system.

2. Camera to Image Coordinates

   Using the intrinsic matrix $\mathbf{K}$, the 3D camera coordinates are projected onto the 2D image plane,

$$p = \mathbf{K} P_c \tag{2.4}$$

3. The pixel coordinates (u, v) are derived from:

$$\begin{aligned} u &= f_x \frac{X_c}{Z_c} + c_x, \\ v &= f_y \frac{Y_c}{Z_c} + c_y \end{aligned} \tag{2.5}$$

2.2 Lens Distortion Models

In order to get better imaging results, a lens is usually installed in front of the camera. However, this introduces image distortions [24]. A perfect lens would

magnify the scene evenly across the field of view. The distortions are a local change in magnification that comes from the geometry of the lens.

In addition to estimating the camera parameters, we can also account for lens distortion in the same optimization. There are many models for distortion, like the *equidistant, fish-eye and radial-tangential* [25]. In this thesis, we have considered only the radial-tangential or *Brown-Conrady model* which considers the two most relevant distortion effects. This model is more applicable to linear cameras used in automotive applications for detecting distant objects with minimal distortion [26], which is also the type of cameras focused in this thesis.

The Brown-Conrady model is a mathematical framework used to describe how real-world lens distortions affect the image coordinates captured by a camera [27]. This model attempts to quantify the relationship between the ideal, undistorted coordinates and the actual, distorted coordinates that appear due to lens imperfections.

To account for lens-induced distortion, the Brown-Conrady model modifies the ideal image coordinates, x_i and y_i, to get the distorted coordinates, x_d and y_d, as described by equation 2.6.

$$\begin{aligned} x_d &= x_i(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 x_i y_i + p_2(r^2 + 2{x_i}^2) \\ y_d &= y_i(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2{y_i}^2) + 2p_2 x_i y_i \end{aligned} \tag{2.6}$$

where x_d and y_d are the distorted coordinates, x_i and y_i are the ideal coordinates, k_1, k_2, k_3 are the radial distortion coefficients and p_1, p_2 are the tangential distortion coefficients.

Radial distortion is due to the fact that the geometric lens shape affects straight lines resulting in local, uneven magnification. Based on its shape, radial distortion is of three types: barrel, pincushion and mustache distortions [24]. In barrel distortion, the image magnification decreases with distance from the optical axis, while in pincushion distortion, the effect is the opposite. Mustache distortion is a mix of barrel and pincushion distortion. The types of radial distortion is demonstrated in Figure 2.4.

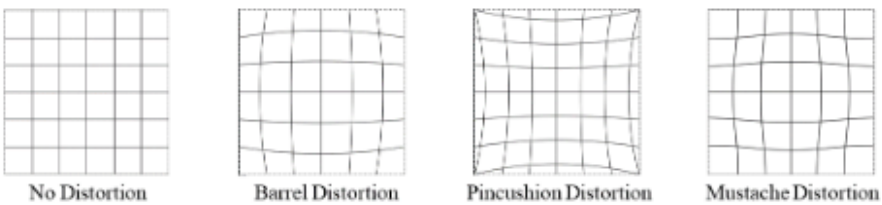


Figure 2.4: Radial distortion

Tangential distortion is the effect that occurs when the image sensor and the lens are not parallel to each other. This misalignment causes the image to appear tilted

or stretched. This effect is shown in Figure 2.5 where the concentric axes suffers from a slight deviation based on its angle.

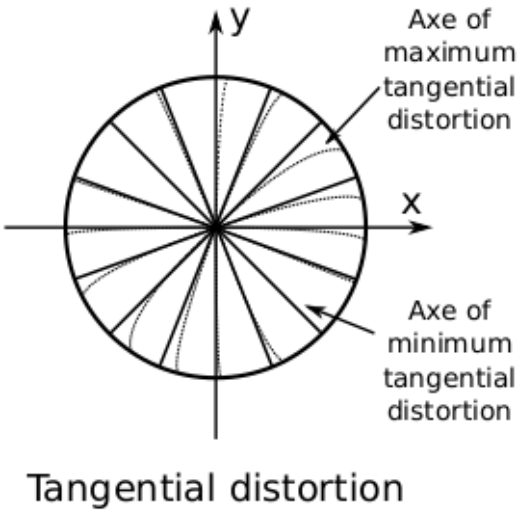


Figure 2.5: Tangential distortion

The procedure to correct lens distortion is iterative. First the undistorted point due to radial distortion is estimated, and then using this corrected point, the tangential distortion is corrected. This is usually done after projecting the points in the image plane.

2.3 System Overview

The overview of the system considered in this thesis is presented in Figure 2.6.

All the experiments in this thesis is performed using Smart Eye's intrinsic calibration software. The software, developed by the in-house research team is based on OpenCV's intrinsic calibration package. The inputs to the software are a batch of images and a configuration describing the model selection and properties of the image and target. The batch of input images consists of two sets of images, one for calibration and the other for verification. The calibration results, or the output from the software are the intrinsic parameters - focal length and principal point, distortion coefficients, RMS reprojection error and detection coverage for both calibration and verification.

The real images consists of multiple views of the fixed planar target, captured using the camera mounted on the robot's end effector.

The synthetic images are generated in Smart Eye's custom build of the open source 3D creation pipeline, Blender with user defined, known intrinsics. The test chart used for real testing is also modeled in this custom build. Rendering and saving of the raw images is automated by using Blender Python API.

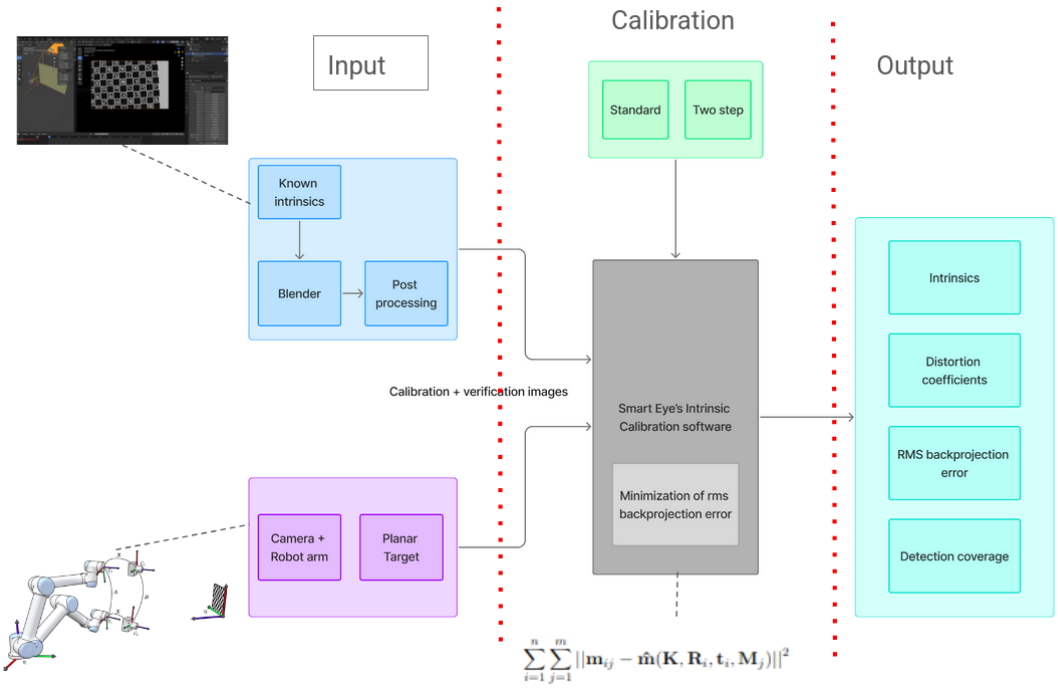


Figure 2.6: System Overview for optimization of intrinsic calibration of cameras

Noise and gaussian blur is added to the raw synthetic images to make them more realistic, mimicking the sensor in the real cameras, using Smart Eye's post processing pipeline.

The synthetic dataset is also considered as the baseline for the calibration software, as the intrinsics are known. For this project, the synthetic images have the same extrinsics as the real images, and the camera intrinsics and distortion coefficients are chosen in the range of the calibration results from the real data set.

3

Methodology

This chapter summarizes the background behind the optimization followed in Smart Eye's intrinsic calibration software. The main topics covered are the maximum-likelihood estimation in the intrinsic calibration and quantities that help ascertain the quality of calibration such as RMS reprojection error and detection coverage.

3.1 Intrinsic Camera Calibration

Smart Eye's intrinsic calibration software is based on OpenCV's intrinsic calibration package.

Like most calibration algorithms, this software is also built upon Zhang's method [4], which consists of computing an analytical closed-form solution of the parameters, followed by a non-linear optimization based on maximum-likelihood criterion.

The analytical solution is found by assuming that the target plane is on $Z = 0$, and calculating the homography between the target plane and the image plane. Then, by imposing orthonormality among columns in the rotation matrix, we arrive at a closed-form of the camera intrinsic parameters as described by equation 3.1

$$\begin{aligned}\mathbf{B} = \mathbf{K}^{-T}\mathbf{K}^{-1} &\equiv \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{{f_x}^2} & -\frac{\gamma}{{f_x}^2 f_y} & \frac{c_y\gamma - c_x f_y}{{f_x}^2 f_y} \\ -\frac{\gamma}{{f_x}^2 f_y} & \frac{\gamma^2}{{f_x}^2 {f_y}} + \frac{1}{{f_y}^2} & -\frac{\gamma(c_y\gamma - c_x f_y)}{{f_x}^2 {f_y}^2} - \frac{c_y}{{f_y}^2} \\ \frac{c_y\gamma - c_x f_y}{{f_x}^2 f_y} & -\frac{\gamma(c_y\gamma - c_x f_y)}{{f_x}^2 {f_y}^2} - \frac{c_y}{{f_y}^2} & -\frac{(c_y\gamma - c_x f_y)^2}{{f_x}^2 {f_y}^2} + \frac{{c_y}^2}{{f_y}^2} + 1 \end{bmatrix}\end{aligned} \tag{3.1}$$

We are given n images of a model plane and there are m points on the model plane. Assuming that the image points are corrupted by independent, and identically distributed noise, the subsequent maximum-likelihood estimation can be obtained by minimizing the squared distance between the detected target points in the image, with the reprojected target points using the estimated target pose, as described in equation 3.2

$$\sum_{i=1}^{n}\sum_{j=1}^{m}||\mathbf{m}_{ij}-\hat{\mathbf{m}}(\mathbf{K},\mathbf{R}_i,\mathbf{t}_i,\mathbf{M}_j)||^2 \tag{3.2}$$

where, m_{ij} denotes the location of corner j in image i, and $\hat{\mathbf{m}}(\mathbf{K},\mathbf{R}_i,\mathbf{t}_i,\mathbf{M}_j)$ is the projection of corner $\mathbf{M}_j$ on the image plane, given intrinsic matrix $\mathbf{K}$, rotation matrix $\mathbf{R}_i$ and translation vector $\mathbf{t}_i$. A rotation $\mathbf{R}$ is parameterized by a vector of three parameters, denoted by $\mathbf{r}$, which is parallel to the rotation axis and whose magnitude is equal to the rotation angle. $\mathbf{R}$ and $\mathbf{r}$ are related by the Rodrigues formula [28]. Minimizing equation 3.2 is a nonlinear minimization problem, which is solved with the Levenberg-Marquardt Algorithm [29].

This optimization is used to refine the accuracy of the calibration, and at the same time include other effects such as lens distortion. The technique only requires the camera to observe the planar pattern from a few different orientations. Although the minimum number of orientations is two if pixels are square, it is recommended to have four or five different orientations for better quality [4].

The algorithm performs the following steps

- Compute the initial intrinsic parameters or read them from the input parameters.
- Estimate the initial camera pose as if the intrinsic parameters are already known. The distortion coefficients are all set to zero initially unless specified otherwise.
- Run the global Levenberg-Marquardt Algorithm [29] to minimize the reprojection error, that is the total sum of square distances between the observed feature points and the projected (using current estimates for camera parameters and poses) object points, as described by equation 3.2.

3.1.1 Standard Calibration Procedure

When the initial intrinsic parameters are computed from the image size as described in equation 3.3, and the above optimization of minimizing rms reprojection error is followed to refine the intrinsic parameters and the distortion coefficients while iterating through each image, for the entire set of images, the process is referred to as standard calibration.

$$\begin{aligned} f &= H \\ (c_x, c_y) &= (\frac{W-1}{2}, \frac{H-1}{2}) \end{aligned} \tag{3.3}$$

where W is the image width and H is the image height, in pixels.

3.1.2 Two-Step Calibration Approach

In two-step calibration, the initial intrinsic parameters are computed from the image size, as described in equation 3.3, and the minimization of rms reprojection error is followed to refine the intrinsic parameters only with the first arbitrarily chosen 5 images. Distortion coefficients are assumed as zero during this step. In the next step, the estimated intrinsic parameters are kept fixed, and the distortion coefficients are estimated by minimizing rms reprojection error, in the remaining set of images.

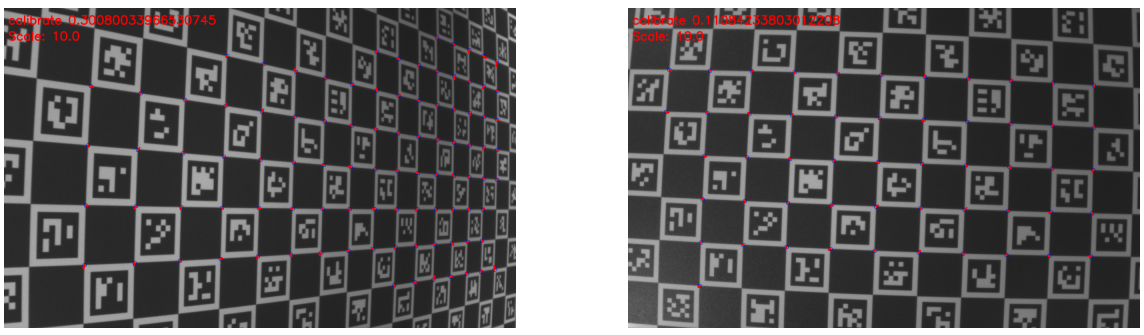
3.2 Quality Metrics

To assess the accuracy of the intrinsic calibration, we mainly consider two parameters - RMS reprojection error and detection coverage.

3.2.1 RMS Reprojection Error

The RMS reprojection error is a measure of how accurately the estimated camera model can project 3D points back onto the 2D image plane. For each calibration point, the known 3D world point is projected onto the image plane using the estimated camera parameters. The euclidean distance between this projected point and the actually observed 2D point in the image is measured. The overall error is reported as the root mean square (RMS) of all these individual errors, which is the result of the minimization of the functional described by equation 3.2.

A lower RMS reprojection error indicates a more accurate camera model. However, a low RMS reprojection error does not always guarantee an accurate calibration. If the detections do not cover the whole image area, the model can overfit to the areas with measurements. This will happen if the target movement was not carefully performed to cover the image corners as shown in Figure 3.1. Figure 3.1b has a low RMS reprojection error while Figure 3.1a has a much higher RMS reprojection error and more image corners. To prevent overfitting, other measures like detection coverage must also be considered.



(a) Image with higher RMS error

(b) Image with lower RMS error

Figure 3.1: Calibration images showing reprojection error vectors

3.2.2 Detection Coverage

Detection coverage measures the percentage of the image area that contains target detections. To compute the detection coverage, the image is partitioned into a 2D grid, and a grid cell is set to true if it contains a measurement (i.e, if a target corner falls within its boundaries). The detection coverage C is then defined as the ratio of the number of cells that contains detections to the total number of cells in the grid as described by equation 3.4.

$$C = \frac{N_{\text{covered}}}{N_{\text{total}}} \times 100\% \tag{3.4}$$

From the Figure 3.1, it can also be observed that the image in Figure 3.1a contains more target detections, thereby contributing to higher detection coverage than the image in Figure 3.1b.

For a good coverage of $\geq 75\%$, the target movement must be so that the target is seen in all image regions and illumination has to enable good image quality in the corners.

4

Experimental Setup

This chapter explains the experimental setup: the different hardware and software considered in this project. This includes specifications of the robot arm, planar target, external illumination and the various software used.

4.1 Charuco board

The planar target used in this thesis is a Charuco board. A Charuco board, short for *Chessboard ArUco*, is a special type of calibration target designed to assist in precise calibration procedures. In the Charuco target, the light squares are uniquely encoded with ArUco markers. Thus, it makes this target possible to carry out calibration even with part occlusion or poor image conditions such as in inhomogeneous lighting, while maintaining the advantage that the intersection of square edges can be easily recovered, while an ordinary checkerboard or circular grid target under these conditions would typically fail [30].

The Charuco board used in this thesis is the `18x25 Checker size 30mm AruCo DICT_5x5` from Calib.io with 18 rows and 25 columns with 30mm checker size as shown in Figure 4.1.

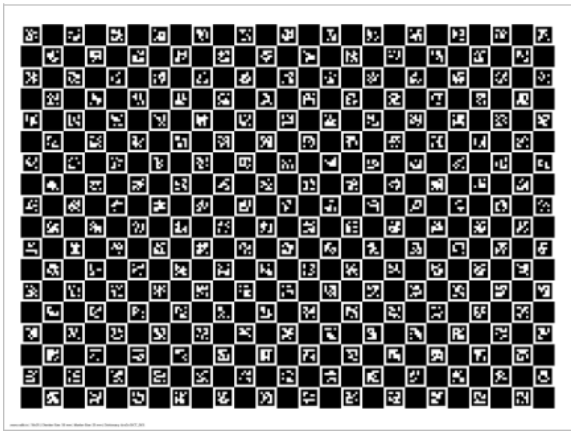


Figure 4.1: Charuco `18x25 Checker size 30mm AruCo DICT_5x5`

The left bottom vertex of the target is assumed as the origin in the left handed

World Coordinate System, WCS in this project, with +x being forward, +y to the right and +z upwards.

4.2 Robot arm

The robot utilized is the Ned2 model from Niryo (Figure 4.2), while the in-house hardware team has developed the external illumination system. The robot is an industrial-grade educational robot with a moderate payload (300g) and long reach (490mm), and is suitable for Smart Eye's research purposes. It can be integrated into several ecosystems. The robot is redeployable and fast to reprogram. This provides flexibility to modify the configuration when required. Further specifications of the robot are given in Table 4.1 [31].

Degrees of freedom	6 rotating joints
Reach (mm)	490
Payload (g)	300
Footprint (mm)	200 × 200
Joint ranges (rad)	$-2.949 \leq$ Joint 1 ≤ 2.949 $-1.83 \leq$ Joint 2 ≤ 0.61 $-1.34 \leq$ Joint 3 ≤ 1.57 $-2.089 \leq$ Joint 4 ≤ 2.089 $-1.919 \leq$ Joint 5 ≤ 1.922 $-2.53 \leq$ Joint 6 ≤ 2.53

Table 4.1: Niryo Ned2 Specifications

The robot can be controlled through multiple interfaces, of which the most user-friendly method is through Niryo Studio, a proprietary Graphical User Interface, GUI based on Blockly visual programming. This platform enables users to design, simulate and execute robotic tasks using an intuitive drag and drop environment. Communication between Niryo Studio and the robot is established via Wi-Fi or Ethernet. This enables remote task deployment and monitoring.

For more advanced operations, the Ned2 provides support for both Python API (PyNiryo) and the Robot Operating System (ROS) framework. The Python API allows for direct, script based control of the robot's joints, end effector and peripheral devices through standard TCP/IP communication.

The external illumination is synchronized to the pulsed illumination emitted by the camera. The internal illumination by the camera is physically blocked when the external illumination is used. To provide constant illumination with minimal impact of unwanted test chart reflexes, the chart is illumined by external sources positioned at the most optimal positions. The illumination setup consists of four LED-clusters, each composed of nine Near Infrared (NIR) diodes. The sources have corresponding wavelength and pulse duration, as the internal flashes, to provide similar exposure and eye safety levels.

Figure 4.2: Niryo Ned2 with the 6 rotating joints

The pulses for the external illumination are triggered by a photodiode mounted in the camera holder, and an external driver is used to power the external illuminators.

The movements are performed by specifying the joint positions as six coordinates (x, y, z, roll, pitch, yaw) in the Robot Coordinate System, RCS with the origin at the centre of the base of the robot.

The base of the robot is located at a distance of 0.7m from the target and has an offset of 0.12m to the right from the left edge of the target.

4.3 Smart Eye's custom build of Blender

Smart Eye's synthetic data team has developed a custom build of Blender, an open-source 3D creation pipeline. The test chart used for real testing is also modeled in this custom build. The synthetic images are generated using Blender's panoramic OpenCV camera object with known intrinsics. Further specifications of the camera is given in Table 4.2.

Image width, W	1280
Image height, H	800
Focal length, f	1644.7
Principal point x-coordinate, c_x	644.96
Principal point y-coordinate, c_x	399.45
Radial distortion coefficient, k_1	-0.2605
Radial distortion coefficient, k_2	0.1387
Radial distortion coefficient, k_3	0
Tangential distortion coefficient, p_1	0
Tangential distortion coefficient, p_2	0

Table 4.2: Blender Camera Specifications

The world origin is set as the test chart object's left bottom vertex, and follows Blender's inherent right handed coordinate system. The camera poses are specified in Euler XYZ format.

Blender has an embedded Python interpreter. This allows the automation of rendering and saving of raw images using Blender Python API [32].

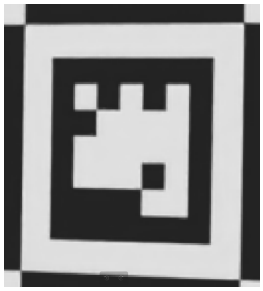
4.4 Smart Eye's post processing pipeline

Smart Eye's post processing custom image augmentation pipeline was utilized to add corrections to the images to make them more realistic. The pipeline is based on the libraries OpenCV and Albumentations, and includes custom transformations to mimic the sensor noise.

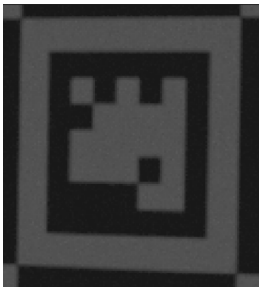
Smart Eye's `AIS6` cameras have the image sensor `OV9284`, and the pipeline is customized as follows:

- ReduceChannels - to go from RGB to Grayscale
- GaussianBlur - to apply fixed Gaussian blur
- DynamicRangeShift - to alter dynamic range[1] of synthetic images to resemble images captured by `OV9284` camera sensor by applying a scaling factor to match the dynamic range, and an offset to match the baseline brightness.
- Float32to10bit - to simulate the digitization step[2], mimicking the output of a real sensor.
- AddSynthNoise - adds synthetic sensor noise to an image, mimicking the behavior of the `OV9284` camera sensor, done in a statistically grounded way, based on real noise distributions sampled from an actual `OV9284` sensor.

The effect of these transformations is presented in Figure 4.3, where a Charuco marker is observed by zooming the original images. The Figure 4.3a is a synthetic image without any imperfections while Figure 4.3b contains noise and blur.



(a) Noise-free synthetic image



(b) Realistic synthetic image

Figure 4.3: Synthetic images - with and without noise and blur

[1]Dynamic range is the ratio between darkest and brightest parts of an image that a sensor can capture at the same time without losing detail.

[2]High Dynamic Range files typically store pixel values as `float32`, very high precision, while real sensors store pixel values as 10-bit unsigned integers.

5

Experiments

5.1 Data collection

This section describes the images that form the inputs to the calibration. This includes details and specifications of the real images captured using the robot arm and processes involved in the generation as well as processing of the synthetic images.

5.1.1 Real images

A previously recorded data set of a batch of Smart Eye's driver monitoring cameras, `AIS6 Batch 3` with normal angular field of view(A-FOV) of $< 120°$ and insignificant tangential distortion was chosen as the real data set for this study. This batch contains twelve cameras with similar intrinsic parameters. For every camera, there is a set of 60 images of which 40 images constitute the calibration set and 20 images constitute the verification set. Images are captured from the same predetermined 60 poses for all the cameras under uniform illumination conditions. The poses are selected such that all the corners of the target are captured from multiple views to ensure sufficient detection coverage and proper estimation of distortion coefficients.

The poses of the cameras while capturing these images are known in the RCS.

A few examples from the calibration set are shown in Figure 5.1. The resolution of the images are 1280x800 pixels.

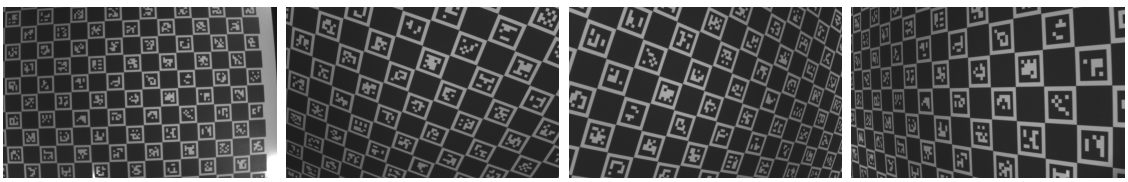


Figure 5.1: Real images with different views, calibration set

After the estimation of intrinsics and distortion coefficients using the calibration images, the algorithm uses the estimated intrinsics and distortion coefficients to compute reprojection errors on the verification images. A few examples from the verification set are presented in Figure 5.2.

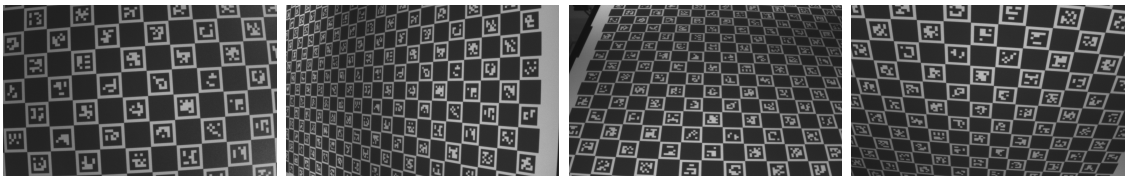


Figure 5.2: Real images with different views, verification set

5.1.2 Synthetic images

Synthetic images with extrinsics that match the real images are generated using Blender. This task also involved finding the transformation between coordinate frames - World Coordinate System (WCS), Robot Coordinate System (RCS) and Blender Coordinate System (BCS) and was done in the following steps -

1. Tranformation from RCS to WCS :

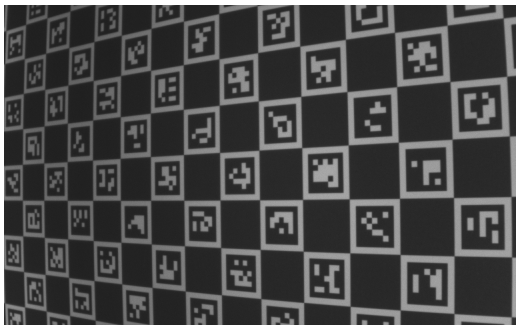
   Rotating counterclockwise around the z-axis by 22.5° to account for the robot's base's yaw, and translating along x and y directions to account for the robot base's offset from the assumed world origin at the bottom left vertex of the Charuco board.

2. Transformation from WCS to BCS :

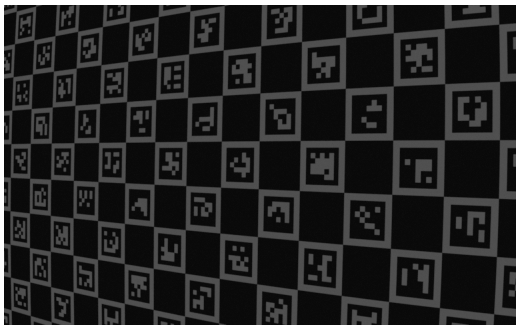
   The x and y coordinates in WCS are y and x coordinates respectively in BCS, and the roll and pitch in WCS are 90°-pitch and -roll respectively in BCS due to left-handedness of WCS and right-handedness of BCS.

The high dynamic range (HDR) synthetic images exported in OpenEXR format are made realistic by adding noise and gaussian blur using Smart Eye's post processing pipeline. Since the ground truth, or the intrinsic parameters for these images are known, performing intrinsic calibration on these images can be considered as a baseline for the calibration software.

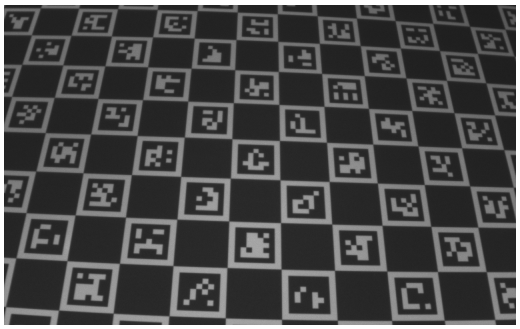
Figure 5.3 shows real and synthetic images for two different views. The difference in brightness is because of the non identical lighting between the real and synthetic data collection.



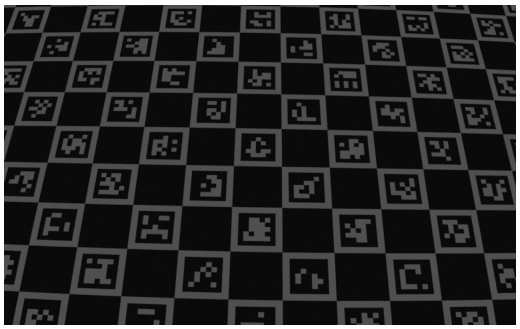
(a) Real image from camera 1, view 1



(b) Realistic synthetic image, view 1



(c) Real image from camera 1, view 2



(d) Realistic synthetic image, view 2

Figure 5.3: Comparison of real and synthetic images for different views

5.2 Model configuration

Smart Eye's intrinsic calibration software optimizes the rms reprojection error subject to configurable constraints. These constraints are called models and are combinations of OpenCV's calibration flags.

Smart Eye's `AIS6 batch 3` DMS cameras have standard lenses with narrow to normal A-FOV, low radial distortion and insignificant tangential distortion. The applicable assumptions to formulate the models are consolidated in Equation 5.1.

$$\begin{aligned} f_x &= f_y \\ p_1, p_2 &= 0 \\ k_3, k_4, k_5, k_6 &= 0 \end{aligned} \tag{5.1}$$

5.2.1 Standard calibration

The models SE and SEPP are configured for standard calibration. Both the models include OpenCV flags that satisfy the assumptions described by Equation 5.1.

The SE model fixes the principal point to the image center. The calibration yields the estimates for focal length, radial distortion coefficients k_1 and k_2, and RMS reprojection error when the principal point is assumed to be at the image center. This model is preferred to get the best global set of intrinsics when calibrating a

large population of cameras.

The SEPP model includes the principal point in the optimization. The calibration yields the estimates for focal length, principal point, radial distortion coefficients k_1 and k_2, and RMS reprojection error. This model is preferred to get exact intrinsics for a particular camera.

5.2.2 Two-step calibration

A model, TSFP is configured for two-step calibration. The first step yields the focal length and principal point from the first five images to be used as fixed parameters in the second step. The second step includes the optimization of radial distortion coefficients k_1 and k_2, and the RMS reprojection error.

5.3 Testing and analysis

As the number of images, N is a quantity of interest, the calibration is tested on batches of images with varying N.

A test script was created to automate:

1. The selection of five random combinations each of N images from the set of calibration images with $N = 5, 10, 15, 20, 25, 30, 35$
2. The testing of Smart Eye's intrinsic calibration software on each subset of images
3. Plotting error bars for every intrinsic parameter and distortion coefficient with the mean and 2σ standard deviations for the corresponding N
4. Plotting rms reprojection error for calibration and verification for the corresponding N
5. Plotting coverage for the corresponding N

The script was used to perform intrinsic calibration and visualize the results for both the real and synthetic image sets.

6

Results

Smart Eye's intrinsic calibration software was used to perform standard and two-step intrinsic calibration of previously recorded real images from `AIS6 batch 3` cameras using the robot arm, and realistic synthetic images with the same poses as that of the real images. The results of the calibration, consisting of the focal length, principal point, distortion coefficients k_1 and k_2, RMS reprojection error and detection coverage are plotted across the number of calibration images, N.

The plots for focal length in the y-direction are omitted because it is the same as the plots for focal length in the x-direction, since the cameras in this study are known to have square pixels and the aspect ratio is maintained constant in calibration.

The plots for two-step calibration start with $N = 10$, since the samples starting from $N = 10$ were used for two-step calibration, using five images for focal length and principal point optimization, and the remaining images for distortion optimization.

Mean values for intrinsic parameters and distortion coefficients across N for all models are tabulated after respective figures for ease of numerical comparison.

The detection coverage is the same for all models, and can be verified in the plots. This is also the reason why only one detection coverage is presented for each camera.

The calibration results for synthetic images are presented first, followed by real images for two cameras.

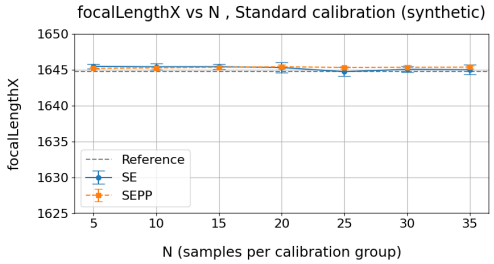
6.1 Synthetic images

6.1.1 Focal length and principal point

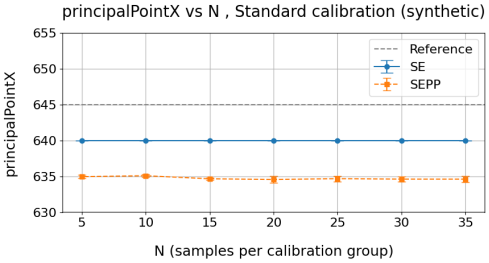
The SE model does not optimize principal point and fixes it at the image center, thereby resulting in a straight line plot with zero standard deviation. This result is included in the plots just for the purpose of visual comparison for the offset in the estimate when the principal point is included in the optimization. The principal point with the SE model is $c_x = 640$ pixels and $c_y = 400$ pixels.

The optimization of focal length and principal point showing the corresponding

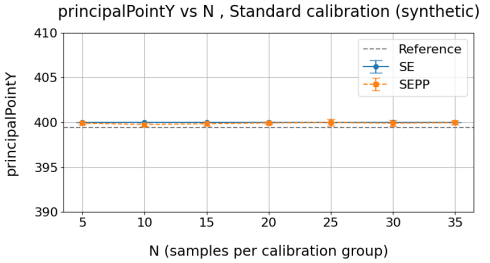
means and 2σ standard deviation across N for the synthetic images are presented in Figure 6.1.



(a) Focal length, f_x across N (synthetic)



(b) Principal point x-coordinate, c_x across N (synthetic)



(c) Principal point y-coordinate, c_y across N (synthetic)

Figure 6.1: Focal length and principal point across N for synthetic images

Figure 6.1a shows how the calibrated focal length f_x varies across the number of images with f_x as the y-axis and N as the x-axis. The reference line represents the ground truth, i.e, focal length of the Blender camera used to generate the synthetic images ($f_x = 1644.7$ pixels).

Figure 6.1b and Figure 6.1c show how the x-coordinate and the y-coordinate of the principal point, c_x and c_y varies across the number of images, N with c_x and c_y as the corresponding y-axis and N as the x-axis. The respective reference lines represent the ground truth principal point used to generate the images in Blender ($c_x = 644.96$ pixels and $c_y = 399.45$ pixels).

The mean values of the estimated focal length for all three models across N are tabulated in Table 6.1a.

The mean values of the estimated c_x are tabulated in Table 6.1b and the mean values of the estimated c_y are tabulated in Table 6.1c for both standard calibration (SEPP) and two-step calibration (TSFP).

N	5	10	15	20	25	30	35
mean f_x, SE	1645.46	1645.4	1645.41	1645.29	1644.71	1645.05	1644.99
mean f_x, SEPP	1645.19	1645.25	1645.33	1645.41	1645.3	1645.33	1645.25
mean f_x, TSFP	-	1644.6	1644.51	1644.42	1644.47	1644.54	1644.58

(a) Mean f_x across N, for synthetic images

N	5	10	15	20	25	30	35
mean c_x , SEPP	634.95	635.08	634.65	634.55	634.67	634.6	634.6
mean c_x , TSFP	-	634.6	634.69	634.69	634.68	634.88	634.45

(b) Mean c_x across N, for synthetic images

N	5	10	15	20	25	30	35
mean c_y, SEPP	399.92	399.78	399.86	399.95	400.0	399.92	399.98
mean c_y, TSFP	-	400.29	400.22	400.04	400.15	400.35	400.05

(c) Mean c_y across N, for synthetic images

Table 6.1: Summary of intrinsic parameter means across N for synthetic images

From Figure 6.1 and Table 6.1, it can be observed that, the optimization of focal length and principal point is consistently good across all values of N, since the estimated values are very close to the corresponding ground truth values.

This gives the conclusion,

- The optimization of focal length and principal point is just as good for lower values of N as it is for higher values of N.

6.1.2 Distortion coefficients

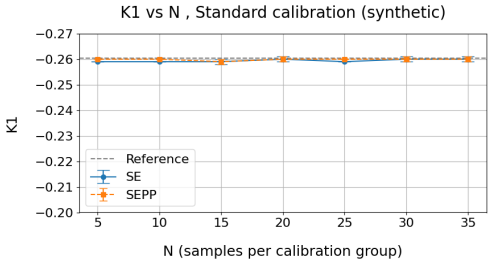
The optimization of radial distortion coefficients k_1 and k_2 across N for the synthetic images are presented in Figure 6.2.

Figure 6.2a and Figure 6.2b shows how the calibrated radial distortion coefficient k_1 varies across the number of images for standard and two-step calibrations respectively, with k_1 along the y-axis and N along the x-axis. The reference line represents the ground truth, i.e, k_1 of the Blender camera used to generate the synthetic images ($k_1 = -0.2604$ pixels).

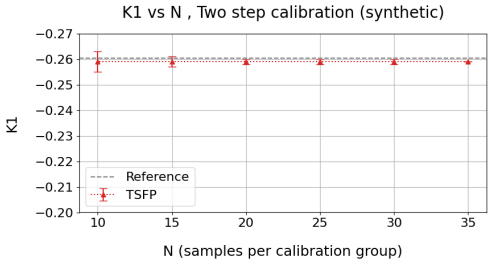
Figure 6.2c and Figure 6.2d shows how the calibrated radial distortion coefficient k_2 varies across the number of images for standard and two-step calibrations respectively, with k_2 along the y-axis and N along the x-axis. The reference line represents the ground truth, i.e, k_2 of the Blender camera used to generate the synthetic images ($k_2 = 0.1387$ pixels).

Mean values of k_1 and k_2 across N for all three models are tabulated in Table 6.2a

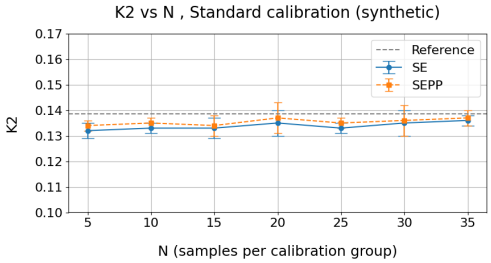
and Table 6.2b respectively.



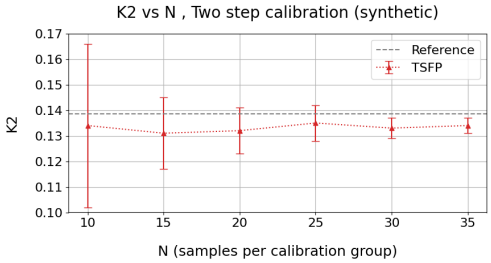
(a) Radial distortion coefficient, k_1 standard calibration (synthetic)



(b) Radial distortion coefficient, k_1 two-step calibration (synthetic)



(c) Radial distortion coefficient, k_2 standard calibration (synthetic)



(d) Radial distortion coefficient, k_2 two-step calibration (synthetic)

Figure 6.2: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on synthetic images

N	5	10	15	20	25	30	35
mean k_1, *SE*	-0.259	-0.259	-0.259	-0.26	-0.259	-0.26	-0.26
mean k_1, *SEPP*	-0.26	-0.26	-0.259	-0.26	-0.26	-0.26	-0.26
mean k_1, *TSFP*	-	-0.259	-0.259	-0.259	-0.259	-0.259	-0.259

(a) Mean k_1 across N, for synthetic images

N	5	10	15	20	25	30	35
mean k_2, SE	0.132	0.133	0.133	0.135	0.133	0.135	0.136
mean k_2, SEPP	0.134	0.135	0.134	0.137	0.135	0.136	0.137
mean k_2, TSFP	-	0.134	0.131	0.132	0.135	0.133	0.134

(b) Mean k_2 across N, for synthetic images

Table 6.2: Summary of distortion coefficient means across N for synthetic images

From Figure 6.2 and Table 6.2, it can be observed that:

- The optimization of radial distortion coefficient k_1 is consistently good across all values of N, since the estimated values are very close to the ground truth value for both standard and two-step calibrations.

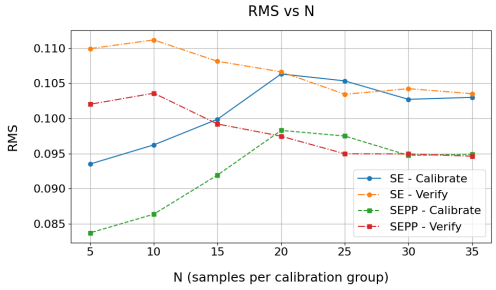
- The optimization of k_1 does not seem to be dependent on the number of images.
- The optimization of k_2 improves significantly and the standard deviation decreases with more number of images for two-step calibration.
- The optimization of k_2 is closest to the true value at $N = 20$ and $N = 35$ in standard calibration.

This gives the conclusion,

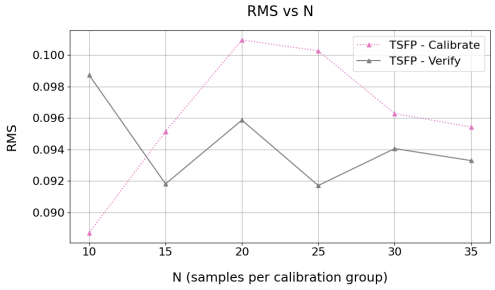
- The optimization of k_1 is good even for the lowest value of N, and the optimization of k_2 is best when $N = 20$ or $N = 35$.

6.1.3 RMS reprojection error and detection coverage

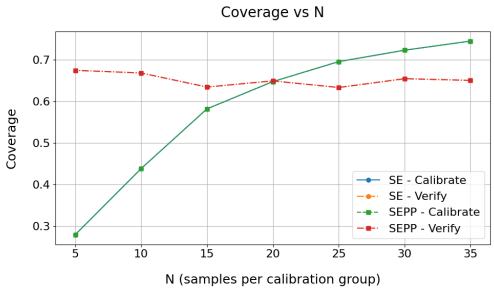
The optimization of RMS reprojection error across N for standard and two-step calibrations for the synthetic images are presented in Figure 6.3a and Figure 6.3b respectively. Figure 6.3c shows the detection coverage of the synthetic images across N.



(a) RMS reprojection error across N, standard calibration (synthetic)



(b) RMS reprojection error across N, two-step calibration (synthetic)



(c) Detection coverage across N, (synthetic)

Figure 6.3: RMS reprojection error and detection coverage across N, for standard and two-step calibration on synthetic images

From Figure 6.3, it can be observed that, the RMS reprojection error is always less

than 0.115, less than the reasonable range of 0.2 to 0.4 pixels indicating accurate calibration results. This is also verified from Figure 6.1, Table 6.1, Figure 6.2 and Table 6.2.

Detection coverage is poor for low values of N such as $N = 5$ and $N = 10$. However, the RMS reprojection error is still less than 0.2, and the calibration results as verified previously are accurate, indicating an overfit.

The detection coverage improves significantly between $N = 10$ and $N = 25$, after which the rate of improvement is low. The verification images have a detection coverage of $60 - 70\%$.

This gives the conclusions,

- RMS reprojection error and detection coverage improves with more number of images.
- Even though the intrinsics optimized at low number of images are close to the ground truth, using them to get accurate reprojections has higher uncertainity.
- In this case, $N = 20$ would be a sufficient number of images with reasonably good estimates, coverage and low RMS reprojection error.

6.2 Real images, camera 1

The calibration results with the real images from the first camera in `AIS6 batch 3`, camera `ID_1_1` are presented below.

6.2.1 Focal length and principal point

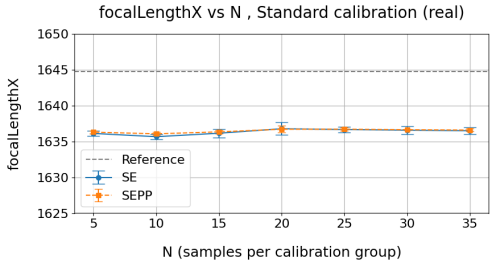
The SE model does not optimize principal point and fixes it at the image center, thereby resulting in a straight line plot with zero standard deviation. This result is included in the plots just for the purpose of visual comparison for the offset in the estimate when the principal point is included in the optimization. The principal point with the SE model is $c_x = 640$ pixels and $c_y = 400$ pixels.

The optimization of focal length and principal point showing the corresponding means and 2σ standard deviation across N for the real images from camera `ID_1_1` are presented in Figure 6.4.

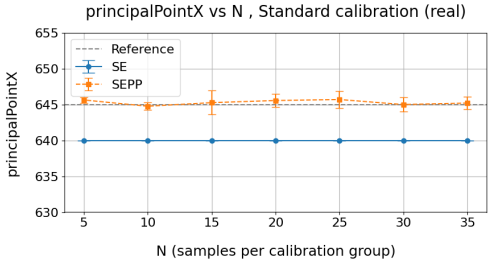
Figure 6.4a shows how the calibrated focal length f_x varies across the number of images. The reference line represents the focal length of the Blender camera used to generate the synthetic images and is only shown for visual comparison between ranges for real and synthetic images.

Figure 6.4b and Figure 6.4c show how the x-coordinate and the y-coordinate of the principal point, c_x and c_y varies across the number of images, N. The respective

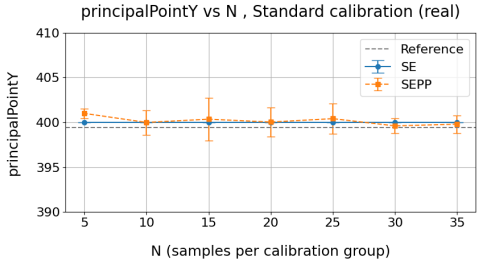
reference lines represent the principal point used to generate the images in Blender, and is only shown for visual comparison between ranges for real and synthetic images.



(a) Focal length, f_x across N (camera 1)



(b) Principal point x-coordinate, c_x across N (camera 1)



(c) Principal point y-coordinate, c_y across N (camera 1)

Figure 6.4: Focal length and principal point across N for real images from camera `ID_1_1`

The mean values of the estimated focal length for all three models across N are tabulated in Table 6.3a.

The mean values of the estimated c_x are tabulated in Table 6.3b and the mean values of the estimated c_y are tabulated in Table 6.3c for both standard calibration (SEPP) and two-step calibration (TSFP).

N	5	10	15	20	25	30	35
mean f_x, SE	1636.11	1635.66	1636.14	1636.77	1636.65	1636.56	1636.49
mean f_x, SEPP	1636.29	1635.06	1636.33	1636.70	1636.70	1636.63	1636.58
mean f_x, TSFP	-	1637.65	1636.83	1636.97	1637.04	1637.04	1637.18

(a) Mean f_x across N, for real images from camera 1

N	5	10	15	20	25	30	35
mean c_x , SEPP	645.64	644.76	645.26	645.56	645.7	645.0	645.21
mean c_x , TSFP	-	640.0	642.39	642.71	642.63	642.58	643.0

(b) Mean c_x across N, for real images from camera 1

N	5	10	15	20	25	30	35
mean c_y, SEPP	401.01	400.0	400.37	400.05	400.43	399.62	399.81
mean c_y, TSFP	-	394.6	397.73	397.61	397.42	396.59	397.64

(c) Mean c_y across N, for real images from camera 1

Table 6.3: Summary of intrinsic parameter means across N for real images from camera `ID_1_1`

From Figure 6.4 and Table 6.3, it can be observed that the optimization of focal length and principal point is consistently good across all values of N, since the estimated values are very close to each other with minimal standard deviation.

This gives the conclusion,

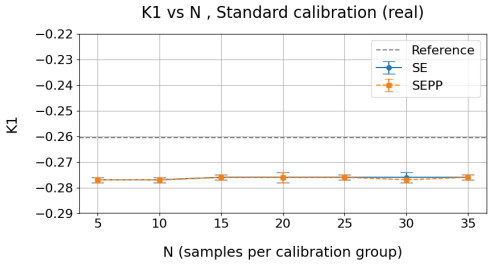
- The optimization of focal length and principal point is just as good for lower values of N as it is for higher values of N.

6.2.2 Distortion coefficients

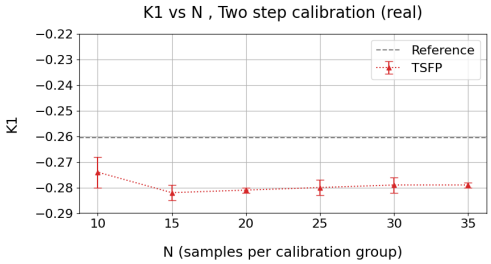
The optimization of radial distortion coefficients k_1 and k_2 across N for the real images from camera `ID_1_1` are presented in Figure 6.5.

Figure 6.5a and Figure 6.5b shows how the calibrated radial distortion coefficient k_1 varies across the number of images for standard and two-step calibrations respectively. The reference line represents the value of k_1 of the Blender camera used to generate the synthetic images and is only shown for visual comparison between ranges for real and synthetic images.

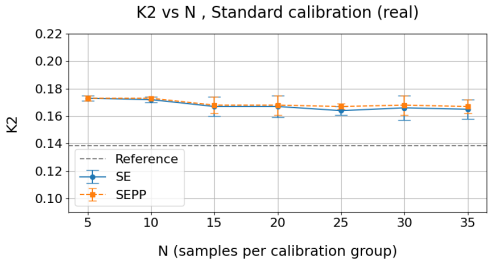
Figure 6.5c and Figure 6.5d shows how the calibrated radial distortion coefficient k_2 varies across the number of images for standard and two-step calibrations respectively. The reference line represents the value of k_2 of the Blender camera used to generate the synthetic images and is only shown for visual comparison between ranges for real and synthetic images.



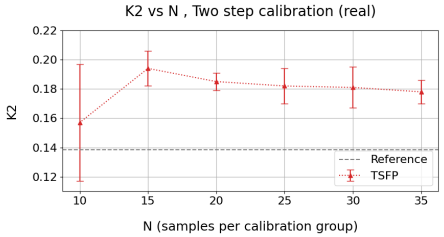
(a) Radial distortion coefficient, k_1 standard calibration (camera 1)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 1)



(c) Radial distortion coefficient, k_2 standard calibration (camera 1)



(d) Radial distortion coefficient, k_2 two-step calibration (camera 1)

Figure 6.5: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_1_1`

Mean values of k_1 and k_2 across N for all three models are tabulated in Table 6.4a and Table 6.4b respectively.

N	5	10	15	20	25	30	35
mean k_1, *SE*	-0.277	-0.277	-0.276	-0.276	-0.276	-0.276	-0.276
mean k_1, *SEPP*	-0.277	-0.277	-0.276	-0.276	-0.276	-0.277	-0.276
mean k_1, *TSFP*	-	-0.274	-0.282	-0.281	-0.28	-0.279	-0.279

(a) Mean k_1 across N, for real images (camera 1)

N	5	10	15	20	25	30	35
mean k_2, *SE*	0.173	0.172	0.167	0.167	0.164	0.166	0.165
mean k_2, *SEPP*	0.173	0.173	0.168	0.168	0.167	0.168	0.167
mean k_2, *TSFP*	-	0.157	0.194	0.185	0.182	0.181	0.178

(b) Mean k_2 across N, for real images (camera 1)

Table 6.4: Summary of distortion coefficient means across N for real images from camera `ID_1_1`

From Figure 6.5 and Table 6.4, it can be observed that:

- The optimization of radial distortion coefficient k_1 is consistently good across

all values of N for standard calibration, since the estimated values are very close to each other with minimal standard deviation.

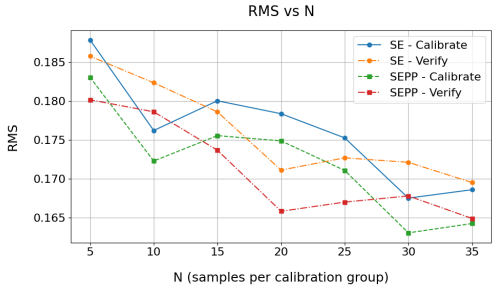
- The optimization of k_1 does not seem to be dependent on the number of images.
- The optimization of k_2 for two-step calibration has significant standard deviation from the optimization of k_2 for standard calibration, and is worst at $N = 10$ for two-step calibration.

This gives the conclusion,

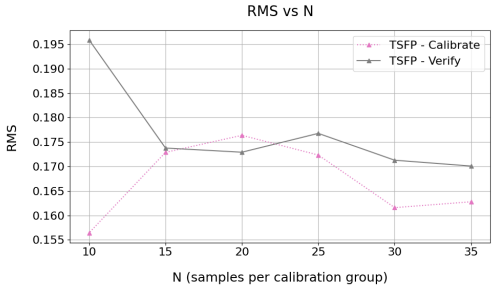
- The optimization of k_1 is good even for low values of N, and the optimization of k_2 is best when $N \geq 20$.

6.2.3 RMS reprojection error and detection coverage

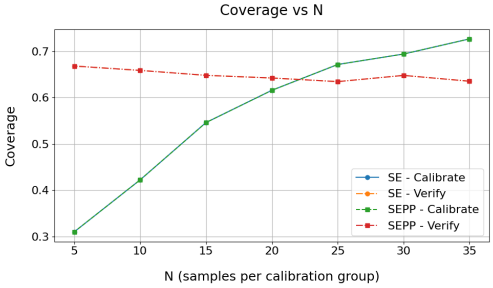
The optimization of RMS reprojection error across N for standard and two-step calibrations for the real images from camera `ID_1_1` are presented in Figure 6.6a and Figure 6.6b respectively. Figure 6.6c shows the detection coverage of the real images across N.



(a) RMS reprojection error across N, standard calibration (camera 1)



(b) RMS reprojection error across N, two-step calibration (camera 1)



(c) Detection coverage across N, (camera 1)

Figure 6.6: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_1_1`

From Figure 6.6, it can be observed that:

- The RMS reprojection error is always less than 0.2, which is less than the reasonable range of 0.2 to 0.4 pixels indicating accurate calibration results.
- Detection coverage is poor for low values of N such as $N = 5$ and $N = 10$.
- The two-step calibration has lowest RMS reprojection error for the calibration set and highest RMS reprojection error for the verification set at $N = 10$. As previously observed from 6.5 and Table 6.4, the optimization of radial distortion coefficient k_1 is worst at $N = 10$, indicating overfit.
- The detection coverage improves significantly between $N = 10$ and $N = 25$, after which the rate of improvement slows. The verification images have a detection coverage of $60 - 70\%$.

This gives the conclusions,

- RMS reprojection error and detection coverage improves with more number of images.
- Even though the intrinsics optimized at low number of images are close to the ground truth, using them to get accurate reprojections has higher uncertainity.
- In this case, $N = 20$ would be a sufficient number of images with reasonably good estimates, coverage and low RMS reprojection error.

6.3 Real images, camera 2

The calibration results with the real images from the second camera in `AIS6 batch 3`, camera `ID_2_1` are presented below.

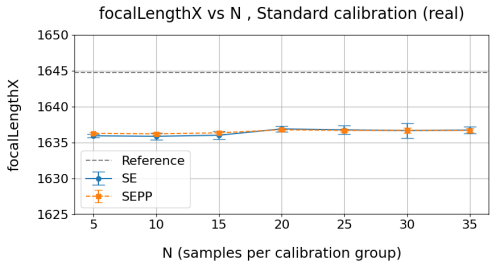
6.3.1 Focal length and principal point

The SE model does not optimize principal point and fixes it at the image center, thereby resulting in a straight line plot with zero standard deviation. This result is included in the plots just for the purpose of visual comparison for the offset in the estimate when the principal point is included in the optimization. The principal point with the SE model is $c_x = 640$ pixels and $c_y = 400$ pixels.

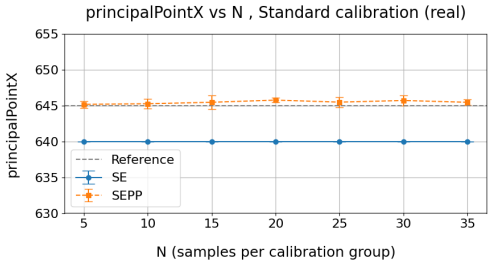
The optimization of focal length and principal point showing the corresponding means and 2σ standard deviation across N for the real images from camera `ID_2_1` are presented in Figure 6.7.

Figure 6.7a shows how the calibrated focal length f_x varies across the number of images. The reference line represents the focal length of the Blender camera used to generate the synthetic images and is only shown for visual comparison between ranges for real and synthetic images.

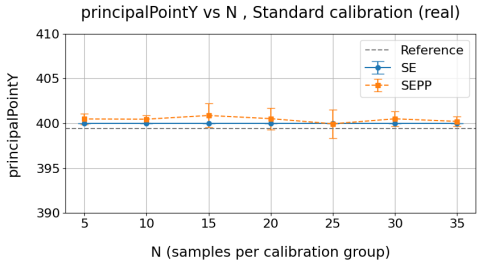
Figure 6.7b and Figure 6.7c show how the x-coordinate and the y-coordinate of the principal point, c_x and c_y varies across the number of images, N. The respective reference lines represent the principal point used to generate the images in Blender, and is only shown for visual comparison between ranges for real and synthetic images.



(a) Focal length, f_x across N (camera 2)



(b) Principal point x-coordinate, c_x across N (camera 2)



(c) Principal point y-coordinate, c_y across N (camera 2)

Figure 6.7: Focal length and principal point across N for real images from camera `ID_2_1`

The mean values of the estimated focal length for all three models across N are tabulated in Table 6.5a.

The mean values of the estimated c_x are tabulated in Table 6.5b and the mean values of the estimated c_y are tabulated in Table 6.5c for both standard calibration (SEPP) and two-step calibration (TSFP).

N	5	10	15	20	25	30	35
mean f_x, SE	1635.92	1635.85	1636.0	1636.88	1636.75	1636.65	1636.72
mean f_x, SEPP	1636.26	1636.2	1636.33	1636.76	1636.67	1636.67	1636.67
mean f_x, TSFP	-	1637.65	1637.14	1636.76	1636.86	1637.21	1636.96

(a) Mean f_x across N, for real images from camera 2

N	5	10	15	20	25	30	35
mean c_x , SEPP	645.17	645.25	645.45	645.76	645.47	645.7	645.45
mean c_x , TSFP	-	640.0	642.55	642.16	642.95	643.58	642.71

(b) Mean c_x across N, for real images from camera 2

N	5	10	15	20	25	30	35
mean c_y, SEPP	400.5	400.48	400.89	400.54	399.96	400.52	400.23
mean c_y, TSFP	-	394.6	397.52	396.61	398.54	397.84	396.93

(c) Mean c_y across N, for real images from camera 2

Table 6.5: Summary of intrinsic parameter means across N for real images from camera `ID_2_1`

From Figure 6.7 and Table 6.5, it can be observed that, the optimization of focal length and principal point is consistently good across all values of N, since the estimated values are very close to each other with minimal standard deviation.

This gives the conclusion,

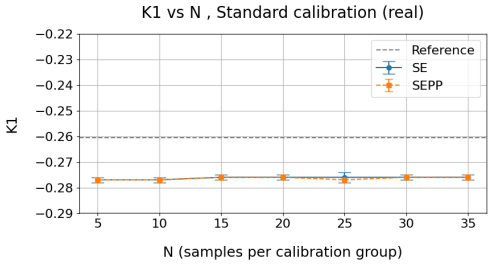
- The optimization of focal length and principal point is just as good for lower values of N as it is for higher values of N.

6.3.2 Distortion coefficients

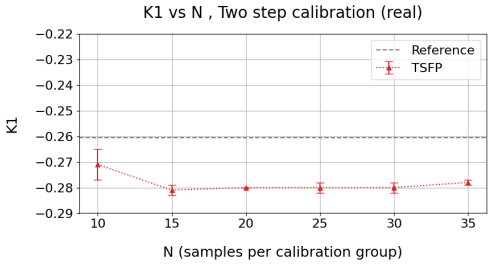
The optimization of radial distortion coefficients k_1 and k_2 across N for the real images from camera `ID_2_1` are presented in Figure 6.8.

Figure 6.8a and Figure 6.8b shows how the calibrated radial distortion coefficient k_1 varies across the number of images for standard and two-step calibrations respectively. The reference line represents the value of k_1 of the Blender camera used to generate the synthetic images and is only shown for visual comparison between ranges for real and synthetic images.

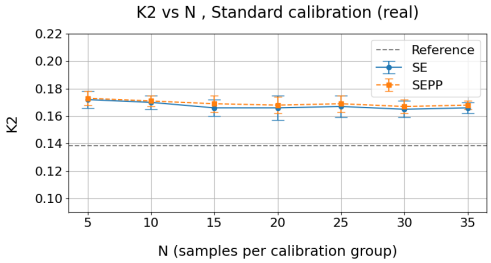
Figure 6.8c and Figure 6.8d shows how the calibrated radial distortion coefficient k_2 varies across the number of images for standard and two-step calibrations respectively. The reference line represents the value of k_2 of the Blender camera used to generate the synthetic images and is only shown for visual comparison between ranges for real and synthetic images.



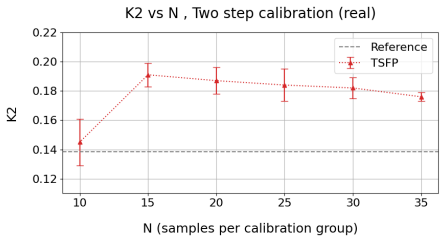
(a) Radial distortion coefficient, k_1 standard calibration (camera 2)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 2)



(c) Radial distortion coefficient, k_2 standard calibration (camera 2)



(d) Radial distortion coefficient, k_2 two-step calibration (camera 2)

Figure 6.8: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_2_1`

Mean values of k_1 and k_2 across N for all three models are tabulated in Table 6.6a and Table 6.6b respectively.

N	5	10	15	20	25	30	35
mean k_1, SE	-0.277	-0.277	-0.276	-0.276	-0.276	-0.276	-0.276
mean k_1, SEPP	-0.277	-0.277	-0.276	-0.276	-0.277	-0.276	-0.276
mean k_1, TSFP	-	-0.271	-0.281	-0.28	-0.28	-0.28	-0.278

(a) Mean k_1 across N, for real images (camera 2)

N	5	10	15	20	25	30	35
mean k_2, SE	0.172	0.17	0.166	0.166	0.167	0.165	0.166
mean k_2, SEPP	0.173	0.171	0.169	0.168	0.169	0.167	0.168
mean k_2, TSFP	-	0.145	0.191	0.187	0.184	0.182	0.176

(b) Mean k_2 across N, for real images (camera 2)

Table 6.6: Summary of distortion coefficient means across N for real images from camera `ID_2_1`

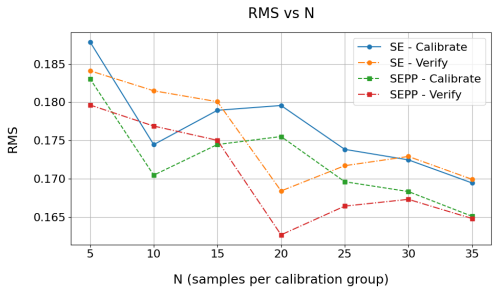
From Figure 6.8 and Table 6.6, it can be observed that:

- The optimization of radial distortion coefficient k_1 is consistently good across all values of N for standard calibration, since the estimated values are very close to each other with minimal standard deviation.
- The optimization of k_1 does not seem to be dependent on the number of images for standard calibration, however for two-step calibration the optimization of k_1 improves from $N \geq 15$.
- The optimization of k_2 for two-step calibration has significant standard deviation from the optimization of k_2 for standard calibration, and is worst at $N = 10$ for two-step calibration.

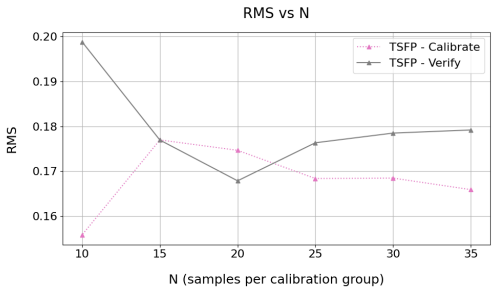
This gives the conclusions,

- The optimization of k_1 is good even for low values of N, and the optimization of k_2 is best when $N \geq 10$ for standard calibration.
- The optimization of both k_1 and k_2 is worst at $N = 10$ for two-step calibration.

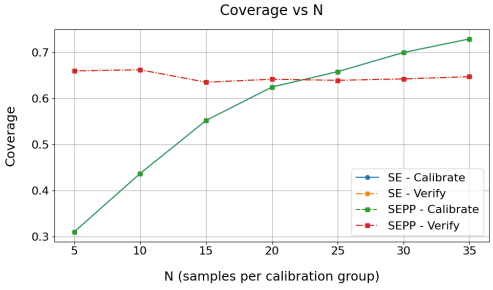
6.3.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 2)



(b) RMS reprojection error across N, two-step calibration (camera 2)



(c) Detection coverage across N, (camera 2)

Figure 6.9: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_2_1`

The optimization of RMS reprojection error across N for standard and two-step calibrations for the real images from camera `ID_2_1` are presented in Figure 6.9a and Figure 6.9b respectively. Figure 6.9c shows the detection coverage of the real images across N.

From Figure 6.9, it can be observed that:

- The RMS reprojection error is always less than 0.2, which is less than the reasonable range of 0.2 to 0.4 pixels indicating accurate calibration results.
- Detection coverage is poor for low values of N such as $N = 5$ and $N = 10$.
- The two-step calibration has lowest RMS reprojection error for the calibration set and highest RMS reprojection error for the verification set at $N = 10$. As previously observed from 6.8 and Table 6.6, the optimization of radial distortion coefficients is worst at $N = 10$, indicating overfit.
- The detection coverage improves significantly between $N = 10$ and $N = 25$, after which the rate of improvement slows. The verification images have a detection coverage of $60 - 70\%$.
- The RMS reprojection error in the verification images is lowest at $N = 20$.

This gives the conclusions,

- RMS reprojection error and detection coverage improves with more number of images.
- Even though the intrinsics optimized at low number of images are close to the ground truth, using them to get accurate reprojections has higher uncertainity.
- In this case, $N = 20$ would be a sufficient number of images with reasonably good estimates, coverage and low RMS reprojection error.

The observations and inferences made for the above cases also extend to the other cameras used in this study. Results of the experiments done on the images from the remaining cameras are presented in the Appendix.

7

Conclusion

This chapter presents the main conclusions drawn from the results, discusses key observations from the experiments, addresses limitations of the study, and suggests ideas for future work.

In general, it is found that for the estimation of focal length and principal point the number of images required is less. For the estimation of radial distortion coefficients, adding more images improves the results, especially while estimating the radial distortion coefficient k_2. Even though the calibration results are of reasonable accuracy, using $N = 20$ images for calibration is recommended to get results which are useful for applications requiring accurate reprojection.

The calibration results from using a reduced number of calibration images ($N = 20$) are comparable to the calibration results from using the original number of calibration images ($N = 40$), thereby resulting in reduced calibration times.

In the first experiment, the intrinsic calibration was performed on synthetic images with known intrinsics. The results revealed that the focal length and principal point estimation was accurate even with the smallest number of images. The estimation of radial distortion coefficient k_1 was also accurate with the smallest number of images for both standard and two-step calibrations. The estimation of the radial distortion coefficient k_2, however, showed significant improvement with increasing the number of images. This effect was more evident in two-step calibration where the estimation of k_2 was the worst at $N = 10$, and improved with increasing N. The RMS reprojection error was always less than the reasonable range of 0.2 to 0.4 pixels indicating accurate calibration, which is also verified from comparing the results with the known ground truth values. The detection coverage showed significant increase until $N = 20$, after which it continued to increase at a slower rate. It is also noticed that including the principal point in the optimization during standard calibration (SEPP) gave slightly lower rms reprojection error compared to when the principal point is assumed at the image center (SE). Two-step calibration did not show any improvement in the results compared to standard calibration. An interesting observation is that, even though the optimization of intrinsic parameters and distortion coefficients are reasonably good with the smallest number of images i.e, $N = 5$, the RMS reprojection error is highest at $N = 5$ which gives uncertainties in using the estimated intrinsic parameters and distortion coefficients for applications

that require accurate reprojection. From the results of calibrating the synthetic images, it was concluded that $N = 20$ offered a reasonable detection coverage and low RMS reprojection errors.

In the second experiment, the intrinsic calibration was performed on the previously recorded real images from `AIS6 batch 3` cameras using the robot arm. It was observed that the conclusions from the synthetic images could be extended to the real images as well. Focal length and principal point estimation was consistently good even with a very small number of images. The estimation of radial distortion coeffcient k_1 is good across all N for standard calibration, but has a slightly higher estimated value at $N = 10$ in two-step calibration compared to standard calibration. The values of k_1 for $N \geq 15$ is similar for both standard and two-step calibrations. The optimization of k_2 seems consistent across all values of N in standard calibration, however is estimated to be much lower at $N = 10$ and much higher for $N \geq 15$ in two-step calibration compared to standard calibration. The two-step calibration did not show any improvement in the optimizations. The RMS reprojection errors decrease with increase in N, and the detection coverage increases with increase in N. From the results of calibrating the real images, it was concluded that $N = 20$ offered a reasonable detection coverage and low RMS reprojection errors.

These experiments revealed that the calibration produces reliable results at a reduced number of images, $N = 20$ for DMS cameras with low distortion, reducing the time required for calibration.

To estimate the poses of the camera in WCS, the transformations were done directly to the available pose data in RCS. The dimensions of the end effector were not taken in consideration, which has introduced slight offsets. The robot, having a small payload has resulted in minor instabilities while changing poses, which also results in differences between the specified poses in RCS and the actual poses of the camera.

Several promising directions exist for extending this study. One possibility is to analyze how the movement of the robotic arm influences the estimation of camera intrinsics and to identify motion patterns that are optimal for calibration. Another is to increase pose diversity in order to improve detection coverage. The study could also be extended by investigating camera models with wider angular fields of view or fisheye lenses, as well as models that include tangential distortion.

Bibliography

[1] R. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal on Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 2003.

[2] J. Liu, Z. Yang, H. Huo, and T. Fang, "Camera calibration method with checkerboard pattern under complicated illumination," *Journal of Electronic Imaging*, vol. 27, no. 4, pp. 043038–043038, 2018.

[3] Y. Liu, M. Liu, Y. Zhou, L. Dong, and L. Kong, "Camera calibration method based on improved dlt algorithm," in *Optoelectronic Imaging and Multimedia Technology X*, vol. 12767, pp. 251–256, SPIE, 2023.

[4] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2002.

[5] C. Mei and P. Rives, "Single view point omnidirectional camera calibration from planar grids," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3945–3950, IEEE, 2007.

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

[7] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easily calibrating omnidirectional cameras," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5695–5701, IEEE, 2006.

[8] S. Shah and J. Aggarwal, "A simple calibration procedure for fish-eye (high distortion) lens camera," in *Proceedings of the 1994 IEEE international Conference on Robotics and Automation*, pp. 3422–3427, IEEE, 1994.

[9] M. Antunes, J. P. Barreto, D. Aouada, and B. Ottersten, "Unsupervised vanishing point detection and camera calibration from a single manhattan image with radial distortion," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4288–4296, 2017.

[10] J. P. Barreto and H. Araujo, "Geometric properties of central catadioptric line images and their application in calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1327–1333, 2005.

[11] C. Brauer-Burchardt and K. Voss, "A new algorithm to correct fish-eye-and strong wide-angle-lens-distortion from single images," in *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, vol. 1, pp. 225–228, IEEE, 2001.

[12] J. P. Barreto and K. Daniilidis, "Wide area multiple camera calibration and estimation of radial distortion," in *Workshop on Omidirectional Vision and Camera Networks*, pp. 5–16, Citeseer, 2004.

[13] M. Li, "Camera calibration of a head-eye system for active vision," in *Computer Vision—ECCV'94: Third European Conference on Computer Vision Stockholm, Sweden, May 2–6, 1994 Proceedings, Volume I 3*, pp. 541–554, Springer, 1994.

[14] M. Kalaitzakis, B. Cain, S. Carroll, A. Ambrosi, C. Whitehead, and N. Vitzilaios, "Fiducial markers for pose estimation," *Journal of Intelligent & Robotic Systems*, vol. 101, p. 71, Mar 2021.

[15] J. Mellor, "Realtime camera calibration for enhanced reality visualization," in *International Conference on Computer Vision, Virtual Reality, and Robotics in Medicine*, pp. 471–475, Springer, 1995.

[16] J. Huai, Y. Shao, G. Jozkow, B. Wang, D. Chen, Y. He, and A. Yilmaz, "Geometric wide-angle camera calibration: A review and comparative study," *Sensors (Basel, Switzerland)*, vol. 24, no. 20, p. 6595, 2024.

[17] J. Sandhu, B. Rheingans, A. Davis, G. Van Harten, and C. J. Bruegge, "Geometric calibration of the multi-angle imager for aerosols (maia)," in *Earth Observing Systems XXVI*, vol. 11829, pp. 230–247, SPIE, 2021.

[18] A. Richardson, J. Strom, and E. Olson, "Aprilcal: Assisted and repeatable camera calibration," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1814–1821, 2013.

[19] K. Kayan, S. Alexandropoulos, R. Jain, Y. Zuo, E. Liang, and J. Deng, "Princeton365: A diverse dataset with accurate camera pose," *arXiv preprint arXiv:2506.09035*, 2025.

[20] M. C. Roshan, M. Isaksson, and A. Pranata, "A geometric calibration method for thermal cameras using a charuco board," *Infrared Physics Technology*, vol. 138, p. 105219, 2024.

[21] G. H. An, S. Lee, M.-W. Seo, K. Yun, W.-S. Cheong, and S.-J. Kang, "Charuco board-based omnidirectional camera calibration method," *Electronics*, vol. 7,

no. 12, p. 421, 2018.

[22] R. Boada Farràs, "Active camera calibration for robotic systems," Master's thesis, Universitat Politècnica de Catalunya, 2016.

[23] P. Sturm, S. Ramalingam, J.-P. Tardif, S. Gasparini, J. Barreto, *et al.*, "Camera models and fundamental concepts used in geometric computer vision," *Foundations and Trends® in Computer Graphics and Vision*, vol. 6, no. 1–2, pp. 1–183, 2011.

[24] R. Fan, L. Wang, M. J. Bocus, and I. Pitas, "Computer stereo vision for autonomous driving," 2020.

[25] C. Ricolfe-Viala and A.-J. Sanchez-Salmeron, "Lens distortion models evaluation," *Applied optics*, vol. 49, no. 30, pp. 5914–5928, 2010.

[26] O. Bogdan, V. Eckstein, F. Rameau, and J.-C. Bazin, "Deepcalib: A deep learning approach for automatic intrinsic calibration of wide field-of-view cameras," in *Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production*, pp. 1–10, 2018.

[27] D. Brown, "Close-range camera calibration photo," 1971.

[28] O. Faugeras, *Three-dimensional computer vision: a geometric viewpoint*. MIT press, 1993.

[29] J. J. Moré, "The levenberg-marquardt algorithm: implementation and theory," in *Numerical analysis: proceedings of the biennial Conference held at Dundee, June 28–July 1, 1977*, pp. 105–116, Springer, 2006.

[30] I. Enebuse, M. Foo, B. S. K. K. Ibrahim, H. Ahmed, F. Supmak, and O. S. Eyobu, "A comparative review of hand-eye calibration techniques for vision guided robots," *IEEE Access*, vol. 9, pp. 113143–113155, 2021.

[31] N. SAS, "Ned2 technical specifications." `https://docs.niryo.com/robots/ned2/technical-specifications/`, 2025.

[32] Blender Foundation, *API Overview: Blender Python API*. Blender Foundation, 2025.

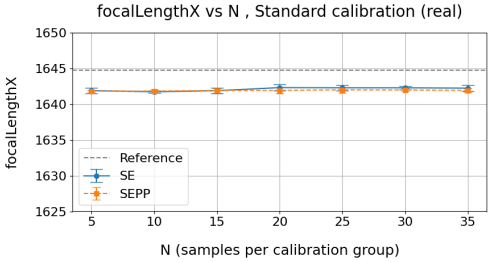
A

Appendix 1

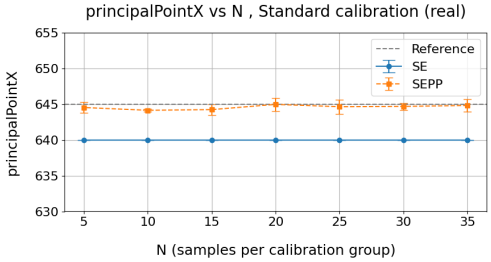
Results of the experiments on real images from the remaining cameras in the batch used in this study are presented in this chapter.

A.1 Real images, camera 3

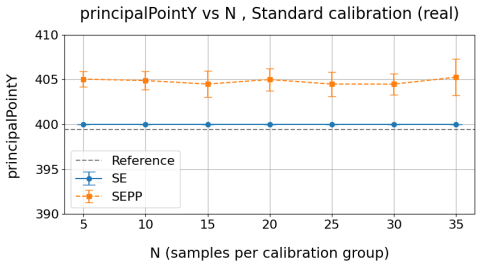
A.1.1 Focal length and principal point



(a) Focal length, f_x across N (camera 3)



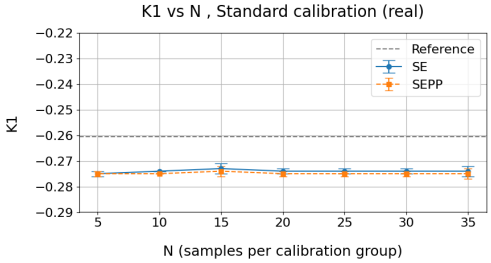
(b) Principal point x-coordinate, c_x across N (camera 3)



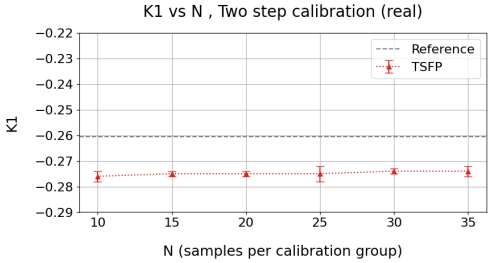
(c) Principal point y-coordinate, c_y across N (camera 3)

Figure A.1: Focal length and principal point across N for real images from camera `ID_3_1`

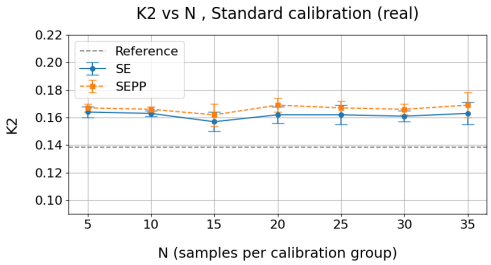
A.1.2 Distortion coefficients



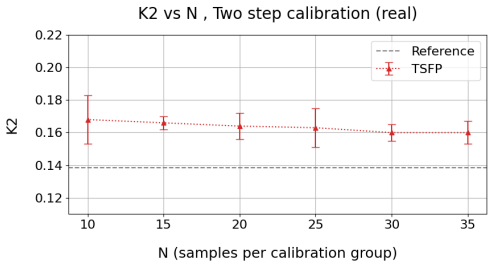
(a) Radial distortion coefficient, k_1 standard calibration (camera 3)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 3)



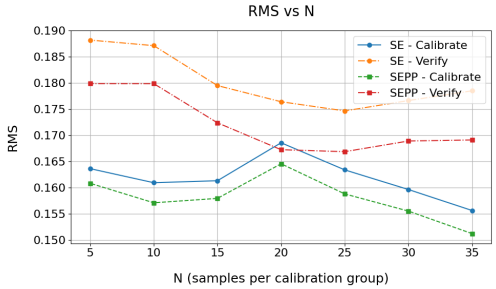
(c) Radial distortion coefficient, k_2 standard calibration (camera 3)



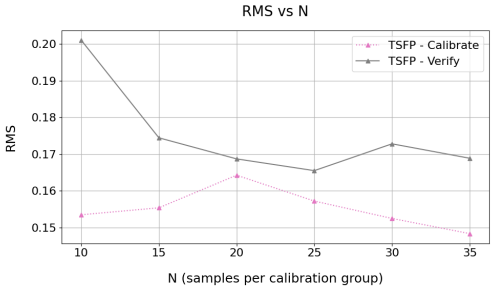
(d) Radial distortion coefficient, k_2 two-step calibration (camera 3)

Figure A.2: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_3_1`

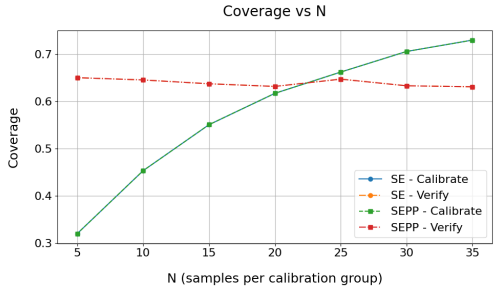
A.1.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 3)



(b) RMS reprojection error across N, two-step calibration (camera 3)

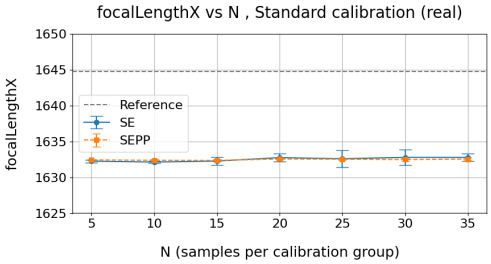


(c) Detection coverage across N, (camera 3)

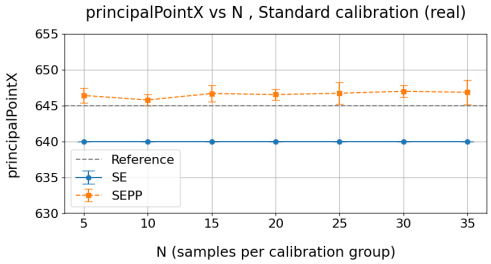
Figure A.3: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_3_1`

A.2 Real images, camera 4

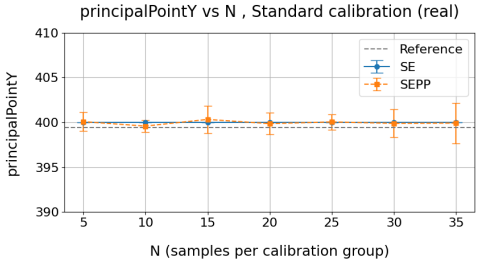
A.2.1 Focal length and principal point



(a) Focal length, f_x across N (camera 4)



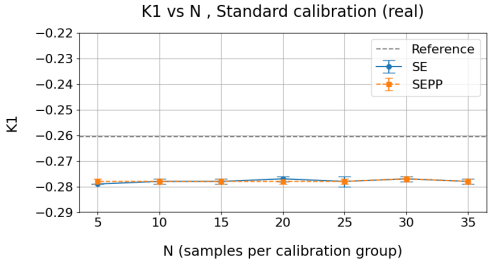
(b) Principal point x-coordinate, c_x across N (camera 4)



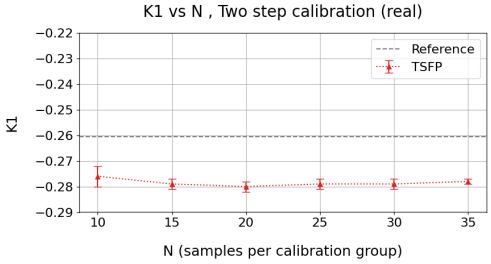
(c) Principal point y-coordinate, c_y across N (camera 4)

Figure A.4: Focal length and principal point across N for real images from camera `ID_4_1`

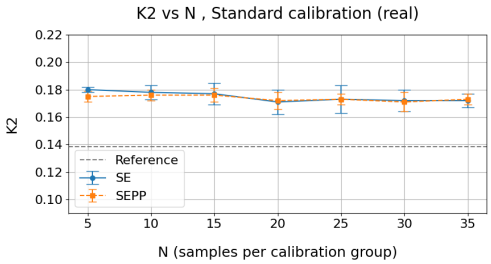
A.2.2 Distortion coefficients



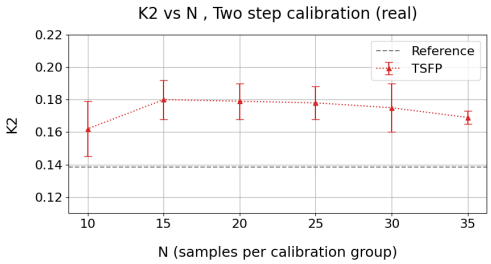
(a) Radial distortion coefficient, k_1 standard calibration (camera 4)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 4)



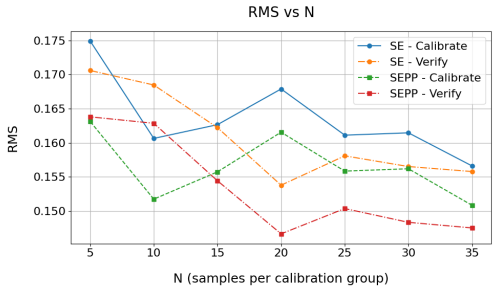
(c) Radial distortion coefficient, k_2 standard calibration (camera 4)



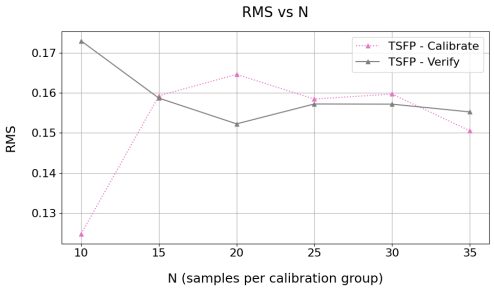
(d) Radial distortion coefficient, k_2 two-step calibration (camera 4)

Figure A.5: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_4_1`

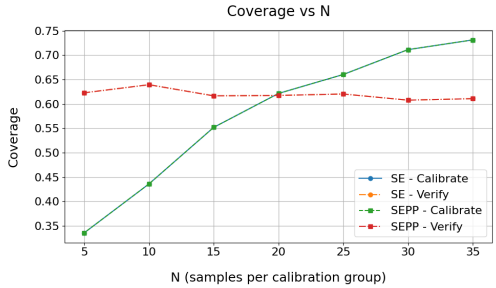
A.2.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 4)



(b) RMS reprojection error across N, two-step calibration (camera 4)

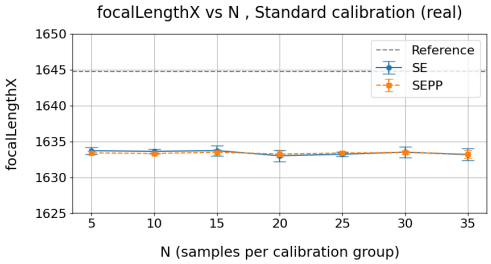


(c) Detection coverage across N, (camera 4)

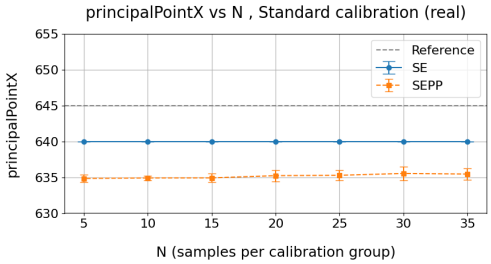
Figure A.6: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_4_1`

A.3 Real images, camera 5

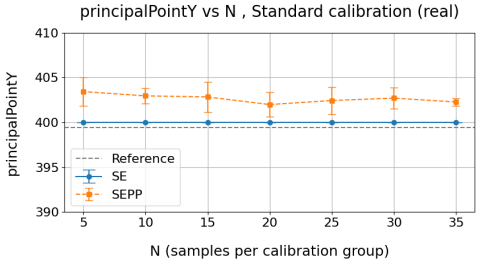
A.3.1 Focal length and principal point



(a) Focal length, f_x across N (camera 5)



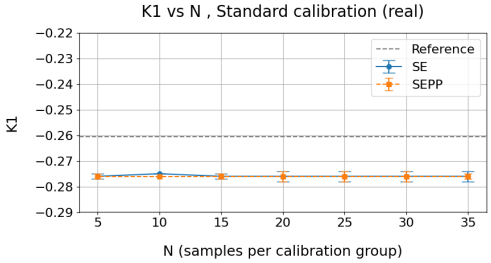
(b) Principal point x-coordinate, c_x across N (camera 5)



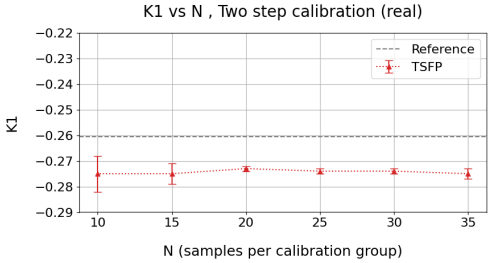
(c) Principal point y-coordinate, c_y across N (camera 5)

Figure A.7: Focal length and principal point across N for real images from camera `ID_5_1`

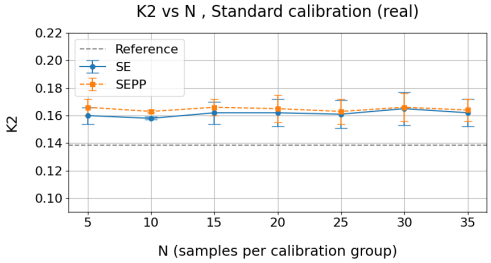
A.3.2 Distortion coefficients



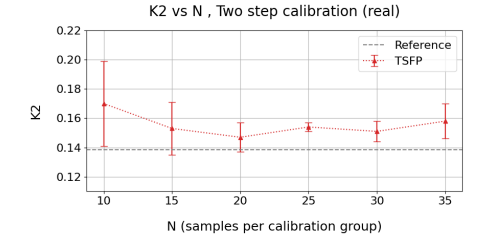
(a) Radial distortion coefficient, k_1 standard calibration (camera 5)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 5)



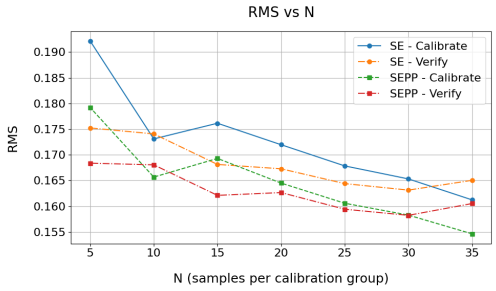
(c) Radial distortion coefficient, k_2 standard calibration (camera 5)



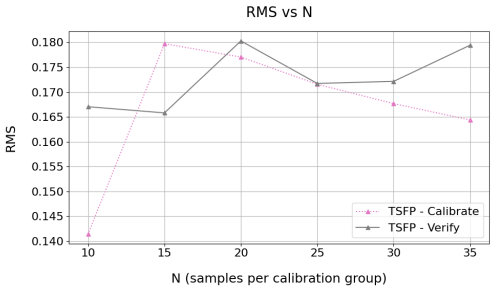
(d) Radial distortion coefficient, k_2 two-step calibration (camera 5)

Figure A.8: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_5_1`

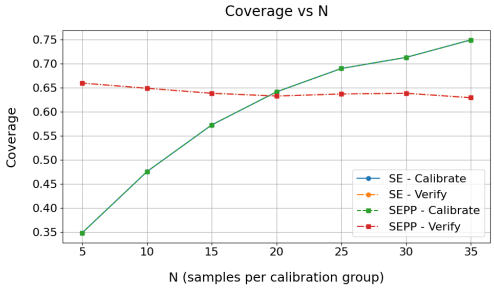
A.3.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 5)



(b) RMS reprojection error across N, two-step calibration (camera 5)

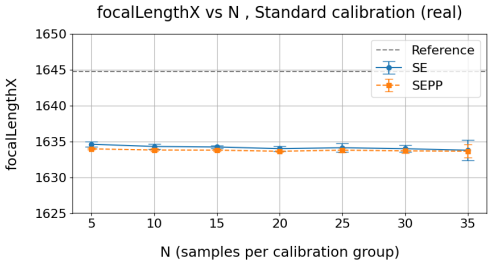


(c) Detection coverage across N, (camera 5)

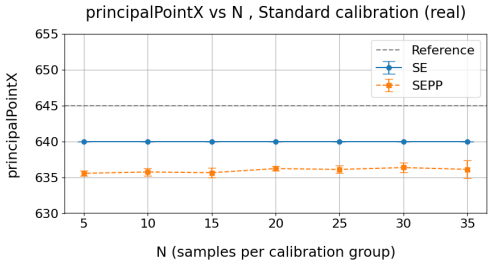
Figure A.9: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_5_1`

A.4 Real images, camera 6

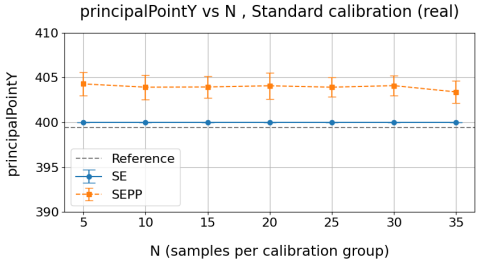
A.4.1 Focal length and principal point



(a) Focal length, f_x across N (camera 6)



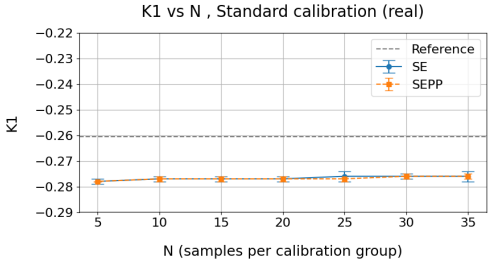
(b) Principal point x-coordinate, c_x across N (camera 6)



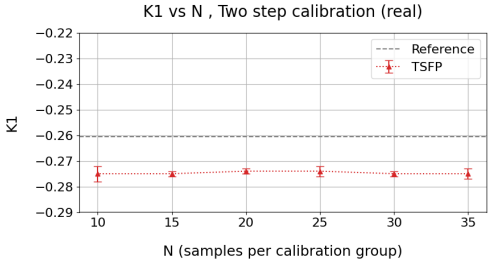
(c) Principal point y-coordinate, c_y across N (camera 6)

Figure A.10: Focal length and principal point across N for real images from camera `ID_5_2`

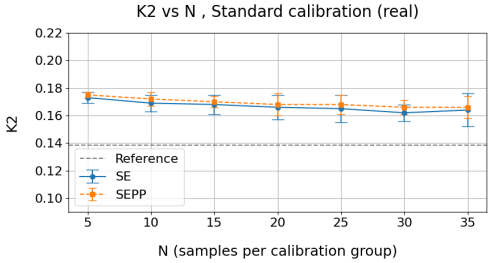
A.4.2 Distortion coefficients



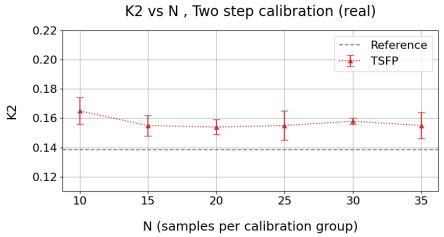
(a) Radial distortion coefficient, k_1 standard calibration (camera 6)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 6)



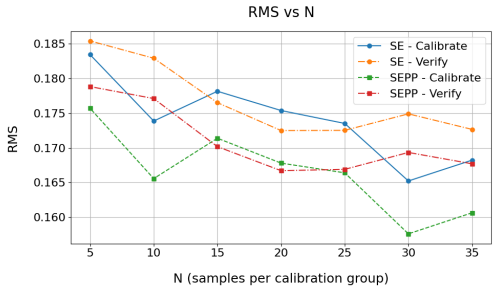
(c) Radial distortion coefficient, k_2 standard calibration (camera 6)



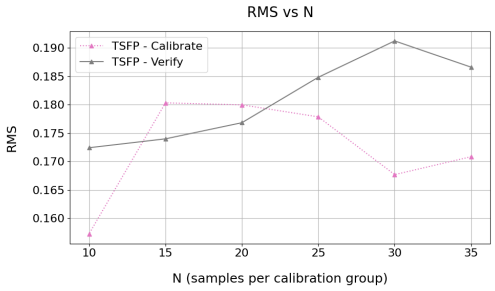
(d) Radial distortion coefficient, k_2 two-step calibration (camera 6)

Figure A.11: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_6_1`

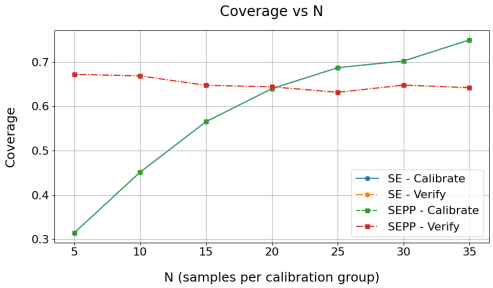
A.4.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 6)



(b) RMS reprojection error across N, two-step calibration (camera 6)

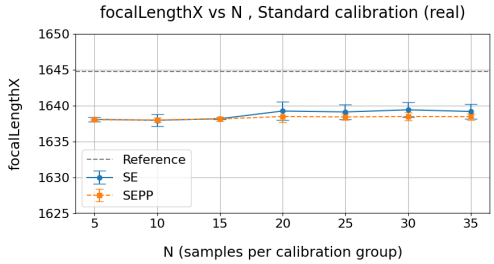


(c) Detection coverage across N, (camera 6)

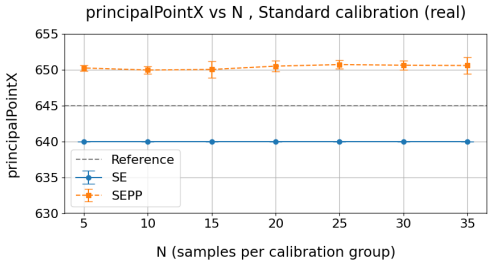
Figure A.12: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_5_2`

A.5 Real images, camera 7

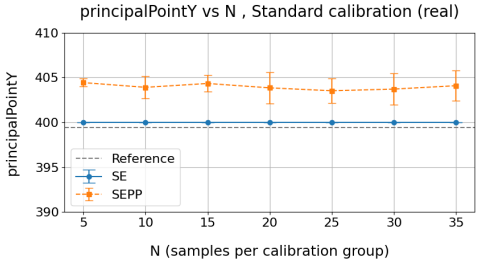
A.5.1 Focal length and principal point



(a) Focal length, f_x across N (camera 7)



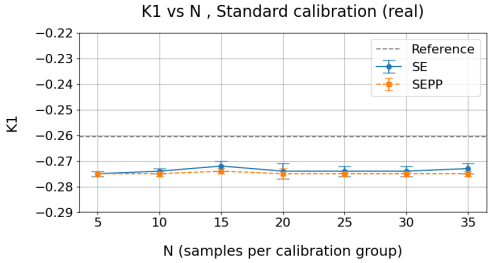
(b) Principal point x-coordinate, c_x across N (camera 7)



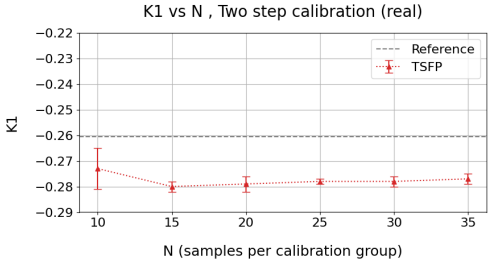
(c) Principal point y-coordinate, c_y across N (camera 7)

Figure A.13: Focal length and principal point across N for real images from camera `ID_6_1`

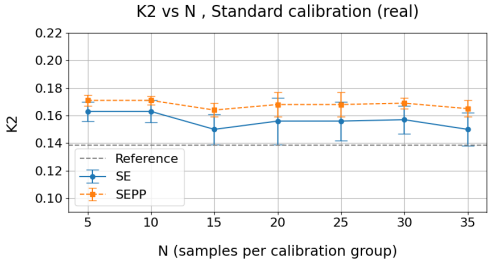
A.5.2 Distortion coefficients



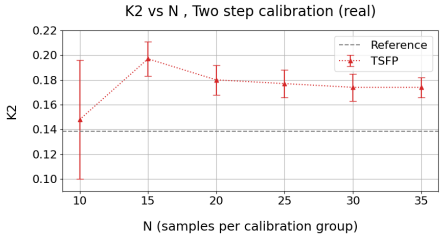
(a) Radial distortion coefficient, k_1 standard calibration (camera 7)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 7)



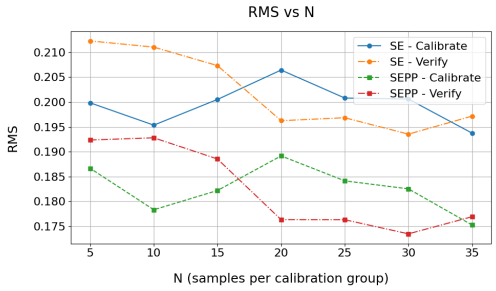
(c) Radial distortion coefficient, k_2 standard calibration (camera 7)



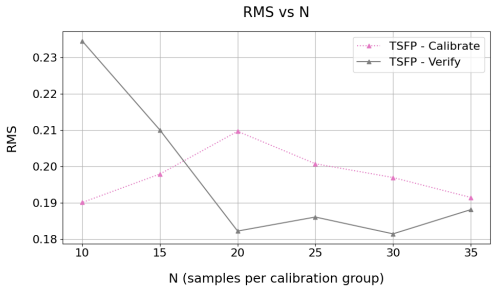
(d) Radial distortion coefficient, k_2 two-step calibration (camera 7)

Figure A.14: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_6_1`

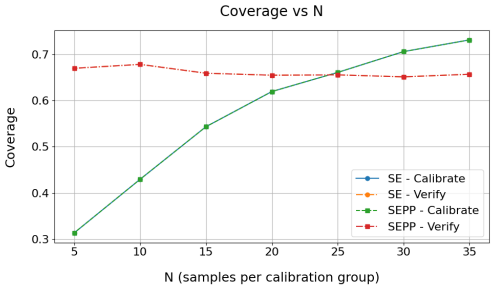
A.5.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 7)



(b) RMS reprojection error across N, two-step calibration (camera 7)

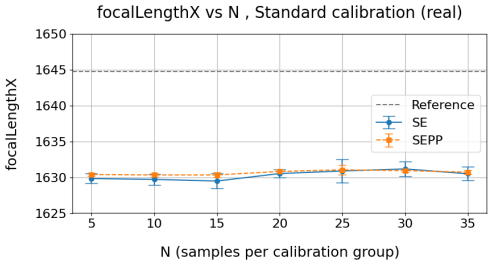


(c) Detection coverage across N, (camera 7)

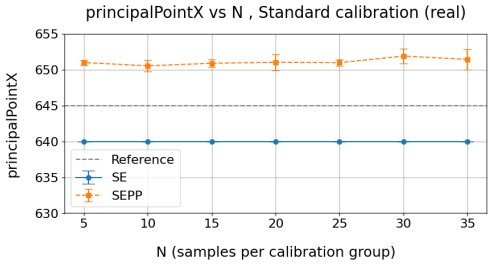
Figure A.15: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_6_1`

A.6 Real images, camera 8

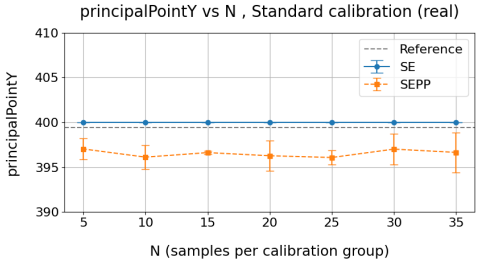
A.6.1 Focal length and principal point



(a) Focal length, f_x across N (camera 8)



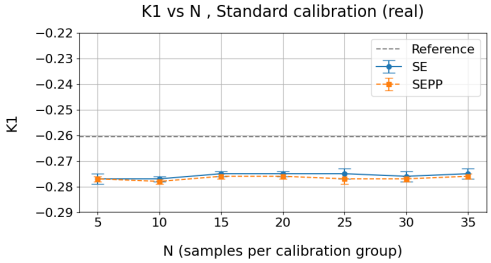
(b) Principal point x-coordinate, c_x across N (camera 8)



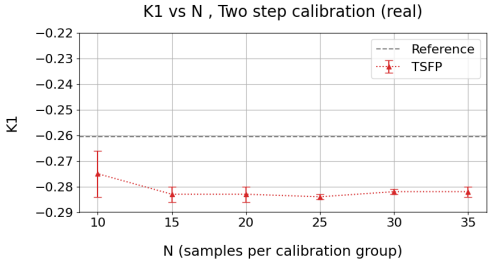
(c) Principal point y-coordinate, c_y across N (camera 8)

Figure A.16: Focal length and principal point across N for real images from camera `ID_7_1`

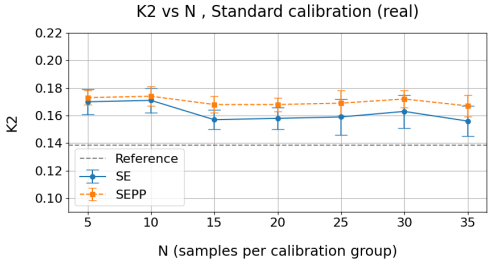
A.6.2 Distortion coefficients



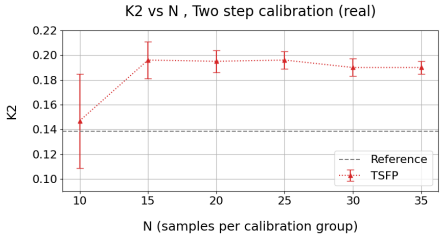
(a) Radial distortion coefficient, k_1 standard calibration (camera 8)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 8)



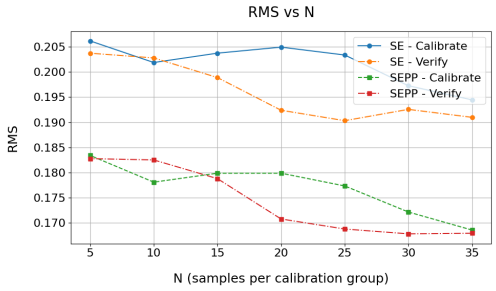
(c) Radial distortion coefficient, k_2 standard calibration (camera 8)



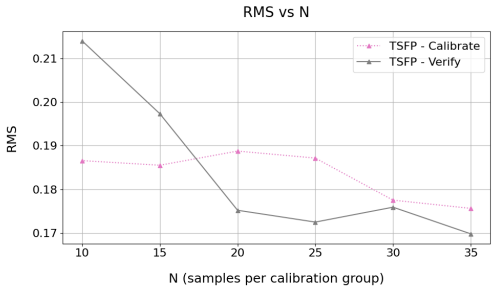
(d) Radial distortion coefficient, k_2 two-step calibration (camera 8)

Figure A.17: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_7_1`

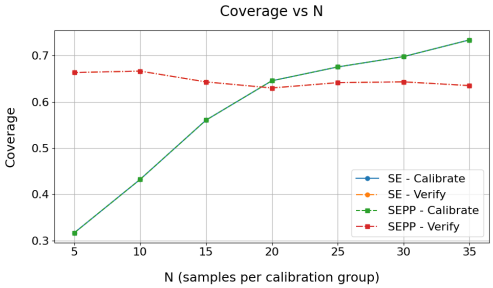
A.6.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 8)



(b) RMS reprojection error across N, two-step calibration (camera 8)

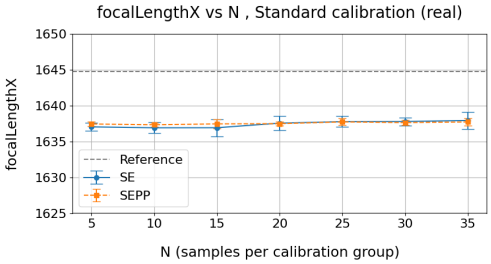


(c) Detection coverage across N, (camera 8)

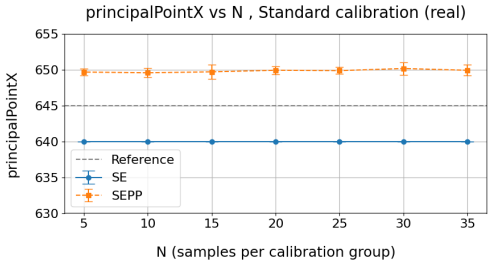
Figure A.18: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_7_1`

A.7 Real images, camera 9

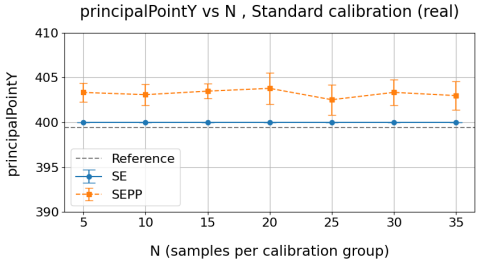
A.7.1 Focal length and principal point



(a) Focal length, f_x across N (camera 9)



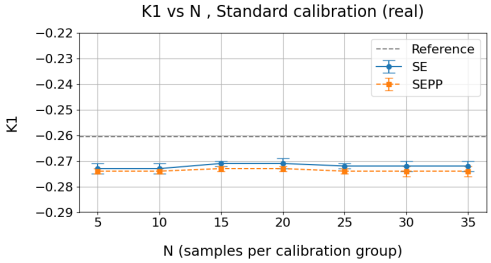
(b) Principal point x-coordinate, c_x across N (camera 9)



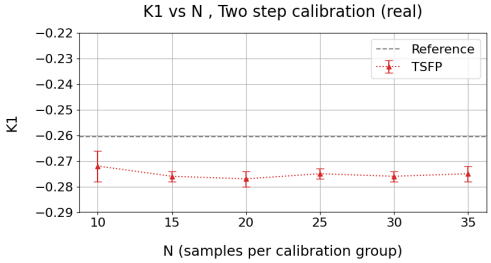
(c) Principal point y-coordinate, c_y across N (camera 9)

Figure A.19: Focal length and principal point across N for real images from camera `ID_8_1`

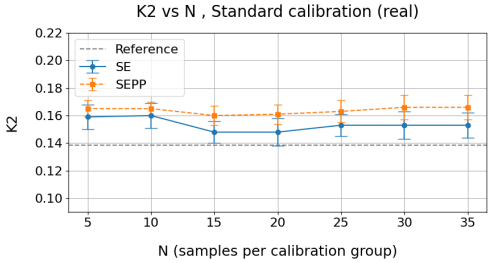
A.7.2 Distortion coefficients



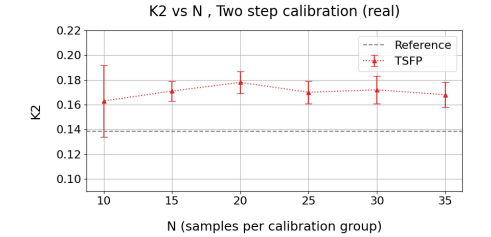
(a) Radial distortion coefficient, k_1 standard calibration (camera 9)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 9)



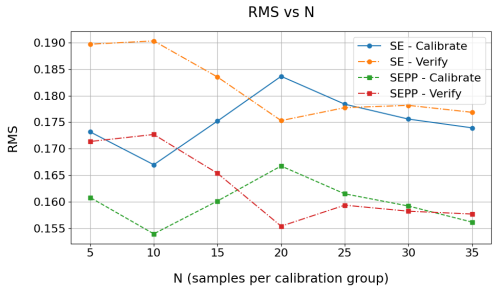
(c) Radial distortion coefficient, k_2 standard calibration (camera 9)



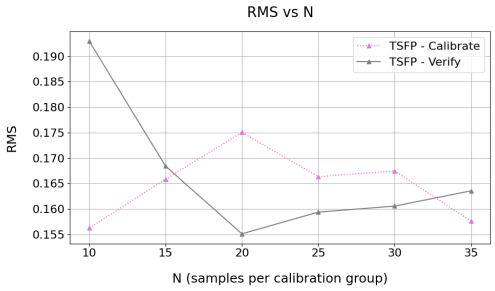
(d) Radial distortion coefficient, k_2 two-step calibration (camera 9)

Figure A.20: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_8_1`

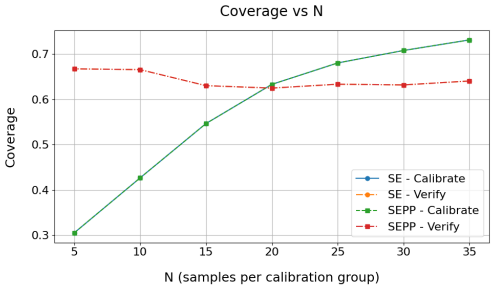
A.7.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 9)



(b) RMS reprojection error across N, two-step calibration (camera 9)

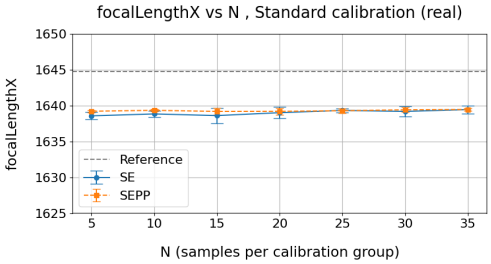


(c) Detection coverage across N, (camera 9)

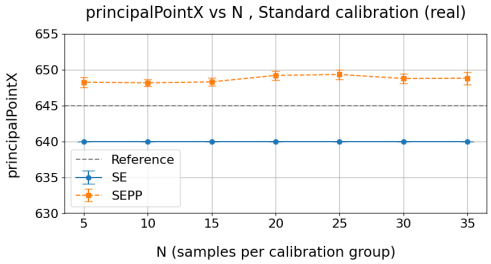
Figure A.21: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_8_1`

A.8 Real images, camera 10

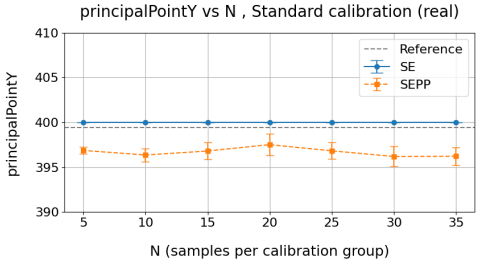
A.8.1 Focal length and principal point



(a) Focal length, f_x across N (camera 10)



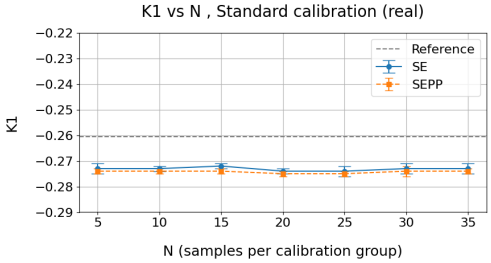
(b) Principal point x-coordinate, c_x across N (camera 10)



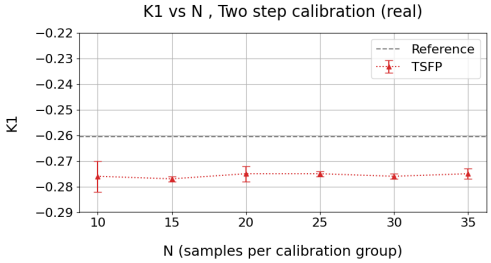
(c) Principal point y-coordinate, c_y across N (camera 10)

Figure A.22: Focal length and principal point across N for real images from camera `ID_9_1`

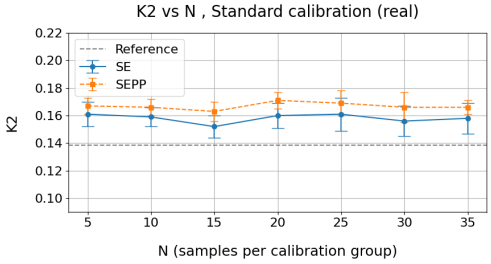
A.8.2 Distortion coefficients



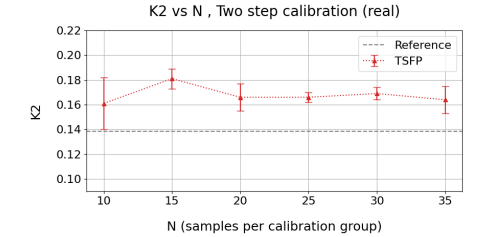
(a) Radial distortion coefficient, k_1 standard calibration (camera 10)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 10)



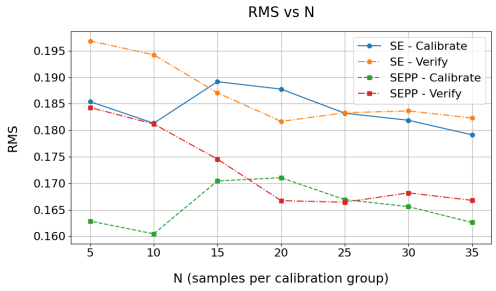
(c) Radial distortion coefficient, k_2 standard calibration (camera 10)



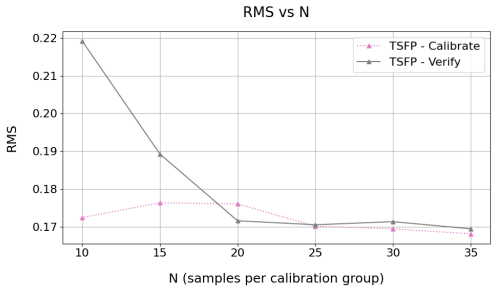
(d) Radial distortion coefficient, k_2 two-step calibration (camera 10)

Figure A.23: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_9_1`

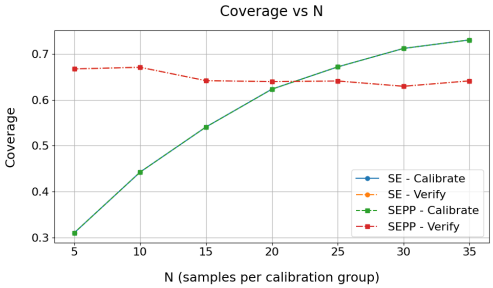
A.8.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 10)



(b) RMS reprojection error across N, two-step calibration (camera 10)

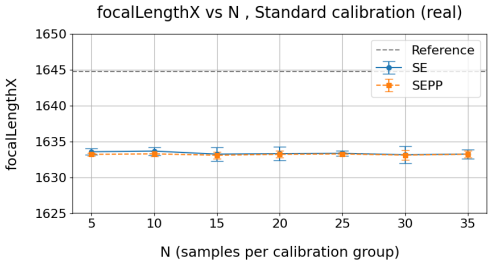


(c) Detection coverage across N, (camera 10)

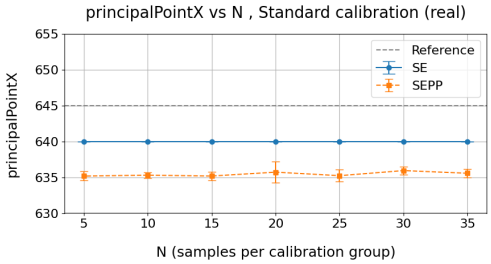
Figure A.24: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_9_1`

A.9 Real images, camera 11

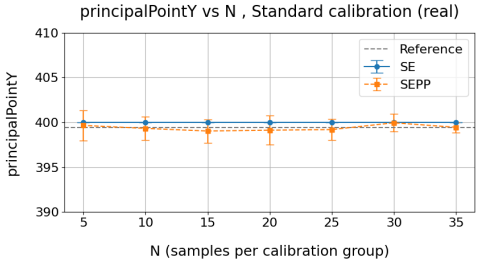
A.9.1 Focal length and principal point



(a) Focal length, f_x across N (camera 11)



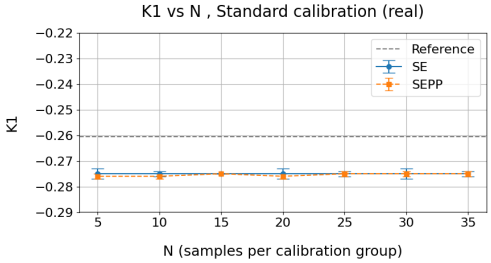
(b) Principal point x-coordinate, c_x across N (camera 11)



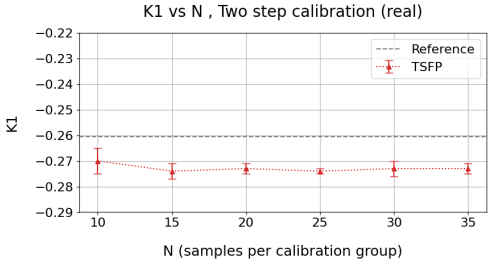
(c) Principal point y-coordinate, c_y across N (camera 11)

Figure A.25: Focal length and principal point across N for real images from camera `ID_10_1`

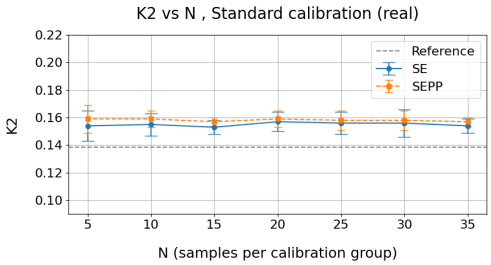
A.9.2 Distortion coefficients



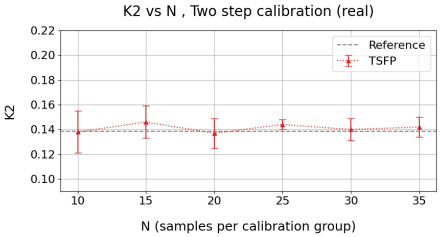
(a) Radial distortion coefficient, k_1 standard calibration (camera 11)



(b) Radial distortion coefficient, k_1 two-step calibration (camera 11)



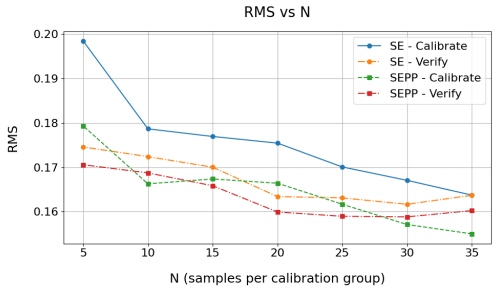
(c) Radial distortion coefficient, k_2 standard calibration (camera 11)



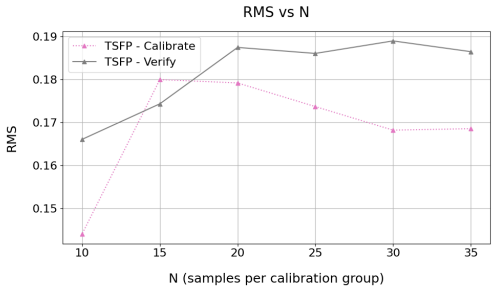
(d) Radial distortion coefficient, k_2 two-step calibration (camera 11)

Figure A.26: Radial distortion coefficients k_1 and k_2 for standard and two-step calibration on real images from camera `ID_10_1`

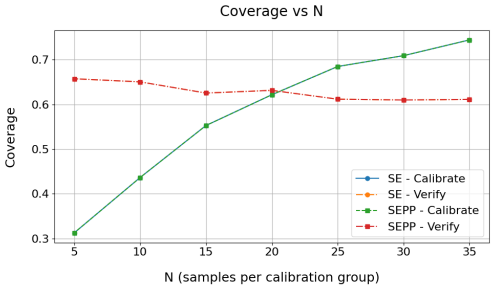
A.9.3 RMS reprojection error and detection coverage



(a) RMS reprojection error across N, standard calibration (camera 11)



(b) RMS reprojection error across N, two-step calibration (camera 11)



(c) Detection coverage across N, (camera 11)

Figure A.27: RMS reprojection error and detection coverage across N, for standard and two-step calibration on real images from camera `ID_10_1`

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY