PORTUGUESE AIR FORCE ACADEMY

Development of additional sensor capabilities for use in unmanned aerial vehicles under CIDIFA

Rafael Batalha Parcelas

Alferes Aluno Engenheiro Electrotécnico 138104-A

Thesis to obtain the Master of Science Degree in

Electrotechnic Engineering

Supervisor(s): Prof. Doutor Ricardo Adriano Ribeiro
Capt. Gonçalo Charters Santos Cruz

Examination Committee

Chairperson: Brigadier General José Augusto Nunes Vicente Passos Morgado
Supervisor: Doctor Ricardo Adriano Ribeiro
Member of the Committee: Doctor José António da Cruz Pinto Gaspar

Sintra, February 2018

To the Portuguese Air Force Academy...

Acknowledgments

I would like to express my gratitude to the Portuguese Air Force Academy for giving me the opportunity to pursue my academic studies. This prestigious institution supported all of my studies during the past six years and, during this dissertation provided the equipment, material, logistical and human conditions, helping me in the development of this research work.

This thesis would not have been possible without the support of many people. I would like to express my deepest gratitude to my supervisor Professor Ricardo Ribeiro for his guidance, support, encouragement, and patience throughout my research period and experimental results analysis. His insight and guidance have shaped this thesis, and i remain deeply grateful for all that he has taught me.

I also would like to thank my co-supervisor Capt. Gonçalo Cruz for his constant support, guidance, availability and constructive suggestions, which were determinant for the accomplishment and success of the work presented in this thesis.

Finally, would like to thank my girlfriend Ana Cláudia Lopes and my family, Domingos Parcelas, Amélia Batalha and João Duarte Parcelas, for all of their encouragement and support.

Resumo

Veículos aéreos não tripulados (VANTs) originalmente foram desenvolvidos para aplicações militares que, pela sua natureza não transportam operadores humanos a bordo. Esta dissertação foi realizada no seguimento do projeto Seagull. Este originou da necessidade da geração de um conhecimento situacional marítimo rigoroso e representa um dos esforços da Força Aérea Portuguesa para o cumprimento da sua missão, de acordo com os objetivos definidos pelo Conceito Estratégico de Defesa Nacional.

Esta tese tem por objectivo expandir as capacidades ja desenvolvidas no projeto Seagull. O trabalho apresentado foi realizado sobre a premissa que o ambiento marítimo é o principal cenário de operação. Assim sendo, um objetivo importante é a capacidade de localizar embarcações a partir dos dados visuais provenientes da aeronave.

Cameras equipadas com lentes de parâmetros variáveis revelam-se mais úteis do que as com lentes de parâmetros fixos. As lentes de parâmetros variáveis permitem a um operador obter imagens com melhor qualidade através de ajustes na configuração das lentes, contudo estas não são usualmente usadas em visão computacional devido a dificuldade inerente em modelar variações continuas das configurações da camera.

Esta tese apresenta uma metodologia para calibração de cameras com zoom variável e a sua utilização numa metodologia de localização geográfica. A metodologia de calibração de cameras apresentada é baseada na técnica de Zhang. Os modelos obtidos dos parâmetros intrínsecos da camera foram testados num equipamento com lentes de zoom numa experiência à escala reduzida do problema da localização geográfica. Utilizando a hipótese da terra-plana, o objetivo desta experiência foi validar tanto os modelos obtidos da calibração como a performance da metodologia para localização de alvos. Identificando a localização em pixels de um alvo junto com a posição e orientação da camera, as coordenadas de um alvo são determinadas no "referencial do mundo".

Uma experiência a escala real do problema de localização geográfica foi conduzida na Academia da Força Aérea Portuguesa, onde um conjunto de locais foram selecionados para testar a metodologia desenvolvida. O objetivo desta experiência foi avaliar o trabalho desenvolvido através da determinação das coordenadas geográficas de um alvo não cooperativo no solo sob condições reais. Através da comparação de resultados entre as estimativas providenciadas pela metodologia de localização geográfica proposta e as coordenadas GPS dos alvos foi possível identificar fatores e causas de erro possíveis de serem mitigadas.

Os resultados obtidos das experiências realizadas revelam uma interdependência entre os parâmetros intrínsecos da camera utilizada e o seu zoom. A metodologia de localização geográfica apresenta resultados promissores para ser utilizada em ambiente marítimo. Contudo, os erros experimentais observados foram categorizados em três fatores. O primeiro fator verificado foi as limitações dos modelos obtidos da calibração da camera. O segundo fator foi a diferença de alturas entre a altitude acima do solo da camera e a altitude acima do alvo. O terceiro e ultimo fator é a influência dos fatores ambientais no ruído da imagem e a sua sensibilidade à prespectiva da imagem para a captura do alvo.

Palavras-chave: calibração de câmeras, localização geográfica, VANTs

Abstract

Unmanned aerial vehicles (UAVs) were originally designed for military applications, which by nature, did not necessarily required human operators on-board. This thesis was developed following the Seagull Project, which represents one of the Portuguese Air Force attempts to fulfill its mission requirements in compliance with the guidelines of the Strategic National Defense Concept. Developed by the Portuguese Air Force Academy, it originated from a need to generate a more accurate maritime situation awareness of the Portuguese maritime territory.

This thesis aims to extend the capabilities previously developed in the Seagull project, which one of the main interests was the ability to geo-locate identified target vessels from the visual data of the aircraft, where the use of camera devices with variable-parameter lenses is more useful than those with fixed-parameter lenses. The variable-parameter lenses enable an operator to obtain better images by adjusting the camera's lenses to the present conditions of a scene. However, variable parameter lenses are not commonly used in computer vision because they are difficult to model for continuous variations of the lenses configuration.

This thesis presents a camera calibration methodology for devices with variable zoom and its employment in a vision-based target geo-location method. The camera calibration method presented in this work is based on Zhang's technique for camera calibration, were a model of the camera intrinsic parameters is obtained by exploring the interdependence between the camera zoom and its parameters.

A small-scale experiment of the geo-location problem was conducted in order to validate both the camera intrinsic parameters models and the proposed geo-location methodology. In this experiment by identifying the pixel location of a target in an image and the measurements of the camera position and pose, the world coordinates of the target are determined. Using the same methodology a full-scale experiment of the geo-location problem was devised. In this a set of location across the Portuguese Air Force Academy Campus were selected to employ the methodology devised. The objective of this experiment was to assess the accuracy of the work presented in this thesis under real-world conditions and the ability to localize an uncooperative ground target using the UAV vision sensor. By comparing the results of the target estimated position obtained from the geo-location methodology with the actual GPS coordinates of the selected locations to identify factors and/or error sources where identified, which can be mitigated.

The geo-location methodology developed is a proof of concept that empirically as shown encouraging promises to be employed in a maritime environment. The errors verified in the experimental procedures were mainly caused by three factors. First the precision of the camera variable-parameters models obtained, these are estimates of the real behavior of the lenses and as such are considered empirical approximations. Second the height difference between the camera altitude above ground and altitude above target, which is linked to the flat-earth hypothesis, as such errors can occur from the relative height difference. Third the environmental factors which induce image noise, which with the increasing obliquity of the captured image this errors can be exacerbated as demonstrated in this work.

Keywords: camera calibration; geo-location; variable-parameter lenses; UAVs.

x

Contents

List of Tables

List of Figures

Glossary

AGL   Above ground level.

AIS   Automatic Identification System.

CIAFA   Centro de Investigação da Academia da Força Aérea.

DEM   Digital Elevation Model.

DIAC   Dual image of the absolute conic.

ECEF   Earth Centered Earth Fixed.

ETP   Payload Ground Station.

GPS   Global Positioning System.

ISR/IST   Institute for Systems and Robotics/Instituto Superior Técnico.

Mavlink   Micro Air Vehicle Communication Protocol.

ROS   Robot Operating System.

SEC2   On-board Command and Control Unit.

SEP   On-board Payload Control Unit.

SFM   Structure from Motion.

UAV   Unmanned Aircraft Vehicle.

WSG84   World Geodetic System 1984.

Chapter 1

Introduction

1.1 Motivation

The Portuguese continental territory is located in the occidental periphery of Europe bordering Spain, covering an area of 89 000 Km^2. The Portuguese territory also includes the archipelagos of Madeira and Açores located in the North Atlantic with a combined 3000 Km^2 area [1]. In total, the country occupies a land area of 92 000 Km^2

Portugal has small land mass territory with scarce natural resources. However, considering its maritime territory, it is one of the largest maritime nations in the world. Several of the most important maritime shipping routes in the Atlantic ocean cross the Portuguese national waters, due its unique maritime geopolitical position of Portugal the nation faces multiple challenges in maintaining the control and surveillance of the national maritime space.

The considerable size of the Portuguese maritime territory, originates unique surveillance challenges which the Portuguese State is confronted. The main challenges are related with the fulfillment of its international obligations and responsibilities in its sovereign territory both in the maritime exclusive economic zone and the search and rescue area. These obligations include: oversight and control of maritime related activities, environmental control, search and rescue of shipwrecked vessels and medical support to their crews and operations in territorial waters.

The Portuguese Strategic National Defense Concept [2] establishes the guidelines that the government must address to fulfill the responsibilities in its sovereign maritime territory. The development of these guidelines into policies falls under institutions with proper jurisdiction. As such, the maritime territory security, safety and surveillance must be integrated in a broad scale perspective with all partners involved.

The Strategic National Defense Concept [2] establishes that the capability for proper surveillance and control of the maritime space must be maintained, with focus on employing the most cost effective means. This document also emphasizes the optimization and coordination between the national institutions; the responsible use of combat means in criminal activities; and the prevention and preparation of protocols in order to respond to natural disasters through an effective observation and alert system [2].

Up until now the Portuguese Air Force has been committed responding to these guidelines by using manned air platforms. However, unmanned aircraft platforms (in contrast to manned aircraft), characterized by their low costs in both maintenance and operation, emerge in this context as an appealing alternative to reduce both the man power employed; the platforms maintenance costs; and the aircraft acquisition costs.

The Seagull project was conducted to test and validate solutions to meet immediate and future needs for the management and information gathering of the maritime situation [3]. The Seagull project started in July 2013 and ended in July 2015. The project was led by Critical Software with the participation of other partners such as: CIAFA (Centro de Investigação da Academia da Força Aérea), ISR/IST (Institute for Systems and Robotics/Instituto Superior Técnico), FEUP (Faculdade de Engenharia da Universidade do Porto) and the Centro de Investigação Naval. The Seagull project aimed to develop an intelligent integrated system for UAV's and its purpose was to generate a more accurate maritime situation awareness knowledge base.

Maritime situation awareness is a large topic and the Seagull project addressed some of the points present in the Strategic National Defense Concept with concrete solutions such as: target detection, classification and tracking, identification of suspicious vessel behavior and monitoring environmental indicators [3]. In order to accomplish these objectives the airborne platform was equipped with an array of sensors and cameras in a multitude of light spectrums (infrared, multi-spectral, visible light). The dimension of this project provided an unique opportunity for the development of vision based algorithms an software architecture design [3]. This dissertation is aligned with the continuity of the project. By working within the developed software and with the already employed sensors in an attempt to expand and properly validate the concepts supporting the extended capabilities of the airborne platform.

1.2 Problem Formulation

One of the primary application of UAVs has been short range aerial reconnaissance, where they have been primarily responsible for collecting information and relaying it to an external operator on the ground. The employment of UAV platforms in operational scenarios is an increasing reality and the development of the Seagull project represents one of the Portuguese Air Force attempts to develop new technologies and capabilities usable in the organization daily operations.

One important sensor that many UAVs have is a camera devise. With a camera, a vehicle can obtain a a lot of information about its surroundings and possible targets of interest. In the Seagull project, one of the most important topics developed was the capability of detecting targets in maritime surveillance environment, specifically the detection of vessels using an optical color sensor mounted on an airborne platform. This topic represents a very complex problem due to the maritime environment where the detection algorithm must be able to deal with variability of the vessels appearance, wave crests, presence of sun glare, change in the lighting conditions, clouds and platform movement [4].

The extension of this capability is the geo-location of vessels once they are identified in an image. Geo-location is defined as the identification or estimation of the real-world geographic location of

an object. This information allows the national institutions to take immediate actions in related to the surveillance and control of the maritime scenario.

In this work, object recognition was not a major concern, rather the focus was to find a target real world coordinates with a variable-parameters lenses camera in order to test a geo-location proof of concept methodology. The solution proposed was designed in order to work as independently as possible from the object recognition algorithm and other systems of detection.

1.2.1 Problem Description

The problem presented in this thesis consists of obtaining real world coordinates (latitude, longitude and altitude) of a target identified in a image. To accomplish this goal, several challenges arise as result of its complexity. The first challenge is the mathematical characterization and modeling of the camera equipment, where a calibration procedure must be employed. The second challenge is evaluating the correctness of the derived models from calibration.

Once a target is identified in an image and the camera device that captured such image is correctly modeled the third challenge is the depth or range estimation of the target regarding the airborne platform. Monocular cameras such as the one used in this work, do not provide this information therefore, an alternative solution must be employed. All of these challenges are part of the vision based geo-location problem proposed in this thesis. In order to overcome them, several objectives were outlined.

1.2.2 Objectives

The research presented in this thesis attempts to fulfill the following 5 objectives:

1. The development of a method to calibrate cameras with variable settings;

2. Modulation of camera intrinsic parameters by the zoom of the camera's lens;

3. The development of a geo-location algorithm;

4. The analysis of the camera calibration model and geo-location algorithm in experimental and real world scenario;

5. Integration with the software developed in the Seagull Software Architecture;

1.3 Literature Review

1.3.1 Camera Calibration

A camera is a device designed for capturing a photographic image or recording a video (stream of images). The calibration of such devices is a process from which it is possible to obtain the camera intrinsic parameters (focal length, lens distortion, optical image center, model parameters) and extrinsic parameters (location and pose). The camera intrinsic parameters are unique to each and every single device and do not change when the camera is moved and its lens configuration is not altered.

The state-of-the-art camera calibration techniques solve the intrinsic and extrinsic camera parameters using a physical calibration pattern (such as a planar checkerboard). Several images from the pattern are captured in different poses, and the camera model parameters are optimized in order to correspond image coordinates to known coordinates of the calibration pattern.

The state-of-the-art methods are Tsai's [5], Heikkilä's [6] and Zhang's [7] methods. These methods mainly differ in how the camera intrinsic parameters are determined and how the the calibration patterns are use .

Salvi [8] traces a comparison of the calibration methods developed between 1982 and 1998 and Tsai's method, shows the best performance despite requiring a higher precision in the input data. The widely used method proposed by Tsai, which is based on a two-step technique modeling only radial lens distortion are considered [5]. This method represents a conventional approach that is based on the radial alignment constraint and requires accurate 3D world-referenced coordinates measurement with respect to a fixed reference (planar pattern).

Heikkilä's method, is also world-reference based, although not included in Salvi's work, employs the more general direct linear transformation technique by making use of the prior knowledge of intrinsic parameters. It also involves a more complete camera model regarding the camera lenses distortions[6].

Zhang's method represents a new leap in camera calibration techniques. This method is a different special case of Heikkilä's formulation, where it combines the benefits of world-reference based and auto-calibration approaches. This enables the linear estimation of all intrinsic parameters.

Auto-calibration or self-calibration approaches determine camera parameters directly from multiple non-calibrated views of a scene by matching corresponding features between views, despite unknown camera motion and even changes in some of the intrinsic parameters. This method introduces a flexible approach in which either the camera or the planar pattern can be moved freely and the calibration procedure is easily repeatable without redoing any measurements.

By examining both Tsai's and Zhang's methods, on the one hand Tsai's reveals to be more precise, if the input data is not corrupted however, very accurate measurements of the calibration pattern are needed. On the other hand Zhang's method does not require neither special measurements of the planar pattern nor from the experimental setup. According to Sun [9] better results are obtain with Zhang's method and the algorithm sensibility to error in the measurements can be improved by increasing the number of corners in the checker pattern used.

Taking in account the results from Salvi [8] and Sun [9], Zhang's method is stated as a state-of-the-

art reference for camera calibration. A version of this method uses a pattern of circles to avoid errors in the coordinates measurements in the pattern although there is not a consensus if it improves the results from a checkerboard pattern.

1.3.2 Geo-location

Vision-based geo-location is an extensively explored topic and has been a well documented problem. From the several approaches developed in the academic community there are four different types which are important to review[10]:

1. The Paper-Map-Based Method

2. Line-of-sight filtering

3. Structure From Motion (SFM)

4. Geo-referenced imagery

When using airborne platforms or UAV's the estimation of the target ground altitude or the target range to the aircraft, is an essential part for solving the ground object geo-location problem. The payload restrictions in this type of platforms represents a major problem which can make this task challenging. Given the fact that this type of platforms cannot carry a heavy range sensor capable of measuring distances of hundreds of meters, most of the methods rely on assuming a flat earth model. This assumption implies that there is no height difference at a ground level between the aircraft and the target locations. In a maritime environment this is a relatively safe assumption because either a GPS receiver or a calibrated barometric sensor respectively assume a mean sea level altitude or an average sea level pressure to operate. In this work the mean sea level corresponds to the ground altitude of the target.

The Paper-Map-Based Method

This is most straightforward targeting method and, assumes that the target is centered in a camera image, and the target geographic location is determined as the intersection between the camera pointing axis and the ground plane[10].

The camera location and attitude must be known and the target must be in line of sight. In this method a Digital Elevation Model (DEM) of the terrain or the flat world assumption hypothesis must be employed for the calculation. This method accuracy relies strongly on the precision of the measurements of the aircraft pitch, roll and heading, and the camera tilt and pan as well as the camera calibration. Errors can also arise from the inaccuracy of the DEM as well as sensor bias and noise. Typical sensors installed on most UAV platforms have low accuracy due to limited payload capabilities. This can originate errors from a single target measurement sample from 20-40 meters with the platform flying at 100 meters [10, 11].

The systematic errors verified are usually biases in the estimation of the camera orientation. They can be the consequence of biases in the UAV attitude and orientation, poor calibration of the magnetic compass sensor or the misalignment of the several inertial measurement units in either the camera unit

or the aircraft. These errors can be mitigated my selecting a flight path that enables more accurate target position estimates.

In [11] it is shown that by flying in a circular loitering pattern and applying a least square filter to multiple observations of the target, the errors from the geo-location decrease from 20-40 meters to less than 5 meters. The flat world assumption was employed in the experiment and the results were obtained using offline flight data. In [12], results show an approximate 4 meters error of the target location, These results were obtained by using a combination of a DEM model of the flight area and a target vector towards a target of interest in the image frame.

Line-of-sight Filtering

This method is similar to the Paper-Map-Based Method however, it does not rely in a DEM or the flat earth hypothesis. The idea behind this method is either to find the intersection point of multiple rays from multiple observations passing through the camera and the target or using stereo cameras. This intersection identifies the 3D coordinates of the target.

Due to sensor bias the rays might never produce a point of intersection and therefore the problem is to identify the closest point between such rays. A Kalman filter can be used to remove possible sensor noise and bias but multiple samples of the target are still required. In [10], this method is implemented and produces a targeting accuracy of 10 meters at an altitude of 100 meters. These results were obtained by using a Raven-B platform.

Structure From Motion (SFM)

The principal source of uncertainty in the line-of-sight filtering method is likely to be an error in the supposed orientation of the vehicle and thus the camera. As an example, at an altitude of 100 meters with a camera pitched down 30º, a wind-induced 1.5º pitch error produces 10 meter of targeting error, and even if this is zero-mean noise, it will damp slowly over many frames of filtering [10]. This method relies on the 2D image coordinates of multiple targets seen in images taken from multiple camera positions and solves for the 3D locations of each target. The method can be simplified and use fewer targets and/or images if some of the 3D positions or orientations are given as input [10]. The mathematical premise of SFM is described in detail in[13, 14].

Geo-referenced imagery

This method consists in registering a on board video and compare it to geo-referenced images, thus extracting the target coordinates directly from the referenced image. This method is independent of the camera pose accuracy.

The sensor information on board can be used for a first estimate of the target position, but the final position is computed after the registration process. This registration process is a limitation of this technique, as well as its dependence on the availability of a reference imagery for a specific region. On

the other hand, the rapid development of imagery tools such as Google Earth makes this method very promising.

These methods have been tested in [15, 16] on large platforms with extremely accurate instrumentation and have demonstrated a high degree of accuracy. However, these methods have yet to be tested on smaller platforms with their relatively inaccurate instrumentation. Furthermore, these methods require a known map of the world with reference imagery precisely aligned to geo-coordinates. This method and his constraint are not detailed in the present document.

1.4   Proposed Approach

The proposed approach in this thesis for the geo-location of vessels in a maritime environment is separated into four different steps.

The first step is the characterization of a camera intrinsic parameters behavior in its zoom range. To accomplish this the OpenCv tool box was used, which relies on Zang's [7] camera calibration technique, in which the camera lenses configuration are not altered during the procedure. In this thesis the used optical sensor is a gimbal camera, further detailed in the next chapter. Gimbal cameras inherently can change their settings and therefore it is proposed a method to calibrate the camera by changing the lenses configuration and determining a the intrinsic parameters each time a change occurs. This modeling approach provides a way to define the camera parameters during its operation.

The second step is to implement a geo-location algorithm based on the Paper-Map-Based method with the assumption of the flat world model. This represents the most reasonable solution, due to the scenario of operation (maritime environment), having a accurate DEM or applying the Geo-Referenced Imagery method reveals to be impractical because of the environment ever changing nature and the impossibility of mapping the sea surface accurately. The Structure From Motion method is not suitable, because this method expects that the airborne platform flies at a considerable variable distances from the target vessel. In this situation being able to acquire several images of a target in several significantly different positions is not possible. In the context of this work, obtaining a geo-location solution based in a single sampled image is more relevant.

The third step is to merge Zang's camera calibration methodology and the geo-location Paper-Map-Based method in laboratory simulations with targets in known locations and compare results with the locations provided by the proposed methodology providing an assessment of the concept devised in this document.

The fourth and final step is a full scenery simulation in order to validate that the designed algorithm and camera calibration provide a desirable solution to the proposed problem. This simulation is conducted under the expected circumstances of the aircraft operation, however some flight parameters of this experiment were purposely maintained constant in order to ensure a controlled scenario.

1.5   Thesis Outline

This thesis is organized as follows: Chapter two describes the developments of the Seagull project and its pertinent software architecture. It also establishes the necessary computer vision concepts and notations involved in the camera calibration based on the textbook of Hartley and Zisserman [13]. Chapter three details the method used to geo-locate targets detected in an image. This chapter details the camera calibration and model used as well as its integration with the coordinate frames and coordinate transformation matrices used throughout this work. Chapter four details the experimental procedures conducted to verify the proposed methodology and their results. In this chapter, three different experiments were conducted. The first to determine a model as a function of its settings, the second to access the validity of the proposed method under the flat world assumption and the third is a full scale experiment under real world conditions and constraints. Chapter five summarizes the results and achievements of this research and offers suggestions for future work.

Chapter 2

Background

This chapter describes the developments of the Seagull project and details the relevant theoretical background in computer vision concerning the research presented in this thesis by establishing the concepts, later used and developed in the following chapters.

Section 2.1 briefly describes the UAV platform used and provides a summary description of the technical reports from the Seagull project [3, 17, 18, 19]. This summary highlights the relevant contents concerning this dissertation, where for further technical information the original documents should be consulted.

Section 2.2 describes the software architecture used during the experimental procedures which defers from the one used in the Seagull project.This is a temporary architecture design, with the objective of not disturbing the existing code of the Seagull project. This is latter removed and a geo-location module integrated in its place.

Section 2.3 establishes the computer vision concepts and notations used. It provides the reader with a knowledge base of the subject to further understand the concepts presented in the following chapters. The material in this section is originally based on the textbook of Hartley and Zisserman [13] and thus, the original text should be consulted if a more detailed description of algebraic projective geometry tools and computer vision algorithms is needed.

2.1   Seagull Project

The Seagull project attempted to develop efficient solutions to increase and answer to the maritime situation awareness challenges. The objective of this project was to create intelligent on-board systems to implement in UAVs. The project main contributions were to: detect, identify, classify and follow possible targets of interest in a maritime environment. This contributions can be integrated in three lines of action: search and rescue, environmental monitoring and maritime oversight which translate in the following operational requirements:

- Provide early warnings and identification of risks and threats in a maritime environment;

- Detection of non-conformity vessel behavior, illegal acts and practices;

- Intercept contacts of interest;

- Collect essential elements of analysis which may influence the subsequent actions that can be projected into events that occur in coastal areas.

The UAV platform used in the Seagull Project was the class I - Extend UAV aircraft with a wingspan of 3.5 meters and 1.9 meters in length. Figure 2.1 presents the technical dimensions of the platform.
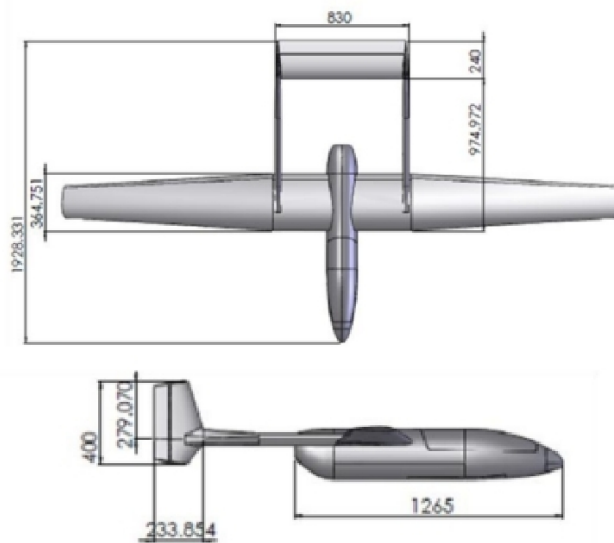
Figure 2.1: Extend UAV platform.

The aircraft's central body component includes the navigation and flight control system, mission payload, landing gear, tail supports and engine firewall. The wing is 1.3 meters in length with a taper ratio of 0.54. The wing root profile is the FX63137 and the tip profile SD7090. The maximum take-off weight is 25 kilograms, 12 of those are accounted by aircraft structure, propulsive system and servo actuators of the control surfaces.
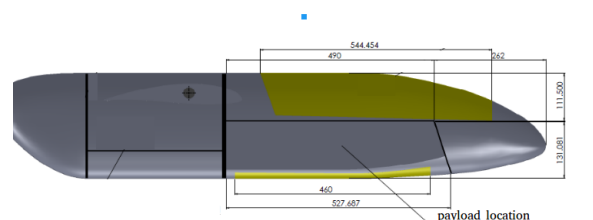
Figure 2.2: Payload location in Extend platform.

Figure 2.2 illustrates the payload location of the platform. The payload equipment used in this document is the TASE 150 gimbal camera manufactured by the Cloud Cap Technology corporation. The

gimbal camera itself encapsulates an array of accelerometers, gyroscopic sensors and a GPS receiver which establish the camera pose, orientation and position.

The TASE 150 is a compact, light weight, low cost daylight gimbal camera payload used in the Seagull project. It is a flexible light wight payload equipment with operator friendly software interfaces. It uses a Sony FCB-EX1000 camera with a 1/4-type Exview Hole Accumulated Diode Charge-Coupled Device sensor.

(a) TASE 150 Gimbal.   (b) Sony FCB-EX1000 camera.

Figure 2.3: Gimbal Camera Payload.

2.1.1 Systems Architecture

In recent years, the on growing development of digital technologies has enabled the rise of new and improved applications. Open-source software (software with source code that anyone can inspect, modify, and enhance) in the context of the new digital technologies has been explored as an option over more traditional solutions. It has lower costs and, in most cases, is only a fraction of the cost of their proprietary counterparts and, is more secure as the code is accessible to everyone and is independent on the company or author that originally created it. Lastly, it uses open standards accessible to everyone, thus eliminate the problem of incompatible formats that exist in proprietary software.

Open software communication standards, when correctly employed, are less likely to be obsolete and, when this occurs, the established openness ensures that the current data and systems can be freely transformed and migrated to updated software systems. The Seagull project was developed using Open-source software in order to maximize systems interchangeability, vendor independence, efficient resource usage, flexibility, and better quality.

When the Seagull software architecture was developed, one of the main concerns was ensure a low level integration between the several system functions. The software architecture is structured in several modules as independent from each other as possible. The main modules are responsible for the aircraft command and control and for the payload control.

Although there is always a residual dependency between the systems, in terms of software architecture, the idea was to develop a system that is aircraft agnostic. This led to the development of a

modular software structure able to satisfy future needs specifically through the easy integration of new sensors and new software features between the operator and the payload requiring fewer modifications as possible to the specific modules.

2.1.2 Physical System Overview

The complete system setup used during the Seagull project includes several computers, were the three main units are: the Piccolo autopilot unit, the on-board command and control unit (SEC2) and the on-board payload control unit (SEP). The ground station computers units are: the Piccolo ground station and the payload ground station(ETP). In the end there is and external system which monitors the operations. This integration of systems and the communication protocols between is illustrated in Figure 2.4.

The Piccolo unit is the autopilot unit of the UAV platform. It directly commands the aircraft control surfaces, provides navigation telemetry to other systems and manages the communication channel with the ground station. The connection between the Piccolo and other systems is ensured via a serial port.

The SEC2 is responsible for the connection between all other Seagull systems and the Piccolo unit. This computer is also responsible for all the command and control tasks related to the sense and avoid (S&A) context.

The on-board payload system computer (SEP) contains all features regarding the UAV payload control, as such it is connected to the array of sensors and equipment in the payload and is connected via Ethernet to the SEC2 unit. Given the nature of the computer vision algorithms employed, the SEP computer has a higher performance requisites possessing a Graphics Processing Unit and compatible with Open Computing Language.

The Piccolo Ground Station is a equipment where there the computer used by the UAV operator on the ground is connected, providing a operation interface to the user. It is connected to the antennas used for communication with the aircraft.

The Payload Ground Station (ETP) is a computer used by the aircraft payload specialist. This computer is responsible for the operation of the payload equipment, data acquisition and transfer regarding the specific payload employed and payload configuration.
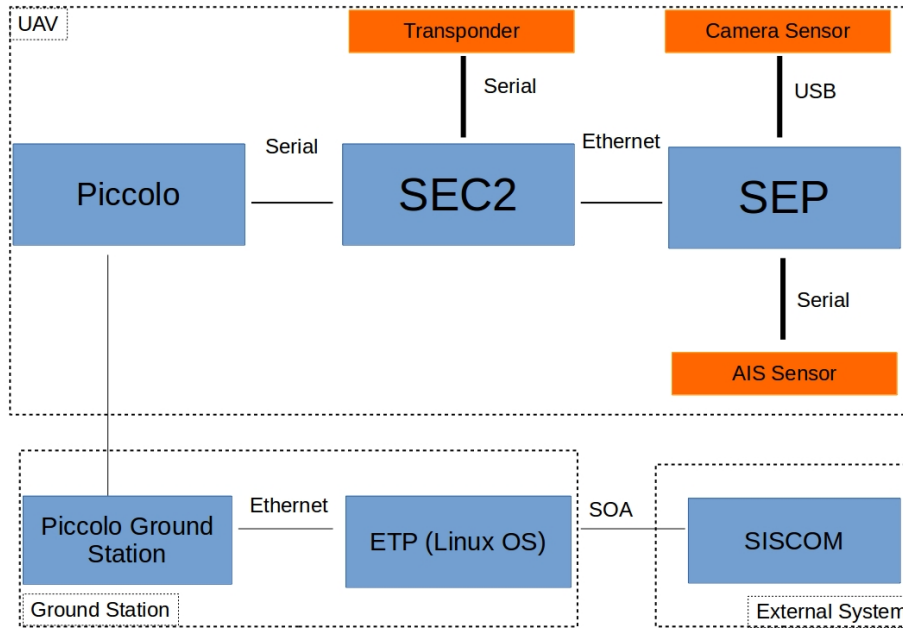
Figure 2.4: Complete Seagull Systems (adapted from Figure 3 in [19]).

2.1.3 On-Board Software Architecture Overview

The Robot Operating System (ROS) is a flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that simplify the task of creating complex and robust robot behaviors across a wide variety of robotic platforms [20]. The software architecture between the SEC2 and SEP computers uses this middleware to ensure the data communication between the modules of each computer, as seen in Figure 2.5.
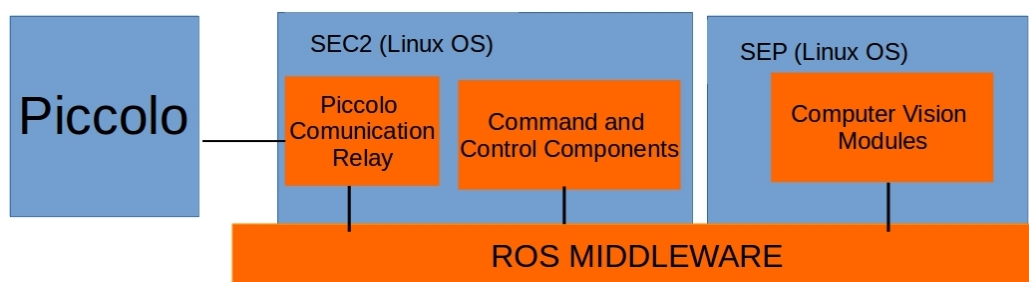
Figure 2.5: Software Architecture of Seagull (adapted from Figure 4 in [19]).

In the context of the Seagull architecture the ROS asynchronous communication is one of the most important feature. This type of communication allows different processes or programs to publish messages on a ROS topic. Such messages are then read by other processes which have a subscription to said topic.

This type of communication allows different programs to share information. ROS topics can be described as a unilateral channel of asynchronous communication without limit of publishers or subscribers.

13

In practice this type of communication can be seen as a data bus where programs subscribed to a topic can read published messages and publisher programs can write messages to the topics.

This mechanism of communication is transparent for all computers in a network and provides the backbone of the communication between the SEC2 and SEP computers, eliminating the need for any other communication mechanism. The Ethernet connection between SEC2 and SEP is predictable and efficient. The ROS communication protocol provides a solution that is both versatile and allows the development and integration of other processes in the architecture.

The ROS middleware has two other major features: the rosbag and the ability to integrate multiple computer languages. The first allows the saving of ROS messages in memory during the execution of a mission. Later this messages can be replayed and used to recreate the mission and studied to access the processes performance and the output results [19]. The second allows to integrate multiple computer languages. As long as they share information about the messages structure, multiple programs or processes in different languages can communicate.

2.1.4 On-Board Command and Control System

The on-board Command and Control System (SEC2) uses a Linux Operating System. It contains a driver responsible for the interactions with the autopilot, communication relay and control supervisor logic units all connected via ROS middleware, as presented in Figure 2.6.
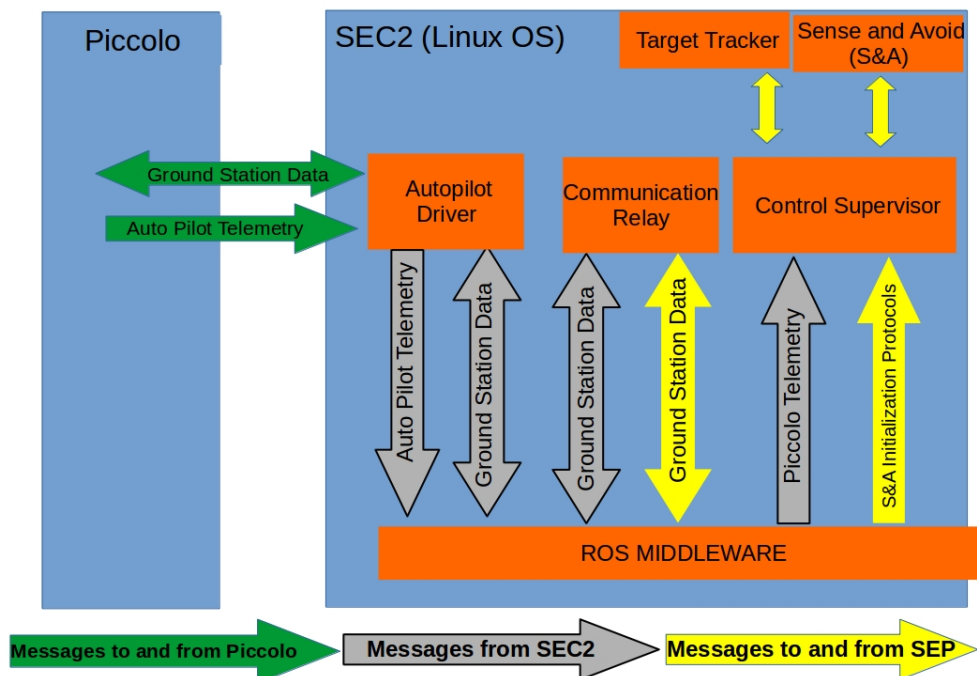
Figure 2.6: On-Board Command and Control Architecture (adapted from Figure 6 in [19]).

The Autopilot Driver connects the Piccolo autopilot and the SEC2 system. It is the only component which directly communicates with the Piccolo autopilot with knowledge of its specific Application Pro-

14

gramming Interface. By using a single software component to communicate between the Piccolo and the SEC2 the on-board software is independent of the autopilot driver. In case of modifications in the software architecture the only component to replace would be the Piccolo unit. This modular aggregation allows for a single device to provide navigation telemetry (e.g. GPS positioning, aircraft heading) and to receive or send data trough the "payload stream" message made available by the Piccolo.

The Communication Relay module, is tasked with managing the communication logic to and from the UAV platform. The module ensures that all communication needs of the platform are meet (in cases where the Piccolo does not provide) specifically the insurance of message deliveries, the sending of large data packages and the correction of the data. This module receives and streamlines the information as an array of bytes being completely agnostic in regard to the messages contents. This type of communication is particularly useful due to its generalization capability where any kind of data can be transmitted without revealing neither the channel characteristics nor the protocol used to communicate.

When a message which requires acknowledgment of reception is sent, the module uses a message identifier to check if the message is received by waiting for a acknowledgment message with that same identifier. In cases where the acknowledgment is not received it resends the message. Due to this method, it is possible that a same message is received several times. In this case, the identifier previously sent ensures that the message is read and only interpreted once.

The Control Supervisor module is responsible for the aircraft control. Under normal conditions, all aircraft control is done via the Piccolo module. However, under two conditions the Control Supervisor assumes control of the aircraft: when the Sense and Avoid (S&A) module warns of a possible collision and when a detected target is engaged in pursue.

The Target Tracking module uses the target and the aircraft position provided respectively by the SEP and Autopilot Driver to ensure the aircraft flies into a circular loitering pattern around the target.

The Sense and Avoid (S&A) module warns of possible collisions with neighboring aircraft platforms present in the area of operation and instructs commands to reroute the aircraft in case of collision. Two rerouting strategies are provided, either by non linear control or by a mixed approach based in cooperative/adversary game theory behavior from platforms in route of a possible collision. When a possible collision is detected the collision avoidance algorithm provides the best navigation solution and alerts the Control Supervisor module. The latter authorizes a high priority navigation decision to avoid collisions.

2.1.5 On-Board Payload System

The on-board computer system uses a Linux operating system. It contains the Image Acquisition and Automatic Identification System (AIS) driver, Seagull Manager, Detection Module and Seagull Actuators modules all connected via the ROS middleware as presented in Figure 2.7.
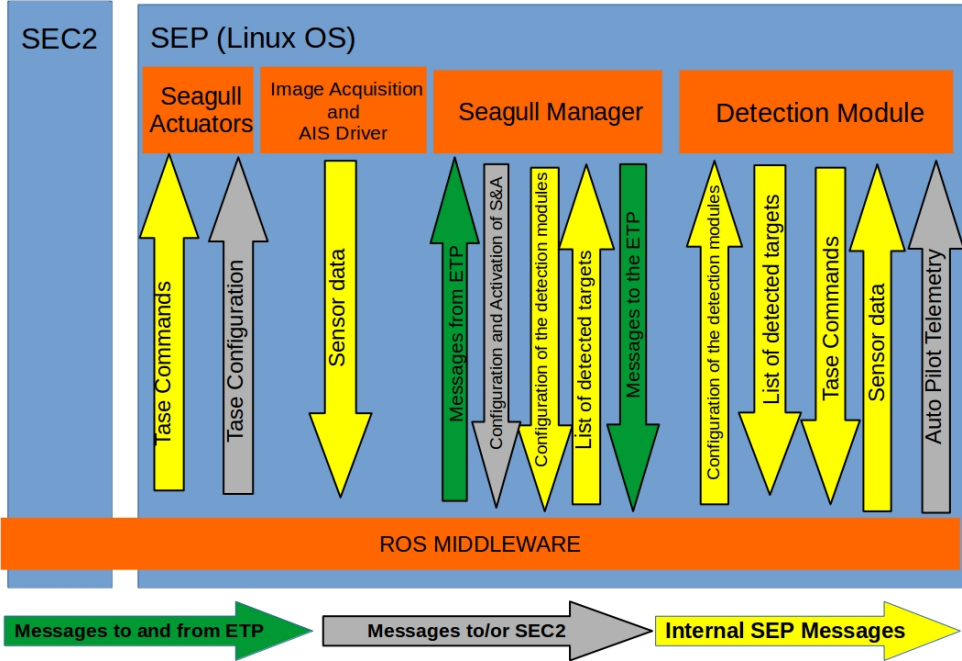
Figure 2.7: On-Board Payload Architecture (adapted from Figure 10 in [19]).

The Image Acquisition modules are responsible for data acquisition from the optical sensors employed in the mission. For each camera device in the payload with support of Video4Linux (available in the systems kernel) and with the ROS OpenCv, an image message is periodically published in specific ROS topics dedicated to each sensor. This allows the Detection module to subscribe to the dedicated topics to access the images

The Automatic Identification System (AIS) is used by maritime platforms, monitoring services in order to locate platforms. This system is mandatory by law to be present in vessels bigger then 15 meters long and transmits information regarding the vessel identification, position, heading and velocity. The data rate transmission varies accordingly to the vessel velocity taking anywhere from 20 seconds to 3 minutes and the maritime monitoring services gather the information of all vessels sailing along the cost or via satellite. The AIS Driver equipped in the Seagull architecture enables the UAV platform to receive the same information that the monitoring services (in the scenario of operation of the aircraft) and communicates it to the SEP unit.

The Seagull Manager is the module responsible for synchronizing the software communication between the SEP and the ETP. The communicated data is then sent to the Communication Relay module in SEC2 which allows generic communication in the format of byte arrays. The Seagull Manager will encode/decode the byte arrays and sent or receive data accordingly to the Mavlink (Micro Air Vehicle

Communication Protocol). This protocol allows the easy structuring of messages and automatic features to encode and decode messages.

Some of the principal interactions between the Seagull Manager and the ETP are:

- Mission planning: The ETP will communicate a mission plan to the Seagull Manager with the mission type and area of operation. The Seagull Manager uses this information to configure the Detection module accordingly;

- Detection notification: The Seagull Manager sends a notification to the ETP each time a target is detected by the Detection Module;

- Order to (or not to) follow a target: The ETP can request SEP the activation or deactivation of the target following feature. By executing this request the Seagull Manager will activate/deactivate such feature in the Control Supervisor of the SEC2;

- Image Request: The ETP can request SEP for a specific target image. The Seagull Manager will redirect that request to the Detection Module. Once an image is acquired it is sent back to the ETP.

2.1.6   Detection Module

The Detection module integrates all the computer vision, filtering, estimation and decision algorithms. The module is itself an aggregate of other sub-modules as presented in Figure 2.8 and is implemented in C++. Each internal sub-module is implemented as a class. This architecture has been chosen due to performance issues.

Some of this performance issues are related with the transmission of images as a ROS message, which can be difficult process particularly if multiple messages are transmitted. Problems related to message lagging can occur either from the publisher or subscriber, wrong message encoding from the parts in communication and memory usage are just some examples of common issues, which are exacerbated if multiple images and/or cameras are used.

The Acquisition sub-model receives images from the several camera drivers and is responsible for the correcting the sampling of images, were it does some pre-processing of the received images (e.g. image format conversion).

The telemetry data can be obtained from several sources either from the autopilot unit or the TASE 150 gimbal. Due to the fact that the Detection module receives different information at different times instances from different sources, the Telemetry Data Aggregation sub-module analyzes all of this information and synchronizes it with the images received to obtain a stream of images and respective telemetry at the time of the capture.

The Vision sub-module employs all target detection algorithms. This sub-module uses the images collected and the detection configuration parameters correspondent with the mission type. It provides descriptors of the targets identified, including not just the dimension and position of the target vessels in the image plane but also a quality quantifier parameter for the detection found.
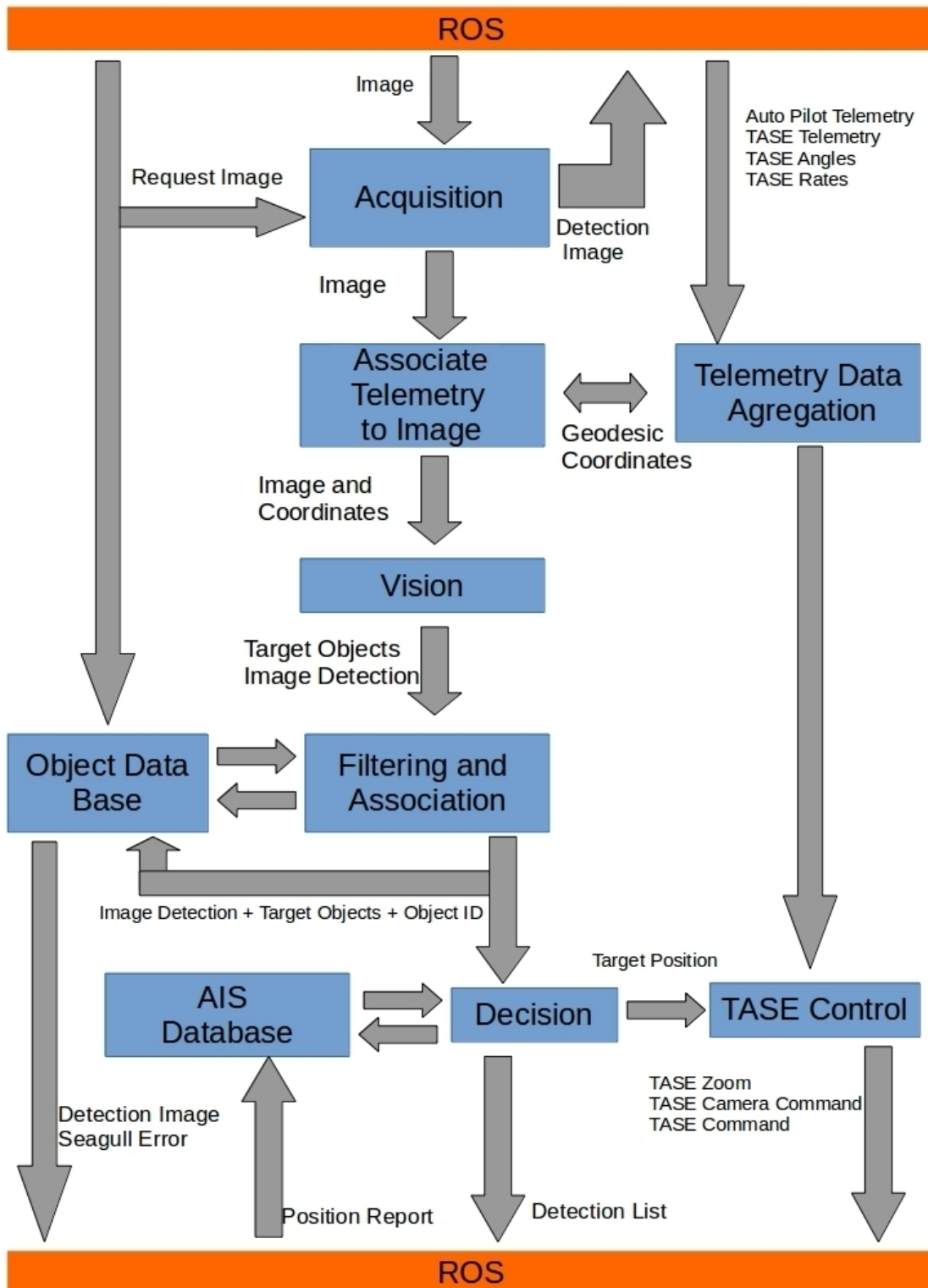
Figure 2.8: Detection Module (adapted from Figure 29 in [18]).

The Filtering and Association sub-module is tasked with two operations. First it filters the target's position, helping the detection of targets in intermittent data and increases the detection quality throughout the mission, consequently, increasing the detection quality quantifier of consistently detected targets. Second, it determines which detection correspond to each target vessel. This association task is required because the Vision sub-module returns a bulk of detected targets without any information of which target vessel originate each detection.

The Object Data Base sub-module stores the location and the images of previously detected targets. This information is updated with the data obtained from the Filtering and Association module. Whenever new target are detected, a new entry is added to the data base.

The Decision sub-module receives all of the information and determines if the target should be pursued or not, according to the current mission requirements. The Decision module can use the location data from the AIS database by comparing it with the targets positions. This module sends the required information for controlling the gimbal camera through the respective control module. The output of the Decision sub-module is a list of detected targets and their relevance to the mission.

The AIS Database module compiles the data transmitted from the vessels. Because this data has a slow update rate it is necessary to store it for later use.

The TASE Control sub-module ensures the movements pan, tilt and zoom of the TASE150 gimbal as long as this payload equipment is present. This sub-module calculates the parameters necessary for the operation of the Seagull Actuators module and publishes them in the a ROS topic "tase_command_in".

2.1.7 Seagull Actuators

The module Seagull Actuators controls the TASE 150 gimbal. This is a payload module with active control over the TASE 150 pan, tilt and zoom, allowing the target object to be followed. The pan, tilt and zoom of the camera can be controlled manually or automatically maintaining fixed rotation rates or angles, keeping a target object in image. This module is divided into two sub-modules: TASE Driver and TASE Comms.
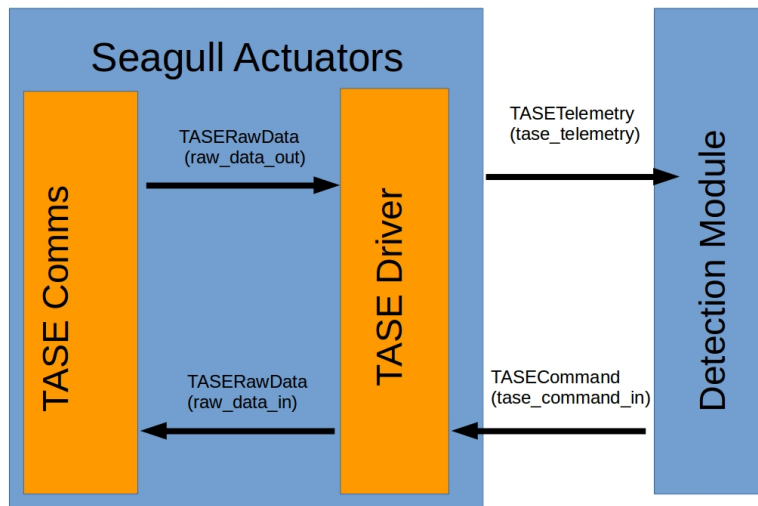
Figure 2.9: Complete Seagull Systems (Figure 32 in [18]).

The TASE Driver sub-module manages all the communication with the TASE Comms. The raw data is received in this node, decoded and aggregated in the ROS messages "TASETelemetry", "TASEAngles" or "TASERates". The TASE Control sub-module in the Detection module presented previously is responsible for publishing the ROS message "TASECommand".

The TASE Comms sub-module manages all communication between the TASE, the Autopilot Driver module and the SEP. The connection to the Autopilot Driver module enables a ground operator to manually control the TASE150. The connection to the SEP allows the Detection module to operate the gimbal automatically. This two connections are mutually exclusive.

2.2 TASE Features

During the experimental procedures presented in this dissertation, further described in latter chapters, a new module named TASE Features was inserted into the Seagull software architecture. This module is used for the experimental procedures only with the objective of isolating the gimbal control part of the remaining components of the Seagull project software (either the Detection module or the autopilot). The module Tase features ensures a more direct control of the TASE 150 gimbal, where the several configurations of the camera can be controlled (pan, tilt, zoom, focus, shutter aperture among

others), without the interference of other modules. This module is a new feature (not coded in the original software architecture of the Seagull project). It was programmed and inserted in the SEP unit for the camera calibration tests only, being latter replaced by the Geo-location module (see section 4.3). Figure 2.10 presents how the Tase features module is integrated in the SEP software architecture.
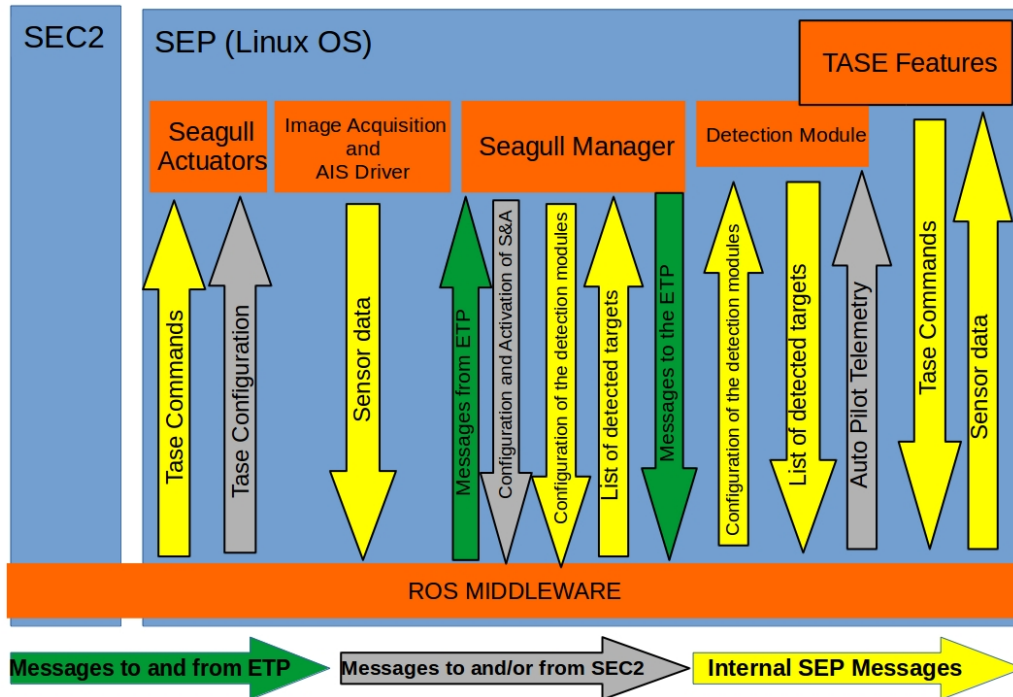
Figure 2.10: SEP experimental software architecture.

The TASE Features module uses the same information as the Detection module from the SEP computer viewpoint and communicates with the Seagull Actuators module via the same ROS topics (which the messages structure is presented in Tables 2.1 and 2.2). Furthermore, the Detection module output (a list of detected targets) is not considered. This provisional architecture is self sufficient to detect the features necessary for the camera calibration procedures and ensures that all ROS communications received or sent by this module are contained in the internal SEP architecture, specifically the ROS messages TASECommand and TASETelemetry.

Table 2.1 sumarizes all parameters structured within a "tase_command_in" ROS topic and the published message TASECommand. This message establishes angular positions or angular speeds commands as well as the stabilization mode and zoom rates. It is transmitted by the Tase Features module and does not require acknowledgment of reception.

Table 2.2 summarizes all parameters structured within a "tase_telemetry" ROS topic and the published message TASETelemetry. This message allows to obtain sensor data of the TASE 150 gimbal and the Sony FCB-EX1000 camera. This information will later provide the necessary parameters to implement the geo-location algorithm.

Message Fields	Format	Units	Description
header	Std_msgs	N/A	Header with additional information
pan	float32	degrees or degrees/second	Agular position or angular speed of pan camera movement.
tilt	float32	degrees or degrees/second	Agular position or angular speed of tilt camera movement.
zoom	int8	degrees or degrees/second	Camera Zoom movement. From -8 to 8 where negative numbers represent zoom out and positive numbers represent zoom in.
zoom_timeout	uint8	10 milliseconds	Specification for the zoom movement time of execution.
flags	uint8	N/A	0-3 Reserved modes of operation. 4 Not applicable. 5 Angular Speed and image stabilization command turn off. 6 Angular Speed and image stabilization command turn on. 7 Position and image stabilization command turn off. 8 Position and image stabilization command turn on. 9 impulse and image stabilization command turn off. 10 impulse and image stabilization command turn on.
impulse_time	uint8	10 milliseconds	Time to respond to a impulse command.

Table 2.1: tase_command_in topic message fields (Table 28 in [18]).

Message Fields	Format	Units	Description
header	Std_msgs	N/A	Header with additional information
flags	uint16	N/A	Indicator for data package
mode	uint8	N/A	Camera operation mode
camera	uint8	N/A	Camera bits information
time	uint32	m/s	Time since last reset
latitude	float32	degrees	Camera GPS latitude coordinates
longitude	float32	degrees	Camera GPS longitude coordinates
altitude	float32	meters	Camera altitude
vnorth	float32	meters/second	North component speed vector
veast	float32	meters/second	East component speed vector
vdown	float32	meters/second	Down component speed vector
mount_roll	float32	degrees	Roll angle of the gimbal body
mount_pitch	float32	degrees	Pitch angle of the gimbal body
mount_yaw	float32	degrees	Yaw angle of the gimbal body
pan	float32	degrees/second	Pan angle of the camera in relation to the gimbal body
tilt	float32	degrees/second	Tilt angle of the camera in relation to the gimbal body
roll	float32	degrees/second	Roll angle of the camera in relation to the gimbal body
hfov	float32	degrees	Horizontal field of view in degrees
vfov	float32	degrees	Vertical field of view in degrees
focus	uint16	N/A	Focus position
board_temp	int8	°C	Temperature
zoom_rate	uint16	X	Zoom ratio
focus_mode	uint8	N/A	Camera focus mode

Table 2.2: tase_telemetry message fields (Table 32 in [18]).

2.3 Projective Geometry

Projective geometry is an important tool when studying camera models and calibration procedures. It is a subject of study in several domains of computer science, mainly in computer vision modeling and computer graphics. This type of geometry establishes the mathematical framework that defines the geometry of cameras and their respective transformations, hence enabling the study of bi-dimensional projections of tri-dimensional scenes/objects.

Although projective geometry is an extension of the Euclidean geometry, it is possible to represent and compute transformations of objects at infinity. This avoids cases of added difficulty. E.g. in projective geometry there are no parallel lines, they have a point of intersection at infinity. This is particularly useful (computation wise) when detailing the camera calibration technique described by Zhang [7].

The contents of this section are introductory concepts and notations, which are used during this dissertation.

2.3.1 Bi-dimensional Projective Geometry

A homogeneous point in bi-dimensional projective geometry is defined by: $X = [x_0 \ x_1 \ x_2]^T$. When $x_2 \neq 0$ this corresponds to a finite set of points in $\mathbb{R}^2$. The set of points with $x_2 = 0$ results in a space with a set of all homogeneous 3-vectors, namely the projective space $\mathbb{P}^2$.

A set of vectors $[kx_0 \ kx_1 \ kx_2]^T$ for varying values of k are a representation of the generic point $X = [x_0 \ x_1 \ x_2]^T$ in $\mathbb{R}^2$ represented by a homogeneous vector. Considering two points A and B in the space $\mathbb{P}^2$ and a line going through them. A third point C belongs to said line if

$$\begin{vmatrix} x_0^A & x_0^B & x_0^C \\ x_1^A & x_1^B & x_1^C \\ x_2^A & x_2^B & x_2^C \end{vmatrix} = 0, \tag{2.1}$$

this establishes a linear dependency where their determinant vanishes.

Defining a line by $L = [l_0 \ l_1 \ l_2]$, different parameters l_0, l_1 and l_2 generate different lines. This parameters can be defined as

$$l_0 = \begin{vmatrix} x_1^A & x_1^B \\ x_2^A & x_2^B \end{vmatrix}, l_1 = - \begin{vmatrix} x_0^A & x_0^B \\ x_2^A & x_2^B \end{vmatrix}, l_2 = \begin{vmatrix} x_0^A & x_0^B \\ x_1^A & x_1^B \end{vmatrix}, \tag{2.2}$$

where l_i are functions of the points coordinates, as such points C belongs to L if

$$l_0 x_0^C + l_1 x_1^C + l_2 x_2^C = \begin{bmatrix} l_0 & l_1 & l_2 \end{bmatrix} \cdot \begin{bmatrix} x_0^C \\ x_1^C \\ x_2^C \end{bmatrix} = 0. \tag{2.3}$$

The representation of a vector can be confused with lines, where $l_0 x_0^C + l_1 x_1^C + l_2 x_2^C = 0$ and $k(l_0 x_0^C + l_1 x_1^C + l_2 x_2^C) = 0$ are the same for any non-zero constant k, representing the same line.

A practical approach of representing $\mathbb{P}^2$ is thinking of it as a set of rays in $\mathbb{R}^3$. The set of all vectors $k\,[x_1 x_2 x_3]$ as k varies forms a ray through the origin. Such a ray may be thought of as representing a single point in $\mathbb{P}^2$ [13].

Similarly to Euclidean geometry, in projective geometry two distinct points define a line and two lines intersect in one point. It is asserted that the role of points and lines may be interchanged in statements concerning the properties of lines and points. The fact that $X^T L = 0$ for line and point is symmetric, implies that $L^T X = 0$, in which the positions of line and point are swapped. This principles is stated as a general principle: the duality principle

> *"The duality principle states that any theorem of bi-dimensional projective geometry there corresponds a dual theorem, which may be derived by interchanging the roles of points and lines in the original theorem"*[13]

The linear transformations of $\mathbb{P}^2$ to itself defined (up to a scale factor) by non-singular $3\,X\,3$ matrices are called homographies, collineation or projective transformation of $\mathbb{P}^2$. The homographie matrix

$$H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}, \tag{2.4}$$

is a homogenous matrix, since the homogenous representation of a point, only the ratio of the matrix elements are relevant. To interpret this assertion, any point in $\mathbb{P}^2$ is represented as a homogeneous 3-vector X and HX is a linear mapping of the homogeneous coordinates. The projectivity is a linear transformation in homogenous coordinates.

The projection along rays through a common point (the center of projection) defines a mapping from one plane to another. Therefore this point-to-point mapping preserves lines from the original plane to the projected plane, preserving the relationship of points and line between planes.

A conic is a curve described by a second-degree equation in the projective plane. In Euclidean geometry, conics are of three main types: hyperbola, ellipse and parabola. The conic coefficient matrix is defined as

$$C = \begin{bmatrix} c_0 & c_1/2 & c_3/2 \\ c_1/2 & c_2 & c_4/2 \\ c_3/2 & c_4/2 & c_5 \end{bmatrix}, \tag{2.5}$$

where the equation of a conic in homogeneous coordinates is given by

$$c_0 x_0^2 + c_1 x_0 x_1 + c_2 x_1^2 + c_3 x_0 + c_4 x_1 + c_5 = 0, \tag{2.6}$$

or rewriting 2.6 in matrix form:

$$\begin{bmatrix} x_0 & x_1 & 1 \end{bmatrix} C \begin{bmatrix} x_0 \\ x_1 \\ 1 \end{bmatrix} = X^T C X = 0. \tag{2.7}$$

The conic coefficient matrix is symmetric. As in the case of homogeneous representation of points and lines, only the coefficients of the matrix are important, since multiplying by a non-zero scalar does not affect the above equations.

The conic defined above by equation 2.6 is more accurately named a point conic, as it defines an equation on points, where X is the vector of homogeneous coordinates associated to points in the plane and C is the homogeneous matrix associated to the conic. Any homography transformation H deforms the conic C and yields a new conic. In the projective plane, there is no distinction between conics that are simply projective deformations of the circle through homography transformations H of the plane.

Given the duality theorem it is possible to assume that interchanging the roles of points and lines defines an equation on lines for a conic. This conic is also defined by a $3X3$ matrix, which is denoted C^* and is called the dual conic.

A line L tangent to the conic C satisfies $L^T C^* L = 0$, illustrated in Figure 2.11. The point X at which the line L is tangent to C is $X = C^{-1}L$. Since X satisfies equation 2.7 the same reasoning can be applied to lines,

$$L^T C^{-1} L = L^T C^* L = 0. \tag{2.8}$$

(a)   (b)
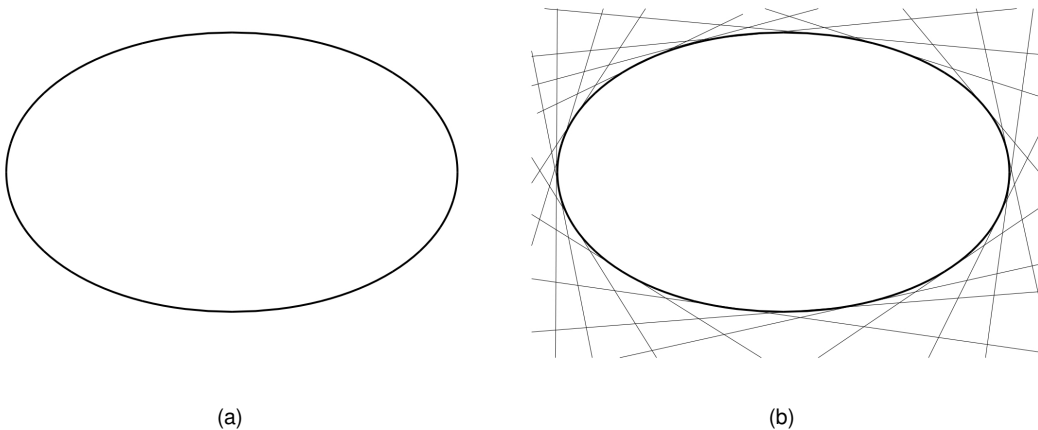
Figure 2.11: (a) Points X satisfiyng $X^T C^* X = 0$ lie on a point conic. (b) Lines L satisfying $L^T C^* L = 0$ are tangent to the point conic C. The conic C is the envelop of lines L (Adapted from Figure 22 in [13]) .

2.3.2 Tri-dimensional Projective Geometry

A point $X = [x \ y \ z]^T$ in a tri-dimensional space $\mathbb{R}^3$ is represented in homogeneous coordinates as a 4-vector. The homogeneous vector $X = [x_1 \ x_2 \ x_3 \ x_4]^T$ with $x_4 \neq 0$ is representative of the point inhomogeneous coordinates in $\mathbb{R}^3$ with coordinates

$$
\begin{cases}
x = x_1/x_4 \\
y = x_2/x_4 \\
z = x_3/x_4
\end{cases}
. \tag{2.9}
$$

As an example, a homogeneous representation of $[x \ y \ z]^T$ is $X = [x_1 \ x_2 \ x_3 \ 1]^T$ where homogeneous points with $x_4 = 0$ represent points at infinity [13].

A plane π in 3 dimensional space can be defined as

$$
\pi_1 x_1 + \pi_2 x_2 + \pi_3 x_3 + \pi_4 x_4 = 0, \tag{2.10}
$$

or more concisely,

$$
\pi^T X = 0, \tag{2.11}
$$

which denotes that point X is on plane π [13].

Assuming that three points X_i are incident with the plane π, then each point must satisfy equation 2.11 and thus $\pi^T X_i = 0$, $i = 1, \therefore, 3$ represented in matrix by

$$
\begin{bmatrix}
X_1^T \\
X_2^T \\
X_3^T
\end{bmatrix}
\pi = 0. \tag{2.12}
$$

The plane π defined by the points is thus obtained as a one dimensional null-space. If the matrix has only a rank 2, and consequently the null-space is two dimesional, then the points are collinear, and define a pencil of planes with the collinear points as the axis.

The absolute conic Ω_∞ is a point conic on π_∞ (the plane at infinity). It plays an important conceptual role in camera calibration and scene reconstruction and is therefore important to examine it more closely. The absolute conic Ω_∞ is expressed as the set of equations:

$$
\begin{cases}
X_1^2 + X_2^2 + X_3^2 = 0 \\
X_4^2 = 0
\end{cases}
. \tag{2.13}
$$

Therefore for points $X_4^2 = 0$ the equation 2.7, when aplied to the tri-dimensional projective space can be defined as

$$\begin{bmatrix} X_1^T & X_2^T & X_3^T \end{bmatrix} \Omega_\infty \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix}^T = 0, \tag{2.14}$$

where Ω_∞ corresponds to a conic C with matrix $C = I$. It represents a conic of purely imaginary points. A more clear definition is the representation of a circle of radius $\sqrt{-1}$ on π_∞. The absolute conic Ω_∞ is invariant to 3 dimension transformations and the π_∞ is invariant to 3 dimensional affine transformations.

The absolute dual quadric Q_∞^* (DAQ) consists of the planes tangent to Ω_∞, so that Ω_∞ is the disk of Q_∞^* at π_∞. Illustrated in Figure 2.12, this dual quadric has the plane at infinity in its envelope, and the absolute conic as the conic formed by the tangent lines defined by the other planes in the envelope, where the absolute dual quadric matrix representation is

$$Q_\infty^* = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.15}$$
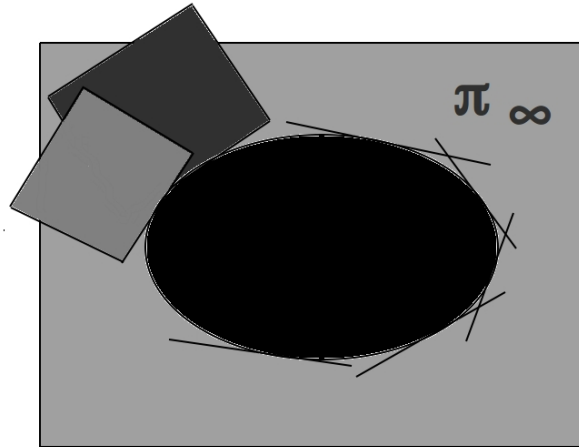
Figure 2.12: Absolute dual quadric representation in π_∞.

Chapter 3

Methodology

This chapter presents the framework of the geo-location method employed in this dissertation. The definition of a camera model and it's behavior (regarding it's variable lenses configuration e.g. zoom, focus) is a fundamental object of study in this work. Furthermore, it's integration with the aircraft motion frames is carefully presented describing the developed methodology to convert a target pixel coordinates to world coordinates.

Section 3.1 presents the camera model and the calibration procedure used by the OpenCv tool box (based in the work of Zhang [7]). The camera lenses imperfections and induced image distortion are taken under consideration and defined by a set of distortion parameters which mimic the lenses behavior.

Section 3.2 outlines the aircraft, gimbal and camera coordinate frames. The content presented in this section is adapted from [21] and presents the coordinate transformation matrices required to employ the paper-map-based based method described the section 3.3.

Section 3.3 details the complete method employed to solve the geo-location problem. It proposes how the camera model and the aircraft motion are integrated to obtain a target world coordinates. The presented method is only dependent of the aircraft on-board sensor reading and a single image of the target.

3.1 Camera Model and Calibration

3.1.1 Camera Model

A camera device is commonly represented by the "pin-hole" model set by a $3X4$ projection matrix. This matrix maps a tri-dimensional space into a bi-dimensional projective space using a perspective transformations. A tri-dimensional point in homogeneous coordinates defined $M = [X\ Y\ Z\ 1]^T$ is transformed into $m = [u\ v\ 1]^T$ in homogeneous coordinates of an image plane. The geometric representation of the "pin-hole" model perspective transformation is represented in Figure 3.1.
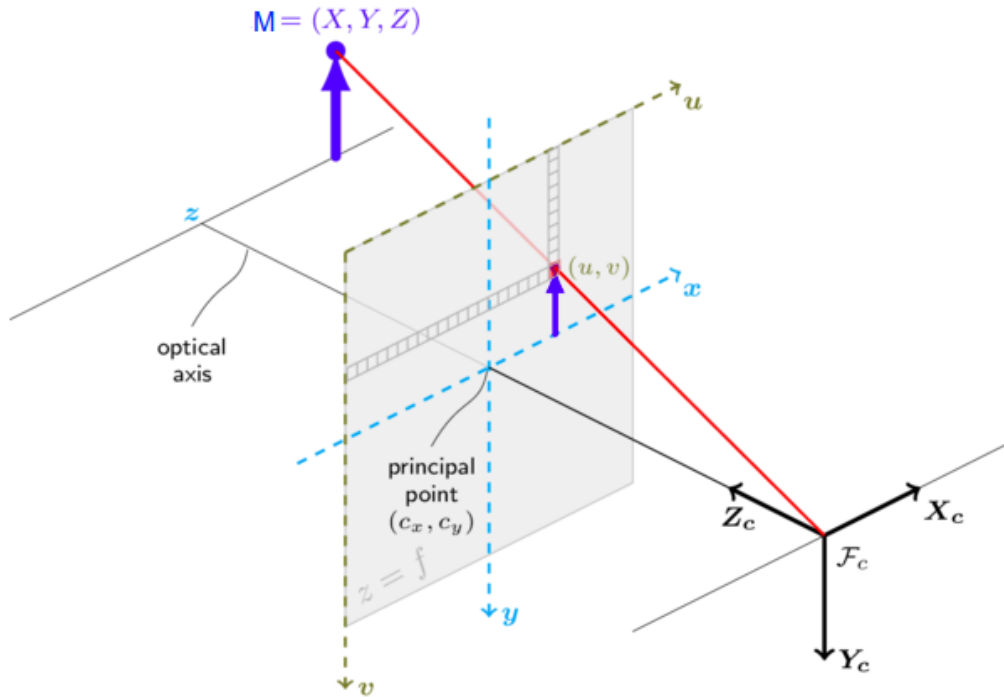
Figure 3.1: Pinhole camera model representation.

The "pin-hole" camera model is defined as

$$sm = A \begin{bmatrix} R|t \end{bmatrix} M,$$ (3.1)

or,

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix},$$ (3.2)

where:

- (u ,v) - Image coordinates of a projected point;

- A - Camera matrix, or matrix of intrinsic parameters;

- $(f_x f_y)$ - Focal lengths in the x and y axis;

- (c_x, c_y) - Principal point coordinates;

- $\begin{bmatrix} R|t \end{bmatrix}$ - Joint rotation-translation matrix (extrinsic parameters);

- $[X\ Y\ Z\ 1]^T$ - Coordinates of a 3D point in the world coordinate space;

- s - Scale factor.

The matrix of intrinsic parameters does not depend on the scene viewed, therefore once the intrinsic parameters are determined, they can be re-used as long as the focal length is fixed [22]. The joint rotation-translation matrix of the extrinsic parameters is used to describe the camera motion around a static scene, or rigid motion of an object in front of a still camera.

When $Z \neq 0$ the equation 3.2 can be rewritten [22]

$$x^{'} = X/Z, \tag{3.3}$$

$$y^{'} = Y/Z, \tag{3.4}$$

$$u = f_x x^{'} + c_x, \tag{3.5}$$

$$v = f_y y^{'} + c_y, \tag{3.6}$$

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \left(R \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + t \right). \tag{3.7}$$

3.1.2 Camera Calibration

Camera calibration is a necessary procedure in order to determine the matrix of intrinsic parameters. This step is fundamental in tri-dimensional computer vision and allows to extract metric information from bi-dimensional images.

Zhang [7] proposes a technique which requires only that a camera observe a planar pattern (or model points) shown in several different poses, where either the camera or the planar pattern can be moved by hand. The proposed approach, which uses bi-dimensional metric information, lies between photogrammetric calibration, which uses a explicit tri-dimensional pattern measurements, and self-calibration, which uses motion rigidity or equivalently implicit tri-dimensional information.

Assuming that the observed pattern is on $(Z = 0)$ of the world coordinate system, equation 3.2 is re-defined as

$$sm = A \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = A \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}, \tag{3.8}$$

therefore, a point M and its image projection m are related by a projective homography matrix H which maps points between π_∞ and the image plane:

$$sm = H M, \tag{3.9}$$

$$H = \lambda A \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}, \tag{3.10}$$

as a simplification, denoting the *i*th column of the joint rotation-translation matrix by R_i. The homography matrix H is denoted as

$$H = \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix} = \lambda A \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} = \lambda A \, R_i, \tag{3.11}$$

this homography mapping method is independent of the camera position and depends only on the camera internal calibration and the model, where λ is an arbitrary scalar [7].

The absolute conic is invariant under Euclidean transformations, therefore its relative position to a moving pattern is constant. For fixed intrinsic camera parameters the image of the absolute conic will also be constant. Since the absolute conic Ω_∞ is on π_∞ it is possible to compute its image projection under the homography matrix H. The image of the conic $C = \Omega_\infty = I$ in π_∞[13] is defined by

$$w = (AR_i)^{-T}\Omega_\infty(AR_i)^{-T} = (AR_i)^{-T}I(AR_i)^{-T} = A^{-T}R_iR_i^{-1}A^{-1} = A^{-T}A^{-1}. \tag{3.12}$$

The expression $w = A^{-T}A^{-1}$ describes the image of the absolute conic w (IAC). Similarly to Ω_∞ the conic w is an imaginary point conic with no real points. Using the duality principle we can extrapolate that the dual image of the absolute conic (DIAC) is

$$w^* = w^{-1} = AA^T, \tag{3.13}$$

this is a line conic, whereas w is a point conic (although it does not contain any real points as stated before), the conic w^* is the image of Q_∞^*. This results points that once w (or the equivalent w^*) is identified in an image then the intrinsic parameters matrix A is also determined. The Absolute conic can be perceived as a calibration object present in all scenes. Once determined, it can be used to reconstruct the metric scene captured by the camera. It is, however, not always so simple to find the absolute conic in the reconstructed space.
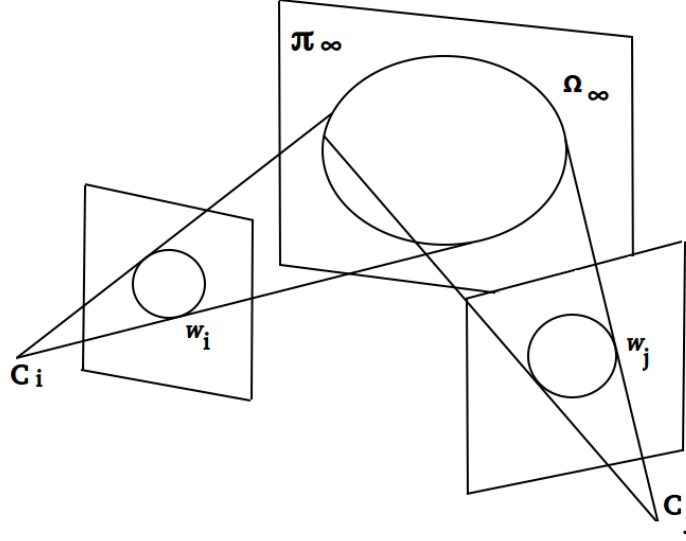
Figure 3.2: The absolute conic Ω_∞ and its image projection w for different camera positions i and j.

The model plane (or model pattern) under the assumption of the simplified notation introduced in equation 3.11 [13], is defined as

$$
\begin{bmatrix} r_3 \\ r_3^T\, t \end{bmatrix}^T \begin{bmatrix} X \\ Y \\ Z \\ 0 \end{bmatrix} = 0,
\tag{3.14}
$$

intersecting the plane π_∞ at a line defined by two particular points $M_1 = [r_1\ 0]$ and $M_2 = [r_2\ 0]$, by intersecting this line with the absolute conic, we obtain:

$$
M_\infty = \begin{bmatrix} r_1 \pm i\, r_2 \\ 0 \end{bmatrix}.
\tag{3.15}
$$

The resulting a pair of complex conjugate points reveals that under Euclidean transformation they are invariant and therefore their projection on the image plane can be derived as

$$
m_\infty = A(r_1 \pm i\, r_2) = (h_1 \pm i\, h_2),
\tag{3.16}
$$

and considering that a point on the image of the absolute conic (w) must verify equation 2.7, presented in the previous chapter, results

$$
(h_1 \pm i\, h_2)^T\, w\, (h_1 \pm i\, h_2) = (h_1 \pm i\, h_2)^T A^{-T} A^{-1} (h_1 \pm i\, h_2) = 0,
\tag{3.17}
$$

where both real and imaginary parts must be zero. This results in two constraints over the intrinsic camera parameters for one given homography:

$$\begin{cases} h_1^T A^{-T} A^{-1} h_2 = 0, & \text{(3.18a)} \\ h_1^T A^{-T} A^{-1} h_1 = h_2^T A^{-T} A^{-1} h_2. & \text{(3.18b)} \end{cases}$$

Defining w as a generic matrix

$$w = A^{-T} A^{-1} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}, \tag{3.19}$$

it holds that w is symmetric and defined by a 6D vector [7]:

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{22} & w_{13} & w_{23} & w_{33} \end{bmatrix}^T. \tag{3.20}$$

By defining the ith column vector of H by $h_i = \begin{bmatrix} h_{i1} & h_{i2} & h_{i3} \end{bmatrix}^T$, we obtain

$$h_i^T w h_j = v_{ij}^T W, \tag{3.21}$$

were the points of the model pattern observed in a image are described as

$$v_{ij} = \begin{bmatrix} h_{i1}h_{j1} & h_{i1}h_{j2} + h_{i2}h_{j1} & h_{i3}h_{j3} & h_{i3}h_{j1} + h_{i1}h_{j3} & h_{i3}h_{j2} + h_{i2}h_{j3} & h_{i3}h_{j3} \end{bmatrix}^T. \tag{3.22}$$

therefore the constraints provided by equations 3.18a and 3.18b, from a given homography, can be rewritten as 2 homogeneous equations in w [7]:

$$\begin{bmatrix} v_{12}^T \\ (v_{11} - v_{22})^T \end{bmatrix} W = 0. \tag{3.23}$$

Once n images of points are observed, by staking n equations 3.23 we obtain [7]:

$$VW = 0. \tag{3.24}$$

In this equation V is defined has a $2n\text{X}6$ matrix. If $n > 3$ we will have in general a unique solution W defined up to a scale factor. The solution to3.23 is well known as the eigenvector of $V^T V$ associated to the smallest eigenvalue (equivalently, the right singular vector of V associated to the smallest singular value). Once W is estimated, we can compute all camera intrinsic matrix A [7].

3.1.3 Camera Distortion

So far in this work it has been assumed that the pin-hole model describing the camera perspective transformations of tri-dimensional entities to bi-dimensional image objects is optimal. However, distortion parameters are present in any camera lenses and must be taken into account.

A distortion can be referred as a lens aberration in which a deviation from rectilinear projection occurs. A lens that exhibits distortion produces slightly curved images of all those lines that do not pass through the center of the image. This factor originates mainly from the lens manufacturing process or assembling and is something that can't be overlooked.

From a geometric viewpoint, the lens distortion relates to the position of image points in the image plane. If a point position in an image is not accurate the results from either the calibration procedures or computer vision algorithms that depend on its pixel coordinates will be erroneous.

Therefore, considering the lens distortion effect in the "pin-hole" camera model [23] presented in equation 3.7, the amount of distortion present in a pixel location can be defined by

$$
\begin{cases}
u^{'} = u + \delta_u(u,v) \\
v^{'} = v + \delta_v(u,v)
\end{cases}
, \tag{3.25}
$$

u and v are the original, distortion-free image coordinates and $u^{'}$ and $v^{'}$ are the corresponding coordinates with distortion observed in a image. Equation 3.25 indicates that the error in each pixel coordinate pair (u,v) is dependent on the position of the point. Two different components of distortion are considered: radial and tangential distortion.

Radial distortion is the symmetric distortion caused by the lens due to imperfections in curvature when the lens was manufactured. This type of distortion is strictly symmetric and causes either an inward or outward shift of a given image point from its ideal location [23].

A negative radial displacement of the image points is commonly referred to as barrel distortion. It causes outer points to be projected in a image plane increasingly together and the scale to decrease. On the other hand positive radial displacement is referred to as pincushion distortion and the opposite occurs causing outer points to spread in the projected image and the scale to increase.

Actual optical systems are subject to various degrees of decentering, that is, the optical centers of lenses are not co-linear. This defect introduces what is called decentering distortion. This distortion has both radial and tangential components

Thin prism distortion arises from imperfection in lens design and manufacturing as well as camera assembly (for example, slight tilt of some lens elements or the image sensing array). This type of distortion can be adequately modeled by adding a thin prism to the optical system, causing additional radial and tangential distortions.

Figures 3.3 and 3.4 illustrate how this two components radial and tangential distortion respectively can affect a image uniquely trough the lens design and manufacturing.

35
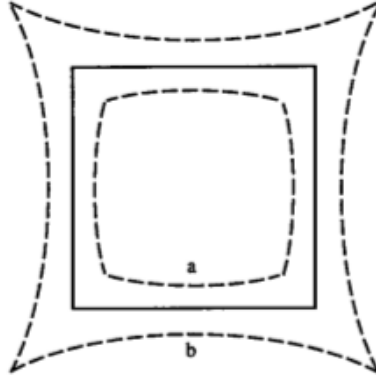
Figure 3.3: Effect of radial distortion. Solid line - no distortion; dashed line: radial distortion (a: negative, b:positive).
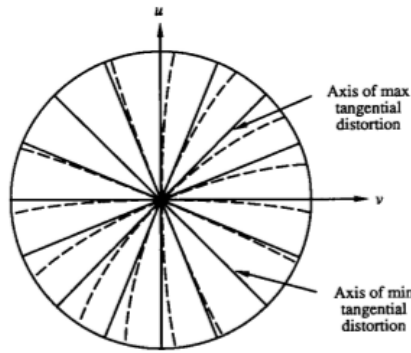
Figure 3.4: Effect of tangential distortion. Solid lines - no distortions; dashed lines - tangential distortion.

Considering this two distortions components, the pin-hole camera model with the assumptions of the equations 3.5 and 3.6 has to be re-framed to take into account the distortion, where

$$u = f_x x^{''} + c_x \tag{3.26}$$

and

$$v = f_y y^{''} + c_y. \tag{3.27}$$

The distorted coordinates $x^{''}$ and $y^{''}$ are redefined accordingly to [22] (considering that $r^2 = (x^{'2} + y^{'2})$) as:

$$x^{''} = x^{'} \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + p_2(r^2 + 2x^{'2}) + 2p_1 x^{'} y^{'}, \tag{3.28}$$

$$y^{''} = y^{'} \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + p_1(r^2 + 2y^{'2}) + 2p_2 x^{'} y^{'}, \tag{3.29}$$

where k_1, k_2, k_3, k_4, k_5 and k_6 represent the radial distortion coefficients and p_1 and p_2 are tangential distortion coefficients.

3.2 Camera, Gimbal and Body Frames

This section describes the coordinate frame transformations that are presented in a airborne platform, which uses a optical sensor to geo-locate a target. Assuming that the origins of the gimbal and camera frame are located in center of mass of the UAV there are five different frames to consider: camera frame, gimbal frame, body frame, vehicle frame and inertial frame.

Coordinate frame transformations are described by two basic operations: rotation and translation. Establishing that all coordinate frames are centered in the aircraft center of mass, translation matrices are not considered, and only rotation transformations are needed.

Any rotation operation can be achieved by composing three elemental rotations about the cartesian axis. This three different rotations are obtained trough three different angles commonly named Euler angles. The Euler angles are used to describe the orientation of a rigid body with respect to a known coordinate frame.

Rotations can be defined as three separate matrices accordingly to the coordinate frames axis x, y and z, considering a generic angle α, the rotation matrices are:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \quad R_y(\alpha) = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \quad R_z(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Figure 3.5 represents a schematic view the geometric relations between the several coordinate frames. The camera, gimbal, body, vehicle and inertial coordinate frames are denoted respectively by: $F^C = (x^C, y^C, z^C)$, $F^G = (x^G, y^G, z^G)$, $F^B = (x^B, y^B, z^B)$, $F^V = (x^V, y^V, z^V)$ and $F^I = (x^I, y^I, z^I)$. The gimbal and body coordinate frames are related by the angles α_{az} and α_{el}, which correspond to the pan and tilt movements of the gimbal camera. The body and inertial coordinate frames are related by the conventional angles of aircraft movement pitch (θ), roll (ϕ) and yaw (ψ).
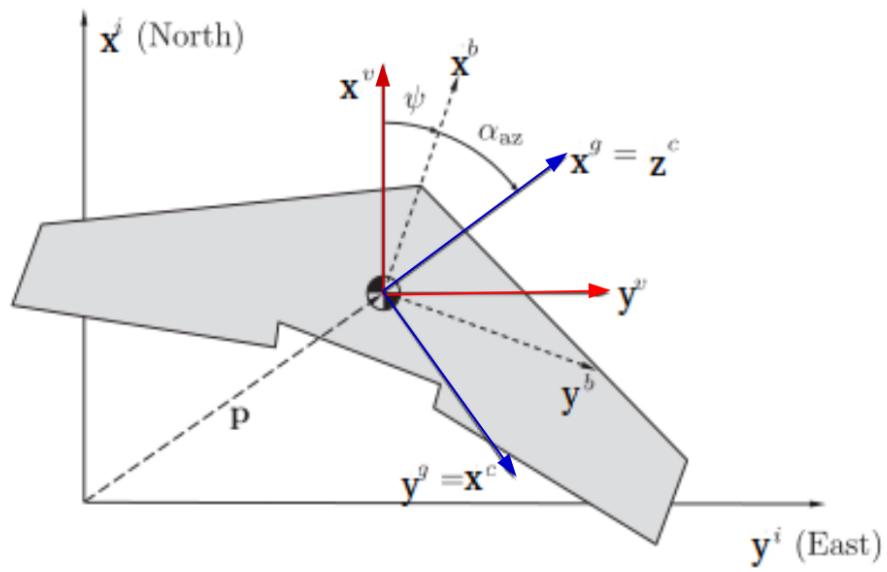
(a)
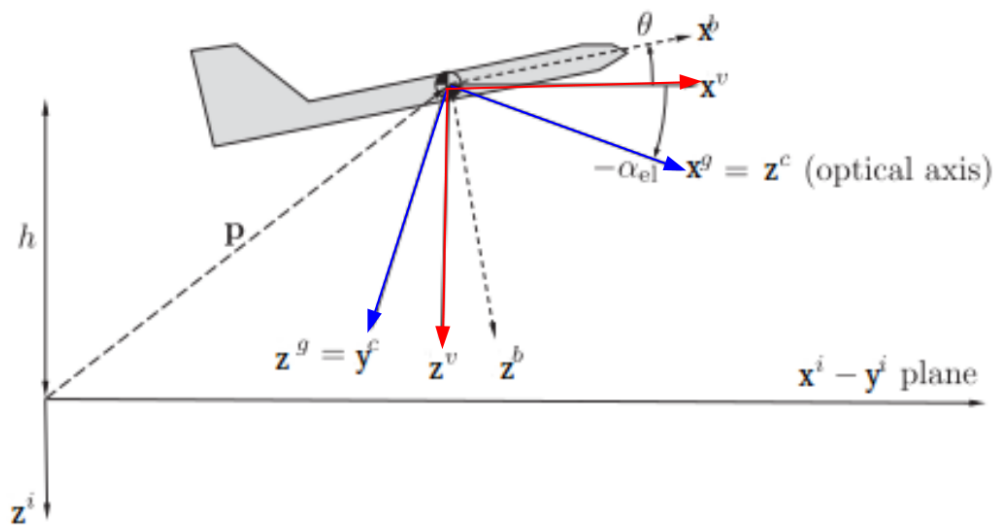
(b)

Figure 3.5: Aircraft and gimbal coordiante frames representation (adapted from [21]). **(a)**-top view. **(b)**-side view

3.2.1 Camera Frame

The origin of the camera coordinate frame is at the optical center of the camera presented in Figure 3.5 in blue. The camera axis x^C points to the right of the image plane, y^C points downward on the image plane and z^C points in the direction of the optical axis of the camera. The coordinate frame rotation from the camera to the gimbal coordinate frame is given by

$$R_C^G = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \tag{3.30}$$

The opposite rotation, from the gimbal to the camera coordinate frame is given by:

$$R_G^C = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}. \tag{3.31}$$

The transformations from the camera frame to the gimbal camera and *vice-versa* represent a simple axis swap as illustrated in Figure 3.5, where x^C, y^C and z^C correspond respectively to the gimbal axis y^G, z^G and x^G.

3.2.2 Gimbal Frame

The gimbal coordinate frame azimuth and elevation angles are respectively the pan and tilt movements of the TASE 150 gimbal, this allows it to rotate in turn of the y^G and z^G axis. The axis x^G points in the camera lens direction and the axis y^G and z^G have the same direction that y^B e z^B as Figure 3.5 illustrate.

The tilt or elevation angle (α_{el}) is defined as a rotation around the axis y^G. The pan or azimuth angle (α_{az}) is defined as a rotation around the axis z^G. Using the notation of the Euler angles the rotation from the gimbal to the body coordinate frame is defined by

$$R_B^G = R_z(-\alpha_{az})R_y(-\alpha_{el}) = \begin{bmatrix} \cos(\alpha_{el})\cos(\alpha_{az}) & -\sin(\alpha_{az}) & \sin(\alpha_{el})\cos(\alpha_{az}) \\ \sin(\alpha_{az})\cos(\alpha_{el}) & \cos(\alpha_{az}) & \sin(\alpha_{az})\sin(\alpha_{el}) \\ -\sin(\alpha_{el}) & 0 & \cos(\alpha_{el}) \end{bmatrix}, \tag{3.32}$$

whereas the opposite rotation, from the body to the gimbal coordinate frame is given by

$$R_G^B = R_y(\alpha_{el})R_z(\alpha_{az}) = \begin{bmatrix} \cos(\alpha_{el})\cos(\alpha_{az}) & \cos(\alpha_{el})\sin(\alpha_{az}) & -\sin(\alpha_{el}) \\ -\sin(\alpha_{az}) & \cos(\alpha_{az}) & 0 \\ \sin(\alpha_{el})\cos(\alpha_{az}) & \sin(\alpha_{el})\sin(\alpha_{az}) & \cos(\alpha_{el}) \end{bmatrix}. \tag{3.33}$$

3.2.3 Body Frame

The body coordinate frame describes the aircraft movement, it is located in the platform center of mass illustrated by the dashed lines in Figure 3.5. The axis x^B points toward the aircraft nose, the axis y^B points toward the aircraft right wing and z^B points to the aircraft belly.

The roll angle (ϕ) corresponds to the movement that the aircraft executes around the x^B axis, similarly the pitch angle (θ) represents the aircraft movement around the y^B axis and lastly the yaw angle (ψ) represents the movement around z^B axis. The transformation from the body to the vehicle coordinate frame is given by

$$R_B^V = R_Z(-\psi)R_Y(-\theta)R_X(-\phi), \tag{3.34}$$

$$R_B^V = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi - \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix}. \tag{3.35}$$

, the opposite transformation, from the vehicle to the body coordinate frame is given by:

$$R_V^B = R_x(\phi)R_y(\theta)R_z(\psi), \tag{3.36}$$

$$R_V^B = \begin{bmatrix} \cos\psi\cos\theta & \cos\theta\sin\phi & -\sin\theta \\ \cos\psi\sin\phi\sin\theta - \cos\phi\sin\psi & \cos\phi\sin\psi + \sin\psi\sin\phi\sin\theta & \cos\theta\sin\phi \\ \sin\phi\sin\psi + \cos\psi\cos\phi\sin\theta & \cos\phi\sin\psi\sin\theta - \cos\psi\sin\phi & \cos\theta\sin\phi \end{bmatrix}. \tag{3.37}$$

3.2.4 Vehicle Frame

The vehicle coordinate frame has its origin in the GPS location of the UAV (assumed to be the aircraft center of mass), were the axis of the coordinate frame are represented in red in Figure 3.5. The aircraft Cartesian coordinates are determined with base on its Geodesic coordinates latitude (ϕ), longitude (λ) and altitude above the geode (h) provided by the GPS receptor.

The ellipsoid of choice to convert the geodesic coordinates to Cartesian ones is the world geodetic system 1984 (WSG84). The WSG84 is an Earth-centered, Earth-fixed terrestrial reference system and geodetic datum, based on a consistent set of constants and model parameters that describe the Earth's size, shape, gravity and geomagnetic fields.

Because it is the standard of the United States Department of Defense definition for a global reference system and geospatial information, it is the main ellipsoid reference when using GPS for platforms positioning. The ellipsoid origin is located in the Earth's center of mass being defined for the whole Earth (including oceans and atmosphere).

The z axis of the WSG84 ellipsoid corresponds to the direction of the Conventional Terrestrial Pole (epoch 1984.0), with an uncertainty of 0.005". The x axis is located in the intersection of the Reference Meridian and the plane passing through the origin and normal to the z axis. The y axis completes a

right-handed, Earth-Centered Earth-Fixed (ECEF) orthogonal coordinate system.

The WGS84 ellipsoid is defined by the following parameters:

- Semi-major Axis - $a = 6378137.0$ (meters);

- Semi-minor Axis - $b = 6356752.0$ (meters);

- Flattening - $f = 1/298.257223563$;

- Eccentricity - $e = 0.08181919$;

- Radius of curvature in the prime vertical - $N = \frac{a}{\sqrt{1 - f(2-f)\sin^2(\varphi)}}$.
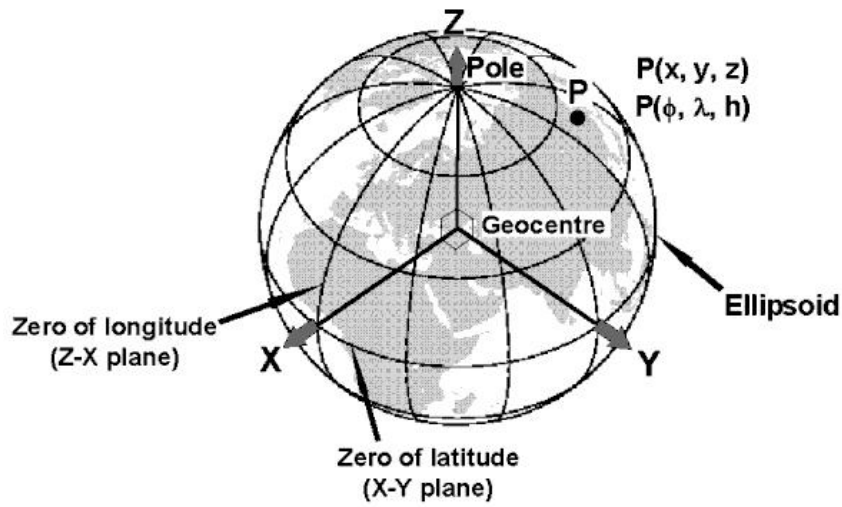
Figure 3.6: Geode representation of WGS84 reference system and geodetic datum.

Considering a point with geodesic coordinates $P = \begin{bmatrix} \phi & \lambda & h \end{bmatrix}^T$, the coordinates in a ECEF coordinate frame is given by:

$$P_{ECEF} = \begin{bmatrix} x_P \\ y_P \\ z_P \end{bmatrix} = \begin{bmatrix} (N+h)\cos(\phi)\cos(\lambda) \\ (N+h)\cos(\phi)\sin(\lambda) \\ ((1-f)^2 N + h)\sin(\phi) \end{bmatrix}.$$ (3.38)

This conversion ensures that the vehicle frame can be defined in an orthogonal coordinate frame and be integrated with the other coordinate frames defined above.

The opposite transformation of ECEF to geodetic coordinates is based in the work of Heikkinen [24] were the exact coordinates conversion is done via the following equations.

$$r = \sqrt{x_P^2 + y_P^2},$$ (3.39)

$$F = 54b^2 z_P^2,$$ (3.40)

$$G = r^2 + (1-e^2)z^2 - e^2(a^2 - b^2),$$ (3.41)

$$c = \frac{e^4 F r^2}{G}, \tag{3.42}$$

$$s = \sqrt[3]{1 + c + \sqrt{c^2 + 2c}}, \tag{3.43}$$

$$T = \frac{F}{3\left(s + \frac{1}{s} + 1\right)^2 G^2}, \tag{3.44}$$

$$Q = \sqrt{1 + 2e^4 T}, \tag{3.45}$$

$$r_0 = -\frac{Te^2 r}{1 + Q} + \sqrt{\frac{a^2}{2}\left(1 + \frac{1}{Q}\right) - \frac{T(1 - e^2)z^2}{Q(1 + Q)} - \frac{Tr^2}{2}}, \tag{3.46}$$

$$U = \sqrt{(r - e^2 r_0)^2 + z^2}, \tag{3.47}$$

$$V = \sqrt{(r - e^2 r_0)^2 + (1 - e^2)z^2}, \tag{3.48}$$

$$z_0 = \frac{b^2 z}{aV}, \tag{3.49}$$

$$e_0 = \frac{a^2 - b^2}{b^2}. \tag{3.50}$$

The altitude is defined by:

$$h = U\left(1 - \frac{b^2}{aV}\right). \tag{3.51}$$

The latitude and longitude are respectively given by:

$$\phi = arctan\left(\frac{z + e_0^2 z_0}{r}\right), \tag{3.52}$$

$$\lambda = arctan\left(\frac{y}{z}\right). \tag{3.53}$$

3.3 Geo-location

The computer vision Paper-Map-based method estimates a target location in a earth-fixed coordinate frame using the local position of the UAV as well as the position of the target relative to the UAV captured by the on-board gimbal camera.

The geo-localization methodology uses the coordinate frames presented in the previous sections and considers the transformation of the target vector (defined in the camera reference frame) among the several coordinate frames (assuming that all coordinate frames are right-handed and orthogonal).

The target range is not directly measured, instead the altitude of the aircraft above the ground (h) is used to define a scale factor to reference the target coordinates relative to the UAV platform. In Figure3.7, this allows to determine the target world coordinates by a vector sum of the inertial-aircraft ($\vec{P}_{UAV}^{I}$) vector and the aircraft-target vector ($\vec{P}_{Target}^{UAV}$).
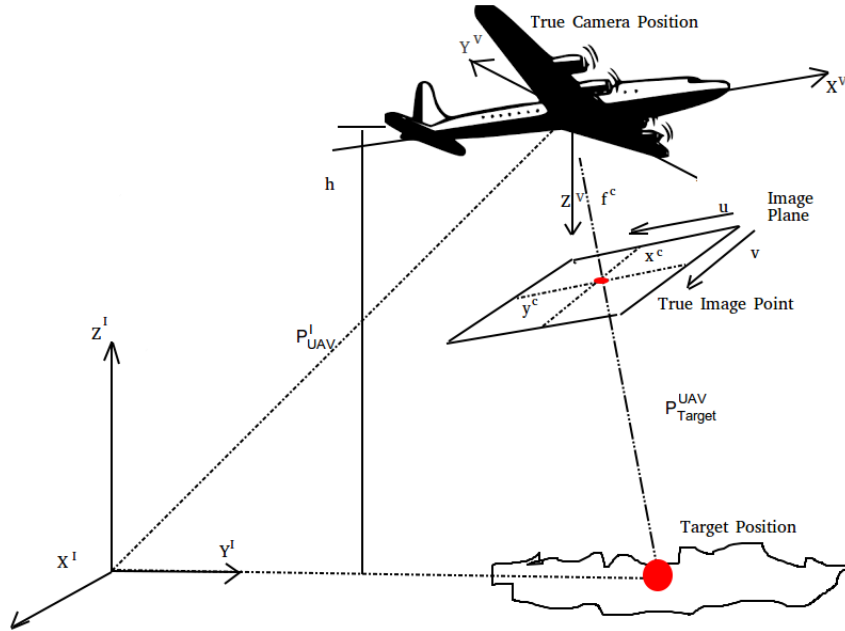
Figure 3.7: Geo-location coordinate frames overview.

The TASE 150 gimbal allows an operator to manipulate the lenses configuration, which in turn changes the intrinsic camera parameters. Assuming that the camera intrinsic parameters can be obtained as a function of the camera zoom and focus, the intrinsic parameters matrix can be defined as:

$$A = \begin{bmatrix} f_x(\textit{zoom},\textit{focus}) & 0 & c_x(\textit{zoom},\textit{focus}) \\ 0 & f_y(\textit{zoom},\textit{focus}) & c_y(\textit{zoom},\textit{focus}) \\ 0 & 0 & 1 \end{bmatrix} . \tag{3.54}$$

This representation establishes the intrinsic camera parameters as a function of the several configurations of the camera, ensuring a generic formulation of the parameters f_x, f_y, c_x and c_y.

Assuming that the target is identified in a generic pixel (u, v), its is possible to define a pointing vector in a tri-dimensional projective space pointing to the pixel where the target is identified with origin in the

camera optical center. The pointing vector in the camera coordinate frame is given by

$$\vec{P}^C_{Target} = \begin{bmatrix} u - c_x \\ v - c_y \\ \frac{f_x + f_y}{2} \end{bmatrix}, \tag{3.55}$$

and, considering the coordinate frames presented before, the vector $\vec{P}^C_{Target}$ is transformed by:

$$\vec{P}^V_{Target} = \begin{bmatrix} x_{target} \\ y_{target} \\ z_{target} \end{bmatrix} = R^V_B R^B_G R^G_C \vec{P}^C_{Target}, \tag{3.56}$$

where $\vec{P}^V_{Target}$ references the target vector in the UAV-carried inertial coordinate frame (vehicle frame, with the same orientation as the inertial coordinate frame).

Obtaining the target range directly from an aircraft optical sensor is not possible. However, using a flat earth model, it is possible to determine a scale factor to determine the intersection of the target vector with the ground. The scale factor is obtained using the aircraft GPS altitude (measured as the geometric altitude above Mean Sea Level) and is defined as

$$\lambda = \frac{h}{z_{target}}, \tag{3.57}$$

therefore the target location in the UAV-carried reference frame is given by

$$\vec{P}^{UAV}_{Target} = \lambda \begin{bmatrix} x_{target} \\ y_{target} \\ z_{target} \end{bmatrix}. \tag{3.58}$$

The final target location in the inertial ECEF coordinate frame is given by the vector sum of of the vector from the ECEF origin coordinate frame to the aircraft and the aircraft to the target.

$$\vec{P}^I_{Target} = \vec{P}^I_{UAV} + \vec{P}^{UAV}_{Target}. \tag{3.59}$$

After the ECEF coordinates of a target are determined using the formulation of Heikkinen [24] it is possible to convert them into geodetic coordinates useful to a ground operator.

Chapter 4

Experimental Procedures

This chapter details the different experiments realized to assert if the presented methodology is valid. Three different experiments were devised: camera calibration, geo-location in a laboratory environment and geo-location in real-world conditions. These experiments were executed incrementally, building and validating assumptions with the objective of providing the most satisfactory solution to the proposed geo-location problem of this dissertation.

Section 4.1 details the experiment conducted ed to calibrate the camera. The approach of this experiment is based in the works of Strum [25] and Zhang [7], were the features of a planar checker pattern are used to obtain the intrinsic parameters of the camera and their behavior according to its variable settings.

Section 4.2 uses the experimental results of the camera calibration from Section 4.1 and employs them into a small-scale experiment of the geo-location problem. This experiment was conducted in a laboratory environment under control conditions. The experiment detailed in this section attempts to corroborate the values of intrinsic parameters obtained and validate the geo-location methodology.

Section 4.4 details a full-scale experiment in a real-world scenario with the objective to assess the performance of the geo-location methodology under real-world conditions. The geo-location algorithm proposed was tested considering its viability as an online instant solution for the geo-location problem. This section presents the final results of this dissertation and their analysis.

4.1 Camera Calibration

4.1.1 Experiment Procedure

The camera calibration experiment was conducted using the software architecture described in Chapter 2, Section 2.2. This software architecture allowed to acquire several videos were a checker pattern was filmed. The calibration procedure was done offline (by sampling the frames of the acquired videos) with a dedicated python script using the open source library OpenCv [22].

The calibration procedure was employed according to the steps defined by Zhang [7] as follows:

1. A printed pattern was attach to a planar surface;

2. A few sample images of the model plane were collected under different plane orientations;

3. The OpenCv library and python script detected the feature points in the images;

4. The intrinsic parameters and all the extrinsic parameters were estimated;

5. The coefficients of the radial and tangential distortion were determined.

The steps defined by Zhang [7] are valid for fixed camera settings, however in the work of Strum [25] the hypothesis of interdependence between the camera intrinsic parameters and its settings is explored. Therefore in this experiment, the theoretical assumptions of Strum were taken into account to obtain the camera intrinsic parameters as a function of its settings (zoom and focus).

To accomplish a full variable camera settings calibration, several images samples were collected across the camera zoom band with a fixed focus (at "infinity") and with auto-focus. Separating the two focus modes allows the verification of the theoretical assertions of Strum [25], despite the fact that in his experimental procedures the focus setting is not considered.
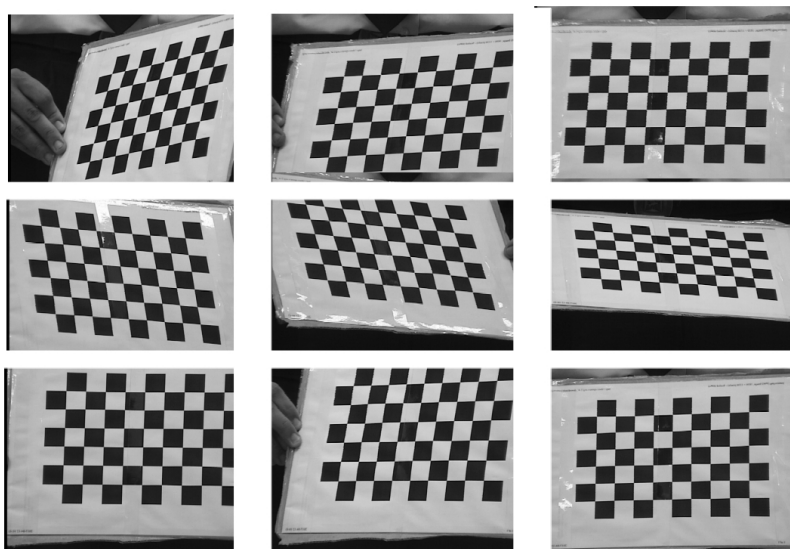
Figure 4.1: Auto-focus samples for calibration.
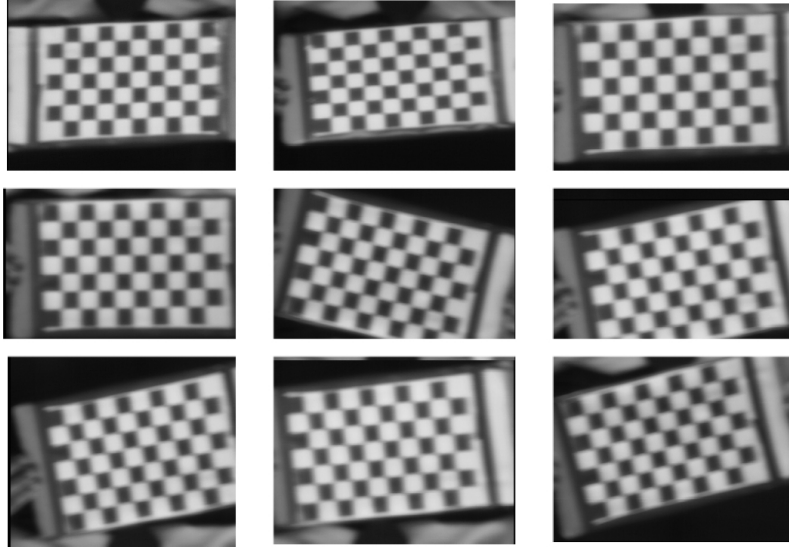
Figure 4.2: Fixed focus samples for calibration.

Figures 4.1 and 4.2 illustrate several samples for both auto-focus and the fixed focus scenario. In both cases, it is possible to obtain the camera intrinsic parameters from the checker pattern features. However, in the case of the fixed focus scenario, with increasing zoom it becomes increasingly difficult to obtain image samples to obtain a calibration. As the zoom increases the image blur becomes more evident and the detection of features in the checker pattern becomes nearly impossible.

4.1.2 Experiment Preparations

The process of collecting image samples for both scenarios illustrated in Figures 4.1 and 4.2 across the entire zoom band is a very time-consuming task [25]. Therefore it is important to assess how the amount of samples affect the computation time of the calibration process in order to make the entire process as much time efficient as possible.

Figure 4.3 illustrates how the required computing time increases in relation to the number of samples provided to the calibration algorithm. In the contrary, the results of the calibration algorithm start to settle around the 35 samples per calibration. This assessment illustrates the fact that even if 35 or more samples are used the calibration results will not be more accurate. Considering that the result of the calibration algorithm only depends on the samples provided and the computing time increases with the number of samples, not leading to a more precise result, a moderate value of 40 image samples per calibration was selected.

This initial assessment test shows that the accuracy of the camera intrinsic results are directly linked to the image samples used. Across the zoom band, a total of 10555 image samples for a fixed focus and 50802 image samples for auto-focus were obtained. The discrepancy in the numbers of image samples verified between the two scenarios is due to image blur when using a fixed focus setting, as observed in Figure 4.2. In this case, some of the collected samples present a high level of blur, making it very difficult to extract the feature points of the checker pattern and leading to a smaller amount of image samples

usable for calibration.

Given the number of image samples collected for both scenarios and the fact that the calibration only depends on the images given to the calibration algorithm, a statistical approach was taken into consideration. This statistical interpretation of the camera intrinsic parameters given by the calibration procedure has two distinct objectives: minimize the influence of the image samples in the calibration and access the performance of the calibration algorithm.

The first objective is accomplished by providing different image samples of each zoom level several times to the calibration algorithm. This ensures that a large variety of images are tested and minimizes the influence of a single image (or group of images) in the calibration results. The second objective is to verify the calibration algorithm performance varies when, for a same scenario, different groups of images are provided and the results of the different values obtained for the intrinsic parameters are compared. Given the fact that all camera settings are the same and only the images samples are different, similar results must be expected regardless of the image samples provided.

For each zoom level, 50 separate calibrations were conducted using 40 randomly selected images samples for the corresponding zoom level. This provided a database of values across the entire zoom band with more than one value for each intrinsic parameter from a multitude of combinations of image samples. The data for the database was collected for both fixed and auto-focus.

Figures 4.4 and 4.5 are the results of the calibrations conducted with a fixed focus scenario. In Figure 4.4 it is possible to verify that the samples average principal point coordinates c_x and c_y, regardless of the zoom level, are more or less consistent with the image center being located at the pixel coordinates $(u, v) = (320, 240)$. In Figure 4.5 it is possible to verify that the focal lengths f_x and f_y gradually increases with the zoom level, however it appears that a ratio of 1:1 is present between f_x and f_y.
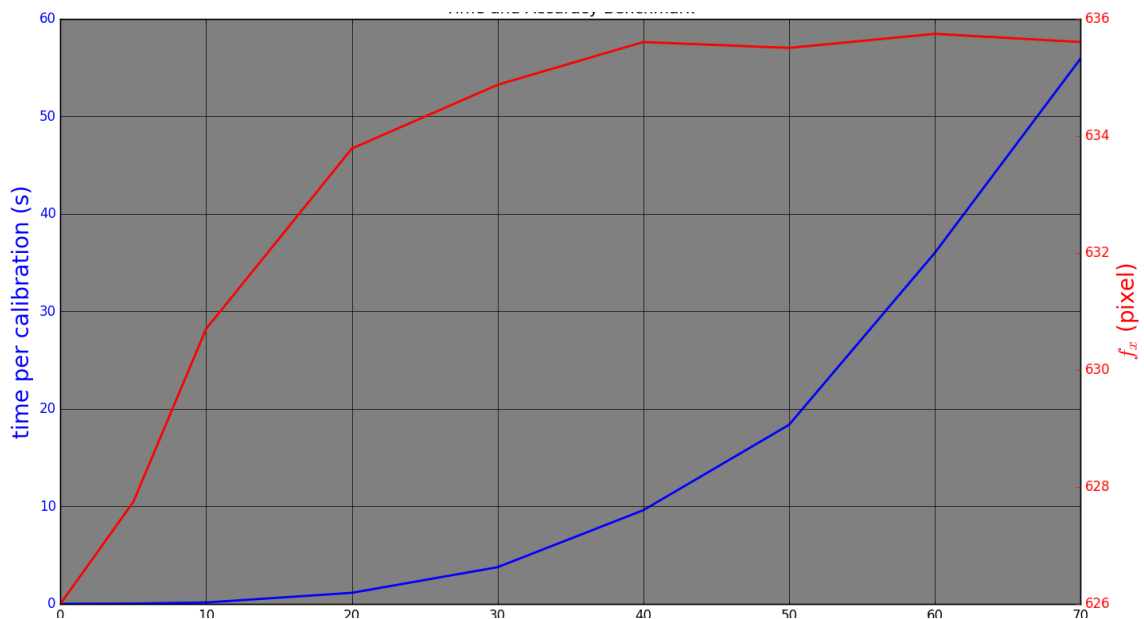
Figure 4.3: Calibration time (seconds) for the focal length at 0% zoom level.
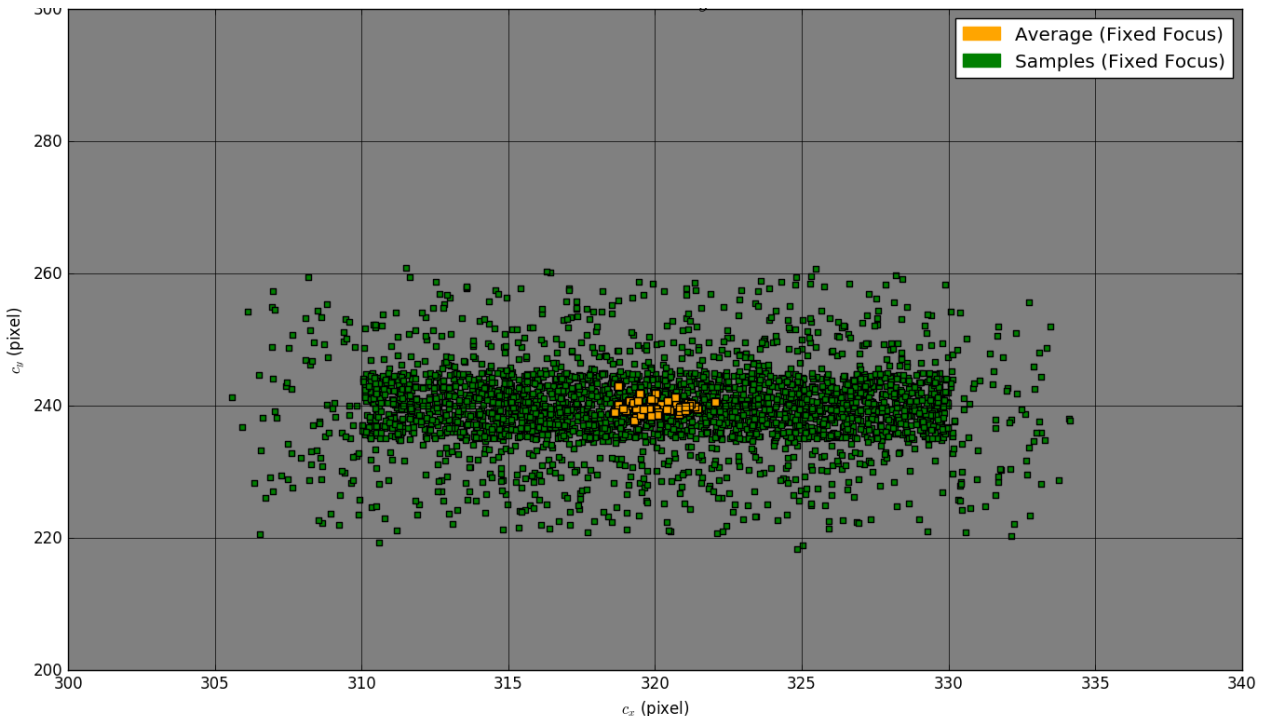
Figure 4.4: Principal point (c_x and c_y) variation through the zoom range. Regardless of the zoom level, the average of the samples collected present values of the principal close to the image center.
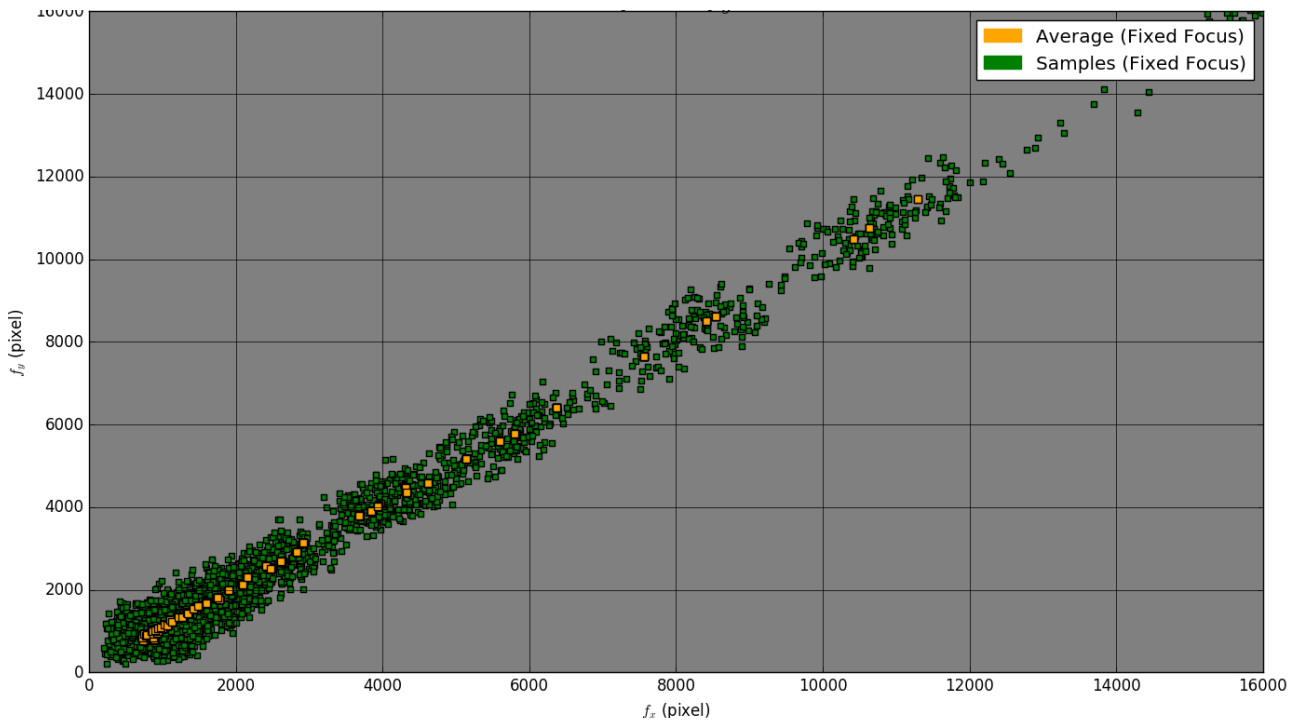
Figure 4.5: Focal length f_x and f_y variation through the zoom range. The focal length increase with the zoom level (lower left corner is at $= 0\%$ zoom and top right corner is at 100% zoom.

49

Figures 4.4 and 4.5 reveal some interesting results respectively regarding the location of the principal point and the focal length which enable the possibility of using two calibration flags from the OpenCv toolbox. The flags of the OpenCv library can be set when certain relations between some of the intrinsic parameters are observed and help to reduce the computing time of the calibration algorithm.

Given that the principal point does not exhibit significant changes, rather fluctuates its position at the image center and the focal length present a ratio of 1:1, in an attempt to decrease the total computing time two OpenCv toolbox flags were selected: **CV_CALIB_FIX_PRINCIPAL_POINT**, which fixes the central point at $(u, v) = (320, 240)$ and **CV_CALIB_FIX_ASPECT_RATIO**, which forces a ratio between the focal lengths during the calibration.

This initial assessment provided a more time efficient approach. If not conducted, each calibration would take around 37.5 seconds which could lead to upwards of 30 minutes to calibrate each zoom level. Considering that a total of 62 zoom levels were tested, this amounts to approximately 32 hours of computing time just for one scenario of the two considered. Using the proposed flags, the entire calibration procedure for each scenario in question, across the entire zoom band, was done in approximately 24 hours.

4.1.3 Experimental Results

One of the indicators of a camera calibration quality is the re-projection error. Defined as the geometric error corresponding to the image distance between a projected point in the image plane and the measured one
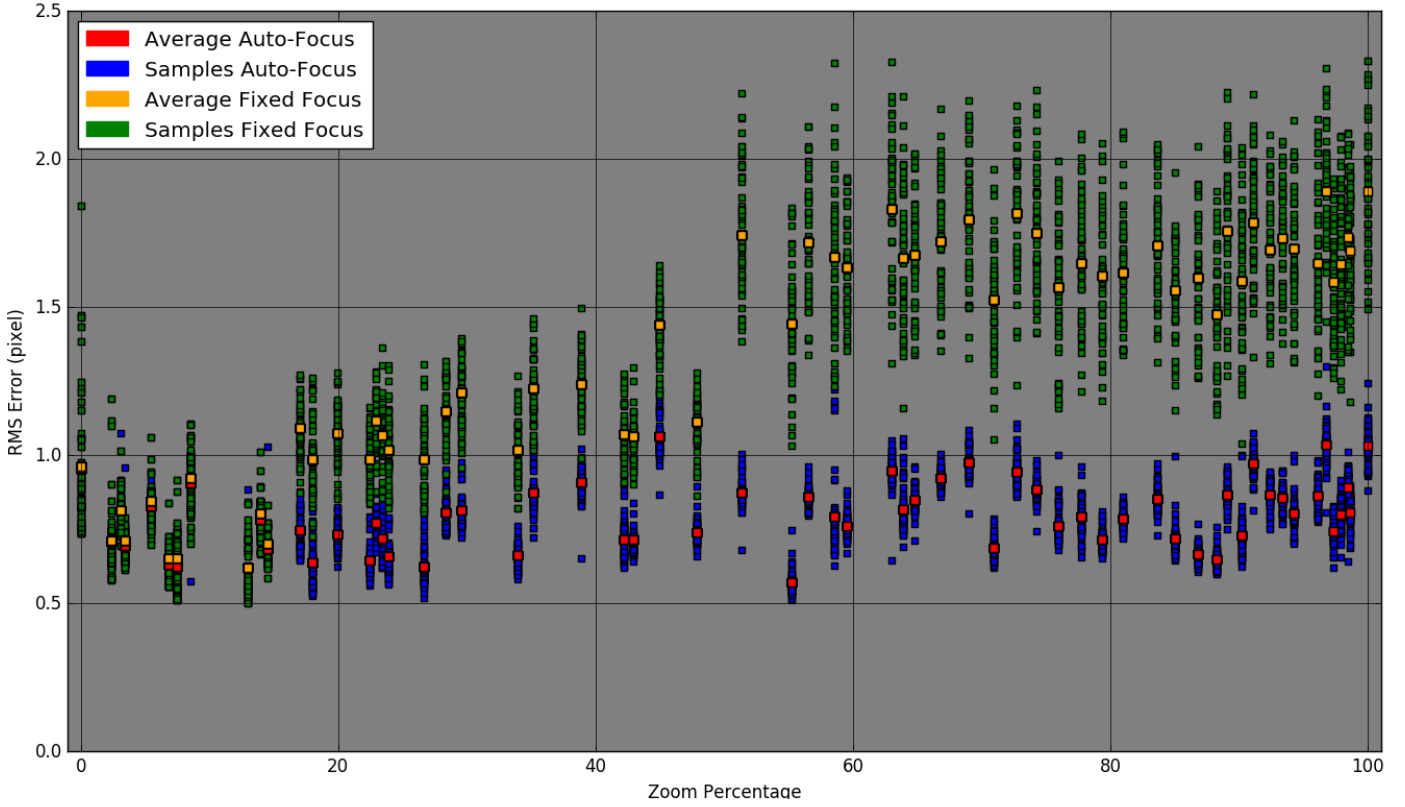
Figure 4.6: Re-projection RMS error through the zoom range.

Establishing that the points in the calibration procedure are perfectly measured in a first image and, considering only the errors in a second image. The more appropriate way to analyze the re-projection error is to minimize the transfer function given by:

$$\sum_i d(x_i', H\bar{x}_i)^2,$$ (4.1)

this represents the Euclidean image distance in the measured points x_i' and the corresponding value of the points $H\bar{x}_i$ corresponding to the first image.

In a cases where image measurement error occur in both images, it is ideal to minimize those errors in both images. One way of obtaining a more realistic error function is by:

$$\sum_i d(x_i, H^{-1}x_i')^2 + d(x_i', Hx_i)^2,$$ (4.2)

where it is considered the forward (H) and backward (H^2) transformation and the sum of the geometric errors in the first and second image, respectively the terms of the sum.

A more simplistic approach is to conceptualize the re-projection root mean square error as the sum of squared distances between the observed projections of the image points and the projected pattern points into the image plane. In Figure 4.6 the results of the calibration for the entire zoom band are presented, where it is possible to verify the different behaviors for the two scenarios considered.

In Figure 4.6 when comparing the auto-focus scenario with fixed focus scenario it is possible to verify that it exhibits a better RMS error throughout the zoom band, it is also possible to verify that the value of the samples of the calibration presents a smaller standard deviation. This fact is inferred from the sample values dispersion around their respective average.

One other fact that can be pointed is the fact that the fixed focus scenario presents an increase in the RMS re-projection error as the zoom level increases, eventually stabilizing. This is due to the fact that image samples collected at a higher zoom level suffer from an increasing blur. Due to the degraded image quality, the calibration algorithm performance is affected leading to higher re-projection errors.
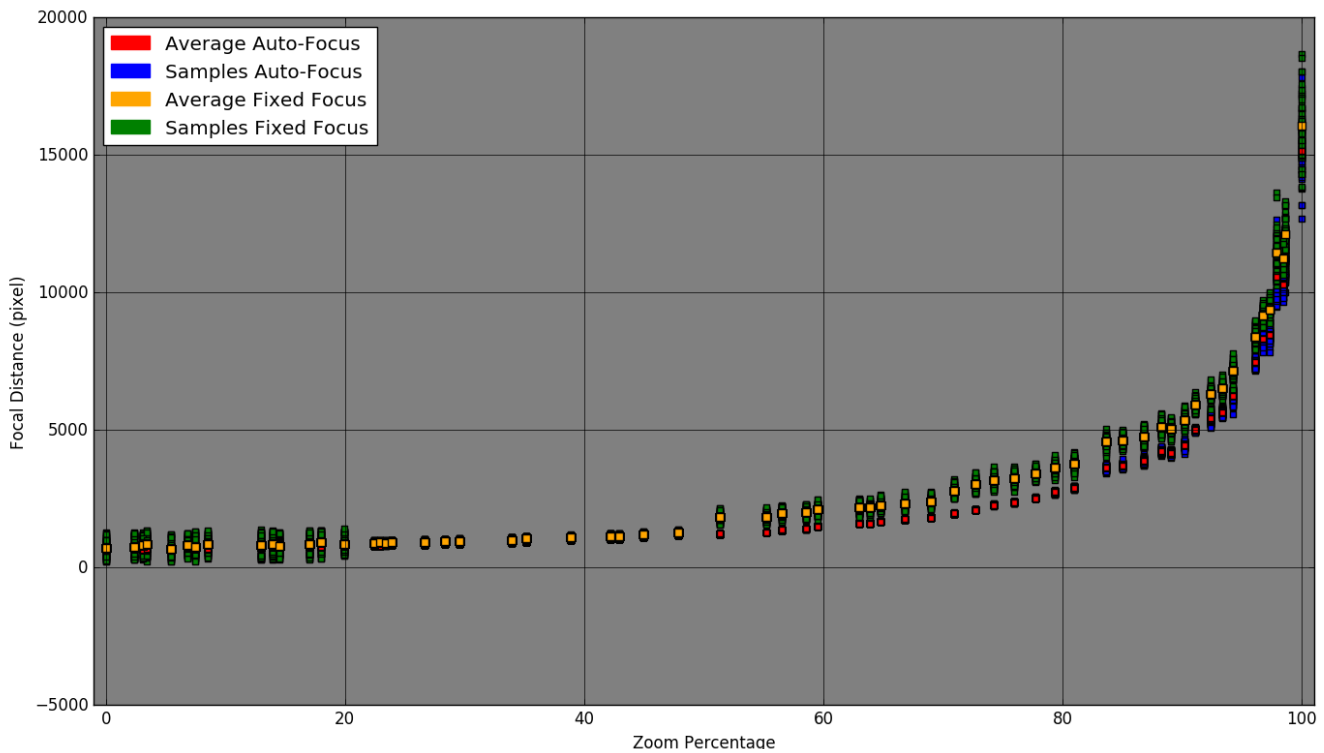
Figure 4.7: Focal length through the zoom range.

Figure 4.7 represents the focal length in pixel units across the zoom band. It is possible to observe that the higher zoom levels the higher the focal length. Between the two scenarios studied, we see that the values of the focal length start very similarly in both case. However, as the zoom increases and passes the 50% mark, the fixed focus scenario starts to exhibit higher values. This once again be justified by the image quality of this scenario, where the image blur induces noise in the results from the calibration algorithm. Another indicator of the noise in the results is the difference in the dispersion of the values between the considered scenarios. As in the RMS re-projection error, the fixed focus scenario shows a higher standard deviation of the samples when compared to the auto-focus.

52
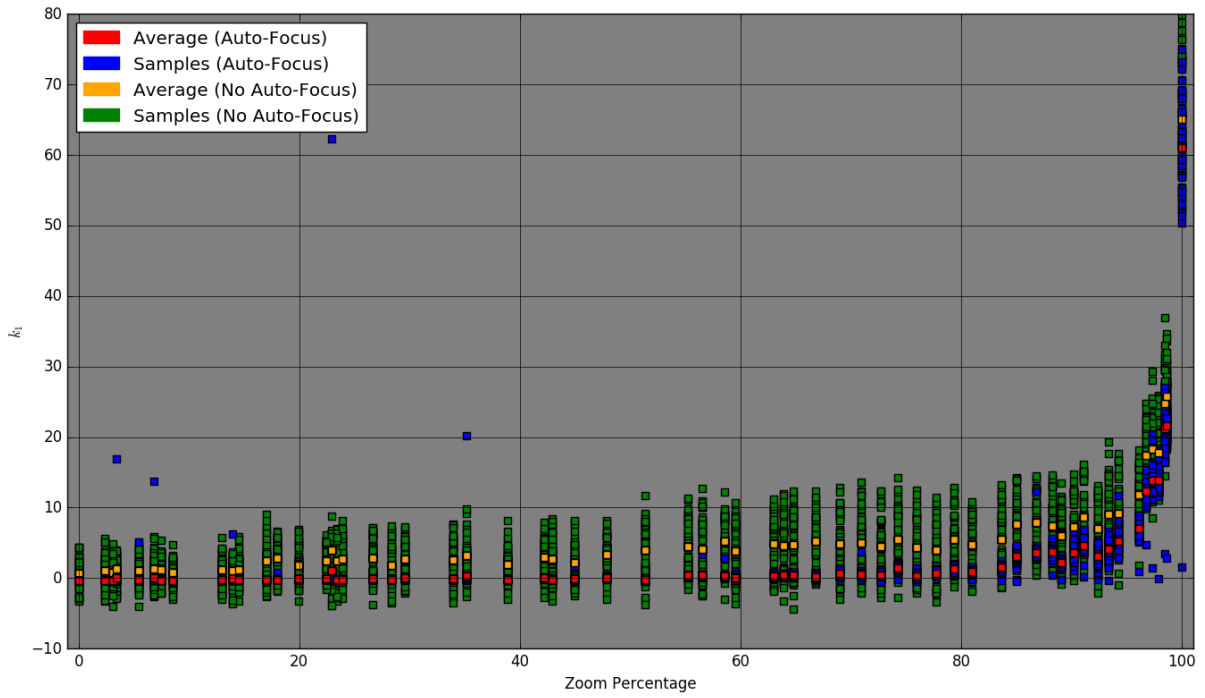
Figure 4.8: Radial distortion parameter k_1 across the zoom range.
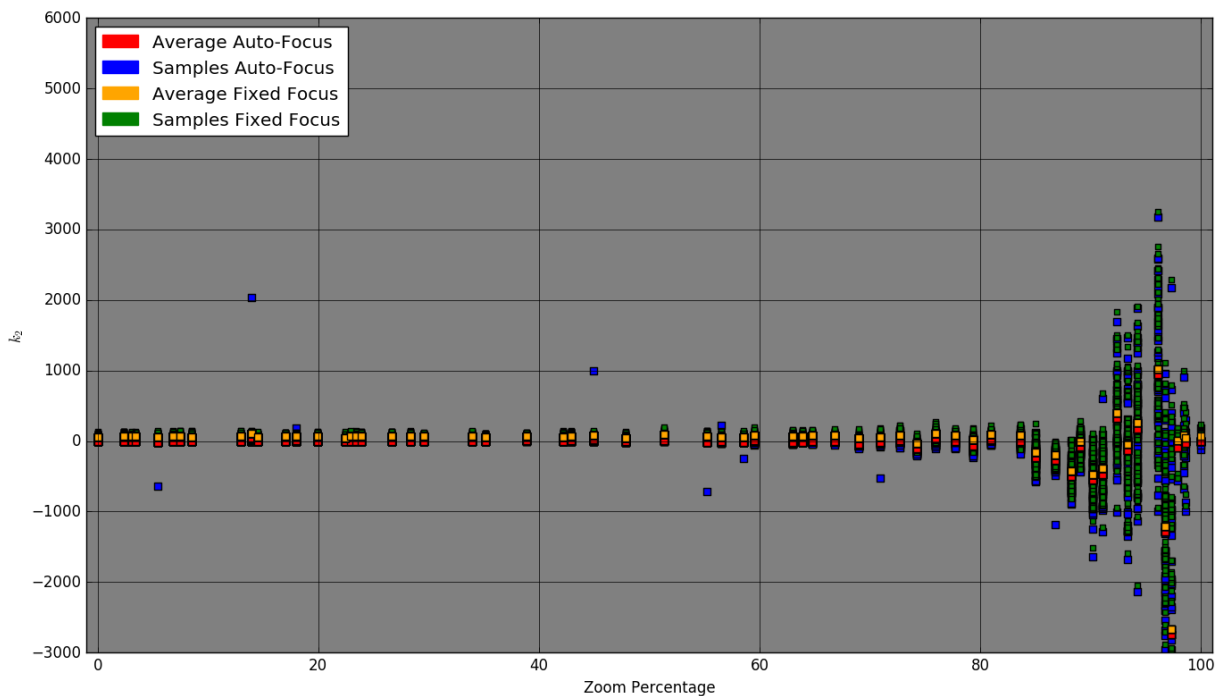
Figure 4.9: Radial distortion parameter k_2 across the zoom range.

Figures 4.8, 4.9 and 4.10 illustrate several radial distortion parameters of the camera lenses. From the data collected it is possible to verify that in Figure 4.8 the parameter k_1 exhibits a very defined behavior were the higher the zoom level the higher its value is. Similarly to the focal length, the data collected between the two scenarios considered presents a higher dispersion of values in the fixed focus

53
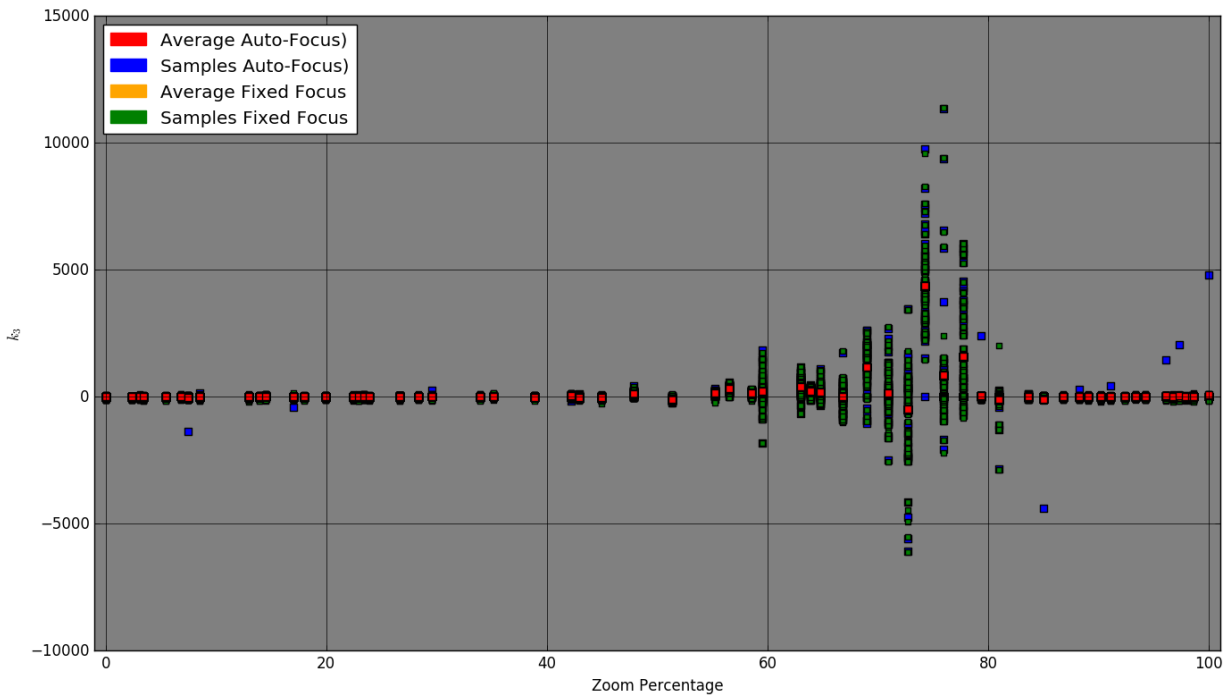
Figure 4.10: Radial distortion parameter k_3 across the zoom range.

scenario when comparing to the auto-focus. The average values also start very similar, however as the zoom increases the average results diverge between the scenarios.

Parameters k_2 and k_3, respectively Figures 4.9 and 4.10 present a more constant values throughout the first half of the zoom band. However, both show erratic results in the range of 80%-100% and 60%-80%, respectively. The data suggests that the calibration algorithm is overloaded, trying to obtain accurate values for very small changes and leading to some unpredictability. This results are revealed to be very hard to model as a continuous function in the ranges affected by the sudden changes in value.

Figures 4.11 and 4.12 present the tangential distortion parameters. These parameters show a constant value close to zero in both cases. Once again the standard deviation of the samples in the case of the fixed focus is considerably higher than the auto-focus scenario, and the values of the average are not so well established as zero, rather they fluctuate around the zero value.
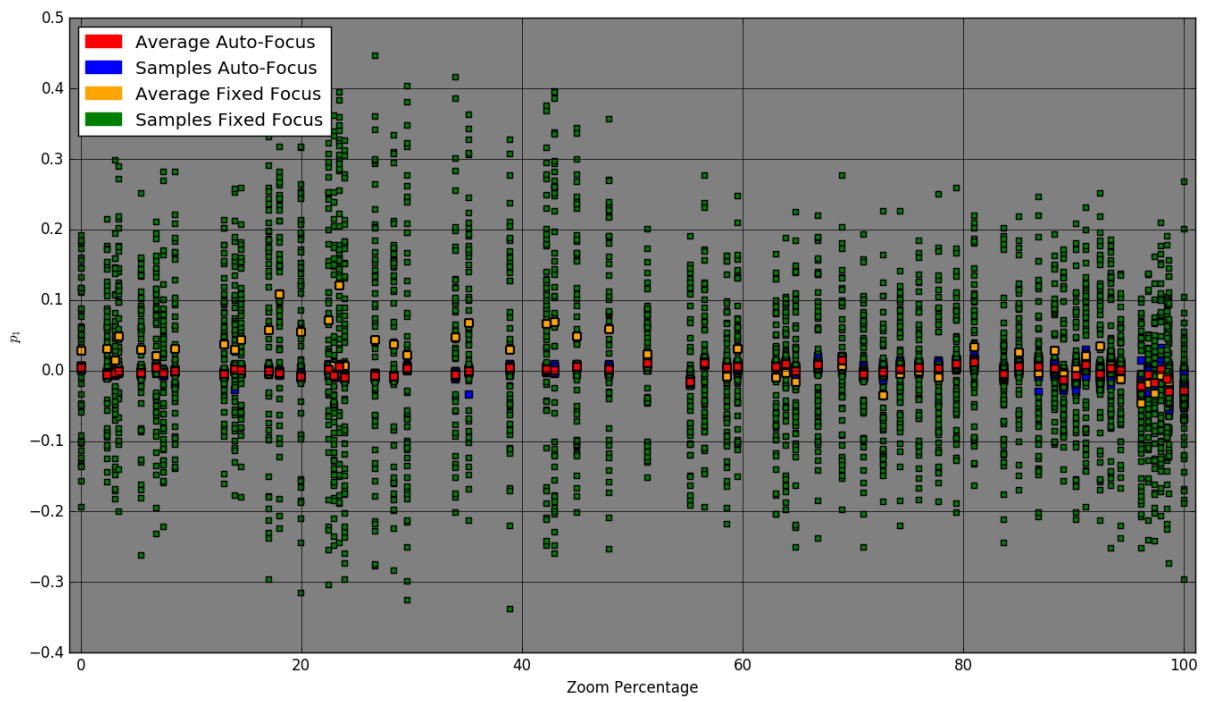
Figure 4.11: Tangential distortion parameter p_1 through the zoom range.
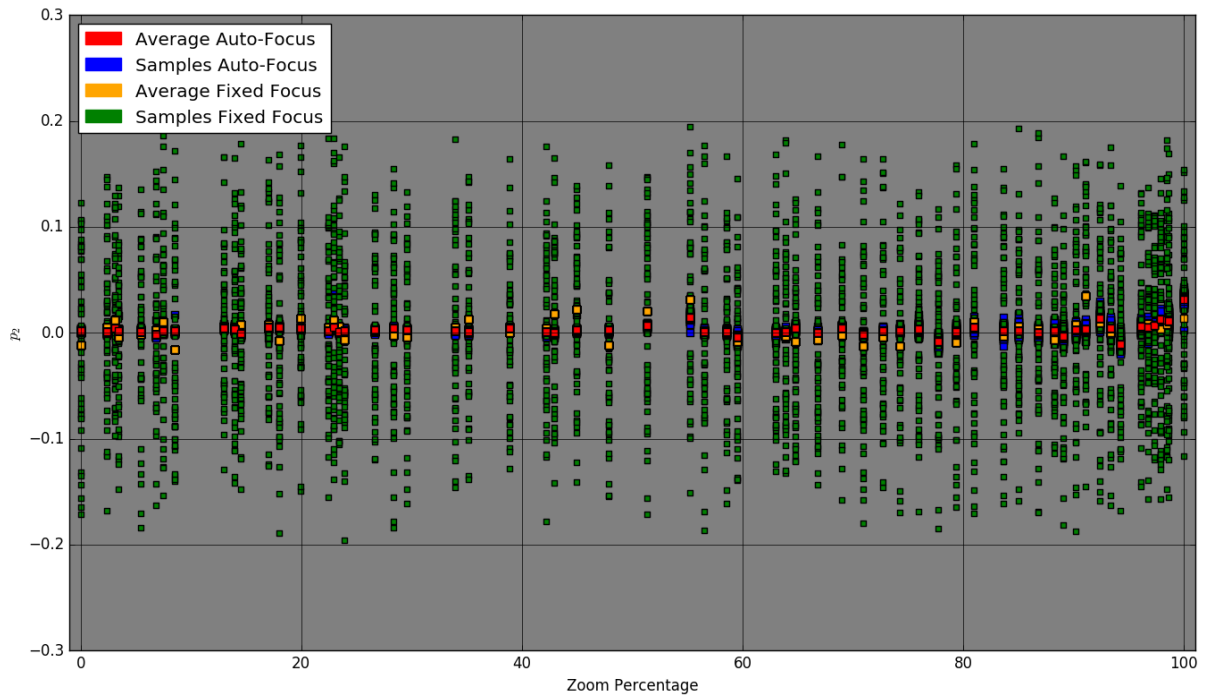
Figure 4.12: Tangential distortion parameter p_2 through the zoom range.

4.1.4 Discussion of Results

The results from the calibration procedure, regardless of the scenario considered, point to a very good camera equipment with a very robust lens assembly practically free of distortion. The statistical analysis conducted verifies the performance of the calibration conducted where, regardless of the images provided, the results are relatively consistent from one calibration to other.

The results of the camera intrinsic and distortion parameters are obtained in pixel units. One drawback encountered when calibrating the camera is the fact that, despite the OpenCv tutorials consulted [26], the introduction of the checkerboard squares size does not influence the calibration results. Several values of the square size and different programs [27] (using the Heikkila [6] method) were tested presenting the similar results in pixel units. Nonetheless, the OpenCv documentation was followed during the calibration procedure.

In Figures 4.7 and 4.8, the data leads to the conclusion that is possible to obtain a polynomial model of the respective intrinsic parameters focal length and k_1 radial distortion parameter across the zoom range in both scenarios studied. The calibration values of the distortion parameters k_2 and k_3 across the zoom band were assumed to be zero due to two factors. First, their values were most of the cases zero, holding no influence over the distortion parameters. Second, their erratic behavior in some zoom ranges reveals a difficult to emulate their behavior as a continuous function. Therefore, they were discarded with the conscience of their possible impact on future results. The tangential parameters p_1 and p_2 present a value close to zero, therefore similarly to k_2 and k_3 they were not considered in the following experiments.

The re-projection RMS error comparison between the scenarios tested is a clear example of the increasing difficulty of obtaining good data for higher zoom level in the case of the fixed focus. Compelling the belief that the calibration values obtained for a fixed focus scenario should not be considered valid, and the adoption of the auto-focus is a better option.

The Appendix A presents a Table with the values used to obtain the models presented in Figures 4.13 and 4.14. Comparing both scenarios, it is possible to verify that, although the mean (μ) at several zoom levels is similar, the standard deviation (σ) of the samples is considerably higher for the fixed focus scenario. The fact that the standard deviation is higher is also a clear indicator that, although the samples averages results are close using a fixed focus is not a desirable option.

The adoption of the auto-focus does not exclude the fact that the focus settings has influence in the camera intrinsic parameters, in fact, Strum [25] theoretically details how this can be modeled and integrated into the interdependence modeling process. However, given the experimental results obtained and the camera equipment, the effect of the focus (as in [25]) is neglected and the camera device is configured to operate in auto-focus.

A generic polynomial function can be described as

$$p(x) = \sum_{i=0}^{n} a_i x^i,$$

(4.3)

where the polynomial coefficients are $a_0 \dots a_n$. The polynomial functions presented in Figures 4.13 and

4.14 model the values of both the focal lengths and distortion parameter k_1 as a function of the zoom. The models were obtained using the data from Table in Appendix A.2 for the auto-focus scenario and their coefficients are presented in Tables 4.1 and 4.2.

Considering the polynomial models which held the best results from the Figures 4.13 and 4.14, and the principal point is centered in image center with coordinates $(u, v) = (320, 240)$. The representation of the intrinsic parameter matrix is given by

$$A = \begin{bmatrix} f(\textbf{\textit{zoom}}) & 0 & 320 \\ 0 & f(\textbf{\textit{zoom}}) & 240 \\ 0 & 0 & 1 \end{bmatrix},$$

(4.4)

and

$$u = fx'' + 320,$$

(4.5)

$$v = fy'' + 240,$$

(4.6)

$$x'' = x' \frac{1 + k_1 r^2}{1},$$

(4.7)

$$y'' = y' \frac{1 + k_1 r^2}{1}.$$

(4.8)
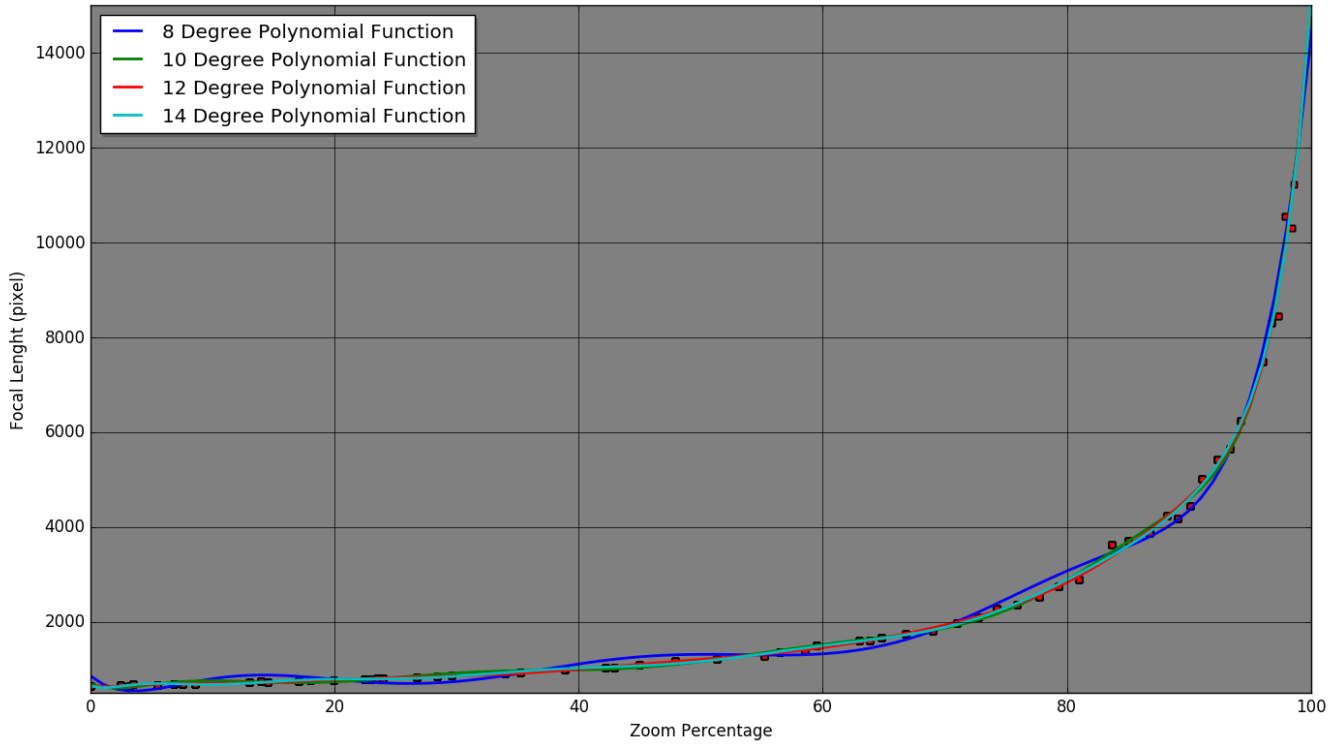
Figure 4.13: Focal length polynomial models.

$f\left(Focal\,Length\right)$				
n	8	10	12	14
a_0	8.57498626e+02	6.95738881e+02	6.51654427e+02	6.47391330e+02
a_1	-2.17075797e+02	-1.10607854e+02	-3.47745891e+01	-9.10040760e+01
a_2	4.93086359e+01	4.22074087e+01	2.20022794e+01	6.91023729e+01
a_3	-4.26675536e+00	-5.87743667e+00	-4.45564203e+00	-1.82927334e+01
a_4	1.82510690e-01	4.15897323e-01	4.63509835e-01	2.50274317e+00
a_5	-4.23934993e-03	-1.69069427e-02	-2.84287052e-02	-2.04714494e-01
a_6	5.44876321e-05	4.18416617e-04	1.10586143e-03	1.08447490e-02
a_7	-3.64086499e-07	-6.40574881e-06	-2.83310726e-05	-3.89908722e-04
a_8	9.87225734e-10	5.92135810e-08	4.84433353e-07	9.76778274e-06
a_9		-3.02710356e-10	-5.47446076e-09	-1.72341626e-07
a_{10}		6.56931179e-13	3.92598243e-11	2.13335870e-09
a_{11}			-1.61857584e-13	-1.81340688e-11
a_{12}				

Table 4.1: Focal length polynomial models coefficients
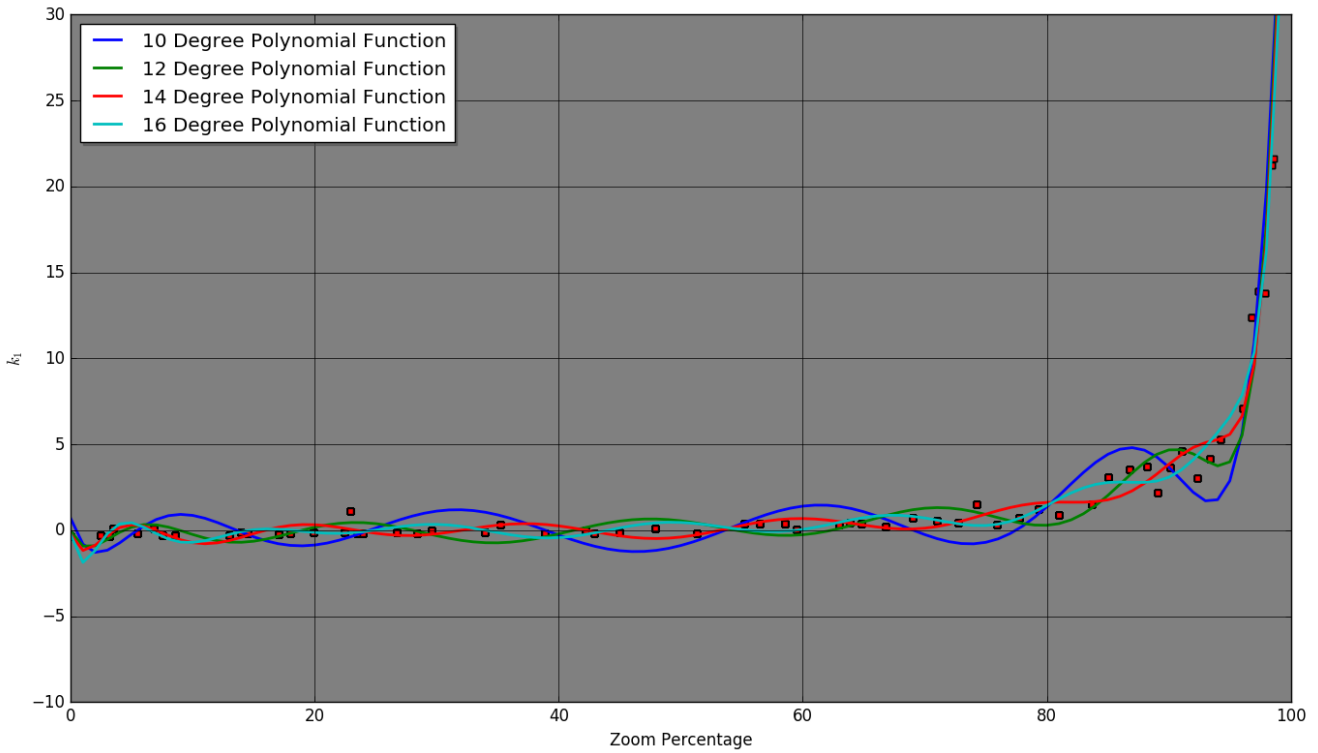
Figure 4.14: Radial distortion parameters k_1 polynomial models.

$k_1(Distortion\,Parameter)$				
n	10	12	14	16
a_0	6.71508346e-01	-5.54348161e-02	-2.45370388e-01	-2.70641172e-01
a_1	-2.16898243e+00	-1.55997709e+00	-2.05450262e+00	-3.95311431e+00
a_2	7.82013271e-01	8.44771000e-01	1.45390097e+00	3.18859805e+00
a_3	-1.07854084e-01	-1.70515101e-01	-3.80361769e-01	-9.89378767e-01
a_4	7.54067939e-03	1.76023280e-02	5.15390412e-02	1.64316730e-01
a_5	-3.02260823e-04	-1.06767965e-03	-4.18904852e-03	-1.68339994e-02
a_6	7.35961009e-06	4.09500214e-05	2.21257815e-04	1.15011237e-03
a_7	-1.10654217e-07	-1.03195565e-06	-7.95586483e-06	-5.49341690e-05
a_8	1.00319924e-09	1.73216170e-08	1.99843912e-07	1.88890056e-06
a_9	-5.02508765e-12	-1.91787362e-10	-3.54294401e-09	-4.75588213e-08
a_{10}	1.06775157e-14	1.34506655e-12	4.41392931e-11	8.83125965e-10

Table 4.2: Radial distortion parameters k_1 polynomial models coefficients.

4.2 Geo-location in Laboratory Environment

4.2.1 Experiment Procedure

Before employing the geo-location algorithm in a full-scale scenario, some hypothesis must be verified in a small scale scenario. The first objective of this experiment is to verify the validity of the several polynomial models obtained from the previous experiment and determine the combination of focal length and distortion parameters models with better results. The second objective is to verify if the proposed methodology is a valid approach to the geo-location problem.

To accomplish the objectives proposed in this experiment, a simulated scenario was assembled. Several targets were placed in known stationary positions and the camera recorded the targets from a stationary position, where Figures 4.15 and 4.16 illustrate the experimental scenario proposed.
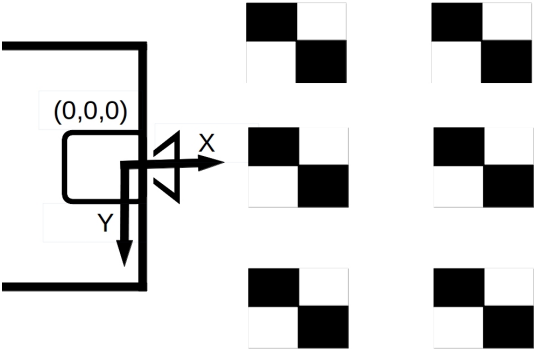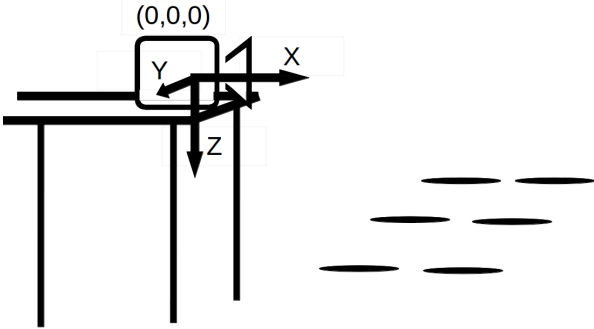
Figure 4.15: Top view   Figure 4.16: Side view

The targets shape emulate the checker pattern and a total of 12 were printed and placed on the floor, as illustrated by Figure 4.17. This experiment sets the perfect scenario for testing both the camera calibration and the paper-map-base geo-location method. The targets were placed in specific known locations on the inertial reference frame and there are no environmental effects (e.g. wind, image glare, sun reflections).
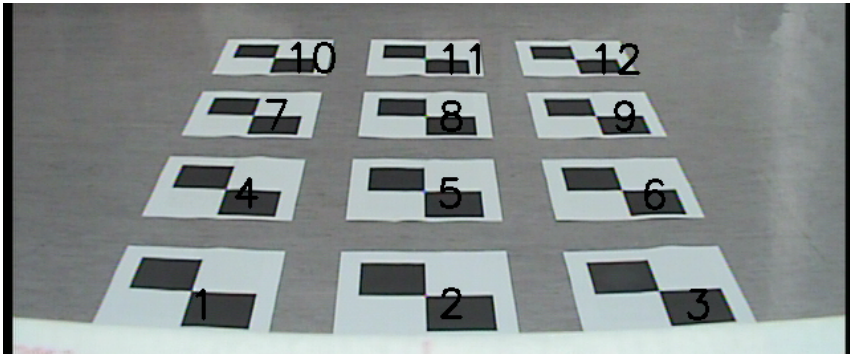
Figure 4.17: Example of the experimental setup. The targets are numbered from 1 to 12 for future reference.
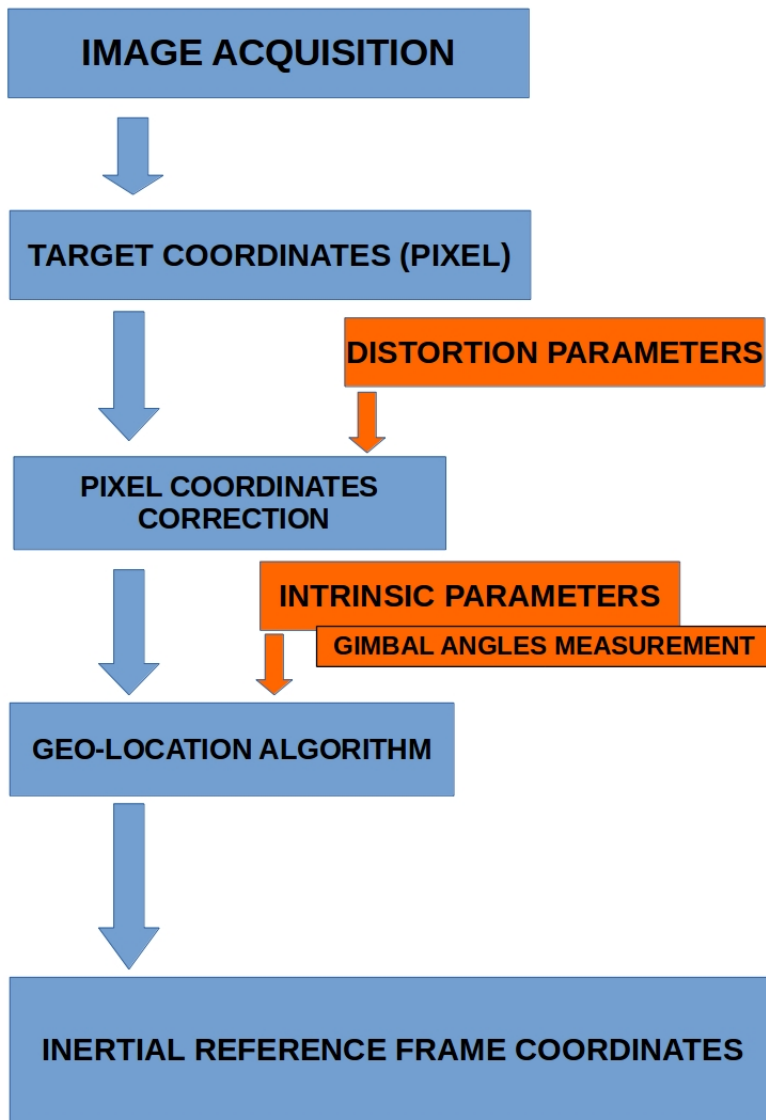
Figure 4.18: Experimental software diagram.

Figure 4.18 illustrates how the intrinsic and distortion parameters are combined with the geo-location algorithm. First, an image sample is acquired, from which the several targets pixel coordinates are determined with a basic template matching detection algorithm. The distortion parameters (according to the zoom level) are calculated using one of the models presented in Figure 4.14. After the distorted target pixel coordinates are corrected to obtain the "true" coordinates (effectively compensating for the camera lens distortion), the camera intrinsic parameters (according to the zoom level) and measured angles of the gimbal camera are supplied to the geo-location algorithm that determines the inertial reference frame target coordinates.

4.2.2 Experiment Preparations

The first preparations concern the geometric location of the perspective center of the camera. This is the point where no parallax errors occur. Depending on the lenses design, the geometric perspective center point located on the optical axis may be behind, within or in front of the lens system and even at an infinite distance from the lens in the case of telecentric systems.

The perspective center point represents the point in a lens for which when rotating the camera or the panoramic head, image stitchings do not appear in sequenced captured images. Therefore, for the purposes of this experiment, it can be interpreted as the geometric center of the lenses assembly, corresponding to the focus point of the "pin-hole" camera model previously.

To determine the perspective center point position two objects must be placed vertically in front of the camera. In the laboratory environment, the legs of a table where used for this purpose. Looking at the images captured in Figure 4.19, the camera was rotated so that the two objects were located at the right of the image frame. Afterwards, the camera was rotated so that the objects are on the left side of the image. Repeating this procedure while translating the camera back and forward relative to the rotation axis there is a point in which the two objects selected (legs of the table) become perfectly aligned in the image regardless of the camera orientation.

Figure 4.19: Perspective center point determination steps.

The perspective center was marked in the camera mount as Figure 4.20 presents. This point here forward is assumed as the origin of the inertial reference frame concerning this experiment, illustrated as the point $(0, 0, 0)$ in Figures 4.15 and 4.16.
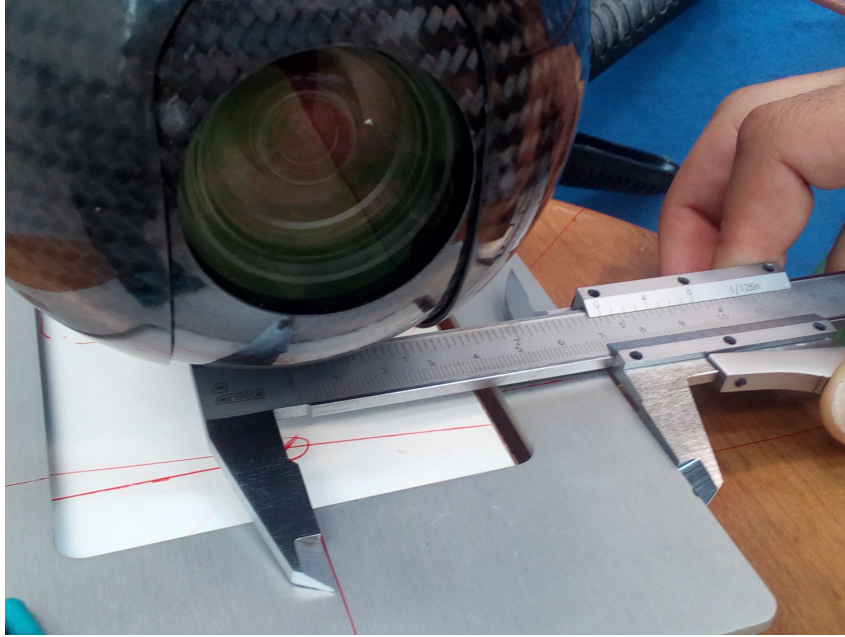
Figure 4.20: Perspective center point markings in the camera mount.

4.2.3 Experimental Results

The experiments conducted cover several zoom levels, as well as diverse camera, poses. Table 4.3 summarizes the test conditions for each test realized. These conditions were purposely varied with the objective of providing a set of different scenarios and comparing them for the several polynomial approaches proposed.

Test Scenario		Test 0	Test 1	Test 2	Test 3	Test 4	Test 5	Units
Test Parameters	Pan	-180	-180	-180	-255	-180	-180	º (Degrees)
	Tilt	-25	-20	-20	-20	-20	-20	º (Degrees)
	Heading	180	180	180	105	180	180	º (Degrees)
	Pitch	0	5	7	10	7	5.2	º (Degrees)
	Roll	3.5	3.5	3.5	2.1	2	3.5	º (Degrees)
	Zoom	0	5	10	20	50	95	% (Percentage)

Table 4.3: Tests parameters summary Table.

To compare the known position of the targets and their estimated positions the target coordinates are given by T and the location estimates are described as E. The location error is given by:

$$\rho = T - E = \begin{bmatrix} x_t & y_t & z_t \end{bmatrix} - \begin{bmatrix} x_e & y_e & z_e \end{bmatrix} = \begin{bmatrix} x & y & z \end{bmatrix}, \tag{4.9}$$

the mean error and root mean square error of the resulting employment of the geo-location algorithm are respectively:

$$\mu_e = \frac{1}{N} \sum_{i=1}^{n} \sqrt{\rho_x^2 + \rho_y^2 + \rho_z^2}, \tag{4.10}$$

$$RMSError = \sqrt{\frac{\sum_{i=1}^{N}(\sqrt{\rho_x^2 + \rho_y^2 + \rho_z^2})}{N}}, \tag{4.11}$$

In Tables 4.4, 4.5 and 4.6 the value of $|\rho|$ is presented according to the test and target considered. The mean error, mean RMS error and standard deviation of the collected samples is also presented. The samples collected are analyzed from a statistical viewpoint due to the several fluctuations induced by the detection algorithm and the angles sensors information of the gimbal camera.

Target	Test 0	Test 1	Test 2	Test 3	Test 4	Test 5	Mean Error	RMS Error
1	0.0269	0.0310	0.0420	NA	NA	NA	0.0333	0.0339
2	0.0510	0.0506	0.0475	NA	NA	NA	0.0497	0.0497
3	0.0180	0.0233	0.0278	NA	NA	NA	0.0230	0.0234
4	0.0447	0.0428	0.0517	0.0270	NA	NA	0.0415	0.0425
5	0.0676	0.0490	0.0478	0.0520	NA	NA	0.0541	0.0547
6	0.0407	0.0564	0.0514	0.0324	NA	NA	0.0452	0.0462
7	0.1052	0.0730	0.1052	0.0839	0.0839	NA	0.0903	0.0912
8	0.1012	0.0956	0.0994	0.0843	0.0885	NA	0.0938	0.0940
9	0.0742	0.1032	0.1030	0.0858	0.0837	NA	0.0900	0.0907
10	0.1531	0.1764	0.1420	0.1570	0.1547	NA	0.1566	0.1570
11	0.1732	0.1281	0.1890	0.1692	0.1771	0.1904	0.1712	0.1724
12	0.1559	0.1934	0.1725	0.1606	0.1591	NA	0.1683	0.1689

Table 4.4: Error vector module ($|\rho|$) when using focal length $14th$ degree model and a $16th$ degree approximation for k_1.

Target	Test 0	Test 1	Test 2	Test 3	Test 4	Test 5	Mean Error	RMS Error
1	0.0270	0.0308	0.0422	NA	NA	NA	0.0334	0.0340
2	0.0524	0.0509	0.0479	NA	NA	NA	0.0504	0.0504
3	0.0187	0.0241	0.0283	NA	NA	NA	0.0237	0.0240
4	0.0449	0.0430	0.0519	0.0270	NA	NA	0.0417	0.0427
5	0.0672	0.0492	0.0472	0.0525	NA	NA	0.0540	0.0546
6	0.0411	0.0565	0.0515	0.0325	NA	NA	0.0454	0.0463
7	0.1053	0.0736	0.1054	0.0841	0.0840	NA	0.0905	0.0914
8	0.1013	0.0954	0.0993	0.0842	0.0883	NA	0.0937	0.0939
9	0.0744	0.1033	0.1032	0.0859	0.0836	NA	0.0901	0.0908
10	0.1528	0.1761	0.1422	0.1572	0.1545	NA	0.1566	0.1569
11	0.1734	0.1278	0.1891	0.1690	0.1773	0.1914	0.1713	0.1726
12	0.1561	0.1960	0.1735	0.1616	0.1571	NA	0.1688	0.1695

Table 4.5: Error vector module ($|\rho|$) when using focal length $12th$ degree model and a $14th$ degree approximation for k_1.

Target	Test 0	Test 1	Test 2	Test 3	Test 4	Test 5	Mean Error	RMS Error
1	0.0270	0.0311	0.0421	NA	NA	NA	0.0334	0.0340
2	0.0509	0.0508	0.0474	NA	NA	NA	0.0497	0.0497
3	0.0183	0.0236	0.0280	NA	NA	NA	0.0233	0.0237
4	0.0448	0.0428	0.0516	0.0273	NA	NA	0.0416	0.0426
5	0.0677	0.0492	0.0482	0.0522	NA	NA	0.0543	0.0549
6	0.0409	0.0562	0.0504	0.0327	NA	NA	0.0450	0.0459
7	0.1055	0.0736	0.1053	0.0834	0.0845	NA	0.0905	0.0914
8	0.1013	0.0958	0.0999	0.0863	0.0889	NA	0.0944	0.0946
9	0.0751	0.1037	0.1033	0.0863	0.0868	NA	0.0910	0.0917
10	0.1533	0.1764	0.1430	0.1573	0.1543	NA	0.1568	0.1572
11	0.1738	0.1283	0.1892	0.1695	0.1767	0.1884	0.1710	0.1722
12	0.1560	0.1936	0.1772	0.1610	0.1594	NA	0.1695	0.1700

Table 4.6: Error vector module ($|\rho|$) when using focal length $10th$ degree model and a $12th$ degree approximation for k_1.

Analyzing the results in Tables 4.4, 4.5 and 4.6 it is possible to conclude that the combination of models regarding Table 4.4 provide the best results for the proposed experiment. Comparing both the mean and RMS errors of the data the combination of the focal length $14th$ degree model and k_1 16_{th} degree provide the smaller position error estimate.

The data presented also reveals two indications. First, the distance increasing of the targets with regard to the inertial frame origin (the point center of projection) results in a higher error. Therefore, the further away a target is located the higher the error and its corresponding position estimate is further away from the real position. Secondly, the data suggests that for different zoom levels the target coordinate errors are similar, and possibly independent of the zoom used.

The first indication reveals some systematic errors. These can occur either from the camera and gimbal sensors measurements or the models obtained from the calibration for the focal length and distortion parameters. The modulation of the camera intrinsic parameters is not an exact value of the parameters, as the manufacture is the only one who might know this information and it is not disclosed in the equipment manual. Therefore, the models obtained should be viewed has an estimate a not has the precise, accurate, and exact models, which describe the interdependence between the camera lenses and its characteristic parameters.

The second indication allows to establish the coherence of the models presented in Figures 4.13 and 4.14. Throughout the zoom levels, the results obtained using the estimated models are very similar and so are the results as presented in Tables 4.4, 4.5 and 4.6, attesting for their validity. The experimental results show that, independently of the zoom level used, the target estimates errors are very similar, validating the hypothesis of using models to emulate the camera parameters.

4.2.4 Discussion of Results

The experiment results reveal a promising outcome for the employment of the geo-location algorithm in a full-scale scenario. Depending on the used polynomial model the results are encouraging. Verifying Figures 4.13 and 4.14 the model functions do not encompass perfectly the intrinsic parameters. However, they provide the best possible approximation of their respective parameters behavior. Because of the oscillations verified (under/over-shooting of these models), the sensor bias of the equipment and detection algorithm, an error in the geo-location is inevitably present.

Based on the observed results (Tables 4.4, 4.5 and 4.6) and the Figures 4.13 and 4.14 with the polynomial models, the combination of models which present the smallest range vector error is the $14th$ degree polynomial model for the focal length and the $16th$ degree polynomial model for the radial distortion parameter k_1.

Target	Target Coordinates			Test 0			Test 1			Test 2		
1	x: 1.1	y: -0.4	z: 1.04	x: 1.09	y: -0.38	z: 1.04	x: 1.092	y: -0.37	z: 1.04	x: 1.099	y: -0.36	z: 1.04
2	x: 1.1	y: 0	z: 1.04	x: 1.09	y: 0.05	z: 1.04	x: 1.116	y: 0.048	z: 1.04	x: 1.115	y: 0.045	z: 1.04
3	x: 1.1	y: 0.4	z: 1.04	x: 1.1	y: 0.418	z: 1.04	x: 1.11	y: 0.421	z: 1.04	x: 1.114	y: 0.424	z: 1.04
4	x: 1.4	y: -0.4	z: 1.04	x: 1.434	y: -0.37	z: 1.04	x: 1.434	y: -0.37	z: 1.04	x: 1.438	y: -0.37	z: 1.04
5	x: 1.4	y: 0	z: 1.04	x: 1.433	y: 0.059	z: 1.04	x: 1.431	y: 0.038	z: 1.04	x: 1.429	y: 0.038	z: 1.04
6	x: 1.4	y: 0.4	z: 1.04	x: 1.436	y: 0.419	z: 1.04	x: 1.451	y: 0.424	z: 1.04	x: 1.441	y: 0.431	z: 1.04
7	x: 1.7	y: -0.4	z: 1.04	x: 1.796	y: -0.36	z: 1.04	x: 1.767	y: -0.37	z: 1.04	x: 1.796	y: -0.36	z: 1.04
8	x: 1.7	y: 0	z: 1.04	x: 1.78	y: 0.062	z: 1.04	x: 1.789	y: 0.035	z: 1.04	x: 1.791	y: 0.04	z: 1.04
9	x: 1.7	y: 0.4	z: 1.04	x: 1.772	y: 0.418	z: 1.04	x: 1.801	y: 0.421	z: 1.04	x: 1.801	y: 0.42	z: 1.04
10	x: 2	y: -0.4	z: 1.04	x: 2.146	y: -0.35	z: 1.04	x: 2.175	y: -0.38	z: 1.04	x: 2.139	y: -0.37	z: 1.04
11	x: 2	y: 0	z: 1.04	x: 2.163	y: 0.059	z: 1.04	x: 2.124	y: 0.032	z: 1.04	x: 2.186	y: 0.034	z: 1.04
12	x: 2	y: 0.4	z: 1.04	x: 2.149	y: 0.446	z: 1.04	x: 2.189	y: 0.441	z: 1.04	x: 2.168	y: 0.439	z: 1.04

Table 4.7: Real and estimated coordinates obtained by using a $14th$ degree polynomial model of the focal length and a $16th$ degree polynomial model of k_1 results for tests 0 to 2 are shown.

Target	Target Coordinates			Test 3			Test 4			Test 5		
1	x: 1.1	y: -0.4	z: 1.04	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA
2	x: 1.1	y: 0	z: 1.04	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA
3	x: 1.1	y: 0.4	z: 1.04	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA
4	x: 1.4	y: -0.4	z: 1.04	x: 1.379	y: -0.38	z: 1.04	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA
5	x: 1.4	y: 0	z: 1.04	x: 1.374	y: 0.045	z: 1.04	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA
6	x: 1.4	y: 0.4	z: 1.04	x: 1.368	y: 0.405	z: 1.04	x: NA	y: NA	z: NA	x: NA	y: NA	z: NA
7	x: 1.7	y: -0.4	z: 1.04	x: 1.781	y: -0.38	z: 1.04	x: 1.781	y: -0.38	z: 1.04	x: NA	y: NA	z: NA
8	x: 1.7	y: 0	z: 1.04	x: 1.77	y: 0.047	z: 1.04	x: 1.775	y: 0.047	z: 1.04	x: NA	y: NA	z: NA
9	x: 1.7	y: 0.4	z: 1.04	x: 1.78	y: 0.431	z: 1.04	x: 1.781	y: 0.421	z: 1.04	x: NA	y: NA	z: NA
10	x: 2	y: -0.4	z: 1.04	x: 2.151	y: -0.36	z: 1.04	x: 2.148	y: -0.36	z: 1.04	x: NA	y: NA	z: NA
11	x: 2	y: 0	z: 1.04	x: 2.162	y: 0.049	z: 1.04	x: 2.171	y: 0.046	z: 1.04	x: 2.185	y: 0.045	z: 1.04
12	x: 2	y: 0.4	z: 1.04	x: 2.158	y: 0.429	z: 1.04	x: 2.156	y: 0.431	z: 1.04	x: NA	y: NA	z: NA

Table 4.8: Real and estimated coordinates obtained by using a $14th$ degree polynomial model of the focal length and a $16th$ degree polynomial model of k_1 results for tests 3 to 5 are shown.

Tables 4.7 and 4.8 present the detailed results of the experiment detailing each measured coordinate and test conducted for the $14th$ degree polynomial model for the focal length and the $16th$ degree polynomial model for the radial distortion parameter k_1. The fact that the distortion parameters k_2, k_3, p_1, p_2 were discarded apparently do not exhibit a a significant influence in errors obtained in this experiment. The fact it was assumed that the discarded parameters were not significant would lead to believe that a higher error would be evident however at the scale of this experiment, the results show otherwise.

The focus setting, however, is a camera setting with a small but evident presence. In every test and target considered it appears that a small deviation to the right in the y axis is present. The deviation can be attributed to the non-modulation of the focus setting. As stated by Strum [25] this would cause a translation of the central point (even if mainly due to the zoom setting). Accounting for the small change in the central point due to the focus change could lead to a more favorable result and possibly eliminate the deviation verified in the y axis.

In summary, the results of this experiment allowed to verify the validity of the calibration procedure proposed as well as the geo-location algorithm and determining the more favorable model of the camera parameters for employment in a full-scale scenario. The results revealed to be encouraging and, although errors were verified, their presence is inevitable due to the choices and assumptions made for the camera intrinsic parameters modulation.

4.3   Geo-location Module in the SEP Software Architecture

The proposed integration of the software developed in this dissertation can be illustrated by Figure 4.21, were the geo-location module is integrated into the sep architecture using the ROS middleware.
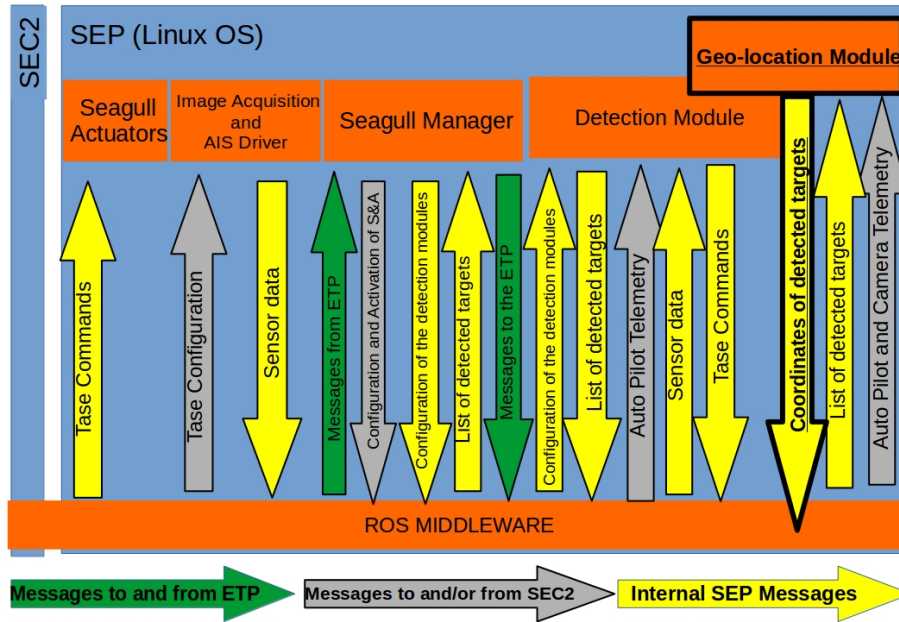
Figure 4.21: SEP software architecture with geo-location module.

The proposed software architecture is illustrated in Figure 4.21. In this architecture, in contrast to the experimental architecture, all the SEP modules are enabled and the Tase Features module is replaced with the Geo-location module. This module is defined as a ROS node and accesses to the memory to load the camera models pre-determined in the camera calibration procedure. The geo-location module receives a list of targets and their pixel coordinates from the detection module as well as the aircraft and camera telemetry and outputs a list of the coordinates of the identified targets.

The methodology employed was already extensively detailed in chapter 3. Figure 4.22 summarizes the geo-location algorithm and implementation in the context of the software architecture. The camera models stored in memory determined in the calibration procedures are combined with a target list in the Camera Frame sub-module, this defines the targets detected in the camera coordinate frame defined by the coordinates (u, v, f), respectively the pixel horizontal and vertical coordinates and the focal length given by the model for the corresponding zoom. In this sub-module, a distortion correction is applied by using the k_1 radial distortion parameter model.

The telemetry data of the aircraft pitch, roll, yaw, altitude above ground level (AGL) and GPS position together with the angles (pan and tilt) of the gimbal camera are combined to obtain the ECEF coordinates of the target (in meters). However to an operator they are not very useful. Therefore, they are transformed into geodetic coordinates latitude, longitude, and altitude before being published in a ROS topic.
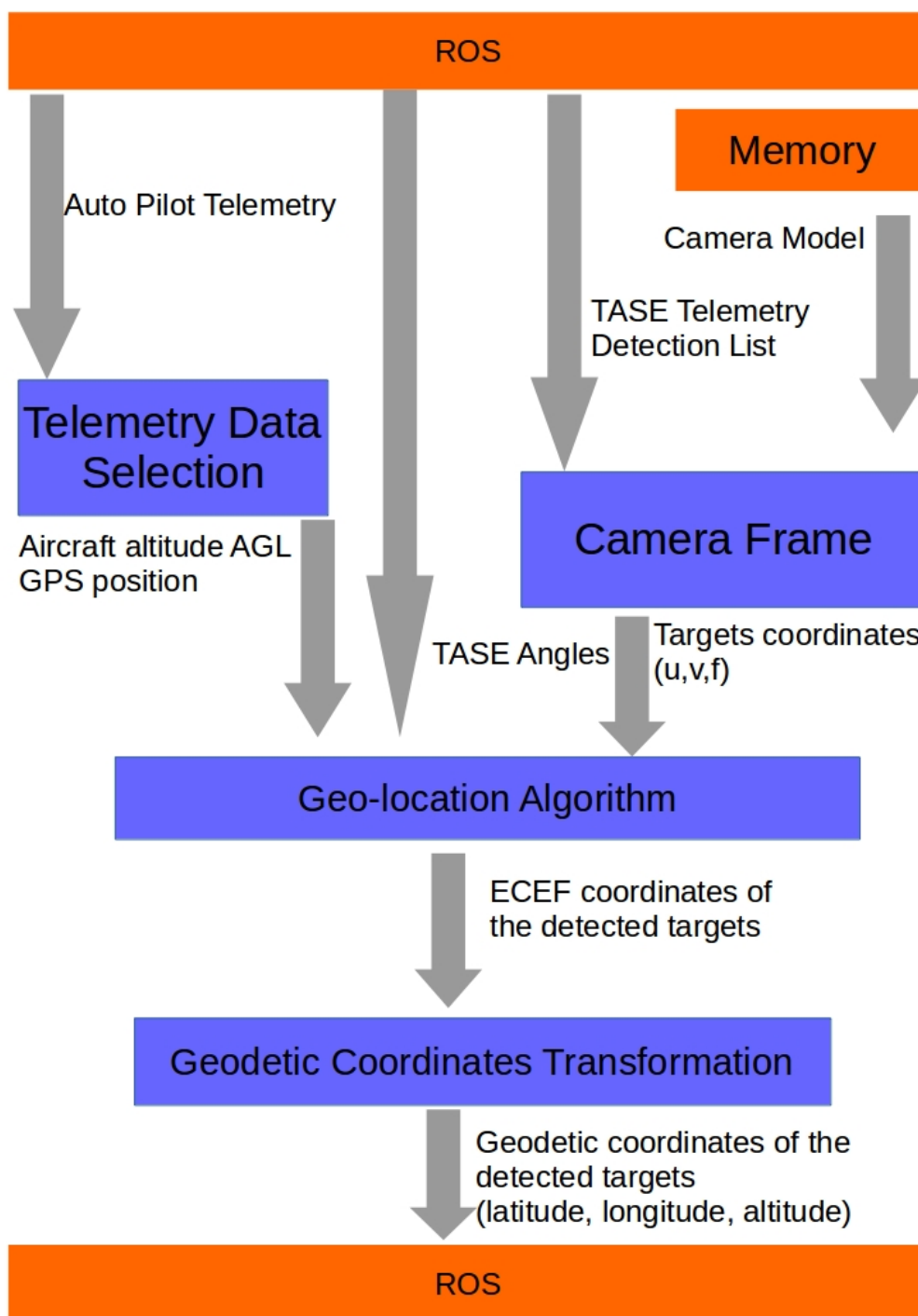
Figure 4.22: Geo-location module.

4.4 Geo-location in Real-World Conditions

4.4.1 Experiment Procedure

This experiment tests the geo-location algorithm in a full-scale real-world scenario. The objectives of this experiment are: final validation of the proposed methodology for calibrating a camera with variable settings and test the geo-location algorithm under real-world conditions and constraints. To accomplish the proposed objectives the $14th$ degree polynomial model for the focal length and the $16th$ degree polynomial model for the radial distortion parameter k_1 were used.

In this experiment, a test target with 2.8 meters in length by 2 meters in width was built (presented in Figure 4.23) and placed in several locations throughout the Portuguese Air Force Academy campus. The gimbal camera setup was placed on the top of the water tower building as presented by Figure 4.24. This location was chosen to ensure a sufficient height to approximate, as close as possible, the aircraft altitude of operation.

Figure 4.23: Test target for the full scale experiment.

Figure 4.26 presents the gimbal camera location as well as the targets locations. In each location, several tests were conducted by varying the pan, tilt or zoom gimbal parameters. During the tests, the gimbal camera was maintained in a static position and the location of the targets was chosen to accommodate a variety of ranges. The targets are numbered from 1 to 10 according to their increasing distance to the camera location.

Each target GPS coordinates were obtained with a receiver to compare the real target coordinates with the estimations made by the geo-location algorithm. In this experimental setup, environmental factors such as wind and sun glare were present and expected to influence the results, in contrast with the laboratory simulation from the previous experiment presented in Section 4.2.

The camera in this experiment was maintained in static position with a fixed heading, pitch, and roll (heading = 150º, pitch=3.15º, roll=0.95º). The target was captured in several tests with variable pan, tilt or zoom summarized in Table 4.25 (a) and (b), which presents the values of the variable parameters accordingly to the test and locations illustrated in Figure 4.26.

Figure 4.24: Camera setup in the water tower of the Portuguese Air Force Academy.

(a) Target locations 1 to 5.

| Target Location | Test | Test Parameters | | |
		Pan (degrees)	Tilt (degrees)	Zoom (%)
1	1	124,996	-30,005	89,5
	2	124,985	-29,994	98,9873
2	1	125,518	-21,990	98,902036
	2	125,518	-22,002	100
	3	126,518	-22,002	95,742138
3	1	-175,496	-23,990	98,934
	2	-175,486	-23,990	88,06
	3	-175,485	-25,990	80,06097
	4	-174,018	-24,488	72,263
	5	-173,978	-23,067	80,7376
4	1	-133,975	-18,982	92,8579
	2	-136,977	-18,982	92,8579
	3	-136,977	-19,034	92,8579
	4	-134,026	-22,036	92,8579
	5	-135,000	-20,993	99,2324
	6	-135,000	-20,500	100
5	1	-118,000	-16,003	84,127493
	2	-114,030	-16,002	84,12749
	3	-115,900	-18,002	84,12749
	4	-116,992	-19,990	93,68937
	5	-116,992	-19,990	93,6893
	6	-114,042	-19,990	93,68937

(b) Target locations 6 to 10.

| Target Location | Test | Test Parameters | | |
		Pan (degrees)	Tilt (degrees)	Zoom (%)
6	1	176,036	-9,992	98,784778
	2	-179,983	-11,012	85,36403
	3	-179,983	-11,012	85,364034
	4	177,015	-9,964	85,364033
	5	175,033	-12,015	85,364033
	6	175,984	-9,987	98,742138
7	1	135,487	-5,002	100
	2	135,487	-5,002	100
	3	136,014	-4,503	100
	4	134,954	-4,503	100
	5	134,954	-5,500	100
8	1	151,020	-5,013	95,832
	2	151,794	-5,202	100
	3	151,450	-4,303	100
9	1	156,486	-4,303	100
	2	155,953	-3,999	100
	3	156,486	-4,303	100
	4	155,466	-3,501	100
10	1	-174,047	-5,001	100
	2	-174,547	-5,002	100
	3	-174,534	-5,012	100
	4	-174,002	-5,500	100

Figure 4.25: Test parameters according to target locations.

Figure 4.26: Camera setup in the water tower of the Portuguese Air Force Academy.

4.4.2 Experiment Results

The environmental factors in this experiment played an important role in the number of valid tests in each location. The wind present in the top of the water tower was the main factor which caused vibrations in the camera mount, causing the collected image samples to be unsuitable for the test purposes as illustrated by Figure 4.27. In this image sample, the wind-induced vibrations in the camera mount caused the impossibility to determine the pixel coordinates of the target.

Figure 4.27: Image noise sample in location 10.

The height estimation using the proposed geo-location method is a key factor, where errors are more prompt to occur. The method relies on the an accurate measurement of the camera height in regard with the target, which in turn affects the geo-location algorithm in the calculation of the scale factor, defined in equation 3.57, and therefore the determination of the target coordinates. As such in this experiment two different altitudes were considered: altitude above target and altitude above ground.

The altitude above target represents the GPS height measured with respect to each specific target and is defined as the GPS measured altitude difference between the camera and the target GPS (WGS84) altitudes for each location. The altitude above target for each location are presented in Table 4.9. The values in Table 4.9 were obtained using the average of the GPS (WGS84) measurements of both the target and the camera locations and represents the altitude above target for each test location. The altitude above ground of 83 meters is the GPS (WGS84) measured height of the water tower and represents the camera GPS height measured with respect to the underlying ground surface (altitude above ground).

Location	1	2	3	4	5	6	7	8	9	10
Height Above Target (m)	83	80	82	83	81	80	80	90	85	105

Table 4.9: Altitude above target (H_1) for each location.

The results of the geo-location algorithm were studied under three different scenarios to access the influence of the altitude estimation in the algorithm performance. Figure 4.28 illustrates the several scenarios proposed and represents a schematic view of the different altitudes considered, were the height H_1 is the more clear representation of the altitude above target (obtained from the GPS average positions of the camera and target) and, H_4 the representation of the altitude above ground.

In the first scenario, it is assumed that the GPS coordinates of the camera are fixed. In this case, the height used to calculate the scale factor of the geo-location algorithm is the fixed altitude above the target for each location according to Table 4.9 (Scenario 1 in Figure 4.28). This means that the scale factor is obtained by using a steady altitude (H_1) above target for each location.

In the second scenario, the coordinates of the camera are obtained directly from its GPS receiver and the target coordinates are fixed. The height measurements to determine the scale factor is the altitude difference between the camera and the target in each time instance (Scenario 2 in Figure 4.28). In this scenario the altitude above target is given by by $H_2 + H_3$. The height H_2 is the online real data coming from the camera's GPS receiver and therefore, varies along the time. The height H_3 is maintained fixed.

In the third scenario, the height measurement to determine the scale factor is given by the altitude above ground of the camera regardless of the target location (Scenario 3 in Figure 4.29). This is illustrated in Figure 4.28 by $H_4 = 83m$.
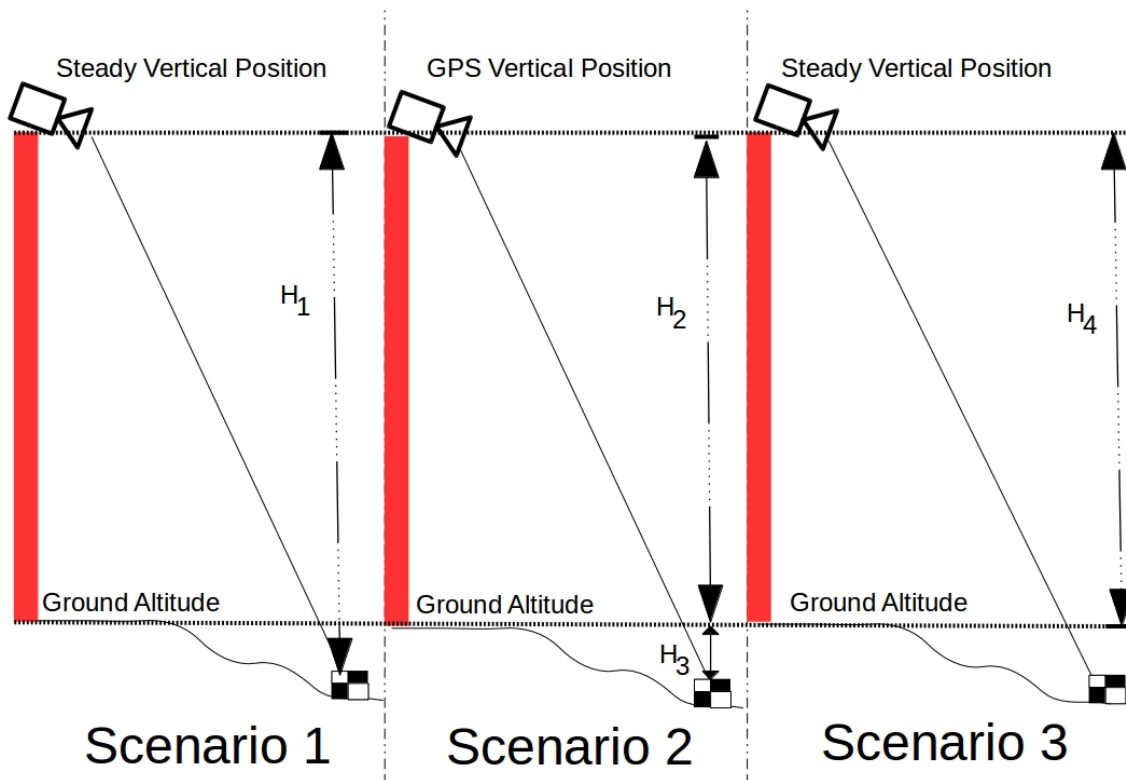
Figure 4.28: Test scenarios proposed.

The first scenario intends to benchmark the best possible results concerning the developed methodology. In this case, the variable parameters of the experiment are limited to the camera models obtain and the angle sensor readings. The second scenario intends to verify the influence of the camera GPS coordinates variation. The third and last scenario intends to verify the flat world assumption, where the camera altitude above target and altitude above ground are different, and how this difference influences

the results.

The results of this analysis are presented as a box plot in Figure 4.29. This Figure graphically depicts the error of the geo-location algorithm estimates of a target position. The box plot have lines extending vertically from the boxes (whiskers) indicating variability outside the upper and lower quartiles of the estimates obtained. The spacing between the different parts of the box indicate the degree of dispersion (spread) and skewness in the samples, and show outliers.

The analysis in this Figure summarizes all the tests for each location, where locations both in proximity to the camera and similar altitudes above ground and target are not affected by the different approaches. However, the targets further-away and with considerable height differences between the altitude above ground and altitude above target to the camera are more prompt to location errors. In Appendix B the histograms of the error for each scenario of analysis, location, and test are detailed.
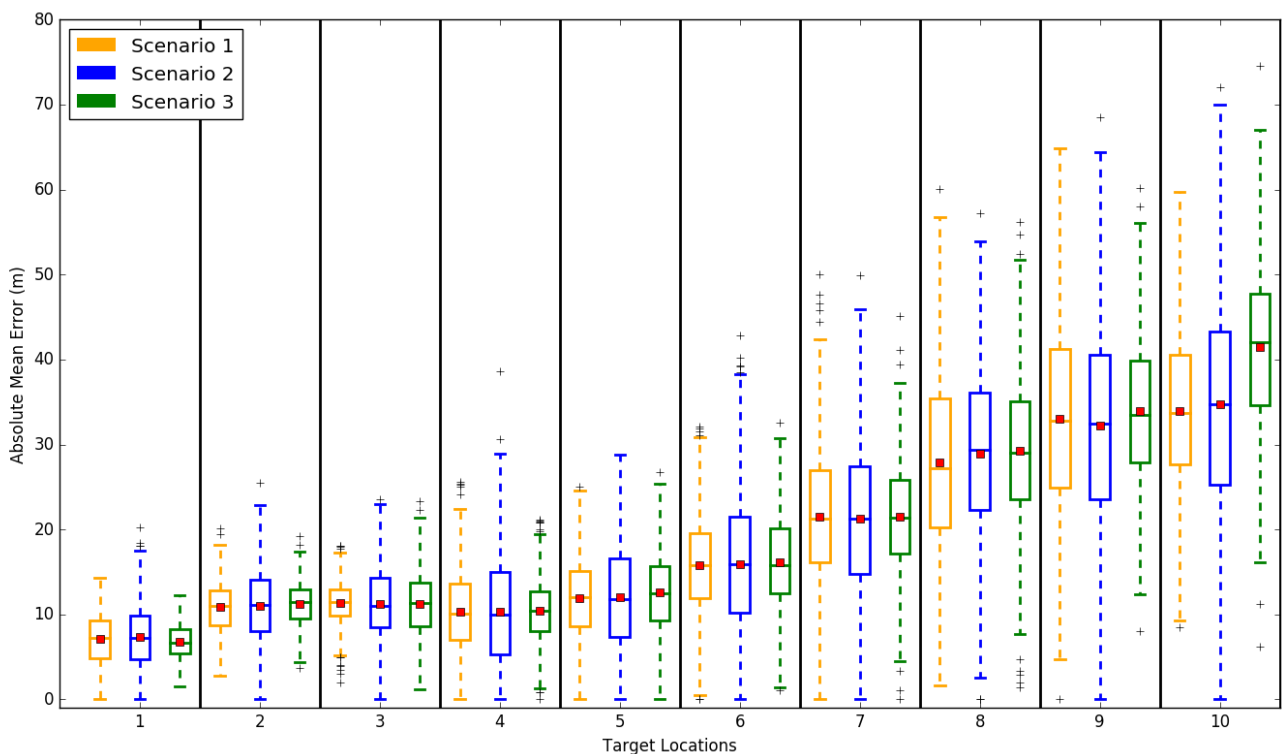
Figure 4.29: Error of each location according the proposed scenario of analysis.

Comparing the first and second scenarios both the median and mean (red square) suggest that using the online instant data from the camera GPS sensor does not significantly affect the location errors. This fact is due to the stable fix acquired by the GPS sensor during the experimental procedure. Because the camera initial position does not change, the location errors consequently are not affected in a significant way.

The locations 1 to 5, which are closer to the camera, present a moderate error location error varying from 7 to 15 meters. Although the outliers of the graph in this cases present errors has high as 15 meters (target location 1) and 23 meters (target location 5) this means that in some cases the the online instant error of the estimated position was bigger then the average error.

In locations 6 to 10, which are further away from the camera location, the positioning error is higher than in the previous locations (1 to 5). In the data collected, the mean error varies from 18 to 35 meters as the target distance to the camera increases. The samples dispersion of the results in this locations also increases, where the location error obtained from an instant online sample in some locations shown by the outliers of the box plot can be as high as 32 to 60 meters. Regardless of this fact, the results from using a constant value for the camera coordinates and using the GPS instant readings are similar. However, the difference between the upper and lower quartile of the box plot is higher in the second scenario. The data points to the negligible effect of the camera GPS position variation in the average position obtained but, with higher dispersion of samples.

Comparing Figures B.1 and B.2 of the Appendix B (respectively referencing the first and second scenarios,) it is possible to see that, across the several tests conducted in each location, the mean absolute error obtained presents a small difference between the first and second scenarios.

If the calibration models obtained were perfect estimations of the intrinsic parameters throughout the zoom band, then the errors would be smaller. However, taking into account that in a small-scale experiment errors of at most 0.17 meters were obtained and the fact this experiment scales the scenario proposed in Section 4.2 the errors obtained are also scaled and therefore understandably present.

Comparing the first and third scenarios in Figure 4.29 it is possible to observe that, in the locations where the altitude above target and the altitude above ground of the camera (83 meters) differ the most, the error increases. In locations 1 to 7 and 9, the height difference between the altitude above ground and altitude above target is smaller, therefore the positioning error is similar to the ones obtained in the first scenario. However, in locations 8 and 10, the height difference between the altitudes is higher and therefore the error increases.

In the first scenario, the induced errors by the camera calibration models and environmental effects are predominant. In the third scenario, there is an additional altitude error which increases the overall positioning error. This suggests that the geo-location algorithm is very sensitive to an accurate measurement of the altitude above target. This is illustrated by Figures B.1 and B.3 of Appendix B (respectively referencing the firs and third scenarios), were considering locations 8 and 10, the location error respectively increases from around 24 to 27 meters and from approximately 25 meters to 34 meters.

Taking into account the formulation in AppendixC and assuming $h = 100$, $\tau = 5^{\mathrm{o}}$ and $\eta = 7^{\mathrm{o}}$, Figure 4.30 denotes the ground area captured by an image as a function of the target distance to the camera. In this figure it is possible to see that the higher the distance of a target to the camera, the higher the ground area mapped by an image assuming a fixed altitude, vertical and horizontal field of view.

Assuming that the camera has a fixed set number of pixels, the the further a target is located the higher the area mapped by a single pixel. Therefore, if a target pixel coordinates vary due to environmental factors or sensor readings (at a small tilt angle) the target coordinates are more prompt to be erroneous.

Figure 4.30 also illustrates the geo-location errors of the experiment according to the target distance. This illustrates as the target distance increases, the experimental errors obtained increase as the area covered by an image also increases. From this analysis a conclusion can be drawn, the further away a

target is located, the more influence the determination of its pixel coordinates determination has in the geo-location method.
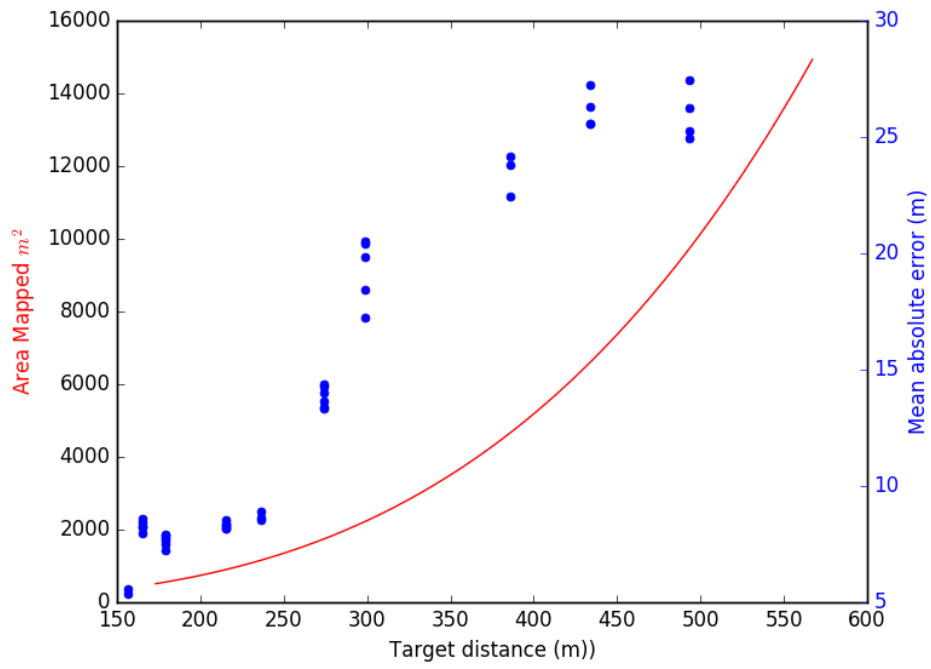

Figure 4.30: Geo-location error as a function of the tilt angle.

4.4.3 Discussion of Results

The experiment procedure conducted and consequent data analysis points to several factors which influence the geo-location method proposed. The first factor is the camera intrinsic parameter models obtained. The models used are approximations of the intrinsic parameters behavior as function of the zoom setting, they're bound to their relative accurateness and therefore the errors made during the model calculations are propagated to this experiment. The fact that, in the small-scale scenario of this experiment, the disregard of some distortion parameters did not present a great influence in the results, at a large scale they undoubtedly influence the results.

The second factor which affects the geo-location algorithm is the height difference between the altitude above target and the altitude above ground. When these altitudes are similar there are no significant errors. However, when the height difference increases, the positioning errors increase, meaning that the proposed geo-location solution is not suitable to be employed in rugged terrain were significant geological formation such as mountains and valleys would result in a significant loss of accuracy.

The third factor which influences the geo-location algorithm performance is the target distance in relation to the camera, where for higher distances the error is consequently higher. The increasing target distance results in a increasing image obliqueness, where images captured at a more oblique position cover a bigger area. Therefore, the further a target is located the more influence the pixel location of the target has in the final estimate of a target coordinates.

The results obtained are encouraging. However, they point some factors which much be taken into consideration for future tests. The fact that the scenario of operation of the geo-location method developed is a maritime environment, where the height difference between the target altitude and the ground (mean sea level) altitude is negligible, mitigates the effect that the height difference influences the proposed method.

The target distance at which the target is points to a careful selection of the flight path of the aircraft. In this, the lower the more oblique an image is the more prompt the location errors are, meaning that an altitude/distance compromise must be verified which minimizes the error.

Chapter 5

Conclusions

In conclusion of this thesis, this chapter describes the main achievements, suggestions of topics for further research and the conclusions drawn from the research conducted.

5.1   Achievements

The camera intrinsic and distortion parameters throughout its zoom band were obtained using a state-of-the-art camera calibration technique, where the assumption that the camera characteristic parameters are interdependent with lenses configuration was validated. The behavior of the camera parameters was approximated in several polynomial approaches and validated in a controlled environment.

Using the camera models which shown more promising results in laboratory environment, a full-scale experience of the geo-location algorithm was conducted. In this experiment a target was placed in several locations across the Portuguese Air Force Academy campus. The analyses of the results of this experiment were done considering a number of scenarios which influence the estimates of a target position.

A geo-location algorithm was designed according to the scenario of employment of the project. The method devised to geo-locate targets is a stand-alone, instant and online solution to determine an image target geodetic coordinates. This geo-location method only depends on the knowledge of the camera parameters model, the camera and aircraft on-board sensors.

The geo-location algorithm was design so that its integration in the Seagull software architecture could be done without any major modifications or upgrades to its structure.

5.2   Future Work

Future work that would improve the research presented in this thesis and improve the capabilities of the Seagull project include:

1. Design of a filtering solution to minimize the results dispersion of the geo-location algorithm;

2. Study of the flight pattern which minimizes the geo-location error;

3. Accurate evaluation of the target geo-location using a stereo camera set-up;

4. Gimbal control loop for automatic target tracking;

5. Development of a cooperative geo-location system allowing multiple platforms to simultaneous locate a target.

5.3   Conclusions

In summary, this thesis has presented a methodology for geo-locating an unknown, uncooperative target object in the maritime environment. In the context of the Seagull project, this work represents an extension of the capabilities already in existence.

This thesis presents a camera calibration methodology, based on state-of-the-art techniques, for cameras with variable zoom. Experiments have proven the validity of the method proposed however, a major drawback is that this method is a time-consuming process. It was not evaluated if the effectiveness of the proposed method is valid for cameras with more complex lenses assembly or camera devices than the one used in this work. The focus setting was briefly studied and the data proved the possibility of neglecting it, without losing rigor in the study. However, the effects of the focus setting can be incorporated but, with different interdependence models.

The work presented in this document shown experimental results which allowed to demonstrate an interdependence between the camera zoom and its intrinsic and distortion parameters, where it was possible to define several polynomial models of the parameters through the zoom band. The experiments results lead to the conclusion that a $14th$ degree polynomial approach for the focal distances and a $16th$ degree polynomial approach of the distortion parameter held the best results.

The proposed geo-location method was devised according to the environment of operation of the aircraft. Tests in a controlled environment reveal the sustainability and validity of the algorithm implemented. However, the location errors increased with the distance of the target location in relation to the camera position. An experiment in real world conditions also verified this fact, were the errors were consequently higher given the target distances considered.

In a real world scenario, several sources of error were identified: wind, the inaccurate measurement of the altitude above target and the target horizontal distance to the camera. The wind produced oscillations in the camera mount leading to image noise (in some cases unusable images). The difference between the altitude above target and the altitude above ground have shown to play an important

role and reveal that the solution proposed is not suitable to be employed in irregular terrain. The target ground distance to the camera is one other factor which influences the results. The higher the distance a target is located, the higher the image obliquity and the more prompt the target estimates are to location errors.

In conclusion, the work presented in this dissertation details a possible methodology to geo-locate targets with a calibrated monocular camera, which shows promising results. The camera calibration method employed held good results as well as the geo-location algorithm, under a controlled environment. Under real-world conditions, the method proposed in this thesis held satisfactory results as a stand-alone instant online solution. However, certain sources of error where verified and highlighted because of their effect in affecting the results. The results obtained in this work are good indicators that the proposed approach is suitable to be more carefully study, and developed to increase the Seagull project capabilities. This in turn can help the Air Force Academy in developing a sustainable solution to Air Force obligations of: control and surveillance in maritime environment.

Bibliography

[1] Estratégia nacional para o mar 2013-2020. Technical report, Governo de Portugal, 2013.

[2] Conceito estratégico de defesa nacional. Technical report, Governo de Portugal, 2013.

[3] A. Bernardino, N. Ferreira, C. Dias, F. Nunes, G. Cruz, and J. Viegas. SEAGULL - Relatório Estado da Arte. Technical report, Critical Software S.A., 2013.

[4] G. Cruz and A. Bernardino. Aerial detection in maritime scenarios using convolutional neural networks. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, 2016. Blanc-Talon J. and Distante C. and Philips W. and Popescu D. and Scheunders P. (eds) Advanced Concepts for Intelligent Vision Systems ACIVS 2016. Lecture Notes in Computer Science, vol 10016, Springer, Cham.

[5] R. Tsai. versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lens. In *IEEE Journal of Robotics and Automation*, 1987.

[6] J. Heikkilä. Geometric camera calibration using circular control points. In *EEE Trans. Pattern Anal. Machine Intell*, page 1066–1077.

[7] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999.

[8] J. Salvi, X. Armangue, and J. Batlle. A comparative review of camera calibrating methods with accuracy evaluation. In *Pattern Recognition*, pages 1617–1635, 2002.

[9] W.Sun and J. R. Cooperstock. An empirical evaluation of factors influencing camera calibration accuracy using three publicly available techniques. In *Machine Vision and Applications*, page 51–67, 2006.

[10] R. Madison, P. DeBitetto, A. Olean, and M. Peebles. Target geolocation from a small unmanned aircraft system. In *IEEE Aerospace Conference*, Big Sky, Montana, March 2008.

[11] D. B. Barber, J. D. Redding, T. W. McLain, R. W. Beard, and C. N. Taylor. Vision-based target geo-location using a fixed-wing miniature air vehicle. In *Journal of Intelligent and Robotic Systems*, volume 47, pages 361–382, December 2006.

[12] J. A. Ross, B. R. Geiger, G. L.Sinsley, J. F. Horn, L. N. Long, and A. F. Niessner. Vision-based target geolocation and optimal surveillance on an unmanned aerial vehicle. In T. P. S. University, editor, *AIAA Guidance, Navigation, and Control Conference*, Haway, August 2008.

[13] R. Hartley and A. Zisserman. *Multiple View Geometery in computer vision*. Cambridge University Press, 2003.

[14] D. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. NJ: Prentice Hall, 2003.

[15] H. Schultz, A. Hanson, E. Riseman, F. Stolle, and Z. Zhu. A system for real-time generation of geo-referenced terrain models. In *SPIE Enabling Technologies for Law Enforcement*, Boston, 2000.

[16] R. Kumar, S. Samarasekera, S. Hsu, and K. Hanna. Registration of highly-oblique and zoomed in aerial video to reference imagery. In *Proceedings of the IEEE Computer Society Computer Vision and Pattern Recognition Conference*, 2000.

[17] A. Bernardino, R. Batista, G. Cruz, T. Oliveira, and R. Ribeiro. SEAGULL - Algoritmos de Payload e Documentação Científica. Technical report, Critical Software S.A, 2013.

[18] D. Salgueiro, G. Cruz, J. Viegas, P. Silva, R. Batista, T. Oliveira, and R. Ribeiro. SEAGULL - Especificação do Software Embarcado. Technical report, Critical Software S.A, 2014.

[19] A. Bernardino, G. Cruz, J. Viegas, N. Ferreira, P. Silva, R. Batista, and S. Ferreira. SEAGULL - Especificação da Arquitectura do Software. Technical report, Critical Software S.A, 2013.

[20] About ros, 2017. URL `http://www.ros.org/about-ros/`.

[21] R. W. Beard and T. W. McLain. *Small Unmanned Aircraft Theory and Practice*. Princeton University Press, 1nd edition, 2011. ISBN:978-0-691-14921-9.

[22] Open Cv. Camera Calibration and 3D Reconstruction, 2017. URL `http://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html`.

[23] M. H. Juayang Weng, Paul Cohen. Camera calibration with distortion models and accuracy evaluation. *IEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 1992.

[24] M. Heikkinen. Cgeschlossene formeln zur berechnung raumlicher geodatischer koordinaten aus rechtwinkligen koordinaten. *Zeitschrift fur Vermessungswesen*, 1982.

[25] P. Sturm. Self-calibration of a moving zoom-lens camera by pre-calibration. *Image and Vision Computing*, 1997.

[26] Open Cv - tutorial. Camera Calibration and 3D Reconstruction, 2017. URL `https://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html`.

[27] Matlab. Camera Calibration Toolbox for Matlab, 2017. URL `http://www.vision.caltech.edu/bouguetj/calib_doc/`.

Appendix A

Camera Calibration Results

Zoom	Auto-Focus		Fixed Focus		Auto-Focus		Fixed Focus		Auto-Focus		Fixed Focus	
	RMS	std	RMS	std	fx	std	fx	std	k1	std	k1	std
0,00	0,96	0,24	0,96	0,24	641,59	5,66	657,49	298,08	-0,32	0,05	0,71	1,85
2,41	0,71	0,13	0,72	0,14	676,33	4,63	782,63	317,70	-0,29	0,04	1,33	2,13
3,14	0,81	0,07	0,81	0,07	647,80	5,23	671,01	282,96	-0,37	0,02	1,62	1,99
3,44	0,70	0,05	0,72	0,22	678,88	4,45	722,67	313,06	0,08	2,42	1,65	3,07
5,44	0,83	0,07	0,85	0,19	674,10	5,70	779,49	271,84	-0,18	0,77	1,07	1,84
6,81	0,63	0,05	0,65	0,14	697,50	2,56	718,11	303,82	0,08	1,98	1,47	2,74
7,48	0,63	0,10	0,64	0,13	688,66	5,16	785,11	313,27	-0,28	0,07	1,10	1,68
8,55	0,91	0,11	0,92	0,13	694,67	4,28	750,27	288,44	-0,27	0,08	0,91	1,61
12,95	0,62	0,10	0,62	0,10	729,44	3,09	874,82	307,18	-0,27	0,03	1,45	2,10
13,93	0,78	0,07	0,80	0,16	740,13	6,12	784,80	306,48	-0,12	0,91	0,91	2,02
14,48	0,68	0,07	0,70	0,15	733,16	3,20	819,99	266,06	-0,18	0,09	0,98	2,09
17,00	0,75	0,07	1,10	0,16	756,24	5,06	882,24	316,87	-0,22	0,04	2,06	2,60
18,02	0,64	0,09	1,01	0,17	761,04	3,61	825,54	327,67	-0,18	0,15	2,61	2,61
19,91	0,73	0,05	1,10	0,12	769,29	2,36	851,39	295,74	-0,13	0,04	2,53	2,47
22,42	0,64	0,05	1,01	0,10	799,32	2,92	902,80	58,55	-0,14	0,04	2,87	2,60
22,93	0,77	0,06	1,13	0,17	793,13	2,24	883,54	55,02	1,10	8,82	3,96	8,87
23,43	0,72	0,06	1,08	0,09	806,18	2,95	906,60	59,67	-0,20	0,03	2,55	2,59
23,93	0,66	0,06	1,02	0,13	811,50	1,96	912,48	61,23	-0,17	0,04	2,38	2,71
26,68	0,63	0,08	0,98	0,12	829,63	2,22	928,46	58,56	-0,12	0,06	2,76	2,83
28,38	0,81	0,06	1,16	0,11	848,15	2,85	948,12	47,09	-0,16	0,08	2,28	2,88
29,57	0,81	0,06	1,18	0,14	867,19	2,05	966,62	57,12	-0,01	0,06	2,72	2,40
33,91	0,66	0,05	0,99	0,11	910,36	4,61	996,63	55,20	-0,13	0,06	2,65	2,71
35,19	0,88	0,08	1,25	0,14	943,21	4,25	1043,48	63,23	0,34	2,87	3,48	3,40
38,89	0,91	0,06	1,26	0,10	989,51	3,23	1088,42	58,13	-0,20	0,06	2,82	2,71
42,22	0,72	0,06	1,08	0,15	1040,49	3,23	1136,62	56,09	0,00	0,07	3,51	2,42
42,93	0,72	0,07	1,07	0,11	1046,09	5,63	1158,44	53,87	-0,16	0,09	2,38	2,39
44,96	1,07	0,07	1,42	0,10	1096,60	5,48	1199,84	51,18	-0,14	0,28	2,15	2,68
47,89	0,74	0,04	1,08	0,11	1170,92	6,17	1267,67	55,84	0,10	0,12	3,52	2,45

Table A.1: Camera calibration experiment results.

Zoom	Auto-Focus		Fixed Focus		Auto-Focus		Fixed Focus		Auto-Focus		Fixed Focus	
	RMS	std	RMS	std	fx	std	fx	std	k1	std	k1	std
51,35	0,87	0,05	1,67	0,21	1227,62	8,59	1816,11	177,42	-0,18	0,12	4,84	3,41
55,19	0,57	0,04	1,46	0,21	1276,39	4,56	1874,81	176,18	0,38	0,60	2,94	3,29
56,49	0,86	0,03	1,65	0,20	1368,00	9,95	1932,40	184,86	0,40	0,44	4,63	3,83
58,52	0,79	0,14	1,65	0,24	1408,63	6,09	2012,75	173,33	0,36	0,38	3,97	3,92
59,48	0,76	0,04	1,62	0,23	1496,96	10,76	2145,06	159,78	0,04	0,26	3,59	4,15
62,96	0,95	0,06	1,78	0,24	1603,92	11,54	2195,94	167,78	0,35	0,22	3,92	3,78
63,86	0,82	0,08	1,69	0,24	1609,58	10,52	2196,97	195,87	0,45	0,14	4,25	3,95
64,81	0,85	0,04	1,72	0,23	1672,83	8,89	2305,37	169,76	0,41	0,21	4,94	3,87
66,79	0,92	0,03	1,75	0,24	1753,88	12,83	2371,41	166,40	0,23	0,23	3,98	3,76
69,00	0,97	0,05	1,76	0,23	1819,94	10,18	2461,24	161,19	0,74	0,39	3,60	3,83
70,95	0,69	0,04	1,57	0,23	1968,26	12,49	2885,91	200,67	0,55	0,52	5,35	4,24
72,70	0,95	0,06	1,85	0,23	2095,07	11,66	2962,36	209,03	0,45	0,39	4,27	3,90
74,28	0,88	0,04	1,75	0,23	2277,68	20,86	3228,16	217,25	1,49	0,38	6,84	4,36
75,93	0,76	0,06	1,61	0,20	2370,59	20,29	3271,41	211,49	0,35	0,36	4,95	3,81
77,73	0,79	0,06	1,61	0,20	2525,98	18,47	3404,98	206,86	0,71	0,28	4,97	4,05
79,31	0,72	0,04	1,56	0,21	2742,69	42,84	3650,03	228,76	1,25	0,42	6,45	3,55
80,97	0,78	0,03	1,60	0,26	2903,41	36,16	3838,18	206,87	0,91	0,31	5,24	3,51
83,66	0,85	0,05	1,68	0,25	3619,63	74,04	4529,44	227,36	1,52	0,43	6,21	3,67
85,03	0,72	0,05	1,59	0,24	3703,18	68,23	4585,97	220,23	3,09	0,84	6,62	3,93
86,78	0,66	0,03	1,53	0,22	3873,30	95,98	4773,75	212,14	3,54	1,33	7,88	4,05
88,23	0,65	0,03	1,47	0,25	4236,55	75,96	5121,67	232,77	3,70	0,78	7,68	4,45
89,05	0,87	0,05	1,72	0,21	4171,98	72,21	5072,86	194,08	2,20	0,59	6,74	3,74
90,15	0,73	0,07	1,61	0,21	4446,28	111,26	5369,65	221,96	3,65	1,03	7,35	3,99
91,09	0,97	0,05	1,81	0,24	5010,19	47,92	5901,55	223,48	4,63	1,07	8,31	4,17
92,34	0,87	0,05	1,75	0,25	5420,80	161,09	6310,01	204,38	3,03	1,18	8,17	3,65
93,37	0,86	0,05	1,66	0,22	5647,84	110,81	6565,24	239,91	4,15	1,38	8,93	3,87
94,23	0,80	0,06	1,59	0,20	6238,81	215,59	7162,70	327,60	5,26	1,53	8,64	3,56
96,07	0,86	0,05	1,71	0,21	7479,63	199,67	8384,72	290,97	7,06	1,30	11,38	4,25
96,76	1,04	0,07	1,87	0,22	8306,09	194,80	9217,11	306,22	12,40	1,66	16,39	4,18
97,34	0,74	0,05	1,59	0,21	8453,49	222,27	9336,53	324,90	13,88	2,92	18,33	4,73
97,88	0,80	0,05	1,63	0,26	10562,62	687,79	11410,57	721,43	13,80	2,44	19,02	4,25
98,45	0,89	0,06	1,71	0,20	10306,27	291,06	11155,54	342,70	21,22	3,41	24,68	4,56
98,62	0,81	0,05	1,65	0,19	11227,40	541,65	12110,01	576,29	21,62	3,13	26,30	5,12
100,00	1,03	0,07	1,90	0,20	15146,08	1060,28	16070,25	1029,85	61,02	10,99	65,59	12,10

Table A.2: Camera calibration experiment results (continuation).

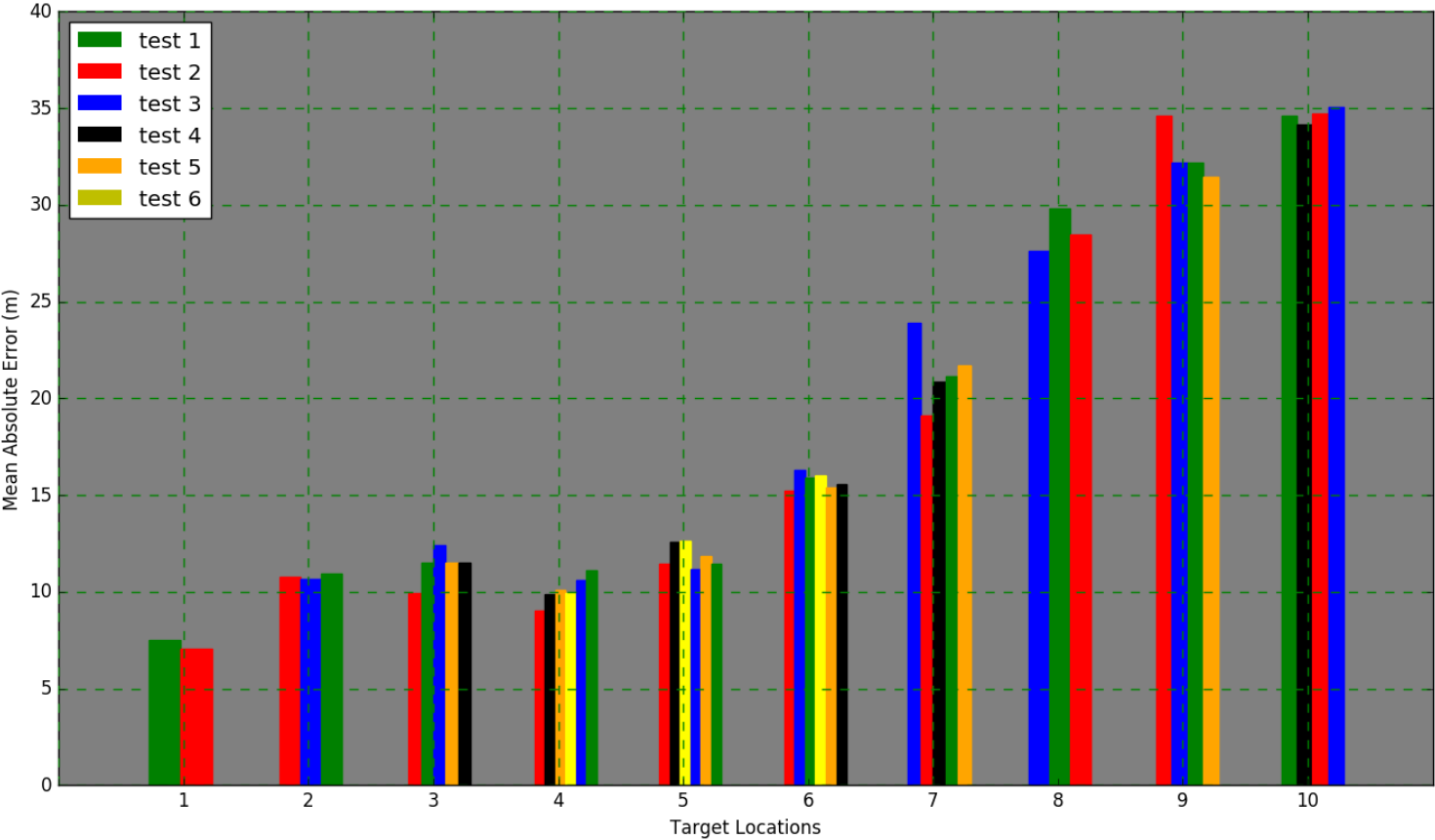# Appendix B

Geo-location Errors

Figure B.1: Camera and target GPS coordinates fixed. Estimation using altitude above target for each location.
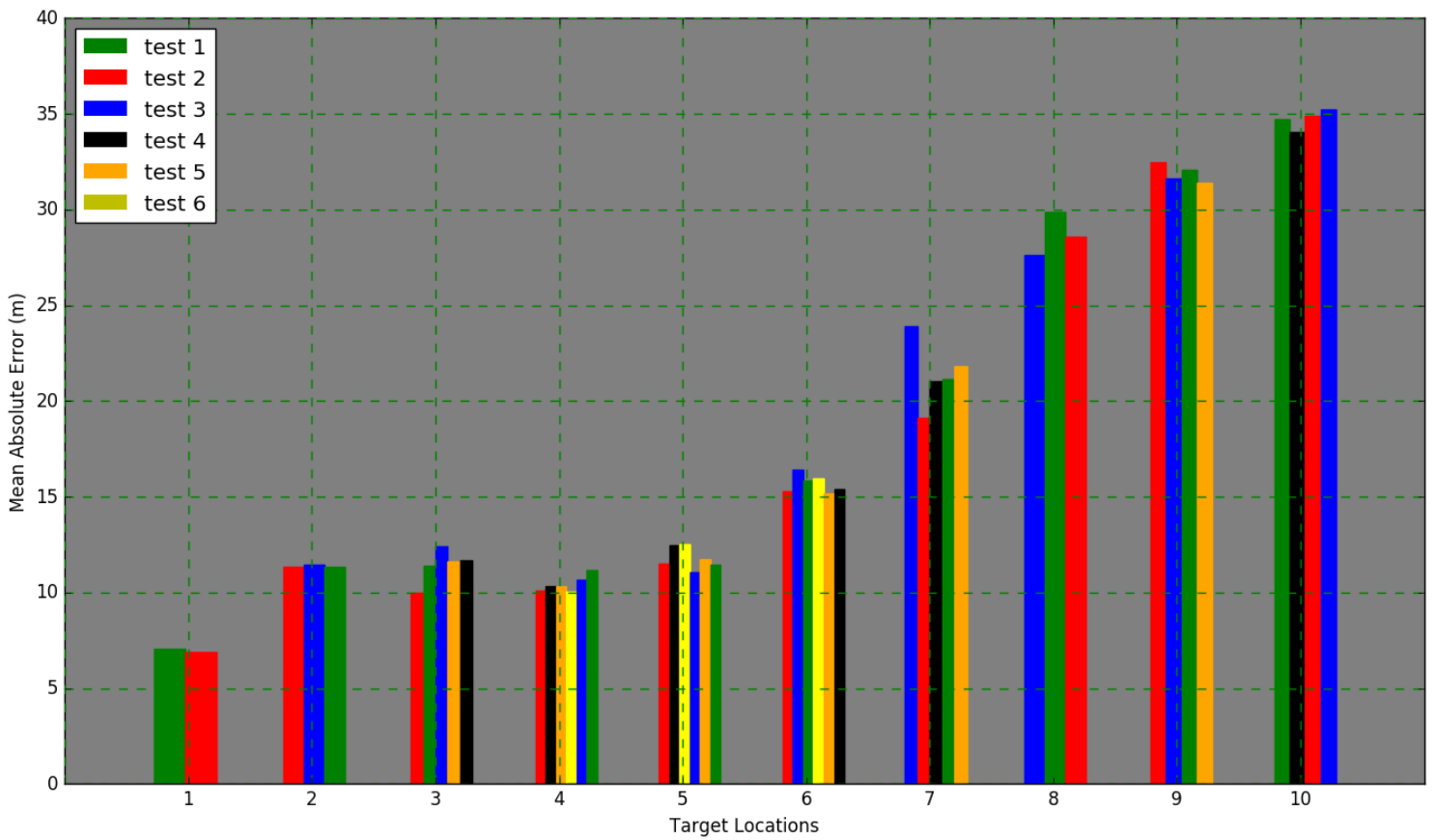
Figure B.2: Camera GPS coordinates variable and target GPS coordinates fixed. Estimation using altitude above target for each location in each time sample.
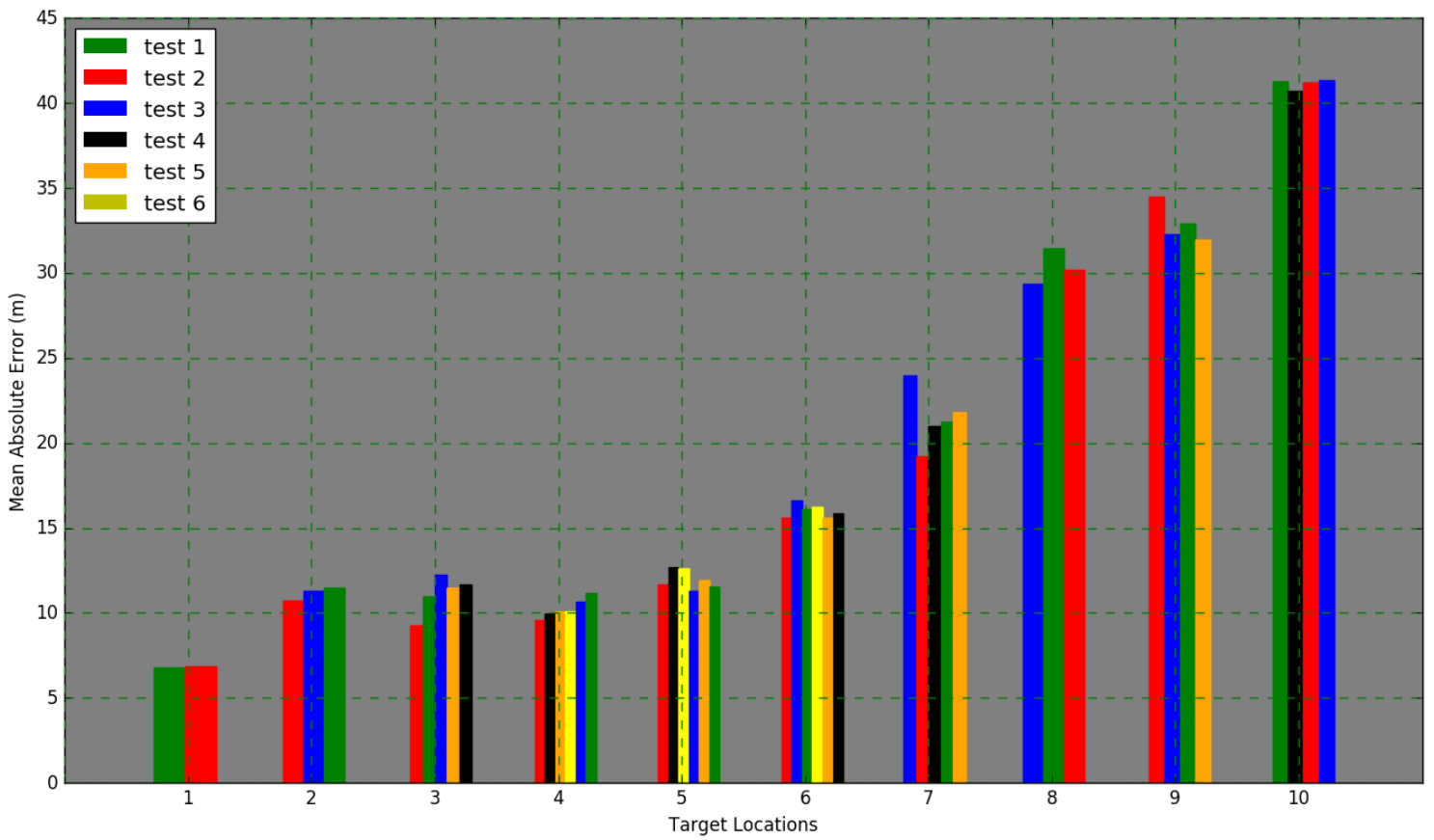
Figure B.3: Camera and target GPS coordinates fixed. Estimation using a fixed altitude above ground.

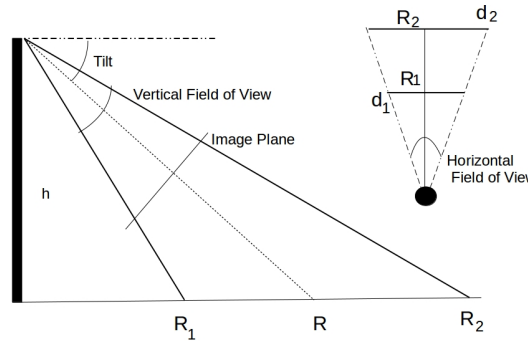# Appendix C

Induced Geometry Errors

Figure C.1: Assuming a flat earth model, the ground distance of a camera image can be computed by knowledge of the tilt angle λ the lens vertical and horizontal field of view respectively τ and η, and the altitude h.

The geometric scene of the experiment can be illustrated by Figure C.1. This, summarizes the top and side perspectives and the measurement relations between the mapped area and the camera image. The distances in Figure C.1 are derived as:

$$R_1 = \frac{h}{\lambda + \frac{\tau}{2}}, \tag{C.1}$$

$$R_2 = \frac{h}{\lambda - \frac{\tau}{2}}, \tag{C.2}$$

$$d_1 = R_1 tan(\frac{\eta}{2}), \tag{C.3}$$

$$d_2 = R_2 tan(\frac{\eta}{2}), \tag{C.4}$$

were the mapped area is given by

$$A = (d_1 + d_2)(R_2 - R_1). \tag{C.5}$$