TÉCNICO LISBOA

Direct Georeferencing of Fire Front Aerial Images using Iterative Ray-Tracing and a Bearings-Range Extended Kalman Filter

Bernardo Maria Gonçalves Santana

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Professor Alexandre José Malheiro Bernardino
Professor Ricardo Adriano Ribeiro

Examination Committee

Chairperson: Professor João Fernando Cardoso Silva Sequeira
Supervisor: Professor Ricardo Adriano Ribeiro
Member of the Committee: Professor João Pedro Castilho Pereira Santos Gomes

January 2021

For my father, who left us too soon

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

I would like to express my gratitude to my mother, Carla Santana, without whom this endeavour would have been lonesome and harder. To my girlfriend Adriana Vazão and her parents, who also supported and encouraged me along the development and writing of this thesis. To my friends, who freshened my mind and gave me strength.

To Professor Alexandre Bernardino, for his guidance, patience and trust I feel he put in me. To him I express my deepest gratitude for offering me the opportunity to participate in such a noble project.

To Professor Ricardo Ribeiro, for his teaching and valuable insights, crucial for the methodology and experimental results analysis.

Finally, to Instituto Superior Técnico, for providing me the knowledge and the skills that will be useful throughout my personal and professional life.

Resumo

Este trabalho propõe o algoritmo IRT (*Iterative Ray-Tracing*) como metodologia de georreferenciação de frentes de incêndio a partir de imagens captadas por um veículo aéreo. O IRT requer que o veículo esteja equipado com um dispositivo GPS (*Global Positioning System*), IMU (*Inertial Measurement Unit*) e um DEM (*Digital Elevation Map*). Além disto, são necessários os IP (*Intrinsic Parameters*) da câmara a utilizar assim como os pixeis a georreferenciar. Tendo em conta que o equipamento está sujeito a erros, propõe-se ainda a utilização da UT (*Unscented Transform*), permitindo assim a caracterização da incerteza associada ao IRT. Por último, desenvolveu-se um novo modelo de filtragem denominado *Bearings-Range*, que permite melhorar a estimativa da posição e incerteza a partir de várias observações do mesmo alvo.

Os algoritmos de georreferenciação e filtragem foram validados por duas simulações em terreno acidentado e plano. Comparou-se a performance do EKF (*Extended Kalman Filter*) com o CKF (*Cubature Kalman Filter*) utilizando os modelos *Bearings-Range* e *Bearings-Only*. Foram alcançados resultados idênticos, com o primeiro a destacar-se pelo menor tempo de processamento, pelo que é o filtro mais adequado para processamento em tempo real.

Tendo o sistema validado em simulação, realizaram-se três experiências com dados reais. Utilizou-se um telemóvel para adquirir imagens de um alvo e telemetria ao longo do percurso pedestre elevado. Além disto, dois vídeos gravados por UAVs (*Unmanned Aerial Vehicles*) foram utilizados para testar o algoritmo. Os resultados obtidos demonstram a aplicabilidade da metodologia proposta para georreferenciar frentes de incêndio.

Palavras-chave

incêndio florestal, veículo aéreo, georreferenciação, GPS, IMU, DEM

Abstract

This work proposes the Iterative Ray-Tracing (IRT) as a forest fire georeferencing algorithm using images captured by an aerial vehicle. The IRT requires that the vehicle is equipped with a Global Positioning System (GPS) device, an Inertial Measurement Unit (IMU) and a Digital Elevation Map (DEM). In addition, the camera's Intrinsic Parameters (IP) and the pixels to georeference must be known. Considering errors in the equipment, the Unscented Transform (UT) is proposed to characterize the uncertainty of the IRT. Furthermore, a novel Bearings-Range filter measurement model is put forward, to improve the target position and reduce its uncertainty.

The georeferencing and filtering algorithms were validated with two simulations on rough and flat terrains. A performance comparison was done between the Extended (EKF) and Cubature Kalman Filters (CKF) using the Bearings-Range and Bearings-Only measurement models. Identical results were achieved, with the EKF performing faster, making it more adequate for real-time processing.

Having validated the system in simulations, three experiments were conducted using real data. A mobile phone was used to acquire imagery of a target and telemetry along an elevated pedestrian path. Furthermore, the algorithm was tested on two videos recorded with an Unmanned Aerial Vehicle (UAV). The results demonstrate the applicability of the proposed methodology to georeference forest fires.

Keywords

forest fire, aerial vehicle, georeferencing, GPS, IMU, DEM

Contents

List of Tables

List of Figures

Nomenclature

BA Bundle Adjustment

BO-CKF Bearings-Only Cubature Kalman Filter

BO-EKF Bearings-Only Extended Kalman Filter

BR-CKF Bearings-Range Cubature Kalman Filter

BR-EKF Bearings-Range Extended Kalman Filter

CKF Cubature Kalman Filter

CRS Coordinate Reference System

DEM Digital Elevation Map

DOF Degrees Of Freedom

ECEF Earth-Centered Earth-Fixed

EKF Extended Kalman Filter

EP Extrinsic Parameters

GCP Ground Control Points

GNSS Global Navigation Satellite System

GPS Global Positioning System

IMU Inertial Measurement Unit

IP Intrinsic Parameters

IPG Iterative Photogrammetry

IRT Iterative Ray-Tracing

MEMS MicroElectroMechanical Systems

OCR Optical Character Reader

QGIS Quantum Geographic Information System

RMSE Root Mean Squared Error

RT Ray-Tracing

SfM Structure from Motion

TIN Triangular Irregular Network

UAV Unmanned Aerial Vehicle

UKF Unscented Kalman Filter

UT Unscented Transform

WGS84 World Geodetic System 1984

1

Introduction

Contents

In recent years, remote sensing has seen an increased interest in the scientific community. The development and widespread use of Unmanned Aerial Vehicles (UAVs) as a cheaper solution when compared to manned aerial vehicles enabled the development of target geolocalization applications. This includes precision agriculture [26], natural disaster management [39] and fire detection and monitoring [24]. This work addresses the latter.

The most commonly used sensors regarding target geolocalization are digital cameras, Global Positioning Systems (GPSs) and Inertial Measurement Units (IMUs). This became known as Direct Georeferencing, since direct sensor orientation is computed by processing the information provided by the onboard sensors [23], i.e., the camera's Extrinsic Parameters (EP) are directly calculated.

Depending on the accuracy requirements, sensors with different specifications can be used. However, the higher the desired accuracy, the higher the cost and/or size of the hardware. Due to payload constraints, UAVs typically use smaller and error-prone IMUs, especially in yaw measurements [13], such as MicroElectroMechanical Systems (MEMS). This lack of quality lead to the development of computer vision algorithms such as Structure from Motion (SfM) that extract and match features between sequential images. The integration of these algorithms with the pose provided by the IMU and GPS greatly increases the accuracy of the georeferencing process.

Alternatively, Indirect Georeferencing can be used for systems that lack navigation equipment. This requires, however, the placement of Ground Control Points (GCP) to determine the camera's EP, which can be time-consuming and costly [20]. Furthermore, in a natural disaster scenario or in rough and inaccessible terrains, it is impracticable to place GCP.

1.1 Motivation

Portugal suffers annually from forest fires. In the last 10 years [01-01-2009, 15-10-2019] Portugal has summed a total of 1 451 692 ha (14 516.92 km^2) of burnt area [25], approximately 15.76% of the country's total area (not taking into account that the same area can burn more than once). Furthermore, in 2017, poor preparation, planning and coordination for such catastrophes resulted in the loss of human lives. For these reasons, forest fires have become a major issue in modern-day Portuguese society, which in turn has led to increasing investment in fire suppression equipment [11], mainly airborne. Moreover, there is ongoing investigation on how to approach forest fires more efficiently and effectively [1, 2].

Fire propagation has been under investigation for many decades. The effect of environment variables such as wind [30] and terrain [29] has been studied and models for the fire propagation have been developed. However, the absence of a method capable of georeferencing the fire front limits the usefulness of said models. Therefore, the aim of this thesis is to fill in this gap and develop an algorithm capable of performing forest fire geolocalization, enabling the use of fire propagation models to improve firefighting strategies.

1.2 Objectives

This work addresses three main objectives. Firstly, as was stated previously, it aims at developing a forest fire georeferencing algorithm using an aerial vehicle equipped with a digital camera, GPS, IMU and gimbal. The aerial imagery is assumed to be preprocessed, with the target pixels segmented from the background.

The data provided by the navigation equipment and gimbal will be affected by errors. The GPS accuracy depends on the topography and satellite coverage, the IMU depends on the quality of the accelerometer, gyroscope and magnetometer and, lastly, the gimbal depends on the precision of its actuators. These errors will be propagated and lead to an uncertain target position. For this reason, the second main objective of this work is to characterize the uncertainty of the target position. This characterization is crucial for the firefighting teams since it functions as a confidence metric of the georeferencing algorithm.

Finally, considering that the aerial vehicle will be flying at considerable heights to avoid disturbing the theatre of operations, the errors will be amplified by the increased distance to the fire. Hence, the third and final objective is the development of a methodology that takes into account the equipment errors and optimizes the estimated target position and uncertainty.

1.3 Thesis Outline

This document is structured in the following manner:

- Chapter 2 reviews the state of the art methodologies for the georeferencing problem and the background knowledge necessary to understand the proposed methodology. The limitations of related works and the proposed approach is also presented in this chapter.
- Chapter 3 explains with detail the proposed geolocalization and optimization algorithms.
- Chapter 4 validates the proposed methodology with simulations on a rough and a flat terrain.
- Chapter 5 details three experiments using a mobile phone and two UAV videos. The respective results are also presented in this chapter.
- Chapter 6 concludes with the achievements of this work and suggestions for future work.

2

Background & State-of-the-Art

Contents

This chapter is divided in three parts: Section 2.1 provides the theoretical background necessary to understand the georeferencing problem, Section 2.2 reviews the current state of the art approaches to solve it and finally Section 2.3 highlights the limitations of the state of the art and the main contributions of the proposed approach.

2.1 Background

2.1.1 Pinhole Camera Model

The pinhole camera model is used to map a tri-dimensional point to a two-dimensional point in the image plane using perspective transformation, and its geometry is represented in Figure 2.1. In order to use matrix notation, points are represented in homogeneous coordinates, so a 3D point is defined as $M = [X\ Y\ Z\ 1]^T$ and a 2D point as $m = [x\ y\ 1]^T$.

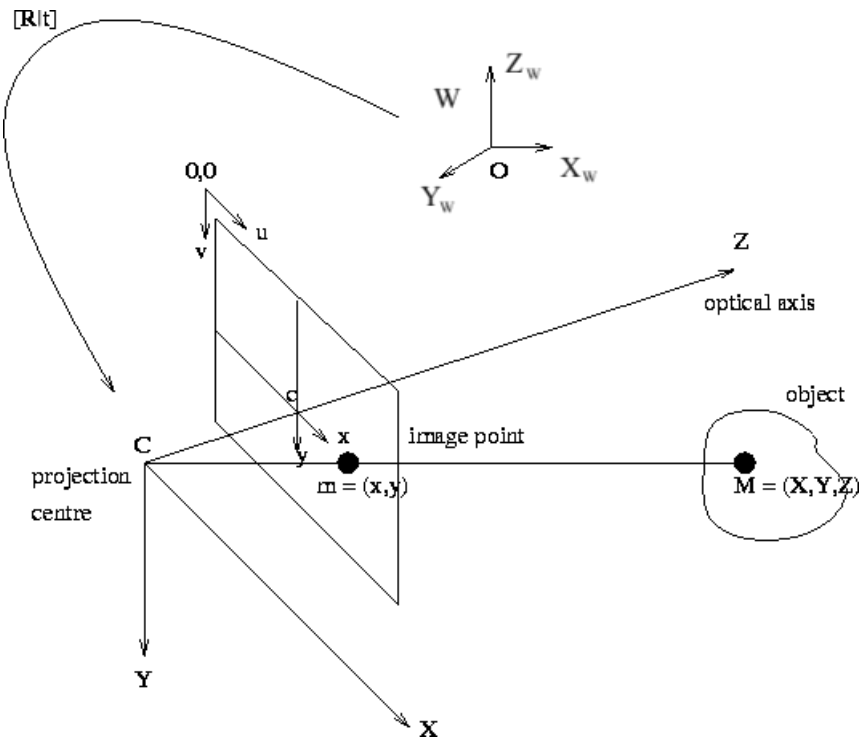


Figure 2.1: Pinhole camera model geometry

The model is described as

$$sm = K_{\text{int}}[R|t]M, \tag{2.1}$$

or

$$s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{2.2}$$

where:

- s - scale factor
- m - image coordinates $[x\ y]^T$ of projected point
- K_{int} - matrix of intrinsic parameters (IP)
- (f_x, f_y) - focal length
- $(c_x\ c_y)$ - principal point
- $[R|t]$ - matrix of extrinsic parameters (EP), which combines the rotation matrix and the translation vector
- M - 3D point coordinates $[X\ Y\ Z\ 1]^T$ in world coordinate space W.

The IP of a camera can be estimated with a calibration procedure and are crucial to extract metrics information from the image. The EP, represented by the joint rotation-translation matrix, establish the relationship between the camera and the world coordinate frame.

2.1.2 Lens Distortion

Real cameras introduce some amount of geometric distortions that must be taken into account if the camera measurements are to be accurate. The most common are tangential and radial distortions and are caused by the curvature and misalignment with the sensor plate, respectively.

Models have been developed [21] to compensate these distortions:

$$\begin{bmatrix} u \\ v \end{bmatrix}_u = d_t + d_r \cdot \begin{bmatrix} u \\ v \end{bmatrix}_d, \tag{2.3}$$

where $[u,\ v]_u^T$ is the undistorted pixel, $[u,\ v]_d^T$ is the distorted pixel, and

$$d_t = \begin{bmatrix} 2p_1uv + p_2(r^2 + 2u^2) \\ 2p_2uv + p_1(r^2 + 2v^2) \end{bmatrix}, \tag{2.4}$$

$$d_r = 1 + k_1r^2 + k_2r^4 + k_3r^6, \tag{2.5}$$

$$r^2 = u^2 + v^2, \tag{2.6}$$

where k_1, k_2, k_3, p_1 and p_2 the model parameters and r the distance to the image center,

2.1.3 Coordinate Frame Transformation

We intend to map points of interest in images to geodetic coordinates (latitude, longitude, elevation), so it is relevant to review coordinate frame transformations. These are composed of two operations, translation and rotation. A translation $t = [t_1\ t_2\ t_3]^T$ is a 3-by-1 vector that points from one coordinate frame origin to another. A rotation can be described as the multiplication of three separate rotations matrices about each Cartesian axis, OX, OY and OZ,

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

These angles of rotation are commonly known as the Euler angles (γ,β,α) and are used to describe the orientation of a rigid body with respect to a known coordinate frame. Using homogeneous coordinates, the matrices become as follows:

$$T_a^b = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.7}$$

where T_a^b stands for transformation from coordinate frame a to coordinate frame b, and r_{ij} are the elements of the rotation matrix. Matrices in the form of (2.7) enable the use of a single multiplication to perform a translation and a rotation. These are multiplied according to the order of the transformations one intends to perform on a coordinate frame.

2.1.4 Uncertainty Propagation

As was stated, one of the main objectives is to characterize the uncertainty of the estimated position. If the flat earth simplification was considered, it would be possible to linearize the projection as a function of the surface. Defining the projection from pixels to world coordinates as $f(x)$, the error propagation is calculated by first computing the Taylor expansion of f,

$$f(x)|_{x=x_0} \approx f(x_0) + Jx_0, \tag{2.8}$$

where x_0 is state variable with the IP, EP and target pixels, and J is the Jacobian matrix of $f(x)$. Given the input error covariance matrix Σ^x, the output covariance matrix Σ^f can be computed as

$$\Sigma^f = J\Sigma^x J^T. \tag{2.9}$$

This procedure is represented in Figure 2.2, where x_s is the camera pose and C_s is the covariance matrix which contains information about position and orientation errors. These are propagated to the world coordinate frame, where x_m is the target position and C_m is the estimated covariance. Target detection uncertainty, C_o, can also be taken into account.

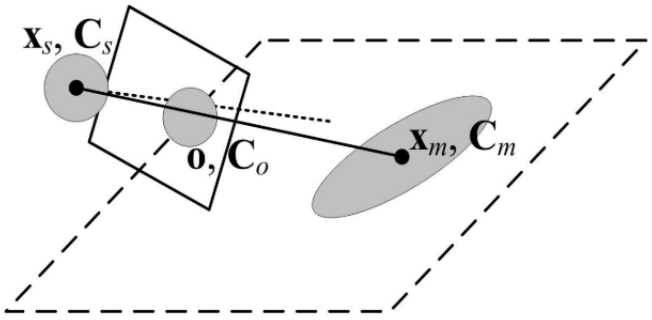


Figure 2.2: Uncertainty propagation (adapted from [31])

However, the flat earth simplification cannot be used for rough terrains. In this case, the error propagation becomes non-linear and needs to be calculated using the UT. A set of so-called sigma points is computed, whose mean and covariance are the same as the input signal, i.e., the EPs. These points approximate a Gaussian density distribution. By applying the non-linear transformation f to the sigma points, the projected points on the DEM capture the mean and covariance of the new density. Figure 2.3 exemplifies the methodology of the UT. The UT

(a) Input density and sigma points

(b) Non-linear transformation applied to the sigma points

(c) Input and resulting output densities

Figure 2.3: Unscented Transform (adapted from [8])

can also be applied to linear functions and the result is the same as the one obtained with the Taylor expansion. In fact, it is better since it does not compute the Jacobian matrix and is, therefore, less computationally demanding.

The UT can be summarized as follows [8]:

1. Sigma point and weight selection

$$\begin{aligned}
\chi^{[0]} &= \mu \\
\chi^{[i]} &= \mu + (\sqrt{(n+\lambda)\Sigma}) \quad \text{for} \quad i = 1, \ldots, n \\
\chi^{[i]} &= \mu - (\sqrt{(n+\lambda)\Sigma}) \quad \text{for} \quad i = n+1, \ldots, 2n \\
w_m^{[0]} &= \frac{\lambda}{n+\lambda} \quad , \quad w_c^{[0]} = w_m^{[0]} + (1 - \alpha^2 + \beta) \\
w_m^{[i]} &= w_c^{[i]} = \frac{1}{2(n+\lambda)} \quad \text{for} \quad i = 1, \ldots, 2n \\
\lambda &= \alpha^2(n+\kappa) - n
\end{aligned} \tag{2.10}$$

   where μ is the is mean, Σ is the covariance matrix, n is the dimension variable, α and κ are tunable variables that influence how far the sigma points are from the mean, β is a tunable variable to incorporate higher order information in the Taylor Series expansion of the covariance (for Gaussians β=2 is the optimal choice), $w_m^{[i]}$ and $w_c^{[i]}$ are the sigma point mean and covariance weights, respectively.

2. Sigma point propagation

$$y^{[i]} = f(\chi^{[i]}), \tag{2.11}$$

   where f is the non-linear transformation.

3. Recover the propagated distribution

$$\mu' = \sum_{i=0}^{2n} w_m^{[i]} y^{[i]} \tag{2.12}$$

$$\Sigma' = \sum_{i=0}^{2n} w_c^{[i]} (y^{[i]} - \mu')(y^{[i]} - \mu')^T, \tag{2.13}$$

where μ' and Σ' are the mean and covariance, respectively, of the propagated sigma points.

2.1.5 Digital Elevation Map

A DEM, also known as Digital Elevation Model or Digital Terrain Model, is a 3D representation of a surface. It is sometimes confused with a Digital Surface Model. While the first only contemplate the height of the Earth's surface, the latter takes into account structures on it, as can be seen in Figure 2.4.

These maps are generally represented as a Triangular Irregular Network (TIN), regular grids or contour lines, as shown in Figure 2.5. TIN concentrates more data points where the terrain has more variation and fewer where the surface is flatter. The regular grid has equally distanced sample points, and this distance is known as the DEM's

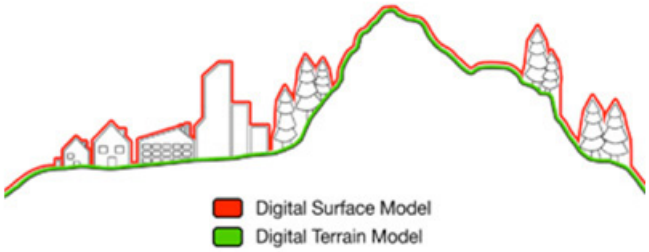


Figure 2.4: Difference between DSM and DEM/DTM (adapted from [5])

resolution. While a TIN manages to represent rough surfaces more accurately, it is more complex, therefore more demanding and time-consuming to process. On the other hand, grid models are easy to store and manipulate [28].

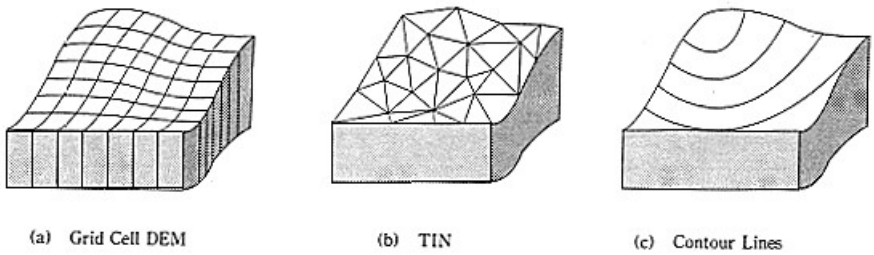


Figure 2.5: Different DEM representations (adapted from [4])

When querying a point of the DEM that does not match the exact location of a sample point, different approaches can be taken, depending on the map's representation. In case of a TIN, the solution is to solve the plane equation for the triangle where the query point lies. In case of a regular grid, different interpolation methods can be used. According to [18], for grids with finer resolutions (up to 30 meters), linear methods such as linear or bi-linear interpolation provide good results with low computational cost and processing time. On the other hand, grids with more coarse resolution (100 to 1000 meters) require polynomial interpolation methods for better results.

2.1.6 Kalman Filter

In this work, two non-linear Kalman filters were considered, the EKF and CKF, and are described in the following Subsections. A discrete non-linear system with Gaussian noise can be characterized as follows:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{w}_k \tag{2.14a}$$

$$\mathbf{z}_{k+1} = \mathbf{h}(\mathbf{x}_{k+1}) + \mathbf{v}_{k+1} \tag{2.14b}$$

where $\mathbf{x}_k$ is the state vector, $\mathbf{z}_k$ the measurement vector, $\mathbf{f}(\mathbf{x}_k)$ and $\mathbf{h}(\mathbf{x}_k)$ the process and measurement models, respectively, and $\mathbf{w}_k$ ($\mathbf{w} \sim \mathcal{N}(0, \mathbf{Q}_k)$)and $\mathbf{v}_k$ ($\mathbf{v} \sim \mathcal{N}(0, \mathbf{R}_k)$) the uncorrelated system and measurement noises with covariances given by $\mathbf{Q}_k$ and $\mathbf{R}_k$.

2.1.6.A Extended Kalman Filter

The EKF was developed as a modification of the original Kalman Filter to deal with systems having non-linear dynamics or nonlinear measurement models. It proposes the linearization of said models by performing a first-order Taylor series expansion.

The EKF algorithm is composed of a prediction step and an update step.

Prediction phase:

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{f}(\hat{\mathbf{x}}_{k|k}), \tag{2.15}$$

$$\hat{\mathbf{z}}_{k+1|k} = \mathbf{h}(\hat{\mathbf{x}}_{k+1|k}), \tag{2.16}$$

$$\mathbf{P}_{k+1|k} = \Phi_{k+1|k}\mathbf{P}_{k|k}\Phi^T_{k+1|k} + \mathbf{Q}_k, \tag{2.17}$$

where $\hat{\mathbf{x}}_{k+1|k}$ is the predicted state at time $k+1$, $\hat{\mathbf{z}}_{k+1|k}$ is the predicted measurement at time $k+1$, $\mathbf{P}_{k+1|k}$ is the error covariance matrix for the estimate at time $k+1$ and $\Phi_{k+1|k}$ is the Jacobian of the system dynamics model with respect to the state.

Kalman Gain:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{H}^T_{k+1}[\mathbf{H}_{k+1}\mathbf{P}_{k+1|k}\mathbf{H}^T_{k+1} + \mathbf{R}_{k+1}]^{-1}\mathbf{H}_{k+1}\mathbf{P}_{k+1|k}, \tag{2.18}$$

where $\mathbf{H}_{k+1}$ is the Jacobian of the measurement model.

Update phase:

$$\mathbf{v}_{k+1} = \mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1|k} \tag{2.19}$$

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1}\mathbf{v}_{k+1} \tag{2.20}$$

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}^T_{k+1}[\mathbf{H}_{k+1}\mathbf{P}_{k+1|k}\mathbf{H}^T_{k+1} + \mathbf{R}_{k+1}]^{-1}\mathbf{H}_{k+1}\mathbf{P}_{k+1|k} \tag{2.21}$$

2.1.6.B Cubature Kalman Filter

For highly nonlinear systems, the EKF becomes unsuitable as the Taylor series expansion induces linearization errors that slow or prevent filter convergence. Following the Unscented Kalman Filter (UKF) approach, the CKF also uses sample points, called cubature points, of equal weight to calculate the mean and covariance of state variables

which after non-linear transformation yield performance of nonlinear optimal approximation, filtering accuracy and numerical stability [16]. These cubature points are chosen based on the spherical-radial transformation and guarantee the exact approximation of the Gaussian distribution up the third moment. The resulting filter ends up being a special case of the UKF, for α = 1 and β = κ = 0.

The CKF algorithm [9] is defined by a time update and measurement update:

Time update:

1. Cubature point generation $\chi_k^{(i)}(i = 1, \ldots, 2n)$:

$$\chi_k^{(i)} = \sqrt{\hat{\mathbf{P}}_k}\xi^{(i)} + \hat{\mathbf{x}}_k, \tag{2.22}$$

where $\xi = \sqrt{n}[1]$, n being the number of state variables and $[1]$ the full permutation of the n dimension unit vector,

$$[1] = \left[\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad , \ldots , \quad \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad , \ldots , \quad \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \right], \quad \text{for } [1]_i \in R^n.$$

2. Propagate the cubature points:

$$\chi_{k+1|k}^{(i)} = \mathbf{f}(\chi_k^{(i)}) \tag{2.23}$$

3. Estimate the propagated state and covariance matrix for time $k + 1$:

$$\overline{\mathbf{x}}_{k+1|k} = \frac{1}{2n}\sum_{i=1}^{2n} \chi_{k+1|k}^{(i)} \tag{2.24}$$

$$\overline{\mathbf{P}}_{k+1|k} = \frac{1}{2n}\sum_{i=1}^{2n} [\chi_{k+1|k}^{(i)} \chi_{k+1|k}^{T(i)}] - [\overline{\mathbf{x}}_{k+1|k}\overline{\mathbf{x}}_{k+1|k}^T] + \mathbf{Q}_k \tag{2.25}$$

Measurement update:

1. Cubature point regeneration $\chi_{k+1}^{(i)}(i = 1, \ldots, 2n)$:

$$\chi_{k+1}^{(i)} = \sqrt{\overline{\mathbf{P}}_{k+1|k}}\xi + \overline{\mathbf{x}}_{k+1|k}, \tag{2.26}$$

2. Propagate cubature points through the measurement model:

$$\gamma_{k+1}^{(i)} = \mathbf{h}(\chi_{k+1}^{(i)}) \tag{2.27}$$

3. Estimate the predicted measurement, innovation covariance matrix and cross-covariance matrix:

$$\overline{\mathbf{y}}_{k+1} = \frac{1}{2n}\sum_{i=1}^{2n}\gamma_{k+1}^{(i)} \tag{2.28}$$

$$\mathbf{P}_{k+1}^{yy} = \frac{1}{2n}\sum_{i=1}^{2n}[\gamma_{k+1}^{(i)}\gamma_{k+1}^{T(i)}] - [\overline{\mathbf{y}}_{k+1}\overline{\mathbf{y}}_{k+1}^{T}] + \mathbf{R}_k \tag{2.29}$$

$$\mathbf{P}_{k+1}^{xy} = \frac{1}{2n}\sum_{i=1}^{2n}[\chi_{k+1}^{(i)}\gamma_{k+1}^{T(i)}] - [\overline{\mathbf{x}}_{k+1|k}\overline{\mathbf{y}}_{k+1}^{T}] \tag{2.30}$$

4. Estimate the Kalman gain, update state and covariance matrix:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^{xy}(\mathbf{P}_{k+1}^{yy})^{-1} \tag{2.31}$$

$$\hat{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_{k+1|k} + \mathbf{K}_k(y_{k+1} - \overline{\mathbf{y}}_{k+1}) \tag{2.32}$$

$$\hat{\mathbf{P}}_{k+1} = \overline{\mathbf{P}}_{k+1|k} - \mathbf{K}_{k+1}\mathbf{P}_{k+1}^{yy}\mathbf{K}_{k+1}^{T} \tag{2.33}$$

2.2 State-of-the-Art

This subsection divides and details the different state-of-the-art approaches for the georeferencing problem according to their base methodology.

2.2.1 Optic-Ray Surface Intersection

When projecting a 3D point from the world to the 2D image plane, there is a loss of information. The 2D points are defined up to an unknown scale factor, defined by the distance from the 3D point to the principal point of the camera. To invert the projection, assuming that the IP and EP are known, either this distance or information regarding the surface where the point lies needs to be known.

2.2.1.A Flat Earth Hypothesis

Leira et al. [27] propose the intersection of the optic ray, whose direction is defined by the pixel to map, with a plane with Z equal to zero. The UAV is equipped with a gimbal and a thermal camera, and the transformation between the camera frame and the inertial frame is known. By doing this assumption, the scaling factor can be calculated and the optic ray is projected to the surface. An accuracy of 7.8 meters was achieved for a variable flight height, between 50 and 100 meters.

Xiang and Tian [38] propose an automatic georeferencing algorithm that estimates the target's world coordinates and mosaics the images together based on their estimated geographical positions. Since the purpose of that paper was to find horizontal coordinates, it assumes that all targets are on the same elevation plane.

Parcelas [33] studies how the variation of the camera's intrinsic and distortion parameters, by varying the lenses configuration, influences the results of the geolocation procedure. The flat earth hypothesis was valid since the purpose was to locate vessels. Different tests were conducted with a variable height between 80 meters and 100 meters, with the mean absolute error varying from 7 meters to 35 meters.

2.2.1.B Digital Surface

Sheng [36] tests three different algorithms to solve the optic ray-DEM intersection: Iterative Photogrammetry (IPG), Ray-Tracing (RT) and Iterative Ray-Tracing (IRT). The IPG's convergence depends on the initial elevation, the view angle and profile inclination angle and is prone to fail with occlusions or rough terrains while IRT's convergence depends on the step chosen to iterate along the optic ray. RT actually calculates the intersection point so it is the most robust method, however, it is more computationally demanding when compared to IRT and IPG. If computational power is available, the best option is the RT, otherwise, the IRT seems the best option as it is not so prone to fail with occlusions as the IPG and its convergence only depends on the step size.

Sheng [37] continue the previous work and review in detail the IPG, as it is the most promising and efficient of the methods. The convergence condition and convergence speed are analyzed.

2.2.2 Image Feature Extraction and Matching

2.2.2.A Structure from Motion

Structure from motion is a method that relies on feature extraction and matching from sequential images captured with different camera poses. Having multiple images with the 2D coordinates of these features enables stereo-vision techniques to solve their 3D coordinates. These coordinates are then georeferenced in two possible ways: the camera pose in the inertial frame is known, via GPS and IMU, or with the placement of GCP. SfM is usually followed by a Bundle Adjustment (BA). BA is an algorithm that takes as input the targets' 3D and 2D positions in the camera and image frames, respectively, and performs a least-squares minimization with the reprojection error as a cost function. In case the camera's IP and EP are known, these too are refined. If not, they can be estimated, increasing the complexity of the algorithm.

Forlani et al. [15] propose the use of SfM to match a set of images acquired under poor Global Navigation Satellite System (GNSS) coverage, designated as the master block, to a set of georeferenced images acquired under nominal GNSS coverage, designated as the auxiliary block. Features are extracted and used to match the images from the master to the auxiliary blocks and are followed by a BA. The method was tested with different camera calibrations. A Root Mean Squared Error (RMSE) of centimetres was achieved for a master block flown at 30 meters and an auxiliary block flown at 100 meters.

He et al. [20] present the mathematical premise of SfM with detail. Prior knowledge of the vehicle trajectory (planar motion) is used to simplify the problem from 6 Degrees Of Freedom (DOF) to 3DOF, enabling a 2-point approach. An incremental approach is compared to a global approach in terms of EP recovery, followed by a BA. The RMSEs obtained for both methodologies are on the centimetre order, with the global strategy performing slightly better than the incremental, for a maximum flight height of 50 meters.

2.2.2.B Geo-Referenced Imagery

This methodology consists of matching images captured by an aerial vehicle with available georeferenced imagery, such as Google Earth. Conte et al. [12] propose an image registration approach by pattern-matching the images collected at 100 meters from a micro aerial vehicle with satellite imagery. Multiple measurements are taken and a recursive least square filter is applied. The method is compared against the intersection of the optic ray assuming the flat earth simplification. The proposed method achieved a RMSE of 2.25m while the intersection method best result was a RMSE of 22 meters.

Hamidi and Samadzadegan [19] propose the IPG algorithm combined with EP refinement using feature matching with georeferenced imagery. The DEM used was the Shuttle Radar Topography Mission [14], with a spatial resolution of 90 meters. On a first stage, the EP are computed using the information provided by the UAV's IMU and GPS. On a second stage, the EP are adjusted and the IPG is applied. The mean UAV height in the experimental procedure was 400 meters. The refinement improved the position accuracy by 100 meters to 14.476 meters.

2.2.3 Line-of-Sight Filtering

Barber et al. [10] applies the flat earth simplification to calculate the target's coordinates, but its main innovation is applying a recursive least square filter to multiple observations while performing a loitering pattern. By doing so, the geo-location errors decreased from 40 meters to less than 5 meters.

Ponda et al. [34] introduces a bearings-only 3D target coordinate estimation with an EKF. This model filters the azimuth and elevation angles between the aerial vehicle and the target for every measurement. Three cases are simulated: stationary target, slow-moving target and fast-moving target. In all cases, a stationary process is assumed and the process noise is tuned to allow unknown target motion. The filter manages to track the target for the stationary and slow-moving cases and fails to do so in the fast-moving case. Trajectory optimization is also studied in that paper, but it is out of the scope of this thesis so it will not be reviewed.

Xu et al. [40] also estimate the target's position by filtering multiple bearing measurements with a CKF [9] while performing a loitering trajectory, centred on the target. The filter's initial state is calculated with the IPG using the ASTER-GDEM V2 [3], which has 30-meter spatial resolution. The CKF method is compared against the standalone IPG and flat earth simplification. Two experiments are performed on different terrains, rough and flat. In the two cases, the standalone IPG achieves similar accuracy, 39.6 and 36.2 meters. In the flat terrain, the CKF and the flat earth simplification achieve similar RMSEs, 10.8 and 12.9 meters respectively. Finally, in the rough terrain, the flat earth has a bad performance (105.6 meters), because this approximation is invalid for this terrain type, whereas the proposed method achieves a RMSE of 13.8 meters. For both experiments, the UAV flew at a maximum height of 500 meters.

2.2.4 Others

Zhang et al. [42] use a stereo vision technique to determine the target relative height with respect to the UAV. A yaw bias estimation algorithm is also proposed. Two different trajectories were tested, overflight and loitering. Firstly, the relative target height and yaw bias are estimated using multiple bearing measurements of the same target. Then its coordinates are estimated. The proposed method achieves a horizontal and vertical accuracy of 0.7 and 0.5 meters, respectively, for the loitering trajectory, and 1.77 and 1.15 meters for the overflight, for maximum flight heights of 20 meters. That work provides an interesting result by showing that the trajectory of the UAV is an important factor to take into account when determining the target's position.

Zhang et al. [41] continue the work developed in [42] and study the influence of the trajectory on the accuracy of the georeferencing algorithm. Trajectory planning is out of the scope of this thesis but was again demonstrated to influence the georeferencing accuracy.

2.3 Proposed Approach

The main goal of this work is to develop a georeferencing algorithm as part of the FIREFRONT project. As input, it has the image pixels that correspond to the fire, the GPS and IMU data from the aerial vehicle and should output the geodetic coordinates of said pixels and the uncertainty of the estimated target positions. Furthermore, we expect some errors in the telemetry data, so we also aim at developing a methodology that takes these errors into account and optimizes the target position.

The following limitations were found in the state of the art with respect to the problem we intended to address. The SfM and image registration techniques achieve very accurate results, however, these methodologies rely on feature extraction and matching. This requires rich, differentiated and stationary texture in the acquired imagery. In a forest environment, the texture is poor, repetitive and is subject to dynamic elements like the wind. In addition, smoke caused by the fire might prevent feature extraction. All these factors render the SfM unreliable. Other works intersect the optic ray with a surface, with the majority assuming the flat earth simplification. In our case, this hypothesis is not suitable, since the algorithm we aim at developing must work on all surfaces, and this simplification would be unreliable in rough terrains. Finally, the line-of-sight filtering requires an accurate initialization of the target position and is, therefore, a methodology that does not work as a solution by itself since it requires some other methodology to perform this initialization. The work that most resembles our proposal is [40], where they estimate the target's position using a Bearings-Only CKF initialized with the IPG algorithm. As was reviewed, the IPG is prone to diverge in rough terrains where the view angle is smaller than the profile inclination angle.

Considering the limitations encountered in the literature, we propose using the IRT algorithm, suitable for all terrains, to calculate the intersection between the optic ray and the EU-DEM v1.1, which has a spatial resolution of 25 meters. This algorithm will be used together with the Unscented Transform, to estimate the target position and characterize its uncertainty. Regarding the errors in the telemetry, we propose a new Bearings-Range filtering measurement model that takes advantage of the available distance between the vehicle and the target. Considering possible linearization errors induced by the EKF, a performance comparison is done with a CKF.

2.4 Contributions

This work's main contribution is a georeferencing algorithm applicable in a forest fire scenario. We propose using for the first time the IRT together with the UT, to calculate the position of the fire and characterize its uncertainty. Furthermore, the selected DEM has a finer spatial resolution when compared to the ones reviewed in the state of the art. Finally, a novel Bearings-Range model is presented to filter multiple observations of the same target, with the purpose of improving its position and uncertainty.

A paper [35], presented in Appendix B, was submitted and accepted in the RECPAD2020 Conference, a remote event that took place on October 30 2020.

Lastly, an application was developed to aid the Civil Protection. An Optical Character Reader (OCR) was

designed and integrated with the georeferencing algorithm, enabling the direct processing of the videos provided by the aerial vehicles. This application is detailed in Appendix C.

3

Methodology

Contents

The proposed approach can be divided in three steps: the pre-flight camera calibration, the georeferencing algorithm and the Bearings-Range filtering algorithm.

3.1 Camera Calibration

The camera calibration is an important procedure in computer vision that calculates the camera's intrinsic parameters and allows the extraction of metric information from bi-dimensional images. For this reason, before taking off, the camera must be calibrated, and assuming that the lenses configuration is not manipulated, these parameters remain constant throughout the flight.

Using one of the many computer vision tools available (OpenCV or MATLAB Image Processing and Computer Vision Toolbox), the calibration is done by moving and changing the pose of a known pattern in the camera's line of sight [43]. Usually this pattern is a checkerboard and the size of the checkerboard square is measured beforehand.

By performing the calibration, the camera's intrinsic and distortion parameters are calculated, crucial to define the optic ray from the projection center to the target pixel.

3.2 Coordinate Frames

This section describes the five coordinate frames considered: camera, gimbal, body, vehicle and inertial. This 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 vehicle frame is a North-East-Down frame centered on the vehicle's centre of mass and is related to the inertial frame by a simple translation. This work considered that the camera, gimbal and UAV centers of mass coincided with the GPS position, meaning that the translations in the frame transformations are zero, $t_A^B = [0\ 0\ 0]^T$. Still, they are presented in the following Subsections for consideration in future works.

3.2.1 Camera Frame

The camera frame, represented in Figure 3.1, has its origin in the optical center, the x^C axis points to the right of the image plane , the y^C axis points downward on the image plane and z^C axis points in the direction of the optical axis of the camera. Defining the rotation from camera to the gimbal coordinates as R_C^G and t_C^G as the translation from the camera optical center to the gimbal center of mass, the transformation from the camera to the gimbal coordinate frame is defined by

$$T_C^G = \left[\begin{array}{c|c} R_C^G & t_C^G \\ \hline 0 & 1 \end{array}\right], \tag{3.1}$$

where

$$R_C^G = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{3.2}$$

and t_C^G the translation from the camera optic center to the gimbal center of mass.

The opposite transformation, from gimbal to camera coordinate frames is defined by

$$T_G^C = \left[\begin{array}{c|c} R_G^C & t_G^C \\ \hline 0 & 1 \end{array}\right], \tag{3.3}$$

where

$$R_G^C = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \tag{3.4}$$

and t_G^C the translation from the gimbal center of mass to the camera optic center.

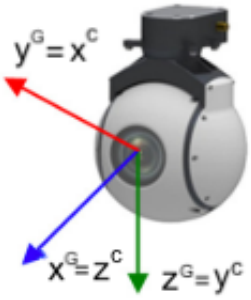


Figure 3.1: Camera and gimbal coordinate frames (adapted from [40])

3.2.2 Gimbal Frame

The gimbal coordinate frame, represented in Figures 3.2 and 3.1, has two degrees of freedom around y^G and z^G due to its pan and tilt movements, respectively. Defining the pan (elevation) and tilt (azimuth) angles as α_{el} and α_{az}, and the translation from the gimbal to the UAV centers of mass as t_G^B, the transformation from the gimbal coordinate frame to body coordinate frame is defined by

$$T_G^B = \left[\begin{array}{c|c} R_G^B & t_G^B \\ \hline 0 & 1 \end{array}\right], \tag{3.5}$$

where

$$R_G^B = R_z(-\alpha_{az})R_y(-\alpha_{el}) = \begin{bmatrix} \cos\alpha_{el}\cos\alpha_{az} & -\sin\alpha_{az} & \sin\alpha_{el}\cos\alpha_{az} \\ \sin\alpha_{az}\cos\alpha_{el} & \cos\alpha_{az} & \sin\alpha_{az}\sin\alpha_{el} \\ -\sin\alpha_{el} & 0 & \cos\alpha_{el} \end{bmatrix} \tag{3.6}$$

and t_G^B is the translation from the gimbal to the UAV center of mass in the body coordinate frame.

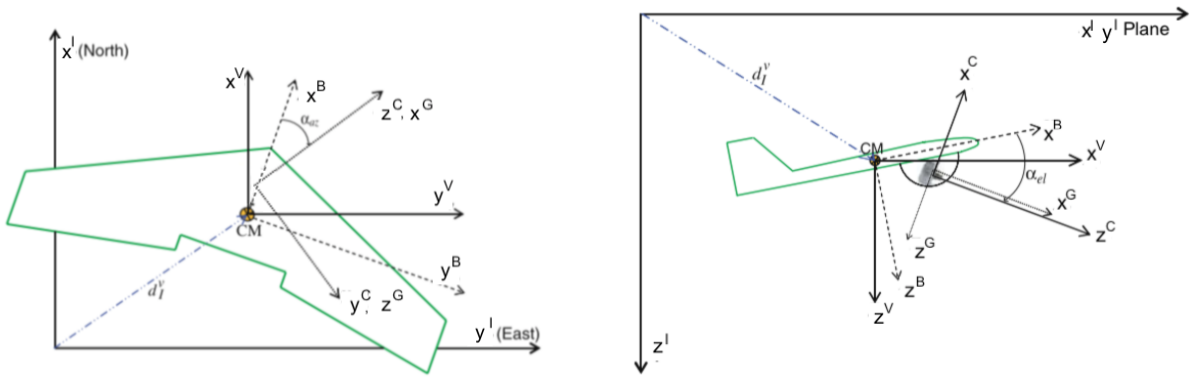
The opposite transformation, from body to gimbal coordinate frames is defined by

$$T_B^G = \left[\begin{array}{c|c} R_B^G & t_B^G \\ \hline 0 & 1 \end{array}\right], \tag{3.7}$$

where

$$R_B^G = R_y(\alpha_{el})R_z(\alpha_{az}) = \begin{bmatrix} \cos\alpha_{el}\cos\alpha_{az} & \sin\alpha_{az}\cos\alpha_{el} & -\sin\alpha_{el} \\ -\sin\alpha_{az} & \cos\alpha_{az} & 0 \\ \sin\alpha_{el}\cos\alpha_{az} & \sin\alpha_{az}\sin\alpha_{el} & \cos\alpha_{el} \end{bmatrix} \tag{3.8}$$

and t_B^G is the translation from the UAV to the gimbal center of mass in the gimbal coordinate frame.



(a) Top view

(b) Side view

Figure 3.2: UAV, gimbal and camera frames (adapted from [10])

3.2.3 Body Frame

The body frame describes the aircraft movement and has its origin in the center of mass of the UAV. The x^B axis points in the direction of the nose, the y^B axis points towards the right wing and the z^B axis points towards the aircraft belly, as represented in Figure 3.2.

Defining the roll (ϕ), pitch (θ) and yaw (ψ) angles as the movement of the UAV around the axis x^B, y^B and z^B, respectively, and t_B^V as the translation from the UAV center of mass to the GPS position, the transformation

from the body to the vehicle coordinate frame is defined as

$$T_B^V = \left[\begin{array}{c|c} R_B^V & t_B^V \\ \hline 0 & 1 \end{array}\right], \tag{3.9}$$

where $R_B^V = R_z(-\psi)R_y(-\theta)R_x(-\phi)$,

$$R_B^V = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\beta\cos\phi \end{bmatrix}, \tag{3.10}$$

and t_B^V is the translation from the UAV center of mass to the GPS position in the body coordinate frame.

The opposite transformation, from vehicle to body coordinate frame, is defined as

$$T_V^B = \left[\begin{array}{c|c} R_V^B & t_V^B \\ \hline 0 & 1 \end{array}\right], \tag{3.11}$$

where $R_V^B = R_x(\phi)R_y(\theta)R_z(\psi)$.

$$R_V^B = \begin{bmatrix} \cos\psi\cos\theta & \sin\psi\cos\theta & -\sin\theta \\ \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \cos\theta\sin\phi \\ \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi & \cos\beta\cos\phi \end{bmatrix}, \tag{3.12}$$

and t_V^B is the translation from the GPS position to the UAV center of mass in the vehicle coordinate frame.

3.2.4 Vehicle Frame

The vehicle coordinate frame is centered on the GPS position and its Cartesian coordinates are calculated based on the its Geodesic coordinates, latitude (ϕ), longitude (λ) and altitude (h) above the geoid provided by the GPS receptor.

To convert geodetic coordinates do Cartesian coordinates and vice-versa, it is necessary to choose an adequate ellipsoid. Considering that the main ellipsoid of reference when using GPS systems is the World Geodetic System 1984 (WGS84), this is the one chosen for this work. The WGS84 defines the Earth-Centered Earth-Fixed (ECEF) coordinate system characterized by the following parameters:

- Semi-major Axis - a = 6378137 meters;
- Semi-minor Axis - b = 6356752 meters;
- Flattening - f = 1/298.257223563;

- Eccentricity - e = 0.08181919;

- Radius of curvature in the prime vertical - $N = \frac{a}{\sqrt{1-f(2-f)\sin^2\phi}}$

Given a point in geodetic coordinates $P = [\phi\ \lambda\ h]^T$, the coordinates in the ECEF system are given by

$$P_{ECEF} = \begin{bmatrix} x_P \\ y_P \\ z_P \end{bmatrix} = \begin{bmatrix} (N+h)\cos(\phi)\cos(\lambda) \\ (N+h)\cos(\phi)\sin(\lambda) \\ ((1-f)^2 N + h)\sin(\phi) \end{bmatrix}. \tag{3.13}$$

According to [44], the most accurate transformation from ECEF to geodetic coordinates is based on the following equations, deduced in [22]:

$$r = \sqrt{x_P^2 + y_P^2} \tag{3.14}$$

$$F = 54 b^2 z_P^2 \tag{3.15}$$

$$G = r^2 + (1+e^2)z^2 - e^2(a^2 - b^2) \tag{3.16}$$

$$c = \frac{e^4 F r^2}{G^3} \tag{3.17}$$

$$s = \sqrt[3]{1 + c + \sqrt{c^2 + 2c}} \tag{3.18}$$

$$T = \frac{F}{3(s + \frac{1}{s} + 1)^2 G^2} \tag{3.19}$$

$$Q = \sqrt{1 + 2e^4 T} \tag{3.20}$$

$$r_0 = -\frac{Te^2r}{1+Q} + \sqrt{\frac{a^2}{2}(1+\frac{1}{Q}) - \frac{T(1-e^2)z^2}{Q(1+Q} - \frac{Tr^2}{2}} \tag{3.21}$$

$$U = \sqrt{(r - e^2 r_0) + z_p^2} \tag{3.22}$$

$$V = \sqrt{(r - e_0^r)^2 + (1-e^2)z_p^2} \tag{3.23}$$

$$z_0 = \frac{b^2 z}{aV} \tag{3.24}$$

$$e_0 = \frac{a^2 - b^2}{b^2} \tag{3.25}$$

The altitude is defined by

$$h = U(1 - \frac{b^2}{aV}) \tag{3.26}$$

and the latitude (ϕ) and longitude (λ) by

$$\phi = \arctan(\frac{z_p + e_0 z_0}{r}) \tag{3.27}$$

$$\lambda = \arctan(\frac{y}{x}) \tag{3.28}$$

3.3 Georeferencing Algorithm

Defining $\vec{P}^V_{Target}$ as the vector pointing from the UAV to the target in the inertial coordinate frame and $\vec{P}^I_V$ as the vector pointing from the inertial frame origin to the UAV, the target position is defined as the sum of these vectors. To obtain $\vec{P}^V_{Target}$, the vector pointing from the camera optical center to the target pixel $[u, v]^T$ must be defined,

assuming $f_x \approx f_y$,

$$\vec{P}^{C}_{Target} = \begin{bmatrix} u - c_x \\ v - c_y \\ \frac{f_x + f_y}{2} \end{bmatrix}, \tag{3.29}$$

and transformed to the vehicle coordinate frame using the coordinate frame transformations described in Section 3.2,

$$\vec{P}^{V}_{Target} = \begin{bmatrix} X'_d \\ Y'_d \\ Z'_d \end{bmatrix} = T^{V}_{B} T^{B}_{G} T^{G}_{C} \vec{P}^{C}_{Target}, \tag{3.30}$$

where $\vec{P}^{V}_{Target}$ references the target vector to the inertial frame.

3.3.1 Digital Elevation Model - *EU-DEM v1.1*

Currently, the DEM with the finest resolution of Portugal's topography is the *EU_DEM v1.1*, with a 25 meter spacial resolution and a RMSE of 7 meters with respect to the vertical accuracy. This map was developed as part of the European Union's Copernicus Programme and its Coordinate Reference System (CRS) is the EPSG:3035 (ETRS89, LAEA). Its ellipsoid of reference is the GRS80 which, in practicality, is identical to the WGS84. More information on this DEM can be found in [6].

Although this version has not been validated yet, recent work [32] has shown that it combines the higher resolution of the ASTER GDEM and the vertical accuracy of both SRTM DEM and ASTER GDEM in low and high slope areas, respectively. Using the Quantum Geographic Information System (QGIS) software, presented in Figure 3.3, this elevation model was converted to WGS84 reference system, making it compatible with the proposed georeferencing algorithm.

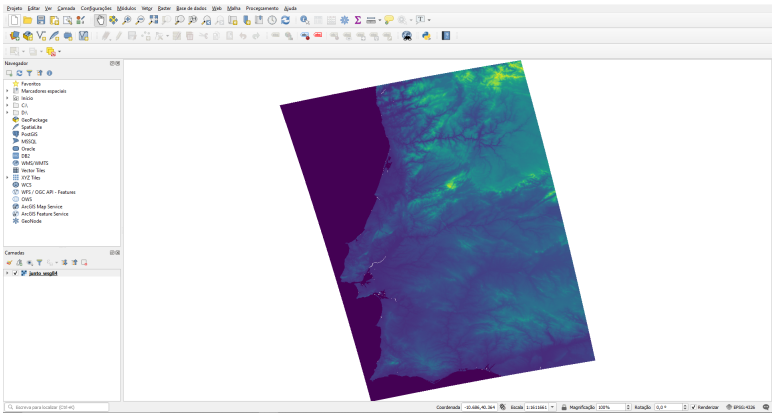


Figure 3.3: QGIS with *EU-DEM v1.1*

3.3.2 Iterative Ray-Tracing

The IRT is the most suitable algorithm to perform the intersection between the optic ray and a DEM. As was reviewed in Section 2.2, the IRT method does not have a convergence condition as the IPG, which prevents it from working properly on all terrains, and is more efficient than the RT. Furthermore, the RT method requires a DEM in TIN format, which is more complex and computationally demanding.

Given the camera position $\mathbf{R}_0 = [X_s\ Y_s\ Z_s]^T$ in the inertial frame and the normalized pointing vector

$$\mathbf{R}_d = \begin{bmatrix} X_d \\ Y_d \\ Z_d \end{bmatrix} = \frac{1}{||\vec{P}^V_{Target}||} \cdot \vec{P}^V_{Target} = \frac{1}{\sqrt{X_d'^2 + Y_d'^2 + Z_d'^2}} \cdot \begin{bmatrix} X_d' \\ Y_d' \\ Z_d' \end{bmatrix} \tag{3.31}$$

from the aerial vehicle to the target, in the vehicle frame, the ray $\mathbf{R}$ that starts in the vehicle and points to the target in the inertial frame is defined as

$$\mathbf{R}(t) = \mathbf{R}_0 + t \cdot \mathbf{R}_d = \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} + t \cdot \begin{bmatrix} X_d \\ Y_d \\ Z_d \end{bmatrix}, \tag{3.32}$$

where t is the step and represents the distance between a point $\mathbf{R}(t)$ on the ray and the origin $\mathbf{R}_0$. This is valid since the inertial and the vehicle frame have the same orientation. When the ray elevation Z_{R} becomes less than the surface elevation Z_{DEM}, the intersection is detected.

Algorithm 1: Iterative Ray-Tracing

Input : Telemetry from GPS and IMU, camera IP, target pixels, DEM
Output: Target coordinates in the inertial coordinate frame $\mathbf{t}_{\mathrm{I}} = [x\ y\ z]$

```
Define ray origin R_0;
Define ray direction R_d;
Define step size t;
while No intersection do
    Extend ray with (3.32);
    if Z_R ≤ Z_DEM then
        Intersection;
    else
        Augment t;
    end
end
```

3.3.2.A Maximum Height Iteration

Apart from the aerial vehicle recording the fire, others will be fighting the fire. It is expected that the former will be flying at higher altitudes in order not to perturb the theater of operations. This places the camera position $\mathbf{R}_0$ at

a much higher altitude, forcing the IRT to perform more iterations or the user to increase step size in order to have less iterations. A solution to avoid this problem is to begin iterating at the maximum height of the loaded DEM, $Z_{\max}$. This way, the method only starts where it is factual that an intersection will occur.

By defining a scaling factor,

$$\lambda = \frac{Z_{\max} - Z_{\mathbf{R}_0}}{Z_d}, \quad \text{for } \lambda > 0 \tag{3.33}$$

where $Z_{\mathbf{R}_0}$ is the height of the camera and Z_d is the third component of the optic ray, it is possible to calculate the new starting point, with $Z = Z_{\max}$

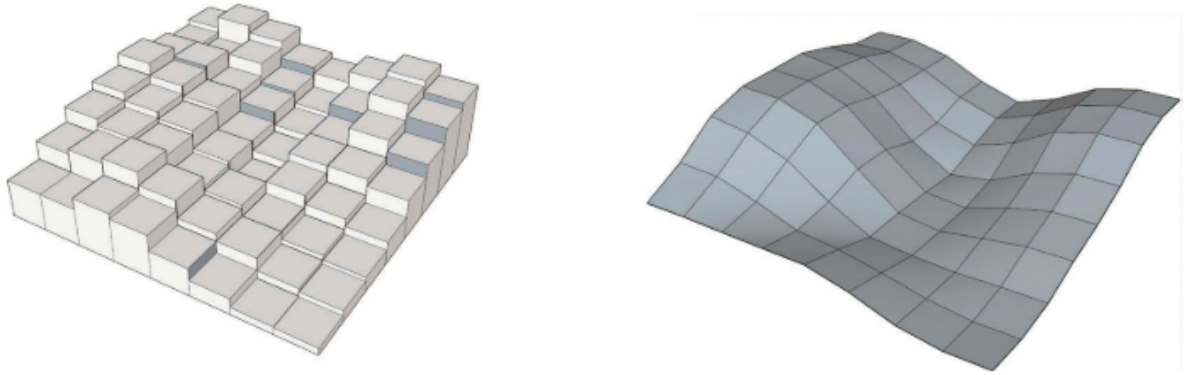
$$\mathbf{R}'_0 = \mathbf{R}_0 + \lambda \cdot \mathbf{R}_d. \tag{3.34}$$

3.3.2.B Dynamic Step

Defining and adequate value for t is crucial as it influences the speed and the accuracy of the algorithm. A bigger step increases convergence speed but decreases the accuracy, while a smaller step decreases speed but increases accuracy. A compromise can be achieved by implementing a dynamic step. The basic idea is the following. The algorithm is initialized with a large step value and when the intersection is detected ($Z_{\mathrm{R}} \leq Z_{\mathrm{DEM}}$), the step size is reduced with a step divider, t_{div}. This way, the number of iterations is reduced without sacrificing the IRT's accuracy. This strategy can be repeated until the step size becomes smaller than a pre-defined threshold, t_{th}.

3.3.2.C Bilinear Interpolation

A gridded DEM such as the *EU-DEM v1.1* is a map sampled at a constant spacial resolution. Each pixel represents a planar surface (area equal to the squared resolution) with constant elevation, when in fact this elevation represents only the center of the cell. This is known as the rigid pixel paradigm. On the other hand there is the surface adjusted paradigm, where the sampled points are interpolated to approximate the real world topography. These paradigms are represented in Figure 3.4.



(a) Rigid pixel paradigm (b) Surface adjusted paradigm

Figure 3.4: Rigid pixel and surface adjusted paradigms (adapted from [17])

As was reviewed in Subsection 2.1.5, for maps with finer resolutions, the bilinear interpolation provides a satisfactory approximation of the real surface without compromising real-time application of the algorithm. Given four known points, the bilinear interpolation consists in performing the linear interpolation across one of the dimensions and with the resulting points perform it again across the other dimension (Figure 3.5).

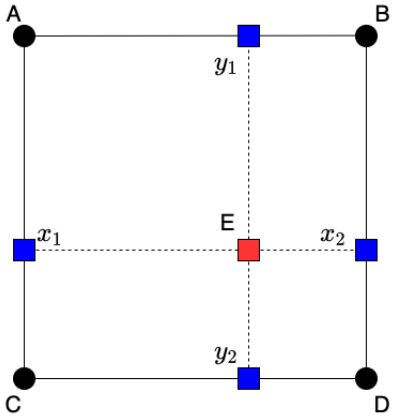


Figure 3.5: Bilinear interpolation

3.3.3 Complete Iterative Ray-Tracing Algorithm

The following algorithm describes the complete IRT algorithm, with the aforementioned variants included:

Algorithm 2: Complete Iterative Ray-Tracing

Input : Telemetry from GPS and IMU, camera IP, target pixels, DEM
Output: Target coordinates in the inertial coordinate frame $\mathbf{t}_{\mathrm{I}} = [x\ y\ z]$

Define ray origin $\mathbf{R}_0$;
Define ray direction $\mathbf{R}_d$;
Define step size t;
Define step size threshold t_{th};
Define step size divider t_{div};
if $Z_{\boldsymbol{R}_0} \geq Z_{\max}$ **then**
    Update ray origin with (3.34);
end
while *No Intersection* **do**
    Extend ray with (3.32);
    Interpolate ray;
    if $Z_{\mathrm{R}} \leq Z_{\mathrm{DEM}}$ *AND* $t \geq t_{\mathrm{th}}$ **then**
        Update step t by diving it by step divider t_{div};
    else if $Z_{\mathrm{R}} \leq Z_{\mathrm{DEM}}$ *AND* $t < t_{\mathrm{th}}$ **then**
        Intersection;
    else
        Augment t with current step value;
    end
end

3.4 Position and Uncertainty Characterization with the Unscented Transform

Three sources of uncertainty were be taken into account: the vehicle GPS, IMU and gimbal. The GPS contributed with three degrees of uncertainty related to the position of the vehicle in the vehicle frame, σ_x, σ_y and σ_z. The vehicle IMU contributed with another three degrees of uncertainty related to the orientation angles of the vehicle with respect to the vehicle frame, roll σ_ϕ, pitch σ_θ, and yaw σ_ψ. Finally, the gimbal contributed with two degrees of uncertainty related to the elevation and azimuth angles that establish the orientation of the gimbal with respect to the body frame, $\sigma_{\alpha_{el}}$ and $\sigma_{\alpha_{az}}$. This makes a total of $n = 8$ degrees of uncertainty. The standard deviations presented in Table 3.1 were assumed independent and the covariance matrix was defined as in (3.35).

Device	Standard Deviation σ_i	Value
GPS	σ_x	10m
	σ_y	10m
	σ_z	10m
IMU	σ_ϕ	1°
	σ_θ	1°
	σ_ψ	3°
Gimbal	$\sigma_{\alpha_{el}}$	1°
	$\sigma_{\alpha_{az}}$	1°

Table 3.1: GPS, IMU and Gimbal standard deviations

$$\Sigma = \text{diag}(\sigma_x^2, \sigma_y^2, \sigma_z^2, \sigma_\phi^2, \sigma_\theta^2, \sigma_\psi^2, \sigma_{\alpha_{el}}^2, \sigma_{\alpha_{az}}^2) \tag{3.35}$$

Parameter	Value
α	$\frac{1}{\sqrt{8}}$
κ	0
β	2

Table 3.2: UT parameters

The UT parameters were set according to Table 3.2. Since we approximate the distribution as a Gaussian, β = 2 is the optimal choice to minimize higher order information from the Taylor Series expansion. As for α and κ, these values were chosen so as to have the sigma points equal to the standard deviations of the equipment.

The target position and uncertainty are calculated by propagating the sigma points with the complete IRT described in Algorithm 2 and applying (2.12) and (2.13):

Algorithm 3: Unscented Transform with IRT

Output: Geodetic target coordinates $\mathbf{t}$ = [ϕ λ h] and uncertainty σ_{xyz} = [σ_x σ_y σ_z]

Generate $2n + 1$ sigma points;

for $i \leftarrow 0$ **to** $2n$ **do**

  Calculate the intersection for each sigma point, $y_i = \mathrm{IRT}(\chi_i)$;

end

Recover mean $\mathbf{t}$ and covariance $\mathbf{P}$;

Convert mean to geodetic coordinates and uncertainty $\sigma_{xyz} = \sqrt{\mathrm{diag}(\mathbf{P})}$;

3.5 Bearings-Range Measurement Model

Considering measurement errors in the GPS and IMU, this section details the proposed vision-based target localization using bearings and range measurements. Regarding previous works [34, 40], a new filter measurement model was developed that takes advantage of the available range information between the vehicle and the target. Taking into account possible linearization errors induced by the EKF, a performance comparison is done with a CKF.

The bearings localization problem is based on the extraction of the azimuth β, and elevation ϕ, angles from the pointing vector, as shown in Figure 3.6. Since the proposed georeferencing algorithm calculates the 3D coordinates of the target, it enables the estimation of the distance r between the vehicle and the target. For this methodology to work, the target must remain in the line-of-sight of the camera.

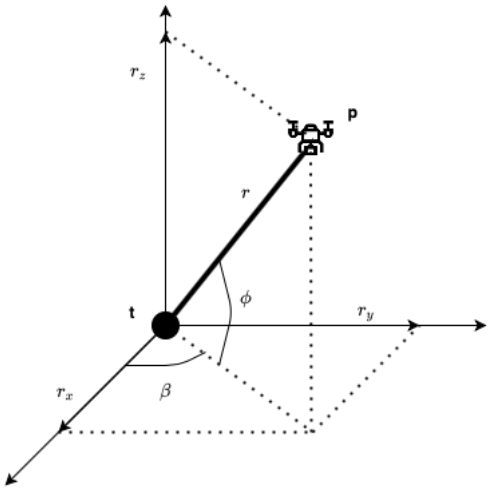


Figure 3.6: Azimuth (β), elevation (ϕ) and range (r) between vehicle ($\mathbf{p}$) and target ($\mathbf{t}$)

The measurements model for the non-linear Gaussian system described in (2.14b) is given by:

$$\mathbf{h}(\mathbf{x}_k) = \begin{bmatrix} \beta \\ \phi \\ r \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{p_y - t_y}{p_x - t_x}\right) \\ \tan^{-1}\left(\frac{p_z - t_z}{\sqrt{(p_x - t_x)^2 + (p_y - t_y)^2}}\right) \\ \sqrt{(p_x - t_x)^2 + (p_y - t_y)^2 + (p_z - t_z)^2} \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{r_y}{r_x}\right) \\ \tan^{-1}\left(\frac{r_z}{\sqrt{r_x^2 + r_y^2}}\right) \\ \sqrt{r_x^2 + r_y^2 + r_z^2} \end{bmatrix} \tag{3.36}$$

where $\mathbf{r}_k = [r_x\ r_y\ r_z]_k^T = [p_x - t_x,\ p_y - t_y,\ p_z - t_z]_k^T$ is the relative vector between the vehicle and the target for the k^{th} measurement, $\mathbf{p}_k = [p_x\ p_y\ p_z]_k^T$ is the position of the vehicle and $\mathbf{x}_k = \mathbf{t}_k = [t_x\ t_y\ t_z]_k^T$ is the position of the target and the state to be estimated.

Assuming that a fire front is stationary when comparing images taken within a small time interval, the target dynamics model used in (2.14a) is given by:

$$\mathbf{f}(\mathbf{x}_k) = \Phi_{k+1|k}\mathbf{x}_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_k \tag{3.37}$$

and the covariance of the system noise w_k is given by

$$\mathbf{Q}_k = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{3.38}$$

For the EKF, the Jacobian of the measurement function with respect to the state is

$$\mathbf{H}_k = \begin{bmatrix} \frac{r_y}{r_x^2 + r_y^2} & -\frac{r_x}{r_x^2 + r_y^2} & 0 \\ \frac{r_x r_z}{(r_x^2 + r_y^2 + r_z^2)\sqrt{r_x^2 + r_y^2}} & \frac{r_y r_z}{(r_x^2 + r_y^2 + r_z^2)\sqrt{r_x^2 + r_y^2}} & -\frac{\sqrt{r_x^2 + r_y^2}}{(r_x^2 + r_y^2 + r_z^2)} \\ -\frac{r_x}{\sqrt{r_x^2 + r_y^2 + r_z^2}} & -\frac{r_y}{\sqrt{r_x^2 + r_y^2 + r_z^2}} & -\frac{r_z}{\sqrt{r_x^2 + r_y^2 + r_z^2}} \end{bmatrix}_k. \tag{3.39}$$

The sensor noise mentioned in Section 3.4 is used to model the noise covariance matrix, tuned to the following values

$$\mathbf{R}_k = \begin{bmatrix} \sigma_{\alpha_{az}}^2 & 0 & 0 \\ 0 & \sigma_{\alpha_{el}}^2 & 0 \\ 0 & 0 & \sigma_r^2 \end{bmatrix} = \begin{bmatrix} 1^2 & 0 & 0 \\ 0 & 1^2 & 0 \\ 0 & 0 & 10^2 \end{bmatrix}, \tag{3.40}$$

where $\sigma_{\alpha_{az}}$ and $\sigma_{\alpha_{el}}$ are the gimbal's azimuth and elevation uncertainties and σ_r is the range uncertainty. The IRT and UT results of the first observation initialize the filters' state and covariance, $\hat{\mathbf{x}}_\mathbf{0}$ and $\hat{\mathbf{P}}_0$.

3.6 Metrics

The metrics presented in this Section will be used throughout the following chapters. The position error, e_p defined in (3.41) is used to determine the distance between the target position, t, and its estimates, $\hat{t}$,

$$e_p = [e_{p_x}\ e_{p_y}\ e_{p_z}]^T = [t_x\ t_y\ t_z]^T - [\hat{t}_x\ \hat{t}_y\ \hat{t}_z]^T. \tag{3.41}$$

To characterize the accuracy of the algorithm, the average position error and RMSE are defined, respectively,

$$\mu_{e_p} = \frac{\sum_{i=1}^{N} ||e_{p_i}||}{N} = \frac{\sum_{i=1}^{N} \sqrt{e_{p_{x_i}}^2 + e_{p_{y_i}}^2 + e_{p_{z_i}}^2}}{N}, \tag{3.42}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} ||e_{p_i}||^2}{N}} = \sqrt{\frac{\sum_{i=1}^{N} e_{p_{x_i}}^2 + e_{p_{y_i}}^2 + e_{p_{z_i}}^2}{N}}. \tag{3.43}$$

While the former provides the average distance between the estimates and the target, the latter characterizes their dispersion.

Finally, the uncertainty is defined as the square root of the sum of the diagonal of matrix $\mathbf{P}$,

$$\mathbf{P} = \begin{bmatrix} \sigma_{xx}^2 & \sigma_{xy}^2 & \sigma_{xz}^2 \\ \sigma_{yx}^2 & \sigma_{yy}^2 & \sigma_{yx}^2 \\ \sigma_{zx}^2 & \sigma_{zy}^2 & \sigma_{zz}^2 \end{bmatrix}, \tag{3.44}$$

$$\sigma_{xyz} = \sqrt{\mathrm{diag}(\mathbf{P})} = [\sigma_{xx},\ \sigma_{yy},\ \sigma_{zz}]^T. \tag{3.45}$$

The average position uncertainty is defined as

$$\mu_{\sigma_{xyz}} = \frac{\sum_{i=1}^{N} \sqrt{\sigma_{xx_i}^2 + \sigma_{yy_i}^2 + \sigma_{zz_i}^2}}{N}. \tag{3.46}$$

4

Simulations

Contents

The following chapter describes the simulations designed to validate the IRT, the uncertainty characterization using the UT and the filtering algorithm.

Section 4.1 describes the datasets used to perform the simulations, their terrain type and the software used.

Section 4.2 analyzes the contribution of each IRT upgrade proposed in the previous chapter considering a fixed system configuration.

In Section 4.3 the UT behaviour is validated with eight different simulation scenarios.

Finally, Section 4.4 describes two simulations planned to validate the Bearings-Range filtering algorithm. The performances of the EKF and CKF are compared with the standalone IRT and conclusions are drawn on which Kalman Filter is better for this measurement model.

4.1 Simulation Environment

In this thesis, two terrain types were considered to perform the simulations, rough and flat, following the example of previous works [40] on this topic.

4.1.1 Rough Terrain Dataset

The chosen rough terrain is located near Coentral, Leiria, where a fire simulacrum planned by the FIREFRONT project took place. There, an airplane was used to acquire RGB and thermal imagery. Some of the RGB images are shown in Figure 4.1. Figure 4.2 shows the chosen rough terrain area and the respective DEM is presented in Figure 4.3. The main characteristics of the terrain are summarized in Table 4.1.

Characteristics	
Maximum Height [m]	1088.00
Minimum Height [m]	850.57
WGS84/EPSG:4326 Coordinates [°]	N[40.060, 40.067] E[-8.166, -8.159]

Table 4.1: Coentral DEM characteristics

Gazebo was used to acquire telemetry and imagery data. Gazebo is an open-source robotic simulator useful for robot simulation, or to simulate a sensor or an actuator. In the context of this thesis, it was used to acquire position and orientation of a vehicle and imagery of a target using a camera attached to said vehicle. A linear trajectory parallel to the elevation was simulated, with constant yaw equal to 48°, roll and pitch equal to 0°, and a variable azimuth and elevation angles that kept the target in sight of the camera. This way, it was possible to manually identify the pixel corresponding to the target on each image. Because there is no fire model available in the Gazebo library, a checkerboard was used to represent the target. Figure 4.4 shows the target surrounded by trees, to resemble a forest environment. The camera parameters presented in Table 4.2 were used throughout all the Gazebo simulations, with a resolution of 640x480. This camera was considered ideal, with no radial or tangential distortions.

(a)

(b)

(c)

Figure 4.1: Imagery from the fire simulacrum near Coentral, Leiria

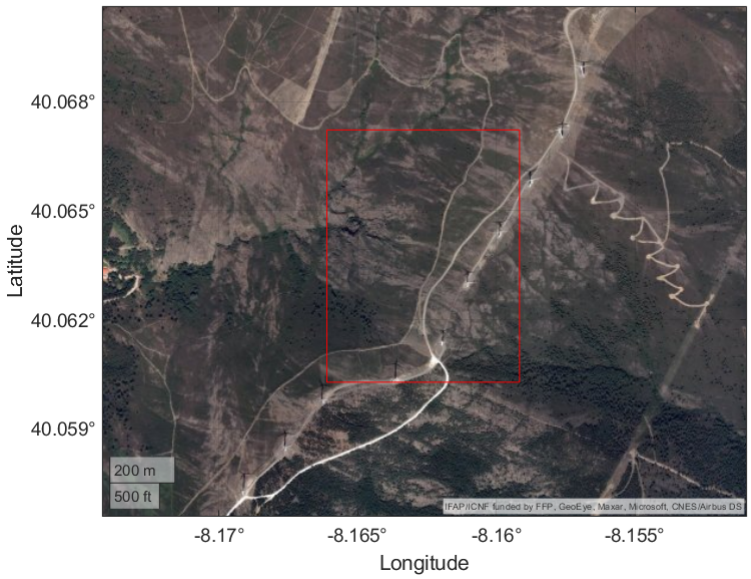


Figure 4.2: Testing area near Coentral, Leiria

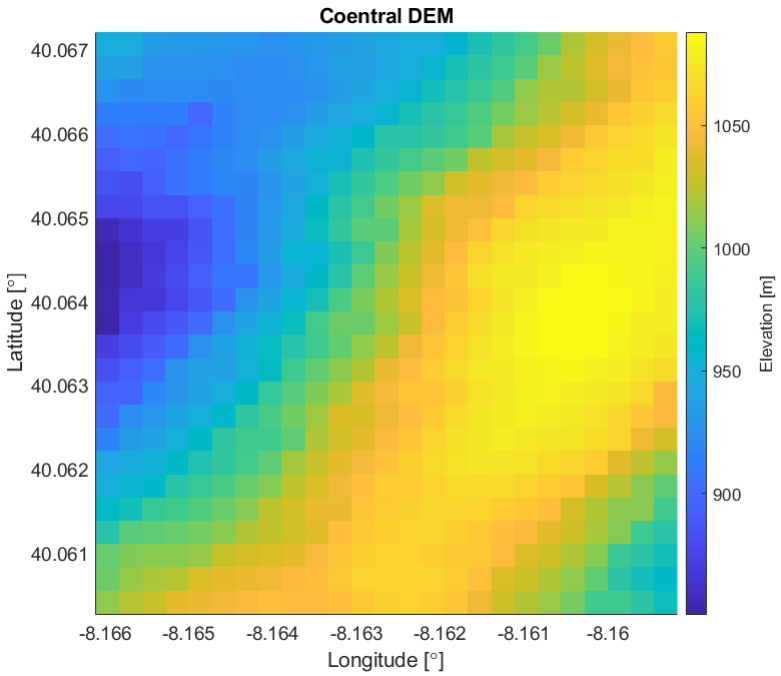


Figure 4.3: DEM of testing area near Coentral, Leiria

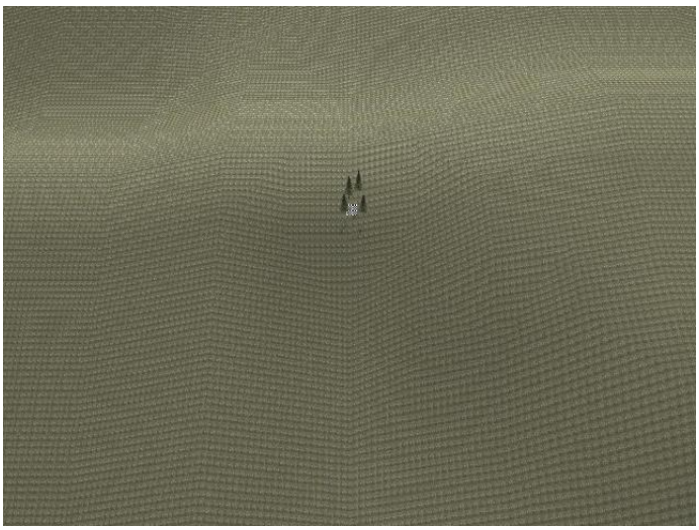


Figure 4.4: Gazebo target setup for image and telemetry acquisition

Parameter	Value
f_x	480.0
f_y	480.0
c_x	319.5
c_y	239.5

Table 4.2: Camera parameters used in the Gazebo environment

4.1.2 Flat Terrain Dataset

Regarding the flat terrain dataset, a small area near Porto de Mós, Leiria, represented in Figure 4.5, was chosen. Its DEM is presented in Figure 4.6 and the main characteristics are summarized in Table 4.3. The same area was later used to perform a real experiment, to be presented in the following chapter. For this terrain type, MATLAB was

Figure 4.5: Testing area near Porto de Mós, Leiria

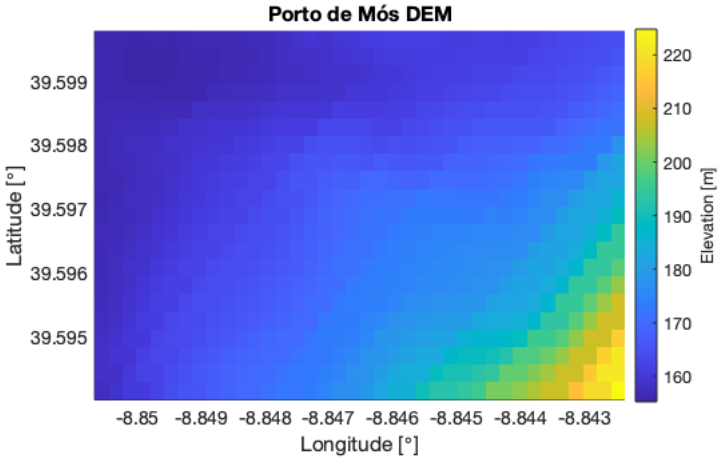


Figure 4.6: Testing area near Porto de Mós, Leiria

used to generate ground truth telemetry data. As in the previous dataset, a linear trajectory was simulated. This time, the vehicle moved East, therefore with the yaw equal to 90°, while the roll and pitch were set to 0°. It was assumed that the target was always centred in the images, so no camera model was needed and the azimuth and

Characteristics	
Maximum Height [m]	224.82
Minimum Height [m]	155.25
WGS84/EPSG:4326 Coordinates [°]	N[39.5940, 39.5998] E[-8.8507, -8.8424]

Table 4.3: Porto de Mós DEM characteristics

elevation angles were calculated based on the trigonometric relation between the aerial vehicle and the target.

4.2 Iterative Ray-Tracing Validation

To validate the Algorithm 2 described in 3.3.3, an experiment was conducted using data from the rough terrain dataset. Each algorithm variant was implemented individually and its performance was tested against the basic IRT described in Algorithm 1. The system's position and attitude were kept constant, as was the target's position.

Pixel		Vehicle Position [m]			Attitude [rad]					Target Position [m]		
u	v	x	y	z	ϕ	θ	ψ	α_{el}	α_{az}	x	y	z
316	249	42.5	582.5	2350.0	0	0	0.75	-1.25	0.75	350.3	312.0	1041.7

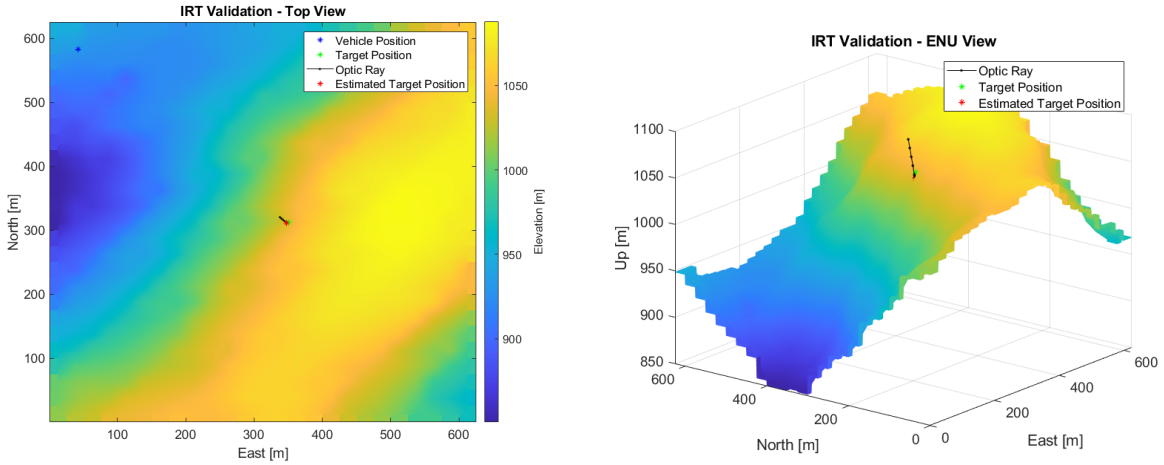
Table 4.4: System and target setup for IRT validation

Three different variables were taken into account when analysing the IRT variant performances: the number of iterations, the processing time, and the position error. The chosen step size t was 10 meters, the step divider t_{div} was 10 and the step threshold t_{th} was 1 meter.

Method	Iterations	Processing Time [ms]	Position Error [m]
Basic IRT	139	4.610	19.405
Dynamic Step	143	7.727	13.570
Maximum Height Iteration	7	1.890	20.839
Bilinear Interpolation	138	5.236	9.792
Combined	10	1.934	5.869

Table 4.5: Performance comparison of the IRT variants

The results presented in Table 4.5 are in agreement with what was expected. The dynamic step increases the accuracy but also increases the number of iterations and, consequentially, the processing time. The maximum height iteration decreases significantly the number of iterations since the maximum height of the loaded map was 1088 meters and the vehicle height was 2350 meters. The bilinear interpolation increases the accuracy of the IRT but also increases the processing time. In conclusion, the combination of all the variants not only achieves the best result in terms of position error but also minimizes their individual disadvantages. Figure 4.7 allows the visualization of the results of the complete IRT algorithm.



(a) Top view

(b) Perspective view

Figure 4.7: IRT combining all algorithm variants

4.3 Characterization of IRT Uncertainty

To validate the UT applied to the characterization of uncertainty of the IRT algorithm, data from the rough terrain dataset was used. A total of 8 different scenarios were simulated, whose parameters are presented in Table 4.6. The main purpose of this set of simulations was to verify the evolution of the uncertainty as a function of the distance to the target.

Test	Pixel		Vehicle Position [m]			Attitude [rad]					Target Position [m]		
	u	v	x	y	z	ϕ	θ	ψ	α_{el}	α_{az}	x	y	z
1	321	241	350.3	312.0	1100.0	0	0	0	$-\frac{\pi}{2}$	0	350.3	312.0	1041.7
2	323	189	272.5	312.5	1100.0	0	0	0	-0.75	0	350.3	312.0	1041.7
3	322	259	192.5	312.5	1200.0	0	0	0	-0.75	0	350.3	312.0	1041.7
4	212	245	192.5	387.5	1200.0	0	0	0.75	-0.75	0.75	350.3	312.0	1041.7
5	322	290	42.5	312.5	1650.0	0	0	0	-1	0	350.3	312.0	1041.7
6	313	230	42.5	582.5	1650.0	0	0	0.75	-1	0.75	350.3	312.0	1041.7
7	320	278	42.5	312.5	2350.0	0	0	0	-1.25	0	350.3	312.0	1041.7
8	316	249	42.5	582.5	2350.0	0	0	0.75	-1.25	0.75	350.3	312.0	1041.7

Table 4.6: System setup for IRT uncertainty characterization with the UT

The results presented in Table 4.7 show that as the distance to the target increases, the larger are the uncertainties due to the GPS, IMU and gimbal pose. Furthermore, the GPS errors are the dominant source of uncertainty for distances closer to the target, while the IMU and gimbal errors become the dominant source of uncertainty for greater distances. As an example, the uncertainty region estimated with the UT for test 8 is plotted in Figure 4.8 using the uncertainties presented in Table 3.1.

Test	Distance to Target [m]	σ_x [m]	σ_y [m]	σ_z [m]	Position Error [m]
1	58.3	10.14	10.05	10.24	3.710
2	97.25	10.01	10.96	7.50	2.741
3	223.54	9.95	13.55	7.52	2.657
4	235.94	10.20	14.07	7.58	2.354
5	681.76	17.36	22.30	16.50	4.095
6	733.46	21.42	26.15	18.88	3.592
7	1346.35	31.37	30.16	26.65	2.917
8	1370.98	31.97	34.84	34.28	5.885

Table 4.7: UT validation results

Device	Standard Deviation σ_i	Value
GPS	σ_x	10m
	σ_y	10m
	σ_z	10m
IMU	σ_ϕ	1°
	σ_θ	1°
	σ_ψ	3°
Gimbal	$\sigma_{\alpha_{el}}$	1°
	$\sigma_{\alpha_{az}}$	1°

Table 3.1: GPS, IMU and Gimbal standard deviations (repeated from page 32)

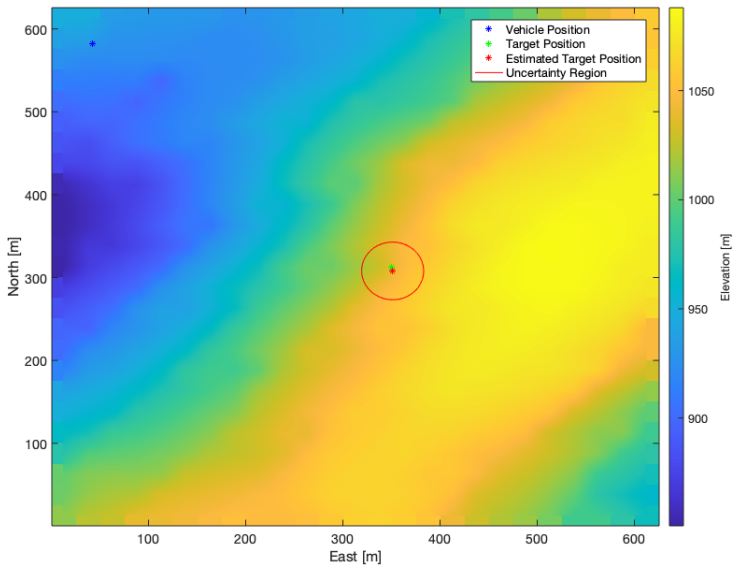


Figure 4.8: Test 8 with uncertainty region

4.4 Bearings-Range Model Validation

This section details two simulations performed with the following purposes: to compare the Bearings-Range with the Bearings-Only measurement models; to conclude on whether the CKF is, in fact, a better alternative, for the Bearings-Range measurement model, than the EKF. In addition, they served as a verification of the expected behaviour of the georeferencing algorithm in rough and flat terrains. In both simulations a linear trajectory was assumed, with the vehicle travelling at a constant speed of 250 km/h and acquiring telemetry and imagery with a frequency of 1 Hz.

The ground truth telemetry data was perturbed using the following model:

$$\hat{\mathbf{x}}_i(t) = \mathbf{x}_i(t) + \mathbf{w}(t), \quad \mathbf{w} \sim \mathcal{N}(0, \sigma_{x_i}), \tag{4.1}$$

where $\hat{\mathbf{x}}_i$ is the noisy variable, $\mathbf{x}_i$ is the ground truth variable, $\mathbf{w}$ is the zero-mean noise with σ_{x_i} standard deviation. The assumed values for σ_{x_i} are the same as the ones used for the UT, presented in Table 3.1. For each simulation, 100 runs were performed with independent noise sequences.

On a first stage, the IRT was applied to each measurement. The resulting estimated positions were then filtered with the Bearings-Range and Bearings-Only models. The IRT results are an average of all measurements, while the filtering results are an average of their final correction.

4.4.1 Rough Terrain Simulation

A total of 25 measurements were collected at a constant height of 1650 meters. Figure 4.9 presents the measurements with undisturbed vehicle position and system attitude. Figure 4.10 shows the resulting measurements following the model presented in (4.1) to disturb the system. The average distance to the target was 880 meters.

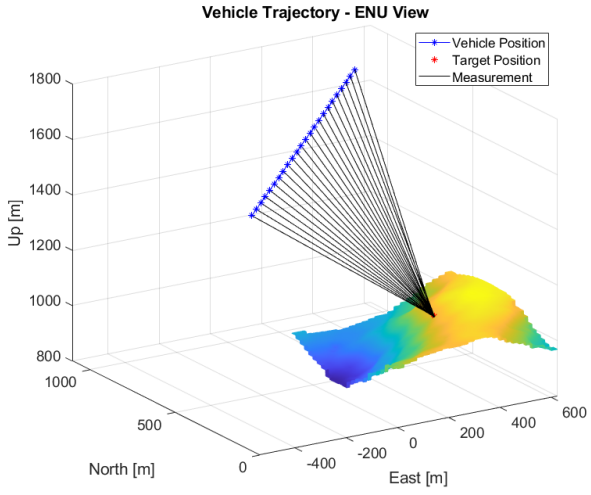


Figure 4.9: Measurements with ground truth position and attitude on the Coentral simulation

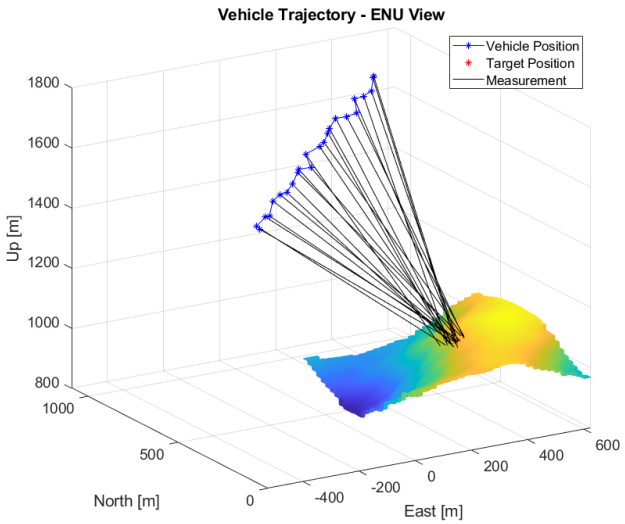


Figure 4.10: Measurements with noise-induced position and attitude on the Coentral simulation

4.4.2 Flat Terrain Simulation

A total of 21 measurements were collected at a constant height of 950 meters. Figure 4.11 presents the measurements with ground truth data. Figure 4.12 shows the resulting measurements with noisy data. The average distance to the target was 985 meters.

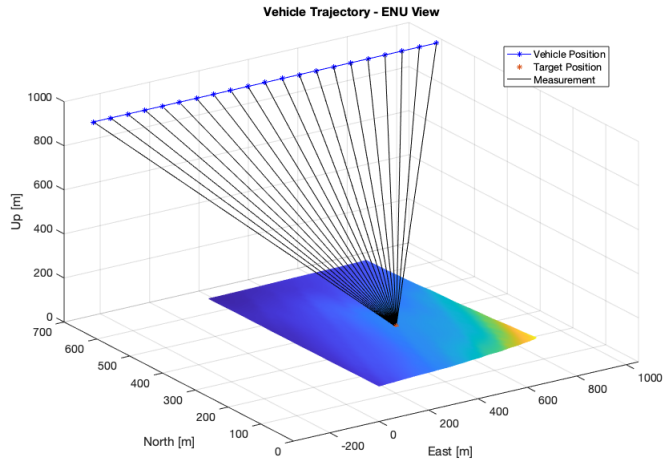


Figure 4.11: Measurements with ground truth position and attitude on the Porto de Mós simulation

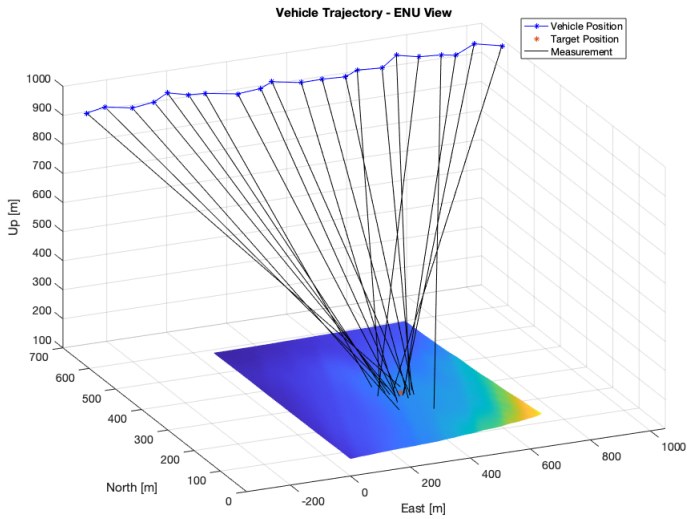


Figure 4.12: Measurements with noise-induced position and attitude on the Porto de Mós simulation

4.4.3 Results

This subsection presents the results of the rough and flat terrain simulations for the Bearings-Only and Bearings-Range measurement models.

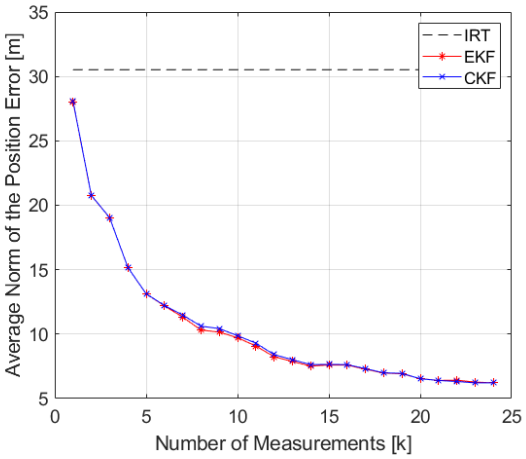
The averaged norm of the position error and uncertainty for the rough case are plotted in Figure 4.13 as a function of the number of measurements. Table 4.9 summarizes the results of the IRT, IRT+Bearings-Range Extended Kalman Filter (BR-EKF), IRT+Bearings-Only Extended Kalman Filter (BO-EKF), IRT+Bearings-Range Cubature Kalman Filter (BR-CKF) and IRT+Bearings-Only Cubature Kalman Filter (BO-CKF). Table 4.11 compares the average processing time per iteration of the EKF and CKF.

Method	Iterations	Norm of Position Error [m]	RMSE [m]	Norm of Position Uncertainty [m]
IRT		30.527	30.743	35.965
IRT+BR-EKF	24	**6.221**	**11.726**	3.837
IRT+BO-EKF	24	8.225	13.071	5.137
IRT+BR-CKF	24	6.222	11.785	**3.782**
IRT+BO-CKF	24	8.193	13.136	5.054

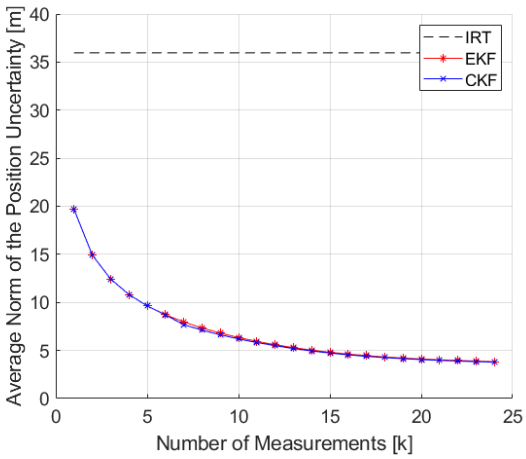
Table 4.9: IRT, IRT+BR-EKF, IRT+BO-EKF, IRT+BR-CKF and IRT+BO-CKF results for the rough terrain scenario

For the flat terrain simulation, Figure 4.14 presents the evolution of the averaged position error and uncertainty norm as a function of the number of measurements. The results are summarized in Table 4.10.

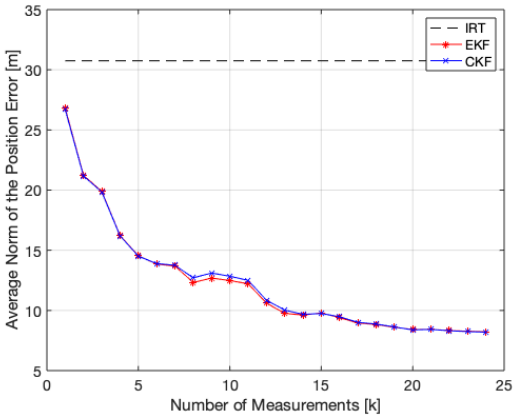
As can be seen in Figures 4.13 and 4.14, for both terrain types, the Bearings-Range model converges faster than the Bearings-Only. This is evident for $k = 10$, for example. This faster convergence is also evident on the results presented in Tables 4.9 and 4.10, with the BR-EKF and BR-CKF achieving lower position errors, RMSE's



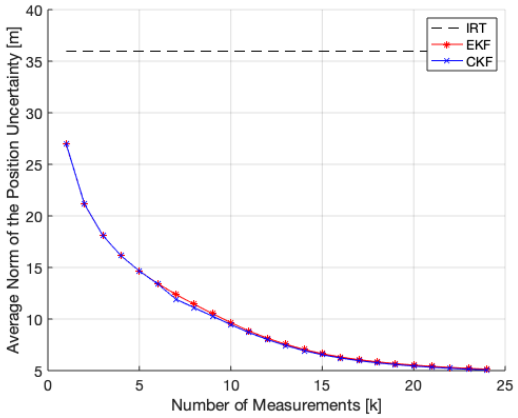
(a) BR-EKF and BR-CKF average norm of the position error



(b) BR-EKF and BR-CKF average norm of the position uncertainty



(c) BO-EKF and BO-CKF average norm of the position error



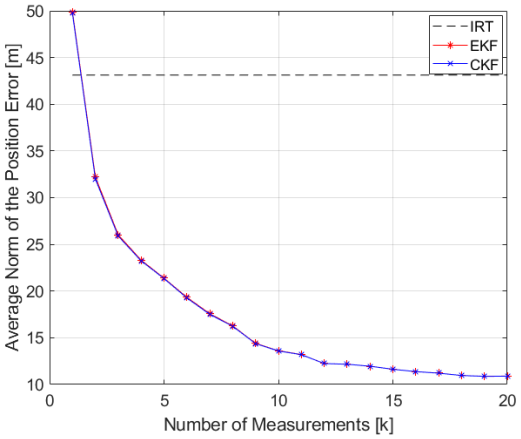
(d) BO-EKF and BO-CKF average norm of the position uncertainty

Figure 4.13: EKF and CKF filtering results for the rough terrain simulation

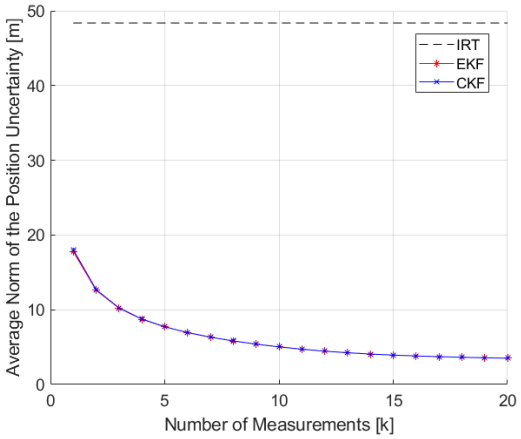
Method	Iterations	Norm of Position Error [m]	RMSE [m]	Norm of Position Uncertainty [m]
IRT		43.121	43.405	48.373
IRT+BR-EKF	20	11.020	19.910	**3.898**
IRT+BO-EKF	20	11.565	21.578	3.909
IRT+BR-CKF	20	**11.000**	**19.842**	3.899
IRT+BO-CKF	20	11.575	21.573	3.911

Table 4.10: IRT, IRT+BR-EKF, IRT+BO-EKF, IRT+BR-CKF and IRT+BO-CKF results for the flat terrain scenario

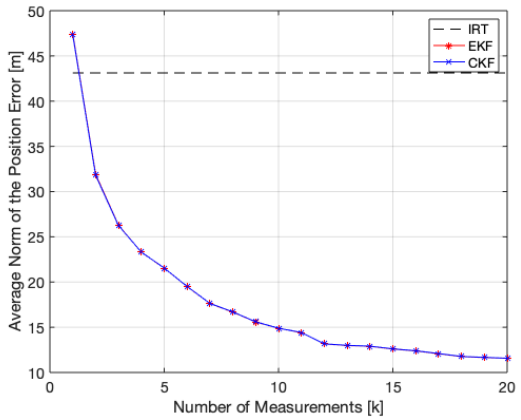
and uncertainties when compared to their Bearings-Only counterparts. Furthermore, the EKF achieves a faster processing time than the CKF, a result that is in agreement with previous works [16] and is due to the differences in their algorithms.



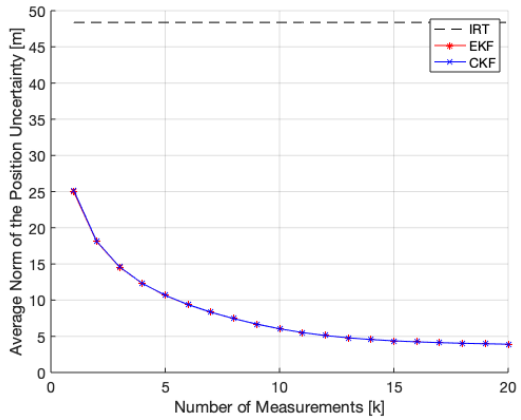
(a) BR-EKF and BR-CKF average norm of the position error



(b) BR-EKF and BR-CKF average norm of the position uncertainty



(c) BO-EKF and BO-CKF average norm of the position error



(d) BO-EKF and BO-CKF average norm of the position uncertainty

Figure 4.14: BR-EKF and BR-CKF filtering results for the flat terrain simulation

Kalman Filter	Average Processing Time [ms]
Extended	**0.795**
Cubature	0.83

Table 4.11: EKF and CKF processing time comparison

4.4.4 Discussion of Results

Comparing Tables 4.9 and 4.10, the rough terrain achieves better results in terms of the standalone IRT. This can be explained by two factors: distance to the target and incidence angle. While the average distance to the target in Porto de Mós was 985 meters, in the Coentral simulation the average distance was 880. As it had already been concluded from the UT validation, the greater the distance, the greater the position error and uncertainty. The

other important factor is the incidence angle. In the Coentral scenario, the optic-ray is more perpendicular to the surface than in Porto de Mós. Considering the study in Appendix A, the more perpendicular to the surface, the less amplified is the uncertainty.

These simulations demonstrate the advantage of including the range information in the filtering algorithm. Both in the rough and flat scenarios, the Bearings-Range measurement model achieves lower position errors, position uncertainties and RMSE's for the same number of measurements, therefore is more accurate than the Bearings-Only measurement model. These results also show a clear improvement on the accuracy of the estimated target position when applying the filtering algorithm. The IRT RMSE is reduced by 61.86% and 54.12% for the rough and flat terrains, respectively, both for the BR-EKF and BR-CKF. Since both filters achieve the same final result with an identical progression, both in the position error and uncertainty, there is no clear advantage in using the BR-CKF over the BR-EKF on this measurement model. In addition, the EKF shows a slightly faster processing time per iteration, making it more appropriate for real-time applications, as is the case of the FIREFRONT project.

Regarding the state-of-the-art, results are in agreement with the ones presented in [40], which is the work that mostly resembles this work. A comparison is done on Table 4.12. There are various key factors that contribute to

Terrain Type	Xu et al. [40]	This work
Rough	13.8	**11.7**
Flat	**10.8**	19.8

Table 4.12: Comparison with state-of-the-art RMSE [m]

the difference in the results. In their work, the UAV performed a circular trajectory around the target, at a height of 500 meters, acquiring at least 60 measurements for the flat terrain and 120 for the rough terrain. In this work, the aerial vehicle followed a linear trajectory, at height of 950 and 1650 meters and acquired no more than 21 and 25 measurements, for the flat and rough terrains, respectively. Furthermore, the equipment measurement errors in that work were more accurate than in this work. Under the same circumstances, as was the case in the simulations presented in this chapter, it is demonstrated that the Bearings-Range is a clear improvement over the Bearings-Only measurement model.

5

Experimental Procedures

Contents

This chapter details the experimental procedures conducted to verify the validity of the proposed methodology. It is divided in two sections, Section 5.1 which details the different experiments, their objectives and the data used, and Section 5.2, that summarizes the results obtained in each experiment and concludes on them.

5.1 Experiments

Three experiments compose this section and are presented in the following manner: Subsection 5.1.1 details the procedure where a mobile phone was used to acquire telemetry and imagery of a target along a pedestrian path, near Porto de Mós, Leiria.

Subsection 5.1.2 describes the georeferencing algorithm applied to a footage recorded by a Portuguese Air Force UAV near Chaves, Vila Real.

Finally, subsection 5.1.3 describes another experiment using footage from a UAVision's UAV recorded near Pombal, Leiria. A minimization problem is proposed to calculate the camera's intrinsic parameters and the results are used on the georeferencing algorithm.

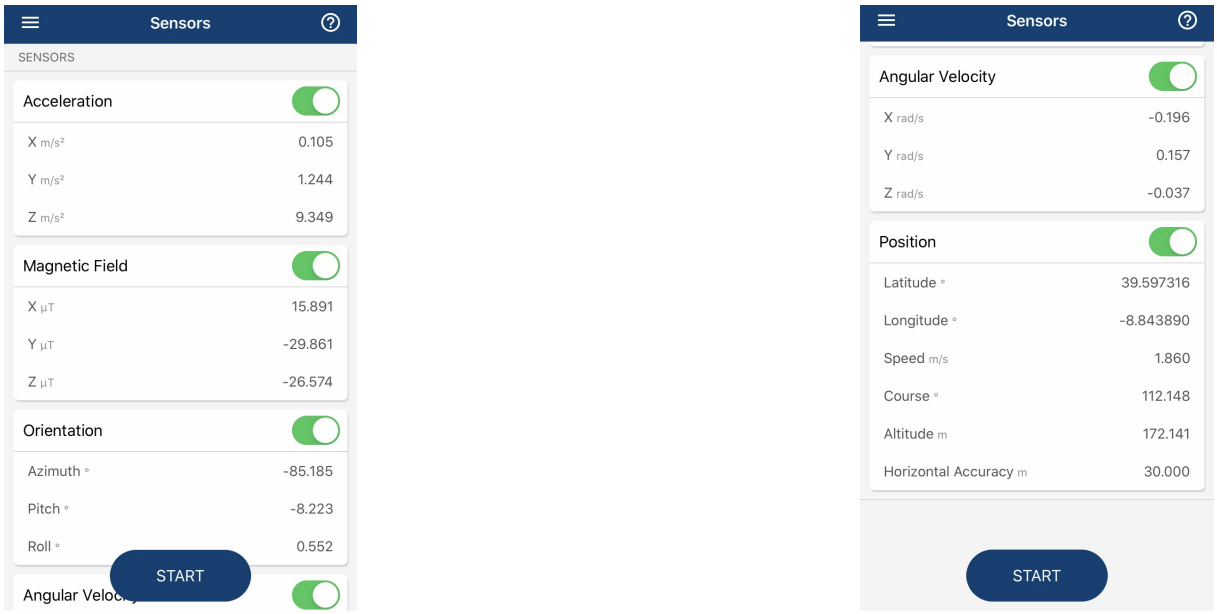
5.1.1 Mobile Phone

The purpose of this procedure was to simulate the flight of an aerial vehicle using a mobile phone. Therefore, 14 images of a target were captured and also its IMU was used to acquire telemetry data.

To calibrate the camera, a set of 20 images of a squared checkerboard pattern was captured and given as input to the MATLAB Camera Calibrator from the Computer Vision Toolbox. In order to access the mobile phone's accelerometer, gyroscope, magnetometer and GPS, the MATLAB Mobile App was used. The App, whose interface is presented in Figure 5.1, allows the logging and upload of raw sensor data with a configurable sample rate, from 0.5 to 100 Hz. It also provides the roll, pitch and azimuth angles calculated by the mobile phone's internal algorithm with respect to the coordinate frame presented in Figure 5.2.. The GPS sample rate is not configurable and is fixed at 1 Hz.

To perform the integration of the raw data from the accelerometer, gyroscope and magnetometer, MATLAB Sensor Fusion and Tracking Toolbox was used. More specifically, *ahrsfilter*, which initializes an indirect Kalman Filter described in [7]. This filter has nothing to do with the BR-EKF. Instead of tracking the actual state vector, the indirect Kalman Filter tracks the errors for the orientation, gyroscope bias, linear acceleration and magnetic disturbance. This filter requires as input the sensors noise covariances in addition to the sensor data. The mobile phone's manufacturer does not disclose the sensor models used, therefore the covariances could not be obtained by looking at the sensors' datasheets. As an alternative, an Allan Variance analysis was employed on 8 hours of static accelerometer and gyroscope. The plots are presented in Figure 5.3 and the results are summarized in Table 5.1.

Different small experiments at 10 Hz frequency were conducted to validate the result of the MATLAB implemented filter by comparing it with the orientation outputted by the mobile phone's internal algorithm. One of these



(a) (b)

Figure 5.1: MATLAB Mobile sensor interface

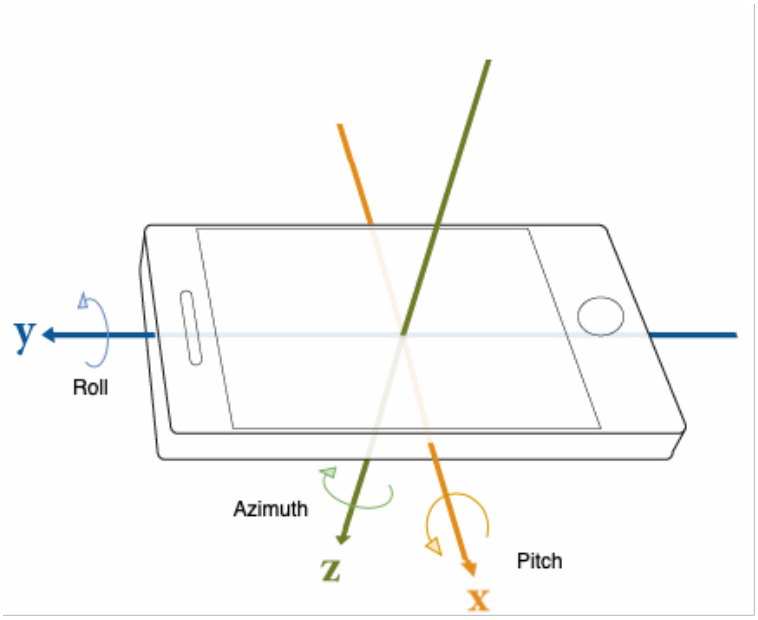
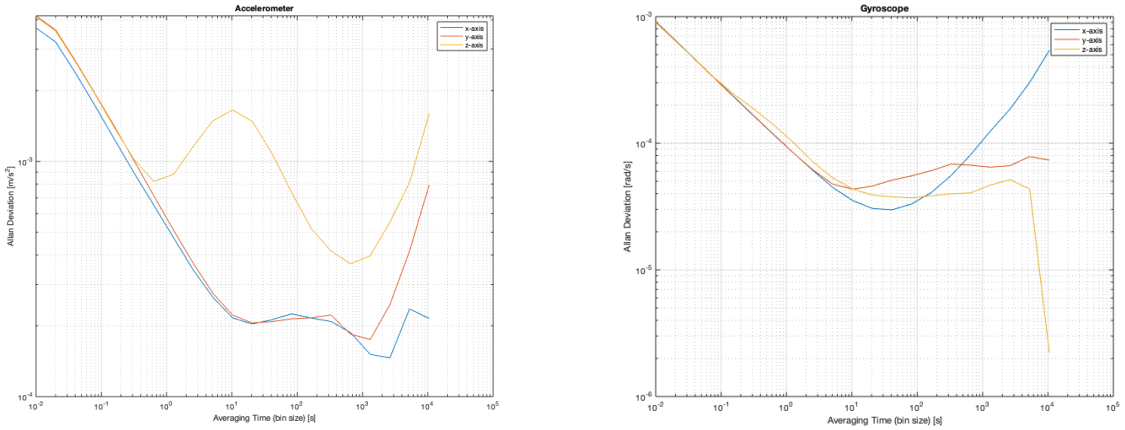


Figure 5.2: iPhone coordinate frame

experiments is presented in Figure 5.4. While the roll and pitch were practically identical in all experiments, an unexpected behaviour was detected in the iPhone outputted yaw, which had a constant offset when compared to the yaw angle estimated by the Kalman Filter. It was latter found that the MATLAB Mobile app always initializes this angle with -90°, and does not calculate it with respect to the magnetic north. Correcting the offset, the curves



(a) Accelerometer

(b) Gyroscope

Figure 5.3: Allan Deviation Plots

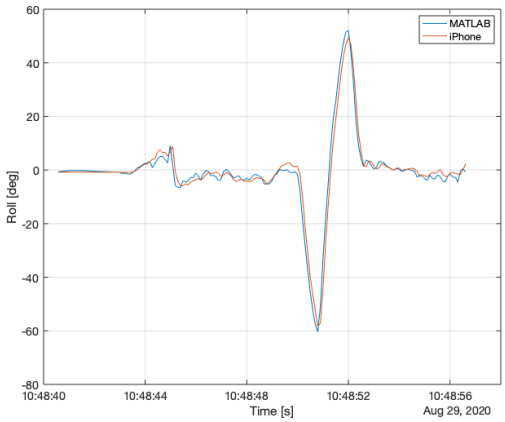
	Noise Density	Random Walk
Accelerometer	5.28 $\times 10^{-4}$ $m/s^2/\sqrt{Hz}$	1.15 $\times 10^{-5}$ $m/s^3/\sqrt{Hz}$
Gyroscope	9.19 $\times 10^{-5}$ $rad/s/\sqrt{Hz}$	3.43 $\times 10^{-6}$ $rad/s^2/\sqrt{Hz}$

Table 5.1: Accelerometer and gyroscope Noise and Random Walk Bias

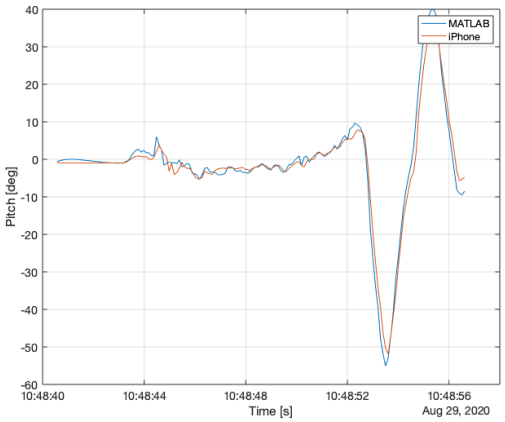
were practically identical, therefore the parameters estimated with the Allan Variance were validated.

Regarding the actual experiment, the water deposit with the WGS84 coordinates N39.5962162 E-8.846342, shown in Figure 5.5, was chosen as the target since it was a landmark that stood out and was easier to detect. In total, a set of 14 images were acquired. For every image, the position and attitude were registered. Although the GPS and IMU data were asynchronous, this did not pose as a major problem since the mobile phone was held at an approximately constant position for more than 1 second when an image was being captured. This ensured that the translation errors caused by GPS and IMU latency were minimized. The image timestamps were also asynchronous with the IMU data so the methodology adopted was to register the attitude and position immediately before the image timestamp.

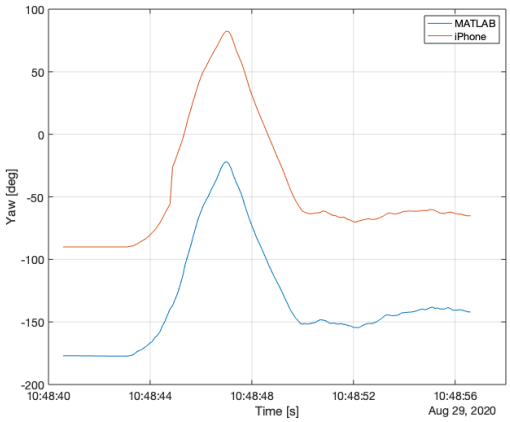
The sample rate was set to 10 Hz in the MATLAB Mobile App and the IMU data was acquired along the trajectory plotted in Figure 5.6. On the left, the trajectory is represented in blue and the target is in red. On the right, the positions where the images were acquired along the trajectory. Some of these images are shown in Figure 5.7 with the target highlighted.



(a) Roll



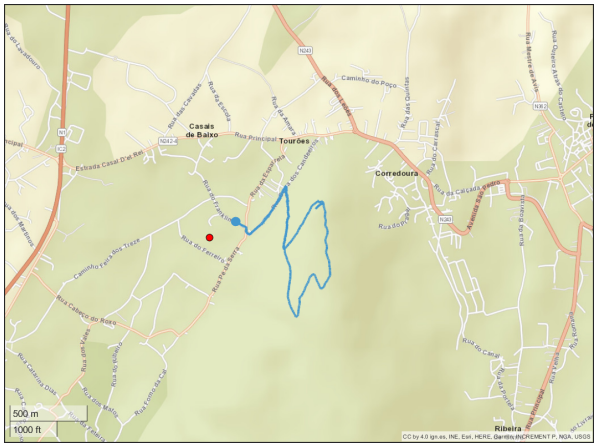
(b) Pitch



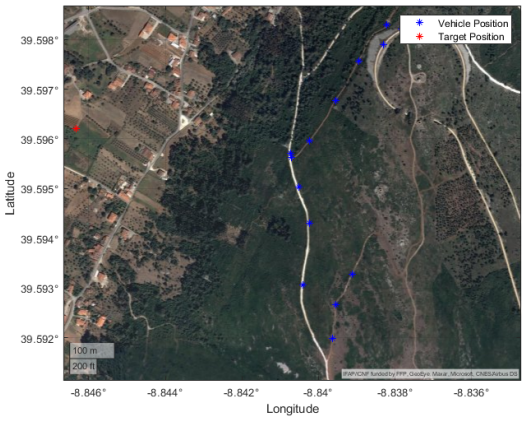
(c) Yaw

Figure 5.4: Comparison between the output iPhone orientation and Kalman Filter estimated orientation

Figure 5.5: Water deposit



(a) Trajectory and target in OpenStreetMap



(b) Trajectory and target satellite view

Figure 5.6: OpenStreetMap and satellite views of the mobile phone experimental procedure configuration

(a) (b)

Figure 5.7: Two images of the mobile phone experimental procedure

5.1.2 Portuguese Air Force UAV Footage

In this video, a UAV recorded a fire burning near Chaves, Vila Real, at 41.631724N -7.465919E, as shown in Figure 5.8. The camera intrinsic parameters were not available. However, the target was locked in the centre of the image, making it possible to geolocate without them because the centre pixel always points in the direction of the camera axis. The gimbal's azimuth and elevation angles are overlayed on the video feed. The former is calculated with respect to the vehicle coordinate frame. The latter, however, is provided with respect to the UAV's body. This was not problematic since the horizon line was parallel to the horizontal image frame. The feed also provided the UAV's position in the WGS84 reference system. A total of 7 frames were collected and used to test the georeferencing algorithm. Due to its confidentiality, no video frame can be presented.

Figure 5.8: Chaves fire location

5.1.3 UAVision UAV Footage

This video recorded a forest fire near Pombal, Leiria, at 39.832856N -8.519885E, as shown in Figure 5.9. As in the previous recording, the overlay displayed information on the azimuth and elevation angles, as well as the UAV's position in the WGS84 reference frame.

Again, the camera was not calibrated, therefore no intrinsic parameters were available. Furthermore, the vehicle performs a slight turn along its trajectory, meaning that the horizon line is not centered and parallel to the horizontal image frame and therefore the elevation angle is not reliable. To overcome these setbacks, three video frames with visible landmarks, presented in Figure 5.10, were selected. This allowed the formulation of a minimization problem to estimate the intrinsic parameters and to refine the rotation matrix of the system when those images were

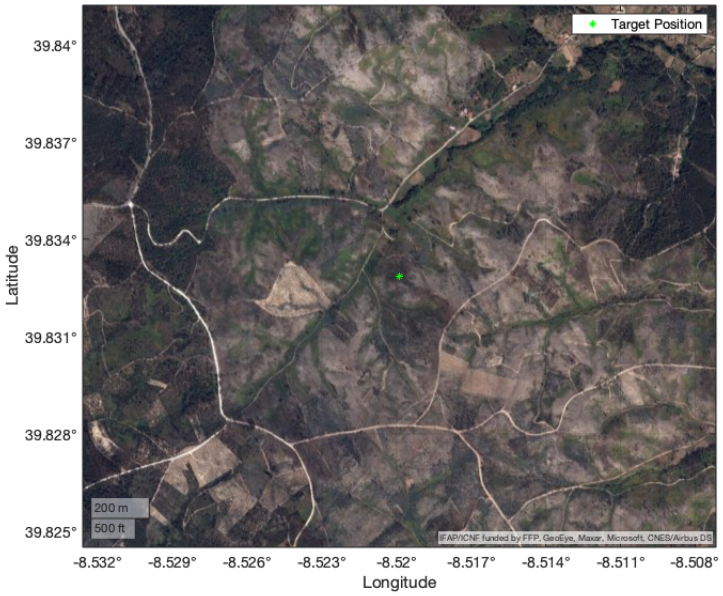


Figure 5.9: Pombal fire location

captured. As a cost function, the reprojection error was used

$$J(R_i, t_i, K) = \sum_{i=1}^{3} \sum_{k=1}^{N_k} (u_{ik} - \hat{u}_{ik})^2 + (v_{ik} - \hat{v}_{ik})^2, \tag{5.1}$$

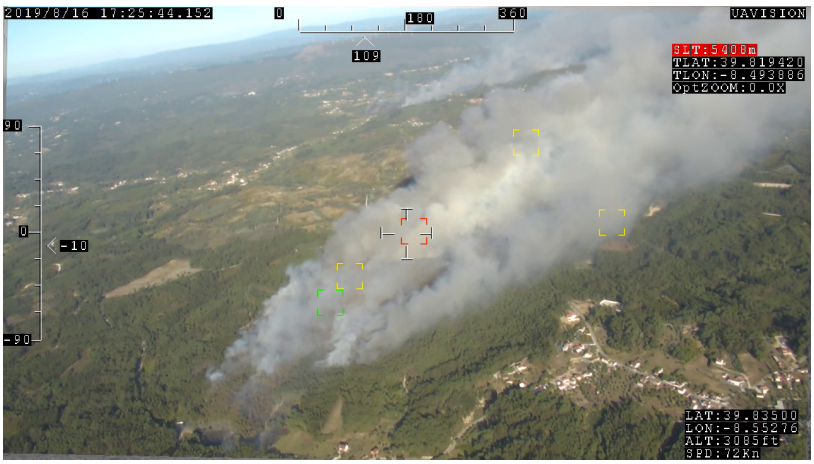
where R_i and t_i are the rotation matrix and translation vector that establish the transformation from world to camera coordinates for frame i, K is the IP matrix, (u_{ik}, v_{ik}) is the measured pixel k in frame i and $(\hat{u}_{ik}, \hat{v}_{ik})$ is the predicted pixel. It was assumed that the skew s was zero, $f_x = f_y$ and that the optical centre matched the image centre, $c_x = 640$ and $c_y = 360$:

$$\begin{aligned} &\arg\min J(R_i, t_i, K) \\ &\quad \text{s.t.} \quad ||r_{i1}|| = 1, \quad ||r_{i2}|| = 1, \quad ||r_{i3}|| = 1 \\ &\qquad\quad r_{i1} r_{i2}^T = 0, \quad r_{i2} r_{i3}^T = 0, \quad r_{i1} r_{i3}^T = 0 \\ &\qquad\quad f_x = f_y \end{aligned}, \tag{5.2}$$

where r_{ik} is the row k of rotation matrix i. The six initial constraints are related to the orthogonality condition of the rotation matrices. The resulting intrinsic parameters and refined rotation matrices were used to calculate the location of some landmarks.

(a) Frame 1

(b) Frame 2



(c) Frame 3

Figure 5.10: Three frames used to calculate the camera's intrinsic parameters

5.2 Experimental Results

5.2.1 Mobile Phone Experimental Results

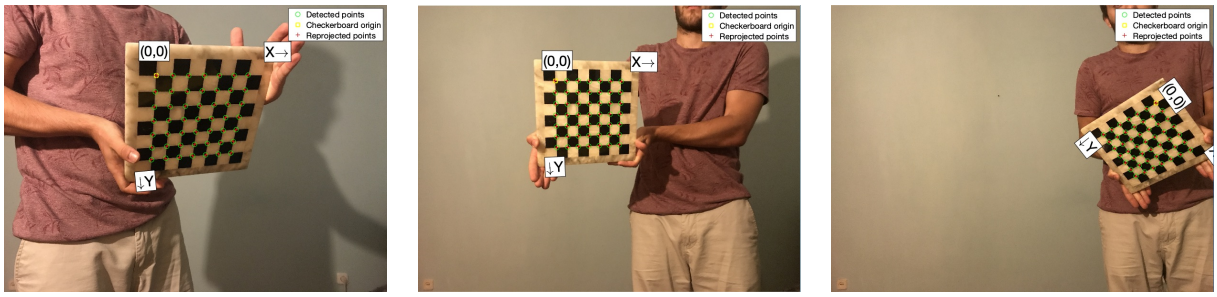
A total of 20 images were used to calibrate the camera. Six are presented in Figure 5.11. As a result, the following intrinsic and distortion parameters were obtained:

$$\mathbf{K_{int}} = \begin{bmatrix} f_x & 0 & cx \\ 0 & f_y & cy \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 3363.507 & 0 & 1967.377 \\ 0 & 3369.501 & 1419.890 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.3}$$

$$\mathbf{k} = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix}^T = \begin{bmatrix} 0.2265 & -1.0227 & 1.7296 \end{bmatrix}^T \tag{5.4}$$

$$\mathbf{p} = \begin{bmatrix} p_1 & p_2 \end{bmatrix}^T = \begin{bmatrix} -0.0098 & -0.0065 \end{bmatrix}^T \tag{5.5}$$

A mean reprojection error of 0.68 pixels was achieved with a 3-parameter radial distortion model.



(a) (b) (c)

Figure 5.11: iPhone image samples for calibration

In Figure 5.12 the trajectory is plotted in a local ENU coordinate system, with the location of the images, the target and the estimated locations of the target. The average distance between the position of the images and target was 640.83 meters, with a maximum of 770.92 meters and 493.90 meters. The results are presented in Table 5.2.

Average Norm of Position Error [m]	RMSE [m]	Average Norm of Position Uncertainty [m]
74.483	77.498	157.035

Table 5.2: Standalone IRT results of the mobile phone experimental procedure

Given the lower quality of the mobile phone IMU, the azimuth and elevation measurement error was tuned to

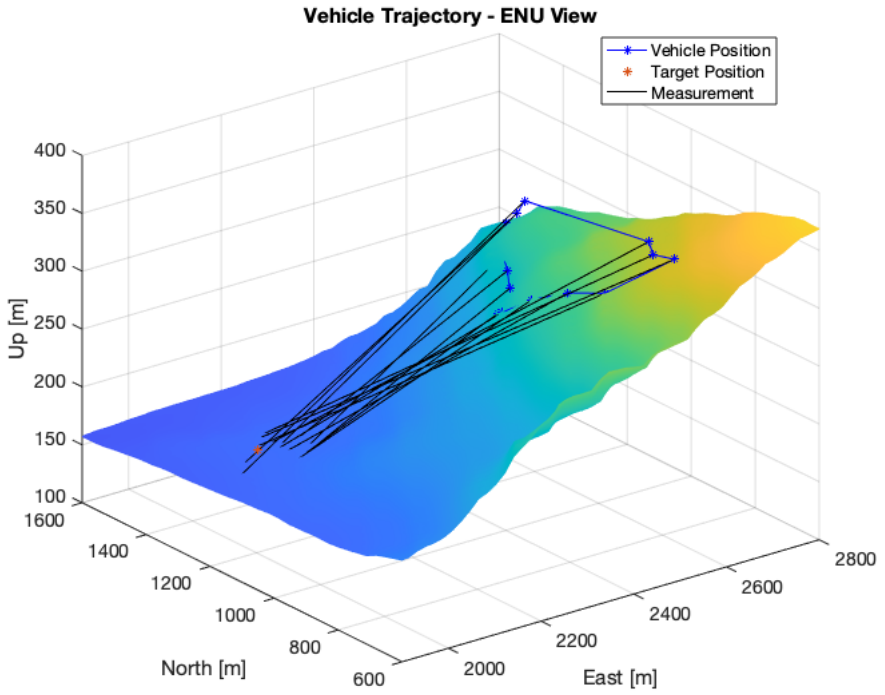


Figure 5.12: Trajectory and IRT results in local ENU coordinate system

5°, i.e.,

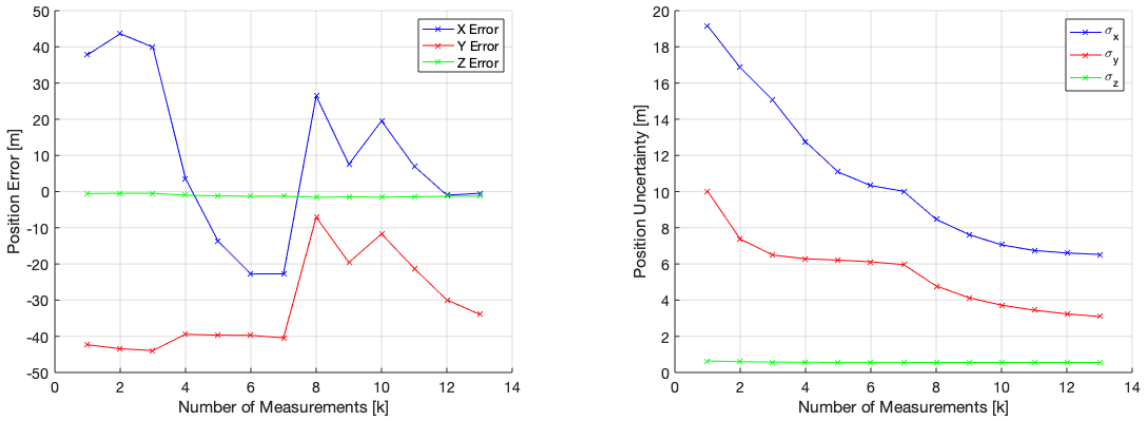
$$\mathbf{R}_k = \begin{bmatrix} 5^2 & 0 & 0 \\ 0 & 5^2 & 0 \\ 0 & 0 & 10^2 \end{bmatrix}. \tag{5.6}$$

The first estimated position was used to initialize the state of the EKF. The other 13 measurements were filtered. Figure 5.13 presents the results of the filtering algorithm. On the left, the estimation error is plotted for each axis and on the right the standard deviation. Table 5.3 presents the final position error, uncertainty and the RMSE.

Norm of Position Error [m]	RMSE [m]	Norm of Position Uncertainty [m]
33.620	41.501	7.250

Table 5.3: BR-EKF filtering results of the mobile phone experimental procedure

Figure 5.14 provides a clear view of the IRT estimated positions and the improvement achieved by applying the EKF. A bias along the positive East direction is visible, since 12 of the 14 estimated positions are on that area. This can be due to the non-ideal experimental setup, i.e., a line-of-sight more parallel to the ground when compared to a more vertical one from an aerial vehicle. The mean position and uncertainty are plotted in Figure 5.15 and evidence this ill-conditioned setup, with a stretched uncertainty region along the line-of-sight direction. This means that a small error on the vertical image direction, be it the selected pixel or the roll angle, is amplified



(a) Position error

(b) Position uncertainty

Figure 5.13: Position error and uncertainty evolution per BR-EKF iteration

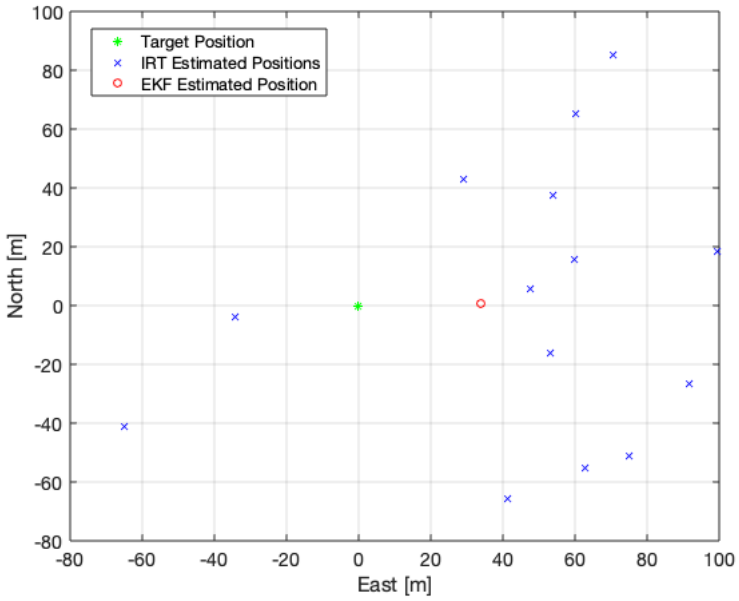


Figure 5.14: Real and estimated target positions by the IRT and BR-EKF algorithms

by this configuration. In addition, the images were taken at approximate positions, limiting the new information added to the BR-EKF.

The same experiment was simulated to investigate the origin of this bias. More specifically, to understand if it was due to the non-ideal perspective or if it was related to IMU or pixel identification errors. The mean position and uncertainty are plotted in Figure 5.16. The results presented in Table 5.4 demonstrate the non-ideal perspective of the experimental procedure, with an increase by almost 100% in the RMSE and 72 meters in the position uncertainty when compared to the results of the experimental procedure presented in Table 5.2. Furthermore, the

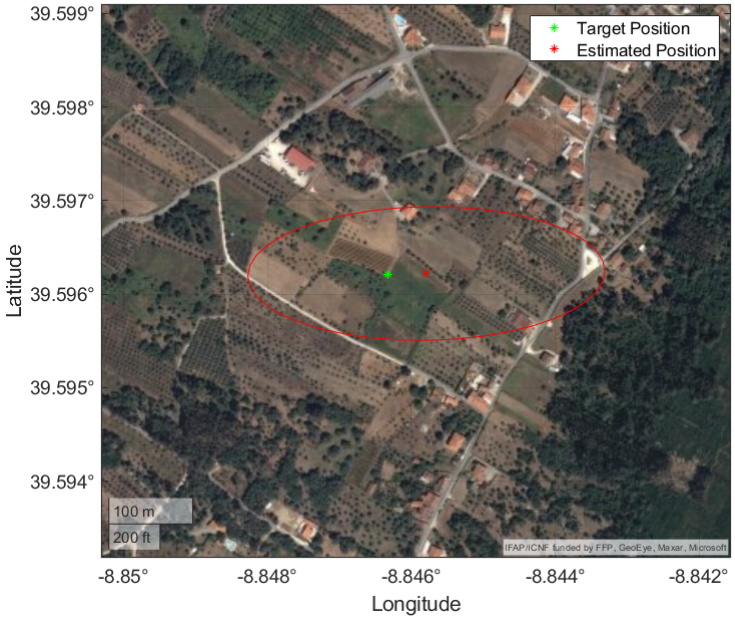


Figure 5.15: Mean position and uncertainty from experimental procedure

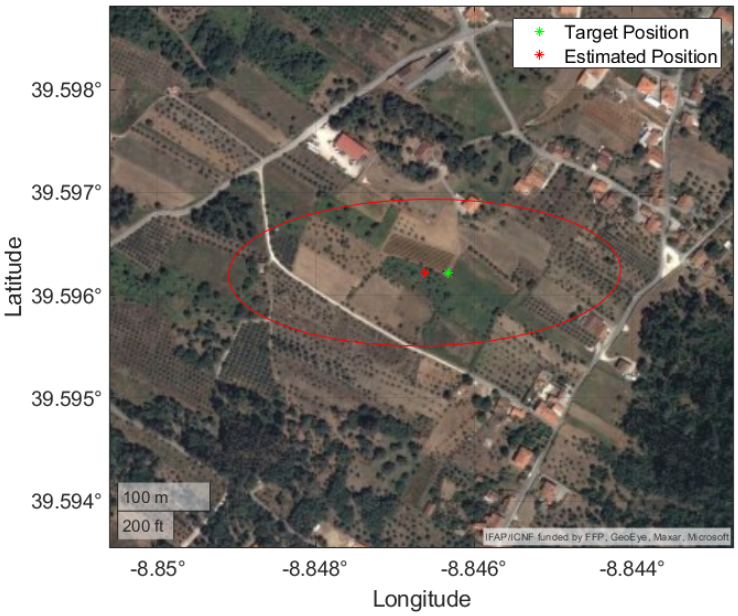


Figure 5.16: Mean position and uncertainty from ground perspective simulation

mean estimated target position presented Figure 5.16 is shifted towards West when compared to the actual target position, showing that the bias identified in the experimental procedure, despite being towards East, is mainly caused by the sub-optimal line-of-sight.

Method	Iterations	Norm of Position Error [m]	RMSE [m]	Norm of Position Uncertainty [m]
IRT		134.377	140.2487	229.571
IRT+BR-EKF	13	73.923	94.686	4.055

Table 5.4: Results from the Porto de Mós experimental procedure simulated with a ground perspective

Finally, the same trajectory was simulated with an aerial perspective at a height of 950 meters. The results presented in Table 5.5 validate the use of aerial vehicles when applying the proposed methodology. The RMSE

Method	Iterations	Norm of Position Error [m]	RMSE [m]	Norm of Position Uncertainty [m]
IRT		47.120	47.605	43.560
IRT+BR-EKF	13	13.908	21.946	5.109

Table 5.5: Results from the Porto de Mós experimental procedure simulated with an aerial perspective

decreased by 36 meters and the position uncertainty was reduced by 113 meters, as a result of a more perpendicular line-of-sight evidenced in Figure 5.17. In addition, the biases identified in the experimental procedure and ground simulation are no longer present since the mean estimated target position is practically identical to the real target position. In conclusion, the use of an aerial vehicle is clearly advantageous when applying the proposed algorithm, providing a more accurate estimate of the target's position, which is further improved with the Bearings-Range EKF.

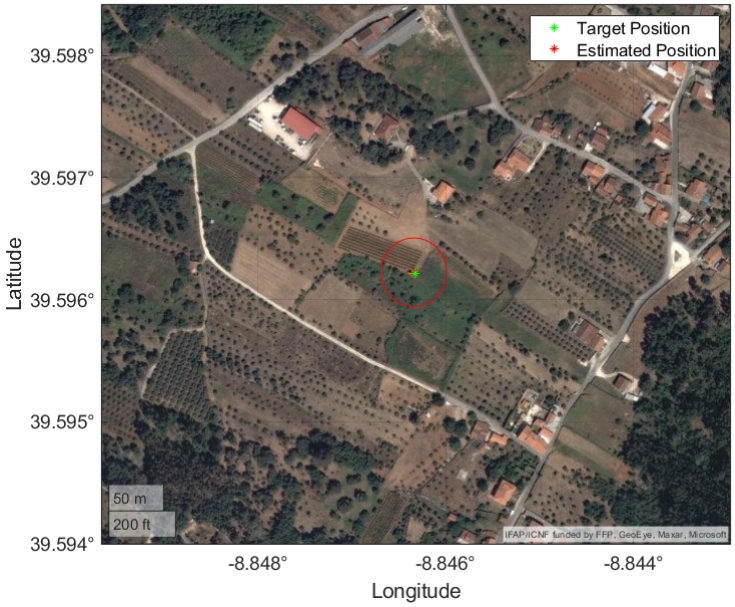
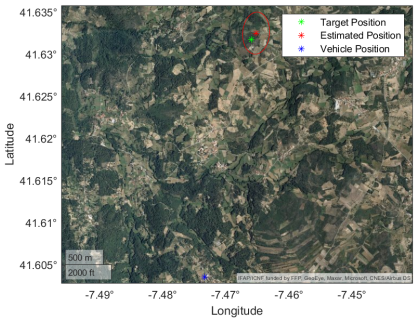


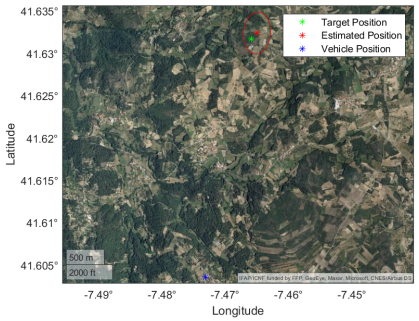
Figure 5.17: Mean position and uncertainty from aerial perspective simulation

5.2.2 Portuguese Air Force UAV Footage Results

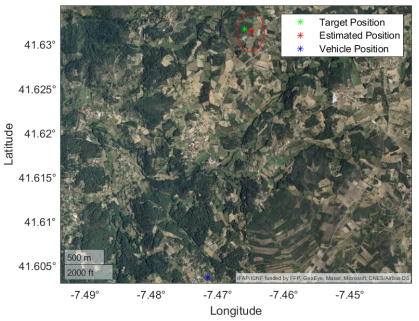
In this experiment, the UAV flew at approximately 1920 meters, and the average distance to the target was 3183 meters, meaning that the horizontal component was 2539 meters. As in the Porto de Mós experimental procedure, the optic ray is more parallel to the ground when compared to the perspective of an aerial vehicle, explaining why the uncertainty regions plotted in Figure 5.18 are stretched along the line-of-sight direction. In 6 out of 7 frames, the target is included in the uncertainty region.



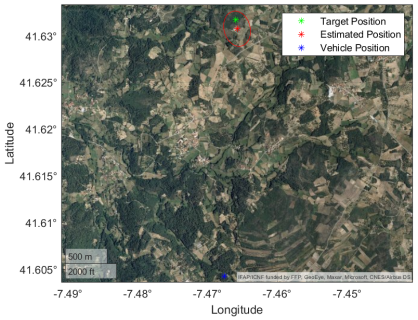
(a)



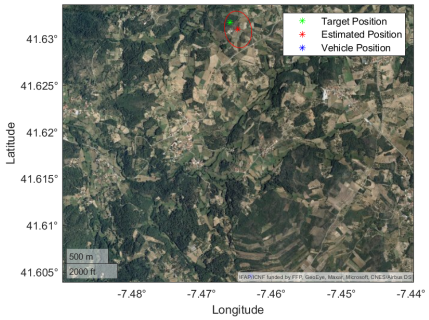
(b)



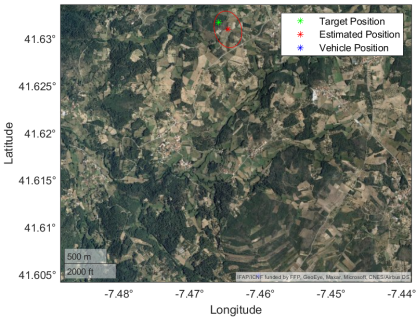
(c)



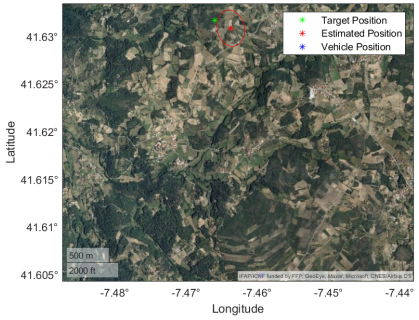
(d)



(e)



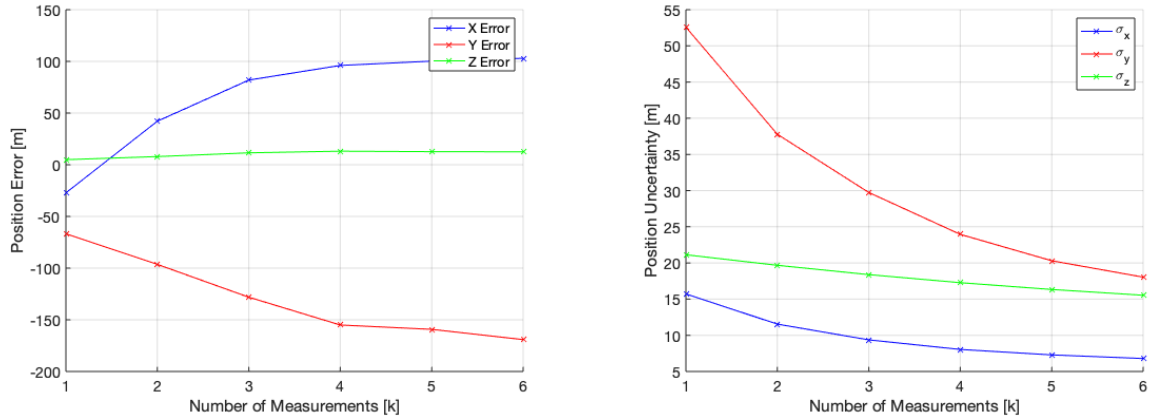
(f)



(g)

Figure 5.18: Portuguese Air Force footage georeferencing results

The fact that the observations are very similar means that no new information is added on each EKF iteration. Furthermore, only 7 measurements were taken, where one was used to initialize the filter state and covariance and only six were left to iterate. These factors contribute to the increase in the position error estimated by the filtering algorithm, as presented in Figure 5.19 and summarized in Table 5.6. An important conclusion can be drawn from this. When georeferencing targets at greater distances, the uncertainties associated with the estimated positions are amplified. Therefore, the number and variability of measurements required for an accurate target position increases.



(a) EKF position error

(b) EKF position uncertainty

Figure 5.19: Chaves BR-EKF position error and uncertainty

Method	Iterations	Norm of Position Error [m]	RMSE [m]	Norm of Position Uncertainty [m]
IRT		136.250	146.049	292.600
IRT+BR-EKF	6	198.434	157.056	24.7578

Table 5.6: Portuguese Air Force IRT and IRT+EKF results

5.2.3 UAVision UAV Footage Results

The estimated intrinsic parameters from the minimization problem were

$$\mathbf{K_{int}} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1063.17 & 0 & 640 \\ 0 & 1063.17 & 360 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.7}$$

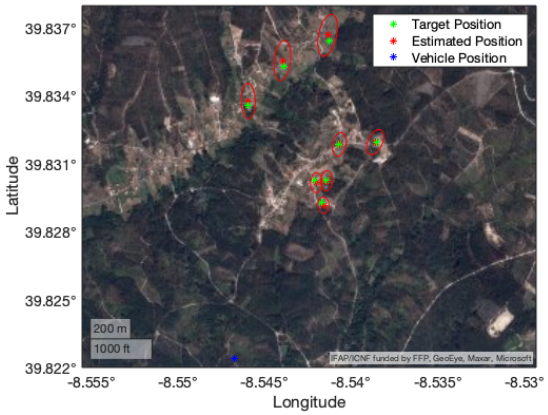
These parameters and the calculated rotation matrices were used to georeference the position of several landmarks. Table 5.7 presents the number of landmarks selected in each frame and the average norm of the position error,

average norm of position uncertainty and RMSE. The targets, estimated positions and uncertainty regions are plotted in Figure 5.20.

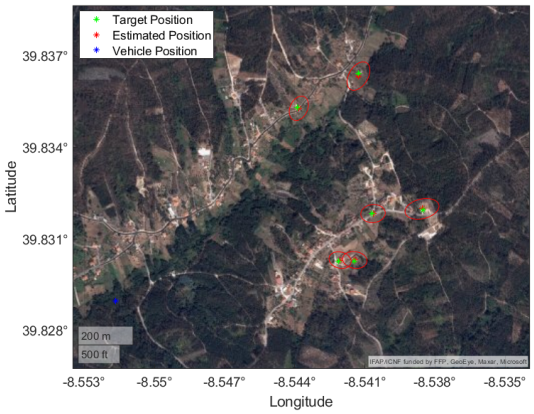
Frame	Landmarks	Norm of Position Error	RMSE [m]	Norm of Position Uncertainty [m]
1	8	15.166	16.432	81.277
2	6	6.598	6.857	60.226
3	4	25.908	26.029	65.896

Table 5.7: UAVision footage results

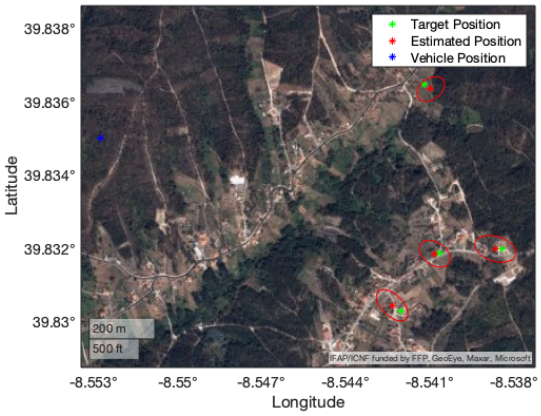
Frames 1 and 2 present similar results in terms of position error and RMSE. Position uncertainty is greater on the first frame because it is georeferencing targets that are further from the vehicle than in the second frame. The third and final frame presents the worst results in terms of position error and RMSE. This is due to being the image with less landmarks, and as a consequence is less refined by the minimization problem when compared to the other frames. Overall, accurate results were obtained for targets that distanced more than one kilometer from the UAV. In a real scenario, were the position error and RMSE will not be available, the uncertainty is taken into account as metric of the georeferencing algorithm, with a lower value representing a more trustworthy estimated position.



(a) Frame 1



(b) Frame 2



(c) Frame 3

Figure 5.20: UAVision georeferencing results

6

Conclusions

Contents

Having validated the proposed methodology, this chapter presents the main achievements of this thesis, its conclusions and suggestion for future work.

6.1 Achievements

The IRT algorithm proposed in this thesis presents a robust solution for forest fire georeferencing. It only requires the camera's IPs, the onboard sensor data (GPS/IMU/gimbal) and a DEM, whilst current state-of-the-art approaches rely on feature identification and matching.

Equipped with the UT, the georeferencing algorithm provides an estimate of the target position and characterizes its uncertainty. That is a relevant achievement for the FIREFRONT project since it provides crucial information regarding the confidence of the estimated target position to the fire fighting personnel.

Finally, a novel Bearings-Range filter measurement model was developed to filter the bearings and range information from relative geometry between the aerial vehicle and the estimated position.

The results obtained in a controlled simulation environment demonstrate the potential of the developed algorithm. At distances of almost 1000 meters, two simulations were run on rough and flat terrains, achieving RMSEs of 30.7 and 43.4 meters with 25 and 21 measurements, respectively. After applying the BR-EKF, these errors reduced to 11.0 and 19.9 meters, a decrease of 58.5% and 54.7%. This relation was also verified in the mobile phone experimental procedure, where an initial RMSE of 77.5 meters was reduced to 41.5 meters with just 14 measurements. Finally, georeferencing with accurate intrinsic and extrinsic parameters, as was the case in the UAVision experimental procedure, resulted in very accurate estimated positions. An average RMSE of 16.33 meters was achieved on a total of 18 landmarks that distanced more than 1000 meters from the UAV.

6.2 Future Work

Regarding the acquisition of multiple images of the same target, it was seen in the literature that the results of the georeferencing algorithms are influenced by the flight pattern. These can also influence the filtering algorithm since it strives on differentiated measurements. For this reason, the next step would be to design flight patterns that minimize the error of the IRT algorithm but also maximize the information extracted from Bearings-Range measurement model.

Another possible future work would be the development of a cooperative georeferencing algorithm that allows multiple aerial vehicles to simultaneously geolocate an object of interest. That would increase the problem complexity and require extra coordination with the aerial vehicles fighting the fire but would allow a faster and more accurate georeferencing.

A mobile phone application could be developed, considering the results obtained with the mobile phone experimental procedure. The user would upload telemetry and fire imagery and these could be used to georeference

the fire together with the data from the aerial vehicle or as an alternative, in case something prevented it from operating.

Lastly, considering the procedure followed in the UAVision footage, an algorithm could be developed that searches recognizable landmarks in the imagery acquired by the aerial vehicle. These landmarks could then be georeferenced using image registration techniques and used to refine the camera's EPs.

Bibliography

[1] Concurso de Projetos de Investigação Científica e Desenvolvimento Tecnológico no Âmbito da Prevenção e Combate a Incêndios Florestais - 2017, . URL https://www.fct.pt/apoios/projectos/consulta/areas.phtml.pt?idElemConcurso=12320. visited on 19-12-2019.

[2] Concurso de Projetos de Investigação Científica e Desenvolvimento Tecnológico no Âmbito da Prevenção e Combate a Incêndios Florestais - 2018, . URL https://www.fct.pt/apoios/projectos/consulta/areas.phtml.pt?idElemConcurso=14042. visited on 19-12-2019.

[3] ASTER GDEM v2. URL https://asterweb.jpl.nasa.gov/gdem.asp. visited on 05-08-2020.

[4] Chapter 3 Digital Terrain Model (DTM). URL http://wtlab.iis.u-tokyo.ac.jp/wataru/lecture/rsgis/giswb/vol2/cp3/cp3-1.htm. visited on 02-08-2020.

[5] DSM vs DEM. URL https://www.geoimage.com.au/DEMS/dems-overview. visited on 02-08-2020.

[6] EU-DEM v1.1. URL https://land.copernicus.eu/imagery-in-situ/eu-dem/eu-dem-v1.1?tab=metadata. visited on 26-07-2020.

[7] Open Source Sensor Fusion. URL https://github.com/memsindustrygroup/Open-Source-Sensor-Fusion/tree/master/docs. visited on 13-08-2020.

[8] Unscented Transform. URL http://ais.informatik.uni-freiburg.de/teaching/ws13/mapping/pdf/slam06-ukf-4.pdf. visited on 25-12-2019.

[9] Ienkaran Arasaratnam and Simon Haykin. Cubature Kalman Filters. *IEEE Transactions on Automatic Control*, 2009.

[10] D. Blake Barber, Joshua D. Redding, Timothy W. McLain, Randal W. Beard, and Clark N. Taylor. Vision-based target geo-location using a fixed-wing miniature air vehicle. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 2006.

[11] Mark Beighley and A C Hyde. Portugal Wildfire Management in a New Era: Assessing Fire Risks , Resources and Reforms. (February), 2018.

[12] Gianpaolo Conte, Maria Hempel, Piotr Rudol, David Lundström, Simone Duranti, Mariusz Wzorek, and Patrick Doherty. High accuracy ground target geo-location using autonomous micro aerial vehicle platforms. *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008.

[13] Davide Antonio Cucci, Martin Rehak, and Jan Skaloud. Bundle adjustment with raw inertial observations in UAV applications. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2017.

[14] Tom G. Farr, Paul A. Rosen, Edward Caro, Robert Crippen, Riley Duren, Scott Hensley, Michael Kobrick, Mimi Paller, Ernesto Rodriguez, Ladislav Roth, David Seal, Scott Shaffer, Joanne Shimada, Jeffrey Umland, Marian Werner, Michael Oskin, Douglas Burbank, and Douglas Alsdorf. The Shuttle Radar Topography Mission. *Reviews of Geophysics*, 2007.

[15] Gianfranco Forlani, Fabrizio Diotri, Umberto Morra di Cella, and Riccardo Roncella. Indirect UAV Strip Georeferencing by On-Board GNSS Data under Poor Satellite Coverage. *Remote Sensing*, 2019.

[16] R. V. Garcia, P. C.P.M. Pardal, H. K. Kuga, and M. C. Zanardi. Nonlinear filtering for sequential spacecraft attitude estimation with real data: Cubature Kalman Filter, Unscented Kalman Filter and Extended Kalman Filter. *Advances in Space Research*, 2019.

[17] Mehran Ghandehari, Barbara P. Buttenfield, and Carson J. Q. Farmer. Cross-Scale Analysis of Sub-pixel Variations in Digital Elevation Models. In *Advances in Cartography and GIScience*. Springer International Publishing, 2017.

[18] Mehran Ghandehari, Barbara P Buttenfield, and Carson J Q Farmer. Comparing the accuracy of estimated terrain elevations across spatial resolution. *International Journal of Remote Sensing*, 2019.

[19] M. Hamidi and F. Samadzadegan. Precise 3D geo-location of UAV images using geo-referenced data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 2015.

[20] Fangning He, Tian Zhou, Weifeng Xiong, Seyyed Meghdad Hasheminnasab, and Ayman Habib. Automated aerial triangulation for UAV-based mapping. *Remote Sensing*, 2018.

[21] Janne Heikkila and Olli Silven. Four-step camera calibration procedure with implicit image correction. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997.

[22] M Heikkinenm. Geschlossene formeln zur berechnung räumlicher geodätischer koordinaten aus rechtwinkligen koordinaten. *Z. Vermess.*, 1982.

[23] Christian Heipke, Krasten Jacobsen, H Wegmann, and Birgir Nilsen. Integrated sensor orientation-an OEEPE Test. *International Archives of the Photogrammetry and Remote Sensing*, 2000.

[24] Elder M. Hemerly. Automatic georeferencing of images acquired by UAV's. *International Journal of Automation and Computing*, 2014.

[25] Instituto da Conservação da Natureza e das Florestas. 8.º relatório provisório de incêndios rurais. 2019. URL `https://www.agroportal.pt/wp-content/uploads/2019-10-16-RPIR-08-01jan-15out.pdf`. visited on 19-12-2020.

[26] Dong Wook Kim, Hee Sup Yun, Sang Jin Jeong, Young Seok Kwon, Suk Gu Kim, Won Suk Lee, and Hak Jin Kim. Modeling and testing of growth status for Chinese cabbage and white radish with UAV-based RGB imagery. *Remote Sensing*, 2018.

[27] Frederik S. Leira, Kenan Trnka, Thor I. Fossen, and Tor Arne Johansen. A ligth-weight thermal camera payload with georeferencing capabilities for small fixed-wing UAVs. *2015 International Conference on Unmanned Aircraft Systems, ICUAS 2015*, 2015.

[28] Xiaoye Liu. Airborne LiDAR for DEM generation: Some critical issues. *Progress in Physical Geography*, 2008.

[29] A. M.G. Lopes, A. C.M. Sousa, and D. X. Viegas. Numerical simulation of turbulent flow and fire propagation in complex topography. *Numerical Heat Transfer; Part A: Applications*, 1995.

[30] A. M.G. Lopes, L. M. Ribeiro, D. X. Viegas, and J. R. Raposo. Simulation of forest fire spread using a two-way coupling algorithm and its application to a real wildfire. *Journal of Wind Engineering and Industrial Aerodynamics*, 2019.

[31] Luis Merino, Fernando Caballero, J. R. Martínez-de Dios, Joaquín Ferruz, and Aníbal Ollero. A cooperative perception system for multiple UAVs: Application to automatic detection of forest fires. *Journal of Field Robotics*, 2006.

[32] Antonios Mouratidis and Dimitrios Ampatzidis. European digital elevation model validation against extensive global navigation satellite systems data and comparison with SRTM DEM and ASTER GDEM in Central Macedonia (Greece). *ISPRS International Journal of Geo-Information*, 2019.

[33] Rafael Batalha Parcelas. Development of additional sensor capabilities for use in unmanned aerial vehicles under CIDIFA Electrotechnic Engineering. Master's thesis, Portuguese Air Force Academy, 2018.

[34] Sameera S. Ponda, Richard M. Kolacinski, and Emilio Frazzoli. Trajectory optimization for target localization using small unmanned aerial vehicles. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August), 2009.

[35] Bernardo Santana, Alexandre Bernardino, and Ricardo Ribeiro. Direct Georeferencing of Forest Fire Aerial Images using Iterative Ray-Tracing and a Bearings-Range Extended Kalman Filter. *Proceedings of RECPAD 2020*, 2020.

[36] Yongwei Sheng. Comparative evaluation of iterative and non-iterative methods to ground coordinate determination from single aerial images. *Computers and Geosciences*, 2004.

[37] Yongwei Sheng. Theoretical analysis of the iterative photogrammetric method to determining ground coordinates from photo coordinates and a DEM. *Photogrammetric Engineering and Remote Sensing*, 2005.

[38] Haitao Xiang and Lei Tian. Method for automatic georeferencing aerial remote sensing (RS) images from an unmanned aerial vehicle (UAV) platform. *Biosystems Engineering*, 2011.

[39] Chen Xiong, Qiangsheng Li, and Xinzheng Lu. Automated regional seismic damage assessment of buildings using an unmanned aerial vehicle and a convolutional neural network. *Automation in Construction*, 2020.

[40] Cheng Xu, Daqing Huang, and Jianye Liu. Target location of unmanned aerial vehicles based on the electro-optical stabilization and tracking platform. *Measurement*, 2019.

[41] Lele Zhang, Jie Chen, Fang Deng, Yingcai Bi, Swee King Phang, and Xudong Chen. Trajectory Planning for Improving Vision-Based Target Geolocation Performance Using a Quad-Rotor UAV. *IEEE Transactions on Aerospace and Electronic Systems*, 2018.

[42] Lele Zhang, Fang Deng, Jie Chen, Yingcai Bi, Swee King Phang, Xudong Chen, and Ben M. Chen. Vision-Based Target Three-Dimensional Geolocation Using Unmanned Aerial Vehicles. *IEEE Transactions on Industrial Electronics*, 2018.

[43] Zhengyou Zhang. Flexible camera calibration by viewing a plane from unknown orientations. *Proceedings of the IEEE International Conference on Computer Vision*, 1999.

[44] J. Zhu. Conversion of Earth-centered Earth-fixed coordinates to geodetic coordinates. 1994.

Uncertainty Propagation on a Linear Surface

A.1 Optic Ray

The optic ray is defined as the vector with origin in the camera's optical centre and pointing at the pixel one intends to project.

This vector is defined in the camera's coordinate frame as

$$P_C = \begin{bmatrix} u - c_x \\ v - c_y \\ \frac{f_x + f_y}{2} \end{bmatrix}, \tag{A.1}$$

assuming $f_x \approx f_y$, where $[u, v]^T$ is the pixel, $[c_x, c_y]^T$ is the principal point and f_x, f_y are the camera's focal lengths with respect to x and y. Normalizing,

$$P'_C = \frac{1}{\sqrt{(u - c_x)^2 + (v - c_y)^2 + (\frac{f_x + f_y}{2})^2}} P_C \tag{A.2}$$

Depending on the system configuration, this vector can be transformed to an inertial coordinate frame, where it is designated as P_I.

A.2 Plane - Optic Ray Intersection

A plane is defined by

$$ax + by + cz + d = 0, \tag{A.3}$$

where (a, b, c) are the coordinates of the plane's normal vector $\vec{n}$ and d is the distance to the origin of the coordinate frame. Any point (x, y, z) belonging to said plane verifies (A.3).

The optic ray can be written with the following equation:

$$r(x, y, z) = (v_x, v_y, v_z) + k(P_{Ix}, P_{Iy}, P_{Iz}), \tag{A.4}$$

where (v_x, v_y, v_z) is the camera's position in the inertial frame.

To calculate the intersection between the optic ray and the plane, one solves (A.5) with respect to *k*, obtaining (A.6),

$$\begin{cases} x = v_x + kP_{Ix} \\ y = v_y + kP_{Iy} \\ z = v_z + kP_{Iz} \\ ax + by + cz + d = 0 \end{cases}, \tag{A.5}$$

$$k = -\frac{d + av_x + bv_y + cv_z}{aP_{Ix} + bP_{Iy} + cP_{Iz}}. \tag{A.6}$$

The coordinates of the intersection are given by

$$\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} v_x - \frac{d+av_x+bv_y+cv_z}{aP_{Ix}+bP_{Iy}+cP_{Iz}} P_{Ix} \\ v_y - \frac{d+av_x+bv_y+cv_z}{aP_{Ix}+bP_{Iy}+cP_{Iz}} P_{Iy} \\ v_z - \frac{d+av_x+bv_y+cv_z}{aP_{Ix}+bP_{Iy}+cP_{Iz}} P_{Iz} \end{bmatrix}. \tag{A.7}$$

The incidence angle, represented in Figure A.1, is given by

$$\theta_i = \cos^{-1}(-\vec{P_I} \cdot \vec{n}), \quad \theta_i \in [0, \frac{\pi}{2}[\tag{A.8}$$

Rewritten,

$$\theta_i = \cos^{-1}(-(aP_{Ix} + bP_{Iy} + cP_{Iz})), \quad \theta_i \in [0, \frac{\pi}{2}[\tag{A.9}$$

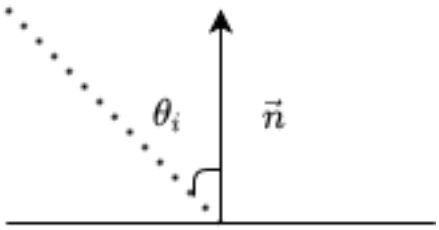


Figure A.1: Incidence angle

Considering (A.7) and (A.9), one can conclude that the intersection coordinates are dependent on the incidence angle. Therefore, (A.7) can be rewritten as

$$\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} v_x + \frac{d+av_x+bv_y+cv_z}{\cos\theta_i} P_{Ix} \\ v_y + \frac{d+av_x+bv_y+cv_z}{\cos\theta_i} P_{Iy} \\ v_z + \frac{d+av_x+bv_y+cv_z}{\cos\theta_i} P_{Iz} \end{bmatrix}. \tag{A.10}$$

The overall configuration is represented in Figure A.2.

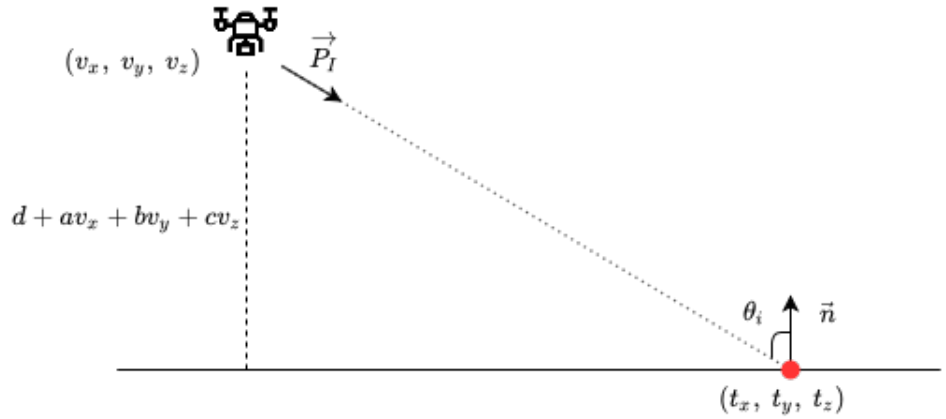


Figure A.2: Optic ray projection to linear surface

A.3 Propagation of Position Uncertainty

Considering that the camera's position is affected by errors, $v'_{x,y,z} = v_{x,y,z} \pm \delta_{x,y,z}$ and replacing in (A.7),

$$\begin{bmatrix} t'_x \\ t'_y \\ t'_z \end{bmatrix} = \begin{bmatrix} (v_x \pm \delta_x) + \frac{d+a(v_x \pm \delta_x)+b(v_y \pm \delta_y)+c(v_z \pm \delta_z)}{\cos\theta_i} P_{Ix} \\ (v_y \pm \delta_y) + \frac{d+a(v_x \pm \delta_x)+b(v_y \pm \delta_y)+c(v_z \pm \delta_z)}{\cos\theta_i} P_{Iy} \\ (v_z \pm \delta_z) + \frac{d+a(v_x \pm \delta_x)+b(v_y \pm \delta_y)+c(v_z \pm \delta_z)}{\cos\theta_i} P_{Iz} \end{bmatrix}. \tag{A.11}$$

The error is calculated subtracting (A.11) to (A.10),

$$\begin{cases} \epsilon_x = \mp\delta_x + \frac{P_{Ix}}{\cos\theta_i}(\mp a\delta_x \mp b\delta_y \mp c\delta_z) \\ \epsilon_y = \mp\delta_y + \frac{P_{Iy}}{\cos\theta_i}(\mp a\delta_x \mp b\delta_y \mp c\delta_z) \\ \epsilon_z = \mp\delta_z + \frac{P_{Iz}}{\cos\theta_i}(\mp a\delta_x \mp b\delta_y \mp c\delta_z) \end{cases}. \tag{A.12}$$

The result deducted in (A.12) shows that as the incidence angle increases, the more amplified are the position errors. The minimum is achieved with $\theta_i = 0°$.

A.4 Propagation of Orientation Uncertainty

Considering now that the camera's orientation is affected by errors,

$$\begin{bmatrix} t'_x \\ t'_y \\ t'_z \end{bmatrix} = \begin{bmatrix} v_x - \frac{d+av_x+bv_y+cv_z}{aP'_{Ix}+bP'_{Iy}+cP'_{Iz}} P'_{Ix} \\ v_y - \frac{d+av_x+bv_y+cv_z}{aP'_{Ix}+bP'_{Iy}+cP'_{Iz}} P'_{Iy} \\ v_z - \frac{d+av_x+bv_y+cv_z}{aP'_{Ix}+bP'_{Iy}+cP'_{Iz}} P'_{Iz} \end{bmatrix}. \tag{A.13}$$

Rewritten,

$$\begin{bmatrix} t'_x \\ t'_y \\ t'_z \end{bmatrix} = \begin{bmatrix} v_x + \frac{d+av_x+bv_y+cv_z}{\cos\theta'_i} P'_{Ix} \\ v_y + \frac{d+av_x+bv_y+cv_z}{\cos\theta'_i} P'_{Iy} \\ v_z + \frac{d+av_x+bv_y+cv_z}{\cos\theta'_i} P'_{Iz} \end{bmatrix}. \tag{A.14}$$

The error is calculated subtracting (A.14) to (A.10),

$$\begin{cases} \epsilon_x = (d + av_x + bv_y + cv_z)(\frac{P_{Ix}}{\cos\theta_i} - \frac{P'_{Ix}}{\cos\theta'_i}) \\ \epsilon_y = (d + av_x + bv_y + cv_z)(\frac{P_{Iy}}{\cos\theta_i} - \frac{P'_{Iy}}{\cos\theta'_i}) \\ \epsilon_z = (d + av_x + bv_y + cv_z)(\frac{P_{Iz}}{\cos\theta_i} - \frac{P'_{Iz}}{\cos\theta'_i}) \end{cases}. \tag{A.15}$$

The result obtained in (A.15) shows that the distance to the plane, $(d+av_x+bv_y+cv_z)$, amplifies the orientation errors. In addition, it is also shown that the incidence angle influences the error propagation, and this influence is proportional to the derivative value on the central point. The derivative of $\cos x$, $\frac{d}{dx}\cos x = -\sin x$, has a minimum for $x = 0$ and maximum for $x = \frac{\pi}{2}$. In conclusion, incidence angles closer to 90°increase significantly the error propagation when compared to incidence angles close to 0°.

B

RECPAD2020 Paper

Direct Georeferencing of Fire Front Aerial Images using Iterative Ray-Tracing and a Bearings-Range Extended Kalman Filter

Bernardo Santana[2]
bernardo.santana@tecnico.ulisboa.pt
Alexandre Bernardino[1]
alex@isr.tecnico.ulisboa.pt
Ricardo Ribeiro[1]
ribeiro@isr.tecnico.ulisboa.pt

[1] Institute for Systems and Robotics
Instituto Superior Técnico
Lisbon, Portugal

[2] MSc Student,
Instituto Superior Técnico
Lisbon, Portugal

Abstract

This paper discusses the design and implementation of the Iterative Ray-Tracing algorithm for forest fire georeferencing using aerial imagery, a Global Positioning System (GPS), an Inertial Measurement Unit (IMU) and a Digital Elevation Model (DEM). Taking into account that measurement errors are amplified by the target distance, an Extended Kalman Filter (EKF) is proposed to filter multiple observations of the same object of interest. This filter extracts the bearings and range information from the geometric relation between the target and the camera in a local coordinate system. A performance comparison is done with a Cubature Kalman Filter (CKF) considering possible linearization errors induced by the EKF.

In order to validate the georeferencing and filtering algorithms, an experiment was conducted. A mobile phone was used to acquire GPS, IMU and 14 images of a target. An average position error of 74.483m was obtained at an average distance of 605m. Applying the Bearings-Range EKF and CKF reduced the error to 33.620 and 33.820, respectively.

1 Introduction

Forest fires are increasingly becoming a frequent problem in modern day society. Their destructive potential makes them a serious concern and a challenge for firefighting authorities.

Fire propagation models have already been studied that take into account weather variables such as wind [7] and also the terrain type [6]. However, these models usefulness is limited since no fire geolocation algorithm has been developed for this purpose. Henceforth, the aim of this work is to fill in this gap and develop a georeferencing algorithm based on images and telemetry recorded by an aerial vehicle. This images are assumed to be pre-processed to identify the pixels that correspond to fire.

1.1 Related Work

Forlani et al. [3] apply direct georeferencing by using the on-board Global Navigation Satellite System with the Real-Time Kinematic option with Structure from Motion and Bundle Adjustment. No ground control points are used. This methodology is, however, not suitable in a forest fire scenario, where the lack of differentiated texture and smoke prevents feature extraction and matching.

Conte et al. [2] propose an image registration approach by pattern-matching the images collected from a Micro Aerial Vehicle with satellite imagery. Multiple measurements are taken and recursive least square filter is applied. Similarly to [3], this technique relies on feature extraction, and is therefore unreliable in a forest fire environment.

Ponda et al. [8] develop a Line-of-Sight Bearings-Only EKF for target localization. This requires, however, a prior knowledge of the target's position, which is not reviewed in that work. Xu et al. [10] propose the same measurement model using a CKF instead, considering possible linearization errors induced by the standard EKF. To determine an initial approximation of the target's position, the Iterative Photogrammetry (IP) algorithm [9] is used. In spite of being efficient, the IP method can diverge when the incidence angle is smaller than the profile inclination angle.

Leira et al. [5] propose the intersection of the optic ray with a flat surface. This generalization, however, is not suitable in rough terrains, as seen in [10].

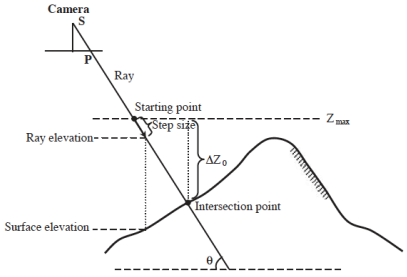


Figure 1: Iterative Ray-Tracing (adapted from [9]).

2 Georeferencing Algorithm - Iterative Ray-Tracing

The proposed georeferencing algorithm is the Iterative Ray-Tracing (IRT) [9], presented in Figure 1, and the DEM used is the EU-DEM v1.1 [1], with a spatial resolution of 25 meters and georeferenced in EPSG:3035. Since the purpose of this work is to output the geodetic coordinates of the target, this map is converted to the EPSG:4326.

The IRT works by extending the optic ray with a step size until it hits the surface. A GPS and IMU are needed to define the origin and direction of this ray, respectively, in a local NED frame. The intersection is detected when the point elevation is equal or smaller than the elevation of the DEM.

Multiple upgrades were introduced in the basic IRT, including a dynamic step size, to increase the accuracy of the algorithm. Furthermore, the starting iteration point was set as the intersection of the ray with the maximum elevation of the loaded DEM. It is expected that the aerial vehicles will operate at heights greater than the local terrain, and this can reduce the number of iterations considerably. Finally, bilinear interpolation was implemented to refine the elevation of the queried point. Ghandehari et al. [4] concluded in their work that for DEM's with finer resolutions, such has the EU-DEM v1.1, this type of interpolation achieves good results with low processing times.

3 Bearings-Range Filter

3.1 Target Dynamic Model

In this work, the target is assumed to be stationary. Therefore, its dynamics can be modeled by $\mathbf{t}_{k+1} = \Phi_{k+1|k}\mathbf{t}_k + \mathbf{Q}_k$, where $\mathbf{t}_k$ represents the target position, $\Phi_{k+1|k}$ the state transition matrix and $\mathbf{Q}_k$ the process covariance matrix:

$$\Phi_{k+1|k} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{Q}_k = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{1}$$

3.2 Bearings-Range Measurement Model

The measurement model is given by $\mathbf{z}_{k+1} = \mathbf{h}(\mathbf{t}_{k+1}) + \mathbf{R}_k$, where $\mathbf{z}_{k+1}$ is the new measurement, $\mathbf{h}$ is the non-linear measurement function and $\mathbf{R}_k$ is the measurement noise covariance matrix.

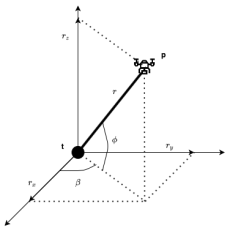


Figure 2: Bearings-Range model geometry.

$$\mathbf{h}(\mathbf{t}_{k+1}) = \begin{bmatrix} \beta \\ \phi \\ r \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{p_y - t_y}{p_x - t_x}\right) \\ \tan^{-1}\left(\frac{p_z - t_z}{\sqrt{(p_x - t_x)^2 + (p_y - t_y)^2}}\right) \\ \sqrt{(p_x - t_x)^2 + (p_y - t_y)^2 + (p_z - t_z)^2} \end{bmatrix}, \quad (2)$$

where β and ϕ are the azimuth and elevation angles, respectively, and r is the distance between the target, $\mathbf{t}$, and the aerial vehicle, $\mathbf{p}$, as presented in Figure 2.

4 Experiment

The unavailability of telemetry and imagery data from an aerial vehicle led to the development of an alternative methodology to validate the proposed algorithm. A mobile phone was used to record GPS, IMU and image data along a pedestrian path. The natural elevation of *Serra dos Candeeiros*, near *Porto de Mós*, *Leiria*, was used to capture images of a target at a lower height, so as to simulate the overview of an aerial vehicle. A total of 14 images were acquired at an average target distance of 605 meters. For the filtering, the IRT result for the first observation is used to initialize the filter state, $\mathbf{t}_0$. The initial state covariance $\mathbf{P}_0$ and measurement noise covariance matrix $\mathbf{R}_k$ were tuned to

$$\mathbf{P}_0 = \begin{bmatrix} 20^2 & 0 & 0 \\ 0 & 50^2 & 0 \\ 0 & 0 & 1^2 \end{bmatrix}, \quad \mathbf{R}_k = \begin{bmatrix} 5^2 & 0 & 0 \\ 0 & 5^2 & 0 \\ 0 & 0 & 10^2 \end{bmatrix}. \quad (3)$$

Details on the EKF and CKF algorithms can be found in [8] and [10], respectively.

The position error is defined as $\mathbf{e_p} = \mathbf{t} - \hat{\mathbf{t}}$, where $\hat{\mathbf{t}}$ is the estimated target. σ_x, σ_y and σ_z are defined as the square root of the filter state covariance matrix diagonal. The results of the standalone IRT, EKF and CKF are summarized in Table 1.

Method	$\|\|e_p\|\|$ [m]	$\|\|\sigma_{x,y,z}\|\|$ [m]
IRT	74.483	n.d.
IRT+EKF	33.620	7.2497
IRT+CKF	33.820	7.2502

Table 1: Norm of the average position error for the standalone IRT and for the final correction of the EKF and CKF.

The IRT results presented in Figure 3 evidence a bias along the positive East direction, which then influences the estimated positions of the EKF and CKF.

5 Conclusions

In this paper, the IRT is proposed as a georeferencing algorithm using the EU-DEM v1.1. Expecting measurement errors from the GPS and IMU, a bearings-range filtering algorithm was developed, with a performance comparison between the EKF and CKF. Preliminary results using the data collected with a mobile phone show evidence of bias susceptibility. This may be due to the non-ideal preliminary experimental setup using a line of sight more parallel to the ground when compared to the more vertical one from an aerial vehicle. Furthermore, the 14 images were captured at approximate positions, limiting the new information added to the filtering algorithm. Still, an improvement of 41 meters is achieved on the 74 meter

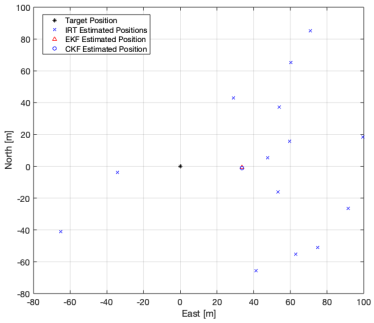


Figure 3: Real and estimated target positions by the IRT, EKF and CKF algorithms.

average position error of the standalone IRT. There is no clear advantage in using the CKF over the EKF for this measurement model.

Acknowledgements

This work was supported by FCT with the LARSyS - FCT Project UIDB/50009/2020 and project FIREFRONT (PCIF/SSI/0096/2017).

References

[1] EU-DEM v1.1. URL `https://land.copernicus.eu/imagery-in-situ/eu-dem/eu-dem-v1.1`. visited on 26-07-2020.

[2] Gianpaolo Conte, Maria Hempel, Piotr Rudol, David Lundström, Simone Duranti, Mariusz Wzorek, and Patrick Doherty. High accuracy ground target geo-location using autonomous micro aerial vehicle platforms. *AIAA Guidance, Navigation and Control Conference and Exhibit*, pages 1–14, 2008.

[3] Gianfranco Forlani, Fabrizio Diotri, Umberto Morra di Cella, and Riccardo Roncella. Indirect UAV Strip Georeferencing by On-Board GNSS Data under Poor Satellite Coverage. *Remote Sensing*, 11(15):1765, 2019.

[4] Mehran Ghandehari, Barbara P Buttenfield, and Carson J Q Farmer. Comparing the accuracy of estimated terrain elevations across spatial resolution. *International Journal of Remote Sensing*, 40(13): 5025–5049, 7 2019.

[5] Frederik S. Leira, Kenan Trnka, Thor I. Fossen, and Tor Arne Johansen. A ligth-weight thermal camera payload with georeferencing capabilities for small fixed-wing UAVs. *2015 International Conference on Unmanned Aircraft Systems, ICUAS 2015*, pages 485–494, 2015.

[6] A. M.G. Lopes, A. C.M. Sousa, and D. X. Viegas. Numerical simulation of turbulent flow and fire propagation in complex topography. *Numerical Heat Transfer; Part A: Applications*, 27(2):229–253, 2 1995.

[7] A. M.G. Lopes, L. M. Ribeiro, D. X. Viegas, and J. R. Raposo. Simulation of forest fire spread using a two-way coupling algorithm and its application to a real wildfire. *Journal of Wind Engineering and Industrial Aerodynamics*, 193(July):103967, 2019.

[8] Sameera S. Ponda, Richard M. Kolacinski, and Emilio Frazzoli. Trajectory optimization for target localization using small unmanned aerial vehicles. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August), 2009.

[9] Yongwei Sheng. Comparative evaluation of iterative and non-iterative methods to ground coordinate determination from single aerial images. *Computers and Geosciences*, 30(3):267–279, 2004.

[10] Cheng Xu, Daqing Huang, and Jianye Liu. Target location of unmanned aerial vehicles based on the electro-optical stabilization and tracking platform. *Measurement*, 147, 12 2019.

C

Application Developed for the Civil Protection

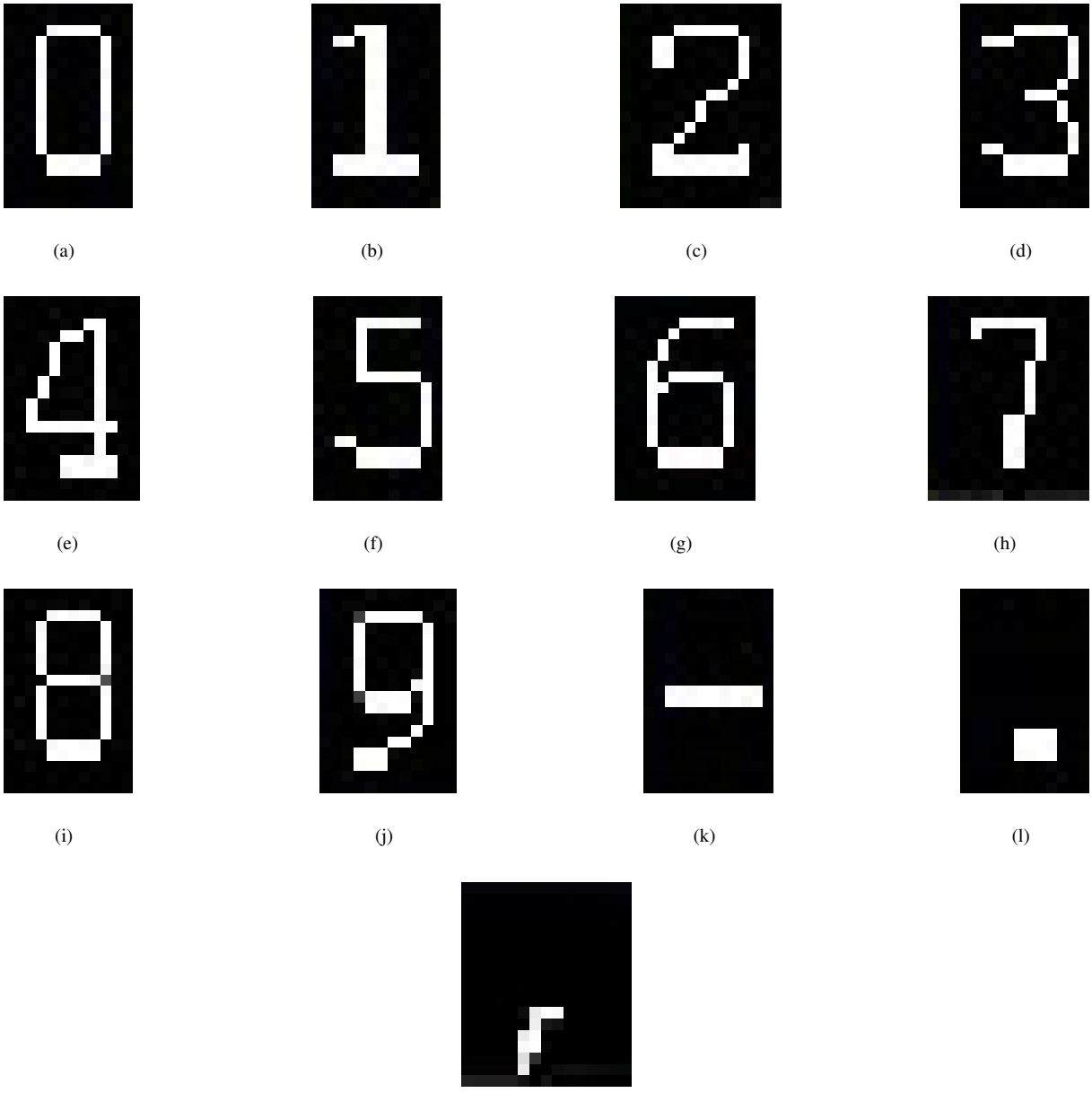
The current georeferencing methodology of the Civil Protection is dependent on the availability of operators to analyze the imagery acquired by the aerial vehicles. These are manually georeferenced by inspection with the help of visible landmarks and satellite imagery. Therefore, this task is too time consuming and prone to errors.

With the purpose of aiding the Civil Protection, an Optical Character Reader (OCR) was developed to read the telemetry overlayed on the video feeds transmitted by the aerial vehicles. This OCR was then successfully integrated with the georeferencing algorithm developed in this thesis, demonstrating its operational potential.

This OCR follows a template-matching approach to extract the desired data (gimbal azimuth, gimbal elevation, camera zoom, vehicle latitude, vehicle longitude and vehicle altitude) from the feed. The characters presented in Figure C.1 were extracted from multiple videos and are used as templates. As can be seen in Figure C.2, the telemetry is placed on a fixed positions with respect to the image frame. To reduce the processing time, the original image is cropped on these positions, as shown in Figure C.3. The gimbal telemetry moves on a bounded area of the image, depending on the angle values, so the image is cropped along this area.

Two approaches were tested for video processing: to analyze all frames and perform the georeferencing algorithm as soon as the data is updated by the UAV software; to process a frame every N frames, where this interval is configurable. The first approach proved to be too time-consuming, and since the feed is updated at least once per second, the errors induced by this delay are negligible, in practicality. Therefore, the second solution was chosen. The output of the program is presented in Figure C.4.

To improve this application, a graphical user interface can be developed where the operator selects the target to georeference and the result is presented directly on satellite imagery. The Python code is available in the following GitHub repository: `https://github.com/bernardosanti/georeferencing_irt_ut`.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

(m)

Figure C.1: Character templates extracted from the overlay and used in the template-matching OCR

Figure C.2: Original image

(a) Gimbal elevation

(b) Gimbal azimuth

(c) Zoom

(d) Vehicle latitude

(e) Vehicle longitude

(f) Vehicle altitude

Figure C.3: Image cropping for the template-matching OCR

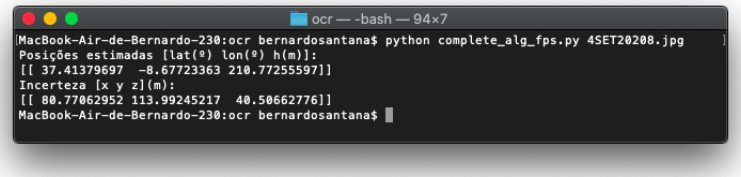


Figure C.4: Software output