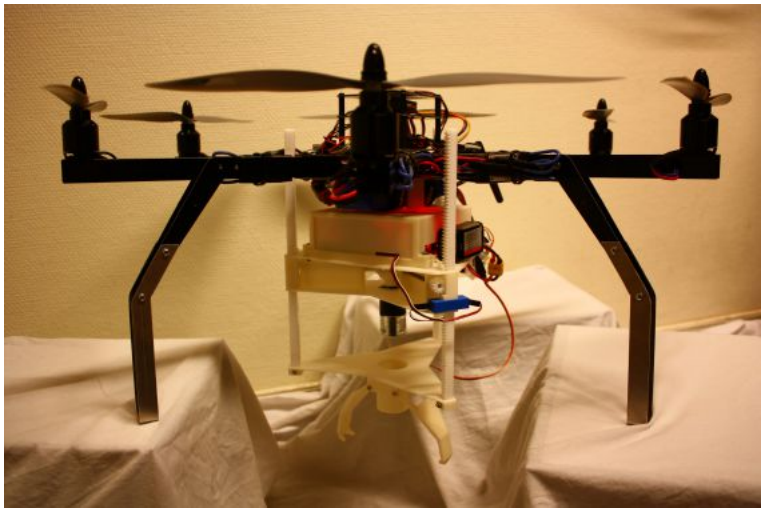
NTNU – Trondheim
Norwegian University of
Science and Technology

Mechanisms for Drop and Recovery of Sensor Nodes Using UAVs

Vegard Voldsund

December 2013

PROJECT THESIS
Department of Engineering Cybernetics
Centre for Autonomous Marine Operations and Systems
Norwegian University of Science and Technology



Supervisor 1: Professor Tor Arne Johansen
Supervisor 2: PhD Candidate Kristian Klausen

Problem Description

Mechanisms for drop and recovery of sensor nodes using UAVs (unmanned aerial vehicles)

Vegard Voldsund
Department of Engineering Cybernetics
NTNU

Project description:
Design and investigate methods for sensor pickup and deployment by multicopters (quad/hex). The following items should be considered:

1. Overall system description with detailed module interaction schemes and protocols.
2. Study and discuss different solution schemes for this kind of mission. Include both simple and complex mechanisms. Discuss necessary assumptions for the design of the sensor node for each proposed solution.
3. Design and create one of the proposed mechanical mechanisms for sensor deployment and pickup to be mounted on a hexacopter.
4. Study how the multicopter should approach the target for pickup/deployment.
5. Study and implement camera-based methods for node tracking needed for pickup.
6. The results should be verified by simulations. Major system components should be tested in experiments.
7. Conclude findings in a report. Include Matlab/C-code as digital appendices.

Supervisor: Tor Arne Johansen
Co-Supervisor: Kristian Klausen

Abstract

The goal of this project is to create a mechanism that can be attached to a multicopter unmanned aerial vehicle (UAV), to make it able to conduct drop and recovery of sensor nodes at sea.

Different design solutions are discussed before one is implemented and tested. The implemented design solution consists of a camera for detection of the sensor node and a lower-able gripper for drop and pickup of sensor nodes.

The use of camera to detect the sensor node gives a flexible system that does not depend on communication with the sensor node. An algorithm to get the position of the sensor node in relation to the UAV is developed and tested with good results. This algorithm will be used for feedback for the controller in the pickup phase.

The use of a gripper gives good accuracy in the pickup phase and flexibility to execute the mission in an efficient manner. The fact that the gripper can be lowered towards the water surface gives a safety margin above the sea surface for the UAV and simplifies operation.

Also the mechanical design is tested with good results, indicating that the implemented design solution will be able to conduct the mission of drop and recovery of sensor nodes in a good manner.

Sammendrag

(Norwegian translation of the abstract)

Målet med dette prosjektet er å designe en mekanisme som kan bli integrert med en multikopter UAV for å gjøre UAVen i stand til å slippe og plukke opp sensornoder fra sjøen.

Forskjellige designløsninger er diskutert og en er blitt implementert og testet. Den implementerte designløsningen består av et kamera for å detektere sensornoden og en nedsenkbar griper som skal slippe og plukke opp sensornoden.

Bruk av kamera for å detektere sensornoden gir et fleksibelt system som er uavhengig av kommunikasjon med sensornoden. En algoritme for å finne posisjonen til sensornoden i forhold til UAVen er utviklet og testet med gode resultater. Denne algoritmen vil bli brukt i tilbakekobling for regulatoren i plukke opp fasen.

Bruk av en griper gir god nøyaktighet i plukke opp fasen og fleksibilitet til å utføre oppdraget på en god måte. Løsningen med en nedsenkbar griper gir en sikkerhetsmargin til vannoverflaten som gjør bruken enklere.

Også det mekaniske designet er testet med gode resultater. Dette indikerer at den implementerte designløsningen vil kunne brukes til å droppe og plukke opp sensornoder på en god måte.

Preface

This project is written as a part of my MSc degree at the Department of Engineering Cybernetics at NTNU and is part of research conducted by the Center of Autonomous Marine Operations and Systems (AMOS). The work in this project will be continued into my master thesis.

The execution of this project would not have been possible without the financial backing from AMOS and the usage of equipment at the electronics and mechanical workshops at the Department of Engineering Cybernetics, NTNU. I will like to thank the guys at these workshops for valuable help and guidance.

I would also like to thank my supervisors Professor Tor Arne Johansen and PhD Candidate Kristian Klausen for guidance and encouragement along the way.

Vegard Voldsund

Acronyms

AMOS	Centre for Autonomous Marine Operations and Systems
APM 2.6	Ardupilot Mega 2.6
BGR	Blue-Green-Red
CEP	Circular Error Probability
CPU	Central Processing Unit
DH Convention	Denavit-Hartenberg Convention
ECEF	Earth-centered Earth-fixed
GPS	Global Positioning System
HSV	Hue-Saturation-Value
IC	Integrated Circuit
IMU	Inertial Measurement Unit
I/O	Input/Output
IR	Infrared Radiation
ISA	Inertial Sensor Assembly
LBP	Local Binary Patterns
LED	Light Emitting Diode
MAVLink	Micr Air Vehicle Protocol
MEMS	Microelectromechanical Systems
MIPI	Mobile Industry Processor Interface
NED	North-East-Down
OpenCV	Open Source Computer Vision Library
PWM	Pulse-Width Modulation
RAM	Random Access Memory
SIFT	Scale-Invariant Feature Transform
SURF	Speeded-Up Robust Features
UART	Universal Asynchronous Receiver/Transmitter
UAV	Unmanned Aerial Vehicle
WGS-84	World Geodetic System 84

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and motivation

Unmanned aerial vehicles (UAVs) are aircraft without a human operator on board. They can fly autonomously or be remotely operated by a pilot on the ground. The use of UAVs have exploded in recent years. This is due to a lot of ongoing research and development conducted by companies, scientists and enthusiasts. Some of this research and development have resulted in open source projects that have made advanced UAV technology availably at very low cost.

The UAV used in this project will be based on one of these open source UAV solutions. This solution is chosen because it is cost efficient, results of the project will be easy to use by others, and it is a way to get going fast and still be able to do some customization.

This project is a part of research conducted by the Center of Autonomous Marine Operations and Systems (AMOS) at NTNU. AMOS is now establishing a new laboratory with field experimental capabilities (UAV-Lab). This project is related to this lab. To be a part of the research at AMOS opens up a lot of opportunities, but it also means that some things have to be standardized. For instance the choice of UAV and the use of PandaBoard as onboard computer follows standards decided by AMOS.

The goal of this project is to create a mechanism that can be used for sensor node pickup and deployment by the use of multicopters. The sensor node will be a lightweight packet that can contain different sensors depending on the mission. These sensor nodes will be dropped into the sea where they will float on the surface. Examples of use for the sensor nodes can be to log temperature, currents, salinity or water quality. Hence they can be very useful in for instance climate research or for detecting oil spills.

Both fixed-wing and multicopter UAVs will be part of the sensor node pickup and deployment system and supplement each other in the UAV-lab. Multicopters will be used in coastal areas and at relatively good weather conditions, while fixed-wing UAVs have a much greater

range and will be used for longer missions and in rougher weather conditions.

The project will be continued into a master thesis where the created mechanism will be used for sensor node pickup and deployment. This means that the goal of this project is not to create a fully operational system with seamless integration between the modules, but to make different modules that are ready to be used in the master thesis. This report should be read in this context.

1.2 Previous work

As already mentioned there has been a lot of research on UAVs in recent years, but most of this research have focused on fixed-wing aerial vehicles. And the applications have usually been limited to monitoring and search [Mellinger et al., 2011].

The latest couple of years have shown some research on rotary-wing aircraft interacting with the environment in different ways. There have been conducted research where the UAV is used to manipulate its environment. Two examples of this using two different strategies are [Jiang and Voyles, 2013] and [Lippiello and Ruggiero, 2012]. Lippiello and Ruggiero are inspired by the use of impedance control in robot manipulation tasks, while Jiang and Voyles alters a hexacopter by tilting the motors, making it possible to create a horizontal force without tilting the hexacopter.

The most relevant applications to highlight in this report are those that use different strategies for drop off and pickup of objects. Possibilities of using an avian catching fish as an inspiration for a quadcopter based gripper system to be able to execute high speed pickup have been explored by [Thomas et al., 2013]. A gripper on the end of a link with two joints is used to replicate an avians leg and claw. Good results were achieved as the quadcopter was able to grasp targets at speeds up to 3 m/s. The trajectory of the pickup was decided pre run time to match the trajectory of the avian. Feedback to the controller was given using VICON[1].

The use of a helicopter to grasp objects on the ground are explored in [Pounds and Dollar, 2014]. The ability to grasp objects of different shapes and without a perfect lined up position is in focus for the gripper design. Tests demonstrated an ability to grasp objects of different shapes and sizes under human control of the helicopter. The most difficult objects were grasped 67 % of the time, while the easiest object was grasped 100 % of the time.

Estimation of payload parameters are explored in [Mellinger et al., 2011] as well as mechanical design and controller for aerial grasping. Relevant parameters to estimate are mass and inertia. These estimates can be used to adapt the controller and to check if the object was successfully picked up or not.

[1] A very accurate external motion capture system

1.3 Contribution and scope of this report

Some of the research presented in the previous section picks up objects lying on the ground (in some of them at exactly known positions). The fact that the aim in the master project is to pick up sensor nodes that are floating affected by waves, wind and currents add another dimension to the challenge. Navigation at sea means that one can not rely on external sensor systems like for instance VICO. This means that accurate positioning above the sensor node will be a challenge. The UAV will have to rely on poor position measurements and some added sensor system, for instance camera, to close the loop between the UAV and the sensor node.

This project aims to connect known techniques and theory from different fields in new ways to make pickup and deployment of sensor nodes by the use of UAVs at sea possible.

1.4 Organization of this report

The UAV used in this project is described in Chapter 2. This description includes the most relevant features to give the reader insight into which possibilities that lie within the UAV. The PandaBoard that is used as onboard computer is also described for the same reasons.

Different possible design solutions are described in Chapter 3. The chapter starts with some design choices that are common for all the solutions and continues with specifics about the different proposals. Both advantages and limitations of the different solutions are considered before the chapter ends in a conclusion of which design solution to implement.

An introduction to the relevant reference frames used for navigation, control and object tracking, as well as some features of the open source computer vision library OpenCV are given in Chapter 4.

Chapter 5 presents the implementation of the chosen design solution. It covers mechanical design, node tracking using camera, communication between the different modules, some circuit design and power supply.

The tests of the design solution with associated results are described in Chapter 6. Some of the key features that are tested are weight limitations on the sensor node, accuracy of the node tracking and robustness of the mechanical design.

The implemented design solution is discussed in Chapter 8, before conclusions are drawn in Chapter 9.

The digital appendix included with this report contains source code used for the experiments and the node tracking algorithm. Solidworks models of the 3D-printed parts are included as a resource if the reader wants to replicate the design solution.

Chapter 2

Description of the UAV

The UAV used in this project as a base for the pickup and deployment mechanism is an ArduCopter Hexacopter (see Figure 2.1). The ArduCopter uses the ArduPilot Mega 2.6 flight control unit (hereinafter referred to as APM). The UAV will also be equipped with a PandaBoard as an onboard computer.

Relevant features of the APM and the PandaBoard are described below.

Figure 2.1: ArduCopter Hexacopter *Courtesy arducopter.co.uk*

2.1 APM

The APM is an open source flight control unit supporting multicopters, traditional helicopters, fixed wing aircraft and rovers [ArduPilot, 2013]. Software is developed and supported

by the DIYDrones community. At the moment the community has more than 45 000 members (November 2013) and an active forum where one could get help and useful tips and tricks.

2.1.1 Modes of Operation

The APM can operate in many different modes of operation [Ardupilot, 2013]. The most relevant for this project are:

- Stabilize - Manual flight mode that automaticly levels the UAV and maintains the current heading
- Auto - The UAV tracks predefined waypoints
- Guided - The next waypoint is defined in flight
- RTL (Return to Launch) - The UAV returns to the position where it was armed and hovers
- LAND - The UAV lands, shut-down the motors and disarms

Control signals in the different modes are given either with PWM-signals[1] or through serial communication using the MAVLink protocol (the MAVLink protocol will be briefly explained below). The PWM-signals are usually sent from a 2.4 GHz radio via a receiver on the UAV, while the serial communication is usually sent from a ground station via a telemetry link to the APM. These signals could easily be replicated by the PandaBoard. A picture of the APM with a voltage regulator ann an external magnetometer and GPS module is found in Figure 2.2.

Figure 2.2: APM 2.6 with a voltage regulator and an external magnetometer and GPS module *Courtesy diydrones.com*

[1]Pulse Width Modulated signals

2.1.2 Sensors

The APM is equipped with several sensors that are utilized for navigation and control. These will be briefly explained below.

Barometer

A barometer is an instrument that is used to measure air pressure [Merriam Webster, 2013]. The barometric formula

$$p(h) = p(0)e^{-\frac{mgh}{kT}} \tag{2.1}$$

relates the pressure $p(h)$ of an isothermal, ideal gas of molecular mass m at some height h to its pressure $p(0)$ at height $h = 0$, where g is the acceleration of gravity, k the Boltzmann constant, and T the temperature. This formula applies reasonably well to the lower troposphere. For altitudes up to 6 km the error is less than 5 % [Berberan-Santos et al., 1997].

The barometer in the APM is based on piezoresistive technology. Piezoresisivity is a common sensing principle for micro machined sensor [Liu, 2011] that uses the fact that resistivity of some materials changes with applied stress [Mason and Thurston, 1957]. This feature is used in the barometer, when the air pressure varies, the pressure on the material in the barometer varies which means that resistivity varies. A mapping from resistivity to pressure is used to calculate altitude referenced to start altitude. Altitude calculations by the use of barometers can be sensitive to changing weather conditions.

The barometer in the APM is the MS5611-01BA03 by Measurement Specialties, which according to the producer has a resolution of 10 cm.

Magnetometer

A magnetometer is an instrument for measurement of magnetic fields. Depending on the setup they can measure strength of a magnetic field or both strength and direction of the field [Store Norske Leksikon, 2013]. The magnetometer in the APM is a three-axes magnetometer. This means that both the strength and direction of the magnetic field can be measured. The magnetometer measures the force created by the magnetic field on an energized conductor. This force is called the Lorentz Force and follows the formula

$$\boldsymbol{F} = q\boldsymbol{v} \times \boldsymbol{B} \tag{2.2}$$

where q is charge, $\boldsymbol{F}$ is the Lorentz Force and $\boldsymbol{B}$ is the magnetic field. The charge q is assumed to be known and $\boldsymbol{F}$ can be measured using piezoresistive principles. Then the magnetic field is easily found using the formula in equation (2.2). This field is pointing towards north (excluding disturbances from for instance the motors on the UAV), which will be utilized in the APMs IMU (described in the next section) to get more accurate attitude measurements.

The magnetometer in the APM is a HMC5883L from Honeywell.

Inertial Measurement Unit

The APM contains an Inertial Measurement Unit (IMU). An IMU consists of an ISA (Inertial Sensor Assembly), hardware and low level software. The ISA is a cluster of three gyroscopes and three accelerometers that measure angular velocity and acceleration respectively [Vik, 2012]. The IMU can also use magnetometer measurements. In the APM the magnetometer is not a part of the IMU, but it has an interface where it communicates with the magnetometer to make it possible to utilize the magnetometer measurements in the calculations of the attitude of the UAV.

Conceptually the accelerometer measures the movement of a damped mass hanging in a spring. To transform this movement into an electric signal, piezoresistive principles are utilized. In this case is it the acceleration that creates deformation in the piezoresistive material.

The gyroscopes are also based on MEMS technology. MEMS-gyroscopes are usually implemented with a tuning fork configuration. Two masses oscillate in opposite directions of each other. When these masses experiences angular velocity the Coriolis force act in opposite directions on the masses. This results in a measurable capacitance change which is proportional to the angular velocity of the UAV [Jay Esfandyari, 2010].

The IMU in the APM is a MPU-6000 from Inven Sense.

GPS

The GPS module that is connected to the APM contains an ublox LEA-6H module [3DRobotics, 2013]. It uses Navstar GPS but can also support GLONASS and Galileo. Communication to the APM is done via UART[2] with an update frequency of 5 Hz. Position accuracy is given by the datasheet to be 2.5 m CEP[3][u blox, 2012]. GPS can also be used for altitude measurements, but the nature of the GPS is that altitude measurements will have even less accuracy than position measurements.

2.1.3 Interfaces

The APM has several interfaces that make it flexible and suitable for research and development. It has dedicated connection points for GPS and telemetry. These are interfaced using UART. It also has an unused UART-port available for other units and applications. Input from the radio and output for the motor controllers have dedicated ports that operates with PWM-signals. A connection point to access the APMs I^2C bus is also available. Several units can communicate using this bus. It has also a lot of unused I/O-pins available for further development. These can for instance be used to connect an Ultrasonic Range Finder (there are several of them that are supported by the APM).

[2]Universal Asynchronous Receiver/Transmitter

[3]CEP (Circular Error Probability) defines the radius of a circle centered in the true position containing 50 % of the GPS measurements [iGage, 2013]

MAVLink

The APM communicates with its surroundings using UART with a subset of the communication protocol MAVLink[4]. MAVLink is a lightweight, header-only marshalling library for micro air vehicles [QGroundControl, 2013]. MAVLink has a lot of predefined messages in addition to the possibility of creating custom messages. An XML-file contains the definition of the different message types. An example of one of the message definitions is shown in Figure 2.3.

```
<message id="24" name="GPS_RAW_INT">
        <description>The global position, as returned by the Global Positioning System (GPS). This is NOT the global
        <field type="uint64_t" name="time_usec">Timestamp (microseconds since UNIX epoch or microseconds since system
        <field type="uint8_t" name="fix_type">0-1: no fix, 2: 2D fix, 3: 3D fix. Some applications will not use the v
        <field type="int32_t" name="lat">Latitude (WGS84), in degrees * 1E7</field>
        <field type="int32_t" name="lon">Longitude (WGS84), in degrees * 1E7</field>
        <field type="int32_t" name="alt">Altitude (WGS84), in meters * 1000 (positive for up)</field>
        <field type="uint16_t" name="eph">GPS HDOP horizontal dilution of position in cm (m*100). If unknown, set to:
        <field type="uint16_t" name="epv">GPS VDOP horizontal dilution of position in cm (m*100). If unknown, set to:
        <field type="uint16_t" name="vel">GPS ground speed (m/s * 100). If unknown, set to: UINT16_MAX</field>
        <field type="uint16_t" name="cog">Course over ground (NOT heading, but direction of movement) in degrees * 10
        <field type="uint8_t" name="satellites_visible">Number of satellites visible. If unknown, set to 255</field>
</message>
```

Figure 2.3: Message definition of message with ID 24 *Courtesy of wikipedia.org*

The MAVLink protocol can be used both to get status information from the APM and to give commands to the APM.

2.2 PandaBoard ES

The version of the PandaBoard used in this project is the PandaBoard ES Revision B2 (hereinafter referred to as PandaBoard). The PandaBoard is a small but powerful computer based on an OMAP™ 4 Processor. The OMAP™ 4 Processor is designed for high performance applications within a low power envelope [Texas Instruments, 2013] and contains a Dual-core ARM®1.2 GHz CPU. The PandaBoard do also have 1 GB RAM and a port to insert a SD-card for additional memory [pandaboard.org, 2013].

2.2.1 Interfaces

The PandaBoard has several interfaces that make it a good platform for development. It has two expansion connectors with 28 pins each. The functions of these pins includes general purpose I/O, SPI, I^2C, USB, UART, audio, power and support for additional memory [pandaboard.org, 2011]. This means that the the PandaBoard is able of communicating with its surroundings using most of the most popular bus standards.

The PandaBoard does also include a camera header available for development of camera solutions. e-con Systems has developed a camera (e-CAM51_44x) especially for the PandaBoard and this header. The camera is a 5 MP auto focus camera able to provide 720p HD

[4]Micro Air Vehicle Communication Protocol

video streaming at 60 fps [e-con Systems, 2013b]. Communication with the camera uses the MIPI[5] CSI-2 standard. CSI-2 is a standard that provides a robust, scalable, low power and high speed interface for imaging solutions [MIPI, 2013].

[5]Mobile Industry Processor Interface

Chapter 3

Possible Design Solutions and Evaluations

During this project several design solutions have been considered. Some of them have been discarded as impractical, too complex or too inaccurate, while others have potential to solve the problem in a efficient and elegant way. This chapter will cover all the different solutions that have been considered, and give a short summary of the designs and the advantages and limitations of the solutions before concluding on which design solution to implement.

3.1 Specifications and Goals for the Design

There are some specifications that the design solution need to meet to be able to execute the drop and recovery operation with the selected UAV. These specifications are given below together with some additional goals that would make the mechanism better suited for the mission.

- The UAV described in Chapter 2 should be used to carry the mechanism
- A PandaBoard should be used as onboard computer
- The full system should weight as little as possible with a maximum value of 800 g
- The mechanism should be able to pickup a sensor node weighting up to 200 g
- Communication with the sensor node is not an option
- It should be made in a way that makes pickup and deployment as simple and robust as possible
- The possibility of dropping the sensor node in an emergency situation is appreciated

3.2 Design Solutions

The specifications above led to some common elements for all the different design solutions. These are presented here before the different design solutions are presented and discussed.

Without communication between the sensor node and the UAV, finding and navigating towards the sensor node will be one of the biggest challenges. There are several possibilities in this matter, one could for instance use radar, lidar or computer vision. Due to simplicity and low weight computer vision is chosen to play a part regardless of which mechanical design that is chosen for the pickup and deployment mechanism.

Another element that is independent of the pickup and deployment mechanism is the sensor node. To be able to handle different sensor nodes a standard mount needs to be developed. This mount needs to be easy to recognize for the camera application and easy to grip or hook onto for the pickup mechanism. The mount designed meets both of these demands. It consists of two ping pong balls with a rod between them as shown in Figure 3.1. The balls are used for the camera to determine the position and orientation of the mount, and the rod is used for the pickup mechanism. The mount is either to be mounted on top of the sensor node such that it is elevated up into the air, or it can be mounted such that it floats next to the sensor node. This will be up to the designer of the sensor node to decide. The choice of colors for the mount will be explained more in detail in Section 5.2.1.

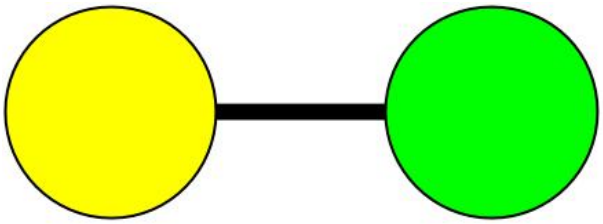


Figure 3.1: Mount for the sensor node

3.2.1 Separate Hook and Release mechanism

Design

One of the simplest pickup mechanisms possible is a hook on a string. Either a small DC-engine in combination with a H-bridge or a continuous rotating servo can be used to hoist and lower the hook. Moreover, one of the simplest drop mechanisms is simply a servo controlled pin that release the sensor node.

Advantages and Limitations

The greatest advantage of this design is its simplicity in the mechanical design. This simple design will also be low weight, which is good for the UAVs range and it would be possible to

pick up heavier sensor nodes.

However, the simplicity also comes at a cost. While the hook is lowered down will its position be highly influenced by wind and the UAVs movements, which makes it almost impossible to determine the position of the hook. Use of encoder to know how much string that is reeled out and use of the camera to determine the position of the sensor node in the picture plane will reduce this problem, but not remove it because of the flexibility of the string. And even if the position of the hook is well known, it could be difficult to get it to the position of the sensor node and get it to hook onto it.

The separate drop and pickup mechanisms mean that a person need to be involved in the process of attaching and detaching the sensor node from the UAV. Ideally the UAV should be able do this itself in order to operate as independently as possible. This solution also makes it impossible to drop the sensor node after pickup if an emergency situation should arise. This could be solved by adding a safety pin in combination with a servo that release the spool with the hook and the sensor node if needed.

The sensor node will be dangling on the hook on the way back after the pickup, this means that there will be a risk of dropping the sensor node unintentionally.

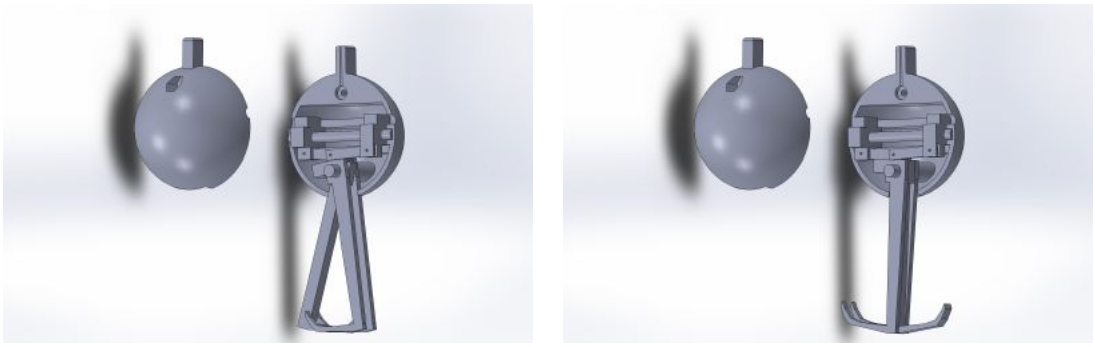
3.2.2 Combined Hook and Release Mechanism

Design

In order to reduce the need of human involvement and remove the need for both a regular drop and an emergency drop mechanism, the hook itself could be designed in such a way that it also could be used for the drop off. An example of such a design is shown in Figure 3.2. A spring will be mounted in the middle of the hook to keep the hook in normal operating position. When the mechanical hook is reeled in, it will go into a docking station on the UAV that will guide it into a position where two servos can be used to push two pins into holes on the side of the hook, which will make the spring contract and the sensor node to be released (see Figure 3.2(a)).

Advantages and Limitations

As mentioned above, this solution will reduce the need for human involvement, but increase the complexity of the hook. The problems of knowing where the hook is continues, and the risk of dropping the sensor node unintentionally are increased due to the flexibility of the hook.



(a) Release Position (b) Normal operating position

Figure 3.2: Mechanical hook

3.2.3 String with Electromagnet

Design

Instead of having a hook on the end of the string, one could have an electromagnet there. If the sensor node contains either a permanent magnet or some metal, the electromagnet will stick to it with contact.

The sensor node will need to have a mount that can be used to fasten the sensor node to the UAV when reeled in using a servo controlled pin. This is needed in order to save power and to make sure that the sensor node will not fall off unintentionally.

Advantages and Limitations

An electromagnet will make it easier to get hold of the sensor node.

If the sensor node contains communication devices or sensitive sensors, a nearby magnetic field will spell trouble. If the sensor node for instance contains a magnetometer for navigation, those sensor data will for sure be corrupted. The sensor data and communication from the pickup phase might not be important so it might be acceptable to corrupt data and disturb the communication, but magnetic fields can in some cases permanently damage electronics.

When the string conductor is reeled in on the spool, an inductor will be created, consequences of this will have to be considered.

3.2.4 Gripper Fixed to Rigid Rod

Design

A gripper could be attached at the end of a rigid rod that is attached under the UAV. This rod could be servo controlled to increase flexibility and depending of the length, get it out of the way for landing.

The gripper could consist of two parts that are connected using gears (see Figure 3.3). This connection makes it possible to control the gripper using only one servo that will be attached to one of the parts.

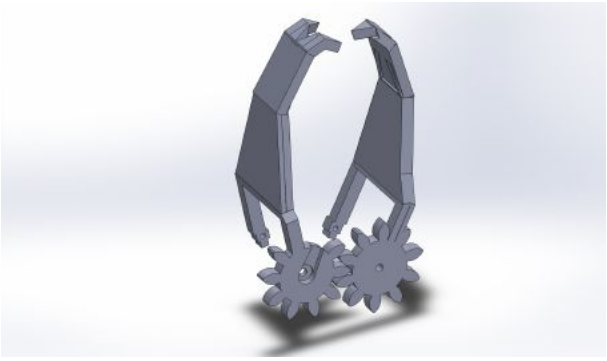


Figure 3.3: Gripper

Advantages and Limitations

The length of the rod will determine how much the weight of the gripper and the sensor node will influence the center of gravity of the UAV. Ideally the center of gravity is in the middle of the APM, but some deviations are OK. If the deviations are along the body z-axes the controller will be able to handle it after some tuning of the controller parameters as the controller will need to use more force to control roll and pitch angles. This mass will stabilize roll and pitch angles, but if center of gravity is moved away from the body z-axis the roll and pitch angles will become unstable, and possibly too unstable for the controller to manage. If the rod is long and its direction is adjustable by a servo, use of this servo will create inertia that can be troublesome for the controller.

If the rod is chosen to be short, these effects are reduced. The distance to the water will be short, which means risk of damage by splashing water created by the propellers or the peripherals of the UAV getting drowned while the controller tries to get to the desired position and attitude. Flying close to the ground/water rotor downwash will create a cushion of air referred to as “ground effect”. As the UAV moves laterally through ground effect, the cushion can be entrained and recirculated by the rotor. This causes thrust to decrease rapidly, causing the UAV to bounce and plunge [Pounds and Dollar, 2014]. This will make it difficult to get into position to grip the sensor node and it could result in a drowned UAV.

3.2.5 Gripper Fixed to Adjustable Rod

Design

The gripper in Figure 3.3 could be mounted on a flexible rod instead of a rigid rod. To lower and raise the gripper one could mount the gripper to telescopic rods, or rods that is controlled using gears. The telescopic rods could be controlled using wire and either a continuous servo or a DC-engine in combination with a H-bridge.

Advantages and Limitations

If the flexible rod is at its most expanded in the pickup phase, the controller will know exactly where the gripper is in relation to the APM. The flexibility of the rod means that the pickup can happen at a safe distance from the water surface while the center of gravity can be kept closer to the APMs center during flight.

The added complexity due to the flexibility of the rod will add some weight and increase the risk of failure.

3.3 Discussion of the Different Designs and Conclusion of Which Solution to Implement

The main difference between the solutions described above is the part that "catches" the sensor node, the hook, electromagnet or the gripper. The electromagnet has some advantages when it comes to "catching" the sensor node, but the risk of corrupting sensor data and communication, or in the worst case permanently damage the electronics makes it a poor choice. The solution with a simple hook seems like the simplest one, but there are some issues that will make it more complex. Like for instance the safety drop mechanism, or the problems of finding the position of the hook, or how to control the hook towards the sensor node.

All of the solutions that include a hook have the problem of securing the sensor node from falling off the hook during flight. This can be solved, but it will once more increase complexity.

A solution with a gripper solves much of the issues related to the hook. Using a gripper, gives no need for a safety mechanism, position is well known and there is no risk of the sensor node falling off during flight. The path planning for pick up will be much simpler because the relationship between the movements of the UAV and the movements of the gripper is strictly defined.

Then there is the question of where to mount the gripper. One of the solutions above suggests to mount it on the end of a servo controlled rod. The servo could be used to cooperate with the controller to avoid moments of inertia created from the sensor node to crash the

UAV. This could be especially useful in the case of high speed pickup, where movements of the rod can counteract a sudden jerk at the pickup. But high speed pickup is not a priority in this project. The rod will need to be short in order to avoid shifting the center of gravity too much, which will force the UAV to fly close to the water surface.

The solution with the flexible rod seems to be the solution that will make path planning and operation simplest. Although the mechanical design will be more complex and heavier than the other solutions, it seems like the best choice. Many components are common between the flexible and the rigid rod solution, which means that implementing the flexible one and evaluating it is a good way to go. If its complexity comes at the cost of robustness, it is easy to simplify the design later on and go for the rigid rod solution. For this reasons, the flexible rod solution will be implemented and tested.

Chapter 4

Background Theory

4.1 Reference Frames

The main reference frames that are relevant for navigation and control are ECEF, NED and BODY. A brief introduction to these reference frames based on [Fossen, 2011] and [Vik, 2012] follows. Denavit-Hartenberg convention (DH convention) is a useful tool for defining reference frames. A brief introduction to DH convention based on [Spong et al., 2005] follows.

ECEF

The Earth-centered Earth-fixed (ECEF) reference frame has its origin fixed to the center of the earth while rotating with the earth. The x-axis is defined to point at the intersection between the 0° longitude and the 0° latitude. The z-axis points along the earths rotation axis and the y-axis complete the right handed orthogonal coordinate system.

The position in the ECEF frame can be expressed both with Cartesian coordinates (x_e, y_e, z_e) and with ellipsoidal coordinates (longitude (l), latitude (μ), height (h)). The transformation from Cartesian ECEF-coordinates to ellipsoidal ECEF-coordinates is given by

$$\begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix} = \begin{bmatrix} (N+h)\cos\mu\cos l \\ (N+h)\cos\mu\sin l \\ (\frac{r_p^2}{r_e^2}N+h)\sin\mu \end{bmatrix} \tag{4.1}$$

where $r_e = 6378137$ m is the equatorial radius of ellipsoid and $r_p = 6356752$ m is the polar axis radius of the ellipsoid as defined in WGS-84. The parameter N is the radius of curvature in prime vectorial obtained from [Vik, 2012].

$$N = \frac{r_e^2}{\sqrt{r_e^2\cos^2\mu + r_p^2\sin^2\mu}} \tag{4.2}$$

The transformation from Cartesian coordinates to ellipsoid coordinates is a bit more complicated. Longitude is calculated straight forward as

$$l = \tan^{-1}(\frac{y_e}{x_e}) \tag{4.3}$$

but the calculations of latitude and height are implicit equations

$$\tan(\mu) = \frac{z}{p}(1 - e^2\frac{N}{N+h})^{-1} \tag{4.4}$$

$$h = \frac{p}{\cos(\mu)} - N \tag{4.5}$$

where e is the eccentricity of the Earth given by

$$e = \sqrt{1 - (\frac{r_p}{r_e})^2} \tag{4.6}$$

There are several algorithms that can be used to solve these implicit equations, see for instance Algorithm 2.4 in [Fossen, 2011].

NED

The north-east-down (NED) reference frame is moving with the body. The x-axis is always pointing north, the y-axis is pointing east and the z-axis i pointing down normal to the Earth surface.

BODY

The BODY reference frame is body fixed and rotates and moves with the body. It is usually defined with the x-axis pointing along the longitudinal axis, the y-axis pointing along the transversal axis and the z-axis pointing along the normal axis of the body.

Transformation Between ECEF and NED

A vector defined in NED can be transformed to the ECEF frame by the use of a rotation matrix that defines the relationship between NED and ECEF reference frames.

$$\boldsymbol{p}^e = \boldsymbol{R}_n^e(\boldsymbol{\Theta}_{en})\boldsymbol{p}^n \tag{4.7}$$

$$\boldsymbol{\Theta}_{en} = \begin{bmatrix} l & \mu & h \end{bmatrix}^T \tag{4.8}$$

Where $\boldsymbol{R}_n^e(\boldsymbol{\Theta}_{en})$ is found by first performing a rotation l about the z-axis and then a rotation $(-\mu - \frac{\pi}{2})$ about the y-axis. This gives

$$\boldsymbol{R}_n^e(\boldsymbol{\Theta}_{en}) = \begin{bmatrix} -\cos(l)\sin(\mu) & -\sin(-l) & -\cos(l)\cos(\mu) \\ -\sin(l)\sin(\mu) & \cos(l) & -\sin(l)\cos(\mu) \\ \cos(\mu) & 0 & -\sin(\mu) \end{bmatrix} \tag{4.9}$$

Transformation Between NED and BODY

A vector defined in BODY can be transformed to the NED frame by the use of the Euler angle rotation matrix.

$$\boldsymbol{p}^n = \boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb})\boldsymbol{p}^b \tag{4.10}$$

Where $\boldsymbol{\Theta}_{nb} = \begin{bmatrix}\phi & \theta & \psi\end{bmatrix}^T$ are the Euler angles roll, pitch and yaw.

$$\boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb}) = \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\phi + c_\psi s_\theta s_\phi & s_\psi s_\phi + c_\psi c_\phi s_\theta \\ s_\psi c_\theta & c_\psi c_\phi + s_\phi s_\theta s_\psi & -c_\psi s_\phi + s_\theta s_\psi c_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \tag{4.11}$$

The notation s and c with an angle as subscript is used to represent $\sin(angle)$ and $cos(angle)$ respectively. This notation is used throughout this report.

Denavit-Hartenberg Convention

The DH convention is a systematic procedure for relating orientation and position of different reference frames, and a tool to select frames. The homogeneous transformation A_i (the transformation matrix from reference system $i-1$ to reference system i) is represented as a product of four basic transformations

$$\begin{aligned} \boldsymbol{A}_i &= \boldsymbol{Rot}_{z,\theta_i}\boldsymbol{Trans}_{z,d_i}\boldsymbol{Trans}_{x,a_i}\boldsymbol{Rot}_{x,\alpha-i} \\ &= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{4.12}$$

where θ_i is rotation around the z-axis, d_i is transversal movement along the z-axis, a_i is the transversal movement along the new x-axis direction and α_i is the rotation around the new x-axis direction. There exists unique values of θ_i, d_i, a_i and α_i to make equation (4.12) valid if the frames have the features:

- The axis x_i is perpendicular to the axes z_{i-1}
- The axis x_i intersects the axis z_{i-1}

The transformation matrix T_j^i expresses the position and orientation of frame $o_j x_j y_j z_j$ with respect to frame $o_i x_i y_i z_i$ and is calculated as

$$\boldsymbol{T}_j^i = \begin{cases} \boldsymbol{A}_{i+1}\boldsymbol{A}_{i+2}...\boldsymbol{A}_{j-1}\boldsymbol{A}_j & \text{if } i < j \\ I & \text{if } i = j \\ (\boldsymbol{T}_i^j)^{-1} & \text{if } j > i \end{cases} \tag{4.13}$$

$\boldsymbol{T}_j^i$ for $i < j$ contain a rotation matrix $\boldsymbol{R}_i^j$ from frame i to j and the position $\boldsymbol{o}_j^i$ of the origin of frame j with respect to frame i.

$$\boldsymbol{T}_j^i = \begin{bmatrix} \boldsymbol{R}_j^i & \boldsymbol{o}_j^i \\ 0 & 1 \end{bmatrix} \tag{4.14}$$

4.2 Computer Vision

4.2.1 OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library that has more than 2500 optimized algorithms [OpenCV.org, 2013a]. A few of these algorithms will be briefly explained here. The ones explained are the most relevant for object recognition.

Color Recognition

A digital picture is essentially a matrix of values describing the picture. This matrix is dependent of the color space the picture is defined in. A picture captured from for instance a web-camera is defined in the BGR (Blue-Green-Red) color space, which means that the colors in the picture are defined by combinations of these colors. The color is defined by the relationship between these values, while the brightness is defined by how high these values are. Hence it could be difficult to select threshold values if one is looking for an object of a specific color, because different light conditions would give very different values for the color. To make the thresholding more intuitive one can transform the picture into the HSV (Hue-Saturation-Value) color space. The hue value is unique for a specific color and describes the base color. The saturation describes the strength of the color and the value is a measure of brightness. This makes it simpler to find intuitive thresholds for specific object colors that can handle different lightning conditions. An visualization of the HSV-color space is found in Figure 4.1.

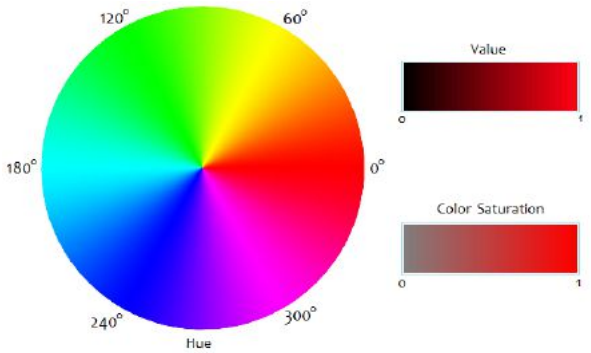


Figure 4.1: HSV-color wheel *Courtesy of had2know.com*

The HSV-picture is thresholded with the desired intervals of the HSV-values. This results in a binary image where the pixel values are one if the color is found and zero if the color is different than the desired color. With good choices of colors for the object and the thresholding values, one would get a good understanding of where the object is.

Canny Edge Detector and Moments

The Canny Edge Detector uses an algorithm presented in [Canny, 1986] that detects edges. It can be used to detect contours. If the Canny Edge Detector is used on a binary image like the one described in the previous section it would find the contour surrounding the area of the detected object.

Then this contour could be fed to the moment function in OpenCV which calculates the center of moment for the contour, which is the center of the outline of the object.

Cascade Classifier

The use of cascade classifiers includes two major stages, training and detection [OpenCV.org, 2013b]. Training is executed once only, while detection is executed run time. Training takes two different sets of samples, positive and negative samples. The positive samples are samples containing the object, while the negative samples are samples without the object. These samples are run through a cascade classifier to create a "rule" of what to look for in the detection phase.

There exists several different cascade classifiers, two of these are implemented in OpenCV. These are Haar- and LBP-classifiers. They use different features and have different runtime. Details on these algorithm are outside the scope of this text.

Accuracy of the use of cascade classifiers is dependent on the object to be detected, and the number of samples used for training. For instance one positive sample can be sufficient for detection of a rigid object, while detection of for instance faces will need hundreds or even thousands of positive samples.

There are tools in OpenCV that take the positive samples and creates many new positive samples of it. This is done by randomly rotating the object around all three axes, changing the objects intensity and placing it on random backgrounds [OpenCV.org, 2013b].

SURF

Speeded Up Robust Features (SURF) is a further development and speeded up algorithm on the basis of SIFT (Scale-Invariant Feature Transform) [Mordvintsev and Abid, 2013]. These algorithms use a single picture of the object for the recognition. Based on different techniques beyond the scope of this text, key points are found describing the object. The key points is assigned an orientation to achieve invariance to image rotation, and the way the key point is

selected make the algorithm scale invariant [OpenCV, 2013].

For commercial use one should note that both SURF and SIFT is patented and part of a non-free module in OpenCV.

Chapter 5

Design Implementation

As a mount and a basis for the selected design solution and possibly other design solutions, a box designed to contain the PandaBoard and eventually some more hardware was 3D-printed. This box is shown in Figure 5.1. The holes in the box are for connections to the PandaBoard, while the pieces sticking up on the top are used for mounting it to the UAV (ArduCopter Hexacopter). The Hexacopter has some slits in its bottom plate that are used to fasten the battery. The battery will fit between these pieces and they will go through the slits. After the pieces are pushed through they will be secured by splints. The cubes with holes in the bottom corners of the box are used for keeping the pieces of the box together and as a mount to fasten the design solution to the UAV.

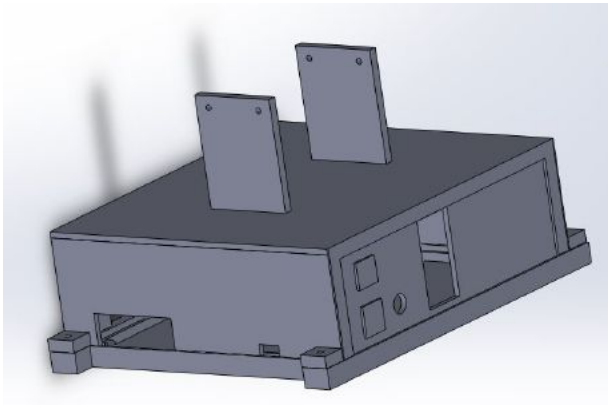


Figure 5.1: Box to contain the PandaBoard

5.1 Mechanical Design

Flexible Rod

There are several ways to design a system where a platform can be lowered down. One could for instance use telescopic poles, gears or some sort of accordian arm. The simplest solution seemed to be telescopic poles. Three lightweight aluminium telescopic poles were made. The

idea was to fasten these with one end to the UAV and one end to a gripper platform. Wires should run through the poles and be used to raise the platform. To lower the platform, these wires should be slackened and gravity would do the rest. Some early stage testing showed that this design was everything but robust. If lowering gets uneven, e.g. due to the UAVs attitude or friction, everything could get stuck. Hence a more robust though complicated solution was developed using gears to push the gripper platform both up and down.

To be able to use gears to push the platform up and down, rack gears were used. The rack gears were fastened to the gripper platform at one end, while the other end went up through the gear platform. To be sure that all the rack gears move at the same time and with the same speed, the same motor was used to drive them all. This meant that a transmission needed to be designed. The platform for the gears was 3D-modelled and 3D-printed, and the gears was mounted. This can be seen in Figure 5.2.

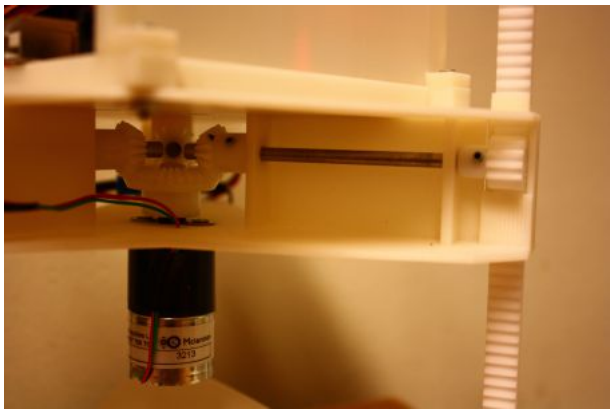


Figure 5.2: Gear

A DC motor was chosen to drive the gears. Alternatively a servo could have been used, but that would have been more spacious in a position where space really matters. It would require a lot of torque to drive the transmission and lift the gripper platform. Hence torque was the most important parameter in the process of choosing a motor. The motor chosen was a 1271 series 188:1 geared DC motor from Mclennan able of providing a torque of 14 Ncm at the same time as its small in size and relatively lightweight. The fact that it is geared increases the torque at the same time as it decreases the speed of the motor. The rated speed is 9 rpm, which means that it will use approximately 20 seconds to raise or lower the gripper platform. This is reckoned to be within acceptable time consumption. To be able to control the DC motor a H-bridge is needed. A dual H-bridge motor driver shield based on the driver chip L298N from ST-microelectronics is used. This uses two signals to set the direction of the motor, and one signal to enable the motor. Speed can be adjusted by using a PWM-signal as the enable signal. But for this project it will be operated at full speed only, hence a digital output is used for the enable signal.

To know when the gripper platform is in the extreme positions a measurement of the plat-

forms position is needed. How this was solved is explained in Section 5.5.

Gripper

When designing the gripper mechanism, several moments needed to be considered. It should be as simple, robust and power efficient as possible. At the same time making sure that it doesn't build to much below the UAV and making sure that the camera gets a good view.

The gripper was designed in a way that it could be controlled using only one servo. This was possible by using gears. The gripper was 3D-modelled and 3D-printed. See Figure 5.3 to understand how the gripper parts interact. The servo used was a MG-14 from Hextronik. This servo was chosen due to the fact that it is relatively light and sufficiently strong, while being metal geared which is good for robustness.

The gripper needs to be mounted on a platform that is connected to the rack gears to be able to raise and lower the gripper. The platform is supposed to be raised up as close to the UAV as possible. The motor connected to the gear platform sticks down below the gear platform and is in the way for the gripper platform. This was solved by designing a hollow cylinder on the gripper platform for the motor to fit into when the gripper platform is raised. The cylinder is there instead of a simple hole to have a position to mount the camera that is centred and have free sight down while searching for the node. This placement of the camera also makes it possible to use the camera to verify if the gripper has caught the node or not. The gripper platform also needed a mount for the rack gear, the gripper and the servo for the gripper. Also the gripper platform was 3D-modelled and 3D-printed.

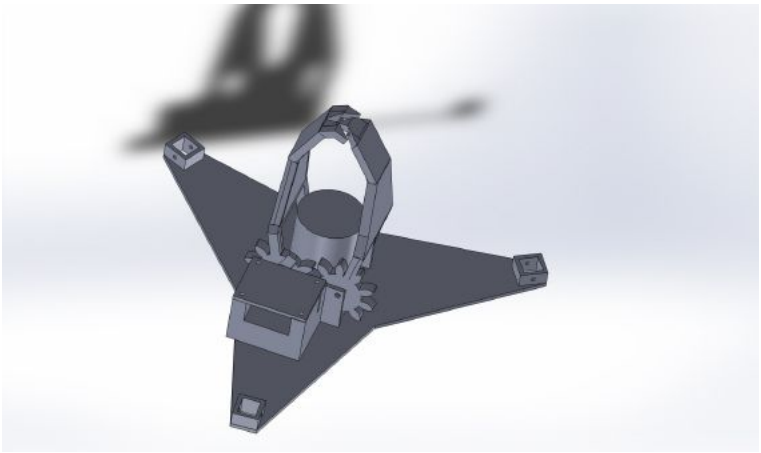


Figure 5.3: Gripper mounted on the gripper platform

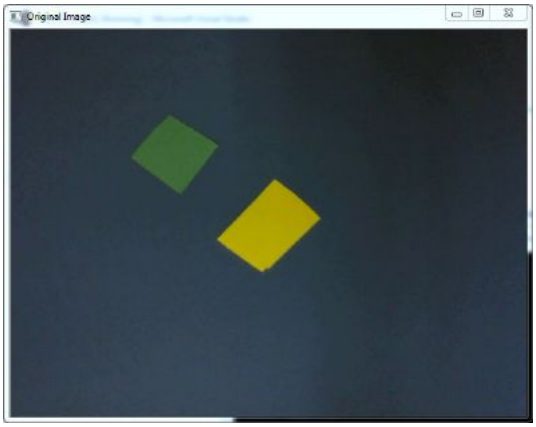
5.2 Camera Application

The main application of the camera is to track the sensor node. The camera will also be used in the pickup to verify whether the gripper got hold of the sensor node or not (one could have used estimation of inertia for this task like [Mellinger et al., 2011], but the position of the camera makes it very convenient to use for this task). To be able to track the sensor node the camera will need to recognize the sensor node. This can be done in multiple ways as described briefly in Section 4.2.1. The SURF method can be efficient under the right circumstances, but some simple tests revealed some weaknesses. It turned out that only a few points on a picture of the pickup mount was marked as corners that it could use to search for. Of course the pickup mount could be made in a way that it would have more detail, but because it is quite small and it should be possible to detect at a distance this method was discarded. A classifier could be developed, but it will have weaknesses when it comes to rotation, and it would need a lot of example pictures to make it robust. Of this reasons the classifier solution was discarded as well. The much simpler solution of color detection was implemented.

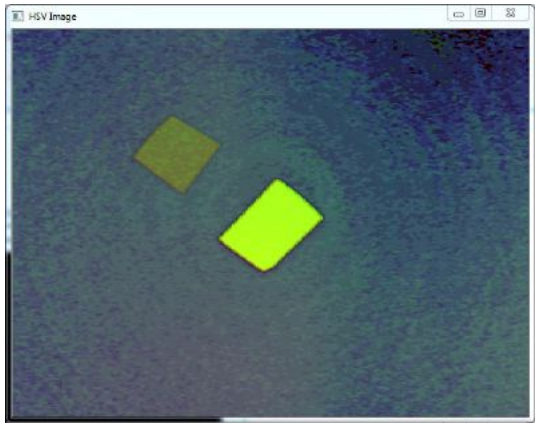
5.2.1 Tracking the Sensor Node

To be able to use color detection the sensor node need to have some colors that sticks out. The mount for the sensor node (Figure 3.1) should have two different colors to make it possible to use color detection to get the orientation of the mount. The ocean is blue, which means that the sensor node should have colors that are as far away from blue on the HSV-wheel (Figure 4.1). Yellow is an obvious choice because it is directly opposite of blue and it has a narrow band, which is good for noise cancellation. The colors next to yellow is red and green. Red contains both the lowest an highest hue values, which will make calculations more complex, hence green is chosen for the other color.

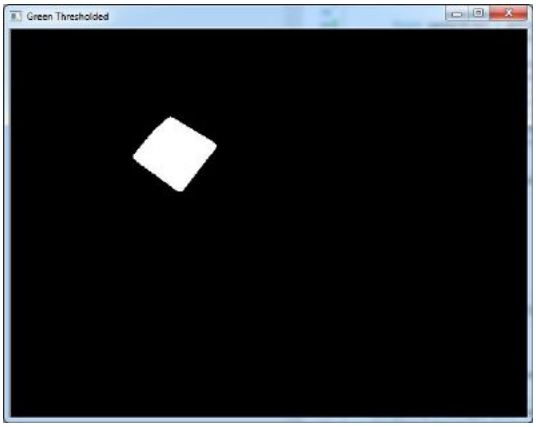
The picture from the camera is converted to the HSV-color space and copied to create two pictures. One of the pictures is thresholded with the HSV-values for yellow, while the other is thresholded with the values for green. This creates two binary pictures where the yellow and green fields are marked respectively. Then the "center of mass" of the two pictures are calculated. This centres of masses are used as centres of the two parts of the pickup mount and used to calculate the orientation of the mount and the center of the mount. This procedure is demonstrated in Figure 5.4. These points are in the picture frame. For the measurements to make sense one need a mapping from this pixel position to the position of the mount in NED.



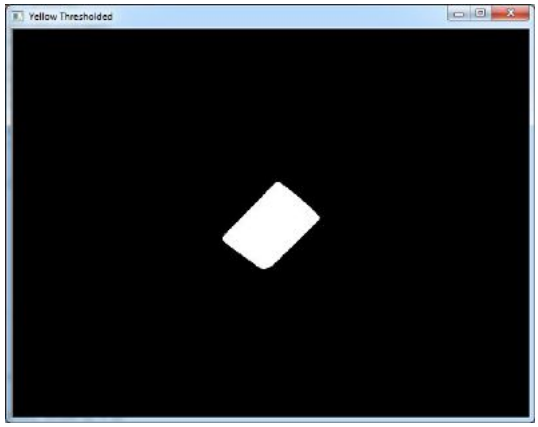
(a) Original image



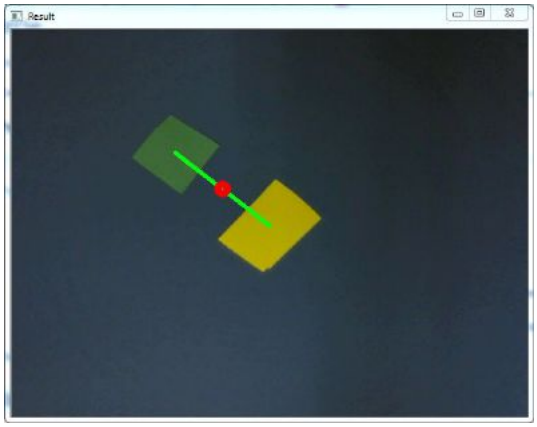
(b) HSV image



(c) Green thresholded



(d) Yellow thresholded



(e) Result

Figure 5.4: Use of color recognition to track the mount of the sensor node

For control purposes a position of the sensor node in NED is good because this will be the same value as the error in position if the controller is trying to get to the exact position of

the sensor node. For estimator and robustness the sensor nodes position should be referenced in ECEF. If the position of the sensor node is referenced in ECEF, it will be simpler to filter out weird measurements, estimation of the movements of the sensor node with current and waves becomes possible and the controller will have something to navigate towards even if the camera looses sight of the sensor node for a moment.

To get the position of the sensor node in NED, first a transformation from BODY to NED is conducted by the use of the rotation matrix from BODY to NED in equation (4.11). DH convention is used to create consequential reference frames from BODY leading up to a reference frame in the sensor node. This is done to exploit the fact that the transformation matrix from BODY to the sensor node reference frame will contain the position of the sensor node reference frame expressed in BODY as seen in equation (4.14).

The first new frame is defined in the center of the camera lens. The homogeneous transformation A_1 from BODY to the camera frame is carried out as a movement d_1 along the BODY z-axis which gives the relation below. This coordinate frame and subsequent frames are visualized in Figure 5.5. The parameters related to the transformations are also marked in the same figure.

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1}$$

The next coordinate system is defined with the z-axis pointing towards the sensor node and the pixel position of the sensor node in the picture frame. This is done by rotating around the z-axis of α degrees and then rotating around the x-axis of β degrees.

To find α and β some calculations needs to be done. The origin of the picture plane is in the topmost left corner. To get the angles to rotate, the origin is moved to the center of the picture frame by defining $\delta x = x_o - x_c$ and $\delta y = y_o - y_c$. The center of the picture frame is the point (x_c, y_c) while the pixel position of the sensor node is the point (x_o, y_o). The picture frame is defined to be at a distance of one meter away from the camera lens and the length between each pixel at this distance (L) is measured using an object of known length. The angles α and β are then calculated.

$$\alpha = atan2(\delta y, \delta x) \tag{5.2}$$
$$d = L\sqrt{\delta x^2 + \delta y^2} \tag{5.3}$$
$$\beta = \tan^{-1}(d) \tag{5.4}$$
$$\tag{5.5}$$

The resulting homogeneous transformation matrix is

$$\boldsymbol{A}_2 = \begin{bmatrix} c_\alpha & -s_\alpha & 0 & 0 \\ s_\alpha & c_\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\beta & -s_\beta & 0 \\ 0 & s_\beta & c_\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_\alpha & -s_\alpha c_\beta & s_\alpha s_\beta & 0 \\ s_\alpha & c_\alpha c_\beta & -c_\alpha s_\beta & 0 \\ 0 & s_\beta & c_\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.6}$$

The last reference frame is defined in the sensor node. To get there a movement of d_2 along the z-axis is necessary. To calculate this distance, the angle (γ) between the z-axis in NED and the z-axis in the picture frame reference system needs to be calculated. The transformation matrix from NED to picture frame is calculated to be able to read out the direction of the z-axis in the picture frame reference system.

$$\boldsymbol{T}_{pf}^{n} = \boldsymbol{R}_{b}^{n}(\boldsymbol{\Theta}_{nb})\boldsymbol{A}_1\boldsymbol{A}_2 \tag{5.7}$$

$$\tag{5.8}$$

Out of this transformation matrix one can read out the direction of the z-axis of the picture frame reference system $\boldsymbol{z}_{pf}^{n}$ while the direction of the z-axis of NED is trivial.

$$\boldsymbol{z}_{pf}^{n} = \begin{bmatrix} s_\alpha s_\beta c_\psi c_\theta - c_\alpha s_\beta(-s_\psi c_\phi + c_\psi s_\theta s_\phi) + c_\beta(c_\psi c_\phi + s_\psi s_\theta s_\phi) \\ s_\alpha s_\beta s_\psi c_\theta - c_\alpha s_\beta(c_\psi c_\phi + s_\psi s_\theta s_\phi) + c_\beta(-c_\psi s_\phi + s_\psi s_\theta s_\phi) \\ -s_\alpha s_\beta s_\theta - c_\alpha s_\beta c_\theta s_\phi + c_\beta c_\theta c_\phi \end{bmatrix} \boldsymbol{z}_{n}^{n} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{5.9}$$

The angle γ between these two vectors can be calculated using dot product, and the fact that the vectors in the rotation matrices are normalized. This combined with basic trigonometry gives these two relations.

$$\cos\gamma = \frac{\boldsymbol{z}_{pf}^{n} \cdot \boldsymbol{z}_{n}^{n}}{|\boldsymbol{z}_{pf}^{n}||\boldsymbol{z}_{n}^{n}|} = z_{pf_3}^{n} \tag{5.10}$$

$$\cos\gamma = \frac{h - d_1 c_\theta c_\phi}{d_2} \tag{5.11}$$

$$\tag{5.12}$$

These relations combined gives the following expression for d_2 and the homogeneous transform.

$$d_2 = \frac{h - d_1 c_\theta c_\phi}{z_{pf_3}^{n}} \tag{5.13}$$

$$\boldsymbol{A}_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.14}$$

$$\boldsymbol{T}_{obj}^{n} = \boldsymbol{T}_{pf}^{n}\boldsymbol{A}_3 \tag{5.15}$$

Calculations of equation (5.15) will read out the position of the origin of the reference frame in the sensor node (according to equation (4.14)). This gives

$$\boldsymbol{p}_{obj}^{n} = \begin{bmatrix} d_2 z_{pf_1}^{n} + d_1(s_\psi s_\phi + c_\psi s_\theta c_\phi) \\ d_2 z_{pf_2}^{n} + d_1(-c_\psi s_\phi + s_\psi s_\theta c_\phi) \\ h \end{bmatrix} \tag{5.16}$$

For the purpose of tracking this vector is then transformed into the ECEF coordinate frame by the transform in equation (4.8).

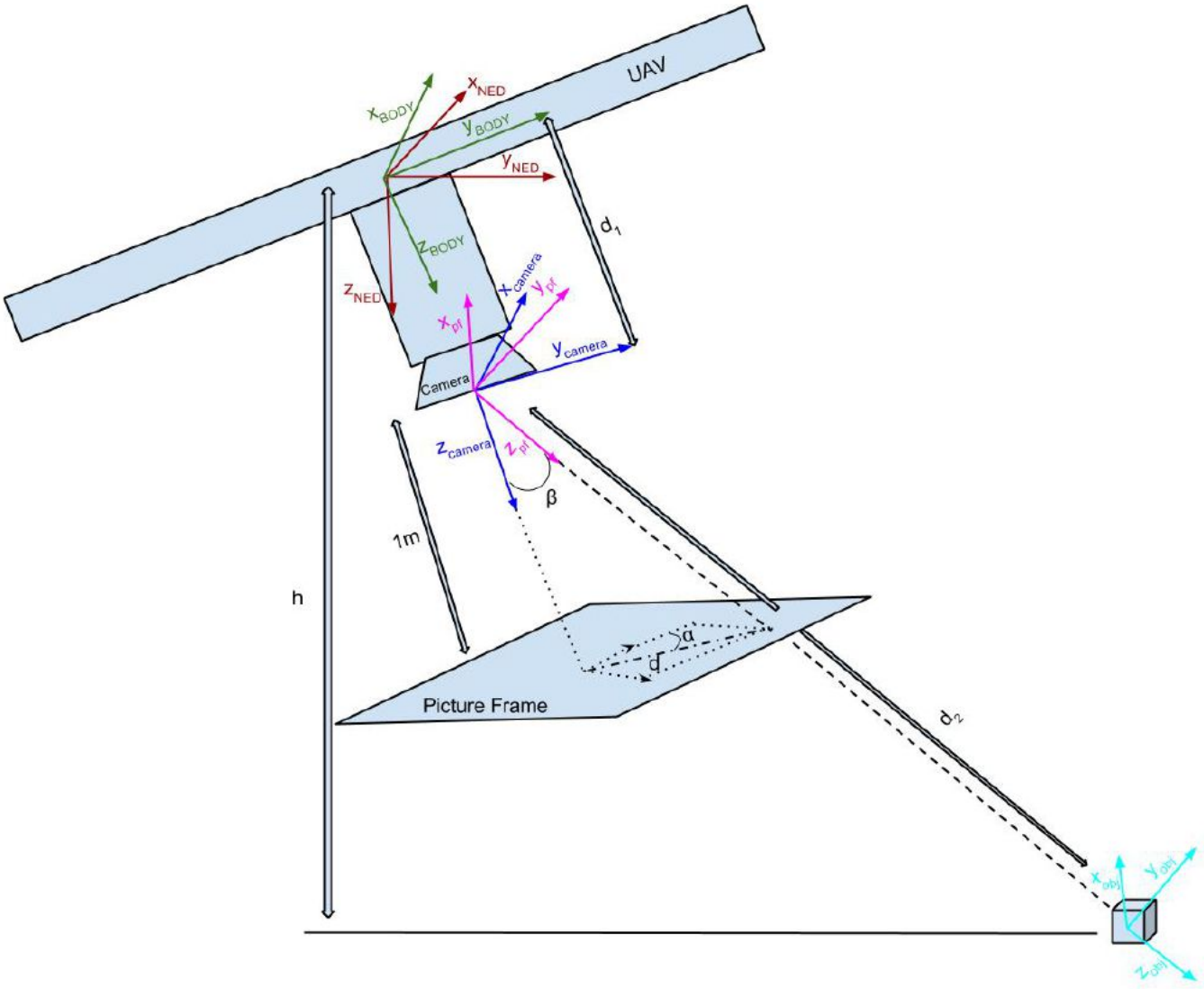


Figure 5.5: Visualization of the different coordinate systems

5.3 Altitude Measurements

The sensor node tracking algorithm described in the previous section is quite vulnerable to inaccurate altitude measurements. As established in Chapter 2, the altitude measurements using barometer and GPS can be very inaccurate. Hence another altitude measurement should be included.

A way to avoid including additional hardware could be to use the camera for height measurements. The mount for the sensor node is of known size. This could be used for a distance measurement by using the correlation between number of yellow or green pixels and distance. This will not increase accuracy of altitude measurements when the sensor node is not visible, but this is not a problem because it is when the sensor node is visible that good altitude measurements are most important. A weakness of this solution is that the entire mount for the sensor node might not be visible. This will give a greater distance reading than the real distance. This solution is not implemented due to this problems.

There are several other possible solutions. The challenge is that it should be relatively lightweight and not to complex. LIDAR are too heavy and complex. There exists a relatively lightweight radar alternative in the Novelda Nanoscale Impulse Radar. Unfortunately is this radar very complex in the way that raw data needs to be processed in order to get distance measurements. Two possible lightweight and simple solutions are ultrasound and laser. There exist small modules of both these solutions especially made for the micro UAV market.

The most relevant laser range finder found is the SF02 from Lightware Optoelectronics. The SF02 weights 75 g and can detect surfaces at a distance of up to 40 m with a resolution of 1 cm. The wavelength of the laser light is 850 nm [Lightware, 2013]. The accuracy profile of the SF02 can be seen in Figure 5.6. This profile shows that there might be a problem with bad accuracy in the pickup phase when the UAV is close to the surface of the sea.

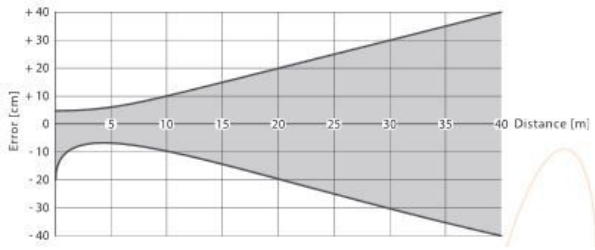


Figure 5.6: Accuracy profile of SF02 *Courtesy of Lightware Optoelectronics*

Another moment that needs to be considered is whether the laser beam will be reflected from the surface of the sea and back to the sensor or not. In [Nejad and Mirsaeidi, 2005] a survey where the reflectance of different wavelengths on different materials is presented. The results of this survey are presented in Figure 5.7.

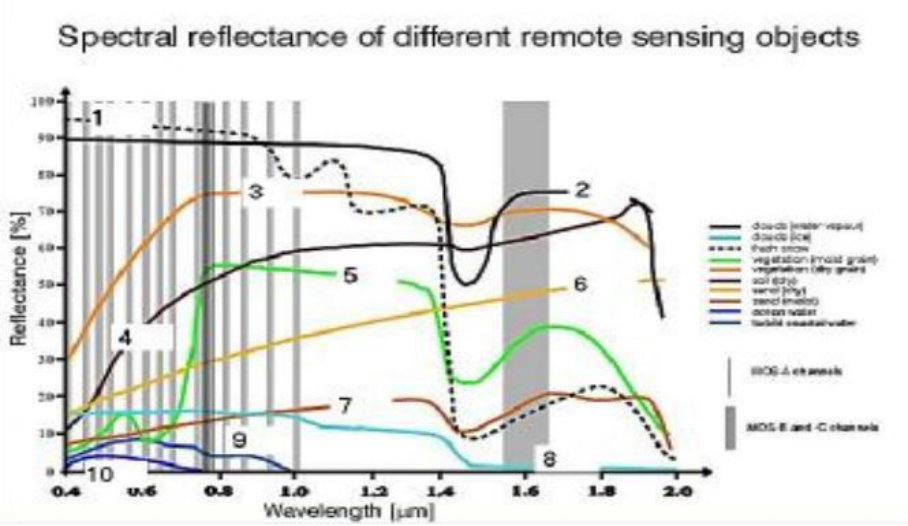


Figure 5.7: Measured reflectance for different objects. 1) snow, 2) cloud, 3) dry vegetation, 4) soil, 5) green vegetation, 6) dry sand, 7) moist sand, 8) ice, 9) turbid coastal water, 10) ocean water [Nejad and Mirsaeidi, 2005]

The survey shows that very little laser light are reflected off the sea surface. Reflectance are better in coastal regions where the water is turbid than on the open sea, but reflectance of the 850 nm wavelength that the SF02 uses is bad in all water conditions, and totally useless on the open sea. This concern led to correspondence with the supplier, who confirmed that there can be some issues on flat sea, but that it would work better if the surface is not flat.

The uncertainty in the pickup phase and the uncertainty to the reflection off the sea, makes the choice of this laser altimeter a poor choice.

There exist some ultrasonic range finders (sonars) that are supported out of the box by the APM. MaxSonar produce these sonars and have written a tutorial for the use of sonars where they conclude that the recommended sonar (that they produce) is the XL-MaxSonar-EZ4 [Gross, 2013]. This sonar has a narrow beam, which will be advantageous when mounting; it will make it easier to mount it a place where it gets free sight to the ground instead of reading out the distance to the gripper platform. The resolution of the sonar is 1 cm within the range 20 cm to 765 cm with an update frequency of 10 Hz [Maxbotix, 2012]. The range that it operate within is perfect for this application and ultrasound is extensively used for level measurements which is promising for measurements of altitude above sea. This sonar is included as a part of the UAV.

5.4 Pickup and Deployment Strategy

A goal for the pickup and deployment is to make the process as independent of human interference as possible. One can picture a scenario where the nodes are stored at a specific location on land, the UAV can pick up one by one and fly out to the drop off point and drop it before it flies back, picks up another node and repeats the process. The same process will be conducted in reverse order for the pickup. The UAV picks up a node from the sea, drops it on land, and returns to pick up the next node. The gripper design makes this process possible.

For deployment, the gripper will simply open and the node will fall down into the sea.

The pickup strategy is bit more complicated. The camera is always facing straight down the z-axis of the BODY-frame. This is a challenge for the node tracking, seeing that if the UAV is flying horizontally towards the sensor node, the camera will be facing away from the sensor node. This problem can be reduced by making sure that the UAV is approaching the sensor node from above.

The first part of the pickup is to get to the approximate position of the sensor node (this position is assumed known). When the UAV gets there it will lower the gripper platform and maintain an altitude of for instance 5 m (some experiments needs to be conducted to know how visible the node is at different heights, and find which search height that is the best), while trying to detect the sensor node. When the UAV detects the sensor node it will try to get directly above the sensor node and start descending while maintaining a position straight above the node and the same orientation as the node. If the sight of the node is lost, the UAV should stop the descent and regain sight of the sensor node. The tracking of the sensor node will continue throughout this operation. When the UAV is sufficiently close to the sensor node with the gripper, the gripper will close and the camera will check if the sensor node is caught. If the sensor node is caught the UAV will ascend, raise the gripper platform and return to land.

It is important that the sensor node is in the picture frame throughout the pickup stage. This is due to the fact that the position measurements are quite poor, making it a better strategy to relay on the camera and the position of the sensor node in NED instead of in ECEF.

5.5 Circuit Design

It is important that the motor on the gripper platform stops when the platform is fully lowered and fully raised. This could be solved by continuously measuring the distance to the platform or by discrete position sensors that tells whether the platform is in one of the extreme positions or not, and which extreme position it is in. The simplest and most robust of these solutions is the discrete alternative.

There are several ways of designing a discrete position sensor. The simplest solution is probably to have a circuit that is connected in position and disconnected if not in position. Another solution could be to use IR-sensors and make sure that the beam is cut if the gripper platform is not in position. The solution with the physical connector could be unstable and jitter could be troublesome. The IR-solution requires a bit more complicated hardware, but seems more robust. Hence the IR-solution is implemented.

The Pandaboards I/O-pins operate at a logical voltage level of 1.8 V while other parts of the hardware used in this project operates at 5 V. This makes a translation necessary for the Pandaboard to be able to communicate with the rest of the hardware.

5.5.1 IR-LED Position Sensor

The essence of the IR-LED position sensor is the IR-LED and the photodiode. These need some support circuitry to function. The circuit with the LED is fairly simple. It contains only of the IR-led and a resistor. The resistor is there to limit the current.

The circuit for the photodiode is a bit more complicated. The photodiode chosen is a PIN type diode from Everlight. It is connected reversed biased, and does not conduct ordinarily, but starts to conduct when exposed to IR. The more IR the more it conducts. This last fact means that the signal will wary between 0 and 5 V depending on how much IR it “sees”. In this application the LED and the photodiode are mounted in a way that the photodiode only “sees” the IR when the gripper platform is in position. Hence it is no need for an analog reading, the signal could be connected directly to a digital I/O port to give ether a in position or an out of position value. An opamp used as an voltage follower is included in the design, this isolates the output from the signal source. The operational amplifier used is an MC3405P from Motorola. A resistor is used to limit current.

A decoupling capacitor was added. Decoupling capacitors are used to give noise a path to ground and keep the power in the circuit smooth [Catsoulis, 2005]. The schematics of the circuit design can be seen in Figure 5.8.

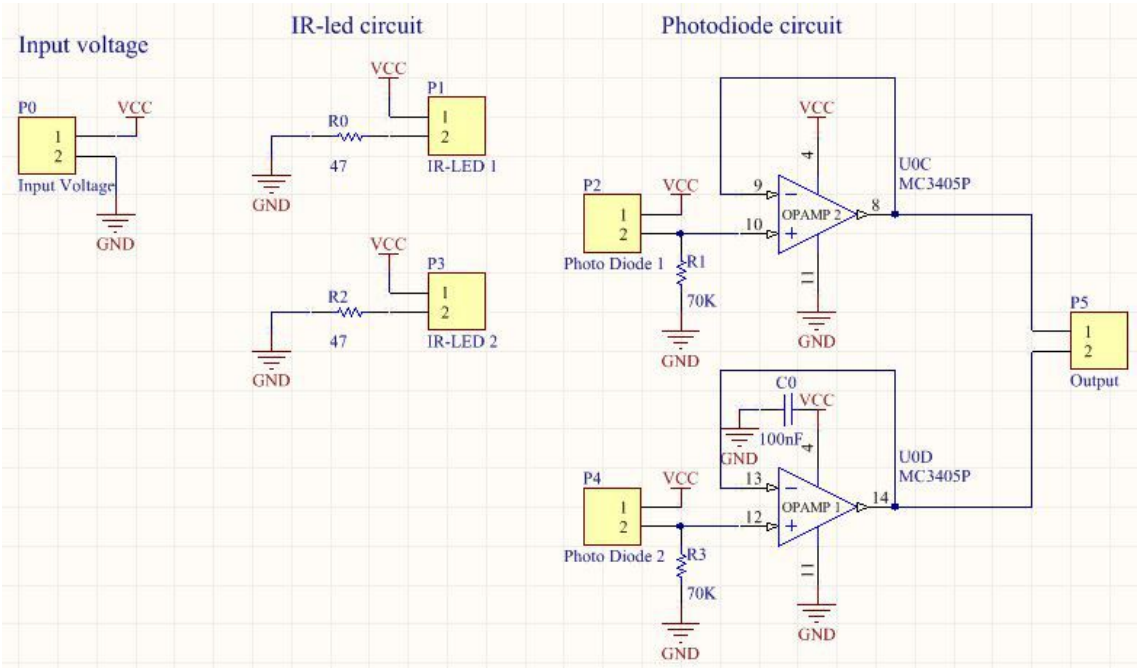


Figure 5.8: Schematics for the IR-LED Position Sensor

5.5.2 Level Translator

Communication will go both to and from the PandaBoard. The level translator will need at least eight channels, three for controlling the H-bridge, one for servo control, two for UART and two for the IR-LED position sensor signals. For this reason the ADG3300BRUZ eight channel bidirectional level translator from Analog Devices was chosen. It is a convenient and easy to use level translator, just connect the desired low voltage and low voltage signals at one side and the desired hight voltage and high voltage signals at the other side. Two decoupling capacitors are also connected. The schematics of the circuit design can be seen in Figure 5.9.

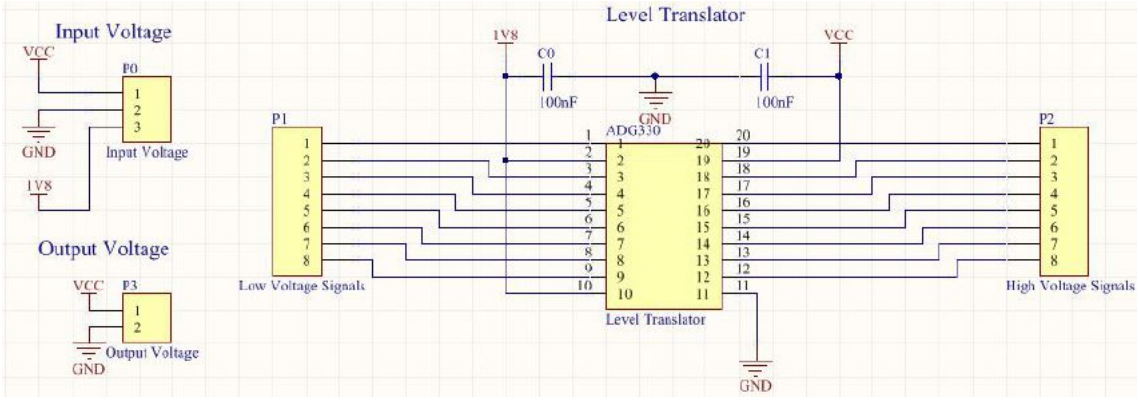


Figure 5.9: Schematics for the Level Translator

5.6 Communication Between the Modules

An overview of the different modules and the communication between them is shown in Figure 5.6.

The PandaBoard communicates with the camera through a dedicated port that uses the MIPI CSI-2 standard. The rest of the communication to and from the PandaBoard goes via the level translator described in the previous section. The servo for the gripper is controlled with a PWM-signal while the H-bridge is controlled using three digital outputs. The communication between the PandaBoard and the APM uses UART and the MAVLink protocol. Relevant information here will be attitude and position data, and set points for the controller in the APM.

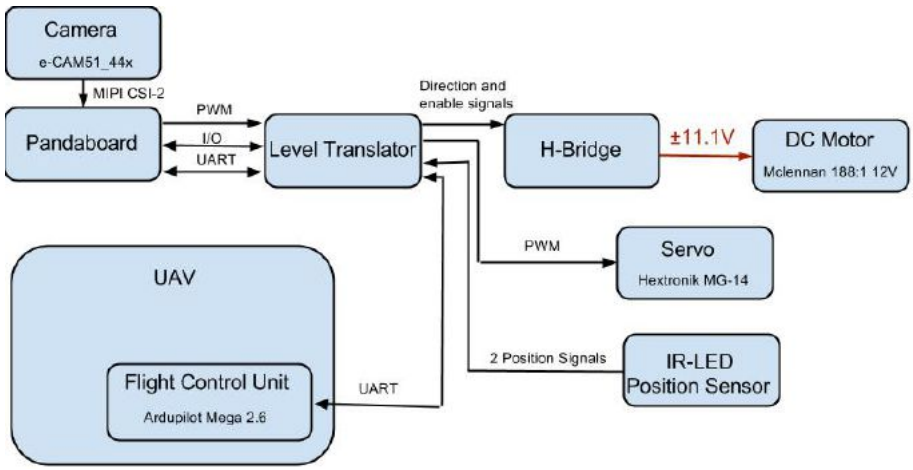


Figure 5.10: Communication between the modules

5.7 Power Supply

The UAV contains a battery and a voltage regulator that regulates the voltage down to 5 V. It is possible to use this voltage source for the parts developed in this project. But using the regulated voltage source could easily result in to much current drawn, which will result in restarts of the APM and the Pandaboard. This could be disastrous for the UAV. To avoid this problem one could use a dedicated voltage regulator for the parts developed in this project, but that will decrease the range of the UAV and increase the chance of losing the UAV in case of problems with the parts developed in this project. Hence the safest and best solution is to add a separate battery and voltage regulator, this will increase weight which will reduce range of the UAV, but this is considered to be a reasonable trade-off. To be able to decide which battery and voltage regulator to use, power demands are investigated and summarized in Table 5.1.

Table 5.1: Power demands

Unit	**Voltage [V]**	**Current Peak [mA]**	**Current Normal [mA]**	**Source of info**
Pandaboard	5	1200	700[1]	Pandaboard FAQ [omappedia.org, 2012]
Servo	5	1500	800	Measured
DC motor	12	50	50	Data sheet [Mclennan, 2013]
Level Translator	5 and 1.8	0.17E-3	5E-3	Data sheet [Analog Devices, 2005]
IR-LED Position Sensor	5	150	450	Measured
Camera	5	150	150	Manufacturer web page [e-con Systems, 2013a]
SUM	-	**3050**	**2150**	

Selecting battery is a trade-off between capacity and weight. The battery should be able to power both the motor and the ICs[2]. Hence a three cell Lithium-Polymer battery is chosen. These kind of batteries deliver 11.1 V, are able to deliver a lot of power fast and able to hold a lot of power in relation to its weight. The specific battery chosen is a 1000 mAh 20-30 C (means that it is able of delivering an instantaneous current of 20-30 times the capacity) battery from Haiyin. The battery is able to support the peak currents, but it might not be able to operate for very long if the servo and motor is used a lot in the mission. The battery will be replaced by a heavier one with greater capacity if necessary.

The H-bridge contains a voltage regulator capable of delivering 500 mA at 5 V. This is not sufficient for the system, so two options arise: to replace the voltage regulator on the H-bridge or to get another one. It was difficult to find a voltage regulator with the needed capacity and with he same footprint as the one on the H-bridge, so the solution became to get another one and use them both. The advantage of using them both is that one gets better safety margins in the power supply for the PandaBoard. The chosen voltage regulator is one from Turningy capable of delivering 3 A steady and 5 A max currents. This means that both peak currents and normal currents are within the limits and the system should be able to operate smoothly. These choices result in the power circuit displayed in Figure 5.11.

[2]Integrated Circuits

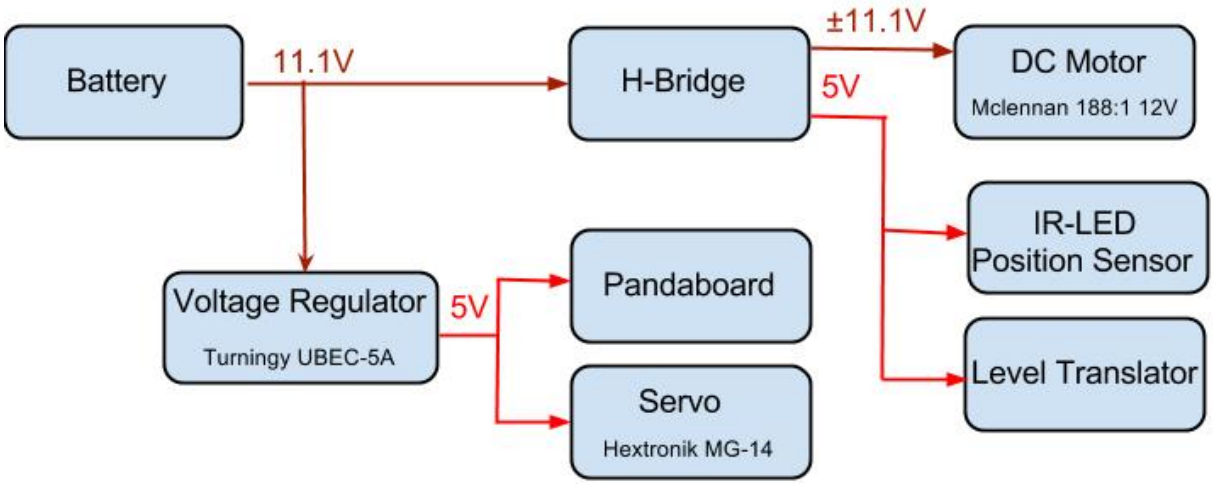


Figure 5.11: Power circuit

Chapter 6

Testing and Results

The implemented mechanical design and the node tracking algorithm are tested to get an understanding of whether they meet the necessary demands for the module to be able to facilitate pickup and deployment of sensor nodes.

6.1 Node Tracking

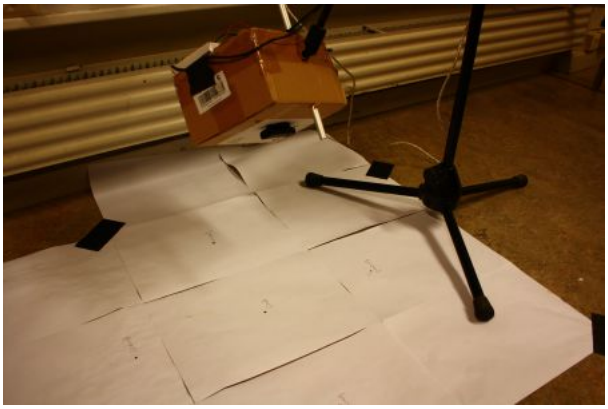
6.1.1 Test Setup

The purpose of the algorithm that is tested is to give a position in NED for an object on the ground based on the height and attitude of the camera and the pixel position of the detected object. This algorithm will be used to find the position of the middle of the sensor node in NED and to find the orientation of the sensor node mount according to NED.

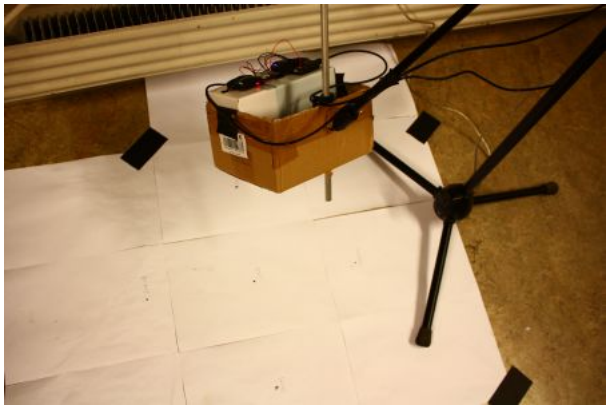
The test was conducted using a computer and a web-camera instead of the PandaBoard with its embedded camera solution. This was done for simplicity and due to the fact that the results will be hardware independent.

The camera, APM and magnetometer were mounted to a microphone stand to be able to test different height and attitudes in a simple manner (see Figures 6.1(a) - 6.1(b)). Mission Planner[1] was used to read out the attitude of the camera and a laser range finder was used to measure the height above the ground of the camera. The use of Mission Planner is shown in Figure 6.1(c). A coordinate system was drawn on the ground below the microphone stand. The points (0, 0), (0, -20), (-20, 0), (0, 20) and (20, 0) were marked (all positions are in centimetres). To get readings that are simple to interpret and compare, the origin was used as a reference and the task became to find the position of the last four points relative to the origin in NED (the x-axis of the coordinate system on the ground points north, while the y-axis points east). In the five first test the heading was measured by eye, and in the last two the heading was measured by the APM. An example of a stream from the camera used to get the positions of the points is shown in Figure 6.1(d).

[1] Ground Control Station for the APM



(a) Camera mounted



(b) APM and magnetometer mounted

(c) Mission Planner

(d) Camera Stream

Figure 6.1: Setup of node tracking test

6.1.2 Result of Test

The mean error of the samples where the heading (yaw angle) was measured by eye is 0.50 cm, while the mean error of the samples where the magnetometer was used is 2.28 cm. The error is calculated as the distance from the true point to the measured point. The results of all the tests are summarized in Table 6.1. The result of the last test are presented in Figure 6.2 to show the effect of inaccurate heading measurements.

Table 6.1: Results of node tracking test

Height [cm]	**Attitude [deg]**			**True Point [cm]**		**Measured Point [cm]**		**Error [cm]**
	Roll	**Pitch**	**Yaw**	**x**	**y**	**x**	**y**	
193	11.61	0.25	0	0	-20	-0.5	-20.0	0.5
				-20	0	-20.3	0.2	0.36
				0	20	0.3	20.2	0.36
				20	0	20.1	0.0	0.1
122	-11.19	-2.45	0	0	-20	0.3	-20.0	0.3
				-20	0	-20.5	0.0	0.5
				0	20	-0.5	20.3	0.58
				20	0	20.3	0.3	0.42
69	-2.00	-4.12	0	0	-20	0.0	-20.2	0.2
				-20	0	-20.6	0.1	0.61
				0	20	-0.2	20.2	0.28
				20	0	19.9	-0.1	0.14
162	-0.11	1.40	-90	0	-20	-0.4	-19.4	0.57
				-20	0	-20.4	0.6	0.72
				0	20	-0.5	20.7	0.86
				20	0	20.0	0.8	0.8
100	-11.24	-1.45	-180	0	-20	-0.2	-20.2	0.28
				-20	0	-20.1	0.0	0.1
				0	20	-0.3	20.0	0.3
				20	0	20.4	-0.3	0.50
154	2.37	-10.43	306.7	0	-20	1.7	-19.6	1.75
				-20	0	-20.3	2.2	2.22
				0	20	-2.2	20.2	2.21
				20	0	20.2	2.3	2.31
155	-10.52	-6.15	243.5	0	-20	-2.5	-19.9	2.50
				-20	0	-20.1	2.3	2.30
				0	20	2.4	20.3	2.42
				20	0	20.4	-2.5	2.53

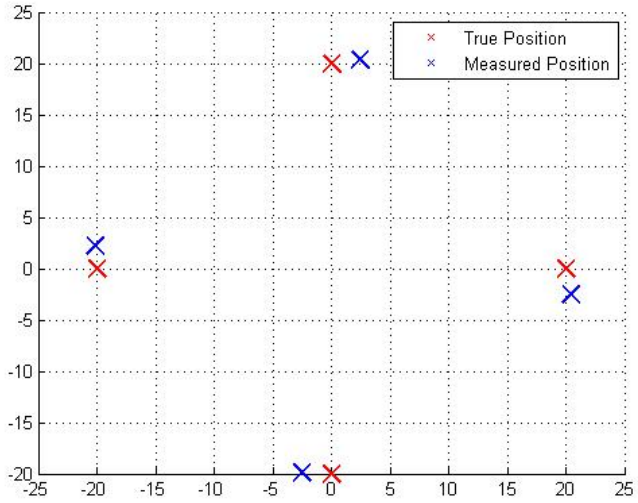


Figure 6.2: Result of test of node tracking using APM for heading measurement

6.2 Pickup Weight Limitations

6.2.1 Test Setup

An Arduino Mega 2560 is used instead of the PandaBoard because of its simplicity and flexibility. Four buttons are connected, one to raise the gripper platform, one to lower the gripper platform, one to open the gripper and one to close the gripper. The Arduino responds to the button clicks by sending control signals to the H-bridge and the servo controlling the gripper.

During the test the platform is mounted with the gripper platform facing straight down. This setup is displayed in Figure 6.3. The gripper platform is raised and lowered with increasingly amount of weight to find its limitations. The time to raise the load are monitored as a measure of performance.

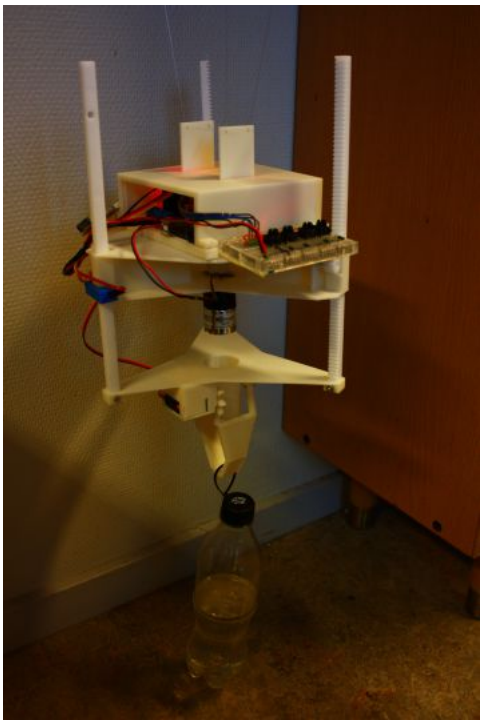


Figure 6.3: Test of pickup weight limitations

6.2.2 Result of Test

The results of the weight limitation test are summarized in Table 6.3.

Table 6.3: Results of the weight limitation test

Weight Load [g]	**Time To Raise Load [s]**	**Comment**
100	13	No problem
200	14	No problem
500	18	No problem
1083	42	Some problems Managed to raise the load by alternating the enable signal

6.3 Functional Tests

6.3.1 Test Setup

In this test the same setup as in the weight limitation test is used, except that the platform is not mounted with the gripper platform facing straight down. The platform is moved around to see if the attitude of the UAV will affect operation. General tests of the grippers abilities and the electronics will also be conducted.

6.3.2 Result of Test

The gripper is responsive, accurate and strong. The servo has no problem holding the gripper closed with a load of 200 g, no matter the attitude of the UAV or if the load is swinging back and forth. The same goes for the raising of the load. The gripper has some barbs on the end to keep the load from opening the gripper. These have some effect, but if the servo is not used to keep the gripper together, one might loose the load.

The electronics works fine. The IR-LED position sensor does a great job ensuring that the gripper platform gets to the desired positions and stays within the legal area.

One should also note that none of the test weights were heavy enough to make the gripper platform go down by itself. This means that one can rely upon the friction of the gears to keep the platform in position without the use of the motor.

Chapter 7

Discussion

7.1 Node Tracking

The camera algorithm showed promising results relating a point on the ground to its own position. The main drawback revealed in the test was that the heading sensor on the APM had poor performance. The results from one of the tests using heading measurements from the APM is shown in Figure 6.2. It is clear from this figure that the measured points are rotated in relation to the true points. This combined with the small errors of 0.5 cm average in the tests where heading were measured by eye, strongly indicates that the position algorithm is accurate, but dependent of good measurements. This algorithm will be used to get the position and orientation of the sensor nodes mount in relation to NED for feedback control. Errors in heading will then not matter too much because the same error will be used by the UAV to navigate. Hence, the error cancels out. On the other hand if the algorithm should be used to track and estimate the sensor nodes position, drift in heading could be troublesome. This combined with the fact that the GPS have an accuracy of only 2.5 m CEP strongly indicates that tracking and estimation of the sensor nodes position will be rather useless. It would be a better solution to make sure that the sensor node is in the camera frame so that the position algorithm can be used directly as feedback to the controller.

7.2 Weight Considerations

There are two main issues connected to weight limitations in this project. For the mechanism to be able to facilitate drop of and recovery of sensor node using an UAV, the UAV need to be able to lift both the mechanism and the sensor node while still being able to navigate and not get a very reduced range. The mechanism should also be able to lift and handle a sensor node.

Combined lifting capacity for the UAV used in this project is approximately 1 kg. Specifications for the sensor node state that the sensor node should weigh 200 g or less. This gives a 800 g weight limitation for the module. The implemented design solution weighs 767 g. In addition some longer legs needed to be manufactured to make room to mount the

mechanism under the UAV. The added weight due to the leg extensions are 39 g. The weight of the added sonar is 6 g. This brings the added weight to the UAV up to a total of 812 g. This is a bit over the specified weight limitations. However, some of the 3D-printed parts could be stripped down a bit to shed sufficient weight to get below the limit. For instance was the box for the PandaBoard designed in a way so that it should be able to attach different design solutions to it. Now when the final design is chosen, the bottom of the box and the platform for the gears can be integrated into each other for a weight saving of up to 38 g. Another way to save weight could be to build some of the parts in materials that are lighter than the plastic used by the 3D printer, for instance aluminium or carbon fiber.

The pickup weight test and the functionality test showed that the motor and the gears had no problems of raising a load of 200 g no matter the attitude of the UAV or if the load is swinging back and forth.

7.3 General Functionality

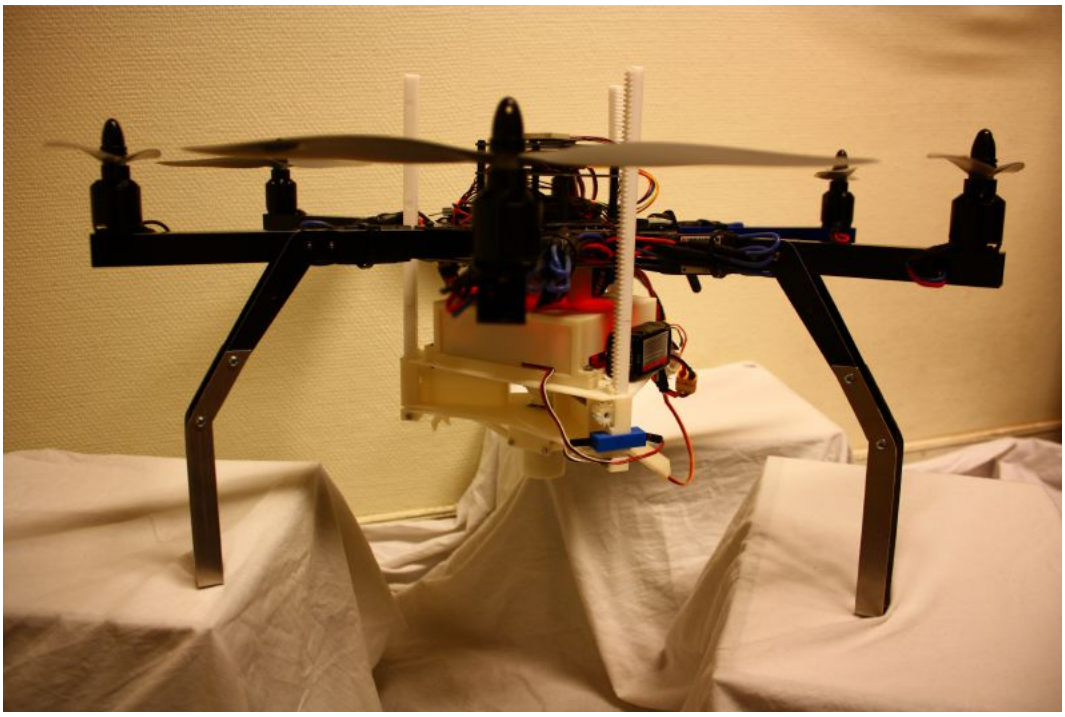
Tests showed promising results for robustness and reliability. One issue that could be troublesome is the battery consumption of the gripper. When the gripper is open, friction will keep it in place so no power is used. But when the gripper is holding the sensor node, some power needs to be used in order to keep the gripper closed. This will increase power demand which in turn will increase the weight of the battery and limit the range of the UAV.

The electronics used in this project have turned out to be reliable, good choices that meets the demands of the project.

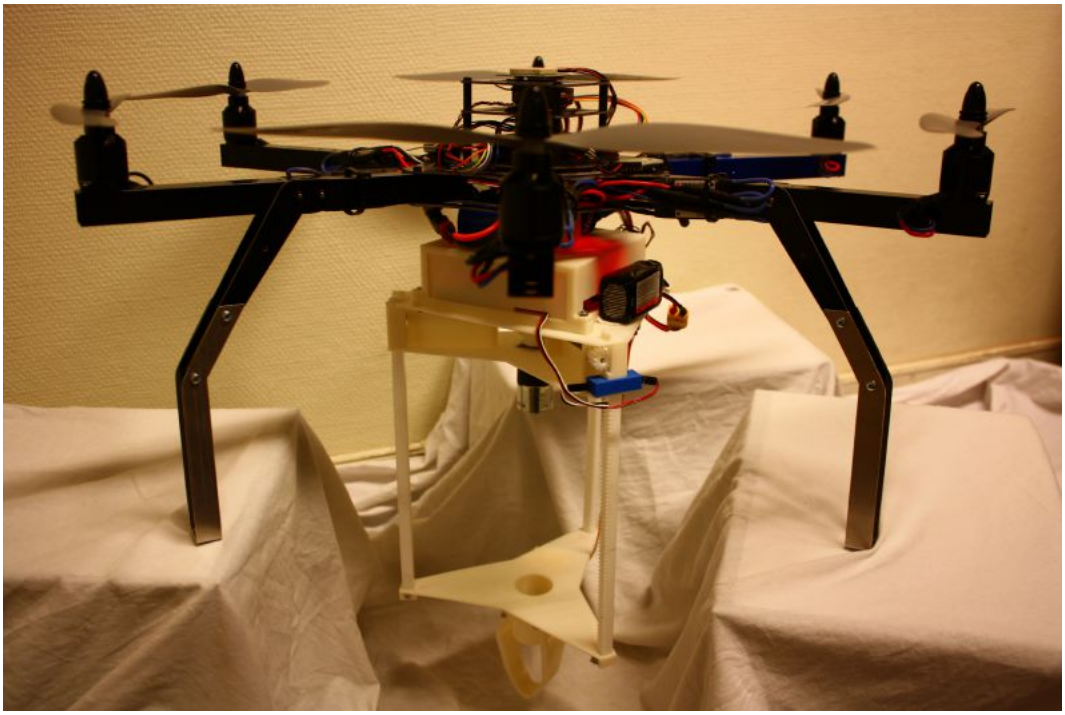
The choice of design solution has turned out to be a good choice that should be able to be used in drop and recovery operations with UAVs. The design is a bit complex, but it functions well and is reliable. The finished design solution is demonstrated in Figure 7.1.

7.4 Future Work

In the continuation of this project, the different parts developed in this project should be interfaced together to form a fully integrated system. The PandaBoard should be interfaced with the APM to be able to take control of the UAV. Control algorithms should be developed to make the UAV able to drop and pickup the sensor node. The parts developed in this project will form a good basis for such operations. The implemented design solution functions well, but with two drawbacks considering weight and power consumption. These drawbacks needs to be monitored and tested further through a flying test with the UAV.



(a) Gripper raised



(b) Gripper lowered

Figure 7.1: Design solution mounted on the UAV

Chapter 8

Conclusion

Several different design solutions for drop and recovery of sensor nodes using UAVs were considered. The implemented solution was a combination of a camera and a lower-able gripper. The camera will be used for feedback to the controller. The reasons for choosing a design with a gripper is that it is flexible and both drop off and pickup operations can be handled in a reliable way. With a gripper design, the controller will know exactly where the gripper is, in contrast to a solution with a hook where it would be difficult to know exactly where the hook is. The drawback is that a gripper would increase power consumption and the precision of the controller needs to be very good for the gripper to catch the sensor node. The reasons for making the platform lower-able is to have a safety margin between the UAV and the sea surface. This will reduce the risk of water damage or in worst case scenarios loss of the UAV.

Tests show promising results for the mechanical design. The mechanical design is a bit heavier than desired, but there are several possibilities for reducing the weight. The functionality of the gripper and the lower mechanism are both precise and robust; they will have no problem handling the sensor node.

The camera application relies upon good measurements. The APM can provide this with exception of position and heading. Measurement errors will affect both the measurement of the position of the sensor node and error used for feedback control while navigating towards the sensor node for pickup. This means that the camera based position measurement are good enough for feedback, but tracking and estimation of the sensor nodes position will produce poor results.

The implemented design solution will be able to facilitate drop and pickup operations using multicopter UAVs in a good fashion with a good control algorithm.

Bibliography

3DRobotics (2013). store.3drobotics.com/products/3dr-gps-ublox-with-compass.

Analog Devices (2005). Adg3300 datasheet. www.analog.com/static/imported-files/data_sheets/ADG3300.pdf.

ArduPilot (2013). Ardupilot development site. dev.ardupilot.com/.

Ardupilot (2013). Setting up flight modes. copter.ardupilot.com/wiki/flight-modes/.

Berberan-Santos, M. N., Bodunov, E. N., and Pogliani, L. (1997). On the barometric formula. *American Journal of Physics*, 65(5):404–412.

Canny, J. (1986). A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(6):679–698.

Catsoulis, J. (2005). *Designing Embedded Hardware*. O'Reilly Media, second edition edition.

e-con Systems (2013a). Camera module features. www.e-consystems.com/OMAP4-MIPI-Camera-Board.asp.

e-con Systems (2013b). e-cam51_44x hardware user manual. www.e-consystems.com/doc_OMAP4_MIPI_Camera.asp.

Fossen, T. I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 1 edition.

Gross, B. (2013). Maxsonar operation on a multi-copter. maxbotix.com/articles/067.htm.

iGage (2013). Gps accuracy. www.igage.com/mp/GPSAccuracy.htm.

Jay Esfandyari, Roberto De Nuccio, G. X. (2010). Introduction to mems gyroscopes. electroiq.com/blog/2010/11/introduction-to-mems-gyroscopes/.

Jiang, G. and Voyles, R. (2013). Hexrotor uav platform enabling dextrous aerial mobile manipulation.

Lightware (2013). Product manual sf02 laser rangefinder. www.lightware.co.za/download/doc/SF02%20-%20Laser%20Rangefinder%20Manual%20Rev%2002.pdf.

Lippiello, V. and Ruggiero, F. (2012). Cartesian impedance control of a uav with a robotic arm. In *10th International IFAC Symposium on Robot Control*.

Liu, C. (2011). *Foundations of MEMS (2nd Edition)*. Prentice Hall, 2 edition.

Mason, W. P. and Thurston, R. N. (1957). Use of piezoresistive materials in the measurement of displacement, force, and torque. *The Journal of the Acoustical Society of America*, 29(10):1096–1101.

Maxbotix (2012). Datasheet for the xl-maxsonar-ez series. `maxbotix.com/documents/XL-MaxSonar-EZ_Datasheet.pdf`.

Mclennan (2013). Geared dc instrument motor 1271 series. `www.farnell.com/datasheets/526521.pdf`.

Mellinger, D., Lindsey, Q., Shomin, M., and Kumar, V. (2011). Design, modeling, estimation and control for aerial grasping and manipulation. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2668–2673.

Merriam Webster (2013). Definition barometer. `www.merriam-webster.com/dictionary/barometer`.

MIPI (2013). Camera interface specifications. `www.mipi.org/specifications/camera-interface#CSI2`.

Mordvintsev, A. and Abid, K. (2013). Introduction to surf (speeded-up robust features). `opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_feature2d/py_surf_intro/py_surf_intro.html`.

Nejad, S. M. and Mirsaeidi, M. H. (2005). Altitude measurement using laser beam reflected from water surface. *Iranian Journal of Electrical & Electronic Engineering*, 1:36.

omappedia.org (2011). Pandaboard test data. `omappedia.org/wiki/Panda_Test_Data`.

omappedia.org (2012). Pandaboard faq. `omappedia.org/wiki/PandaBoard_FAQ`.

OpenCV (2013). Introduction to sift (scale-invariant feature transform). `docs.opencv.org/trunk/doc/py_tutorials/py_feature2d/py_sift_intro/py_sift_intro.html`.

OpenCV.org (2013a). About opencv. `opencv.org/about.html`.

OpenCV.org (2013b). Cascade classifier training. `docs.opencv.org/doc/user_guide/ug_traincascade.html`.

pandaboard.org (2011). Pandaboard es system reference manual. `pandaboard.org/sites/default/files/board_reference/ES/Panda_Board_Spec_DOC-21054_REV0_1.pdf`.

pandaboard.org (2013). Pandaboard es technical specs. `pandaboard.org/content/platform`.

Pounds, P. and Dollar, A. (2014). Aerial grasping from a helicopter uav platform. In Khatib, O., Kumar, V., and Sukhatme, G., editors, *Experimental Robotics*, volume 79 of *Springer Tracts in Advanced Robotics*, pages 269–283. Springer Berlin Heidelberg.

QGroundControl (2013). Mavlink micro air vehicle communication protocol. `qgroundcontrol.org/mavlink/start`.

Spong, M. W., Hutchinson, S., and Vidyasagar, M. (2005). *Robot Modeling and Control*. Wiley, 1 edition.

Store Norske Leksikon (2013). Definition magnetometer. `snl.no/magnetometer`.

Texas Instruments (2013). Omap^{tm} 4 processors. `www.ti.com/lsds/ti/omap-applications-processors/omap-4-processors-products.page`.

Thomas, J., Polin, J., Sreenath, K., and Kumar, V. (2013). Avian-Inspired Grasping for Quadrotor Micro UAVs. In *IDETC/CIE*. ASME.

u blox (2012). Datasheet lea-6 series. `www.u-blox.com/images/downloads/Product_Docs/LEA-6_ProductSummary_%28GPS.G6-HW-09002%29.pdf`.

Vik, B. (2012). *Integrated Satellite and Inertial Navigation Systems*. Deperatment of Engineering Cybernetics, NTNU.