Master of Science Thesis in Electrical Engineering
Department of Electrical Engineering, Linköping University, 2016

Wildlife Surveillance Using a UAV and Thermal Imagery

Albin Flodell and Cornelis Christensson

LINKÖPING
UNIVERSITY

Master of Science Thesis in Electrical Engineering

Wildlife Surveillance Using a UAV and Thermal Imagery

Albin Flodell and Cornelis Christensson

LiTH-ISY-EX--16/4968--SE

Supervisor: **Clas Veibäck**
ISY, Linköpings universitet

Examiner: **Fredrik Gustafsson**
ISY, Linköpings universitet

Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden

Copyright © 2016 Albin Flodell and Cornelis Christensson

Tillägnas alla noshörningar

Sammanfattning

På senare år har tjuvjakten på noshörningar resulterat i ett kritiskt lågt bestånd. Detta examensarbete är en del av ett initiativ för att stoppa denna utveckling. Målet är att använda en UAV, utrustad med GPS och attitydsensorer, samt en värmekamera placerad på en gimbal, till att övervaka vilda djur. Genom att använda en värmekamera kan djuren lätt detekteras eftersom de antas vara varmare än sin omgivning. En modell av marken vid testområdet har använts för att möjliggöra positionering av detekterade djur, samt analys av vilka områden på marken som ses av kameran.

Termen övervakning inkluderar detektion av djur, målföljning och planering av rutt för UAV:n. UAV:n ska kunna söka av ett område efter djur. För att göra detta krävs planering av trajektoria för UAV:n samt hur gimbalen ska förflyttas. Flera metoder för detta har utvärderats. UAV:n ska även kunna målfölja djur som har detekterats. Till detta har ett partikelfilter använts. För att associera mätningar till spår har *Nearest Neighbor*-metoden använts. Djuren detekteras genom att bildbehandla på videoströmmen som ges från värmekameran. För bildbehandlingen har flertalet metoder testats.

Dessutom presenteras en omfattande beskrivning av hur en UAV fungerar och är uppbyggd. I denna beskrivs även nödvändiga delar för ett UAV-system. På grund av begränsningar i budgeten har ingen UAV inköpts. Istället har tester utförts från en gondol i Kolmården. Gondolen åker runt i testområdet med en konstant hastighet.

Djur kunde lätt detekteras och målföljas givet en kall bakgrund. Då solen värmer upp marken är det svårare att särskilja djuren från marken och fler feldetektioner görs av bildbehandlingen.

Abstract

In recent years, the poaching of rhinoceros has decreased its numbers to critical levels. This thesis project is a part of an initiative to stop this development. The aim of this master thesis project is to use a UAV equipped with positioning and attitude sensors as well as a thermal camera, placed onto a gimbal, to perform wildlife surveillance. By using a thermal camera, the animals are easily detected as they are assumed to be warmer than the background.

The term wildlife surveillance includes detection of animals, tracking, and planning of the UAV. The UAV should be able to search an area for animals, for this planning of the UAV trajectory and gimbal attitude is needed. Several approaches for this have been tested, both online and offline planning. The UAV should also be able to track the animals that are detected, for this a particle filter has been used. Here a problem of associating measurements to tracks arises. This has been solved by using the Nearest Neighbor algorithm together with gating. The animals are detected by performing image processing on the images received from the thermal camera. Multiple approaches have been evaluated.

Furthermore, a thoroughly worked description of how a UAV is working as well as how it is built up is presented. Here also necessary parts to make up a full unmanned aerial system are described. This chapter can be seen as a good guide for beginners, to the UAV field, interested in knowing how a UAV works and the most common parts of such a system.

A ground model of Kolmården, where the testing has been conducted, has been used in this thesis. The use of this enables positioning of the detected animals and checking if an area is occluded for the camera.

Unfortunately, due to budget limitations, no UAV was purchased. Instead, testing has been conducted from a gondola in Kolmården traveling across the test area with a constant speed. To use the gondola as the platform, for the sensors and the thermal camera, is essentially the same as using a UAV as both alternatives are located in the air above the animals, both are traveling around the map and both are stable for good weather conditions.

The animals could easily be detected and tracked given a cold background. When the sun heats up the ground, it is harder to distinguish the animals in the thermal video, and more false detections in the image processing appear.

Acknowledgments

First of all we would like to thank Kolmårdens djurpark for letting us use parts of the park and the gondola as test area. They have been welcoming and generous with their time. We would also like to give our thanks to FLIR, for lending us a high-end thermal camera. The camera was invaluable during our field tests at Kolmårdens djurpark.

An important person in this thesis project was our supervisor Clas Veibäck. He has given good advices on the project, and good suggestions for how to improve this report. He has also been a good friend.

We would also like to thank our examiner Fredrik Gustafsson, for the opportunity to make this thesis project. He has shown interest in the project and been a great inspiration and support. Also Gustaf Hendeby have been a great support, with his realistic point of view. Both of you show a great inspirational passion for science and for sensor fusion.

Two very important persons for us, are our fellow friends, Adam and Erik. We would like to give them our greatest thanks for company and for sharing their equipment with us. You have contributed to the good spirit in our office.

Another thanks goes to Emily, for helping us find all the missing commas.

Linköping, June 2016
Albin Flodell and Cornelis Christensson

Contents

Notation

Abbreviations

Abbreviation	Meaning
BEC	Battery Eliminator Circuit
DIY	Do It Yourself
DOF	Degrees Of Freedom
ENU	East North Up
EO	Electro-Optical
ESC	Electronic Speed Control
FOV	Field Of View
FPV	First-Person View
GCS	Ground Control System
GPS	Global Positioning System
IERS	International Earth Rotation Service
IMU	Internal Measurement Unit
IR	Infrared
Li-ion	Lithium-ion
LOS	Line Of Sight
MAVLink	Micro Air Vehicle Link
NED	North East Down
OS	Operating System
POI	Point Of Interest
RC	Radio Control
RHC	Receding Horizon Control
RTF	Ready To Fly
RTSP	Real Time Streaming Protocol
SDK	Software Development Kit
SIR	Sampling Importance Resampling
SIS	Sampling Importance Sampling
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
WGS84	World Geodetic System 1984

1 Introduction

With an autonomous UAV (Unmanned Aerial Vehicle) guarding on the savanna, the endangered animals there could be safer. Park rangers could guard more efficiently since they could focus their efforts on the important areas. This master thesis project investigates which hardware, and which ways of planning and tracking to use, for the development of an autonomous surveillance UAV equipped with a thermal camera.

The chapter starts with a more thorough background, continues with a problem description and delimitations for the project. It ends with a short literature study, showing what have been done previously on the subject.

1.1 Background

The world is going through a wildlife crisis. Biological diversity decreases as endangered species are hunted down to critical numbers by poachers. In the 1970s there were 20000 black rhinos in Kenya. Today there are only 650 left [16]. This sad development must, and may, come to an end. Linköping University has started an initiative, Smart Savannahs, as a technical demonstrator for wildlife security [20]. The demonstrator is first to be deployed in Project Ngulia, which is a co-operation to protect the rhinos at Ngulia Rhino Sanctuary in Tsavo West National Park, Kenya, from poaching. The long term goal is to scale up the project to include other national parks in Kenya and beyond. The project contains many different subprojects, such as 3D modeling, acoustic tracking, radar surveillance, aerial surveillance, radio localization, and image learning.

The following study is a part of Project Ngulia, and will consider aerial surveillance. Aerial surveillance through UAVs is an important part of the project, since they generate many opportunities for a low cost. A UAV equipped with appropriate sensors can easily get an overview of a large area in a short time. For Project

Ngulia the first goal is to use autonomous UAVs to enable easy monitoring of the location of the rhinos. The second goal is to use UAVs to search for poachers such that the park security can be warned of intruders. With the help of UAVs the amount of rhinos killed by poachers can hopefully be reduced.

With a thermal camera, objects such as humans or animals can be detected from a distance. The test site for this thesis project will be Kolmårdens djurpark (Kolmården Wildlife Park), Sweden. The test site is as good of a resemblance of Ngulia Rhino Sanctuary as possible in Sweden. Kolmårdens djurpark has an area called safari, about 1000x500 m in size, which consist of several enclosures. Each of these enclosures represent an environment across the globe, examples are the savanna and the nordic. Several species are often roaming in the same enclosure.

1.2 Problem Description

This thesis project investigates the possibilities of using UAVs to perform autonomous wildlife surveillance. The UAV will be equipped with a thermal imaging sensor mounted on a gimbal. This enables easier detection of animals in the video stream compared to using an EO (Electro-Optical) camera. The term autonomous wildlife surveillance in this case includes autonomous searching and tracking of animals. The UAV must also plan a trajectory which enables coverage of the ground. The problem can be divided into five sub-problems:

- Hardware research
- Autonomous planning of route with camera movements, to cover the ground without occlusions
- Detection of animals
- Tracking of animals on the ground
- Visualization of data

To solve the problem, a ground model is available. The ground model is a height map over Kolmårdens djurpark, which will be the test area for the probject. The ground model is described in Section 2.1. By the use of a ground model, the bearing information from a camera can be transformed to a position on the ground. This enables tracking of the animals.

1.2.1 Hardware Research

This section describes the problem of finding hardware that enables autonomous planning, searching and tracking using a thermal camera. The idea of the thesis project is to use a UAV to do wildlife surveillance. This means that the UAV must have load capacity to carry a thermal camera with gimbal, a communication link to the ground, a control unit, and several sensors such as a GPS (Global Positioning System) and IMUs (Internal Measurement Unit). Also, a computer is needed to perform the planning and image processing. This can either be placed on-board the UAV or on the ground. The communication link must be able to stream video directly from the thermal camera down to the ground as well as send commands from a ground station to the UAV. More DOFs (Degree of Freedoms) are added to the problem as the gimbal can point the camera in another direction than the UAV. This enables coverage of a larger area in less distance since the gimbal can rotate the camera to point at the surrounding area. However a problem arises here since the attitude of the camera must be known to be able to distinguish what each picture portrays.

1.2.2 Planning

This section describes the problem of planning a route which enables complete coverage of the ground. The problem is simple if the ground is considered flat, meaning no occlusions will arise. However this approximation of reality is poor if the ground is rugged. This characteristic applies to the area where the demonstration of the thesis project will take place, thus the approximation of the ground as flat would lead to inadequate results. Instead the planning of the route must be done using the information given in the 3D map. The planning problem can be broken down further into smaller problems.

The first problem is to determine the view of the camera. Given the pose and the FOV (Field Of View) for the camera an algorithm must be developed which determines the footprint of the camera on the ground. The map does not include information of trees or buildings, meaning the planning algorithm has to take this into consideration.

An extension to this problem is to determine the footprint given a trajectory of the UAV, and vice versa the trajectory given a desired footprint. Here it is important to consider that the camera is able to change pose so that the surrounding area can also be seen. However as the field of view of the camera is limited the movement of the gimbal must be planned.

The problem of planning how the gimbal should move is highly connected to the problem of planning the route of the UAV. The goal of the planning is to find all animals within one enclosure and then move on to the next enclosure. The algorithm must also consider occlusion effects which might appear. This problem is harder than it sounds as the animals can move and the assumption that all animals will be able to fit in one picture is not always valid. This means that the animals must be tracked and often a big part of the enclosure must be searched. The planning algorithm should continuously take decisions given the current information.

1.2.3 Detection

The detection problem is to decide whether an animal is present in the current image or not. This can be decided by using image processing algorithms. It is assumed that the animals are warmer than the background.

1.2.4 Association and Tracking

The association problem is to associate detected animals to known targets, or to new targets. Each object should be tracked such that the system has an estimation of where all objects are located. This is necessary to be able to count the amount of animals within one enclosure. If no tracking was used each observation would be seen as a new object, and each object not in the field of view would be forgotten.

1.2.5 Visualization

The outcome of the program must be able to be visualized to the user for an easy understanding. A 3D map is already available. The detected animals should be visualized in this 3D model as well as the uncertainties of the position of animals. Furthermore, the trajectory of the sensor platform should be displayed, together with the FOV to clearly show what is visible.

1.3 Delimitations

Mapping is not a part of this thesis project, instead the map is known at all times. The map is represented as a 3D-map, described in Chapter 2.1.

An interesting task is to classify the animals. A possible approach to this problem is to use a secondary EO-camera mounted on the UAV and overlay the two video streams. In this thesis project, only data from a thermal camera will be used, so all animals will be considered to be one animal type. No further classification will be done.

Whilst visualization is a powerful fault detection tool the main focus in the thesis project will not be on this part. However basic visualizations will be produced in the project.

1.4 Literature Study

In recent years the intensity of research in the UAV area has increased rapidly. One example of a research project utilizing the possibilities of UAVs is the NOAH project [43]. The project aims to protect threatened species in Africa using various technologies. One of these technologies is using UAVs to detect and track poachers. They are focusing on fixed-wing UAVs equipped with both thermal and EO cameras to perform autonomous patrolling of an area. This is very related to the long-term goal of project Ngulia.

Several image processing strategies are discussed in the thesis [48], which also discusses the design and implementation of a UAS (Unmanned Aerial System) including an infra-red camera. The UAS should perform detection and tracking of humans. For tracking a Kalman filter is used together with a linear motion model.

Another thesis relevant to this work is [45] which solely focuses on the image processing problem. Here the specific problem of finding rhinos with both an infra-red and an EO video stream is investigated and several strategies proposed. The focus is on image processing algorithms, and how to use feature descriptors and classifiers to classify humans and animals.

When it comes to planning the trajectory of the UAV there are two main approaches. The first approach described in [46] uses information based path planning. The goal is to plan the trajectory for a fixed-wing UAV performing search and tracking. The UAV is equipped with a camera with limited FOV. The planning tries to maximize the information in a limited time horizon. The level of information depends on the uncertainty of each target position estimation as well as the number of targets found. The planning considers both the movement of the UAV and the movement of the gimbal directing the camera. Also occlusions from buildings are taken into account when planning the trajectory.

The second approach to the planning problem is the deterministic approach where the trajectory is calculated from the beginning. In [26] several deterministic trajectory planning algorithms are compared by both simulating and testing in a real environment using a UAV. More specific, the planning was made for search and rescue missions. These are however, similar to the surveillance problem; to search an area as quickly as possible.

In [41] the problem of using a UAV indoors to track targets by using an EO camera is investigated. To estimate the position of the aircraft the EKF algorithm is used. Since the aircraft is indoors no GPS is available, instead only data from an IMU is used for positioning the aircraft. Targets are detected using a pinhole camera model. The UAV uses a deterministic planning algorithm to control the UAV.

2

Preliminaries

This chapter describes preliminary theory and components used in the thesis project. More extensive theory such as theory for planning, detection, and tracking, are assigned individual chapters and are described in Chapters 4 and 5.

The outline of this chapter is as follows. First, the ground model used in the project is described together with its representation. All coordinate systems used in the project are then presented together with conversion methods between them. Many coordinate systems have been used so it is important to review this theory extensively. The last part of this chapter considers the LOS (Line Of Sight) problem, the problem of how to determine what part of the ground that the camera is able to see given its position and attitude.

2.1 The Ground Model

The ground model is a topographic map over the Kolmården area, which is the test area of the project. The map is represented by a 2D matrix where each element holds the altitude of the current position. The altitude scale represents meters above the sea level. In the x- and y-direction the resolution of the map is 2 meter. The resolution of the altitude is 0.01 meter. The ground model only describes the ground profile and thus, does not include buildings or trees. More specific, the map shows the safari area of Kolmården which consists of large enclosures where animals from different natural habitats roam. The size of that area is about 1000x500 m and the map can be seen in Figure 2.1. The ground model also includes information about the location of the different enclosures. These areas are represented by a set of points which define the corners of the enclosures. The safari area where the tests will be conducted together with the enclosures are visualized in Figure 2.2

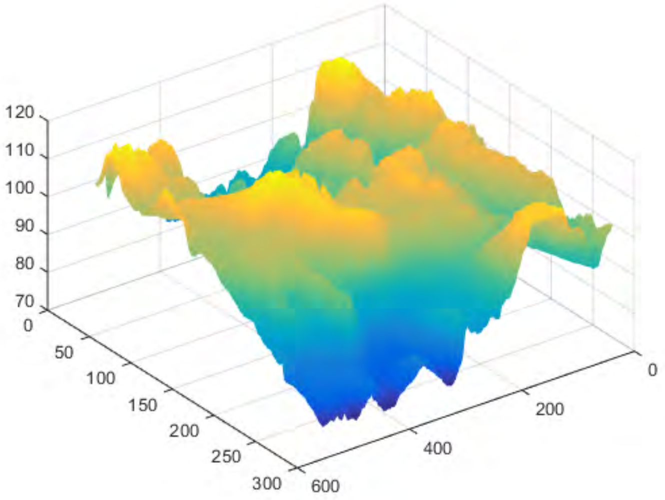


Figure 2.1: *The 3D map of the safari area in Kolmården Zoo.*

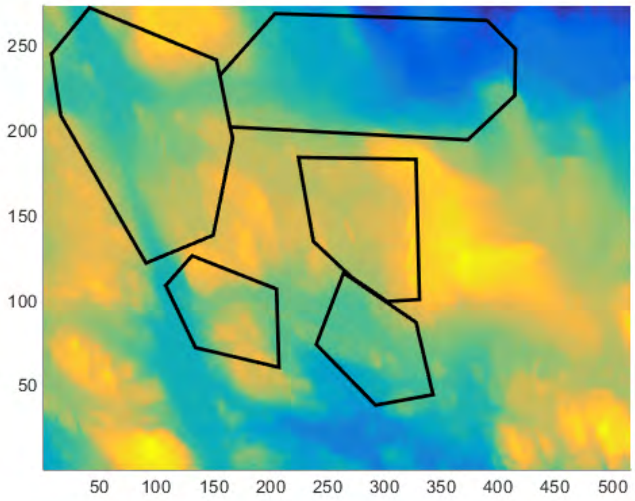


Figure 2.2: *The safari area with the different enclosures marked in black.*

2.2 Coordinate Systems

Several coordinate systems have been used in the thesis project. In this section they will all be presented. Each coordinate system is described below. Table 2.1 summarizes the different coordinate systems used in this thesis project. The coordinate systems are:

- **WGS84:** WGS84 (World Geodetic System 1984) is a global coordinate system commonly used in cartography and navigation [42]. The origin of the coordinate system is defined as the center of mass of the Earth. The z-axis is defined as the direction of the IERS (International Earth Rotation Service) Reference Pole. The x-axis is defined as the intersection of the IERS Reference Meridian and a plane which is going through the origin and normal of the z-axis. The y-axis is defined as the direction which fulfills the demands of a right-handed, orthogonal and Earth-centered coordinate system, following the rotation of the earth [42]. The WGS84 coordinates can be expressed in different ways, one of the more common ones is to express it in decimal form of latitude and longitude [30] and altitude expressed as meters above the sea level. This is also the form of WGS84 coordinates which have been used in this thesis project.

- **ENU:** ENU (East North Up) is the global coordinate system, where the base vectors represents the coordinates in east, north and upward vertical direction. The origin is defined as a specific position in the WGS84 frame. One should observe that the ENU system does not follow the curvature of the earth. The curvature of the earth is approximated as flat around the origin. This makes the ENU system a poor choice of coordinate system to use for large distances [30].

- **NED:** NED (North East Down) is a coordinate system widely used within aerospace. The origin is the same as for the ENU system and the base vectors are pointing in north, east and downward vertical direction respectively [30].

- **Local NED:** The local NED coordinate system has the same axes as the NED system but the origin is instead located at the center of the platform [30].

- **Body:** The Body coordinate system follows the platform. The origin is located in the center of the platform and the base vectors are pointing in the forward, right and down direction relative to the platform [30]. The roll, pitch and yaw angles are defined as the rotation angles from the NED to the Body coordinate system [30].

- **Camera:** The Camera coordinate system follows the camera located on the platform. It is defined in the same way as the Body coordinate system but relative to the camera instead of the platform. The origin is located at the focal point of the camera.

- **Image:** The Image coordinate system is the only coordinate system with only two base vectors. All images are described in this coordinate system. The origin is located at the top left corner and the base vectors are pointing in the right and down direction of the image.

Table 2.1: *Coordinate systems*

Denotation	**Base vectors**	**Directions**
ENU	$x_{ENU}, y_{ENU}, z_{ENU}$	east, north, up
NED	$x_{NED}, y_{NED}, z_{NED}$	north, east, down
Local NED	$x_{locNED}, y_{locNED}, z_{locNED}$	north, east, down
Body	x_b, y_b, z_b	forward, right, down
Camera	$x_{cam}, y_{cam}, z_{cam}$	forward, right, down
Image	x_{im}, y_{im}	right, down

2.3 Conversion between Coordinate Systems

In this section a description of how the conversion between different coordinate systems is presented. Where the conversion from one coordinate system to another is represented by a rotation, the inverse conversion is represented by the inverse rotation. A rotation from frame m to frame n can be expressed as a rotation matrix R_n^m. One important property of a rotation matrix is that it is always orthogonal, meaning that the inverse of that rotation is the same as transposing the matrix [30]. In this section X_m corresponds to the position in frame m, while (x_m, y_m, z_m) corresponds to the coordinates of each axis in frame m.

2.3.1 WGS84 and ENU

The origin for the ENU system is given as a specific position, (ϕ_0, λ_0), in the WGS84 system, where ϕ_0 is a latitude and λ_0 is a longitude. Here this position is chosen close to the test area at Kolmårdens djurpark in Sweden. The exact coordinates are

$$(\phi_0, \lambda_0) = (58.662064,\ 16.433337). \tag{2.1}$$

It is a position some hundred meters from the Kolmårdens djurpark. It was acquired from usage of QGIS [17] and other map tools.

As the decimal form of WGS84 is describing coordinates on an ellipsoid, the exact conversion to the ENU system is complicated. However, the area of interest in this thesis project is very small in comparison to the Earth making a linearization suitable [30]. The result of the linearization yields the following conversion from WGS84 to the ENU system

$$\begin{pmatrix} x_{ENU} \\ y_{ENU} \end{pmatrix} = \begin{pmatrix} k_\phi & k_\lambda \end{pmatrix} \begin{pmatrix} \phi - \phi_0 \\ \lambda - \lambda_0 \end{pmatrix} \tag{2.2}$$

where k_ϕ and k_λ are two constants describing the scaling when converting from degrees to meters. According to [47] the constants are found by

$$\begin{aligned} k_\phi &= 111132.92 - 559.82\cos(2\phi_0) + 1.175\cos(4\phi_0) - 0.0023\cos(6\phi_0) \\ k_\lambda &= 111412.84\cos(\phi_0) - 93.5\cos(3\phi_0) - 0.118\cos(5\phi_0). \end{aligned} \tag{2.3}$$

Figure 2.3 shows how the two coordinate systems are related to each other.

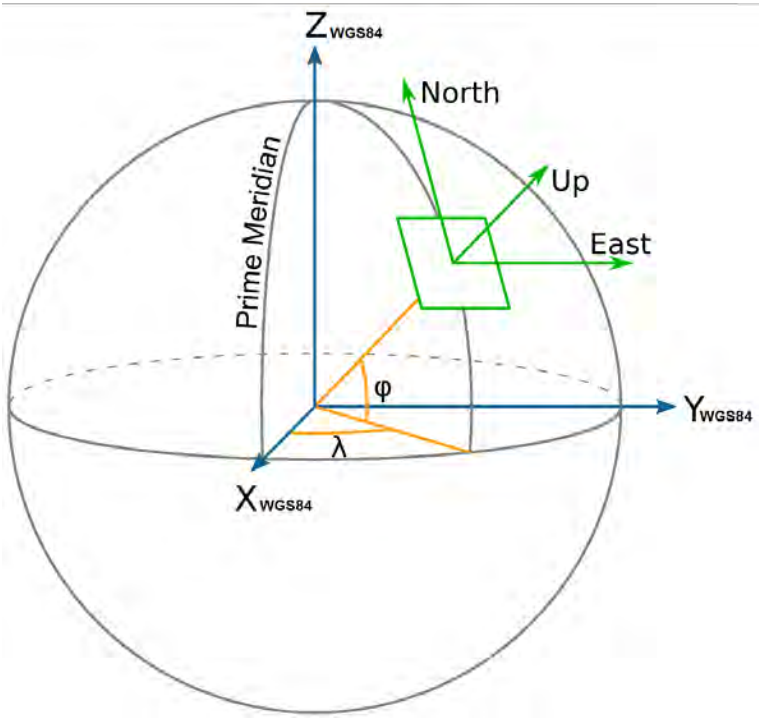


Figure 2.3: *Illustration of the WGS84 and the ENU systems.*
Inspired by image taken from `https://commons.wikimedia.org/wiki/File:ECEF_ENU_Longitude_Latitude_relationships.svg` the 5th of May 2016

2.3.2 ENU and NED

Both the ENU and NED coordinate systems are fixed in the global frame with the same origin, thus no translation is needed and the rotation between the systems is constant. The conversion between ENU coordinates and NED coordinates is

$$X_{NED} = \begin{pmatrix} x_{NED} \\ y_{NED} \\ z_{NED} \end{pmatrix} = R_{NED}^{ENU} X_{ENU} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} x_{ENU} \\ y_{ENU} \\ z_{ENU} \end{pmatrix}. \tag{2.4}$$

To get the inverse conversion the rotation matrix has to be transposed. Since R_{NED}^{ENU} is also symmetric, the same rotation matrix is used for the conversion between these systems in both ways.

2.3.3 NED and Local NED

The axes for the NED and Local NED coordinate systems are parallel, meaning no rotation is needed for the conversion. Instead just a translation is needed. The origin for the Local NED system is defined as the center of the platform while the origin for the NED system is defined as a specific position, often on the ground. The conversion from NED to local NED is defined as

$$X_{locNED} = X_{NED} - X_{NED}^{platform} \tag{2.5}$$

where $X_{NED}^{platform}$ is the position of the platform in the NED system [30]. The inverse conversion is given by

$$X_{NED} = X_{locNED} + X_{NED}^{platform}. \tag{2.6}$$

2.3.4 Local NED and Body

The origin for the Local NED system and the Body system is at the same position, the center of the platform. There is however, a rotation between these systems as the Body system always follows the rotation of the platform and the Local NED system is constant in the global frame. In aviation, Euler angles are commonly used to describe the conversion between these systems [30]. The rotation between the systems can be expressed as a series of rotations

$$R_b^{locNED} = R_b^2 R_2^1 R_1^{locNED} \tag{2.7}$$

where each rotation matrix R_j^i corresponds to a rotation from frame i to frame j. Frame 1 and 2 are here temporary and only used to simplify the conversion. For each rotation one of the Euler angles is considered. The Euler angles, which are called yaw, pitch, and roll, are here defined as:

- **Yaw, ψ:** The rotation around the z_{locNED} axis. After the rotation the axes have changed and are here called x_1, y_1, z_1.
- **Pitch, θ:** The rotation around the y_1-axis. After the rotation the axes have changed and are here called x_2, y_2, z_2.
- **Roll, ϕ:** The rotation around the x_2-axis. After the rotation the conversion to the body system is complete.

Each of the three sub-rotations in (2.7) are visualized in Figure 2.4 to 2.6. The resulting coordinate system after performing all three rotations is the Body system [30].

The total conversion can be written as:

$$X_b = R_b^{locNED} X_{locNED}, \tag{2.8}$$

where R_b^{locNED} is the rotation matrix and given by

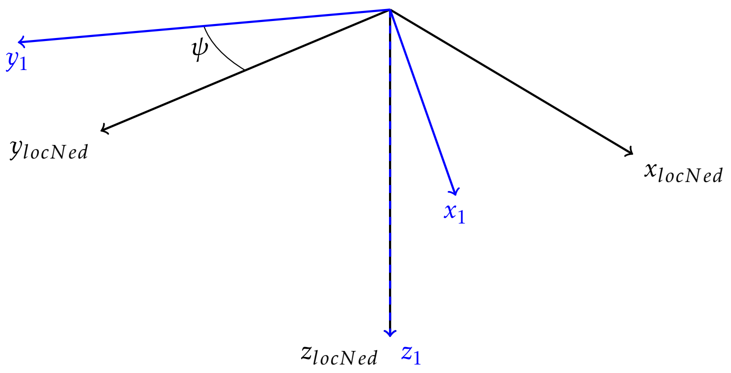


Figure 2.4: *Rotation around the z_{locNed} axis with angle yaw, ψ.*

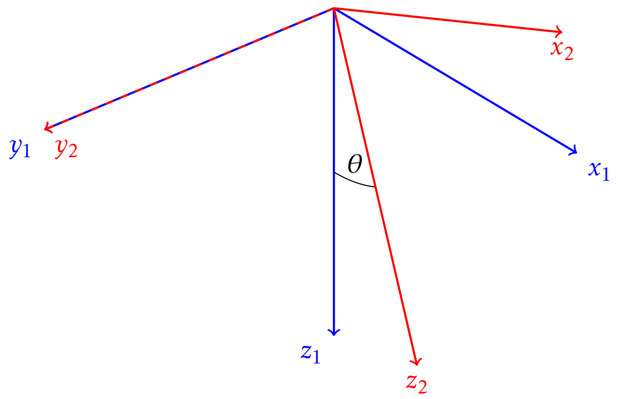


Figure 2.5: *Rotation around the y_1 axis with angle pitch, θ.*

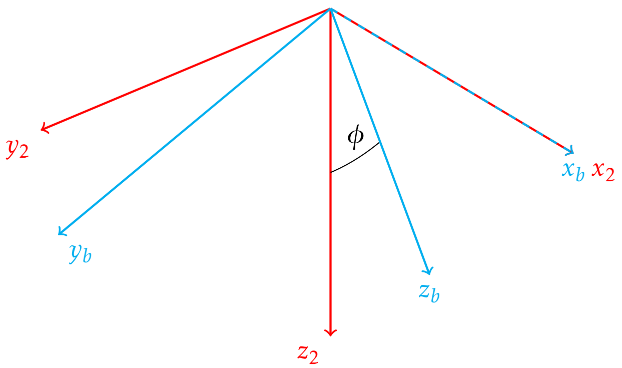


Figure 2.6: *Rotation around the x_2 axis with angle roll, ϕ.*

$$R_b^{locNED} = \begin{pmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{pmatrix} \tag{2.9}$$

where c corresponds to the *cosine* function and s to the *sine* function [30]. The inverse transformation is given by the transpose of this matrix. The order of rotation is important as the same angles used in a different sequence will result in a different rotation and thereby a different coordinate system than the Body system. Here the rotation order used is z-y-x.

2.3.5 Local NED and Camera

The transformation between these systems are essentially the same as for the Local NED and Body systems with the difference that the Camera system has its own Euler angles. The camera might be directed in another way than straight forward in the Body frame. Also there is a translation between the systems as the camera is not located at the center of the platform. It is easiest to calculate the translation before the rotation is done. This is because the offset is constant in the Body frame and that offset can easily be converted to an offset in the Local NED frame using the result described in Section 2.3.4. The total transformation looks as follows:

$$X_{cam} = R_{cam}^{locNED}\left(X_{locNED} - R_{locNED}^{b} X_b^{camOffset}\right) \tag{2.10}$$

where $X_b^{camOffset}$ is the offset of the camera from the center of the platform in the Body frame and R_{cam}^{locENU} is described by (2.9). The inverse transformation is described by

$$X_{locNED} = {R_{cam}^{locNED}}^T X_{cam} + R_{locNED}^{b} X_b^{camOffset}. \tag{2.11}$$

2.3.6 Camera and Image

To convert between the Camera and Image coordinate systems a model of the camera is necessary. We need to know how the camera depicts the surrounding world into the Image frame. The camera modeling can be performed with different levels of refinement. Normally, the conversion between these systems is only defined in one way, from the Camera to the Image system. In this case a three dimensional space is projected onto a two dimensional space. This means that a line of points in the Camera system will be projected only onto one position in the Image plane, if the line is in the radial direction from the optical center. The inverted conversion, from Image to Camera system is not uniquely defined because of this. A point in the Image frame corresponds to a vector in the Camera frame. This is a major problem as measurements of animals will be received in the Image plane and these need to be associated to already known animals. A way to work around this problem is to use the information given in the ground model, described in Section 2.1, and use an algorithm to project the resulting vector, from the camera model, onto the ground. The algorithm is described in Section 2.5.2. Two different camera models are here described, one simpler called the pinhole camera model and one more advanced which we will here call the advanced camera model. The camera models describe the transformation between an image position and its corresponding vector in the Camera frame.

Pinhole Camera Model

The pinhole camera model describes the transformation between a three dimensional coordinate in the Camera frame and its projected two dimensional coordinate in the Image frame, for an ideal pinhole camera. The model can be described as

$$\begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix} = \frac{1}{x_{cam}} \begin{pmatrix} 0 & f & 0 \\ 0 & 0 & f \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{pmatrix} \tag{2.12}$$

where x_{im} and y_{cam} as well as y_{im} and z_{cam} are parallel [40]. The third base vector in the camera system, x_{cam}, points forward relative to the camera. Figure 2.7 shows an illustration of a pinhole camera model. The model is a rough approximation of a camera where all pixels are considered to be square and the optical center is assumed to be known [40], also no skew and no distortions are considered. For the image coordinates this means that the origin will be at the center of the image. Normally for images the origin of the coordinates are considered to be at the top left corner of the image. The intrinsic camera model takes these approximations into account.

Intrinsic Camera Model

The intrinsic camera model does not assume square pixels, a known optical center and no skew for the camera. Based on the pinhole camera model, described in (2.12), to remove the assumption of known optical center an offset, (c_x, c_y), is added to change the origin of the Image frame to the top left corner. The modified model then becomes

$$\begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix} = \frac{1}{x_{cam}} \begin{pmatrix} c_x & f & 0 \\ c_y & 0 & f \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{pmatrix} \tag{2.13}$$

where (c_x, c_y) is called the principal point and describes the offset [40]. To remove the assumption of square pixels, the model has to incorporate the description of different focal lengths in each direction. The new model becomes

$$\begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix} = \frac{1}{x_{cam}} \begin{pmatrix} c_x & f_x & 0 \\ c_y & 0 & f_y \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{pmatrix} \tag{2.14}$$

where f_x and f_y are the focal lengths in each direction [40]. To remove the assumption of the skew for the pixels an additional parameter, α has to be included [40].

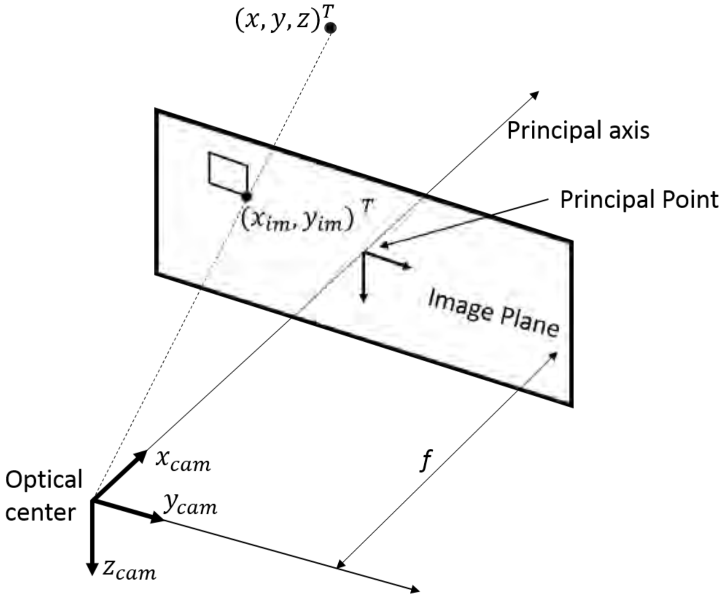


Figure 2.7: *Illustration of a pinhole camera model where a position in world coordinates are projected onto the image plane.*

The full intrinsic camera model then becomes

$$\begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix} = \frac{1}{x_{cam}} \begin{pmatrix} c_x & f_x & \alpha \\ c_y & 0 & f_y \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{pmatrix} \tag{2.15}$$

where α describes the skew of the pixels [40].

All lenses are not ideal but distort the image to some extent. The lens distortion can be described by

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \begin{pmatrix} x_n \\ y_n \end{pmatrix} + \begin{pmatrix} 2k_3 x_n y_n + k_4 \left(r^2 + 2x_n^2\right) \\ k_3 \left(r^2 + 2y_n^2\right) + 2k_4 x_n y_n \end{pmatrix} \tag{2.16}$$

where X_d are the distorted coordinates [35],

$$r^2 = x_n^2 + y_n^2$$

and X_n is the normalized image projection

$$X_n = \begin{pmatrix} x_n \\ y_n \end{pmatrix} = \frac{1}{x_{cam}} \begin{pmatrix} y_{cam} \\ z_{cam} \end{pmatrix}. \tag{2.17}$$

The first term in (2.16) describes the radial distortion while the second term describes the tangential distortion [35]. A common approximation is to only include the second order radial distortion, meaning that only k_1 is used and k_2 to k_5 are considered to be zero. This is a distortion model good for narrow FOV lenses [35]. The simplified distortion model thus becomes

$$X_d = \left(1 + k_1 r^2\right) X_n \tag{2.18}$$

There is no analytical inverse to this equation, but an approximate solution is given in [35]. The inverse mapping from distorted to undistorted coordinates is given by

$$X_n \approx \frac{X_d}{1 + k_1 \left(\frac{x_d^2 + y_d^2}{1 + k_1 \left(x_d^2 + y_d^2 \right)} \right)} \tag{2.19}$$

where X_d is the result from the inverse non-distorted camera model and X_n is the normalized image projection. X_n can easily be converted to a vector in the Camera frame according to

$$\begin{pmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{pmatrix} = \begin{pmatrix} 1 \\ x_n \\ y_n \end{pmatrix}$$

due to the definition of the Camera frame.

The result of using the distortion model in (2.18) and the intrinsic camera model in (2.15) is a transformation between a point in the Image plane and a line in the Camera frame, which considers radial distortions, non-square pixels, pixel skewing and an unknown optical center. The transformation from the Camera frame to the Image plane considering lens distortions becomes

$$\begin{pmatrix} x_n \\ y_n \end{pmatrix} = \frac{1}{x_{cam}} \begin{pmatrix} y_{cam} \\ z_{cam} \end{pmatrix} \tag{2.20a}$$

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = \left(1 + k_1 \left(x_n^2 + y_n^2\right)\right) \begin{pmatrix} x_n \\ y_n \end{pmatrix} \tag{2.20b}$$

$$\begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & \alpha & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_d \\ y_d \\ 1 \end{pmatrix} \tag{2.20c}$$

The transformation from Image coordinates to a vector in the Camera frame becomes

$$\begin{pmatrix} x_d \\ y_d \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & \alpha & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} x_{im} \\ y_{im} \\ 1 \end{pmatrix} \tag{2.21a}$$

$$\begin{pmatrix} x_n \\ y_n \end{pmatrix} \approx \frac{1}{1 + k_1 \left(\frac{x_d^2 + y_d^2}{1 + k_1 \left(x_d^2 + y_d^2 \right)} \right)} \begin{pmatrix} x_d \\ y_d \end{pmatrix} \tag{2.21b}$$

$$\begin{pmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{pmatrix} = \begin{pmatrix} 1 \\ x_n \\ y_n \end{pmatrix} \tag{2.21c}$$

2.4 Modeling of Thermal Camera

To estimate the camera parameters in the model described by (2.20) the *Camera Calibration Toolbox for Matlab* [28] has been used. This is a toolbox with the purpose of modeling cameras. The input to the toolbox is a set of pictures taken with the camera on a calibration chess board from different angles. Figure 2.8 shows two examples of chess boards used for calibrating cameras.

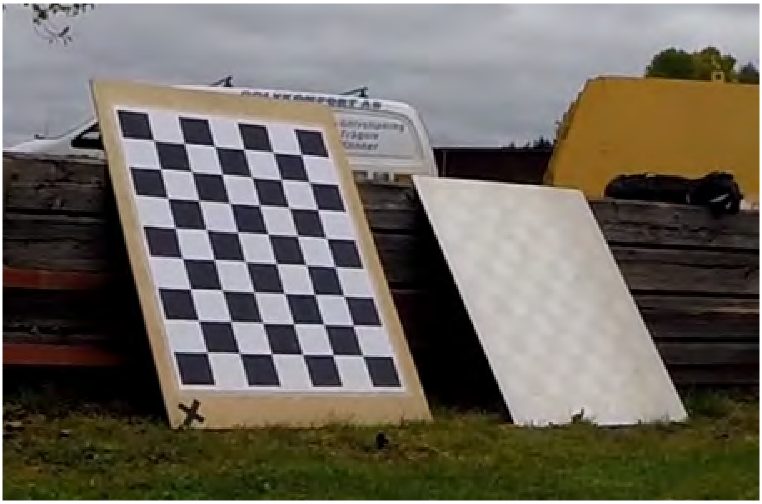


Figure 2.8: *Two chess boards used for camera calibration. The left one is used for EO cameras while the right one is used for thermal cameras.*

The important feature of the chess board is the high contrast between the squares. For a thermal camera this means that the squares must appear as differently tempered. An ordinary chess board placed in the sun will not have this behavior as heat will leak between the squares. Instead a chess board where every other square is seen as a mirror by the thermal camera is used. If the chess board is then placed such as the thermal camera is reflected to the sky, a clear high-contrast chess board will appear to the thermal camera. This can be seen in Figure 2.9.

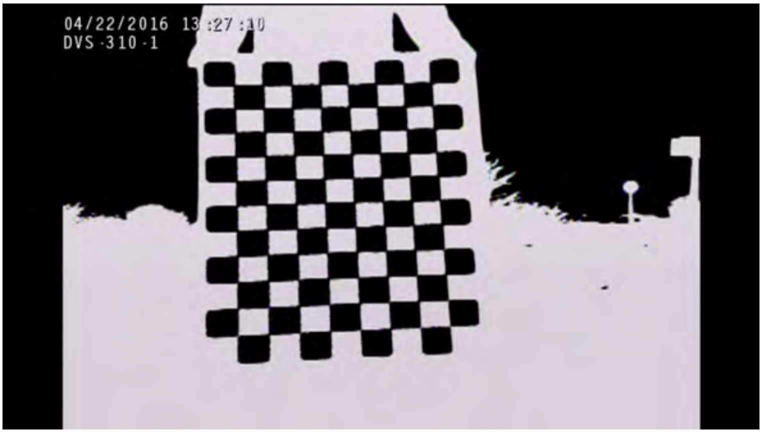


Figure 2.9: *The chess board used for modeling, as seen by the thermal camera.*

Given the set of pictures of the chess board taken from different angles, the toolbox calculates the camera parameters. The output parameters from the toolbox are $f_x, f_y, c_x, c_y, \alpha, k_{1-5}$. A good approximation is to disregard the tangential distortion described by k_3 and k_4 as well as the higher order (≥ 2) radial distortions described by k_2 and k_5. The resulting parameters from the modeling of the thermal camera is presented in Section 7.1.

2.5 Determination of Line of Sight

The LOS represents the footprint of the camera, i.e. the area on the ground that the camera is able to see. To be able to use cameras to position targets, the LOS of the camera has to be known. The LOS is also of use to determine the areas of the ground which have been viewed. This information can be used by a planning algorithm. The LOS depends on the position and the attitude for the camera as well as the camera parameters describing the camera lens. The way that the LOS is calculated here is to first determine the FOV and then check what is not occluded in the FOV. This is a common problem in computer graphics, so many approaches exist.

2.5.1 Field of View

This section describes one approach to determine the FOV which describes a volume which a camera is able to see without considering occlusions. For a camera, the FOV angles are often specified. Given a coordinate in the Camera frame, see

Section 2.2, the goal is to determine whether this is within the FOV for the camera. This is done by first projecting the point onto an Image plane where the FOV can easily be described. A position X_{cam} in the Camera frame is projected onto the image point $(u,\ v)^T$ according to the ideal perspective formula [46]

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{1}{x_{cam}} \begin{pmatrix} y_{cam} \\ z_{cam} \end{pmatrix}. \tag{2.22}$$

The FOV in this Image plane is described as an area according to

$$\mathcal{A} = \left\{ \begin{pmatrix} u \\ v \end{pmatrix} \middle| \, |\arctan u| \leq \frac{\alpha_u}{2}, |\arctan v| \leq \frac{\alpha_v}{2} \right\} \tag{2.23}$$

where (α_u, α_v) represent the FOV angles for the camera [46]. A point $(u, v)^T \in \mathcal{A}$ is considered to be in the FOV. If this does not hold, the point is outside of the FOV. Figure 2.10 shows the FOV for a camera. The FOV problem is cheap to solve and has a linear complexity $\mathcal{O}(n)$ where n is the number of grid points in the map.

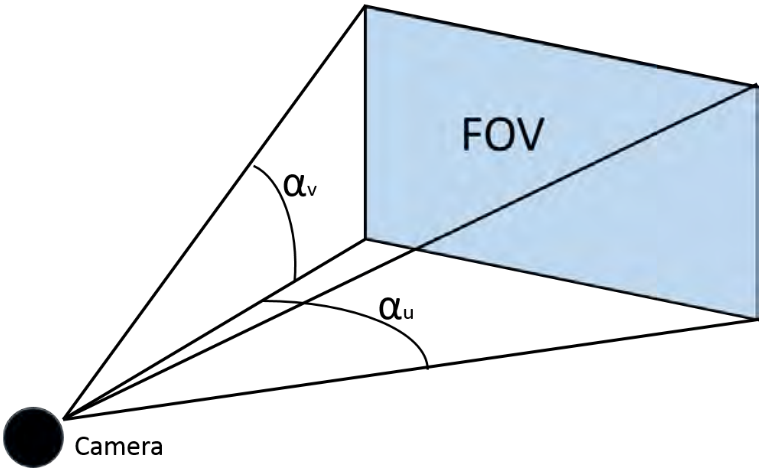


Figure 2.10: *The FOV for a camera with the FOV angles marked in the picture.*

2.5.2 Check for Occlusions

Once the FOV for the camera is determined it is necessary to check for occlusions within the FOV area. This is necessary since the ground is not flat. The check for occlusions utilizes a 3D model of the ground. It is more expensive, from a computational point of view, to check for occlusion of a point compared to check if a point is within the FOV. In this thesis project, two methods for solving the occlusion problem were tested. The first method is an implementation of the *Ray Casting* method [44], and the second method is based on *Bresenham's Line Algorithm* [39]. The two methods are described in the following sections.

Ray Casting

The first method to solve the occlusion problem is called *Ray Casting*. This method is commonly used in computer graphics which is easy to parallelize and thus is suitable to be performed in a GPU [24]. The idea of this method is to invert the image generation process. Instead of light coming into the camera, rays are sent out from the camera also known as *Ray Casting*. The first intersection of the rays with the map determines what the camera sees. To determine if a POI (Point Of Interest) is occluded, a ray between the POI and the camera is generated. If the first intersection of the ray traveling from the camera is somewhere else than at the POI, the POI is considered occluded. Figure 2.11 shows the principle of *Ray Casting*.

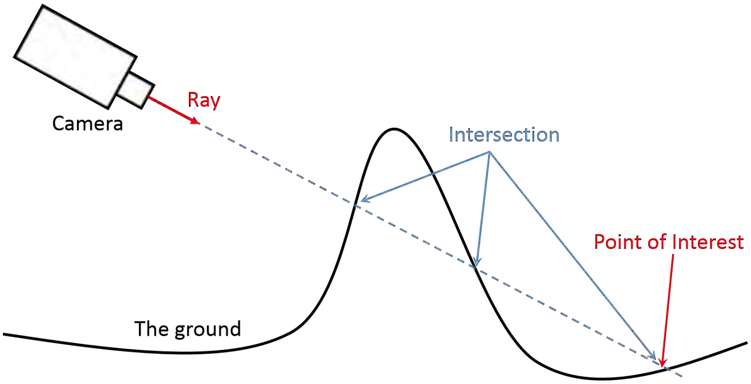


Figure 2.11: *The principle of the Ray Casting algorithm. Here a check for occlusion on the POI is being performed. There is an intersection before reaching the POI meaning that the POI is occluded.*

To check what part of the ground model that the camera is able to see, one ray is generated for each grid point $(x_{ENU}, y_{ENU})^T$ inside the FOV. The grid has a resolution of 2 meters since this is the resolution of the ground model. All the rays will have a slightly different heading but they all start at the same point, the position of the camera. A linear search algorithm perform a test which determines the position of the map that the ray first intersects with, by linearly increasing the length of the ray until it is below the ground level. Once the ray intersects with the ground that position is compared to the grid point, $(x_{ENU}, y_{ENU})^T$, that was tested. If the distance is too large the grid point is considered to be occluded.

The Ray Casting algorithm can also be used to project a vector onto the ground. This is used when an image coordinate needs to be converted to an ENU coordinate. The image coordinate is converted to a vector in the camera frame by taking the inverse of the camera matrix and then compensating for lens distortions, see Section 2.3.6. The camera vector can then easily be converted to a vector in the ENU frame by using the transformations presented in Section 2.3. The ENU vector is then projected onto the ground by the Ray Casting algorithm. The algorithm is summarized in Algorithm 1.

Algorithm 1: Ray Casting Algorithm

```
Data: Platform position and POI (Point Of Interest)
Result: Occluded
begin
    Create a ray starting at the platform and directed to the POI;
    Normalize the ray;
    while Ray is above ground level do
        Increase ray length;
    if Ray at POI then
        Occluded = False;
    else
        Occluded = True;
```

Since the ground model is known, one optimization can be done for the Ray Casting algorithm. By finding the maximum altitude within the FOV, the linear search for an intersection can start at this altitude. This is especially more efficient, compared to performing the linear search from the platform, when the platform has a high altitude relative to the surrounding ground.

Extended Bresenham's Line Algorithm

The *Extended Bresenham's Line Algorithm* compares angles instead of casting out rays to determine if a point is occluded for the camera. This is done by first determining the grid points for which the angle should be computed. The angle of interest is the angle between the camera and the grid point calculated according to

$$\theta_{i,j} = \arctan\left(\frac{z_{ENU}^{platform} - z_{ENU}^{Map(i,j)}}{\sqrt{\left(x_{ENU}^{platform} - x_{ENU}^{Map(i,j)}\right)^2 + \left(y_{ENU}^{platform} - y_{ENU}^{Map(i,j)}\right)^2}}\right) \tag{2.24}$$

where (i, j) is the position of the POI in the map. The angle for the POI is denoted θ_0. To determine which points in the map that the θ angle needs to be computed for, a vector $(\delta x_{ENU}, \ \delta y_{ENU})^T$ is created between the POI and the camera position projected onto the ground plane. Figure 2.12 shows how the grid points are chosen. If $\delta x_{ENU} \leq \delta y_{ENU}$, the vector is scaled to have length 1 in the y_{ENU}-direction, otherwise it is scaled to have length 1 in the x_{ENU}-direction [39]. A number of steps, corresponding to the distance in the furthest direction is then performed. For each step, the grid point, $(x_{ENU}, \ y_{ENU})^T$, closest to the resulting position is selected. For all selected grid points the angle will now be compared. If $\exists \theta_{i,j}$ such that $\theta_{i,j} \leq \theta_0$, the POI is considered to be occluded [39].

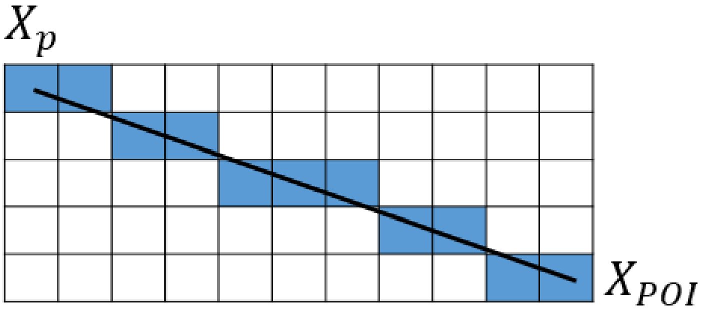


Figure 2.12: *The principle of how the grid points are chosen for which the θ angle is computed. A line is drawn between the projected positions of the platform and the POI. Given this line the grid points are chosen according to Bresenham's Line Algorithm.*

2.6 Sensors

This section shortly describes the sensors necessary to provide position and attitude estimates.

2.6.1 GPS

The GPS is necessary to provide a position estimate for the platform used in the thesis project. The GPS receiver estimates the position by using triangulation in space from GPS satellites [9]. The satellites transmit their location and time continuously. Through a TOA (Time of Arrival) framework [34] the location and the time of the GPS receiver can be calculated. The GPS receiver must have at least 4 satellites in view to be able to calculate its position. The output of the GPS receiver is a position given in latitude and longitude. The altitude is also given, but with a larger uncertainty.

2.6.2 IMU

An IMU (Internal Measurement Unit) consists of three different sensors, an accelerometer, a gyro and a magnetometer [49]. By using sensor fusion, an accurate estimate of the attitude can be obtained. This is a major component for inertial navigation systems in aircrafts, spacecraft and also in mobile phones. The gyro measures the angular rate, the accelerometer measures forces on the IMU and the magnetometer measures the magnetic field. All sensors gives data in three-dimensions.

3 Detection

To be able to track objects, detections of the objects are necessary. Detections can be retrieved from different sensors and in different ways. In this thesis project, a thermal camera is used for detection. The image processing is done in OpenCV [13]. This chapter describes how the thermal images from the camera are retrieved and processed to extract detections.

3.1 Thermal Camera Parameters

The thermal measurements from the thermal camera are propagated through a linear transfer function and mapped onto a color palette. The thermal camera is presented to a viewer where one color, such as white, corresponds to a warmer area while another color, such as black, corresponds to a colder area. Many color palettes exist that the measurements can be mapped to. The linear transfer function can be seen as a window with two parameters, brightness and contrast, to change the location and width of the window. Temperatures lower than the start of the window is mapped onto the color corresponding to the lowest temperature and temperatures higher than the end of the window are mapped onto the color corresponding to the highest temperature. The temperatures inside the window are linearly mapped onto the color palette.

To get a good thermal image, these parameters have to be adjusted in the field. The same parameters will give different images in different weather conditions, due to different thermal environments. The brightness parameter sets the location of the window, while the contrast parameter changes the width of the window. Increasing the contrast will give a more detailed image in a smaller range around the mean. Since the animals typically are warmer than the surrounding environment, the mean should be set between the ground and animal temperature, and the contrast should be set relatively high. This should yield a rather noisy image where the animals should be clearly distinguishable from the background. It is important to not set the gain too high, as this will reduce the width of the temperatures mapped onto the color palette and thereby destroy information.

3.2 Image Processing

The purpose of the image processing is to extract as much relevant information as possible from an image. In this thesis project, the images are processed frame by frame, so no information from previous frames are used in the processing of the current frame. There are several caveats when only using thermal images for detection. First of all, the temperature of the animals has to differ from the surrounding environment. In this thesis project, the animals are considered to be warmer than the background. This is however not always true as the sun heats up the ground during the day. Another problem is that no size information can be extracted solely from the thermal image. Since the pixel size of an animal depends on the distance from the camera, small objects close to the camera could be misinterpreted as animals further away.

The second problem could be solved through the use of position and attitude data in the image processing algorithm together with the ground model. A detection could be ray casted to get the true ground position, which gives a good lead of the size of the detected target. If the difference is too large, the detection should be discarded.

The main problem for the image processing algorithm is noise. A received image has to be filtered to remove this noise from the background. An example of noise could be a warm stone the size of an animal. The heat signature from a warm stone could be similar to the signature of a rhino, making it difficult to differentiate. One approach which may work is to use form detection, this is however not investigated.

The following methods, thermal enhancement and background subtraction, are used to emphasize objects warmer than the background. The methods thresholding and adaptive thresholding are used to find the animals in the improved image. Contour detection are used to extract the information from the threshold methods to detect the targets.

3.2.1 Conversion to Grayscale

Before the more advanced image processing methods are applied, the thermal image is converted to grayscale by a function in the OpenCV library. The conversion is done according to

$$I_{gray}(x,y) = 0.299 I_R(x,y) + 0.587 I_G(x,y) + 0.114 I_B(x,y) \tag{3.1}$$

where I_R, I_G, and I_B, are the color components of the image, and I_{gray} is the grayscale converted image [3]. The reason for the grayscale conversion is that we are only interested in intensity, not color. After the conversion the intensity of each pixel is described with a scalar value between 0 and 255.

3.2.2 Thermal Enhancement

It is assumed that the animals are warmer than the surrounding environment. Thus, the most relevant areas of the image are the warm ones. Thermal enhancement [37] is a method intended to emphasize these regions by increasing the contrast according to

$$I_{enh}(x,y) = 2I(x,y) - \max(I(x,y)) \tag{3.2}$$

where I_{enh} is the resulting enhanced image. Figure 3.1 shows an image before and after the thermal enhancement.

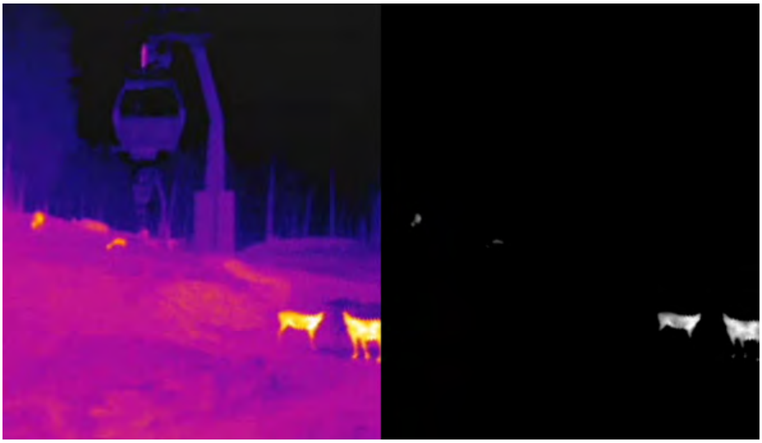


Figure 3.1: *A thermal enhanced image.*

3.2.3 Background Subtraction

Another method to emphasize the hotter areas is called background subtraction, which is similar to the thermal enhancement, see Section 3.2.2. Background subtraction subtracts each pixel with the mean value of the image according to

$$I_{BS}(x,y) = I(x,y) - \text{mean}(I(x,y)) \tag{3.3}$$

where $\text{mean}(I(x,y))$ is the average intensity of the image. In Figure 3.2 an example is shown.

3.2.4 Thresholding

The extraction of interesting regions in an image is often done by thresholding. The basic idea of thresholding is that every pixel with intensity below the threshold level is set to black, and every pixel above the threshold is set to white. This can be seen in figure Figure 3.3. A problem with a fixed threshold is that it is sensitive to changes in the background. If the background intensity increases, the whole image could be above the threshold, and thus result in a completely white image.

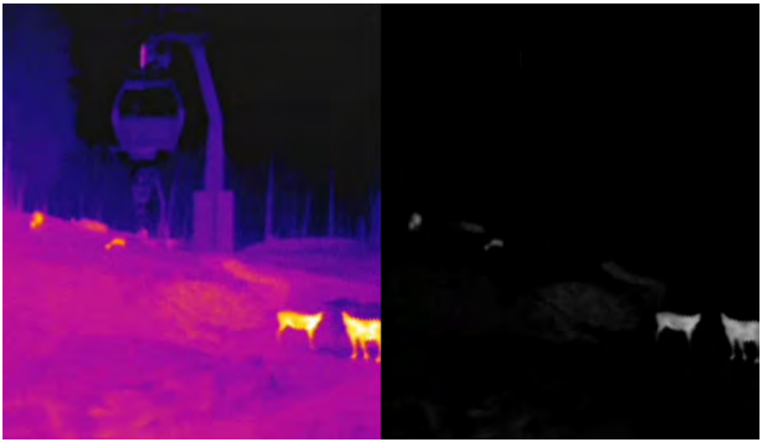


Figure 3.2: *A background subtracted image.*

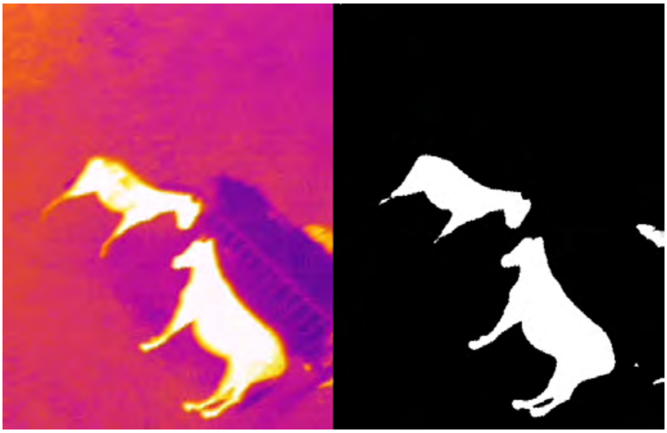


Figure 3.3: *Thresholding of a thermal image. Pixels with an intensity over the threshold level is set to white, while pixels with an intensity below the threshold level is set to black.*

3.2.5 Adaptive Thresholding

Adaptive thresholding is similar to thresholding, but instead calculates the threshold value depending on the intensities in a region around the pixel of interest. The form and size of the region could be set to different values. If a smaller region is used, only the contours of the objects will be captured since the threshold level will be adjusted to a higher level in the middle of larger objects. Adaptive thresholding handles the problem of using a constant threshold level, since a change of intensity in the background will change the threshold level. Although the target must still be warmer than the background for this method to work.

3.2.6 Contour Detection

When the thresholding is done, the resulting image contains only black or white pixels. The thresholded image is then run through a contour detection, where a contour is retrieved from consecutive areas with white pixels. Every contour with sufficient size is seen as a detection. The last step is to check the positions of the detections, and remove detections too close to each other. These should be considered to be from the same target.

4 Association and Tracking

This chapter describes the association and tracking problem and also the methodology to counter these. First a brief introduction to filter theory is given, then the tracking problem is described and a solution proposed. When multiple targets exist, as in this thesis project, association is a part of the problem. The chapter ends with a discussion and proposed solution to the association problem.

4.1 Filter theory

Tracking is an extension of localization, where the time-varying position of a target is estimated. There are several different tracking filters, and for this thesis project the particle filter was considered. The reason for that is the negative information which can easily be fused into the filter. In a particle filter negative measurements (measurements without any detections) are easily fused into the filter, see section 4.2.2.

4.1.1 Measurement Model

The measurement equation describes how the measurements are acquired. In the general case, the nonlinear measurement model is given by

$$y = h(x, e) \tag{4.1}$$

where y is the measurement, x is the unknown state, and e is noise. A special case of (4.1) is given by

$$y = h(x) + e \tag{4.2}$$

where the measurement is thought to be acquired as a nonlinear function of the state with the addition of noise. For an easy implementation e is in this thesis project seen as Gaussian noise modeled as

$$e \sim \mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}\sigma^2 & 0\\0 & \sigma^2\end{pmatrix}\right) \tag{4.3}$$

where σ is a design parameter describing the standard deviation. As can be seen in (4.3), the measurement noise is considered to be uncorrelated.

In this thesis project, the sensor is a camera. The measurements are received in the image coordinate system and the state space is given by ENU coordinates. The measurement model is thus a transformation from the ENU to the image system, so h will thus be a transformation described by (2.4), (2.5), (2.10) and (2.20).

4.1.2 Motion Model

The motion model aims to describe how the target moves through updating the state vector. The position of a car, for example, could be adequately described by a constant position (CP) model - if it is parked. In contrary, this model would not be very good if the car is moving on a highway. In this case the assumption of constant velocity (CV) is more appropriate.

The general motion model is

$$x_{k+1} = f(x_k, w_k) \tag{4.4}$$

where x_k is the state and w_k is process noise which is assumed to be independent and identically distributed.

The following equations are examples of the CP and CV models in the discrete two dimensional case. In this case the CP model is given by

$$x_{k+1} = \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix} x_k + \begin{pmatrix}T & 0\\0 & T\end{pmatrix} w_k \tag{4.5}$$

where x_k is the state, containing the two dimensional position, w_k is the process noise and T is the sample time. The second term could be seen as a velocity input, which is integrated to get the change of position. The CP model can be referred to as *random walk* if the process noise has zero mean. The CV model is given by

$$x_{k+1} = \begin{pmatrix}1 & 0 & T & 0\\0 & 1 & 0 & T\\0 & 0 & 1 & 0\\0 & 0 & 0 & 1\end{pmatrix} x_k + \begin{pmatrix}0 & 0\\0 & 0\\T & 0\\0 & T\end{pmatrix} w_k \tag{4.6}$$

where the state vector is

$$x_k = \begin{pmatrix}p_k^x\\p_k^y\\v_k^x\\v_k^y\end{pmatrix}. \tag{4.7}$$

The position and velocity, at time k, in each direction is given by p_k^x, p_k^y and v_k^x, v_k^y respectively. In (4.6) the process noise w_k can be seen as an acceleration input, which will affect the velocity directly, and affect the position as a double integral of the noise.

Selection of Motion Model

In this thesis project, the targets to be tracked are animals, so the movements are not as predictable as the movements of a car. Which model to use, also depends on the time horizon, and how long periods of time the targets are out of sight. In a short perspective, the constant velocity model is more accurate, since it is reasonable to guess that an animal moving in a direction at time k will continue to move in that direction at time $k + 1$. The problem in this thesis project is that the sensor scans large areas, so targets will be out of sight for long periods of time. Information about the animal movements degenerates rapidly when the animal is out of sight, so for this case the random walk model should be more relevant than the constant velocity model. Also the goal for the thesis project should be considered, and since it is not to have a high precision tracking, a random walk model could also be used for the animals in sight. The implementation becomes simpler and the information loss is not a problem in the scope of this project. The model is defined according to (4.5) where the process noise w_k is modeled as

$$w \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} Q & 0 \\ 0 & Q \end{pmatrix}\right) \tag{4.8}$$

where Q is a design parameter reflecting how much a typical animal moves.

4.2 The Particle Filter

The particle filter is a relatively new approach to non-linear filtering beginning to rise in popularity since a seminal paper was published in 1993 [34]. It is a computationally complex, but intuitive, algorithm for filtering. It is similar to the point mass filter [34], but instead of gridding up the state space deterministically, the particle filter uses an adaptive stochastic grid that will use the grid points (particles) more effectively. [33] A particle filter is suitable when the state space is of a low dimensionality. The performance is still good in a two or three dimensional state space, but in state spaces of higher dimensionality, the particle representation soon gets too sparse. As the dimension of the state increases, the number of particles to maintain the particle density increases according to $\mathcal{O}\left(c^N\right)$, where c is a constant representing particle density and N is the number of dimensions. Though particle filters could be used in many different areas, the following theory will focus on the case when the goal is to estimate the position of a moving target.

4.2.1 Overview

In the particle filter, the possible states are represented by particles. The particles are operated on in four different steps:

1. Time update
2. Measurement update
3. Estimation
4. Resampling

These particles are time updated according to a motion model, describing the motion of the moving target. The time update is a prediction of the future, and if the model is perfect, no measurements are needed. However, in reality this is not the case. The prediction has to be evaluated by the measurement update. Given a measurement, all particles are assigned new weights according to the probability for the particle. A weighted mean and covariance can then be calculated to give a state estimate. A crucial step in the particle filter is the resampling step. In the resampling step, a new set of particles is generated from the old set. Here the efficiency is increased, since the particle density is increased close to more probable states. But the resampling step also removes information, so it should not be performed too often. A visualization of the particle filter can be seen in Figure 4.1.

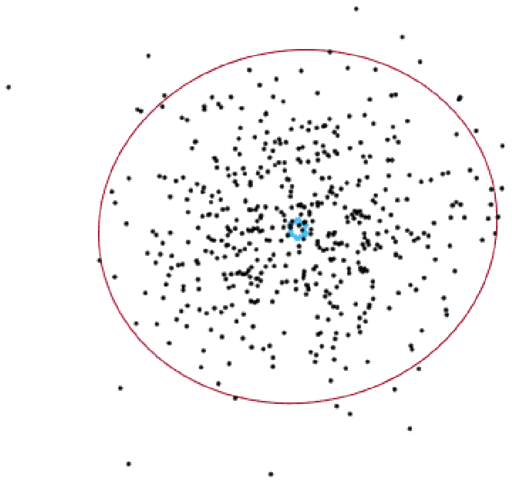


Figure 4.1: *A particle filter with 500 particles. The true target position can be seen in the middle. The red ellipse is a 95% confidence interval for the state estimation.*

4.2.2 Measurement Update

The measurement update is the addition of information. When a measurement is received and associated to a target, the uncertainty of the position of the target decreases. For each particle, x^i, $i \in \{1 \dots N\}$, update the weights according to

$$w^i_{k|k} = \frac{1}{c_k} w^i_{k|k-1} p(\mathbf{y_k}|x^i_k) \tag{4.9}$$

in [34] where k is the time index, and $\mathbf{y_k}$ is a vector of measurements associated to the track. The normalization weight c_k is given by

$$c_k = \sum_{i=1}^{N} w^i_{k|k-1} p(\mathbf{y_k}|x^i_k) \tag{4.10}$$

The distribution $p(\mathbf{y_k}|x^i_k)$ is the probability to get the measurements $\mathbf{y_k}$ given particle x^i_k, at time index k. This is given by

$$p(\mathbf{y_k}|x^i_k) = \prod_{m=1}^{M} p(\mathbf{y^m_k}|x^i_k) \propto \prod_{m=1}^{M} \exp\left\{-\frac{1}{2}\frac{\|\mathbf{y^m_k} - h(x^i_k)\|^2}{\sigma^2}\right\} \tag{4.11}$$

where h is the measurement equation described in Section 4.1.1, and m is an index for the associated measurements. Equation (4.11) propagates the particles to the image system and adjusts the weights with respect to the probability for the measurement given a particle.

Combining (4.9) and (4.10), the resulting weights will be normalized and lie between 0 and 1. Particles more probable, given the measurement and prediction, will have higher weights. The sum of all weights will be equal to 1.

Negative Measurement Update

When no measurement has been associated to a track, information to the track can still be gained. If the track has particles in the FOV, meaning that they should be visible to the camera, the weight of those particles should be decreased. It is less likely that the particles represent the true state of the target that is tracked. By this operation, the uncertainty of the target position is lowered even though no detection occurred. This is a straightforward advantage with the particle filter that would be more difficult to achieve in, for example an EKF which only saves the estimated state and covariance.

4.2.3 Estimation

The position and covariance of a track is estimated through

$$\hat{x}_k = \sum_{i=1}^{N} w^i_{k|k} x^i_k \tag{4.12}$$

and

$$\hat{P} = \sum_{i=1}^{N} w^i_{k|k}(\hat{x}_k - x^i_k)(\hat{x}_k - x^i_k)^T \tag{4.13}$$

where $\hat{x}_k$ is the weighted mean estimate of the position, and $\hat{P}$ is the weighted covariance estimate of the covariance of the track.

4.2.4 Resampling

The resampling step is important for the particle filter to avoid diverging. This is the feedback from observations to predictions, and without it the filter would suffer from sample depletion. Without resampling, the particles would diverge and all relative weights would go to 0 except one that would go to 1. In the resampling step, a new set of particles are sampled from the whole set of particles. N samples are taken with replacement, where the probability to take a sample i is equal to its weight $w^i_{k|k}$. After the resampling step, the weights of all particles are set to $1/N$.

In the standard algorithm of the particle filter, SIR (Sampling Importance Resampling) is used, where resampling is performed each iteration. Since resampling destroys information, it could be better to not sample each time. An alternative is then to use SIS (Sampling Importance Sampling), where the resampling is done when the depletion has grown too large.

Effective Number of Samples

As an indicator of the degree of depletion, the effective number of samples is used. Equation (4.14) provides an approximation from [34] of this

$$\hat{N}_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} (w^i_{k|k})^2}. \tag{4.14}$$

With the effective number of samples as a measure of sample depletion, a resampling condition can be defined as $\hat{N}_{\text{eff}} \leq N_{\text{th}}$, where the threshold could be chosen as $N_{\text{th}} = 2N/3$.

4.2.5 Time Update

The time update generates predictions for all particles and updates all weights according to a proposal distribution, $q\left(x^i_{k+1}|x^i_k, y_{k+1}\right)$. The proposal distribution is a probability function proposing the next state of the particle. The simplest and most common selection of proposal distribution is referred to as prior sampling, where the dynamic model itself is used [34]. This leads to the proposal distribution

$$q(x_{k+1}|x^i_k, y_{k+1}) = p\left(x_{k+1}|x^i_k\right) \tag{4.15}$$

where $p\left(x_{k+1}|x_k^i\right)$ is given by (4.5). The time update is performed through generating predictions for all particles according to

$$x_{k+1}^i \sim q(x_{k+1}|x_k^i, y_{k+1}) \tag{4.16}$$

The weights must also be adjusted with respect to the proposal distribution. This is done by

$$w_{k+1|k}^i = w_{k|k}^i \frac{p\left(x_{k+1}^i|x_k^i\right)}{q\left(x_{k+1}^i|x_k^i, y_{k+1}\right)} \tag{4.17}$$

where we can see that the use of the dynamic model as proposal distribution, makes the weight update trivial. Combining (4.15) with (4.17) the weight update becomes

$$w_{k+1|k}^i = w_{k|k}^i \tag{4.18}$$

4.3 Association

For the case when multiple targets exist, the measurements have to be associated to the correct filter [27]. A measurement could originate from a target not previously observed. In this case a new track should be created. These decisions are handled by an association algorithm. The algorithm should, given an arbitrary number of tracks and an arbitrary number of measurements, associate each measurement to the corresponding filter or create new filters for unassociated measurements.

4.3.1 Nearest Neighbor

The nearest neighbor algorithm is one of the simpler algorithms that can be used for association. This algorithm normally associates a track to the nearest measurement and for the unassociated measurements new tracks are initialized. In this thesis project, this algorithm has been slightly modified. This has been done to decrease the amount of tracks for groups of animals. In the modified version, for every measurement, the closest track is associated if this is within the gate. This means that one track can have several measurements associated if these lie closely to each other. Different measures of distance can be used, whereas two examples are the straight line distance and the *Mahalanobis distance*. In this thesis project, the Mahalanobis distance is used since the uncertainty is considered by this measure. To decide which measurement is most probable, the Mahalanobis distance between the track and each measurement is calculated according to (4.19). If it is larger than a value corresponding to the α% confidence interval, where α is a design parameter, the measurement and track are considered to be from different targets.

Mahalanobis Distance

The Mahalanobis distance is a probability measure, measuring the distance between two objects according to

$$D_M = \sqrt{\left(y_k - h(\hat{x}_{k|k-1})\right)^T S_{k|k-1}^{-1} \left(y_k - h(\hat{x}_{k|k-1})\right)} \tag{4.19}$$

where S is the innovation covariance matrix for the track [34]. The advantage with the Mahalanobis distance, is that it takes the uncertainty into account, not just the straight line distance. This can be seen in Figure 4.2 where the measurement lies closer to track 2 in terms of space distance, but closer to track 1 in terms of Mahalanobis distance. This is due to the different uncertainties of the tracks were track 1 is more uncertain than track 2.

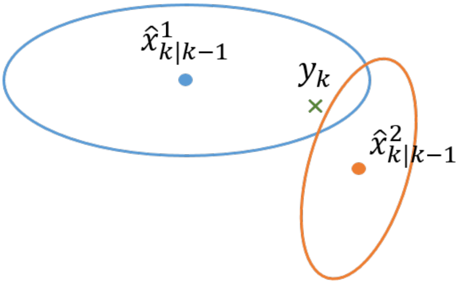


Figure 4.2: An example where the use of Mahalanobis distance would give a better result than the Euclidean distance. The ellipses represent confidence intervals for the two tracks. Here the measurement y_k would be associated to the track with state estimate $\hat{x}^1_{k|k-1}$.

Gating

To not associate measurements which are very unlikely to be a detection of a track, a maximum Mahalanobis distance is set as a limit for the association. This is called gating [27]. The gate corresponds to a confidence interval of a certain probability α, which is a design parameter. For a measurement to lay inside the gate the following must hold

$$D_M < \gamma_\alpha \tag{4.20}$$

where D_M is χ^2_2 distributed and γ_α is the threshold. The threshold is the largest value to lie in an α % confidence interval for a χ^2 distribution with two DOF (Degrees Of Freedom). Two DOF are used since the measurements are given in two dimensions.

5 Planning

This chapter handles theories and approaches for the planning of both the UAV and the gimbal movement. The goal of the planning is to find all animals in the area as fast as possible. This is done by making the UAV move around the area while trying to locate the targets. The chapter starts with a description of the planning problem and then continues with two approaches to solve this problem. One approach is to make decisions online as targets are detected, and the other is to plan a trajectory for the UAV offline i.e., before take-off.

5.1 The Planning Problem

The planning problem in this project is to find all targets in the area. The area is described by a map, see Section 2.1. The area of the map consists of several enclosures with animals. The solution of the planning problem defines the route of the UAV and how the camera should be angled. The prior information is a 3D map of the area which also contains information of the location of the enclosures. Furthermore, the number of animals within each enclosure is known. Henceforth, the animals are referred to as targets and the UAV as the platform.

The FOV of the thermal camera is limited, making it impossible to observe the entire area at one time instant. Thus, all targets can probably not be observed simultaneously which means that the platform has to move around the area to be able to observe all targets. As the covariance of the position of a known target grows with time, each target has to be observed frequently to not be considered lost. Here, a problem arises with the area. If the area is too big, it will be impossible for the platform to search the whole area without losing known targets because of a too high covariance. The speed of the platform is limited making it difficult to reach all targets within a limited time when the operating area grows.

The planning problem should yield a planned route for the UAV given information about the enclosures and the map. However, it is not only the position of the platform which can be controlled but also the attitude of the camera. This is done by controlling the gimbal, described in Section 6.1.9, which the camera is mounted on. By also controlling the attitude of the camera, additional DOF are added to the solution making it harder to find an optimum.

For the platform planning, it is only of interest to plan the position, X_{ENU}, not the attitude. This due to the fact that the gimbal decouples the attitude of the camera from the attitude of the platform. The gimbal is faster than the platform, making it possible to cancel rotations. For the gimbal planning it is the attitude which is of interest.

5.2 The Offline Planning Approach

The offline planning approach is more basic than the online planning approach as no information gained during the flight is considered for the planning. Instead everything is planned beforehand. This section covers the planning of the route for the platform. To plan the route, a pattern based approach has been used. In this approach the idea is to let the platform move in a pattern across the map. The prior information of the enclosures are considered such that one enclosure at a time is being searched. However, the prior information of the outlook of the ground is not considered. When the planning of the route for the platform is performed offline, a good approach is to move across the area in a systematic way, also called a pattern based planning. The problem then becomes how to choose the pattern for which the platform should move across the map. The pattern should have a low probability of missing targets which may move and the area should be covered in a time-efficient way.

One strength of the pattern based planning algorithm, given that the pattern is chosen in a carefully considered way, is that the risk of missing moving targets is reduced since the same area is revisited within a limited time-span. One pattern that has this characteristics, here called the lawn mower pattern, can be seen in Figure 5.1. This is a basic and intuitive pattern which guarantees that a convex area[1] will be completely covered, where occlusions is not taken into consideration.

The lawn mower pattern is essentially letting the platform move across the map from side to side. One design parameter is the distance between each track going up and down. By increasing the distance, the execution time to search the map decreases, but also the chance of missing moving animals. Also, if this parameter is chosen too big the whole map will not be seen from the platform as the FOV is limited for the camera. This parameter has to be chosen carefully to ensure complete coverage of the map.

The area at Kolmården consists of several enclosures. Here the lawn mower pattern has been used according to Figure 5.2. The platform searches one enclo-

[1] A convex area is defined as an area in which no line segment between any two points on the border of the area will go outside the area. Another way of defining a convex area is that no internal angles are greater than 180 degrees.

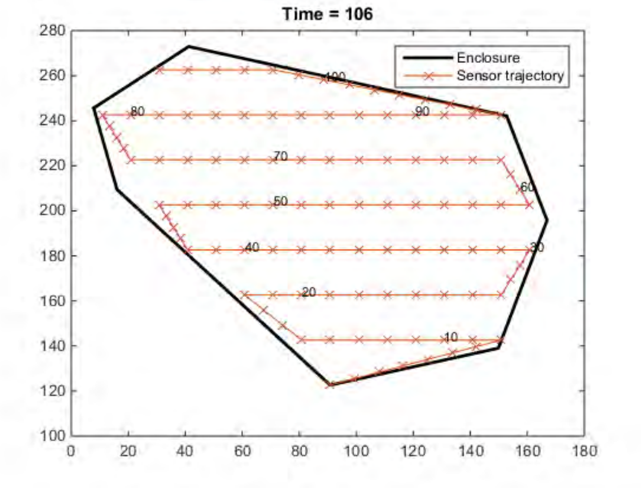


Figure 5.1: An illustration of a lawn mower pattern where the platform starts at the bottom of the enclosure and systematically searches the full enclosure.

sure at a time. As soon as one enclosure has been searched it moves on to the next. The search order of the enclosures can be customized as well as the search direction. As soon as all enclosures have been searched the platform is done searching the area.

As we will see later, the greatest weakness of information based planning in Section 5.3.1, is also the greatest strength of pattern based planning; computational complexity. For the pattern based planning algorithm, occlusions or the 3D-map are not taken into consideration. Instead it is just how the map is limited that is taken into account when planning the trajectory. This reduces the computational complexity significantly as the information does not have to be calculated in the planning step.

The offline planning does not have to make use of patterns, instead the route of the platform can be customized to fulfill certain demands. This may be highly desirable for project Ngulia, where the UAV should visit certain parts of the national park more frequently than others. Interesting locations might be the border of the national park to search for poachers or waterholes which the animals visit often.

5.3 The Online Planning Approach

The online approach of the planning problem makes use of the prior information of the enclosures. We divide the planning area into subareas where each subarea is representing an enclosure. Instead of trying to track all targets within the area the problem becomes to track all targets within an enclosure. When all targets

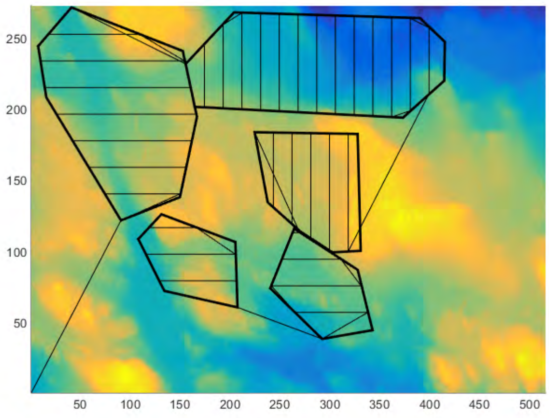


Figure 5.2: *An illustration of a lawn mower pattern for multiple enclosures. Here the area in Kolmården is being searched.*

within an enclosure has been positioned, that enclosure is considered scanned and the platform continues to the next enclosure.

As the planning area is divided into subareas of enclosures, the complexity of the planning problem is reduced. The problem is still essentially the same; to find and track all targets within an area. When the platform is entering a new enclosure, no targets are known which means that the platform has to search for new targets. As the platform moves around the enclosure, more targets will be found. These targets will however not always be in the FOV, thus the covariance of these targets will increase with time. If the covariance of a target increases beyond a certain level the target will be considered lost. To not lose the targets, they must be revisited from time to time.

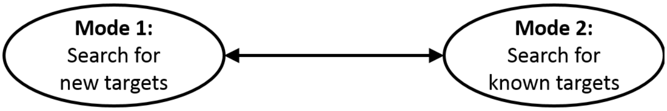


Figure 5.3: *The planning modes of the state-machine. The high-level planner is deciding the active mode.*

To search for new targets and to search for already known targets are in conflict with each other. New targets will most likely be located in an area yet not visited while the known targets most likely will be located in an already observed area. In [46] a state-machine is proposed to solve this problem. The state-machine contains two different states, representing two different planning modes and a high-level planner which chooses between the two different planning modes. The first mode is to search for new targets and the second mode is to search for known targets. The different modes can be seen in Figure 5.3 and a short description of each mode follows:

- **Search for New Targets:** The goal of this mode is to search for new targets. There are two different approaches, Section 5.3.2 and Section 5.3.3 describe the different problems of this mode and how the planning algorithms for this mode works.

- **Search for Known Targets:** The goal of this mode is to search for known targets. As time passes the covariance of known targets increases. It is therefore necessary to revisit a target from time to time to keep its covariance low, otherwise that target will be considered lost. Section 5.3.4 describes the planning in this mode.

Two different approaches have been used for the high-level planner; one based on an information measure and one based on distance between the track and the platform combined with the covariance of the track. In Section 5.3.1 the high-level planner described in [46] is introduced followed by each one of the planning modes.

5.3.1 The High-Level Planner

The task of the high-level planner is to determine which planning mode should be active. In general the task is to determine whether the platform should search for new targets or search for an already known one. It is desirable for the high-level planner to make a good compromise between finding new targets and minimizing the covariance of known targets. Following are two objective functions used for the different kinds of planning methods in Sections 5.3.2 and 5.3.3, information based planning and pattern based planning.

Information Measure

One approach is to design an objective function with the desired properties. The following objective function is proposed in [46]

$$V(t) = \sum_{j=1}^{M_t} e^{-\alpha \mathrm{Tr}[P_t^j]} \tag{5.1}$$

where α is a design parameter, M_t is the number of known targets and P_t^j is the covariance of target j at time t. A very small value for α leads to a measure which is maximized by finding as many targets as possible, while a value around 0.5 leads to a measure which is dominated by the covariances of the known targets. The value of α should be a compromise between these two. This objective function is used for the information based planning.

Covariance and Distance Measure

To determine the planning state when using pattern based planning to find new targets, another objective function has been defined. First the utility is calculated according to

$$U(j) = \begin{cases} \frac{\mathrm{Tr}\left(P_t^j\right)^\alpha}{||X_t^j - S_t||} & \text{if } j = 1, \dots, m \\ \frac{c}{||X^{\mathrm{WP}} - S_t||} & \text{if } j = m+1 \end{cases} \tag{5.2}$$

where m is the number of targets, α and c are design parameters, P_j is the covariance of target j, X_j is the position of target j, S is the position of the platform, X^{WP} is the position of the next waypoint, and t is the time index. The upper case is to calculate the utility for known targets while the lower case is to calculate the utility to search for new targets. The denominator in the two cases can be seen as a cost, defined as the distance to the goal position. The nominator can be seen as a reward. For known targets, it is desirable to visit targets with a high covariance. For new targets, the design parameter c defines this reward.

The function determining which target to be located is

$$k = \underset{j}{\mathrm{argmax}}\ U(j) \tag{5.3}$$

where k is the target to be found. If $k = m + 1$ then the platform should search for new targets. The choice of (5.2) as the objective function makes the high-level planner greedy as only the current situation is considered.

5.3.2 Information Based Planning

A possible planning method when searching for new targets is to always perform the action that maximizes the total information in a given time horizon. The following sections are based on the research in [46].

Measure of Information

To be able to plan the trajectory of the platform and the movements of the gimbal in the information based planning, a measure of information is needed. One way is to make a grid spanning over the area, where each grid point, x_i, holds the information on that position. We define the information matrix, $\mathcal{Y}_t$, as the inverse of the covariance matrix, P_t. Thus the information in each grid point can be represented by $\mathcal{Y}_t$, which is a 3 by 3 matrix. It is now possible to use the information matrix in a similar way to the covariance matrix, but in another form of the Kalman filter: the Information filter [34] (p. 160-162). We get an additive relation between the information in different time steps, according to the measurement update in the extended information filter

$$\mathcal{Y}_{t|t} = \mathcal{Y}_{t|t-1} + H_t^T R^{-1} H_t \tag{5.4}$$

in [46] where H_t is a linearization of the measurement model $h(x)$. The linearization is a first-order linearization, the Jacobian matrix of $h(x)$ evaluated in $x = \hat{x}_{t|t-1}$.

The elements on the diagonal can be interpreted as the information in each direction, and the other elements as the covariance between information in different directions. From the start, the information in all grid points is initialized to 0. When a grid point is seen, the information in that grid point increases according to equation (5.4). This can be seen as an addition of the inverse of the measurement uncertainty, depending on R, which corresponds to an addition of the measurement certainty. As information ages, a time update decreases the information in all grid points. This is done according to equation (5.5) below.

$$\mathcal{Y}_{t+1|t} = (F_t \mathcal{Y}_{t|t}^{-1} F_t^T + G_t Q_t G_t^T)^{-1} \tag{5.5}$$

In this equation F_t and G_t are the linearized versions of the motion model with respect to the state and the noise respectively. Like the case with H_t these are the first-order linearizations. In case of a random walk motion model, $F_t = G_t = I_{2x2}$. Q_t is here the covariance of the process noise.

Planning Problem

The planning problem is to calculate actions maximizing the total information of the world in a given time horizon. This requires an objective function, which is defined in the section below. Solving the planning problem is a complex task when using a platform with six degrees of freedom. To decrease the computational complexity, planning of gimbal movement and the trajectory of the platform are seen as two different problems.

Objective Function

To solve the planning problem the objective function (5.6), where Y is defined as $\text{blkdiag}\left(\mathcal{Y}^1, \mathcal{Y}^2, ..., \mathcal{Y}^N\right)$, is proposed in [46] as

$$L(Y_M) = \text{tr}\left(Y_M^{-1}\right) = \sum_{i=1}^{N} \text{tr}\left((\mathcal{Y}_M^i)^{-1}\right) \tag{5.6}$$

where M is the planning horizon and $\mathcal{Y}_M^i$ is the information in grid point x_i after M time steps. When the total information of the area increases, $L(Y_M)$ decreases, so the actions that minimize $L(Y_M)$ is the best possible actions to perform if the goal is to maximize information of the area. This definition of the objective function rewards exploration of unexplored areas, rather than further exploration of an area with a higher amount of information. It should be observed that this formulation neglects the correlation between different grid points.

Measurement Model

A camera can be approximated to be a bearings only sensor, which outputs the relative angles between the camera position $X^c = (x^c\ y^c\ z^c)^T$ and an arbitrary point, e.g. a grid point position $X^i = (x^i \ y^i \ z^i)^T$. Assuming the coordinates are given in the global coordinate system, the angles are given by

$$Y_t^i = \begin{pmatrix} \phi_t^i \\ \theta_t^i \end{pmatrix} = h(X^i; X_t^c) + e_t^i = \begin{pmatrix} \arctan_2(y^i - y_t^c, x^i - x_t^c) \\ \arctan_2(z^i - z_t^c, \sqrt{(x^i - x_t^c)^2 + (y^i - y_t^c)^2}) \end{pmatrix} + e_t^i \quad (5.7)$$

where $i \in [1, 2, ..., N]$ and e_t^i is noise modeled as a Gaussian distribution with mean 0, and the covariance is a parameter [46].

Probability of Detection

The measurement model explained in the previous section is valid only for an ideal sensor with unlimited field of view, and perfectly transparent world without occlusions. To model the quality – the probability of detection – of a measurement, several factors could be considered, such as, field of view, occlusions, range, sensor resolution, motion blur, and weather conditions. The probability of detection model is closely related to the definition in [46], defined as

$$P_D(X^i; X_t^c) = P_D^r(X^i; X_t^c) P_D^{occ}(X^i; X_t^c) P_D^f(X^i; X_t^c), \quad (5.8)$$

where P_D^r, P_D^f is the range respectively FOV depending likelihood functions, and P_D^{occ} is the likelihood function depending on occlusions.

The range dependent function models atmospheric condition and limited resolution of the sensor according to

$$P_D^r(X^i; X_t^c) = \exp\left(-\frac{\|X^i - X_t^c\|^2}{r^2}\right). \quad (5.9)$$

The FOV and occlusion depending likelihood functions $P_D^f(X^i; X_t^c)$ and $P_D^{occ}(X^i; X_t^c)$ will take the value 0 or 1. They are calculated according to the theory in section 2.5.1 and section 2.5.2, and share the property that they will return 1 if a point X^i is detected by the sensor; otherwise they will return 0. This makes sense, because the probability of detection for an occluded object or an object outside the sensor scope will be zero.

Update of the Information Matrix

The exploration algorithm works in a RHC (Receding Horizon Control) manner, which means that actions for several time steps will be planned, but only a part (typically the first) of the actions will be performed. In the planning it is reasonable to approximate Q_t in (5.5) to zero, because for small time horizons the aging of information can be disregarded. The update of the information for each grid point will according to [46] be

$$\mathcal{Y}_M^i = \bar{\mathcal{Y}}_{0|0}^i + \sum_{t=1}^{M} P_D(X^i; X_t^s) H_t^{iT} R^{-1} H_t^i \quad (5.10)$$

where $\bar{\mathcal{Y}}^{i}_{0|0}$ is the information matrix initialized to a small value times I_{3x3}, for numerical stability. For an exact description of how the information matrix is updated, see Appendix A.

5.3.3 Pattern Based Planning

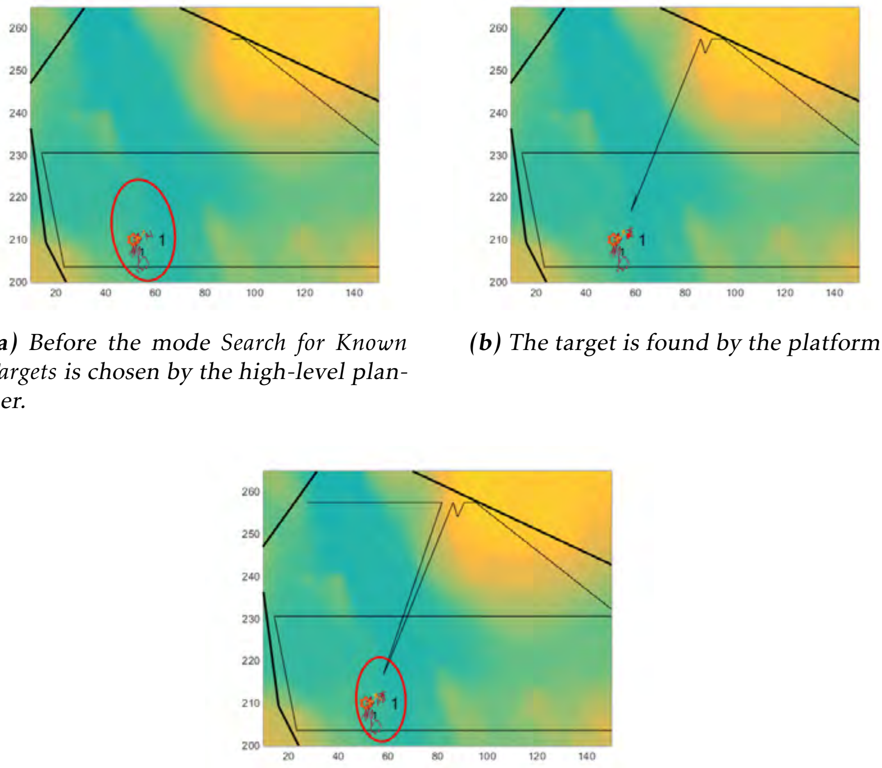
The pattern based planning approach to search for new targets make use of the theory described in Section 5.2. Compared to the information based planning, pattern based planning uses a completely different approach to solve the planning problem. Instead of finding the trajectory that increases the information the most, the idea is to move across the map in a systematic way, a pattern. The idea is that the planning is being done offline and the result is a set of waypoints. The waypoints should be visited in a certain order and thereby forms a pattern. Once the mode *Search for New Targets* is active, the platform is set to move to the first waypoint in the set marked unvisited. Once that waypoint has been visited, it is marked so. This enables the platform to continue on the planned route when the high-level planner switches the active mode to *Search for Known Targets* and then back.

5.3.4 Search for Known Targets

In this mode, the goal is to revisit old targets for which the covariance has grown too large. The problem then becomes to find the target of interest. This is done by visiting the most likely location of the target. For the target, an estimated position and covariance is yielded. One simple approach is to let the platform move to the estimated position and then scan the nearby area with the gimbal. If the target cannot be found then it is considered lost. This strategy works well if the platform is located on a high altitude relative to the target. The tracks are updated according to the negative measurement update which is described in Section 4.2.2. This update lowers the weight of particles in LOS when no measurement is associated to the track of the particles. Figure 5.4 shows the procedure of finding an old track and how the covariance of the track is varying throughout the sequence. When the covariance is too large the high-level planner decides to revisit the track. When the target has been found the covariance is lowered significantly. The platform then continues back to the planned trajectory, and the covariance starts to grow again. A result of the high-level planner being greedy can be seen in Figure 5.4. as the platform makes a zigzag movement just before moving down to the target.

5.4 Gimbal Planning

This section describes different gimbal movement strategies used in the thesis project. The goal of the gimbal movement strategy is to maximize the ground coverage of the camera given movement of the platform. The most basic strategy is to let the camera have a fixed direction pointing downwards. This strategy is



(a) Before the mode *Search for Known Targets* is chosen by the high-level planner.

(b) The target is found by the platform.

(c) The enclosure has been searched.

Figure 5.4: *Sequence of revisiting a known target. The red ellipses represent the covariance of the tracks.*

very easy to implement, however the camera footprint is at its minimum. There are many ways to improve this when a camera is used.

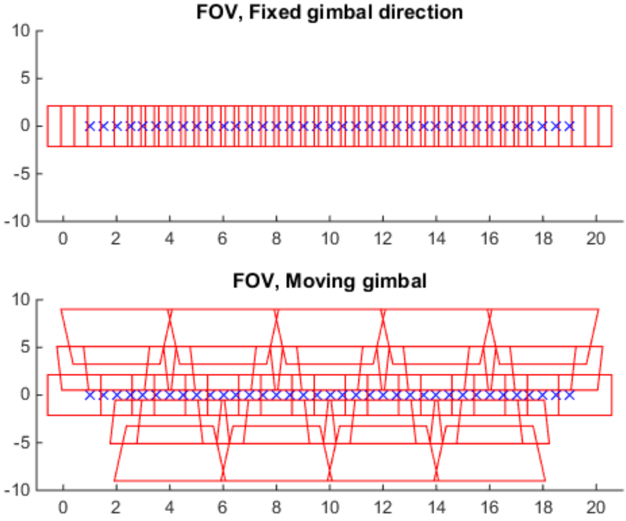


Figure 5.5: Comparison of the two gimbal movement strategies.

For example, when the platform is moving according to a lawn mower planning strategy, the number of times the platform has to move back and forth can be decreased. To do this, the camera footprint must be increased when the platform moves in one direction. One way to do this is to let the gimbal direct the camera from side to side of the platform. It is important that the gimbal is able to rotate back and forth quickly such that no ground is left behind unseen when the platform is passing. One way to assure this, while reducing the gimbal speed, is to reduce the speed of the platform. Moving the gimbal quickly has a negative impact on the quality of the pictures, as this introduces noise. This is a result of not having infinitesimally small shutter time for the camera. Thus, a trade-off between the speed of the platform and high quality pictures appears. In Figure 5.5 the moving gimbal strategy is compared to the strategy of having the gimbal point downwards at all times. The camera footprint of the first strategy has a lot of redundancy, the same area is seen multiple times. For the second strategy this redundancy is reduced significantly and a larger area is covered.

5.4.1 Time Based Gimbal Strategies

Two different time based gimbal strategies have been used in the project. The first strategy is to let the gimbal rotate around its roll axis and having both the pitch and yaw angles be fixed. The result can be seen in Figure 5.6. The pitch angle of the gimbal can be varied. When the platform is flying on a low altitude the camera can point slightly forward by increasing the pitch from minus 90 degrees, meaning that the camera is directed straight down. This will make the camera have a slightly larger LOS. When flying on a high altitude, increasing the pitch too much will make the camera only see object in the far distance. Other design parameters are the roll rotation speed and the limit angles for the roll.

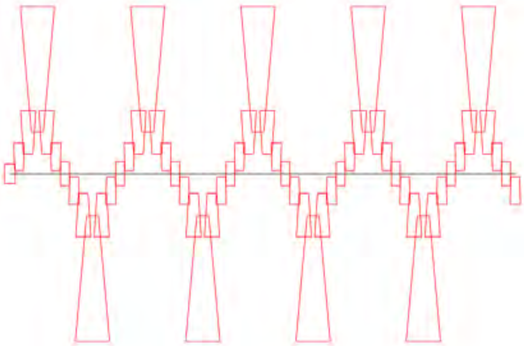


Figure 5.6: A simulation of the FOV for the roll rotation gimbal movement strategy.

The second time based gimbal strategy which has been implemented is to let the gimbal instead rotate around its yaw axis and let the roll and pitch axis be fixed. The pitch angle can be varied with the same advantages and disadvantages as before. The roll angle should be zero to make the video be horizontal. This is an advantage for this strategy compared to the prior strategy, the video will be horizontal. This will make the video nicer for operators to analyze, and nicer for presentations of the work. Figure 5.7 shows how the FOV changes for the gimbal movements in this strategy.

5.4.2 Adaptive Gimbal Strategies

Another approach for the gimbal movement is to use the information of the location for the enclosures. It is desirable to observe as much as possible of the enclosures and increase the oscillation frequency if possible, to maximize information. One way to do this is to let the limit angle vary between two angles $[\phi_{min}\phi_{max}]$ which define an interval. To ensure as much of the enclosure is observed, the gimbal should change direction as soon as the FOV is not completely within the enclosure. By changing the rotation direction at a lower limit angle the gimbal oscillation frequency increases, thus more of the enclosure will be seen. The lower limit angle ϕ_{min} defines the minimum rotation angle limit and prevents the gimbal from just directing the camera downwards when the platform is outside of an enclosure.

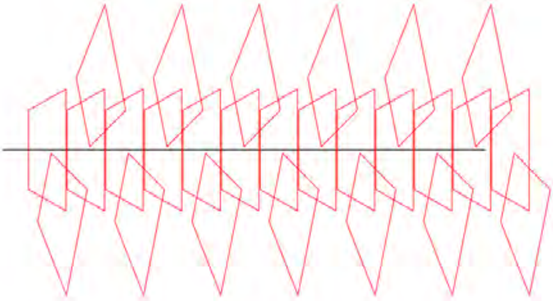


Figure 5.7: *A simulation of the FOV for the yaw rotation gimbal movement strategy.*

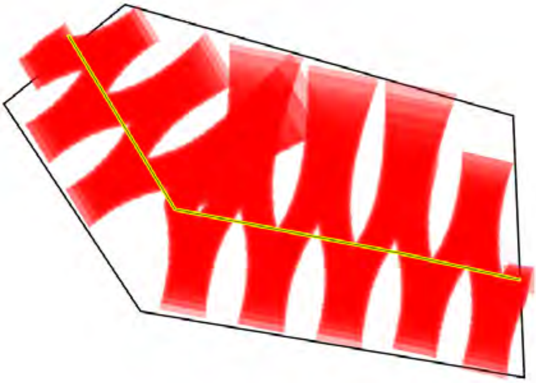


Figure 5.8: *The adaptive roll rotation gimbal movement strategy.*

Figure 5.8 shows how this gimbal movement strategy works. As soon as the gimbal starts pointing outside of the FOV it changes the rotation direction. In Figure 5.8 the rotation is around the roll axis, this can however also easily be applied to a rotation around the yaw axis. The algorithm is briefly described in Algorithm 2.

Algorithm 2: Adaptive Gimbal Movement

```
Data: Platform position, camera attitude, enclosure, desired fixed pitch
      and maximum roll
Result: Adaptive gimbal movement around roll axis
begin
    rollGoal = maximumRoll;
    pitchGoal = fixedPitch;
    direction = clockwise;
    while true do
        if $rollCamera \geq rollMinGoal$ and $|FOV \cap Enclosures| \neq |FOV|$
         then
            if direction == clockwise then
                rollGoal = -maximumRoll;
                direction = counterClockwise;
            else
                rollGoal = maximumRoll;
                direction = Clockwise;
        else if $rollCamera \geq rollMaxGoal$ then
            if direction == clockwise then
                rollGoal = -maximumRoll;
                direction = counterClockwise;
            else
                rollGoal = maximumRoll;
                direction = Clockwise;
```

6
The System

Today, the usage of drones or UAVs are being adapted in more and more fields. One application is to transport goods, for example Amazon recently announced that they will be using drones to deliver products to customers [2]. However, most applications of UAVs utilize the ability of drones to access and view locations impossible for humans to access. UAVs can be equipped with cameras and have the video transmitted to the ground where an operator can view the video stream on a computer. This opens possibilities to get a quick overview of a large area. For example fire fighters can use UAVs to get a picture over a forest fire, or movie makers can use UAVs to record scenes from a new perspective. UAVs can also be utilized for surveillance, which is the area of application for this thesis project.

The idea of this thesis project is to use a UAV to perform tracking, searching and planning. At the time of writing, the UAV market is in a great expansion phase. There are a lot of UAVs offered on the market. The vast majority however are closed systems which does not support autonomous planning. Instead these UAVs are designed to be controlled by an operator using a remote controller. The search of a UAV which can be controlled by a planning program is a major task in this project.

In this chapter an overview of a general UAS will be given followed by a more detailed description of each subsystem making up the UAS. For each subsystem, it will be described how this affects the overall system properties. After this, the hardware search for the specific task of the thesis will be described and a setup will be proposed. The chapter ends with a section about the software that has been developed and used in the thesis.

6.1 Overview of a UAS

According to [23] the term UAV is defined as following:

> A powered, aerial vehicle that does not carry a human operator, uses aerodynamic forces to provide vehicle lift, can fly autonomously or be piloted remotely, can be expendable or recoverable, and can carry a lethal or nonlethal payload. Ballistic or semiballistic vehicles, cruise missiles, and artillery projectiles are not considered unmanned aerial vehicles.

This is a wide definition which means that the term UAV includes a wide spectrum of vehicles designed for completely differently purposes. Essentially, any vehicle designed to fly unmanned with the ability to be controlled fits within the term UAV. Examples of such vehicles are unmanned planes, helicopters, multicopters but also unmanned air balloons. A UAV however only includes the vehicle and leaves out essential parts such as wireless communication, ground PC and ground-control software. A better term which includes not only the UAV but also all components building up the system is UAS. From here on the term UAV only refers to the aircraft while the term UAS refers to the whole system.

Figure 6.1 shows two examples of UAVs. Most UAVs are either controlled from a remote location or controlled automatically by a computer placed onboard the aircraft. In recent years UASs have become very popular in many areas. Today UASs are used in both military and civilian applications. Examples of civilian applications include video production, search and rescue, agriculture, and surveillance. The big advantage of UASs is the ability to access and view locations impossible to humans. The most popular equipment for a drone to carry is cameras of both EO and IR types. The video is transmitted to the ground where an operator can view the video stream on a computer. This opens possibilities to get quick overview of a large area. This also makes a UASs ideal for surveillance as a large area can be searched quickly. For example in [43] it is estimated that a UAS could replace up to eighty human rangers to patrol an area in an African wildlife park.

6.1.1 Common Parts of a UAS

This section aims to break down the UAS into smaller subsystems. The level of each subsystem corresponds to the parts that are often purchased separately when building a UAS. Here, each subsystem will be presented and its purpose described very briefly. In the following sections each and every subsystem will be described more in detail.

The body of the aircraft is called frame. The frame will carry all other subsystems to be placed on-board. When purchasing the frame motors and rotors are often included, otherwise these have to be bought separately. The aircraft gets powered typically from Li-ion batteries, but there are also alternatives where a combustion engine is used. The latter alternative gets more common as the size

(a) A multicopter.
By courtesy of Alexander Glinz. The photograph is taken from https://commons.wikimedia.org/wiki/File:Hexacopter_Multicopter_DJI-S800_on-air_credit_Alexander_Glinz.jpg *the 3rd of May 2016.*

(b) An unmanned plane.
By courtesy of Stefan Sundkvist. The photograph is taken from https://www.flickr.com/photos/stefansundkvist/4697864162/in/photostream/ *the 3rd of May 2016.*

***Figure 6.1:** Example of different UASs.*

of the vehicle increases. The UAS also requires an autopilot which controls the aircraft and receives commands from an operator on the ground. These commands are sent over a radio link which is either connected to a remote controller or a computer. To enable a radio link, the aircraft must have a radio receiver and the ground system must have a radio transmitter. It is optional to send telemetry data from the aircraft down to the ground. If this is to be done, a radio link in the opposite direction is needed. A GPS is often connected to the autopilot to enable a robust positioning of the aircraft. The aircraft can be equipped with various sensors. One of the more common ones are cameras. These are often mounted onto a gimbal whose purpose is to stabilize the camera and point it in different directions. The video is sent down to the ground over a video link which works in the same way as a radio link. Down on the ground, the video is received and displayed on a screen. Sometimes it is of interest to perform calculations on-board the aircraft, for example to control the aircraft autonomously or perform image processing. For this, a small computer is needed.

In Figure 6.2 an example of an on-board system is shown. This system communicates with a ground system through radio links. An example of a ground system can be seen in Figure 6.3.

6.1.2 System Parameters

Depending on the contrived application of usage for the UAV system the desired system parameters will differ significantly. These system parameters have to be taken into consideration in the design process. For example, a UAS contrived to be used for racing will require high speed and good dynamic performance whilst the altitude or range might not be critical parameters. On the other hand, for a UAS designed to be used for forest fire detection, range as well as altitude will be of greater importance. Some examples of system parameters are:

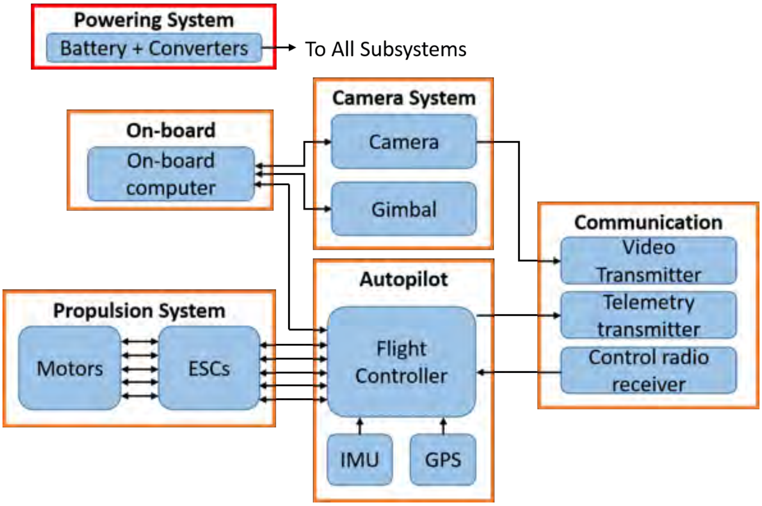


Figure 6.2: An example of an on-board system for a UAS.

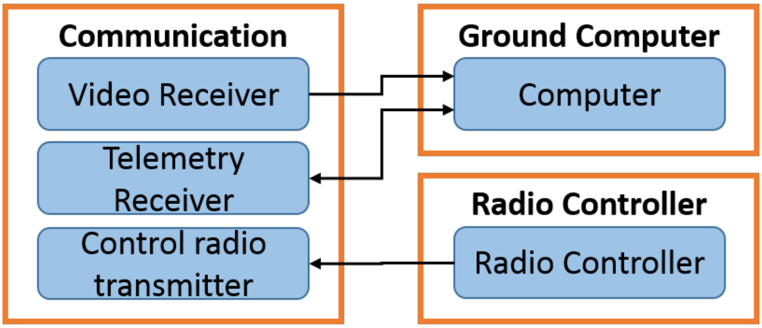


Figure 6.3: An example of a ground system for a UAS.

- **Flight time:** The maximum operating time of the UAS. This depends on how the UAV is controlled. Normally this parameter is defined as the operating time when the UAV hovers in the air.
- **Range:** The maximum communication range for the UAS, i.e. the farthest distance the UAV can fly from the operator before losing radio connection.
- **Weight:** The weight of the UAV.
- **Payload capacity:** The maximum weight of additional equipment which the UAV can carry.
- **Speed:** The maximum flight speed of the UAV. This is often broken down into ascent speed, descent speed and translational speed.
- **Altitude:** The maximum operating altitude of the UAV.
- **Cost:** The cost for purchasing the UAV and all necessary equipment for the UAS.
- **Ability to be controlled from computer:** Can the UAV be controlled from an on-board computer, and thus enable autonomous behavior?
- **Ability to capture video or photos:** Does the UAS include a camera mounted onto the UAV?
- **Dynamic performance:** A good dynamical performance can both mean how well the UAV can handle windy conditions but also how quickly it can change direction.

In the following sections it will be discussed how the different subsystems of the UAS correspond to the system parameters.

6.1.3 The Frame

The frame can be considered the skeleton of the aircraft as all other parts of the UAV will be mounted onto this. Thus, the frame should offer possibilities to carry equipment and be stable. The frame is also the part which classifies the UAV. The UAV classifications which are discussed here are planes and multicopters.

Planes and multicopters utilize completely different tools to be able to fly. While planes utilize both wings and rotors, multicopters only use rotors. The number of rotors on a multicopter is usually between three to eight, but technically the number of rotors are not limited. Every rotor requires a motor and an ESC (Electronic Speed Control). The ESC controls the rotation speed and direction of the motor it is connected to. The ESC is connected to both a motor and the autopilot. The autopilot gives a reference value to the ESC for the rotation speed of the motor.

A plane requires less energy compared to a multicopter to keep itself in the air because of the lifting capacity of its wings. A multicopter must instead always rotate its rotors to generate enough lifting force to keep itself flying. This means that a plane has a better flight time compared to a multicopter. Planes can also fly faster than multicopters. Since planes have both a longer flight time and higher flight speed than multicopters, they are able to fly a longer distance during a single flight. An advantage for the multicopter compared to the plane is that it can hover. A plane must always move forward to get lift from its wings. Also, a plane cannot change direction as fast as a multicopter. The ability to hover in place is important as the vehicle can stay in one place and collect data.

6.1.4 Multicopter Frames

As mentioned earlier the number of rotors for a multicopter varies. The multicopter on the market with the fewest number of rotors is the tricopter. This is the cheapest type of a multicopter since only three motors and rotors are needed. However it is harder to control and keeping stable, thus making it inappropriate to use for filming, but a great platform for learning how to fly a UAV. The multicopter next in size is the quadcopter which is equipped with four rotors. This is the most common type of a multicopter as it is relatively cheap and easy to control. On the market there are also hexacopters and octacopters which are more stable and robust against wind. They are however more expensive due to the increased amount of motors and rotors. Both the hexacopter and the octacopter can handle the failure of one of its motors and still be able to fly. This does not hold for quad- or tricopters. This might be an important feature if the UAV is carrying expensive equipment.

Multicopter Movement

This section contains a brief explanation of how a quadcopter is being controlled and how rotor rotation speeds corresponds to quadcopter movements. The philosophy can easily be applied to multicopters with a different number of rotors. The quadcopter configuration is selected as this is the most basic form of a multicopter. First off, the dynamics of a rotor will be explained to get an understanding of how the rotation of the rotor affects the quadcopter, followed by each basic movement of the vehicle.

Multicopters utilize multiple rotors to generate the force to control the aircraft. The number of rotor blades usually varies from two and up to four, however more blades can be used. An important factor when designing a rotor is to locate the center of gravity at the center of the rotation. Otherwise the rotor will become unbalanced and create major vibrations. When the rotor is rotating two major forces are created, see Figure 6.4. The thrust is the force which lifts the aircraft and enables it to fly. The torque is a result from the drag of the blades, this has to be considered to keep the quadcopter stable in the air. For example, if all four rotors of the quadcopter would rotate clockwise, the quadcopter would start to spin uncontrollably in the counter-clockwise direction.

To avoid this the rotation direction of two of the rotors are switched, as can be seen in Figure 6.5, thus the resulting torque will be zero when all rotors have the same rotation speed. By changing the rotation speed for all rotors the thrust

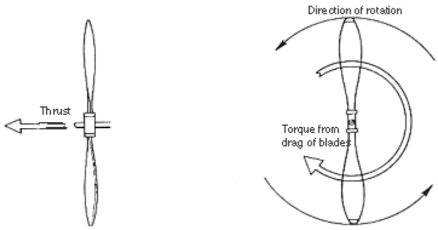


Figure 6.4: *Dynamics of a rotor.*
By courtesy of Donald James. Image taken from `http://quest.nasa.gov/aero/planetary/atmospheric/propulsion1.html` *the 3rd of May 2016.*

generated can be changed [36]. To make the aircraft ascend the thrust generated has to be greater than the force of gravity. To make the aircraft hover, the thrust must instead be equal to the force of gravity. In the same way to make the aircraft descend in a controllable fashion the thrust must be slightly less than the force of gravity.

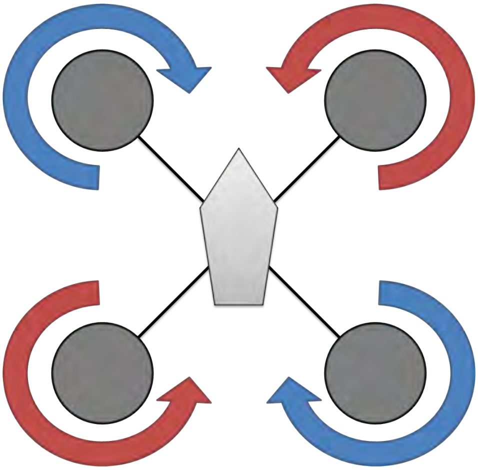


Figure 6.5: Altitude Change: *All rotors spin with the same speed. This results in the quadcopter either rising, declining, or hovering, depending on the rotational speed. Here the blue arrows symbolize rotation clockwise. The red arrows symbolize rotation counterclockwise.*

The torque generated by the rotors can also be utilized to rotate the aircraft around its yaw axis [36]. This can be seen in Figure 6.6, where the rotation speed of the diagonal-pair rotors differ. Since the rotors within each diagonal-pair have the same rotation direction this situation will result in a torque which is not equal to zero, thus making the aircraft rotate.

To make the quadcopter move sideways, more thrust has to be generated on one side of the aircraft. This will make the aircraft and its rotors lean and thus it will move sideways. This is illustrated in Figure 6.7 where a positive rotation

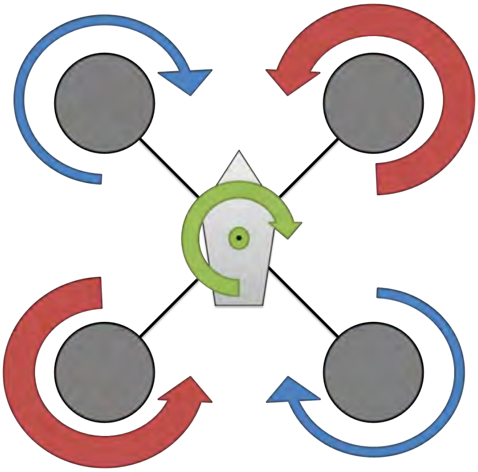


Figure 6.6: ***Yaw Rotation:*** *The rotation speed is different for the two diagonal-pairs. In the figure the quadcopter will rotate clockwise as a result of the negative yaw rotation.*

around the roll axis will be made.

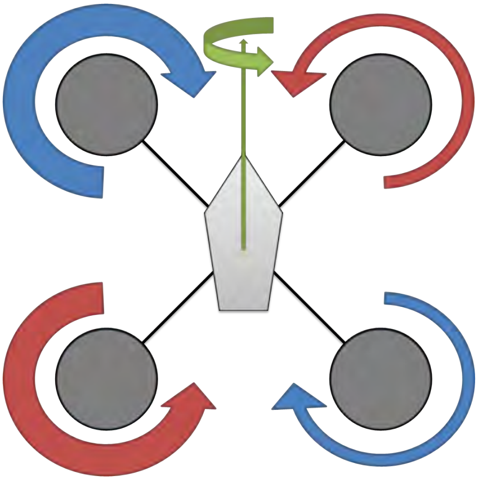


Figure 6.7: Roll Rotation: *The rotation speed is different for each side of the quadcopter. In the figure the quadcopter will move to the right as a result of the positive roll rotation.*

In a similar fashion, to make the quadcopter move forward or backwards, the front and the back rotors must have different rotation speeds such that the thrust will be different between the front and the back. This will make the aircraft lean forward or backward. The situation is described in Figure 6.8 where the situation of the quadcopter moving forward is described.

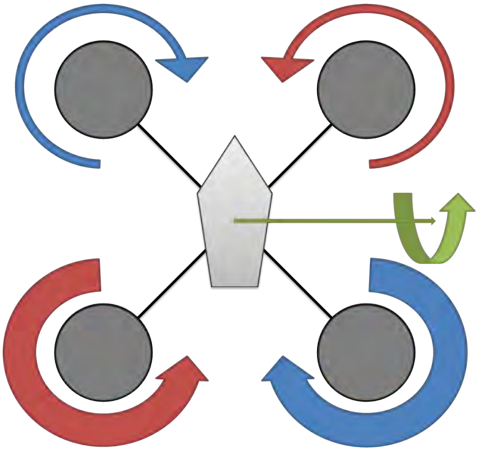


Figure 6.8: Pitch Rotation: *The rotation speed is different for the front and the back of the quadcopter. In the figure the quadcopter will move forward as a result of the negative pitch rotation.*

6.1.5 Thermal Camera

The human eye is sensitive to electromagnetic radiation with wavelengths between 400 to 700 nm [29]. Just above this spectrum, we have IR (Infrared Radiation), which covers wavelengths from 0.7 μm to 1000 μm. This type of electromagnetic radiation is not visible to the human eye, but still contains a lot of information that could be used for various applications. With a thermal camera, the thermal information from a scene can be converted to a visual image. High speed thermal cameras are considered to be dual-use items, and are thus export regulated from the US. To not need an export license, the frame rate has to be below 9 Hz.

Design

The thermal camera is similar to the common digital EO camera, and contains optics, a sensor, and a video processing circuit that converts the thermal image to an image visible for the human eye.

Because glass is not transparent for infrared radiation, the optics in an infrared camera are made of other materials, such as silicon or germanium. Like in EO cameras, antireflective coatings are used on the optics. A good IR camera lens will transmit almost 100% of the incident radiation [32].

There are two main types of sensors: quantum and thermal detectors. The former operates on the basis of the photoelectric effect. Quantum detectors are cooled to a temperature where no electrons are in the conduction band. When photons hit the sensor the electrons of the material are elevated to a higher energy state, causing a voltage and current that is measured [32]. A common type of the other sensor type, the thermal detector, is the microbolometer. The working principle of the microbolometer is that photons are absorbed by different pixels of the sensor, changing the temperatures of these regions. This leads to a change of resistance that can be sensed through applying a voltage over the pixel [50]. The advantage with the microbolometer sensor is that it is low cost, could be made small, has a broad (flat) response curve, and does not require cooling. Quantum detectors in general have lower response time, higher resolution and produce better pictures because of higher sensitivity. The drawbacks are that they are heavier, as cooling equipment is needed, and more expensive.

An important task for the image processing electronics is to convert the thermal image to a visible image. The conversion is done by mapping the received energy of the thermal radiation in a pixel to a corresponding color. Several color schemes can be used, and there is often a possibility to choose color scheme dynamically. Common color schemes are grayscale, where white is hot and black is cool (or the opposite), or color schemes that goes from blue, to red, to yellow with increasing temperature.

Properties

The properties for thermal cameras are the same as for EO cameras, except the ability to select temperature range. The resolution is an important property. Infrared cameras have lower resolution than EO cameras. The resolution can range from about 160 × 120 pixels up to around 1024 × 1024 pixels. A property connected to the resolution is the FOV of the camera. It depends on the sensor size, the distance from an object, and the properties of the optics.

The temperature measurement range is often selectable by the user. When the interesting region has large temperature differences this is a valuable feature. Through decreasing the temperature range the resolution in that range is increased.

Some thermal cameras have the possibility to output absolute measurements of temperature. With this feature the temperature of different objects in the image is available to the user. This problem is harder than just mapping received energy to a color for each pixel. To be able to get absolute measurements the emissivity of the material has to be taken into account.

Reasons for Use in the Project

In this project, the thermal camera is the perfect tool to use to get an overview of an area. Large animals like rhinos, could be detected from a further distance, and then classified more precisely with an EO camera for example. In addition, the use of a thermal camera ensures that the tracking can be performed also at night.

6.1.6 EO Cameras in UAVs

EO cameras are common equipment to mount on drones. They are utilized in many of the different applications for drones, such as; surveillance, mapping and agriculture. By placing an EO camera on a drone views can be captured from locations otherwise inaccessible to humans. The video is often transferred to the ground station through a wireless video link to be viewed by an operator. Cameras are also used to provide a first-person-view (FPV) of the drone to help the operator in controlling the aircraft. This is particularly useful when the aircraft is far away from the operator and it is hard to distinguish the orientation of the aircraft.

6.1.7 Video Encoder

A video encoder, also known as video server, is a device used for converting analog video to digital video. They are mostly used for security systems where the use of video encoders also makes it possible to connect and control many cameras over an IP network [15]. A video encoder takes analog video, e.g. NTSC or PAL, from a source and converts it to digital video. Common formats for the digital video is H.264 and MJPEG.

6.1.8 Autopilot

A multicopter system is not stable when flying, so for an operator to try to operate the aircraft without any type of automatic control would be very difficult if not impossible. It is the autopilot which handles the fast control loops, enabling easy control of the aircraft. The autopilot contains the flight controller as well as internal or external sensors of various types such as; barometer, magnetometer, gyroscope and accelerometer. The flight controller is essentially a processor computing the desired rotation spin of each rotor given measurements from the sensors and control commands from the operator. The commands can either be sent from an operator on the ground or an on-board computer. It is also common to have a GPS connected to the autopilot, especially if the multicopter is intended to be flown outside. Most autopilots support several flight modes such as waypoints, circular motion, area coverage etc.

Flight Modes

Today UAVs come with more and more support for different flight modes. The basic flight mode is for the UAV to be controlled through a remote controller by an operator. The commands are sent wirelessly from the remote controller to the autopilot mounted on the aircraft. This flight mode is given the highest priority in many autopilot systems as it is desirable for the operator to always have the option of taking control of the aircraft. For example, in case of an emergency where the vehicle is about to fly into a building or tree, the operator should be able to regain control of the vehicle and avoid the crash manually.

Another important flight mode commonly used outdoors, is the use of waypoints. The operator then sends a set of waypoints, which are defined either in GPS coordinates or coordinates relative the start position of the current flight, to the autopilot. The autopilot then makes the vehicle go to the waypoints one after another. This is a very powerful flight mode as the autopilot handles all control given certain parameters such as flight speed and altitude. This means that a computer can potentially send the autopilot a set of waypoints and thereby the vehicle flies autonomously.

A flight mode which has become more popular in recent years, much thanks to the successful marketing of the Lily drone [11], is the follow target mode. The Lily drone uses a tracking device which the target must carry. The tracking device emits a signal which the drone can track and thereby follow. Today there are other UAVs which uses this flight mode but does not require a tracking device. Instead, images from the camera recording the target are analyzed by using image processing to find the target in the video frames. From this an estimation of the target position relative to the drone is gained. One example of a project using this technique is a project at Linköping University [19].

A flight mode which is very good for safety reasons is the mode to fly home and land. The start position of the flight is saved at take-off. Thus the vehicle knows one location where it is safe to land. In this mode, the home location is basically seen as a waypoint where the vehicle will land when it has arrived at the waypoint. This flight mode is often activated when the batteries of the UAV is running dangerously low or when radio connection to the operator is lost.

6.1.9 Gimbal

The gimbal is the device which stabilizes the camera and makes it possible to produce professionally good-looking video recordings. Gimbals can both be handheld or mounted on drones. Gimbals are mounted on every drone used for professional photography or videography, and costs vary between \$30 up to hundreds of thousands of dollars. The gimbal can also point the camera in different directions except for just stabilizing the camera. Gimbals are most commonly mounted underneath the body of the vehicle to enable viewing in all directions. Another possibility is to place the gimbal in the front or the back of the aircraft. The gimbals are controlled by either an operator or an on-board computer.

The gimbal contains an IMU which is mounted on the same plane as the camera. This gives measurements of the attitude which the board of the gimbal uses to provide control commands to the motor of each axis. Every motor is controlled by a controller, often a PID-controller. For the gimbal to work properly and without draining an excess of power, it has to be balanced. This means that the gimbal does not require any power to keep the gimbal still.

2 Versus 3 Axes Gimbals

The difference between the 2 and the 3 axes gimbals is the number of axes which the camera is stabilized in. Both gimbal types stabilize the pitch and roll axes, additionally for the 3 axes gimbal the yaw axis is stabilized. The 3 axes gimbal is heavier and requires more power as an additional motor is added compared to the 2 axes gimbal. The additional weight will shorten the flight time of the vehicle for the 3 axes gimbal compared to the former. It is common for a 3 axes gimbal to be controlled by an additional operator due to the amount of degrees of freedom. The 2 axes gimbal can still be controlled by the same operator who controls the vehicle as the camera will always look forward in a sense (the pitch angle will differ).

6.1.10 On-board Computer

While the autopilot handles the fast control loops, the on-board computer works on a higher level with planning and decision making. It also handles the image processing, to find and track targets. Since the weight criterion is important, the most commonly used on-board computers is of the single-board type.

Interface

The on-board computer has to be connected to the autopilot to retrieve the current position and attitude of the camera. When decisions have been made these have to be sent to the autopilot for realization.

Common Requirements

As with all equipment on an UAV the weight and power consumption is the most important variables. But when it comes to computer vision, computing power is also an important factor.

6.1.11 Radio Communication

In the case of a fully autonomous drone, radio communication would not be needed. However, for safety and debugging reasons it should be seen as compulsory equipment. There are three main types of communication: remote control, video and telemetry. One should note that currently, there is no open radio system transferring these streams over the same channel. The normal way to build a system is to have three separate units communicating on different frequencies to not interfere with each other. Which frequencies are used depends on jurisdiction in different countries. For example, the 900 MHz band heavily used in the US, is in Sweden allocated for mobile communication.

Remote Control

A link for remote control is needed to be able to manually steer the drone. The link consists of a radio on the ground, transmitting signals to the receiver on the UAV. A typical radio can be seen in Figure 6.9. The ground radio has two sticks, the left for throttle and yaw, and the right for pitch and roll. With these four control signals, it is possible to control the drone completely. There are also additional channels on the radio for other functionality such as gimbal control, mode control and sometimes even parachute control. The radio communication concerning control of UAVs normally uses the 2.4 GHz or the 5.8 GHz band.

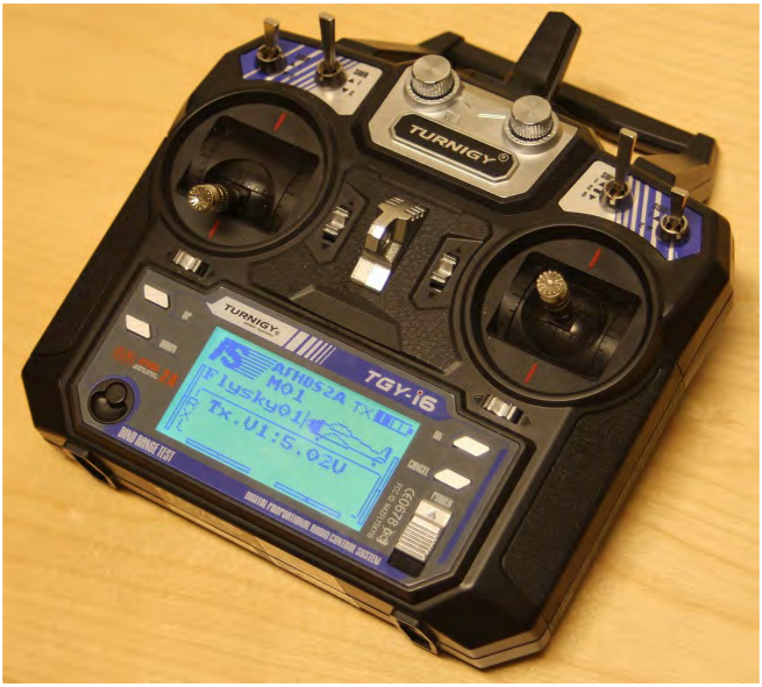


Figure 6.9: *A typical remote control, with a radio transmitter, for aerial vehicles.*

Video

In the UAV world it is common to use an FPV (first person view) camera and a link to transmit the FPV video to the operator. This is a valuable feature to extend the range of control. The limitation when controlling a drone is normally the eyes of the operator. After a certain distance it becomes hard to see the direction of the UAV, and when that happens the chances of a successful flight drop low. For FPV video transmission there is a lot of frequency alternatives, but the most common is to use 5.8 GHz. If this frequency is used for video, the remote control signals usually are transmitted on the 2.4 GHz band.

Telemetry

To be able to control the vehicle by e.g. sending waypoints a telemetry channel has to be used. This is normally bi-directional communication. The UAV sends signals about its position, velocity and attitude which can be received by a receiver connected to a GCS (Ground Control System), running on a computer. From the GCS waypoints or other control commands can be transmitted to the telemetry unit on the drone and thereafter sent to the autopilot.

6.1.12 Powering

The powering of a UAV is an issue, since they mostly operate on the principle "heavier than air" and thus have to create a lift force by pushing something downwards. The rapid development of batteries, especially lithium based batteries, has given rise to the equally rapid development on the UAV field. For a battery supposed to power a UAV the charge density is important. In Figure 6.10 different types of batteries are compared. We can see that the lithium based battery types is in the top considering energy per weight. Therefore this type of battery has been adopted by the RC (Radio Control) world, and consequently also in the drone world.

The Lithium Ion Battery

When searching for a sufficient battery for an application, several properties such as voltage, discharge rate and weight, are important to consider. The last criterion, weight, is what makes the lithium ion battery appealing. The voltage and current requirements are different for different UAVs. The nominal voltage of a single cell is 3.7 V. This means that 2 cells in series would output 7.4 V, three cells in series 11.1 V, and so on. A measure commonly used for the voltage of Li-ion batteries is the number of cells in series which means that the previous numbers would be written as 1S, 2S, and 3S. Another big advantage with the Li-ion battery is that it can deliver a high current. Discharge rates are measured in units of C, which tells how much current a battery can deliver. The C-rate is related to how much energy the battery can store, and a value of 1C means that the battery could be fully charged or discharged in an hour. The discharge rate is normally higher than the charge rate, and for example a battery could have the capacity 3000 mAh

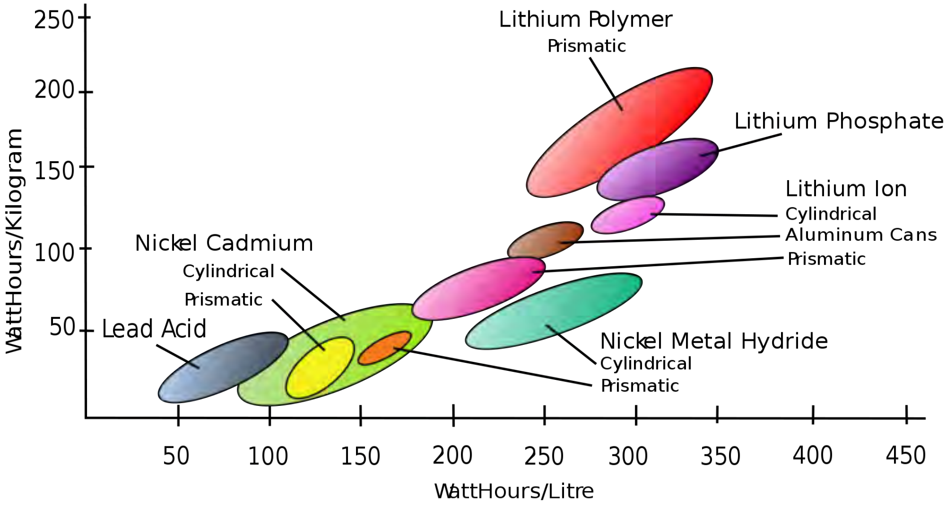


Figure 6.10: *Charge density for different types of batteries. In the upper right part of the graph batteries with high charge density can be found, such as lithium based batteries.*
By courtesy of Barrie Lawson. Image taken from `https://commons.wikimedia.org/wiki/File:Secondary_cell_energy_density.svg` *the 17th of May 2016.*

- 10C for discharge, and 1C for charge. This battery could theoretically deliver 30A and be charged at 3A [38].

6.1.13 Voltage Regulator

On a UAV, many subsystems are cooperating: gimbal, autopilot, propulsion system, camera, etc. This means that a lot of devices, with different powering setups, should be powered on the same platform. Some devices need a stable input voltage, and could thus not be driven directly from the battery because of the varying voltage. For voltage conversions a BEC (battery eliminator circuit) could be used. This is a switched DC-DC circuit which works in either step-down (buck) or step-up (boost) mode, or a combination. The efficiency of a switching voltage regulator is often around 90%. A BEC does exactly what the name suggest; it eliminates the need of an additional battery and makes charging and coupling easier. Figure 6.11 shows an example of a BEC.

6.1.14 Ground Control Station

When the complexity of a UAV increases, the operators need more information and a better interface to control it. A ground control station (GCS) could come in many forms, where the simplest solutions only includes a radio, and the more profound solutions consist of multiple radios, computers and screens.

QM12V5A-UBEC
QUANUM
Output current : 5A (Max.8A)
Output voltage : 7.2V / 12V
Input voltage : 7.2V~25.2V (2~6S LiPo)
Dual output sockets
Input
Output
Switch
12V
7.2V

Figure 6.11: *An example of a BEC.*

Functionality

Common functionality is the ability to view the position of the UAV on a map, view and log attitude data, send waypoints and movement patterns, change parameters, etc. The communication between the GCS and the UAV could for example be over a bidirectional telemetry channel.

6.2 Hardware Proposal

There are two different approaches when it comes to where the calculations for tracking, searching and planning, should be performed. The first approach is to have an on-board computer which performs the image processing, tracking and planning calculations. The second approach is to use the UAV as a recorder and perform computations on the ground. For both approaches the UAV needs to be able to carry a camera with gimbal, batteries, communication module and a computer to control the UAV and perform commands. Having more equipment placed on the UAV increases the payload which in turn requires more power keeping the UAV flying, resulting in less flight time. The video stream from the camera also has to be transmitted to the ground computer where it should be shown to the operator. The difference between the two approaches is where the heavy computations take place. If these are being performed on-board, a light-weight computer must interpret the video stream and perform the image processing. Otherwise the video stream has to be transmitted with a communication link with low delay and high capacity.

The ideal system for this thesis would have long flight time, high payload capacity, and long control range. It would also be small, cheap, and have the possibility to simultaneously carry an EO camera, a thermal camera and be able to stream two video streams to the ground computer. To be able to plan a search, sensor data from the gimbal must be available. The system also would be able to perform image processing, tracking and planning either on-board or on the ground computer. The communication link between the UAV and the ground computer should support two-way communication. Another important requirement is that the system must be easy to assembly and work with. These requirements are difficult to satisfy. To get measurable requirements, reasonable numbers were used for the flight time, maximum payload, transmission range, size and price. Table 6.1 shows the requirements on the system at the start of the hardware research.

Table 6.1: *Starting requirements on the system.*

Type	Requirement
Flight time with payload	> 20 min
Maximum payload	> 1 kg
Data transmission range	> 1 km
Size (diameter)	< 1.5 m
Price	< 30000 SEK
Camera	IR and EO
Multiple video streams	Available
Gimbal angles sensor data	Available
Data communication	Two way communication
Image processing, tracking and planning	On-board or on ground computer
Plug and play solution	Available

Both individual components and ready-to-fly platforms were considered. Discussions with several firms within the UAV market about product of theirs and product interfaces were also performed.

6.2.1 On-board computation

Having all computation on the UAV results in more payload since an extra computer must be added to the payload. Also the computer uses power which either has to come from the batteries powering the motors of the UAV or an additional battery has to be added to the payload. This means that the flight time of this alternative is reduced compared to the other alternative. The big advantage for on-board computing is that there is no risk of errors being introduced to the video stream getting while being wirelessly transmitted to the ground computer.

6.2.2 Computations on the Ground

By having the computations on the ground computer, more computational power is available and the flight time of the UAV is increased, compared to on-board computation. The drawback is that the computations are being made with the video that has been transmitted from the UAV. This communication is generally not lossless, instead redundancy must be used. Increased redundancy however also means increased latency.

6.2.3 Cameras and Gimbals

One of the requirements in Table 6.1 says that an EO and an IR camera should be available. For the tracking, planning and searching only a thermal camera is needed. The EO camera simplifies control for the operator, as it is easier to distinguish what an EO camera is pointing at compared to a thermal camera. This is completely natural as the eyes of humans detects the same wavelengths as an EO camera. When using a thermal camera it is easier to distinguish animals from the landscape due to the thermal difference. The animals will be seen as blobs in the pictures. These can easily be detected by using image processing. Thus, the use of a thermal camera instead of an EO camera is highly desirable.

There are some gimbals on the market where two cameras can be mounted. Most of these gimbals are used for one IR camera and one EO camera. These gimbals are naturally more expensive than single camera gimbals.

6.2.4 DIY Versus RTF Solutions

Another choice that has to be made is whether to buy different components and then assemble them together or buy a full package. The two alternatives are called do-it-yourself (DIY) for buying components and ready-to-fly (RTF) for buying a package. The advantages and disadvantages for each solution will be described below.

The DIY solution is more customizable since each component can be chosen. This makes it easier to fulfill the requirements on the system. It is also cheaper to buy the components than a package which means that the resulting UAV of equal price will have better specifications than an RTF solution. The drawbacks of the DIY solution is that a lot of effort would have to be made to get the UAV flying. Also, more hardware research would have to be made to ensure that the interfaces of the different components match.

The RTF solutions are more expensive but on the other hand less time is needed to get the UAV flying. Another major drawback is that a lot of RTF drones are hard to modify or make autonomous. Most RTF drones come with a remote controller from which it is difficult to get communication to a computer. This communication is necessary to perform image processing, tracking and planning on the ground.

The time is limited for this thesis project making an RTF solution more desirable. However, an RTF solution which fulfills the requirements to a great extent must be found. If this is not possible a DIY solution should be considered.

First, effort was made to find an RTF solution which would be suitable for this project. The setup which fulfilled the requirements to the greatest extent is described in Section 6.2.5. It was later discovered however that this solution did not yet officially support any thermal camera gimbals and that the payload limit of the UAV was too low to be able to carry the necessary devices for this project. Thus, a DIY alternative was instead considered. This alternative is presented in Section 6.2.6.

6.2.5 RTF Setup

The result of the research was a platform from DJI, one of the leading suppliers in the civilian-drone industry, called Matrice 100. This platform targets academic researchers and drone-application developers. The drone is customizable and additional equipment can be added onto the drone such as a DJI developed on-board computer. This on-board computer has been announced but yet not released to the market. There is also the alternative to connect an Android smartphone or tablet to the remote controller and run own-developed applications on these devices. To aid the developers DJI has an SDK (Software Development Kit) available for mobile applications. Since the on-board computer is not yet released the computations will be performed on the ground computer. However with this setup it is later possible to buy the on-board computer and perform the computations on-board. The specifications of Matrice 100 are shown in Table 6.2 [5].

Table 6.2: *Specifications for Matrice 100.*

Type	Specification
Weight	1755 g excluding batteries
Maximum weight	3400 g
Data transmission range	> 5 km
Diagonal wheelbase	350 mm
Diameter (including rotors)	996 mm
Price	3599 €
Flight Time	~16 min
Camera	Not included, prepared for one camera
Multiple video streams	Available
Gimbal angles sensor data	Available
Data communication	Two way communication
On-board computing	Possible with the not yet released Manifold computer.
Plug and play solution	RTF solution
SDK	Available for Android and iOS devices

There are two battery options available for the Matrice 100. The two options are compared in Table 6.3. Since the payload of the UAV is weight critical it is important to have as efficient batteries as possible. By looking at the capacity per weight, the TB48D is more efficient than TB47D making this a better alternative. However, there is the possibility for the UAV to carry two batteries, and thereby increasing the flight time, as long as the maximum weight is not exceeded.

The problem was then to find a gimbal which could be mounted on Matrice 100 and be able to carry a thermal camera. There were no gimbals carrying a thermal camera which was officially supported by DJI. However, some minor companies supplied thermal camera gimbals which would work together with Matrice 100. Unfortunately these were very expensive. This together with tight

Table 6.3: *Battery options for Matrice 100.*

Name	Capacity	Weight	Price	Capacity per weight
TB47D	4500 mAh	600 g	~$249	7.5 mAh/g
TB48D	5700 mAh	676 g	~$299	8.4 mAh/g

weight constraints made the Matrice 100 setup not useful for this project. Instead effort was made to find a DIY solution.

6.2.6 DIY Setup

In this section the different parts of the DIY setup is presented.

The Frame

The first step when constructing a DIY UAV platform is to select the frame. The frame is the backbone of the system and will set the limit regarding weight and payload constraints. DJI S900 was selected as the most promising alternative. As seen in Table 6.4 the payload that the frame can carry is greatly increased compared to the Matrice 100. It is also cheaper but there is a lot of additional equipment which is necessary to purchase.

Table 6.4: *Specifications for DJI S900.*

Type	Specification
Weight	3.3 Kg
Maximum Takeoff Weight	8.2 Kg
Diagonal Wheelbase	900 mm
Price	1190 €
Flight Time (depending on payload and batteries)	~15 min

Remote Controller

The UAV must be able to be controlled by an operator on the ground which means that a remote controller and a radio link is needed. A popular choice for the remote controller is FrSky Taranis Plus [7] which supports 16 channels. As both the UAV and the gimbal should be able to be controlled from the ground, two remote controllers are needed. The remote controllers include radio transmitters but radio receivers have to be bought as well, one for each radio transmitter. The chosen receiver, FrSky X8R [8], supports up to 16 channels.

The Autopilot

The frame does not come with an autopilot. The chosen autopilot is the popular open-source autopilot, the Pixhawk [14]. The Pixhawk itself contains a lot of sensors; accelerometers, gyro, magnetometer and barometer. However to enable the more advanced flight modes a GPS is also needed. The chosen GPS module [1] by 3DR is supported by the Pixhawk autopilot.

To get a reliable state estimation, the raw data from each of the sensors are fused together using an extended Kalman filter. The resulting state contains the velocity, the position and the attitude of the vehicle. The Pixhawk is connected the ESCs of the aircraft which in turn are connected to the motors. The Pixhawk is connected to a radio link such that it can be receiving commands from a remote controller on the ground. Also, the Pixhawk can communicate with an on-board computer through either a USB-port or a UART-port. The Pixhawk has a telemtry port which a telemetry radio link can be connected to for sending telemetry data down to the ground. The Pixhawk communicates via the communication protocol MAVLink (Micro Air Vehicle Link) [12] which is commonly used for UAVs. Through the Mavlink protocol, the Pixhawk can receive commands from a remote controller. Furthermore, the computer used in the setup will be communicating over this protocol. The computer will send waypoint commands to the Pixhawk and receive the estimated position and attitude of the Pixhawk.

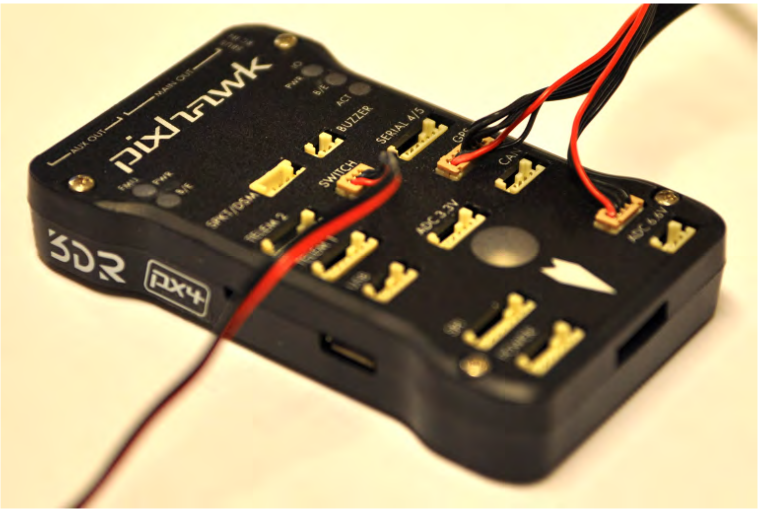


Figure 6.12: *The Pixhawk autopilot.*

Batteries

The UAV must also be powered. As the flight time of the DJI S900 is about 15 minutes, depending on the payload and equipped batteries, it is desirable to have multiple batteries available. This way the battery can be exchanged for another one when the power is low. Thus, enabling more flight time without having to wait for the battery to recharge. Three batteries should be sufficient for this project, as this enables the UAV to fly for about 45 minutes before having to recharge the batteries. There are a lot of batteries on the market. The chosen ones are Tattu 16 000 mAh [21] which are designed specifically to be used for UAVs. Also a compatible battery charger has to be purchased.

Thermal Camera

The next step is to choose a thermal camera. In the UAV industry almost all thermal cameras are uncooled due to the weight constraints. FLIR is the biggest supplier in the thermal camera industry and offers several cameras designed for UAVs. Table 6.5 compares the different camera options. The difference between the Vue and the Tau2 series is that the Vue series only provide analog video while it is possible to get digital video from the Tau2 series with an extension board. For this thesis project FLIR in Sweden has contributed by lending a FLIR Tau2 640 camera with a 19 mm lens. The camera can be seen in Figure 6.13.

Table 6.5: *Comparision of thermal cameras.*

Name	Weight	Price	Resolution	Op. Temp. Range
FLIR Vue 336	92.1 – 113.4 g	$1499	336x256	-20°C to +50°C
FLIR Vue 640	92.1 – 113.4 g	$2999	640x512	-20°C to +50°C
FLIR Tau2 324	~72 g	$2750	324×256	-40°C to +80°C
FLIR Tau2 336	~72 g	$2750	336x256	-40°C to +80°C
FLIR Tau2 640	~72 g	$7350	640x512	-40°C to +80°C

EO Camera

The EO camera will only be used as a reference video to compare the thermal video with after a flight when analyzing the behavior of the UAS. Thus, the video does not have to be sent to the ground via a video link but can be stored on a memory card in the camera. The camera should be light-weight and small. A good option which supports all these requirements is the GoPro Hero 4 camera [10]. This camera only weights 89 g and its dimensions are: 41.0 mm height, 59.0 mm width, 29.6 mm depth. It also has a wide FOV and a battery time, when recording video, of more than an hour. The GoPro is shown in Figure 6.13.

Gimbal

One of the main requirements for the gimbal is that it should be possible to acquire the attitude of itself. This is not very common in the UAV world as most operators are not interested in knowing exactly in which direction the gimbal is pointing, instead it is sufficient to assume that the gimbal is pointing in the direction that it is commanded to do. There is a gimbal controller card on the market

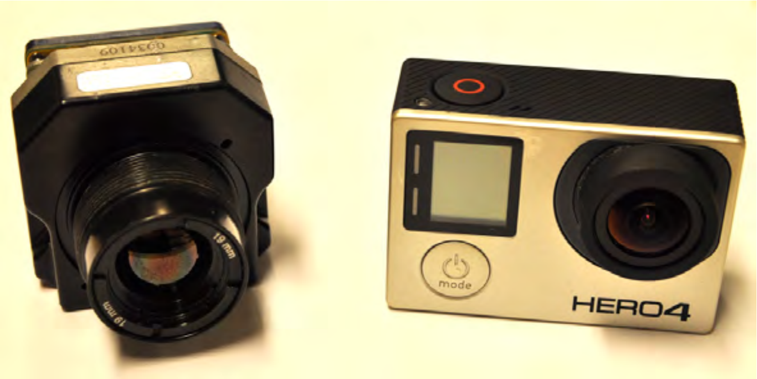


Figure 6.13: The FLIR Tau2 and the GoPro Hero 4.

called AlexMos which originates as an open-source product which can send the attitude of the gimbal to a computer connected through a USB cable. Through the protocol, [25], it is possible to control the gimbal. The gimbal controller can either set the gimbal to move with a constant rotation speed for one of its axes or make the gimbal point in a specific direction. Also, the protocol enables reading of attitude data from the gimbal. The gimbal should preferably be able to carry two cameras, an EO- and a thermal camera. One solution which offers both dual camera mounting and is equipped with an AlexMos gimbal controller card is a gimbal offered by the German company Global Flight Aerial Solutions. The gimbal, which can be seen here [6], supports the FLIR Tau thermal camera as well as GoPro 3 or 4 which is perfect considering the chosen cameras.

Video Link

The video link transmits the video from the thermal camera down onto the ground where it can be displayed to an operator. The chosen video link is DJI AVL58 [31] which has a transmission range of about 1 km. The transmitter only weights 39 g and it is powered by a 3S-6S Li-ion battery. The transmission frequency is 5.8 GHz and the video link is analog.

Video Converter

The video from the thermal camera is analog which is great for the video link which needs analog video. However, for the on-board computer, a problem occurs since the video must be in digital format to be able to be processed. Thus, a video converter is needed. The chosen video converter is the D-Link DVS-310-1 [4]. The analog input video is connected by a coaxial cable. The output stream is connected over Ethernet. The converter weights 232 g.

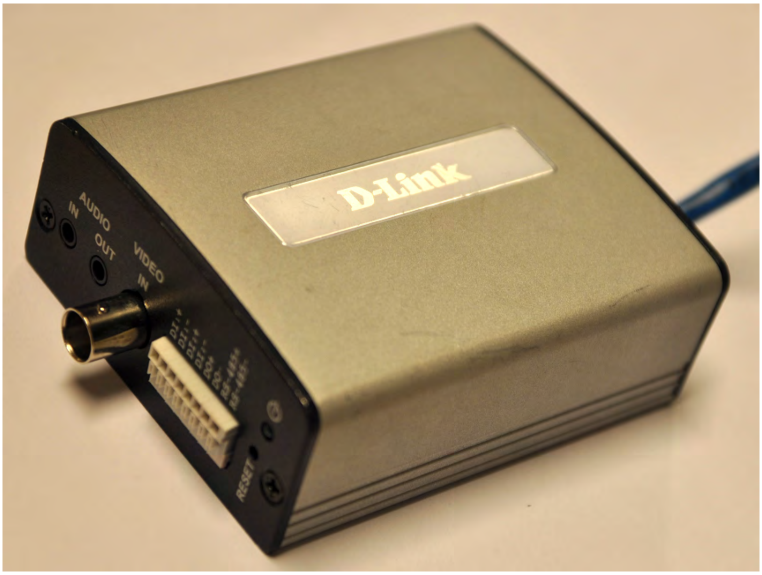


Figure 6.14: The D-Link DVS-310-1 video encoder.

On-board Computer

The on-board computer should be of low weight yet powerful. There are several computers which are supported by the Pixhawk. In Table 6.6 different computers are compared. The computer of lowest weight is the Odroid C1. However the computer which gives most computational power to weight is the Odroid XU4. The computer also runs on Linux which is the desirable operating system in this project.

Table 6.6: Comparison of on-board computers supported by Pixhawk.

Name	Weight, g	Price, SEK	Size, mm	OS
Raspberry Pi 2	45	374	85.6 × 56.5 x -	Multiple
Odroid XU4	60	902	82 x 58 x 22	Linux
Odroid C1	40	350	85 x 56 x -	Linux
Intel NUC	-	4129	115 x 111 x 51.6	Windows 10

6.2.7 Proposed Setup

The proposed setup for the project is summarized in Table 6.7. The total cost for the setup is 113065 SEK. However the camera has been borrowed by FLIR reducing the cost to 54365 SEK. The total weight of the UAV would be about 6.2 kg which is less than the maximum takeoff weight for the DJI S900 which is 8.2

kg. Figure 6.15 shows how the Pixhawk autopilot should be connected to the rest of the system.

Table 6.7: *The different components of the chosen setup.*

Type	Name	Price, SEK	Amount	Weight, g
Frame	DJI S900	11305	1	3300
RC	FrSky Taranis Plus	1800	2	N/A
Receiver	FrSky X8R	317	2	17
Autopilot	3DR Pixhawk	1195	1	38
GPS	3DR uBlox GPS	995	1	17
Battery	Tattu 16000 mAh	2695	3	1932
IR Camera	FLIR Tau2 640	58700	1	72
EO Camera	GoPro Hero 4	5149	1	89
Gimbal	Dual brushless gimbal	18350	1	~400
Video Link	DJI AVL58	1750	2	39
Video Converter	D-Link DVS-310-1	2400	1	232
Computer	Odroid XU4	902	1	60
Total:	-	113065	-	6196

6.3 Software

This section describes the different softwares used in the thesis project, both already developed programs and programs developed in the project are included.

6.3.1 Qt

The software in this thesis project is written in Qt [18] which is a popular C++ framework. The first version of Qt was released in 1995, and the framework has been developed continuously since then. In Qt a graphical user interface can easily be designed with Qt Designer. One of the most important Qt extensions of C++ is the signals and slots mechanism that simplifies event handling and callbacks. A signal in an object can be connected to a slot, which is a function, in another object, that will be run when the signal is emitted. Qt is a cross-platform framework working on several platforms including Windows, Linux, and OS X.

6.3.2 Ground Model Representation in the Software

The ground model is in the software represented by an 8-bit grayscale PNG file where each pixel holds the altitude at the position of the pixel. Since a bit depth of 8 is very small, the map is stored as

$$\text{PNG}(x, y) = 5\,(M(x, y) - 71) \tag{6.1}$$

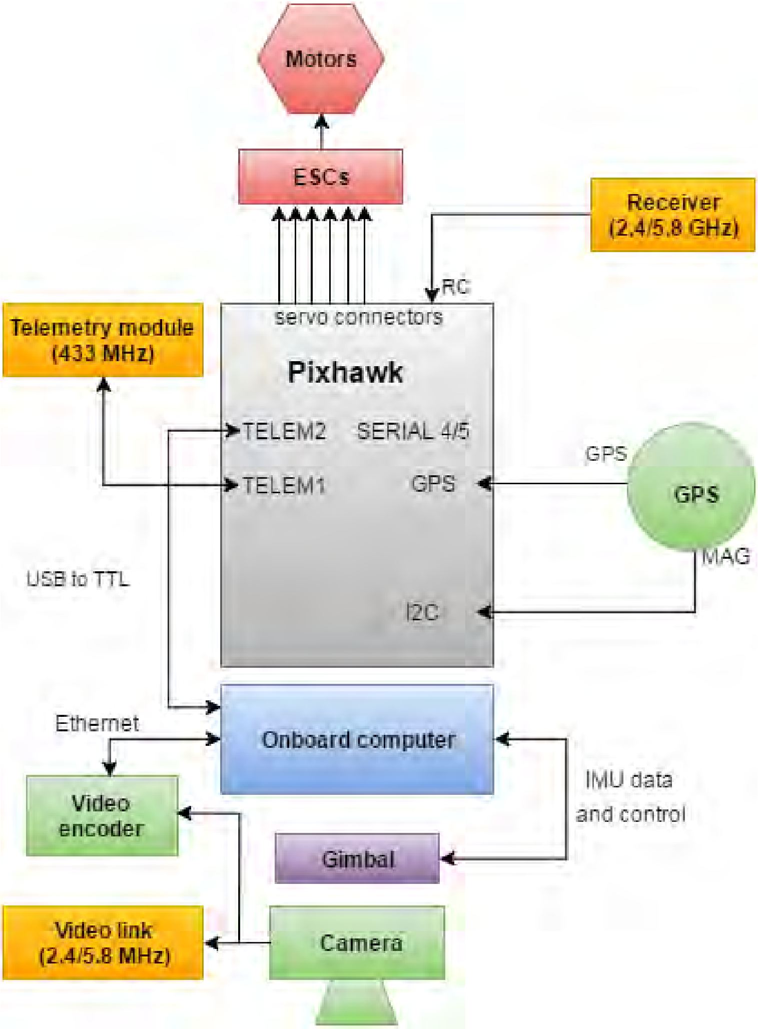


Figure 6.15: *The figure shows how the Pixhawk should be connected to the rest of the on-board system.*

where M is the original map from Section 2.1. The offset of 71 meter and multiplication by 5 is to get a integer representation, but retain precision. The value 71 is chosen because this is the lowest altitude in the map. The precision will be 0.2 meter and this representation will be able to hold altitudes between 71 and 122 meter. Figure 6.16 shows the PNG representation of the map, where darker areas are of lower altitude, and lighter areas are of higher altitude.

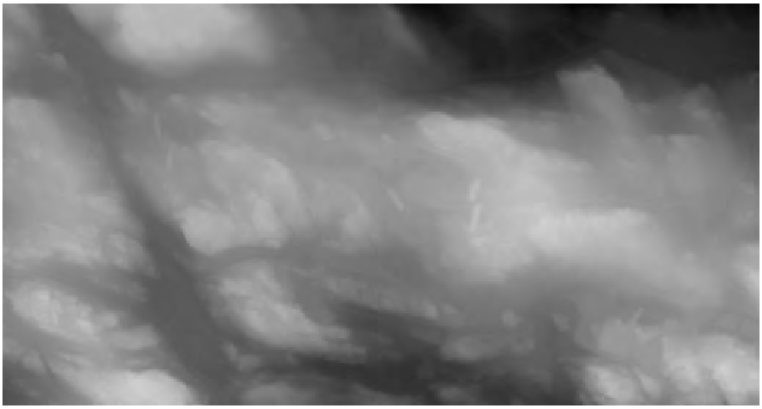


Figure 6.16: *The 8-bit grayscale PNG image. It can be seen as a topographic map, where high altitudes are lighter, and low altitudes are darker.*

6.3.3 The Program

The program is written in Qt on Ubuntu [22]. The reason for the use of Ubuntu as OS (Operating System) was to gain flexibility. Ubuntu is a common choice of OS for single board computers, which is the preferred type of computer to put on a UAV. However, it should be possible to cross-compile the application for Windows since Qt is a cross-platform framework.

Overview

An overview of the program can be seen in Figure 6.17. The program works as a small ground control station where received data can be recorded to a log file. Since the video stream also could be recorded, everything needed to recreate a scenario is logged. The video stream could also be processed in real time, for real time tracking. Parameters for the image processing can be changed by the user. The program also includes gimbal control in several modes. In the program information is displayed graphically for the user in a user interface.

User Interface

The program has a user interface, where the position of the UAV can be seen on a map. The footprint of the camera can also be seen. There are different settings

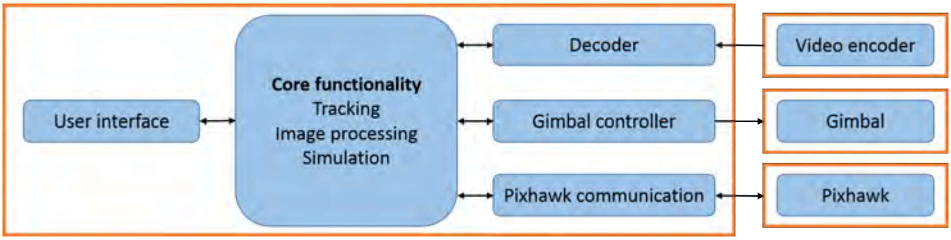


Figure 6.17: *An overview of the program. The logging functionality is included in the classes with incoming external communication.*

for the image processing unit, and also a window for gimbal control. The map window is the most important for the user, where the tracking of animals can be seen in real time. It includes controls where it can be selected how much filter information should be displayed.

Hardware Communication

For full functionality the program needs communication with a Pixhawk, a gimbal and a video encoder. The Pixhawk is connected through a USB, and the communication is performed with the MAVLink communication protocol [12] which is a header-only message library for small UAVs. By the use of different MAVLink commands, information about the state of the UAV can be requested and interpreted by the program. Through Mavlink, it is also possible to send waypoints to the UAV and thereby make the UAV move in patterns. The most important information needed from the Pixhawk is the position and attitude angles, which enables tracking and use of the ground model.

The program also has support for gimbal communication, where a gimbal with the BaseCam (formerly AlexMOS) brushless gimbal controllers [25] could be controlled. Two different control modes could be used: time based and adaptive gimbal control, described in Section 5.4. Communication with a video encoder connected to a thermal camera is needed for the tracking part. The video encoder is connected to the computer over Ethernet, and the video stream is transmitted with RTSP (Real Time Streaming Protocol).

Simulation from Data

To simplify the development process the program contains a simulation mode, when recorded video files and log files could be used to recreate a scenario. The data is time synced by the use of timestamps added in the data acquisition. An additional parameter can be used for manual sync, to increase precision.

Tracking

The whole chain from measurement to prediction is performed according to the descriptions in Chapter 3 and Chapter 4. The tracking uses particle filters for estimation of the animal trajectories. The image processing uses OpenCV [13], which is an open source computer vision library.

7 Results

In this chapter the results of the thesis are presented. First the resulting camera parameters are presented, followed by a comparison of two different LOS algorithms. Then, results from different image processing algorithms are shown, followed by simulation results in Matlab. The chapter is ended by presenting results from the system in practice at the test site Kolmården.

7.1 Thermal Camera Modeling

The resulting camera parameters from the *Camera Calibration Toolbox for Matlab* [28] are:

- $f_x = 1267$
- $f_y = 1117$
- $c_x = 356.0$
- $c_y = 216.6$
- $k_1 = 0.446$

The parameters α and k_2 to k_5 are assumed to be zero, and therefore not estimated in the modeling.

7.2 Comparison of Ray Casting and Extended Bresenham's Line Algorithm

An attempt has been made to determine which of the two algorithms, Extended Bresenham's Line Algorithm and Ray Casting, that is faster computationally. This test has been conducted by simulating in Matlab and measuring the execution time for each algorithm. It is also of interest that each algorithm behaves as the platform changes altitude relative to the ground, thus simulations has been made for multiple altitudes. The result can be seen in Figure 7.1 where the y-axis represents execution time and the x-axis represent the altitude of the platform relative to the ground. The simulations have been done using the 3D map of Kolmården to test the algorithms close to an environment where they will be used. The execution time for the different algorithms are very similar with respect to the platform altitude. Since the Ray Casting algorithm will be used to project ENU vectors to the ground, this is also the choice of algorithm for determining the LOS.

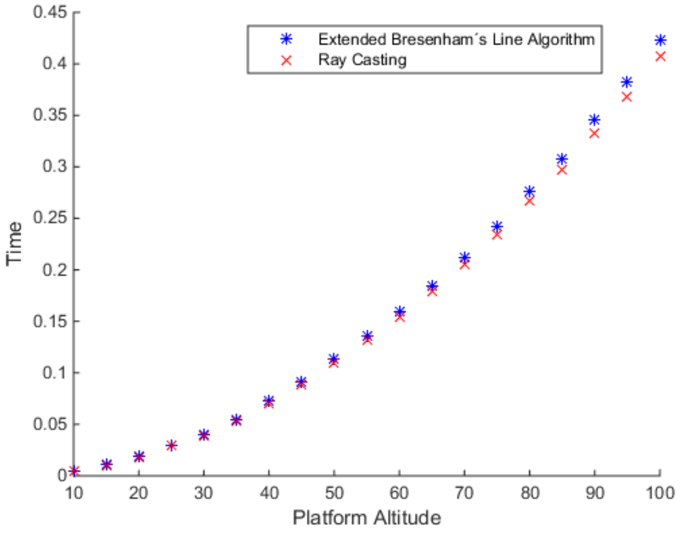


Figure 7.1: Result of simulations to compare the execution time for the Extended Bresenham's Line Algorithm versus the Ray Casting algorithm.

7.3 Choice of Hardware

A lot of time has been spent on finding hardware suitable for the project. The proposed setup however did not stay within the budget for the project. Instead a different approach had to be tried. Instead of using a UAV to mount the cameras and sensors on, a cheaper alternative had to be found. Fortunately, there is a gondola which travels around the safari area. The gondola is technically not a UAV but has the same behavior as a UAV flying a preset route. The difference is that this route cannot be chosen by the operator but is set beforehand. The gondola travels around the map with a fixed speed of 1.5 m/s. The trajectory of the gondola in the map can be seen in Figure 7.2, where it moves clockwise around the map.

By the use of the gondola instead of a UAV the amount of equipment needed to collect data was reduced, as only the data collecting parts were necessary. For data collection a camera, sensor package, and batteries were needed. The cam-

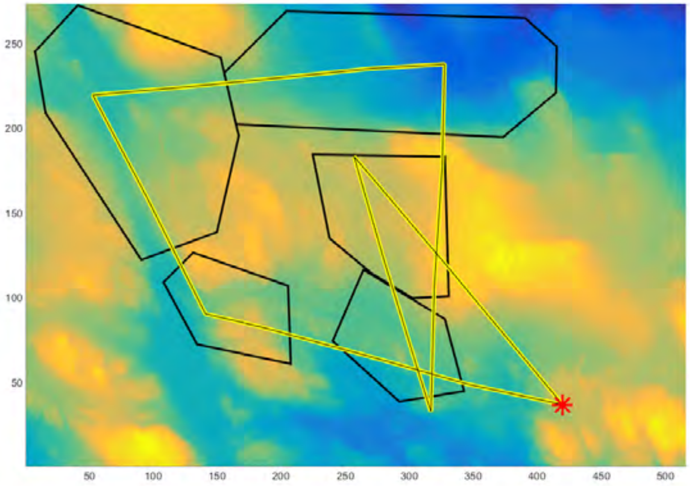


Figure 7.2: *The trajectory of the gondola in the Safari area of Kolmården. The red star marks the start and end point of the gondola.*

era and sensor package is mounted onto a gimbal. The gimbal used is a handheld 3-axis gimbal from Basecam. The camera and sensor package consists of two cameras, a GoPro Hero 4 and a FLIR Tau 2, and a Pixhawk autopilot. To the Pixhawk a GPS module is connected to provide position data. The Pixhawk is only used to collect data such as position and attitude data for the cameras. The cameras are mounted next to each other with the same attitude. The autopilot is also mounted with the same attitude and close to the cameras such that the measurements are accurate for the cameras. The camera and sensor package mounted onto the gimbal can be seen in Figure 7.3. The Pixhawk, the video encoder and the gimbal are all connected to a laptop where the software developed in the thesis project are running to collect data, perform tracking and visualize it in a user-friendly way.

7.4 Detection / Image Processing

The data acquisition mostly took place during cloudy weather, but some results were gathered in sunlight. An example of a thermal image from sunny weather is shown in Figure 7.4. Most of the other examples is from cloudy weather. Several methods and combinations of these were tested. Thermal enhancement and background subtraction are used to emphasize objects warmer than the background. A threshold could then be applied to retrieve a black and white image. This threshold could be constant or adaptive depending on a region around the pixel considered. When a threshold image is created it could be run through a contour detection, extracting the contours of the image.

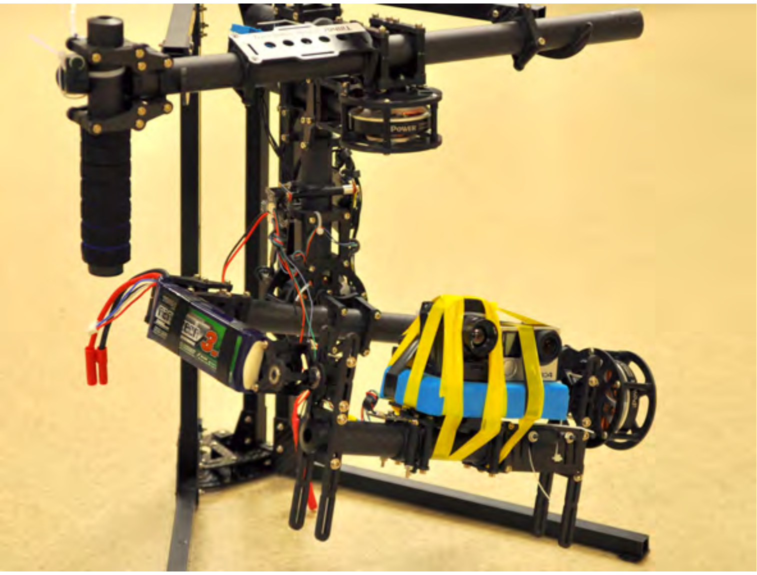


Figure 7.3: The gimbal with the camera, thermal camera, and Pixhawk mounted.

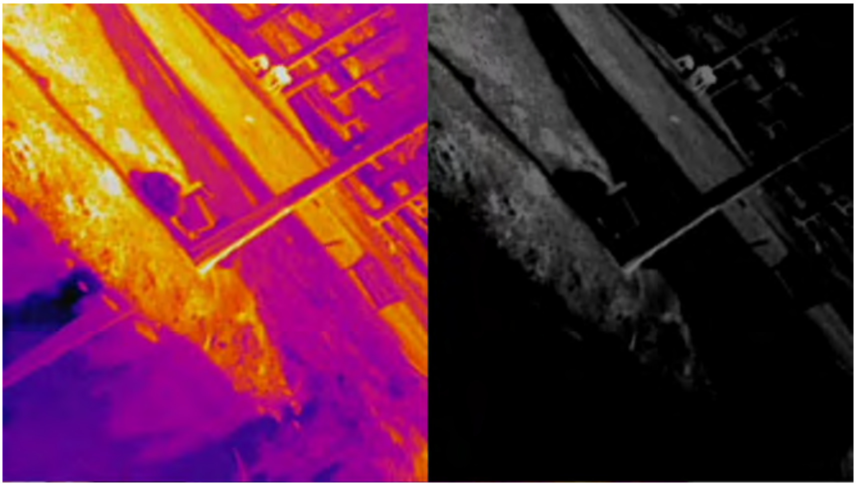
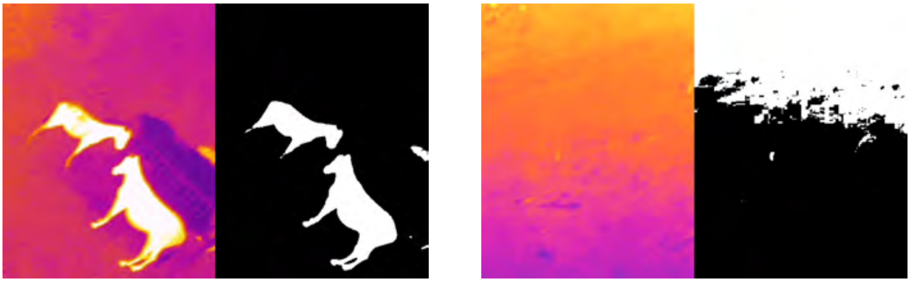


Figure 7.4: A thermal image captured in sunlight. In the grayscale image it is difficult to distinguish animals from hot parts of the ground.

7.4.1 Thresholding

In Figure 7.5 a constant threshold has been used without any background subtraction or thermal enhancement.



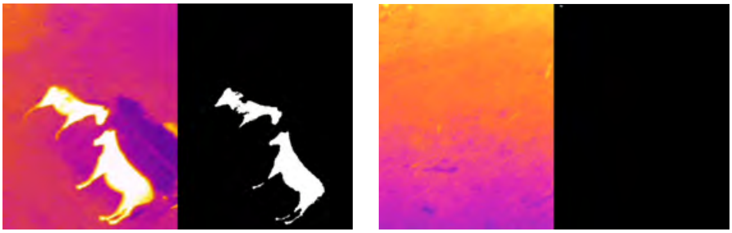
*(**a**) Constant threshold, val 150.*

*(**b**) Constant threshold, val 150.*

Figure 7.5: *Two examples from a sequence where the constant threshold is not enough. The thermal radiation from the background changes too much.*

7.4.2 Adaptive Thresholding

In Figure 7.6 an adaptive threshold has been used without any background subtraction or thermal enhancement.



*(**a**) Adaptive threshold, size 451, val 150.*

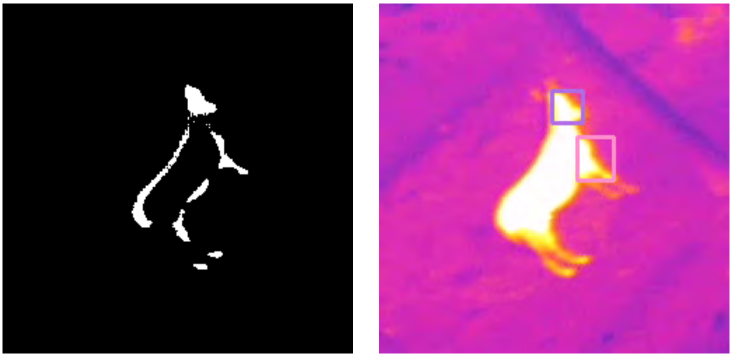
*(**b**) Adaptive threshold, size 451, val 150.*

Figure 7.6: *Two examples from a sequence where adaptive threshold handles changes in the background well.*

In Figure 7.7 and Figure 7.8 it can be seen that an adaptive threshold with a small region emphasizes the contour of the object. The adaptive threshold with a large region extracts the whole animal.

7.4.3 Thermal Enhancement and Background Subtraction

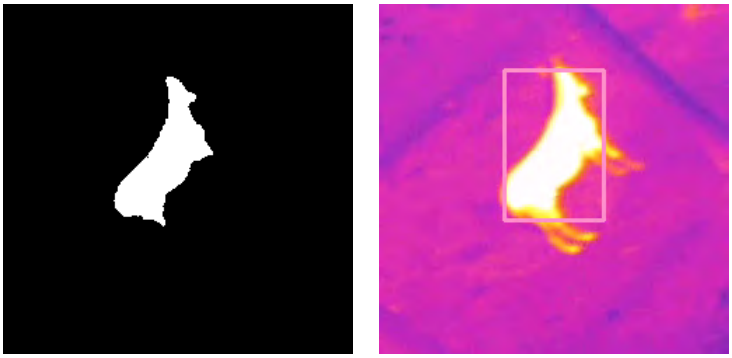
The thermal enhancement and background subtraction described in Chapter 3 are both methods used to extract objects warmer than the background from an



(a) *The thresholded image.*

(b) *The input image with detections.*

Figure 7.7: *Adaptive threshold with small region, size 51. The animal is segmented and generates several detections.*

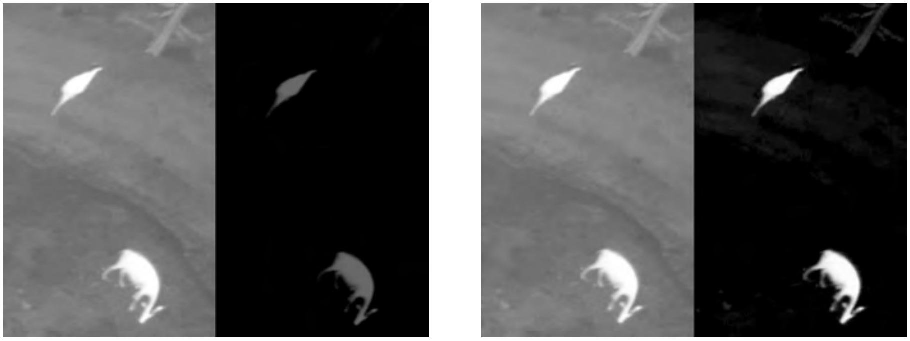


(a) *The thresholded image.*

(b) *The input image with detections.*

Figure 7.8: *Adaptive threshold with large region, size 251. The animal is one segment and generates one detection.*

image. The result can be compared in Figure 7.9. In the figures it can be seen that the thermal enhancement captures more of the background. The background dims the whole picture through subtracting each pixel intensity with the mean of the image. The result is a dim image with objects hotter than the background remaining.



(a) Using background subtraction to extract hot areas.

(b) Using thermal enhancement to extract hot areas.

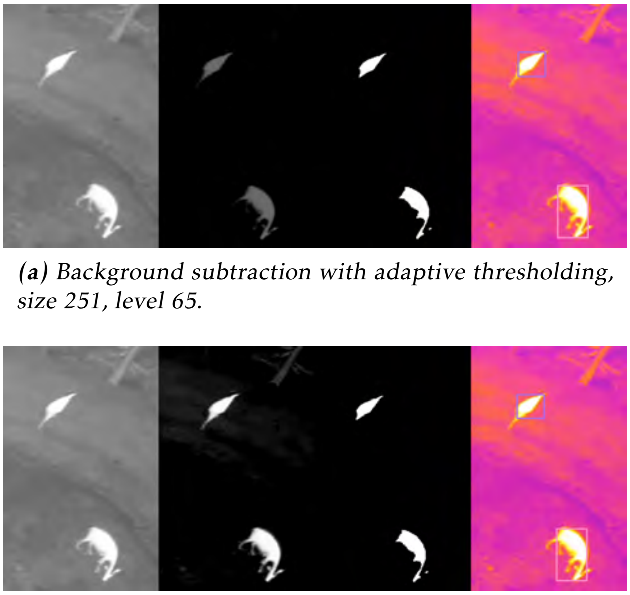
Figure 7.9: *The two methods for emphasizing objects warmer than the background.*

7.4.4 Combination of Different Methods

The combination of methods that have given the best results is thermal enhancement or background subtraction together with an adaptive threshold. In Figure 7.10 and Figure 7.11 a comparison between combinations of the methods can be seen.

7.5 Matlab Simulations

Simulations in Matlab have been done both for a UAV traveling around the map and for the gondola. Simulations of both ground coverage and tracking of animals are presented. The information based planning problem is not continuous, so optimizers in Matlab could not be used. In [46] this is solved by using shifted arctan functions to describe the FOV of the camera and thereby get a continuous problem to optimize. Effort has not been made to implement this in the thesis project due to time limitations. An information based approach to the planning problem is however an interesting approach which could improve the performance but with the drawback of being computationally expensive.



***(a)** Background subtraction with adaptive thresholding, size 251, level 65.*

***(b)** Thermal enhancement with adaptive thresholding size 251, level 160.*

***Figure 7.10:** A situation both methods work well.*

7.5.1 Simulations Using a UAV

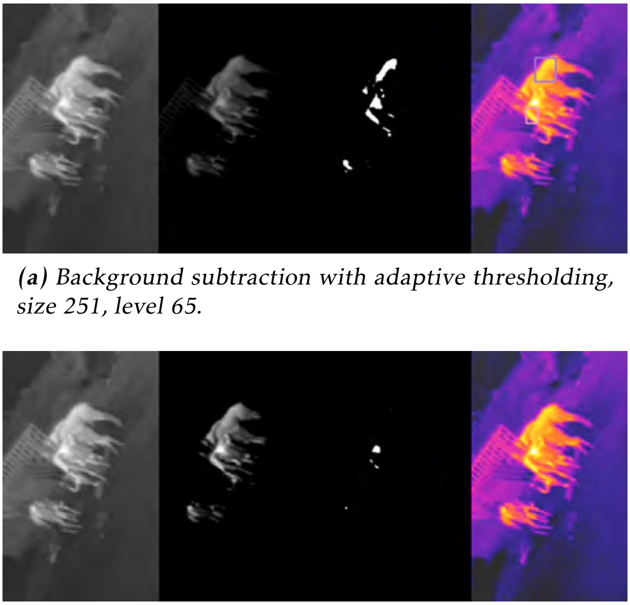
In this section results from simulations using a UAV as a platform is presented. The planning is done online with a high-level planner and pattern based planning is used to search for new targets.

Ground Coverage

This section shows simulations of ground coverage when using a UAV as the platform for the camera. In Figure 7.12 two gimbal strategies for the same planning conditions are compared, then the altitude and the properties of the lawn mower pattern are changed. By using the yaw rotation gimbal strategy, instead of having the camera directed straight down at all times, the ground coverage is increased significantly. Also, when the altitude of the UAV is reduced, the distance between the rows in the lawn mower pattern have to be reduced to get the same ground coverage.

Tracking

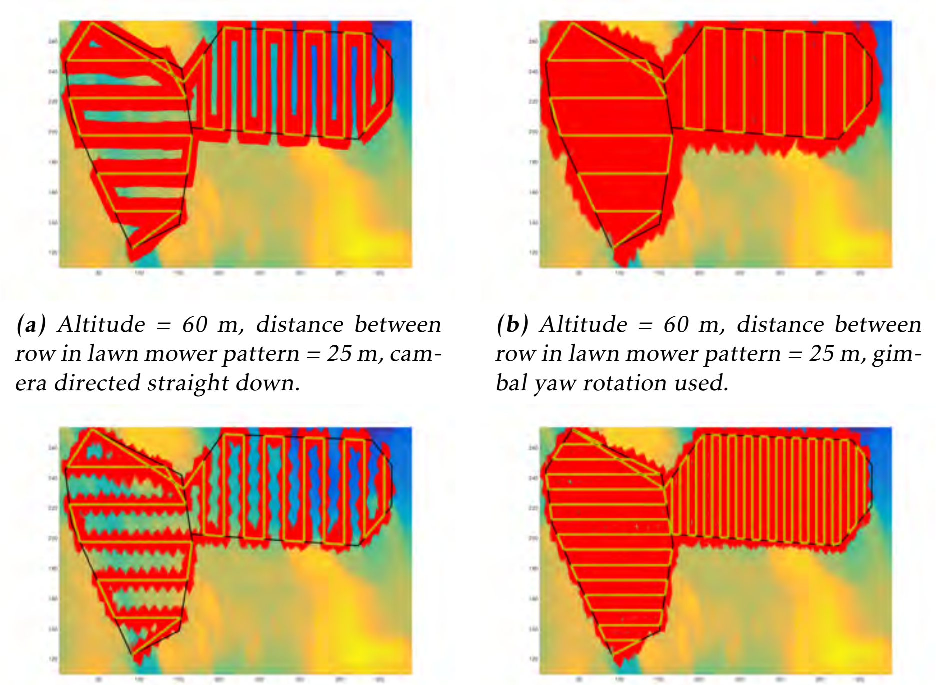
In this section some results regarding simulations of animal tracking are presented. In Figure 7.13 a full simulation is shown. The UAV moves across the map according to a lawn mower pattern, designed for full ground coverage within the enclosures. The altitude of the UAV is set to 50 m above the ground level. The



(a) Background subtraction with adaptive thresholding, size 251, level 65.

(b) Thermal enhancement with adaptive thresholding, size 251, level 160.

Figure 7.11: *A situation where the method with thermal enhancement misses detections.*



***(a)** Altitude = 60 m, distance between row in lawn mower pattern = 25 m, camera directed straight down.*

***(b)** Altitude = 60 m, distance between row in lawn mower pattern = 25 m, gimbal yaw rotation used.*

***(c)** Altitude = 30 m, distance between row in lawn mower pattern = 25 m, gimbal yaw rotation used.*

***(d)** Altitude = 30 m, distance between row in lawn mower pattern = 10 m, gimbal yaw rotation used.*

***Figure 7.12:** Ground coverage for the thermal camera when placed onto a UAV for different altitudes relative to the ground and different settings for the lawn mower pattern. When the altitude for the UAV is decreased, the distance between the rows in the lawn mower pattern have to be reduced as well.*

UAV uses a roll rotation strategy for the gimbal where the pitch limit $\theta_{max} = 60$. An animal move according to the random walk model, and the start location is randomized within its assigned enclosure. In Figure 7.14 a sequence of revisiting known targets are shown.

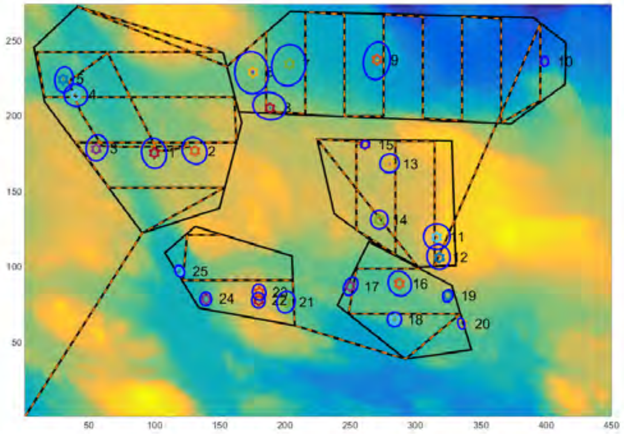


Figure 7.13: Tracking of animals using a UAV as the platform. Planning is done online, and pattern based planning is used to search for new targets.

7.5.2 Simulations Using the Gondola

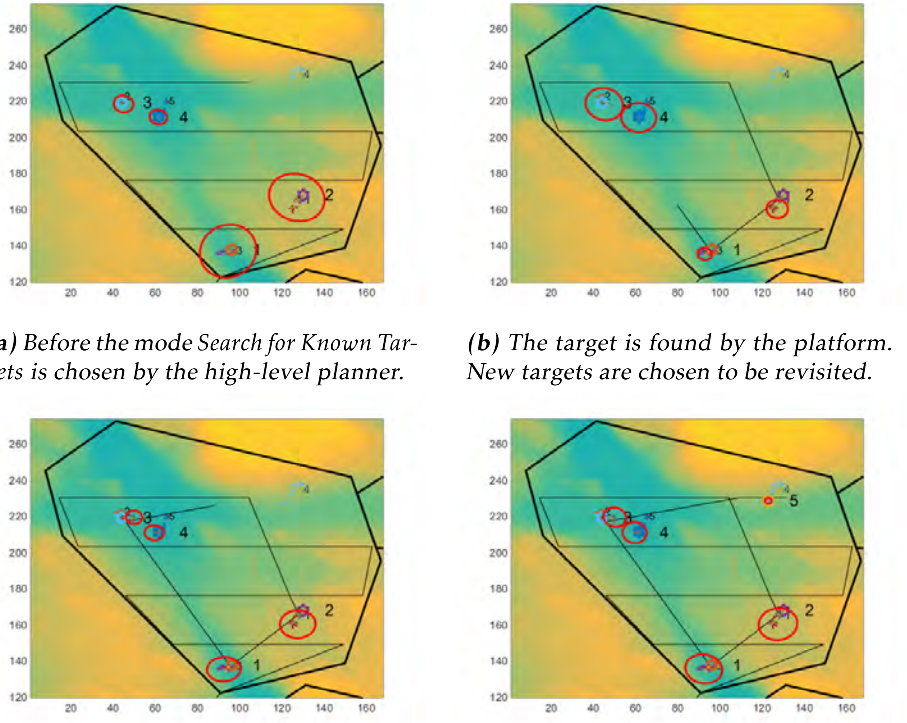
Here results from simulations when using the gondola as the platform are presented. First ground coverage for the different gimbal movement strategies are presented for some different parameter values defining the maximum rotation for each strategy. Then some examples of simulations when tracking animals from the gondola are presented.

Ground Coverage

Here simulations of the ground coverage using four different gimbal strategies are presented. In the following figures, the red color corresponds to FOV area at any time, the enclosures are marked in black and the trajectory of the gondola is given in yellow/black. First, the most basic gimbal strategy is shown in Figure 7.15. Here the camera is directed straight down at all times.

The second gimbal strategy presented, is to let the gimbal rotate around its yaw axis and having a fixed pitch angle. In Figure 7.16 the ground coverage for different pitch angles are shown.

The third gimbal strategy presented, is to let the gimbal rotate back and forth around its roll axis. The limit for the roll rotation is the same for both sides but



(a) *Before the mode Search for Known Targets is chosen by the high-level planner.*

(b) *The target is found by the platform. New targets are chosen to be revisited.*

(c) *All targets have been revisited. The mode is switched to Search for New Targets.*

(d) *A new target has been found by the UAV.*

Figure 7.14: *Sequence of revisiting known targets. The red ellipses represent the covariance of the tracks.*

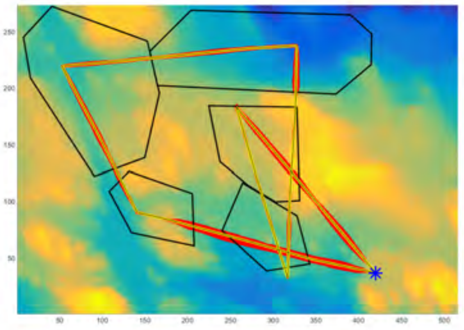
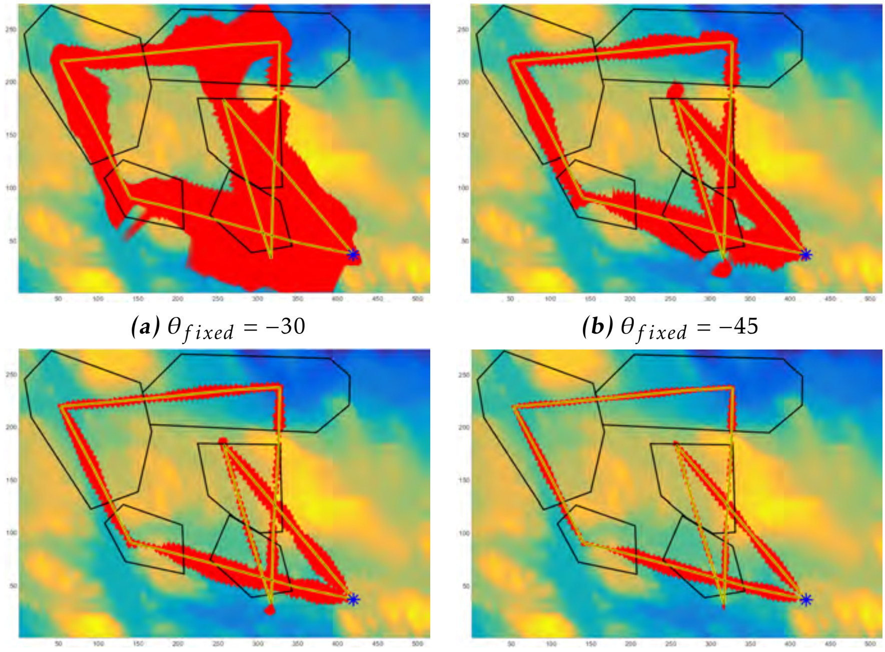


Figure 7.15: *Ground coverage when the camera is directed straight down at all times.*



(a) $\theta_{fixed} = -30$

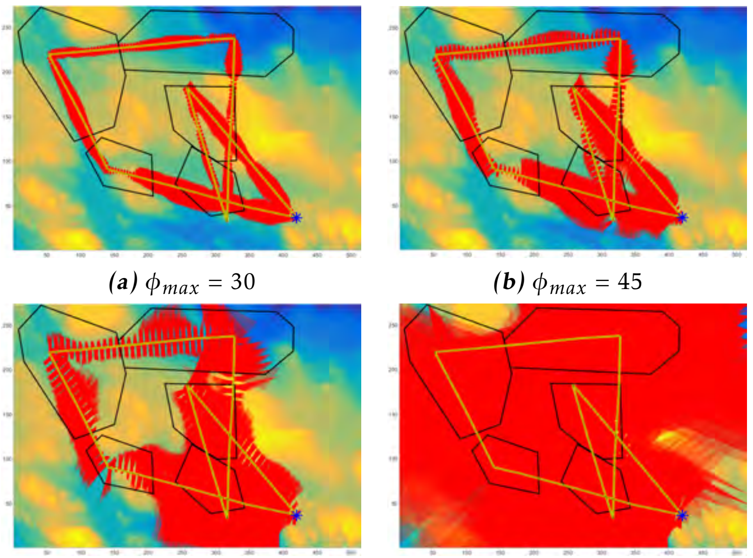
(b) $\theta_{fixed} = -45$

(c) $\theta_{fixed} = -60$

(d) $\theta_{fixed} = -75$

Figure 7.16: *The gimbal is rotating back and forth around its yaw axis. The yaw limit is 45 degrees for both sides for all figures while the fixed pitch angle, θ_{fixed}, is varying.*

varies in the presented subfigures. In Figure 7.17 the ground coverage for the different parameters are presented.



(a) $\phi_{max} = 30$ ***(b)*** $\phi_{max} = 45$

(c) $\phi_{max} = 60$ ***(d)*** $\phi_{max} = 75$

Figure 7.17: *The gimbal is rotating back and forth around its roll axis. The roll limit,* ϕ_{max}*, is varying in the different examples.*

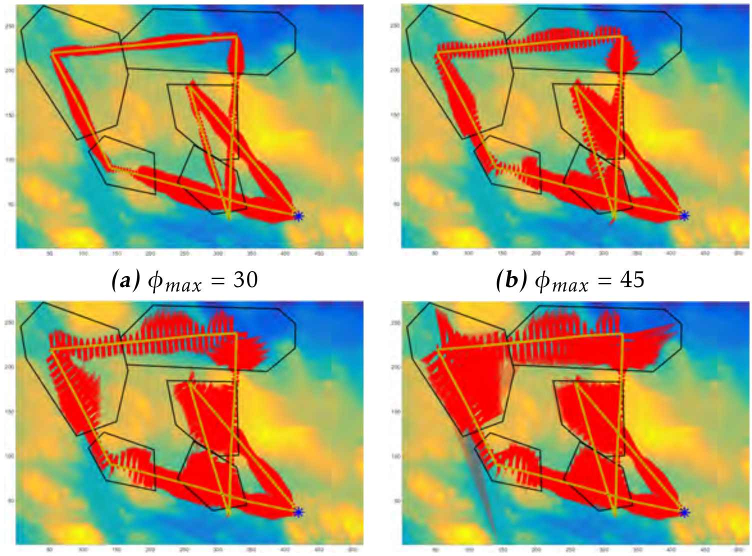
The final presented gimbal strategy, is to let the gimbal change rotation direction adaptively. The gimbal changes rotation direction when the FOV is not completely within any enclosure, or when the roll limit angle is reached. This is presented in Figure 7.18.

Tracking

This section shows simulation for tracking of animals when the camera is placed on the gondola. In each enclosure 5 animals are located. The trajectory of the animals are generated through a random walk motion model. The gimbal movement strategy is the adaptive roll rotation.

7.6 The Program

In Figure 7.20 a screenshot from the computer program can be seen. A number of targets have been seen, but only one of them is in the FOV and detected as a target. For more examples, see Appendix B.



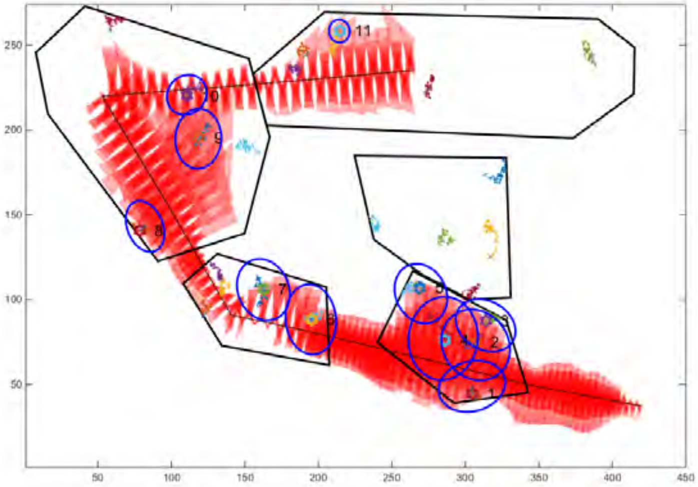
(a) $\phi_{max} = 30$

(b) $\phi_{max} = 45$

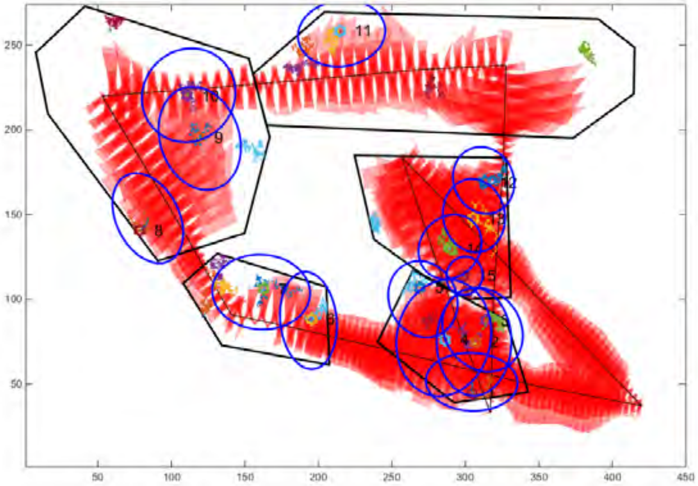
(c) $\phi_{max} = 60$

(d) $\phi_{max} = 75$

Figure 7.18: *The gimbal is moving according to the adaptive roll rotation strategy, described in Section 5.4.2. The maximum roll limit, ϕ, is varying in the different examples.*

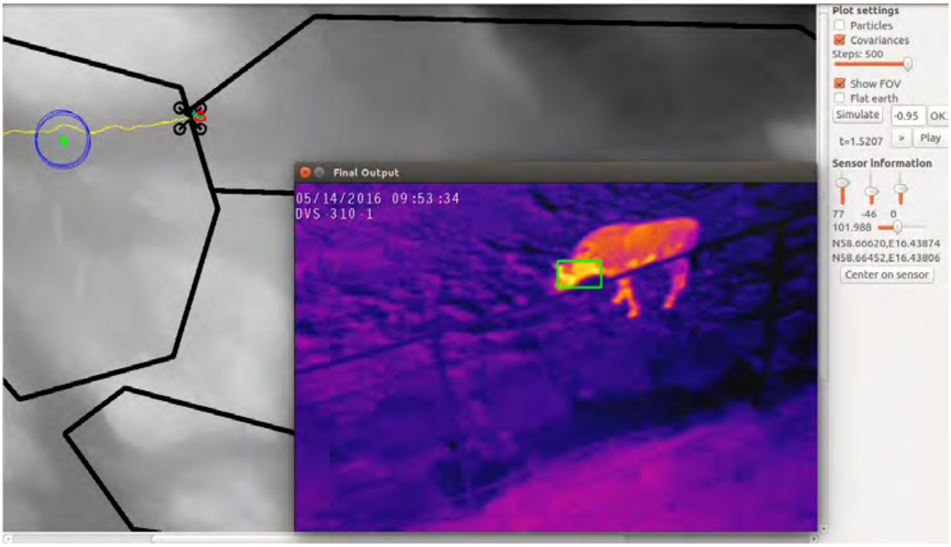


(a) The gondola is half-way through one lap around the map.



(b) The gondola has completed one lap around the Safari area.

Figure 7.19: *Simulation of tracking of animals from the gondola. In each enclosure five animals are located. They move according to the random walk model. It takes about thirty minutes to travel one lap with the gondola.*



Plot settings
☐ Particles
☑ Covariances
Steps: 500
☑ Show FOV
☐ Flat earth
-0.95
t=1.5207

Sensor information
77 -46 0
101.988
N58.66620,E16.43874
N58.66452,E16.43806

Figure 7.20: *A screenshot from the computer program.*

8

Conclusions

The goal for this thesis project was to develop an autonomous UAV with thermal imagery for wildlife surveillance. The greatest challenge was to find appropriate hardware to solve the other subproblems. Chapter 6 contains a thorough research and a result where suitable hardware is presented. Unfortunately there was a lack of time and money, making it difficult to implement this in practice. The other parts of the projects were still reviewed and this chapter contains conclusions for these parts. The chapter ends with a section containing approaches and ideas of improvements not tested.

8.1 Detection

In this thesis project the reason to use a thermal camera is that animals generate heat. Since they are considered to be warmer than the background, weather conditions are an important factor. Sunlight may break the assumption of animals being warmer than the background. This can be seen in Figure 7.4. Thus, if the system developed would be used on the savannas in Africa, it should be used at dawn and dusk, or in the dark. This may be the greatest advantage with a thermal camera, that animals could be detected without ambient light.

8.1.1 Image Processing Algorithm

Image processing has not been the main subject in this thesis project, so a good enough, easy implementation was preferred. The result from the simplest implementation, to only use a constant threshold, is shown in Figure 7.5. It does not perform well, since variations in the background is not considered. Thermal enhancement, background subtraction, and adaptive thresholding, are solutions to this problem. With adaptive thresholding, another problem arises. If the region size is chosen too small, the target could be divided into more targets, since hot areas in the middle of the animal could get a too high threshold value. This can be seen in Figure 7.7. For future improvement, Ray Casting could be used to determine how large the adaptive threshold region size should be, depending on the position and attitude of the sensor.

A combination of background subtraction and constant thresholding is easy to understand, and produces good results. Since the mean is subtracted from the image before the thresholding is done, variations in the background are taken into account. However, when many animals are present, the solution of subtracting the mean gets biased. The animals increase the mean and gives a higher threshold value than needed. A future improvement would be to take the median value instead of the mean.

8.2 Planning

This thesis project is a part of the earlier mentioned Project Ngulia where the goal is to protect wildlife, and especially rhinos, against poachers, by equipping the local park rangers with new technology. The technology should be easy to understand and use. Because of this, deterministic planning approaches should be preferred. While the information based planning approach has some possible interesting advantages, such as a lower search time, compared to the pattern based planning, it is not deterministic. A good first step, for Project Ngulia, is to have a platform search areas deterministically for animals or poachers.

In the ground model, trees are not taken into account. This means that densely forested areas will be occluded in reality, but in the calculations these areas will be seen as covered by the camera. The influence of not taking trees into account increases with the incidence angle of the camera to the ground. Thus, it is desirable to have a high altitude and have the camera directed straight down. When using the gondola as the platform the trajectory of the camera is determined by the gondola trajectory, and the altitude relative to the ground, varies between 3 to 20 meters. Here, larger rotations, and thereby larger incidence angles, are necessary to get a decent ground coverage.

The gimbal movement strategy that shows the most promise is the adaptive gimbal planning, as this minimizes the covering of the area outside of the enclosures, which is not of any interest. The adaptive gimbal planning can make use of yaw and roll rotations, respectively. Which of these that is most desirable depends on the situation.

When using the gondola as the platform, the ground coverage should be maximized given the constant speed of the gondola, at the same time as the incidence angle is minimized. In Figure 7.16 to 7.18, it can be seen that the yaw rotation when using a low pitch angle has the best ground coverage close to the trajectory of the gondola. The downfall of rotating around the yaw axis is that the heading of the platform has to be known. This can be achieved either by using the known trajectory of the gondola or having an IMU placed on the platform.

When using a UAV as the platform, it is easier to have a low incidence angle as the altitude of the platform can be varied. Another consequence of this is that the FOV is increased linearly with respect to the altitude. Thus, a high altitude is desired as long as animals can still easily be detected by the thermal camera. The gimbal movement strategy used here should not have large rotations, as the UAV itself can travel around the map.

8.3 Tracking

The particle filter performs well when it comes to tracking the animals. It is however, more computationally expensive than an EKF, but this has not been a limitation in this thesis project. The advantage of using a particle filter is that negative measurement updates can be used. A negative measurement update consists of reducing the weights of particles within the FOV when a measurement has not been associated to the track of those particles.

When using the gondola as the platform, the usage of tracking becomes limited as each animal is only seen for a couple of seconds. The advantage is that not just measurements are displayed but these are associated to tracks corresponding to animals. It takes about 30 minutes to travel one lap with the gondola, thus, tracks from the previous lap have very large covariances, since no measurement update has been performed for a long time. This means that it is hard to associate measurements from animals to the correct track one lap later. Essentially the same problem still holds for the case when using a UAV as the platform. The area to be searched is large compared to the FOV of the camera. Only a small part of the full area can be observed at once.

8.4 Future Work

A valuable improvement, that should result in much better classification is to fuse the information from the EO and the IR camera. Another improvement, regarding the image processing and classification, is to use shape recognition to detect animals. In combination with machine learning, the performance and efficiency of the image processing should increase.

Another improvement for the image processing when using adaptive thresholding is to vary the size of the region depending on the distance to the ground. The distance to the ground can be estimated by ray casting a pixel in the middle of the image. When the distance is large the region size can be decreased while it can be increased when the ground is closer.

In this thesis project, no absolute measurements were given by the thermal camera. Instead measurements were mapped onto a color palette. By having absolute measurements, areas too hot could be neglected, and thereby less false detections would occur. It would also be easier to set the threshold values for the image processing, making it easier to detect animals.

An interesting approach is to use a more advanced motion model in the tracking filter. When an animal is in sight, a constant velocity model could be used. When the animal is out of sight, the particles could be propagated with higher concentrations around more probable locations, such as water holes and common paths.

Another task left for the future, which was a part of the goal of this thesis project, is to implement the tracking, planning, and detection onto a UAS. The program could either be implemented on a ground computer or an on-board computer. The planning of the UAV could also consider locations of special interest, where animals often are located, such as water holes. The locations could then be visited more frequently than other places. A more advanced approach for the future is to have multiple platforms work together to search an area, for either poachers or animals. The platforms could then share information with each other to cover an area more quickly.

Appendix

A

Update of the Information Matrix

Let $[\mathcal{Y}]_{ij}$ be the element of the information matrix at row i and column j. Let also $R = \sigma_e^2 I_{2x2}$. According to [46] the update of the information matrix in (5.10) can be written as

$$
\begin{aligned}
[\mathcal{Y}_M^i]_{11} &= [\bar{\mathcal{Y}}_{0|0}^i]_{11} + \sum_{t=1}^{M} q_t^i \left(\sin(\phi_t^i)^2 + \cos(\phi_t^i)^2 \sin(\theta_t^i)^2\right) \\
[\mathcal{Y}_M^i]_{12} &= [\bar{\mathcal{Y}}_{0|0}^i]_{12} - \sum_{t=1}^{M} q_t^i \cos(\phi_t^i) \cos(\theta_t^i)^2 \sin(\phi_t^i) \\
[\mathcal{Y}_M^i]_{21} &= [\mathcal{Y}_M^i]_{12} \\
[\mathcal{Y}_M^i]_{13} &= [\bar{\mathcal{Y}}_{0|0}^i]_{13} + \sum_{t=1}^{M} q_t^i \cos(\phi_t^i) \cos(\theta_t^i) \sin(\theta_t^i) \\
[\mathcal{Y}_M^i]_{31} &= [\mathcal{Y}_M^i]_{13} \\
[\mathcal{Y}_M^i]_{22} &= [\bar{\mathcal{Y}}_{0|0}^i]_{22} + \sum_{t=1}^{M} q_t^i \left(\cos(\phi_t^i)^2 + \sin(\phi_t^i)^2 \sin(\theta_t^i)^2\right) \\
[\mathcal{Y}_M^i]_{23} &= [\bar{\mathcal{Y}}_{0|0}^i]_{23} + \sum_{t=1}^{M} q_t^i \cos(\theta_t^i) \sin(\phi_t^i) \sin(\theta_t^i) \\
[\mathcal{Y}_M^i]_{32} &= [\mathcal{Y}_M^i]_{23} \\
[\mathcal{Y}_M^i]_{33} &= [\bar{\mathcal{Y}}_{0|0}^i]_{33} + \sum_{t=1}^{M} q_t^i \sin(\theta_t^i)
\end{aligned}
\tag{A.1}
$$

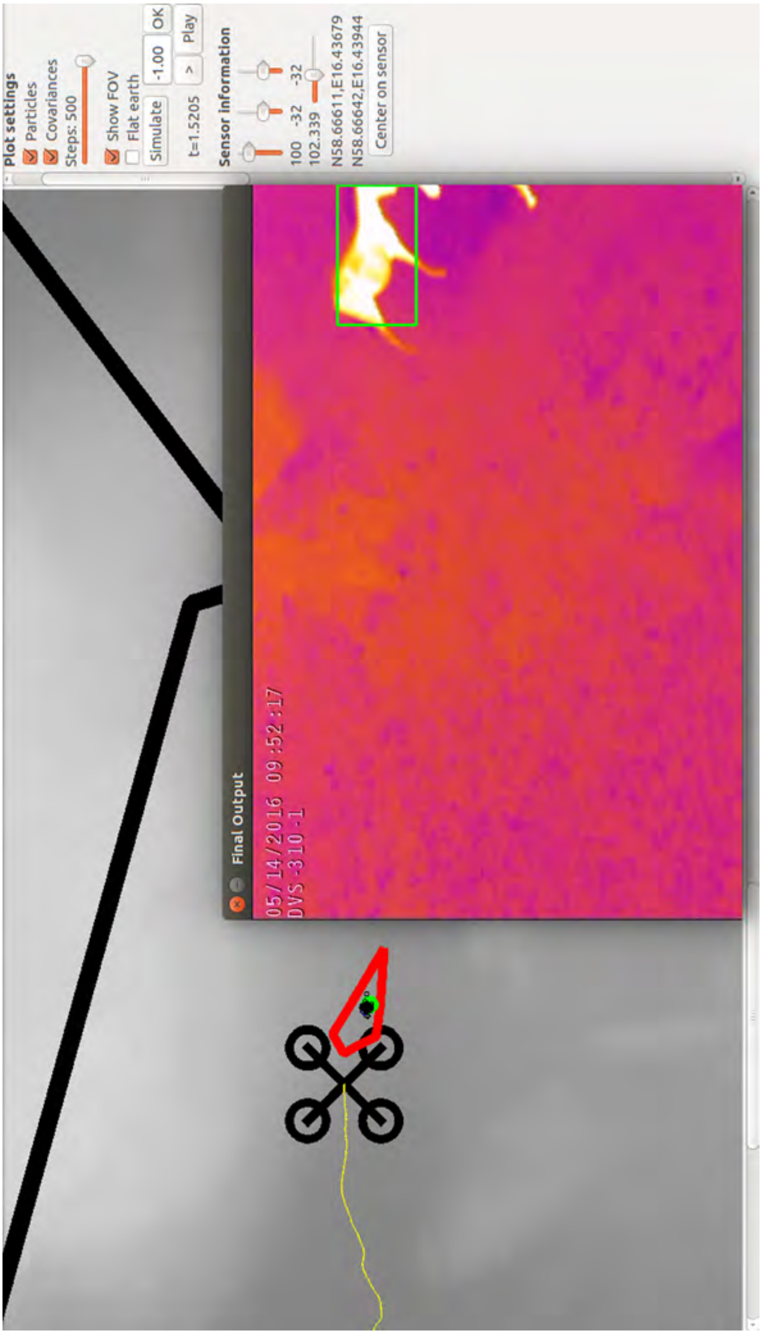
where ϕ_t^i and θ_t^i is obtained from a noise free measurement according to (5.7). The variable q_t^i is defined by (A.2) below.

$$q_t^i = \frac{P_D(X^i; X_t^c)}{\sigma_e^2 \left((x^i - x_t^c)^2 + (y^i - y_t^c)^2 + (z^i - z_t^c)^2 \right)} \tag{A.2}$$

B

Screenshots of the Computer Program

In Figure B.1 and Figure B.2 two screenshots from the computer program can be seen.



Plot settings

☑ Particles

☑ Covariances

Steps: 500

☑ Show FOV

□ Flat earth

-1.00

t=1.5205

Sensor information

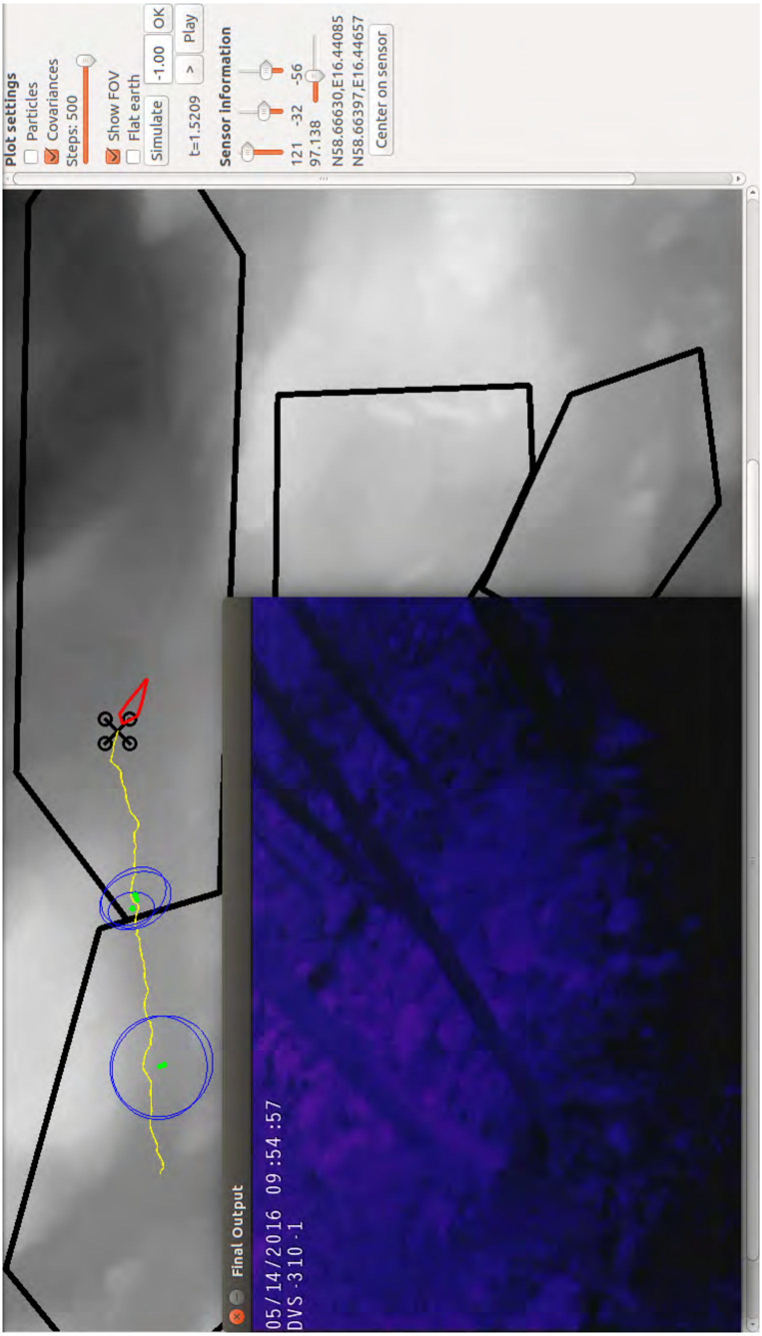
100 -32 -32

102.339

N58.66611,E16.43679

N58.66642,E16.43944

Figure B.1: A screenshot of the computer program. One animal is currently detected.



Plot settings

□ Particles

☑ Covariances

Steps: 500

☑ Show FOV

□ Flat earth

Simulate -1.00 OK

t=1.5209 > Play

Sensor information

121 -32 -56

97.138

N58.66630,E16.44085

N58.66397,E16.44657

Center on sensor

Final Output

05/14/2016 09:54:57

DVS-310-1

Figure B.2: *A screenshot of the computer program. No animal is currently in the FOV, but there are several tracking filters running.*

Bibliography

[1] 3dr gps module. https://store.3dr.com/products/3dr-gps-ublox-with-compass. Accessed: 2016-03-16. Cited on page 74.

[2] Amazon prime air. http://www.amazon.com/b?node=8037720011. Accessed: 2016-03-16. Cited on page 53.

[3] cvtColor from OpenCV reference. http://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous_transformations.html#cvtcolor. Accessed: 2016-06-03. Cited on page 26.

[4] D-link dvs-310-1 video encoder. http://www.dlink.com/uk/en/business-solutions/ip-surveillance/accessories/dvs-310-1--1-channel-h264-poe-video-encoder. Accessed: 2016-05-02. Cited on page 76.

[5] Dji matrice 100. https://developer.dji.com/matrice-100/. Accessed: 2016-01-22. Cited on page 72.

[6] Dual brushless gimbal for flir tau and gopro hero 3/4. http://www.globe-flight.de/Dual-brushless-gimbal-for-FLIR-Tau-336-and-GoPro-Hero-3/4. Accessed: 2016-05-02. Cited on page 76.

[7] Frsky taranis remote controller. http://www.frsky-rc.com/product/pro.php?pro_id=113. Accessed: 2016-03-16. Cited on page 73.

[8] Frsky x8r receiver. http://www.frsky-rc.com/product/pro.php?pro_id=105. Accessed: 2016-03-16. Cited on page 73.

[9] The global positioning system. http://www.gps.gov/systems/gps/performance/accuracy/. Accessed: 2016-05-17. Cited on page 23.

[10] Gopro hero 4 black edition. http://shop.gopro.com/EMEA/cameras/hero4-black/CHDHX-401-EU.html. Accessed: 2016-05-02. Cited on page 75.

[11] The lily drone. https://www.lily.camera/. Accessed: 2016-04-27. Cited on page 64.

[12] Mavlink protocol. https://pixhawk.ethz.ch/mavlink/. Accessed: 2016-05-02. Cited on pages 74 and 81.

[13] OpenCV. http://opencv.org/. Accessed: 2016-05-17. Cited on pages 25 and 82.

[14] Pixhawk. https://pixhawk.org/start. Accessed: 2016-03-16. Cited on page 74.

[15] Principle of a video encoder. http://www.axis.com/global/en/learning/web-articles/technical-guide-to-network-video/video-encoders. Accessed: 2016-05-02. Cited on page 63.

[16] Project Ngulia. http://projectngulia.org/. Accessed: 2016-01-21. Cited on page 1.

[17] Qgis. http://www.qgis.org/. Accessed: 2016-05-09. Cited on page 10.

[18] Qt. https://www.qt.io/. Accessed: 2016-05-13. Cited on page 78.

[19] Quadcopter project at Linköping university. http://www.isy.liu.se/edu/projekt/tsrt10/2015/quadcopter/. Accessed: 2016-04-27. Cited on page 64.

[20] Smart Savannahs. http://wildlifesecurity.se/smart-savannahs/. Accessed: 2016-01-21. Cited on page 1.

[21] Tattu plus 16000 mAh battery. http://www.gensace.de/tattu/tattu-plus-16000mah-22-2v-15c-6s1p-lipo-battery-pack.html. Accessed: 2016-03-16. Cited on page 75.

[22] Ubuntu. https://www.ubuntu.com/. Accessed: 2016-05-13. Cited on page 80.

[23] Unmanned aerial vehicle. http://www.thefreedictionary.com/Unmanned+Aerial+Vehicle. Accessed: 2016-04-27. Cited on page 54.

[24] What is gpu accelerated computing? http://www.nvidia.com/object/what-is-gpu-computing.html. Accessed: 2016-05-06. Cited on page 21.

[25] SimpleBGC 2.4 serial protocol specification, Basecam Electronics. http://www.basecamelectronics.com/files/SimpleBGC_2_4_Serial_Protocol_Specification.pdf, 2015. Cited on pages 76 and 81.

[26] H.L. Andersen. Path planning for search and rescue mission using multicopters. Master's thesis, Norwegian University of Science and Technology, 2014. Cited on page 5.

[27] S.S. Blackman and R. Popoli. *Design and analysis of modern tracking systems.* Artech House radar library. Artech House, Boston, London, 1999. Cited on pages 37 and 38.

[28] J.Y. Bouguet. Camera calibration toolbox for matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/parameters.html`. Accessed: 2016-05-05. Cited on pages 18 and 83.

[29] P.A. Buser and M. Imbert. *Vision.* A Bradford Book, 1992. Cited on page 61.

[30] G. Cai, B.M. Chen, and T.H. Lee. *Unmanned Rotorcraft Systems.* Advances in Industrial Control. Springer London, 2011. Cited on pages 9, 10, 12, and 14.

[31] DJI. *AVL58 5.8G Video Link User Manual*, 2012. Cited on page 76.

[32] FLIR Systems, Inc. *The Ultimate Infrared Handbook for R&D Professionals*, 2012. Cited on page 61.

[33] F. Gustafsson. Particle filter theory and practice with positioning applications. *IEEE Aerospace and Electronic Systems Magazine*, pages 53–82, 2010. Cited on page 33.

[34] F. Gustafsson. *Statistical Sensor Fusion.* Studentlitteratur, second edition, 2012. Cited on pages 23, 33, 35, 36, 38, and 44.

[35] Heikkilä, J. Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1066–1077, 2000. Cited on pages 16 and 17.

[36] G.M. Hoffmann, H. Huang, S.L. Wasl, and E.C.J. Tomlin. Quadrotor helicopter flight dynamics and control: Theory and experiment. In *In Proc. of the AIAA Guidance, Navigation, and Control Conference*, 2007. Cited on page 59.

[37] M.S. Jadin and S. Taib. Infrared image enhancement and segmentation for extracting the thermal anomalies in electrical equipment. *Elektronika ir Elektrotechnika*, 2012. Cited on page 27.

[38] D. Linden and T.B. Reddy. *Handbook of batteries.* McGraw-Hill handbooks. McGraw-Hill, 2002. Cited on page 68.

[39] Madan Lokesh, Kislay Anand, and Bharat Bhushan. Bresenham's lines algorithm. *International Journal of Research in Science and Technology*, 2014. Cited on pages 20 and 22.

[40] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models.* SpringerVerlag, 2003. Cited on pages 15 and 16.

[41] M.A. Ma'sum, M.K. Arrofi, G. Jati, F. Arifin, M.N. Kurniawan, P. Mursanto, and W. Jatmiko. Simulation of intelligent unmanned aerial vehicle (uav) for military surveillance. In *Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on*, pages 161–166, Sept 2013. Cited on page 5.

[42] W.M. Mularie. World geodetic system 1984. its definition and relationships with local geodetic systems. Technical report, National Imagery and Mapping Agency, 2000. Cited on page 9.

[43] M.A. Olivares-Méndez, T. Bissyandé, K. Somasundar, J. Klein, H. Voos, and Y. Le Traon. The NOAH project: Giving a chance to threatened species in africa with uavs. In *e-Infrastructure and e-Services for Developing Countries - 5th International Conference, AFRICOMM 2013, Blantyre, Malawi, November 25-27, 2013, Revised Selected Papers*, 2013. Cited on pages 4 and 54.

[44] Scott D. Roth. Ray Casting for Modeling Solids. *Computer Graphics and Image Processing*, 18(2):109–144, February 1982. Cited on page 20.

[45] C.K. Schmidt. Rhino and human detection in overlapping rgb and lwir images. Master's thesis, Linköping University, 2015. Cited on page 5.

[46] P. Skoglar. *Tracking and Planning for Surveillance Applications*. PhD thesis, Department of Electrical Engineering, Linköpings universitet, SE-581 83 Linköping, Sweden, 2012. ISBN: 978-91-7519-941-2. Cited on pages 5, 20, 42, 43, 44, 45, 46, 89, and 107.

[47] Sneider, JP. Map projections - a working manual. Technical report, Geological Survey (U.S.), 1987. Cited on page 11.

[48] F. Stendahl Leira. Infrared object detection & tracking in uavs. Master's thesis, Norwegian University of Science and Technology, 2013. Cited on page 4.

[49] David H. Titterton and J. L. John L. Weston. *Strapdown inertial navigation technology*. Progress in astronautics and aeronautics. American Institute of Aeronautics and Astronautics Stevenage, U.K. Institution of Electrical Engineers, Reston, VA, 2004. Cited on page 23.

[50] H. Wang, X. Yi, G. Huang, J. Xiao, X. Li, and S. Chen. IR microbolometer with self-supporting structure operating at room temperature. *Infrared Physics and Technology*, 45:53–57, January 2004. Cited on page 62.