OULUN YLIOPISTO

TEKNILLINEN TIEDEKUNTA

Drone Heading Calculation Indoors

Johannes Jyrkkä

Degree Programme of Mechanical Engineering

Master´s thesis

September 2020

ABSTRACT

Drone heading calculation indoors

Johannes Jyrkkä

University of Oulu, Degree Programme of Mechanical Engineering

Master’s thesis 2020, 71 pp

Supervisor(s) at the university: Toni Liedes

Aim of this master´s thesis was to study drone flying indoors and propose a drone-implemented system that enables the drone heading calculation. In the outdoors, the heading is calculated effectively with a drone´s sensors but using them indoors is limited. Indoor positioning currently has not both low-cost and reliable solution for drone heading calculating. The differences between indoor flying principles and outdoor flying principles of the drone are described in the beginning of the thesis. Then different ways to determine the drone's heading indoors and how they compare with one another are discussed. Finally, two different heading calculation methods are implemented and tested. The methods are based on using multiple location measurements on the drone and using machine vision together with machine learning. Both methods are affordable and are evaluated to see if they could enable drone flying indoors. First method gives out potential results based on testing results, but it needs further development to be able to always provide reliable heading. Second method shows poor results based on verification.

Keywords: Drone, navigation, indoor positioning, machine vision

TIIVISTELMÄ

Dronen lentosuunnan laskenta sisätiloissa

Johannes Jyrkkä

Oulun yliopisto, Konetekniikan tutkinto-ohjelma

Diplomityö 2020, 71 s

Työn ohjaaja(t) yliopistolla: Toni Liedes

Työn tavoitteena oli tutkia dronen lentämistä sisätiloissa ja ehdottaa sitä varten droneen implementoitavaa systeemiä, joka mahdollistaa dronen suunnan laskennan. Ulkona suuntatieto saadaan dronen sensorien avulla, mutta sisätiloissa niiden tarkkuus ei riitä samalla tavalla. Sisätilapaikannuksessa ei ole olemassa sekä edullista että luotettavaa ratkaisua dronen suunnan laskentaan. Työssä perehdytään aluksi dronen lentämisen periaatteisiin sisätiloissa ja miten ne eroavat ulkona lentämisestä. Sitten kerrotaan erilaisista keinoista määrittää dronen suunta sisätiloissa ja niiden keskinäisestä vertailusta. Lopuksi testataan kahta erilaista suunnan-laskenta-menetelmää, jotka perustuvat paikkatiedon käyttöön ja konenäköön yhdessä koneoppimisen kanssa. Menetelmät ovat edullisia ja niiden sopivuutta dronen sisälennätykseen arvioidaan. Ensimmäinen menetelmä antaa hyviä testituloksia mutta tarvitsee lisää jatkokehitystä, jotta se voisi antaa aina luotettavaa suuntatietoa. Toinen menetelmä antaa heikkoja tuloksia verifioinnin perusteella.

Asiasanat: Droonit, navigointi, sisätilapaikannus, konenäkö

PREFACE

This thesis and practical testing part of it were done during spring and summer of 2020 at Nokia Networks and Solutions in Oulu. I would like to thank my co-workers in Nokia for helping with the work and giving me valuable feedback. Special thanks for Janne Väättäri for teaching drone practicalities and supervising on Nokia´s behalf. I would also like to thank supervisor Dr. Toni Liedes from the University of Oulu for continuous support with the thesis and overall great teaching over the years. Thank you my friends for sharing remote coffee breaks during COVID-19 but also for helping and sharing great memories during our studies. Lastly, I would like to thank my family for all the love and support they have provided.

Oulu, 15.9.2020

Johannes Jyrkkä

TABLE OF CONTENTS

ABBREVIATIONS AND SYMBOLS

AI	Artificial Intelligence
CNN	Convolutional neural network
DL	Deep learning
DOF	Degree-of-freedom
GCS	Ground control station
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GUI	Graphical user interface
IMU	Inertial Measurement Units
INS	Inertial Navigation System
IR	Infrared radiation
JSON	JavaScript Object Notation
LIDAR	Light detection and ranging
MAV	Micro Air Vehicles
MEMS	Micro-electro-mechanical systems
MQTT	Message Queuing Telemetry Transport
NIR	Near Infrared
NN	Neural network
RADAR	Radio detection and ranging
RPA	Remotely Piloted Aircraft
RTK	Real Time Kinematic
SODAR	Sonic detection and ranging
TCP/IP	Transmission Control Protocol/Internet Protocol
TDOA	Time Difference of Arrival
TOF	Time-of-flight
TWR	Two Way Ranging
UAS	Unmanned Aircraft Systems
UAV	Unmanned Aerial Vehicle
UDP	User Datagram Protocol
UWB	Ultra-wideband
VR	Virtual Reality

ω	Angular velocity
α	Azimuth
h	Elevation
θ	Heading
λ	Latitude
φ	Longitude
μ_x	Random noise
s_x	Scale error
b_x	Sensor bias

1 INTRODUCTION

Drones have been used for a century, but the use cases have been solely for military purposes until the 2000s. First concept of drone dates to 1849 when Austrian soldiers used unmanned balloons filled with explosives to attack the city of Venice (Consortiq 2020). Improvement in aviation technology resulted in the first pilotless winged aircraft during the first World War by the United States in 1916. The aircraft was called Ruston Proctor Aerial Target and it was controlled with a radio controller similarly to drones are controlled these days. (William and K Munson 1977) The Aerial target was basically a flying bomb meaning that it was designed to crash with other aircrafts or ground targets. Later in 1980s and 1990s, technology advancements and the miniaturization of the associated hardware led to more extensive uses. (Tsouros, Bibi and Sarigiannidis 2019, 2) Drones were used to for example reconnaissance enemies' locations and jam radio communications. In the 2006, drones started in earnest to have non-military use cases when Federal Aviation Admirations issued the first commercial drone permits. Even though the permit opened new opportunities for commercial applications it still took few years before drones started to really interest the common people. At the beginning of 2010s, drones surprisingly started to become more popular as a result of rapid growth in the usage of smartphones. Increased usage reduced prices of microcontrollers, accelerometers and camera sensors. These sensors were ideal for radio-controlled drone hobbyist. (Kashyap Vyas 2018) Drones started to become smaller in size and thus more portable leading to be convenient for surveillance and crowd control use cases for police and firefighters. In the recent years, drones have obtained more advanced features, for example in 2016, DJI introduced Phantom 4 which has computer vision and machine learning technology implemented in it. This enables the drone to avoid obstacles and automatically track people or objects.

Although drones were originally meant for outdoor applications, new potential use cases also want to utilise drones indoors. One example could be packet shipment company, which uses drones for transfer. Currently, sending packets with an automatic outdoor flying drone is possible and it has been done for example by the Amazon (TechCrunch 2019) and Google owned Wing (Medium 2019) companies. But the drones in either companies cannot navigate indoors. This limits the packet transfer for example to the roof of a building, whereas if it could reliably navigate indoors, the packet could be taken further. This would enable for example sending important medicines to a patient or providing maintenance package to someplace with restricted human access.

Indoor navigation is a challenge because of location and orientation. There are multiple location solutions for indoor flying drones. For example, Jin, Ko and Lee (2018) used a stereo vision camera and proved that using it on drone makes position tracking indoors possible. The orientation and more specifically heading calculation on the other hand doesn't have both reliable and low-cost solution

The Aim of this thesis is to solve the heading calculation problem and propose a drone-implemented system that enables the heading calculation indoors. Drone navigation principles and reasons why heading calculation cannot be reliably done indoors are explained in chapter two. Then, in chapter three, different available heading calculation methods are investigated. In addition, heading calculation methods from other industries are examined to find out how the problem has been successfully solved in each of them. Both pros and cons are compared and used to inspect if implementing these features into a drone could enable drone heading calculation. This way, better understanding of the problem and potential solutions are achieved. After that, two different methods will be proposed and tested in chapters four and five. First method is a multi-antenna method which is based on using multiple location points to calculate the heading angle compared with a global reference. Second method uses machine vision together with machine learning to learn heading angles from pictures around the flying area. Objective is to prove if either or both methods could enable reliable drone navigation indoors. Navigation is considered reliable when it gives out sufficiently accurate angle data constantly.

The heading calculation task was given by Nokia and the work was done in cooperation with the company. As will be discussed in chapter three, there are already capable systems that can calculate drone heading indoors. But these methods are expensive and for that reason a specific requirement for the task was that the solution must be low-cost in addition to being reliable. Low-cost can be understood in many ways, so money wise this should be interpreted as a budget being much closer to 10 k€ instead of 100 k€. In addition to price, the low-cost includes the amount of work it needs time wise in this use case. It means that needed work hours for a fully working system using such as setup and calibration should be minimum. This is necessary because testing environments can vary and thus system shouldn’t be dependent on location or certain features in the testing area. Minimum work hours also mean that system can be scaled up easily when necessary.

Heading calculation enables automatically flying the drone indoors. Nokia´s objective is to use drones for automatic and closely repeatable measurement flights. During the flights, a drone´s sensors could be directed to a specific direction. Drone orientation is important due to the flight principle of the drones. But because of the use case for Nokia, only rotation is focused on instead of calculating 6DOF (Degrees of freedom) position of the drone. Some indoor use cases for example intend to fly automatically a certain route and evade any possible obstacles. For this kind of navigation, the 6DOF position would be crucial. It is assumed that a drone´s inbuilt altitude sensors handle stabilizing the drone so that only 2D is concerned.

2 DRONE NAVIGATION INDOORS

A drone is an unmanned aircraft which can be operated both manually and automatically. Sometimes drones are also referred to as UAV (unmanned aerial vehicle) or UAS (unmanned aircraft systems). It usually consists of 2 to 8 rotors and different movements such as turning or moving are accomplished by changing the spin rate of one or more of the rotors. Having multiple rotors sets drones apart from many of the other aircrafts, but it still has similarities with them. Main advantage between drone and other aircrafts such as an- airplane is that a drone can hover and rotate in place with multiple rotors. It can also move up, down and in all 4 directions in addition to just moving forward. Even though a drone can move other directions than it is facing, it still needs the heading information to navigate. Without the heading, it is impossible to navigate from one place to another. For example, moving 50 meters forward doesn't mean anything if you don't know which direction forward is. Figure 1 shows a simple example for drone navigation.

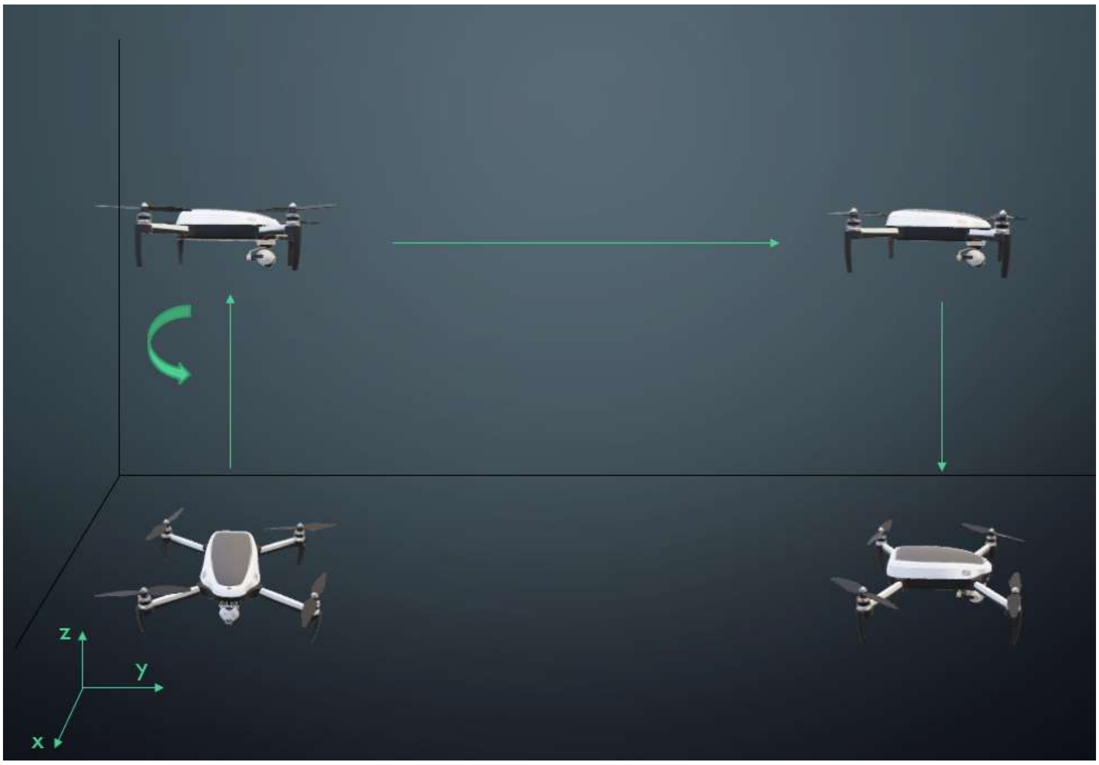

Figure 1. Drone navigation example. Heading information links drone´s 3D coordinates to global coordinates. That enables a simple navigation task such as moving a certain amount to Y-direction in global coordinates.

Outdoors, the drone calculates the heading using on-board sensors. Inertial Navigation System of the drone provides acceptable position data for short flights. But for longer-duration flights, it is obligatory to update navigation because INS accumulate error over time. (Schmidt 2011) For drone heading, a corresponding term for this error is drift, which means the angle between the heading of the object and the desired track. (Jung-Sup Um 2019, 166) For successful navigation, a drift needs to be minimized. Outdoors, this is done using reference heading data from a magnetic field sensor also called a magnetometer. Since the direction of the earth´s magnetic field is close to a constant, a magnetometer can provide an absolute yaw angle which is then compared with the calculated yaw. (Jung-Sup Um 2019, 172 - 173) Lack of this reference data indoors is an important factor for indoors heading calculation problem.

Drone sensors and how each of them affects both navigation and indoor navigation are explained in this chapter. Then, drone navigation principles and challenges for indoor flying are also touch upon.

2.1 Drone sensors

The key technology components in the drone are sensors and wireless networks. (Jung-Sup Um 2019, 33) Most common location sensors are GPS, acceleration sensor, gyroscope, magnetometer and altimeter. These provide necessary navigation data which is then used to fly the drone. Additional sensors can enhance drone´s capability. For example, Lidar or camera can be mounted at the bottom of the drone and used as imaging sensor for different applications such as surveillance or hover stabilization.

Drones have limited sensors availability due to the weight limitations. In general, consumer available drones are small and have decent flying time. Heavy load significantly reduces flight time of the drone. For this reason, both the sensors and parts needed for them such as power supplies and processors must be as light as possible. One big contributor for smaller and lighter sensors on drones has been MEMS (Micro Electromechanical System) technology. (Jung-Sup Um 2019, 34) Due to MEMS, cheaper and smaller sensors are possible to manufacture. Below, is a more detailed description of all sensors and how each is involved in indoor navigation. Especially gyroscope and magnetometer are important because those are heavily involved with heading calculation.

2.1.1 GPS

A global position system is a well-known GNSS (Global Navigation Satellite System) as it was first a satellite navigation system that offered a world-wide satellite positioning service available for public. (Essentials of Satellite Navigation Compendium 2007) Most likely due to that, GPS as a term has become a popular synonym for satellite navigation. The main difference between GPS and GNSS that should be kept in mind is that GNSS-compatible receivers can use more networks beyond GPS system. Meaning that GNSS is more accurate and reliable because it uses more satellites. (TerrisGPS 2015) GPS is arguably the most important sensor for drones outdoors because it is the only reliable way to obtain a drone´s location with high update rate. Due to that, GPS has improved drone navigation significantly and it has enabled autonomous flying missions outdoors. Predefined waypoints can be followed with precise location data. The data is received with an equipped GPS receiver on the drone. The receiver can obtain signals from multiple satellites and use trilateration to determine the relative position. Using more than three satellites also enables altitude calculation and calculating time for each position. Eynard et al. (2010) state that a standard GPS has a vertical precision between 25 meters and 50 meters. Meaning that it is somewhat inaccurate, and a drone usually has a separate altimeter to calculate altitude.

In addition to connecting to multiple satellites to obtain a more accurate position, drones can utilise satellite positioning principle RTK (Real Time Kinematic). RTK means using two satellites data together, one satellite signal is obtained from a stationary reference and another one from the moving drone. Then, combining these two data together a drone can obtain position accuracy of 1 to 5 cm. (manualsdir.com 1999, 45) Using precise timestamps with locations both heading and ground speed for drone can be calculated. GPS signal loses strength when passing through objects and that is why it is not reliable indoors. This is the main reason why both position and heading must be calculated using alternative ways compared with outdoors.

2.1.2 Accelerometer

Accelerometer measures proper acceleration forces. Accelerometer output is the following:

$$\hat{a} = (1 + s_a)a + b_a + \mu_a \quad (1)$$

where â is the measurement value, s_a is the scale error, a is the true value, b_a is the accelerometer bias, and μ_a is the random noise. (Chao et al. 2010) Multi-axis accelerometers can determine both magnitude and the direction of the proper acceleration as a vector quantity. Meaning that it can detect both orientation changes and linear movements. MEMS technology has contributed to the accelerometers in a way that accelerometers are increasingly present in portable electronic devices to detect changes in position. (Wikipedia Contributors 2019)

For drones, accelerometers serve a few purposes, it can stabilize the drone and provide navigation inputs. Commonly, a drone has 3 accelerometers for 3 different axes, as shown in Figure 2. Those can detect movement in X, Y and Z axes. Because a drone is constantly under continuous force of gravity, an accelerometer can be also used to determine the pitch and roll rotations of the drone. An accelerometer can´t detect a yaw because in that rotation gravity force doesn't change. Sudden changes in drone speed such as a gust of wind are also detected, and it can be compensated to improve hovering stability. Acceleration can be integrated twice to obtain position data. This position data can be fused together with GPS to achieve even higher accuracy in location.

2.1.3 Gyroscope

Gyroscope is a very essential sensor in drone navigation. A gyro is a spinning wheel (mass) which measures orientation and angular velocity from the spinning mass. (pilotfriend.com 2018) The spinning mass maintains rotation and stays stable in axis of rotation. This way, gyroscope resists turning and thus detects any deviation. For drones, gyroscope measures rotation movement around the aircraft principal axes yaw, pitch and roll. Its main use in a drone is to maintain orientation for stable hover similar to an accelerometer. These days, gyroscopes are so essential to the stable operation of a drone that a malfunction in the gyro sensors is considered as a fatal error and it will result in ending an automatic mission or in the worst case in a crash.

Gyroscope accumulates error due to gyro bias and scale-factor instability. The gyro error model can be expressed as equation (2):

$$\hat{\omega} = (1 + s_g)\omega + b_g + \mu_g \,, \qquad (2)$$

where $\hat{\omega}$ is measurement value, s_g is the scale error, is the true value, b_g is the gyro bias and μ_g is the random noise. (Chao et al. 2010) Gyro bias is a temperature-sensitive variable error, which affects the measurement all the time, thus, a gyro gives out output even while it is not moving. Scale-factor on the other hand occurs only when the object is moving. (El-Rabbany 2002, 121) Because the output is angular velocity, it must be integrated once to obtain heading information. This further increases the accumulating error and that is why gyroscope needs reference data to maintain accurate orientation. Outdoors, this reference data is obtained from GPS and magnetometer sensors.

Gyroscopes and accelerometers are usually together called IMU (Inertial measurement unit). Combining 3 accelerometers and 3 gyroscopes is called 6DOF IMU. An example of it is given in Figure 2.

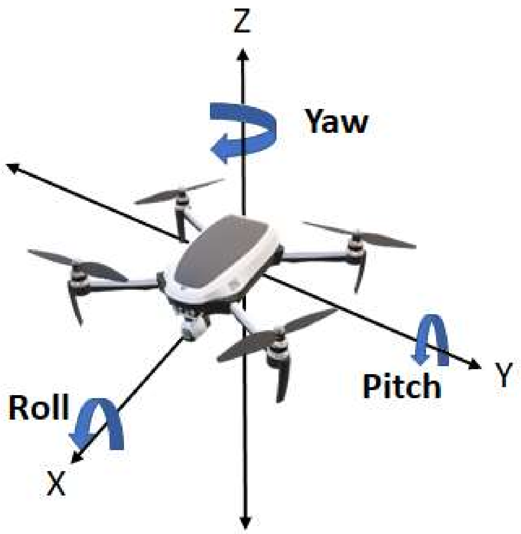

Figure 2. Accelerometers can detect movement along X, Y and Z axes. These also detect roll and pitch from changes in gravity forces. Gyroscopes measure yaw rotation in addition to roll and pitch. Combining 3 accelerometers and 3 gyroscopes forms 6DOF IMU.

2.1.4 Magnetometer

Magnetometer measures strength and direction of the magnetic field. For a drone and other aircrafts in general, it is used to determine the direction of Magnetic North. The Magnetic North is the direction of the horizontal component of the Earth's magnetic field and from that direction a drone can determine its heading. (uavnavigation.com 2018) Magnetic North doesn't change direction. It is used as a reference direction to compensate drift error for IMU and GPS heading. Even though Magnetic North doesn't change direction the magnetometer can still easily be biased. There are two type of categories which can cause interference, hard and soft irons. Hard-iron distortion comes from materials that emit a magnetic field such as magnet in a speaker. When material doesn't produce a magnetic field, but it influences the magnetic field, it is called soft-iron distortion. (FierceElectronics 2009) Soft-iron distortion occurs with metal and for example with a car or other electrical devices. Errors from magnetic disturbances can be reduced with filtering and sensor fusion but not completely removed. For example, Wang and Gao (2005, 155) used a deep learning algorithm to calibrate a magnetic compass to reduce disturbances. In many cases, interfering obstacles can´t be avoided indoors and that is why magnetometer is either unreliable or completely useless in indoor use cases.

2.1.5 Altimeter

Altimeter also called a rangefinder measure a drone´s altitude compared with a fix reference which is usually the ground. Altitude estimation of the drone is extremely important with flight manoeuvres such as landing, take-off and steady flying. (Cherian et al. 2009) Altitude can be obtained by different technologies. Barometer is the most common altitude sensor used in airplanes and vehicles; it measures the altitude from the atmospheric pressure. (Nelson 1998, 26) Other common commercial rangefinders are SONAR, LIDAR and RADAR based. All three of them have the same basic principle where altitude is calculated from arrival time of the reflected signal. The only difference is whether the signal is sonar, laser or electromagnetic pulse. These sensors are characterised as TOF sensors. (Hentschke et al. 2018)

Newest additions are optical sensors which utilise computer vision to estimate altitude. A drone is for example equipped with a downward-looking camera which takes pictures. Pictures from different heights and environments are used to teach the machine learning algorithm which then calculates altitude from features extracted from a camera´s pictures. As an advantage to the computer vision, the on-board camera is lighter and less power hungry compared with other sensors (e.g. laser). But it can´t be used in applications with texture less surfaces and it is limited to low altitude situations because pictures start to become more imprecise when altitude rises (Cherian et al. 2009).

Altimeter becomes crucial in altitude definition indoors. Precise altitude is needed because most indoor use cases have ceiling and in the worst-case drone can crash if altitude information varies too much.

2.2 Drone navigation

In this section drone navigation and related matters are discussed. Navigation principles have great impact on the heading calculation because it defines the needed accuracy for it. Automated drone navigation and ground control station are touch upon because those are related to Nokia´s objective to fly closely repeatable flights.

2.2.1 Drone navigation principles

Without few exceptions, drone navigation means using GPS for navigation. Outdoors the working principle of the GPS based navigation works in a similar way as dead reckoning. (Jung-Sup Um 2019, 144) Dead reckoning in navigation means calculation of up-to-date position from a previously known position based on known direction and speed. Dead reckoning is subject to cumulative errors and GPS navigation has made it old-fashioned. (Wikipedia 2020) Because a drone cannot access GPS indoors, it must rely on dead reckoning navigation. Due to that fact, successful indoor navigation needs reliable heading information in addition to position. Reliability is achieved by updating location data and compensating errors from heading calculation.

Drone navigation is usually done using waypoints. A drone flies from point-to-point according to the parameters. These parameters include longitude, latitude, elevation and azimuths. The first three parameters are respectively denoted by φ, λ, h and they come from Geodetic coordinate system. (Cai, Chen and Lee 2011) In the system, longitude is a line that intersects the defined position and is parallel to a prime meridian line. Latitude is also a line that intersects the defined position, but it is parallel to an equator line. (Patrik et al. 2019) Elevation means the distance between drone and surface of the earth. Combining these three values, a drone exact position can be defined. Last parameter azimuths indicate an angle from true north. Using these four parameters, a drone can navigate using waypoints. The use of Geodetic coordinate system has because common for drone navigation due to use of GPS. An example of the waypoint flying can be seen in Figure 3.

Indoor navigation heading accuracy needs to be highlighted. Because dead reckoning is based on estimations, it has accuracy and acceptable limits. When navigation happens within these boundaries, it can be considered successful. Outdoors automatic waypoint following can be realised with 1–5 ° heading accuracy even using low cost GNSS receivers, MEMS-based inertial sensors and magnetometer. (Vetrella et al. 2016, 2) Based on that heading accuracy, indoor should also have target accuracy under 5 °. If the heading overshoots these limits, a drone will gradually try to compensate error by moving sideways. This can result in a drone rotating around the waypoint and never getting there. For indoor navigation, particularly this kind of behaviour is critical for two reasons. First reason is that indoor space is limited, so an over compensated movement can result in a drone crashing to obstacles or walls. Other reason is that due to limited space waypoints must be close to each other. That gives a drone less time to compensate the error and thus a bigger adjustment movement is needed. This further emphasizes the risk of crashing the drone.

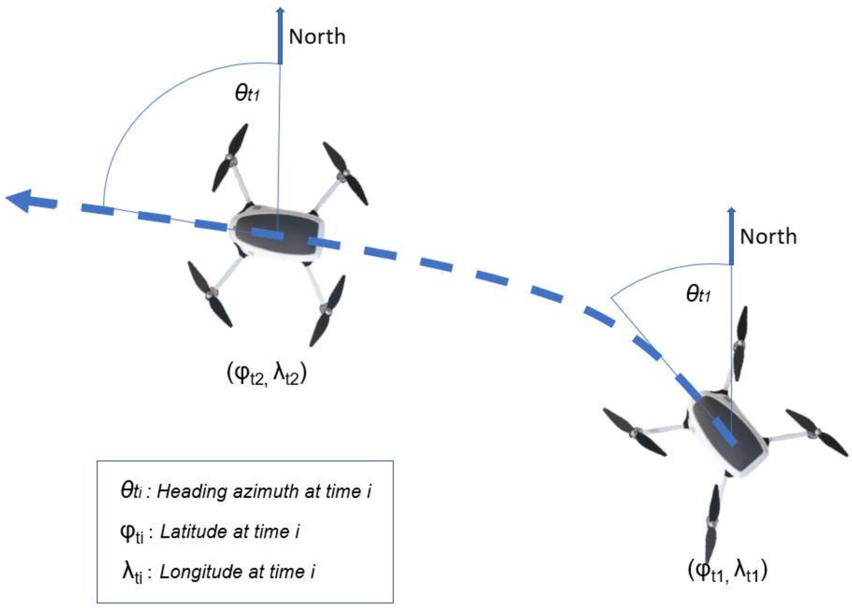

Figure 3. Dead reckoning for drone navigation.

2.2.2 Ground control station

Waypoints and other navigation related issues are handled with GCS (ground control station). Drone GCS is the complete ground-based hardware and software systems used to control the UAV. It acts as a UI for the drone operator and it includes all the necessary HW and SW before and during the flight mission. For example, HW includes telemetry and data links to the drone. SW shows the operator map and different outputs from all the drone´s sensors. There are many different commercial GSC available for both desktops and smartphones. A few worth mentioning for desktop are Mission planner and APM Planner 2 and for smartphones MAVPilot and DroidPlanner 3. (ardupilot.org 2020)

2.2.3 Automated drone navigation

A drone can be operated both manually and autonomously. Manual operation means controlling a drone for example with a radio controller. Automated drone navigation means that a drone uses inputs it receives from different sensors to navigate in an automatic manner. Therefore, it doesn’t require any man-made signals from outside the drone. (Schmidt 2011) Automatic navigation can mean for example following a certain object or flying predefined waypoints. It also includes hovering and stabilising at an unchanging position. These kinds of actions are executed using a closed-loop system. Almost without exception, closed-loop systems for drones are proportional-integral-derivative controllers. PID is a vastly used controller which has a simple structure and in addition it is easy to tune.

Automatic navigation offers many advantages compared to the manual operation. A drone in autopilot mode is much more stable than manually controlled one. This is especially useful for example in aerial photography where precision is necessary. Another upside is that automatic navigation enables flying the same route almost identically multiple times. This kind of flying enables doing measurement flights with a drone because margin of error due to flight decreases.

Currently, automatic navigation indoors is not common but not impossible either. Gageik et al. (2013) were able to do autonomous drone navigation indoors using an optical flow sensor which points downward. An optical flow sensor can detect visual motion and output displacement measurement based on changing pixel positions. This way, a sensor can point to the ground and give out position data for the drone. Using it together with commercial IMU, Gageik et al. obtained accuracy of 20 cm for autonomous navigation and 10 cm for position hold. But the downside is that they don't have any way to compensate a drift from the gyroscope. An effect from the drift is visualized in their research by pointing out that for 6 minutes hovering, the drone rotates about 13 degrees. Another drawback in the system is that position is not tied to any global coordinate system. (X_0,Y_0) coordinates are defined to be where the mission positioning starts. There are also other challenges for indoor navigation which will be touch upon in the last part of this chapter.

2.3 Challenges for indoors flying

Indoor flying means manoeuvring the drone inside a building or a similar structure such as a subway tunnel. Most important difference regarding the traditional outdoor flying is that some of the drone´s sensors won´t work the way they do indoors as already previously touch upon in this chapter. The crucial ones are GPS and magnetometer. Even though a drone could receive satellite signal thru windows and other openings, it is too varied to be used as it is. Instead, position information must be obtained using other methods. Similarly, even though a magnetometer could work indoors, it is affected by sudden changes in magnetic fields like a car parked next to the flying area. It biases the magnetometer thus making the heading information incorrect. Precise position and orientation are important because indoors tend to have less space compared with outdoors. Due to that, a drone can more easily bump into obstacles near the correct waypoints. De Croon and De Wagter (2018) stated that in addition to less space, automatic indoor navigation poses two more problems, different visual appearance and denied visibility of the sky. A different visual appearance means that indoors tend to have less colours and textures compared with outdoors. An example of this can be seen in Figure 4. This affects for example using machine vision algorithms because outdoors usually have fewer visual clues compared with outdoors. Visibility of the sky enables for example attitude estimation from main light source and polarization of the sky could be used as compass. (Chahl and Mizutani 2012, pp.289–297, Pfeifer et al. 1998)

Figure 4. Comparison between indoor and outdoor environments.

Safety indoors is not regulated the same way as outdoors even though arguably outdoors flying is safer than indoors due to more available sensor data. According the Federal Aviation Administration, FAA rules and regulations apply to operations conducted outdoors. (Faa.gov 2019) This means that actions to ensure safe flights are left for the pilots to take care of. Previously mentioned space limitations indoor pose danger but in addition to that indoor has potential interference signals which can affect flying. These interferences are for example Wi-fi, FM radio signals and Bluetooth.

One more challenge for indoor navigation is the lack of suitable SW dedicated to it. Indoor navigation must work with the already existing autonomous robotic systems. (Tiemann, Schweikowski and Wietfeld 2015) As an example, the position is obtained from GPS receiver, so the indoor coordinates have two options to work. Either coordinates are converted to longitude and latitude values to fit the current system. Otherwise, the system must be modified so that it can use the indoor coordinates. This potentially poses a problem because many platforms are not open source.

3 HEADING CALCULATION METHODS

There are several methods that can be utilised to calculate heading of an object. Purpose of this chapter is to identify and research these methods. Because focus is on indoor heading calculation, outdoor navigation options are mostly bypassed. Gade (2016) argued that categorising heading methods is not evident. This is due to several different methods available which are seemingly unrelated to each other. Moreover, different methods can be used together by turns or simultaneously and thus makes defining even more harder. Nevertheless, Gade summarised different methods to seven categories. The different methods and how they can potentially be used with drone indoor heading calculation are shown in Table 1.

Table 1. Heading calculation methods and their suitability for drone indoor usage

Method	Usage for drone indoor use case	Reason
1. Magnetic Compass	No	Magnetometer is not reliable indoors
2. Gyro compassing	No	Gyrocompass is too expensive
3. Observing multiple external objects	Yes	Can be done for example with downward looking camera
4. Measure bearing to object with known position	Yes	Can be done for example with computer vision
5. Multi-antenna GNSS	Yes	Indoor can be done with other position technics instead of GNSS
6. Vehicle velocity	No	Drone movement is needed
7. Vehicle acceleration	No	Drone movement is needed

In this chapter, potential methods are evaluated. Examples from other research fields where heading problem has been successfully solved are examined as well, assuming they would be suitable a for drone in this use case. Example of these are Virtual Reality and pedestrian dead reckoning. From unusable methods, magnetic and gyro compassing are briefly discussed because both have main roles in need for drone heading calculation indoors. Heading calculation from vehicle velocity and acceleration are omitted. Since a drone must be able have heading even when not moving and those methods become more accurate with higher speed and acceleration. Neither high velocity nor acceleration is usually possible because limitations of movement in indoor spaces. Another argument against heading from acceleration is that even though a drone has available acceleration data from an accelerometer, the data is mostly not fit to be used. This is because angle estimates from accelerometers suffer from high frequency noise when the drones are moving. (Chao et al. 2010) But this might change soon because attitude estimation for drones has recently gained great attention to it. Al-Sharman et al. (2020) were able to train a deep neural network with measurement noise from IMU and use the network to filter out the noise. Results from their measurement prove that attitude estimation works greatly with hover.

3.1 Magnetic and gyro compassing

Before turning to with the working and potential heading methods, magnetic and gyro compassing should be tackled first. The lack of the magnetic compass is a key factor for the need of indoor heading calculation. It is also related to IMU data fusion, which is often referred when talking about drone heading calculation.

A magnetic compass can´t be reliably used indoors because electromagnetic interferences. These interferences may be caused for example by a car next to the drone area. Lack of magnetometer is a big lose because a magnetic compass is probably the simplest heading calculation method. But in addition to that, it is also the most common way to calibrate the IMU. IMU, and more specifically the gyro, can give heading information for a short period of time. But because the gyro accumulates errors over time, it quickly becomes useless without the calibration. That is why IMU can´t be relied on for indoor heading calculation unless it is calibrated by other means.

There also exist high quality gyros which don't need outsourcing calibration to work. These military grade gyros could be used for heading calculation and the method is referred to as gyro compassing. But this method is too expensive to be used with consumer UAVs instead it is used for example in airplanes and submarines.

3.2 Observing multiple external objects

In this method, a heading vector is calculated from two known objects O_1 and O_2. The objects can be either on the drone or in the environment. These objects can be virtually anything and those can be recognised with many sensors, such as lidar, sonar and cameras. For example, objects could be features of wall border from a picture with a downward-looking camera on the drone. The method doesn't require the position of the drone for it to work but it is worth mentioning that it must be obtained by other means to navigate a drone.

First example of the method is one of the most potential heading calculations for a drone on the market. It is 3D tracking with OptiTrack. It can follow a fast-moving object with update rate up to 360 fps with high accuracy. The error in position is less than 0.3 mm and rotational error is less than 0.05 ° (OptiTrack 2020a) Because the accuracy is top tier, it enables OptiTrack to be a reference system for other heading and position calculating systems. For example, OptiTrack was used as a reference system for Oculus quest when it was in development state. (tech.fb.com 2019) It has also been very successful in drone industry which was proven when the Drone Racing League adopted this technology (sporttechie.com 2019). DRL used OptiTrack to record exact position of the drone with update rate of 300 frames per second. The data was used to compare a real drone position to a virtual drone in computer simulator. In the article, it was stated that precise position was the factor which made a realistic simulator possible. Working principle of the OptiTrack is the following: The position of a moving object is captured using multiple synchronised cameras which are installed around the target. 2D images are captured from each camera and using overlapping 2D positions, 3D position is calculated using triangulation. The accuracy is optimised with the use of trackable markers and high computational filtering algorithms. (OptiTrack 2020b) The best of the market is not the cheapest one which comes as no surprise. Relatively high price is not problem for many industries when the quality is the highest priority. But for the use case in this thesis, it is not a suitable system for that very reason.

Another example where the observable objects are in indoor environment instead of on the drone was accomplished by Bills, Chen and Saxena (2011). Bills et al. presented a method which used together sonars and cameras to detect corridors and stairs. During flying the heading of the drone was adjusted so that a drone would be in the centre of the stair or corridor based on the data obtained from sensors. Even though heading and automatic flying were successfully accomplished this method can´t be used to freely navigate indoors. It is limited to specific targets and thus can´t be used to freely navigate indoors.

3.3 Measure bearing to object with known position

Working principle of the second potential method is to know own position B_1 and other known position O_1 in environment. Then, heading is calculated from the vector $\vec{P}_{B1O1}$. This method has been successfully implemented indoors with two different ways in VR.

Firstly, it is done with a laser in the VR systems such as HTC Vive. Precise heading tracking is crucial in VR because the user will feel nauseous if the virtual reality doesn't change similarly as the real-world view. In HTC Vive, which was launched 2016, the tracking was done using a laser implemented system. The system hardware consists of laser emitting base station and receiving headset with laser detecting photodiodes. The base station emits horizontal and vertical IR light scans. Each scan is detected in the headset with photodiodes in the pockets in front of the headset. Every scan gives out various angles and distances which are then calculated relative to the base station. This provides reference data for orientation detection and it is a clear replacement for a magnetometer. Meaning that the drift error can be compensated and together with IMU system of the headset, precise orientation is obtained. (Steven 2019)

Secondly, in 2019, was launched Oculus Quest which has a different take on the heading tracking problem compared to other virtual reality equipment's. Fundamental difference between the systems is that Quest is entirely wireless, and it doesn't use any base stations for tracking. Instead, tracking is done using SLAM (simultaneous localization and mapping). It is done by combining computer vision with highly trained machine learning algorithms. Both the algorithms and VR content are processed with a mobile chipset. (tech.fb.com 2019)

6DOF headset tracking in Oculus Quest is accomplished by combining data from IMUs and cameras. Position and specifically orientation are initially calculated from IMU input data. Then, 3D map, generated from image data from ultra-wide-angle cameras, is used to pinpoint landmarks such as corners from indoors. These landmarks are then used continually to check a position and compensate for drift error. This is enough to replace a magnetometer. (ai.facebook.com 2019) In addition to data fusion, Oculus Quest uses specifically for VR purpose developed algorithms which are refined with machine learning. The algorithms predict next human movements before they occur and thus the anticipation speeds up the process and increases accuracy. The heading calculation principle in Oculus Quest could work with a drone. But current algorithms are not compatible with a drone because it has been optimised with machine learning for a human user not for a drone. The difference between human and drone is for example that a drone can move more freely up and down in various heights and rotate faster than human because drones don't feel nauseous. The whole system should be designed to fit the drone from the start for it to work.

At the time of writing this thesis there are already some attempts to use VR technology with the drone. For example, Bitcraze company sells a component for their Crazyflie drone that uses HTC Vive base stations for drone positioning. While still being in an early access state, it also has capability to calculate drone pose (Bitcraze Store 2020). Also, Espinosa and Rubenstein (2018) were able to use the self-made controller together with HTC Vive to hold drone position in a virtual box and obtain 3D position. But they didn't do actual flight with automatic control and feedback.

Machine vision has many upsides compared to other sensors such as low weight, rich amount of data and option to use for other purposes also such as inspection and so on. For that reason, a machine vision-based method will be implemented and tested in this thesis as well.

3.4 Multi-antenna

Last potential indoor heading calculation is done using multiple antennas on the drone. Outdoor, this would be done using the GPS position. For reference, Hirokawa and Ebinuma (2009) presented tightly coupled GPS/INS with multiple GPS antennas. Using two auxiliary GPS antennas together with IMU module Hirokawa and Ebinuma obtained 0,1 yaw degree accuracy with an experiment using a manned aircraft. But indoors instead of

GPS the position must be obtained other ways. Currently available alternatives are Bluetooth, Wi-Fi (IEEE 802.11) and UWB. However, Bluetooth has so low accuracy that it is usually discarded for most of the indoor position applications. (Poza-Luján et al. 2018, 132)

Using a single antenna, heading is obtained when a drone is moving. Heading vector is updated each time when a drone obtains new position. The downside is that heading cannot be calculated when a drone is staying in one place. This problem is solved by adding a second antenna and then two positions can be measured during a single moment. This principle is shown in Figure 5. From at least two different points of the drone, vector $\vec{P}_{B1B2}$ is drawn between measurement points P_{B1} and P_{B2}. Then, heading θ_B is calculated with trigonometry by comparing $\vec{P}_{B1B2}$ to a known vector $\vec{P}_{A1A2}$. $\vec{P}_{A1A2}$ can be directly pointing at magnetic north or it can be aligned with it by rotating it θ_A.

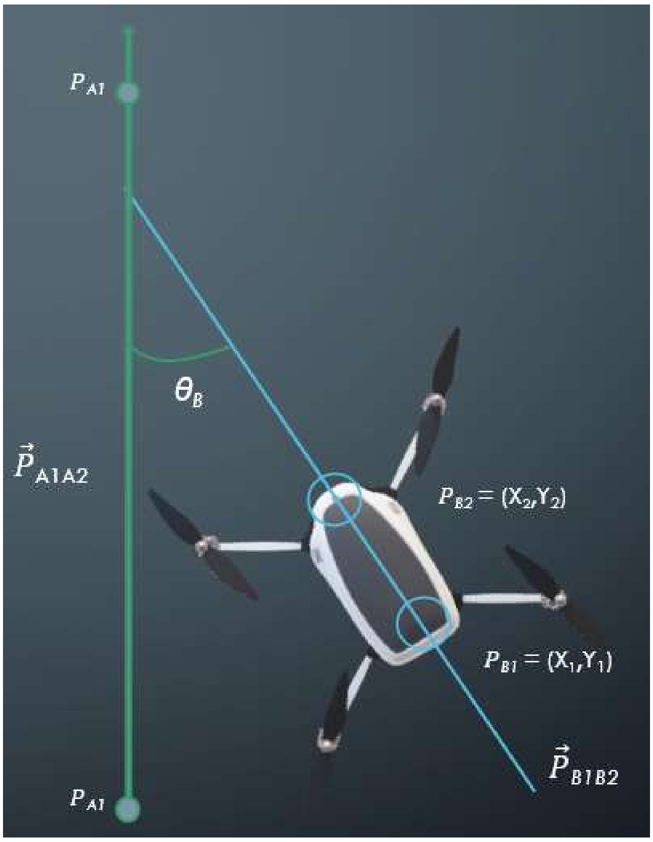

Figure 5. Multi-antenna heading calculation method. Heading θ_B is calculated by comparing vectors $\vec{P}_{A1A2}$ and $\vec{P}_{B1B2}$.

Cho, Kim and Kim (2012) proved that the multi-antenna principle can be used indoors with Wi-Fi. The heading calculation was used for a PRD (pedestrian dead reckoning) application. Cho, Kim and Kim obtained azimuth error between 20–30 ° degrees and average position error of 3 meters. This accuracy is too low to enable drone heading calculation indoors. The most efficient way to obtain a better heading angle is to improve position accuracy. That is why Ultra-wideband is the next potential candidate. UWB positioning has been tested before with a drone before and it has been successful. Tiemann, Schweikowski and Wietfeld (2015) implemented UWB system on the UAV which was able to hold its position in a radius of under 50 cm. These results are clearly better than the ones obtained with Wi-Fi positioning. For these reasons, UWB based multi-antenna heading calculation will be implemented and tested in chapters 4 and 5.

4 TESTING OBJECTIVES AND METHODS

In this work, two implement ready heading calculation methods are proposed. First one is multi-antenna method and the other one is heading with machine vision together with machine learning. The methods were chosen based on the criteria from Nokia as explained in the first chapter. Integrating the methods into a drone and successfully using them to navigate automatically was decided to be beyond the scope of this thesis. Meaning that the objective is to only propose methods which are proven to be working with tests. First, methods´ different working principles are explained. Then, testing plans are introduced, and lastly, SW and HW components are examined.

4.1 Methods working principle

Multi-antenna heading calculation

As stated in chapter 3, multi-antenna method uses two measurement points on the drone to determine a vector which is then compared to a known vector to obtain heading. The measurement points will be obtained from UWB system. Accuracy is improved by using a total of 4 tags instead of only 2 tags. Tags will be placed in cross formation, each 50 cm away from the centre. The position data is assumed to be better compared to a single tag because an average from all the tags can be calculated. Also, heading calculation can be done using two pairs of antennas and thus average of two measurements can also be used.

Machine vision-based heading calculation

For the heading calculation with machine vision, two principles were tested in this thesis. Both use a convolutional neural network to predict the current heading angle. CNN or convnets is a class of deep learning neural networks. It includes convolution neural layer, which together with backpropagation enables to learn features from pictures. A common usage example for it is to recognise handwritten digits. (Schmidhuber 2015, 90-93) In this use case, one principle uses CNN for image recognition and other one uses it for regression.

In image recognition, AI (artificial intelligence) is trained with pictures from different angles. The camera stays in one place and rotates for full 360 degrees while taking pictures. These pictures form a training data which is called a dataset. Every angle is an output and for each output hundreds to thousands of images are needed to properly train the neural network which is the main part in the deep learning algorithm. After the neural network is trained, it will have a total of 360 outputs. Each output will present the prediction for a single angle degree. After training, machine vision will take a new picture and provide the neural network a new input. The input gives out prediction for each output with different certainties, usually indicated with %-mark. Meaning that image recognition is made to the input and each output is considered as a possible solution. In an ideal case the correct angle has prediction with close to 99 % certainty while other angles are closer to 1 %. This information is then used to determine the current heading angle.

Second principle also uses DL together with machine vision. Training is done exactly like in the image recognition example and only the difference is that CNN is used to calculate regression instead of image recognition. Main difference is that the regression system has single output compared to multiple outputs. After training, the new input is predicted and the result is given as a floating value. The floating value is between 0 and 360 and it directly tells the result of the angle. Figure 6 is a flowchart which shows workflow for machine vision and machine learning principle.

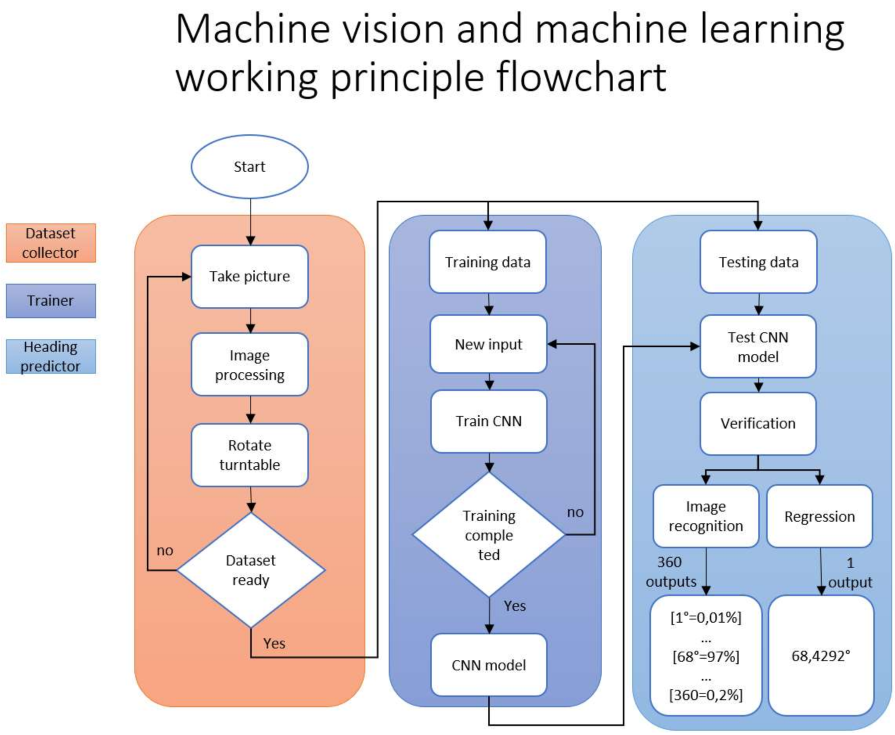

Figure 6. Machine vision and machine learning workflow.

Deep learning algorithms are a complicated and time-consuming subject. Building and training a working algorithm from scratch takes a lot of time and requires good knowledge about the subject. Many ready-to-go solutions exist for different kind of applications but those so called pretrained networks usually solve only the problem they were trained for. Those kinds of networks lack the necessary flexibility. That is why for the purpose of this testing, transfer learning was applied. Transfer learning means using a pretrained network as a base layer which is then modified by training the network with a new dataset. It saves time because the user doesn't have to make the code and neural network parameters fine tuning. The main downside compared to a fully fine-tuned NN is that the most optimal solution is not most likely found.

Angle conversion to azimuths

After receiving heading from either of the methods, the heading information must be converted to global azimuths if it is to be used in automatic flying. An alternative method would be to modify currently existing navigation controllers. An illustrative image for the convert process is in Figure 7. First, an indoor position system using anchors or similar locators is setup, so that the positioning area formed in the middle of the locators. Because a drone´s heading is dependent on the accurate location data, the positioning area is also the available flying area for the drone. Usually, anchors using the system consist of 3 to 8 anchors. One of the anchors is determined as a reference anchor and its position in the local coordinate system will be $(X_0,Y_0) = (0,0)$. Then, line $\vec{P}_{A1A2}$ is drawn in y-direction starting from (X_0,Y_0) . (X_0,Y_0) position is measured and defined with GPS coordinates latitude and longitude. Lastly, the angle θ_A between $\vec{P}_{A1A2}$ and magnetic north is calculated. With θ_A and (X_0,Y_0), local position and heading of the drone can be defined as global GPS location and azimuth. This method has the advantage that the same location data can also be used to determine a drone´s position.

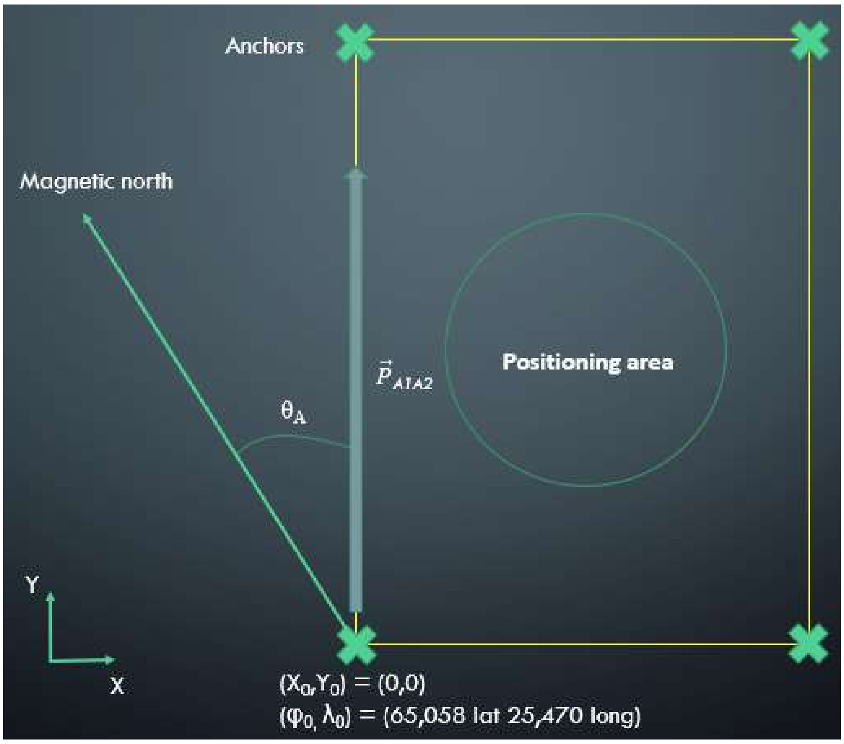

Figure 7. Positioning area from anchors is linked to global coordinates.

4.2 Testing plans

For drone navigation there are four different and important actions that need to be tested. Those are hover, rotation, forward moving along an axis and height changing movement meaning up and down movements. In simplified terms every automatic drone navigation can be done with these four different manoeuvres. And for that reason, it should be enough from the simulation perspective. Drone flying is usually a combination of different actions for example moving forward while also moving up. Different manoeuvres are shown in Figure 8. Below are more specific plans for each manoeuvre.

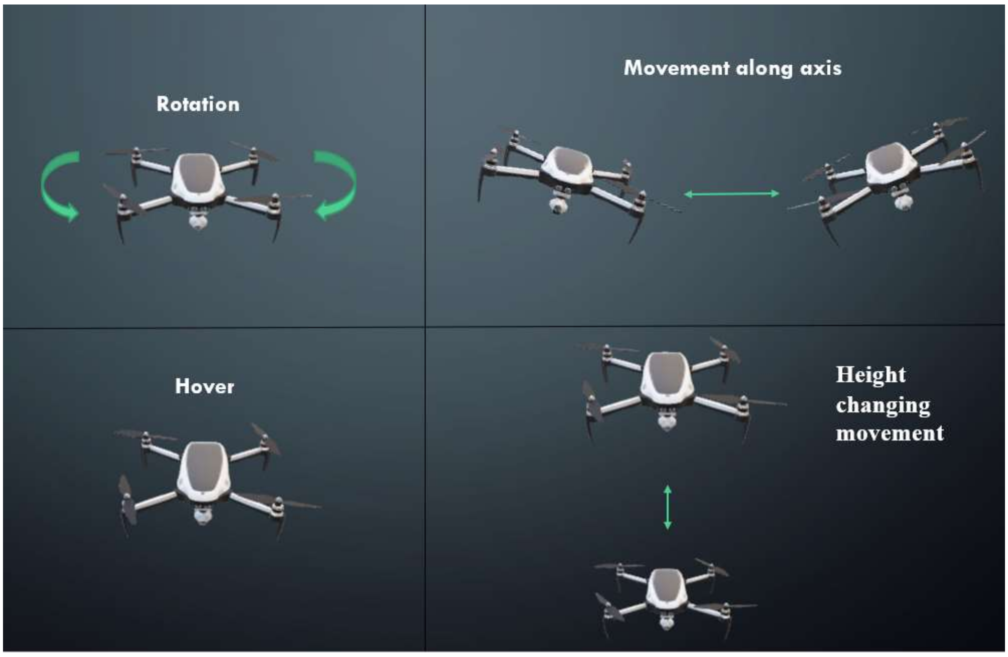

Figure 8. Four different manoeuvres in drone navigation.

Hover means that a drone stays in one place and holds it position. This is the most stable manoeuvre and because of that it should give out most exact heading. This can be considered as a foundation test which shows how good the system can be. To measure that, a reference system is necessary. For both heading calculation methods, ET250-3D turntable is used to determine exact angles and hold the angle for a certain period. Hover should stay in place for 30–60 s and it should be tested with multiple angles.

After the hover, the test shows how good the system can be in the best case and the rotation test is a natural continuation. The purpose of the rotation test is to find out how well calculated heading keeps up with the true angle while moving. Similarly, as in hover testing, ET250-3D turntable will be used for the reference angle. The turntable has only one speed option so different velocities can´t be easily compared. Comparing different speeds would be interesting because every heading calculating algorithm has latency meaning that at some speed the heading can´t keep up with the rotation. When the number of errors in heading calculation exceed a certain threshold, it is deemed as invalid. Finding out those limits could be considered as an upper limit to how quickly a drone can rotate without losing reliability in heading calculation. A simple possible solution to test different speeds a with turntable is to place tags to different distances from the centre.

Pozyx system has an option to calculate altitude but height changing movement doesn't have any reference system. In addition, the final setup doesn't have an option for 3D positioning. For that reason, height changing movement tests are done only in pre-tests and the results will be roughly vague. Nevertheless, the test is important to show how reliable a system is when a drone elevates. Big changes in elevation are typical for drones and heading calculation must be able to cope with that.

In addition to rotation, it is important to test movement along axis. Because IMU can provide momentary heading data, the reference system must stay the same as closely as possible while moving. This movement is not only limited to forward moving because a drone can move in four directions in addition of just forward. As with the height changing movement, movement along axis doesn't have a reference system. The machine vision method was left out for both elevation and movement along axis due to its complexity. Basically, teaching the deep learning algorithm works similarly as in rotation and hover movements but obtaining the photos and testing the results is harder without a reference system. In the future this could be done with a drone which flies indoors.

All of tests should be automatic when possible. Automatic testing means that tests are done with software and human assistance is minimum. Automatic testing has many benefits, for example better repeatability and comparability because every test is identic with each other compared to testing with human interactions. Often, automation also makes testing easier which means that more tests can be conducted in the same period.

4.3 Test equipment and environment

Equipment's

The selected indoors position system was Pozyx Enterprise system. It is a UWB based system which consists of tags and anchors. The system can reach up to 10 cm accuracy with up to 100 hz update rate for a single tag according the provider. Both accurate location information and high update rate are needed for the drone navigation. The system supports two different working principles, TDOA (Time Difference of Arrival) and TWR (Two Way Ranging). TDOA means that a single tag sends out an unscheduled message with the chosen update rate. This message is then collected in all the anchors. Because the distance between the tag and different anchors differs, the time when each anchor receives the message also differs. These timestamps are then used to calculate the tag´s position using trilateration. In the TWR method three messages are sent between one tag and each anchor. First, the system must know which tags it needs to listen. Then, a tag starts the positioning by sending the first message to the anchor, the anchor sends a message back to the tag and finally the tag returns the final message to the anchor. Calculating how long it took to receive this final message is used to determine the location. At least three communications are needed, and an ideal position is achieved with four communications. In the Enterprise edition, all the anchors are connected to each other with ethernet cables and these anchors are in the same way also connected to a local processing server called gateway. (Pozyx NV 2020a)

To obtain precise heading information for both test cases, a turntable was used. ET250-3D as seen in Figure 9 is a turntable which can be controlled remotely using TCP/IP network protocol and it has an accuracy of 0,5 °. (Outline 2020)

Figure 9. ET250-3D turntable.

ET250-3D is connected to the internet using ethernet cable. Measurement PC is connected to the same network and thus they can communicate together. Outline has a graphical user interface ET commander 2012 which enables to control ET250-3D wirelessly. This GUI is limited in a sense that it only shows the angle value to the user, but these values cannot be collected by any means. That is why, the control will be integrated into the same software with heading calculation codes. This enabled to control ET250-3D and obtain precise rotation angle value logs while simultaneously making measurements. The same turntable will be used for both multi-antenna and machine vision testing.

The last main equipment used for tests were simple testing platforms, as seen in Figure 10. Platforms´ purpose is like a fixture in manufacturing industry, it holds a position to support work. A multi-antenna platform´s idea is to keep a certain distance between positioning tags. It also has a power bank integrated into it which powers ups all the tags. Shape of the platform is like a quadcopter drone. Distance between tag and centre of the platform is 50 centimetres. Because the turntable doesn’t have speed options, velocity varieties must be obtained by changing tags distance from the centre. The further tag is from the centre the more velocity it has. In addition to the setup in Figure 10, another platform was used in multi-antenna testing. It is a 4-meter-long wood plank attached to the turntable. It was used to place tags approximately 2,8 m and 3,8 m away from the centre. This way, three different speeds can be potentially compared to each other.

For the machine vision platform, the idea is to hold components in place. The platform consists of camera, company computer, memory stick and battery. Each component is connected so that the platform doesn’t have any cables connected to the turntable. This way, the platform can rotate freely. Intel realsense d435i camera and 32 Gb memory stick are connected to the LattePanda computer with USB connection. Training dataset pictures are collected straight to the memory stick and thus the dataset is easy to export to the PC which handles more demanding training processing. All the components are powered up with 4S 14.8 V LiPo battery which is down converted to 11 V to suit the LattePanda.

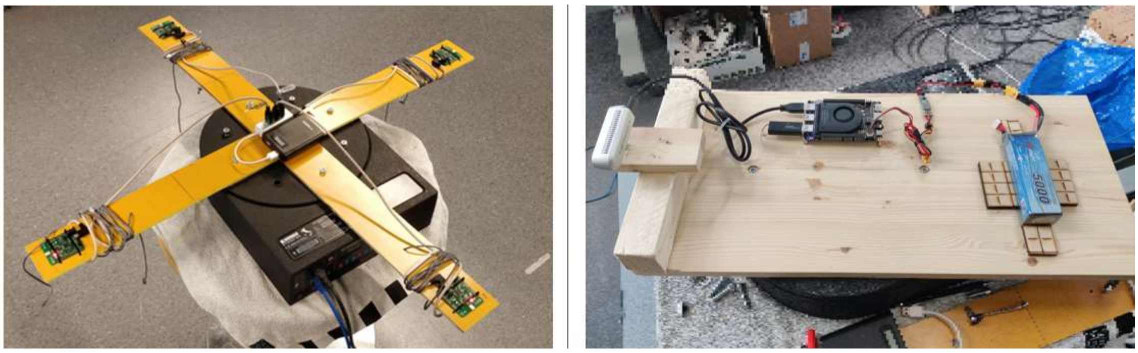

Figure 10. Testing platforms which will be used in the tests. Left platform is for multi-antenna method and right one is for machine vision method.

Test environments

Tests were conducted in two different premises. Figure 11 shows both test areas. Pre-tests were done in a smaller 9 m x 5 m x 3 m office area. The Pozyx system´s anchors were integrated into ceiling grid by hanging them in different heights. The area was a temporary placement before the final placement, so attachments for Pozyx anchors were made with that in mind. Temporary attachment enabled for example testing multiple anchor configurations conveniently. Anchors position was calculated by hand with a laser rangefinder. Final tests were done in a bigger open area size of 13 m x 13 m x 6 m. Anchors were permanently attached to the ceiling and their accurate position was calculated using a laser scanner to obtain 1 mm accuracy. The anchors were placed at similar heights relative to each other around the positioning area. This provides most optimal 2D positioning but the lack of height difference means that 3D position can´t be done with Pozyx. But this is not a mandatory because the goal is to get altitude information from the drone with an altimeter.

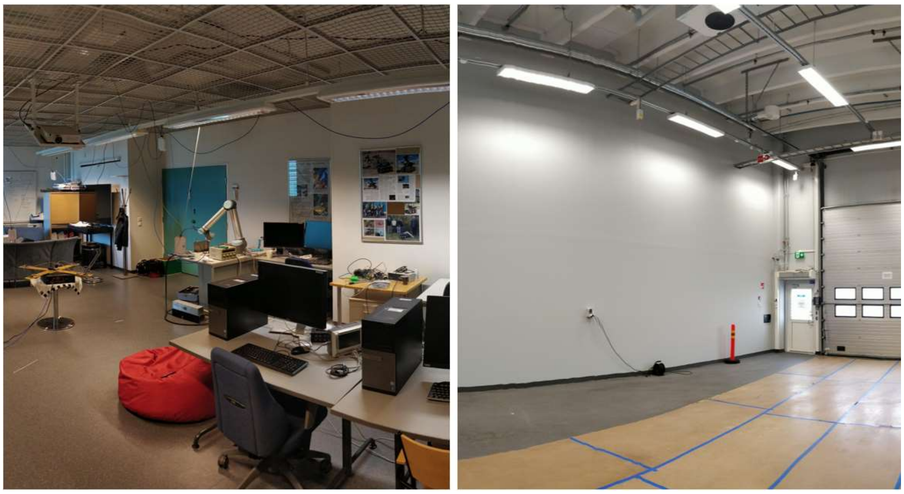

Figure 11. Test areas. Left is office area for pre-tests. Right picture is final placement and final testing area.

4.4 Method testing software

Software used to development the heading calculating methods are python based. Because the software is made with python it can be implemented for example to Arduino and Raspberry Pi with minimal modification. The final version has been thought to be used with a company computer and it only needs to provide a heading angle. Drone navigation can be handled with another computer. Both machine vision and multi-antenna method software are examined below. Two software share similarities because both are doing heading calculating and turntable control. The turntable is controlled with UDP protocol. First, the turntable is connected to the same network as the company computer with a lan cable. Then, UDP connection between the computer and turntable is opened and the computer can start sending request messages to the turntable. These messages can for example move the turntable to a certain direction or request the current heading angle. Using threading in python, the turntable control will run simultaneously with heading algorithm.

Multi-antenna software

Pozyx´s gateway communicates using MQTT protocol. Data is sent out whenever a tag sends out a message. Different tags are distinguished with personal tag IDs and these ID:s need to be changed if the used tags changes. Output data is JSON file consisting position, anchor data and optionally data from multiple sensors.

For each tag X and Y, coordinates are fetched from JSON file and used to calculate θ_{B1} and θ_{B2} with equation (3).

$$\theta_B = \arctan\left(\frac{\Delta y}{\Delta x}\right), \tag{3}$$

where θ_B is the angle between X-axis and a tag pair. Because there are 4 tags the θ_{B2} must be deducted by 90 degrees to be parallel with θ_{B1}. Then, θ_B can be calculated from the average of the two angles θ_{B1} and θ_{B2}. Then, θ_B angle is changed to azimuth with simple equation (4)

$$\alpha_B = 360\,° - \theta_B, \tag{4}$$

This average calculation has an error when heading is close to 0 degrees. This is because 2 different headings might vary between both side of 0 ° and 360 ° meaning that one heading could be for example 357 ° and other 1 ° Even though the difference is 4 degrees, the average from this example would be 179 ° This is taken into account in a way that if the difference between two headings is more than 300 degrees, then a bigger one is left out and used heading is a smaller one divided by two. This method can correct heading when a drone is rotating toward and beyond 0 angle, but it can possibly bias data when heading is constantly 0 degrees. This should be considered when heading is kept in 0 degree while doing the measurements.

Finally, the heading value is imported. In the final version data must be imported into a different computer. In the current version, it is enough that heading is visualised for the user interface and measurement data is imported after measurement. Measurement import is done by writing the data to a .csv comma separated value file and then it will be post processed for example in Excel.

One important factor to make this method reliable is that it needs to provide enough accurate position data. If the positions are changing too much, then heading calculation also changes and renders the result useless. That is why, data filtering is applied to the data to obtain most precise heading. Filtering can be applied in two areas: measurement points and calculated heading angles.

The system used in this thesis has 4 tags with each having 100 Hz update rate. This means that position data is updated roughly 400 times in one second. Because the position doesn't have to be updated so frequently, it can be averaged. Every measurement value is added to a buffer and when the buffer reaches a certain amount of values, it will calculate average and forward one averaged position (x,y) pair for each tag.

Before adding measurement points to the buffer, it can be filtered so that obvious outliers are filtered out. This can be done by comparing the current value with the previous value. If the change is bigger than a threshold value, then it is deemed as a wrong value and it will not be used in the heading calculation. Instead, a previous point will be used and send forward to heading calculation. A threshold value should be chosen so that it doesn't bias the data. Choosing too small a threshold value could filter out real values if a tag is moving and values are supposed to be changing between every data point.

Even though tags are moving, the distance between them will stay the same if the platform where tags are connected doesn't move. This fact can be used to filter out data points in the buffer. Certain upper and lower threshold values are chosen based on the distance between the tags and accuracy of the positioning system. If the two positions have a bigger distance between each other than the upper threshold, then both values are invalidated. Similarly, if the distance is smaller than lower a threshold, then those values are also invalidated. This same principle can be applied to all possible tag pairs between 4 different tags and not just the ones opposite of each other. Pozyx has integrated few position filters into the system which will be touch upon in pretests.

One more way to filter a calculated heading value is to apply a similar comparer to it as is done with the position values. A threshold angle value is determined and if the heading angle changes more than the threshold value, then it is invalidated, and the past value is used. This is not a preferred option in a sense that, at this point of filter, heading update rate is already low. Filtering it means that update rate will go down even more. That is why filtering should be done with position data because it has a much more higher update rate. If heading values are to be filtered, it should be done for example with Extended Kalman filter which could be used together with other sensors like gyro.

Machine vision software

Machine vision software consist of three parts: dataset collector, trainer and heading predictor.

Dataset collector controls camera and turntable. Working principle is that the camera takes a certain number of pictures and then the turntable moves to the next angle. With intel realsense D435i, pictures can be either taken with RGB mode or NIR mode. NIR mode means that the camera takes Near Infrared pictures, which operate in a range from 700 nm to 1400 nm. NIR removes color wavelengths but improves visibility, thus making it more suitable to long a range. (Infiniti Electro-Optics 2016) In raw mode, distance variations are marked in the pictures, as seen in Figure 12. Hence, the NIR mode gives out more data compared to the normal and it was used to collect the dataset instead of using any camera. Pictures are immediately converted to a smaller size to later fit training algorithm. Saving the pictures to the smaller size has also a benefit of using less memory. For image recognition, used size was 227 x 227 pixels and for regression 60 x 60 pixels. Pictures are collected to a separate USB driver. This way, pictures are easy to export to another PC which handles the training algorithm.

Figure 12. Two pictures taken with intel realsense D435i. Left one is a normal picture with RGB mode and right is raw data for the depth detection with NIR mode.

Training is done with a capable computer because training the algorithm is computationally heavy and thus takes time. Training algorithm is done using a pre-trained network. It means that all layers in the network are preselected and only the amount of inputs and outputs are changed. Inputs are the number of pictures for each training angle and output is the total amount of different trained angles. For image recognition with 1-degree interval the output would be 360 and with CNN regression output is 1. After successful training, the trainer gives out a fully connected network also called as a model. The model has parameters which are specially tuned with the dataset. This unique model is lastly used for the heading prediction.

The last part in the machine vision algorithm is the heading prediction. The basic idea is that the predictor takes new inputs for the previously trained model. Input can be obtained for example from new picture or frame of video clip. Then prediction using the model is made. Lastly, the predictor will give a heading angle as the output based on the calculation of the trained network. This output can be then used for drone navigation. Because the actual drone will not be flown in this thesis work, a predictor will not be implemented. Instead validation is used. Validation means that the dataset, which is not used for the training, is used to test how well the model works with new inputs. Validation dataset is usually done by splitting the dataset during training. The most important thing for the validation is that the dataset doesn’t include the same pictures used with training since that is the only way to know if the model really works universally or not.

5 HEADING CALCULATION TESTS

The purpose of the tests is to find out if the methods could work with an actual drone and if those could be used for reliable heading calculation. Tests described in this chapter were done in two phases, pre-test and final tests. Results from the pre-tests showed that both methods have limitations and those were considered in the final tests. Lastly, results and heading accuracy for each method from the final tests are presented in their own sections.

5.1 Pre-tests

Pre-tests were done in a smaller office area and the motivation was to test different things before doing the real tests in an area where configurating the system is harder. Especially indoor position Pozyx for the multi-antenna method must be optimised to provide best possible position data. Heading angle accuracy depends heavily on the position accuracy.

Indoor position system pre-tests

Before starting the heading measurements, it is important to test and optimize the indoor positioning system for a most accurate position. The first considered aspect was a position method. Pozyx Enterprise version has at the time of writing this thesis two available positioning methods, TWR and TDOA. Position methods were tested in three different anchors configurations, as seen in Figure 13. TDOA proved out to be better because of a few reasons. First, it has a higher update rate than TWR, but accuracy was the same for both methods, except in the second case with anchors being both near and far from the tags. In that scenario TWR was clearly more unstable because in TWR only 4 anchors can be utilised at the same time. Thus, position varies when used anchors bounce between closer and further ones. TDOA on the other hand uses all the available anchors, which makes it more reliable in that sense. In Pozyx enterprise edition, TDOA method has integrated filter options. These options include selectable modes of unpredictable and predictable movement and a choice for the freedom of movement. Mode is chosen based on how fast tags move compared to update rate. Freedom of movement strength on the other hand is chosen based on expected speed variances. Options used in tests were unpredictable movement and weak freedom of movement because the system has both low speed and speed variances. Testing showed that increasing freedom of movement increases position variety exponentially meaning that weak freedom of movement filters out obvious outliers.

Pozyx system can do 2D or 3D positioning depending on the height placements of the anchors. If there is a height difference like in the cube configuration, then 3D positioning is possible. Testing showed that position accuracy stays the same for 2D and 3D positioning when a tag was kept in the same height. Because height is irrelevant for this use case, final placement of the anchors in the bigger testing area is scattered around the testing area. This ensures that positioning has best possible coverage for 2D positioning in the biggest possible area. The last initial observation was that the office area was little bit too small for optimal positioning. The system needs more distance between anchors to provide the most optimal coverage. The observation was done using Pozyx enterprise internal tool HDOP. HDOP stands for Horizontal Dilution of Precision and it shows position quality based on the anchors´ position in a certain height. (Pozyx NV 2020b)

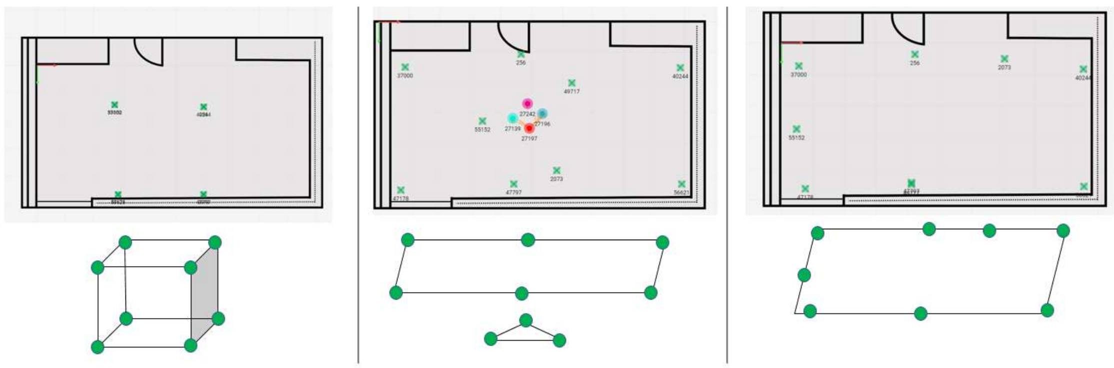

Figure 13. Three different anchor configurations. From left to right: Cube, three anchors close and anchors scattered around the room.

Multi-antenna pre-tests

In the beginning of the testing, the system needs to be calibrated. Purpose of the calibration is to find a correct 0 angle for the system. Calibration can only be done when the system has a reference angle. When calibration is possible, XY directions can be tentatively set visually along the axes of the room. But the starting direction can be basically anything because calibration can handle any starting angle. After choosing a starting angle, a single measurement with 45-degree intervals is performed.

Calibration process is the following: First, the system rotates 45 degrees for about 22 seconds and then waits for 50 seconds. After the first 22 seconds, when the system has reached 45 degrees, it will measure 40 heading values. All the values are collected in a list. After 40 values measuring stops and average of the list is calculated. Then, the system prints the average and adds it to another list and waits until 50 seconds has been reached. After 50 seconds, the system rotates 45 degrees and repeats the process. This continues as a loop until 360 degrees is achieved. Then, all the averages will be examined, and the system will print out the following information: median, average, minimum and maximum average. Then, the user can change 0 angle based on the median and average. Usually, those values are similar but sometimes the system has big outliers. Those outliers can be detected from minimum and maximum values. In those cases, median should be used instead of average.

Table 2 shows example calibration where the starting angle was purposely about 90 degrees offset. After a successful test, the 0 degree of the ET250-3D is changed based on the total correction from the measurement. This configuration is then used for all the rest of the tests.

Table 2. Calibration measurement example. Starting angle was visually 90-degree offset.

Measurements:

Calibration round	1st	2nd	3rd	True angle (°)
Average correction (°)	81,65	4	2	45
	84,85	0,2	0,56	90
	79,05	1,05	2,56	135
	79,98	4,85	2,15	180
	82,93	3,8	2,37	225
	80,85	1,1	0,15	270
	269,34	2,27	1,24	315
	79,85	5,46	348,83	360

Results:

Calibration round	1st	2nd	3rd
Average (°)	104,81	2,84	44,98
Median (°)	81,25	3,04	2,07
Minimum (°)	79,05	0,2	0,15
Maximum (°)	269,34	5,46	348,83
Rotation correction (°)	81	3	-

After calibration, hover pre-test was planned to be conducted. But during the testing, results showed that a heading value varies depending on the chosen angle. For that reason, instead of hover testing first, pre-test was a position accuracy test. Because the tags lack a reference position, other ways to determine a true position had to be done. The position for centre of the turntable was calculated similarly to the anchors position using a handheld laser rangefinder. After calculating the centre point, the perimeter of the circle was calculated knowing that tags are 50 cm away from the centre. Then, the position test was done using the turntable. One tag rotated to 8 different positions on the perimeter with 45 degrees intervals. Lastly, the distance between tag position and point at 45 degrees on the perimeter is calculated. Test results for that position test can be seen in Figure 14. Results show that a position varies to multiple directions. Further testing brought to attention a problem in the positioning system. The position had similar varieties in the same type of measurements. Meaning that precision is good, but the accuracy is not perfect. This kind of error is hard to filter out because errors will have different varieties in other positions.

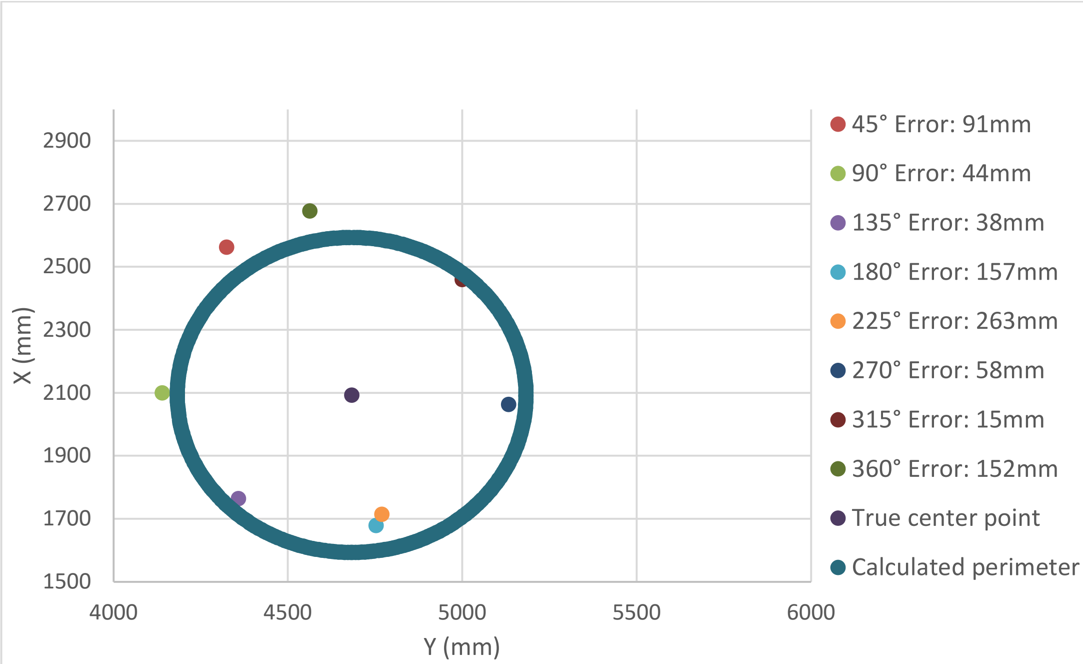

Figure 14. Position measurement results with a single tag. Circle around middle point indicates target assumed values and other points are measurement averages.

The same error occurring in position pre-test can be seen better in rotation pre-test in Figure 15. Two rotation measurements were done with the same calibration and setup except that the second measurement had an update rate cut in half. The turntable was rotated 180 degrees while calculating a reference. Results show that heading changes similarly and not randomly. These repeatable errors are the highest error spikes in the heading calculation.

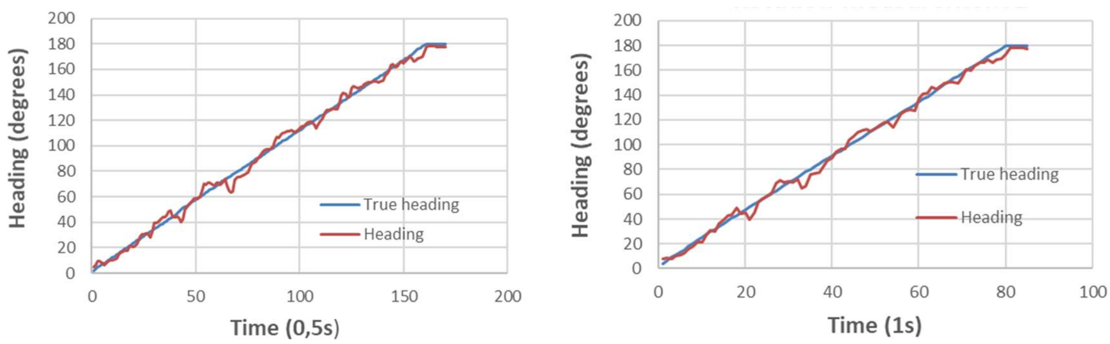

Figure 15. Two different rotation measurements with almost identical results.

Similar errors have been also detected in other studies. Masiero et al. (2017) reported in their detailed analysis that a systematic error indeed exists. According to their studies, the systematic error varies significantly with respect to both distance and relative orientation. This error will be taken into account when conducting final tests in the bigger testing area.

Height changing movement and movement along the axis were also briefly tested in the pretest phase. Neither of the measurements did have a reference system and measurements were done by the user. In the height changing movement measurement, the platform is lifted with two strings, one of them is connected to the end of each wing and the other one to the centre of the platform. The user started the measurement and then lifted the platform. When a certain height was visually achieved the platform was held there for approximately three seconds. After that, the platform was slowly returned to the starting position, kept there approximately three seconds. Finally, the same loop was repeated and after that the platform was kept in the starting position. This measurement practice was done five times. Movement along the axis measurement was also started and then manually moved by the user. The platform was moved to the other side of the room slowly and held at the end of the cable for about 10 seconds. Then, the platform was returned to the starting position and measurement ended.

Figure 16 shows the results from the height changing measurement. The upper value is the height calculated from the average of all the tags height values and the lower value is a corresponding heading at the same moment of time. Results show that height doesn't affect the heading value. Target heading was 270 degrees and the value changes up to 10 degrees in a positive or negative direction. Lastly Figure 17 shows results from the movement along axis measurement. Without filtering, heading changes momentarily more than 15 degrees but filtering efficiently takes out the outliers and maximum change is about 5 degrees in a positive or negative direction.

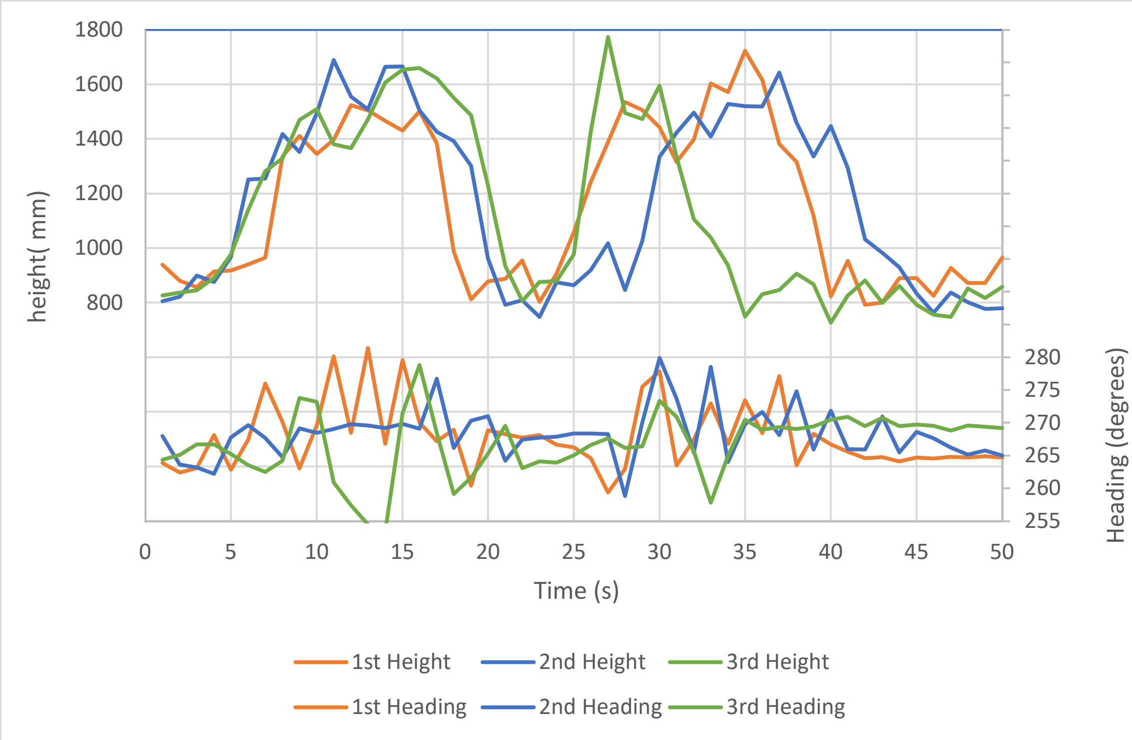

Figure 16. Height changing measurement. Upper values are heights and lower values with matching colours indicate changing in heading angle.

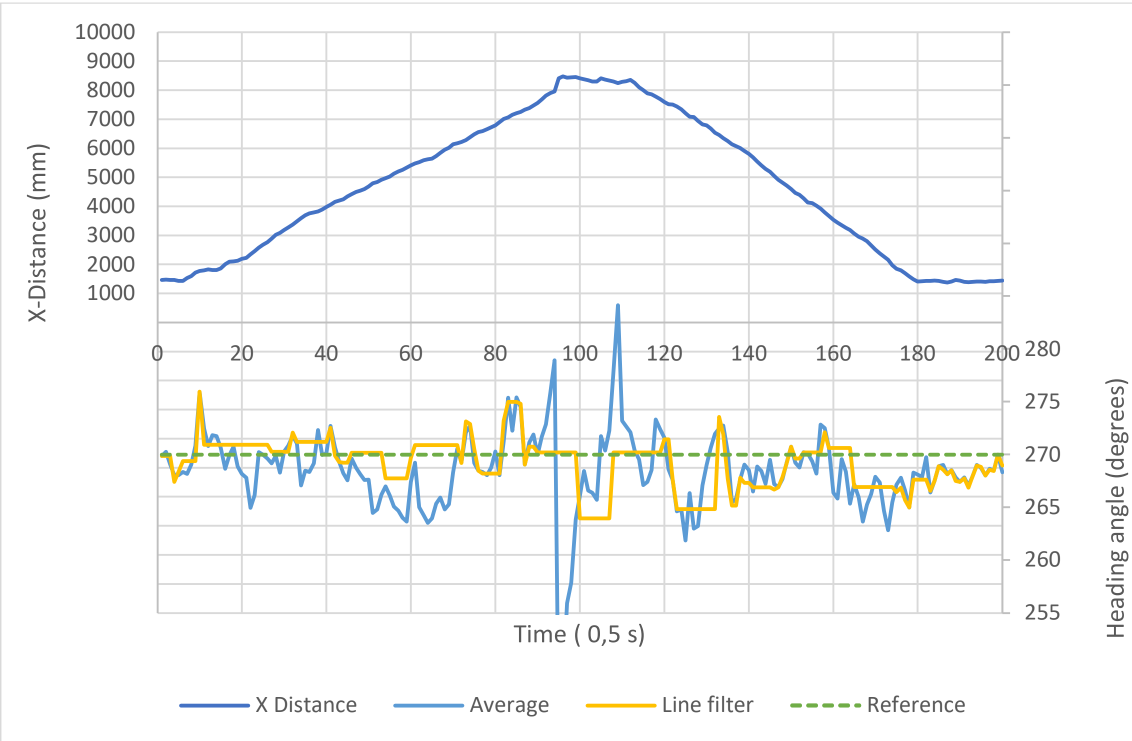

Figure 17. Movement along axis measurement results. Line filter leaves out heading values with position pairs, including invalid values. Invalid values are either too close or too far from each other.

Machine vision pre-tests

During the pretest phase, only the image recognition algorithm was tested. Testing started by collecting the dataset. Collecting was done by combining D435i depth camera with turntable ET250-3D together. The system was controlled with SW that can take pictures with about 30 Hz update rate and rotate after taking a certain number of pictures. The dataset should have a lot of pictures and the raw data reserve a lot of memory space. That is why images where resized to the ready-to-use state for the deep learning algorithm as the needed size is smaller than the original size. Needed size is 227 x 227 pixels. SW starts taking pictures from angle 0 degrees. It takes 2,000 pictures and then moves forward 1 degree. This loop is automatically repeated until 360 degrees are completed. During the measurement the pictures are written to the specific folders according the angles. The result is 360 separate folders with 2,000 pictures each. Each folder is named with a corresponding angle value. This way, the dataset is easy to import to the training software.

Training was done with a pretrained AlexNet and process of development followed guidelines from Mathworks online course Deep Learning Onramp (matlabacademy.mathworks.com 2020) Results can be seen in Figure 18. Training was done with 1,500 samples for each angle and rest of the 500 pictures were used for validation. The outcome was an algorithm that learned the dataset too well. Validation had 100,00 % success rate. It means that the algorithm learned the dataset perfectly, but it can´t be used in any other occasion. If the system is moved for example 15 cm to left, it doesn’t work anymore. This was most likely due to the dataset being too similar. Next time, training data should be taken so that turntable location is moved to different positions in a small area. Then, different angles will have variation and the algorithm will be more adaptive.

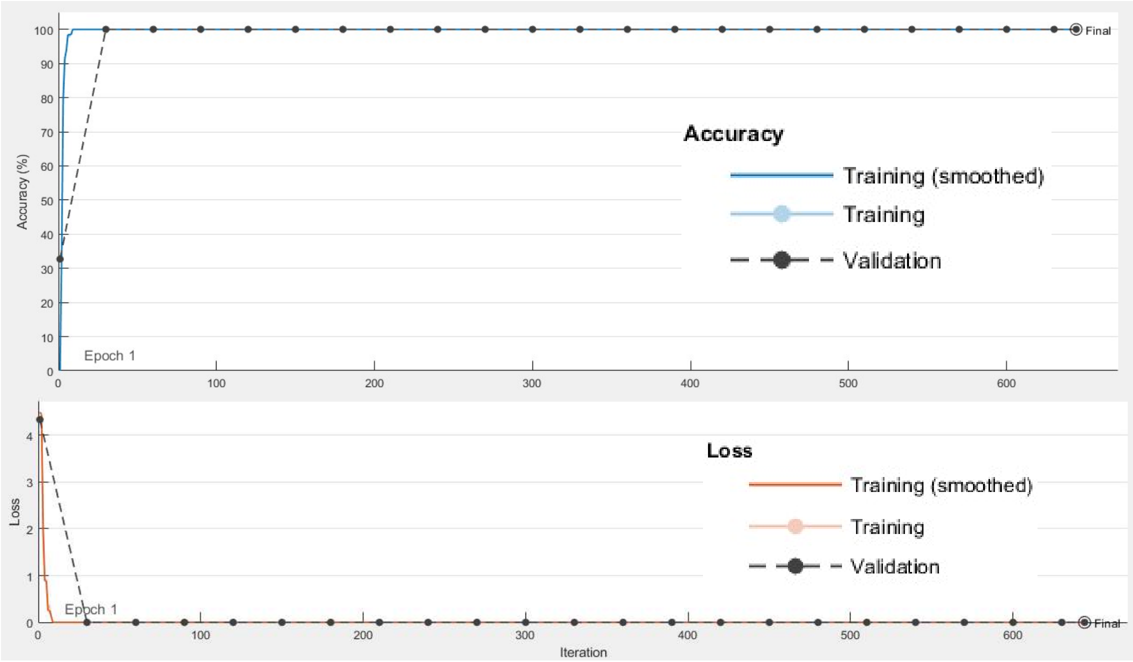

Figure 18. Results from the pre-test machine vision heading calculation. Training was done with 1,500 samples for each angle and 500 for validation. Validation success was 100,00 % meaning that the dataset was learned completely but the algorithm is not adaptive at all.

5.2 Multi-antenna testing

After pre-tests, Pozyx indoor positioning system was integrated into a bigger testing area. Position accuracy is presumably better compared to results from the pre-test office area. This is because a bigger distance between the anchors means better coverage. Another improvement was that anchors´ position was also calculated with 1mm accuracy. Used equipment was Z+F IMAGER 5016, 3D Laser Scanner. (zf-laser.com 2020) In addition to anchors position, a few other positions in the testing environment were measured. These positions can be later used as reference points in the measurements.

First Pozyx position accuracy was tested in the new environment. The turntable was manually placed so that the centre of the turntable was on an accurately measured position. The measured position has high accuracy but because the turntable was placed manually with visual inspection there are bound to an error of few centimetres in the accurate position. After the measurement equipment was ready the same 45 degrees interval position test as previously done in pre-tests was conducted. The only difference was that this time, the position was obtained from each of the tags instead of a single tag. Results are shown in Figure 19. Measurement was done three times in a row and the results suggest the same outcome as in the pre-tests. Position accuracy still differs to multiple directions, but all the measurements are almost identical meaning that precision is excellent. This naturally affects results in both rotation and hover tests.

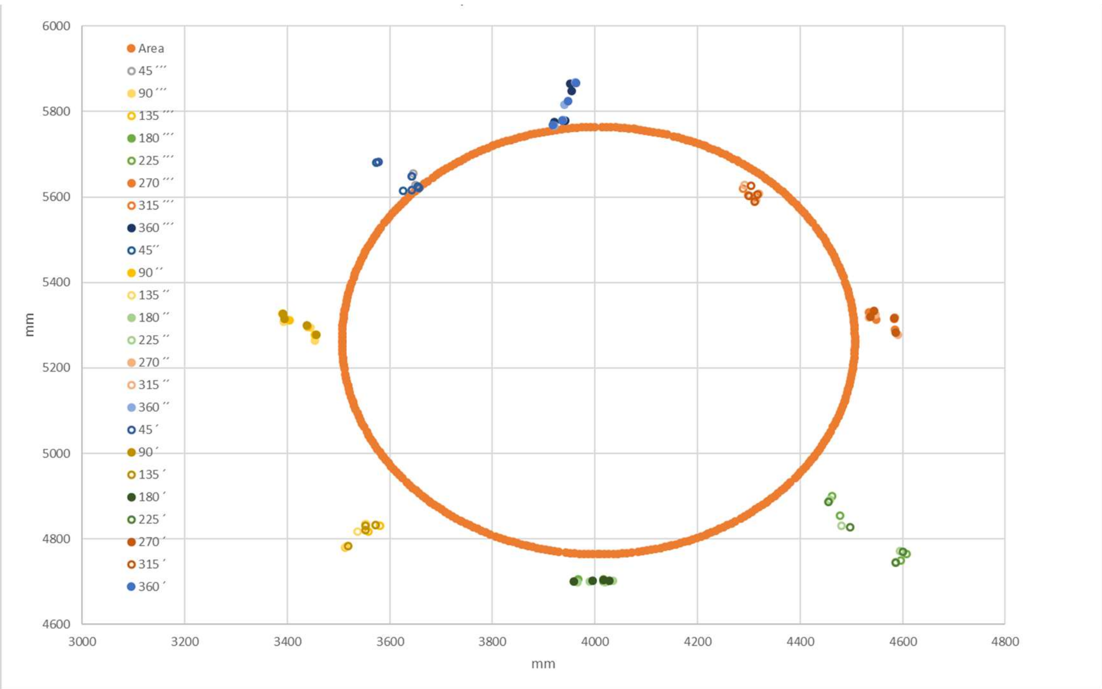

Figure 19. Position accuracy measurements in a new environment. Total of 3 measurements with 4 tags.

First hovering was tested in the same 45 degrees interval as the position measurement. Tags had 30 Hz update rate and total update rate was approximately 120 Hz. Heading was updated after 100 samples 40 times meaning that each angle was measured about 40 seconds. After 40 seconds, the average heading was calculated as listed in Table 3. Results in the table show that in an optimal condition heading can achieve 1-degree accuracy but at its worst it can overshoot about 10 degrees. Overshoot is most likely due to inaccurate heading calculation in 0 angle. Figure 20 demonstrates an example histogram to the distribution of the first 45 degrees measurement. Data is distributed with a small spread between 47–48 degrees.

Table 3. Results for three hovering tests.

Measurement:	1st		2nd		3rd	
True angle	Average	difference	Average	difference	Average	difference
45	47,03	2,03	47,23	2,23	47,22	2,22
90	92,3	2,3	92,34	2,34	92,25	2,25
135	136,2	1,2	136,17	1,17	136,21	1,21
180	183,34	3,34	183,35	3,35	183,4	3,4
225	226,32	1,32	226,29	1,29	226,35	1,35
270	269,22	0,78	269,68	0,32	269,39	0,61
315	317,19	2,19	317,17	2,17	317,16	2,16
360	359,66	0,34	350,82	9,18	350,84	9,16

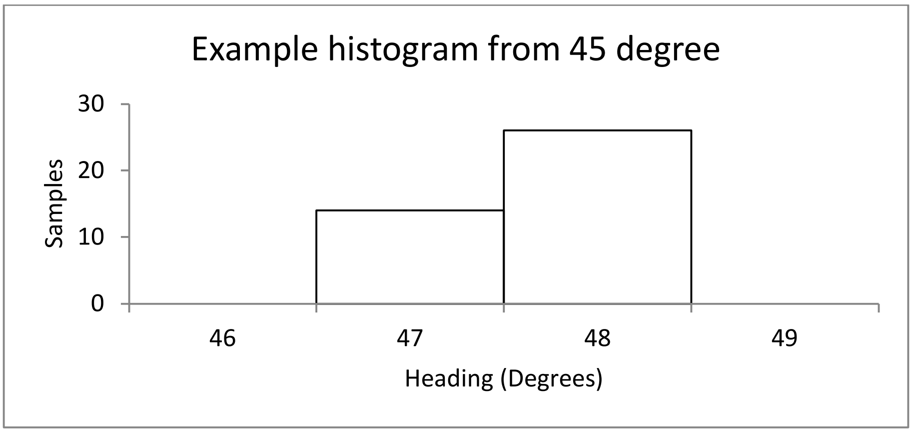

Figure 20. Example histogram from 1st 45 degree measurement. The scatter is visually small and all of the values are in 47–48 degrees range.

After hover the heading was also tested in rotation. Then, the same 360 degrees rotation test was done with a multi-antenna method and comparison results are seen in Figure 21. Results include tag pairs A1 and B1 and their average difference compared to the reference angle. As seen from the results, the average heading difference is low, but it has some local biases where it overshoots. These overshoots come from either one tag pair overshooting a lot or both pairs overshooting at the same time. Same measurement was repeated 5 times with similar results, so the reliability is quite good.

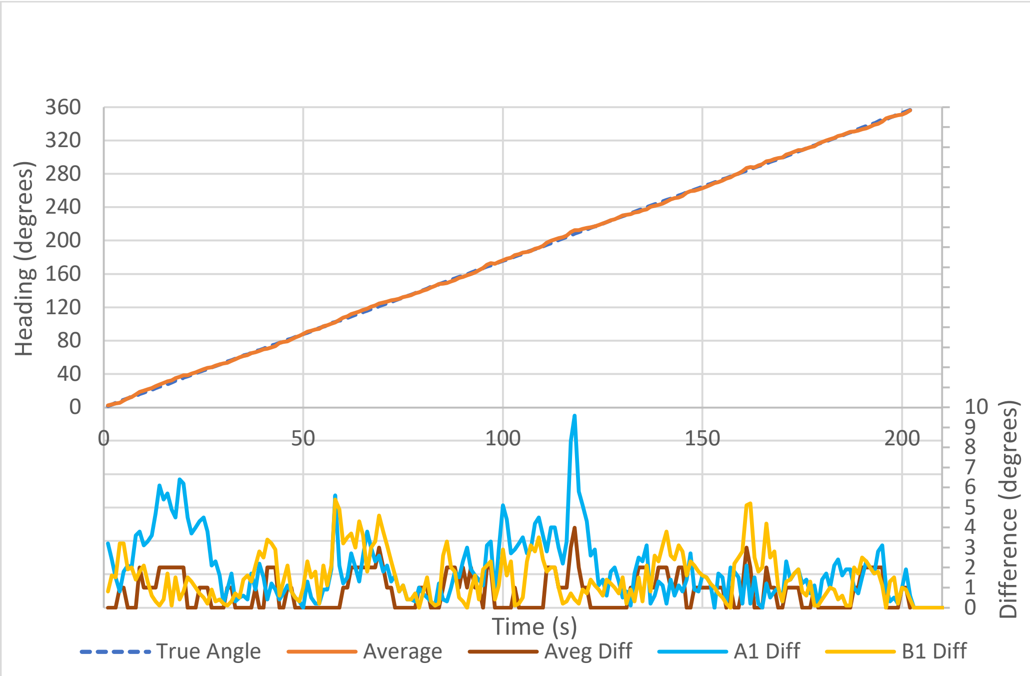

Figure 4. Rotation heading measurement results.

The final measurement for the multi-antenna method were done with a longer platform. Because the turntable doesn't have speed options, using a longer distance from the centre of the turntable was the only available method to test different speeds for tags. The circumference of the circle is calculated as in equation (5)

$$s = 2\pi r, \tag{5}$$

where r is the radius of the circle. The period for one rotation took 164 seconds and we mark is as T. Then, the speed of the object travelling the circle is shown in equation (6)

$$v = \frac{s}{T} = \frac{2\pi r}{T}, \tag{6}$$

The tags were placed to the platform at radius of 2,8 m and 3,8 m, so the respective speeds are $V_1 = 0{,}11$ m/s and $V_2 = 0{,}15$ m/s. The speeds are not high compared to average drone flying speed but it is quite close to optimal for the measurement flight speed indoors. The first position measurement was conducted and position was calculated with about one second interval. Position measurement results in Figure 22 show that while speed is increased, the tags can still keep up with the system. 2,8 m radius tag had minimum distance of 71,4 mm and maximum of 310,6 mm. The average difference was 174,7 mm and 88,1 % of the positions were spread between 150–250 mm. For the 3,8 m radius, tag minimum distance was 128,5 mm and maximum was 417,4 mm. Meaning that the accuracy was clearly worse. Average was 243,9 mm and position differences were spread between 200–300 mm with 82,4 % of the whole measurement. Measurement results are based on calculating the position by hand using a rangefinder, so they have some uncertainty. Most difference is seen in the close to 0 values in X-axis, which means that tags are close to the wall. Being close to the wall could be a possible solution difference in position.

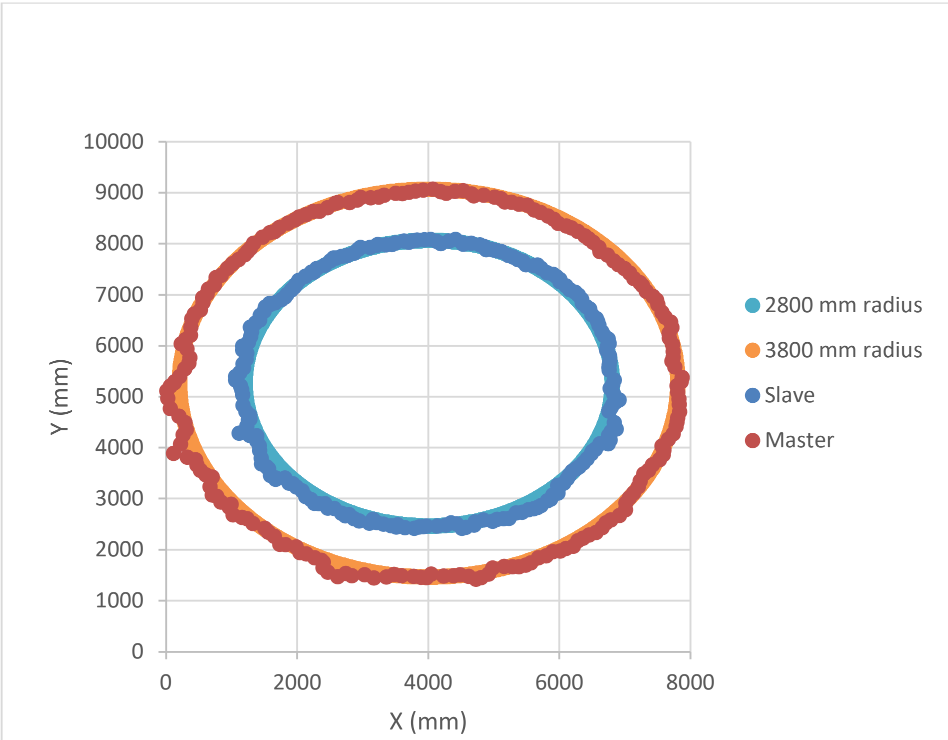

Figure 5. Position measurement with long radius setup.

Heading was also calculated with a longer radius setup but only two tags were used. Meaning that compared to other measurements, heading was obtained from a single pair of tags. Results in Figure 23 show a similar but slightly higher difference in heading accuracy compared to other measurements. Maximum heading was 13,7 degrees, heading varied between 1–5 degrees with 55,5 % and between 1–8 degrees with 87 %. Results are understandable because having only two tags compared to 4 tags decreases accuracy. Because systems are not identical in that sense, results are not directly proportional, but it can be stated based on two different measurements that increasing speeds decreases heading accuracy.

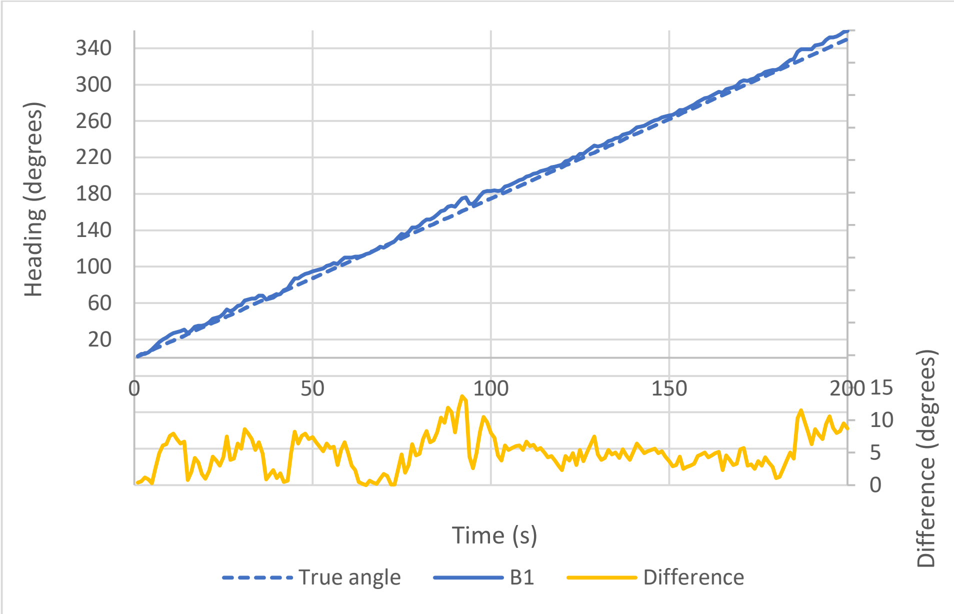

Figure 6. Rotation heading measurement with long radius setup.

Based on all the measurements, it can be concluded that an accurate heading angle, even up to 1-degree accuracy, is possible to obtain using only with the multi-antenna method. But the system has local biases which are hard to filter out. It means that heading information is not always reliable and thus it should not be used as the only heading calculation method. This is due to the systematic error in the positiong system. That is why, a multi-antenna method should be used instead, together with other heading calculation ways to obtain always reliable heading.

5.3 Heading calculation tests with machine vision

Machine vision testing initially continued where it was left off in pre-test phase. Based on the results from the pre-tests, the camera must be moved to multiple positions for one training. This way, the algorithm will be more robust to movement. In the new environment testing was done with the same platform setup as in the pre-tests. First testing was 90 degrees measurement with multiple positions in 50 cm radius. Results were the same as in the pre-test with 100 % validation accuracy. Second attempt was done by placing the turntable in 4 different positions in a 2 square meter area. Results from this training can be seen in Figure 24. Rotation was done with a 5 degrees interval up to 90 degrees. 100 pictures were taken for each position resulting in a total of 400 pictures for each angle. As seen from the results, the validation accuracy was roughly 90 %. But further testing and validation with data that had pictures in the middle of the area had close to 0 percent success meaning that system is not stable.

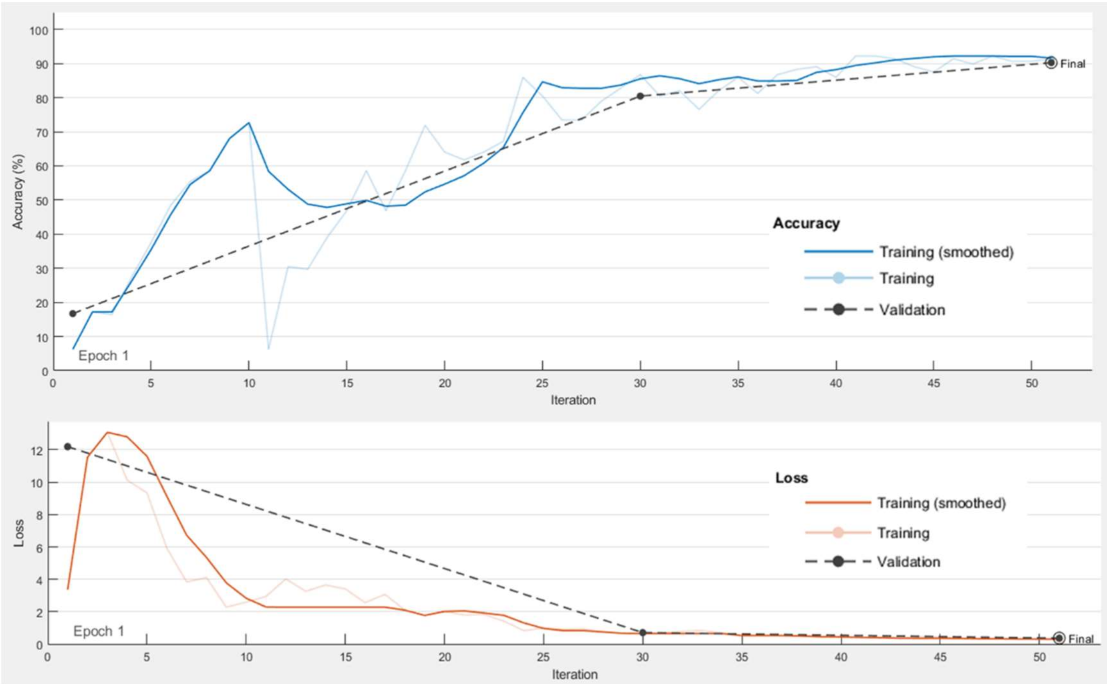

Figure 24. Results from 90 degrees rotation measurement with 5 degrees interval and 4 corner positions in 2 square meter area.

During the testing phase another algorithm was also implemented and tested. As mentioned in more details in chapter 4, the other method is also based on using DL together with machine vision. But instead of image recognition, the neural network is used to calculate regression. The tested principle is based on Adrian Rosebrock´s tutorial where CNN is used to predict regression. (PyImageSearch 2019) The principle is based on using Keras with the TensorFlow backend.

The dataset for the neural network was obtained the same way using the machine platform as in the previous tests. Images were taken inside a small 50 cm radius so that data has little variance but the overall similar looking pictures. 100 pictures were taken a total of seven times for each angle resulting to 525 training images and 175 validation images for each angle. For the regression method, images were resized to 60 x 60 pixels. After resizing, datasets were scaled to be between 0 and 1 by dividing it with 360. This will lead to better training and convergence. After data is prepared, the Convolutional Neural Network is created and then the model is compiled. The network consists of two hidden layers, as shown in architecture Figure 25. Model uses mean absolute percentage error as loss, meaning that the algorithm seeks to minimize the absolute percentage difference between heading prediction and actual heading. Lastly, training is done based on the pre-trained network parameters and results are demonstrated with verification.

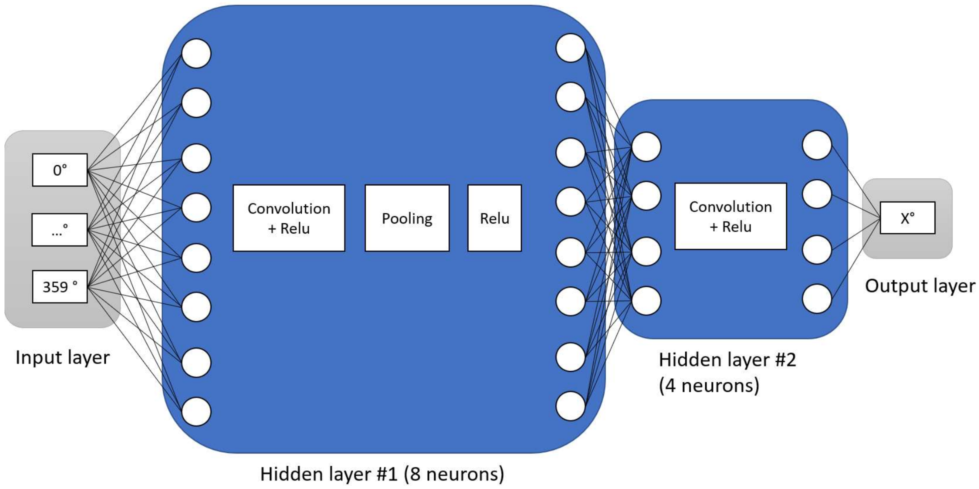

Figure 25. Heading CNN regression architecture diagram.

Table 4 shows results from the regression measurement verification. Results are clearly poor because even the best results are still more than 2,000 % off, even though the number of angles was kept at 90. Having more angles would make the system more complex and would presumably further worsen the results. Due to using a pre-trained network with a ready-made architecture, testing of different parameters was limited. The only variable that had a positive effect on the results was epoch which means how many times dataset is passed through a neural network. For example, changing parameters which affect learning rate such as batch size only made the results worse. The only viable solution to make the current setup work better would be to completely change the architecture of the Neural Network.

Table 4. Regression measurement validation results.

#	Epoch	Angles (°)	Interval (°)	training time (s)	Training samples	Validation samples	Validation loss (%)
1	3	90	5	168	12825	4275	6242,24
2	200	90	5	11200	12825	4275	4081,56
3	400	90	5	22 400	12825	4275	8443,12
4	800	90	5	44800	12825	4275	2303,15

Based on the results from both AI methods, it can be concluded that training data is too similar with each other. Taking a lot of pictures doesn't help if all of them have been taken from the same situation with the same variables. One way to increase diversity in the pictures is to use different outputs from the camera. This way multiple new pictures could be taken automatically. But most likely it doesn't fix the issue because the pictures would still share too much similarities. Instead of only changing one aspect like colour the whole system should be changed entirely. One example could be to use a drone with a camera to obtain the dataset. This way, a drone´s own vibration would add diversity to the different angles and on top of that pictures from different heights could be obtained more easily

Another problem with the system is also related to the dataset. The problem is that the dataset only included a few positions. If the system would be scaled more bigger so that it could work in the whole target area, then the number of pictures would drastically increase. In addition to multiple positions, drone flies in different heights which would even further increase complicity and worsen the results. More position and heights increase complicity and worsen the results.

6 CONCLUSION

In this thesis, drone indoor flying and heading calculation were studied. Different heading calculation methods were discussed and two methods for the indoor use case were proposed and tested. The work was done in collaboration with Nokia.

The first tested method was a multi-antenna method which showed potential results. With testing, it could be proven that heading accuracy can even reach up to 1-degree with good repeatability using only the multi-antenna method. But the system has local biases which make it not suitable to be used alone for automatic drone navigation. Heading should be always reliable and for that very reason biases that affect heading are not acceptable. The second method was based on using machine vision together with machine learning. The results from the verification were poor. Due to poor results in verification the method could not be used with the same testing setup as the multi-antenna method. AI was trained with two different methods but neither of them worked as intended. Results were either too precise with no adaptivity or too inaccurate to enable heading calculation.

Currently the multi-antenna method seems a more potential method, so the next development is to test it with the real drone. Using an actual drone, the testing can be done indoors by controlling the drone manually. Such testing will provide more comprehensive results for changing height measurements and enables more realistic testing in general. For example, using the drone, it is possible to test heading while moving forward and going up or rotating while moving up and down. The main concern in the used multi-antenna method was the local biases which were hard to filter out. For that reason, heading calculation should be refined with other sensor data. One testing worthy example is a flow sensor which has already been proven to work with drones. Using position data from a flow sensor, the multi-antenna method position could be filtered so that its more robust for short bias changes. Other way to use multi-antenna with a drone would be to replace a magnetometer and use it instead to compensate gyroscope drift. It is already proven that a multi-antenna method can give out accurate heading so instead of using it as main heading source, it could potentially help keep a gyro from drifting. In this example, local biases would still exist, so their effect could possibly ruin this method.

As for the machine vision part, the machine learning component should be left out. Learning from pictures is time consuming and is proven to be inefficient. The most troubling aspect is that to make drone heading calculation work universally in the environment, there needs to be pictures from many heights and position. For that kind of work, it would be convenient to use an automatically flying drone which could take pictures from all the possible heights and positions. But that kind of system needs already working indoor heading calculation meaning that it defeats the purpose. Instead of using machine learning, focus should be on the more traditional machine vision. Determination of orientation would be based on for example detecting corners of the room and calculating changes in those. This way, the system could be easily imported to other places also. A simpler solution could be to mark environment floor and use a downward looking camera to determine orientation with machine vision. This method has only the downside that it is dependent on the environment.

Based on the research in different heading calculations, the HTC Vive VR system will also be tested as a next development. VR technology is becoming more available and prices are bound to decrease while new VR equipment keep on coming to the market. There are already components on the market which developers can use to make custom objects for tracking. While writing this thesis Bitcraze has also released an early access product which uses HTC Vive base stations for pose and position calculation. While the Vive system has limitations with environment dependency, it offers promising accuracy and high update rate. It could work as a ground truth for other heading measurement testing’s if it doesn’t enable automatic navigation.

7 REFERENCE

ai.facebook.com, 2019. Powered by AI: Oculus Insight. [online] Available at: https://ai.facebook.com/blog/powered-by-ai-oculus-insight [Accessed 3 Apr. 2020].
ardupilot.org, 2020. Choosing a Ground Station — Copter documentation. [online] Available at: https://ardupilot.org/copter/docs/common-choosing-a-ground-station.html [Accessed 7 Jul. 2020].

Bills, C., Chen, J. and Saxena, A. (2011). Autonomous MAV flight in indoor environments using single image perspective cues. [online] Available at: http://www.cs.cornell.edu/~asaxena/MAV/saxena_MAV_perspectivecues_stairs.pdf. [Accessed 27 Apr. 2020].

Bitcraze Store. (2020). Lighthouse positioning deck. [online] Available at: https://store.bitcraze.io/collections/decks/products/lighthouse-positioning-deck [Accessed 15 Sep. 2020].

Cai, G., Chen, B.M. and Lee, T.H. (2011). Unmanned Rotorcraft Systems. [online] Google Books. Springer Science & Business Media. Available at: https://books.google.fi/books?hl=fi&lr=&id=RJYtO51KZ6kC&oi=fnd&pg=PR4&ots=UVvVU0-SWw&sig=sH5cKcvEGnbnaVDlLMxQf-aXBNM&redir_esc=y#v=onepage&q=latitude&f=false [Accessed 15 Jul. 2020].

Chahl, J. and Mizutani, A. (2012). Biomimetic Attitude and Orientation Sensors. IEEE Sensors Journal, [online] 12(2), pp.289–297. Available at: https://ieeexplore.ieee.org/document/5585652 [Accessed 9 Aug. 2020].

Chao, H., Coopmans, C., Di, L. and Chen, Y. (2010). A comparative evaluation of low-cost IMUs for unmanned autonomous systems. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/5604460/?arnumber=5604460 [Accessed 9 Aug. 2020].

Cherian, A., Andersh, J., Morellas, V., Papanikolopoulos, N. and Mettler, B. (2009). Autonomous altitude estimation of a UAV using a single onboard camera. [online] IEEE

Xplore. Available at: https://ieeexplore.ieee.org/document/5354307 [Accessed 6 Jul. 2020].

Cho, H.H., Kim, M.-I. and Kim, S.-H. (2012). A Study on PDR Heading Angle Error Correction Algorithm Using WLAN Based Localization Information. Lecture Notes in Electrical Engineering, pp.63–72.

Consortiq. (2020). A Short History of Unmanned Aerial Vehicles (UAV). [online] Available at: https://consortiq.com/short-history-unmanned-aerial-vehicles-uavs/.

Croon, Guido & De Wagter, Christophe. (2018). Autonomous flight of small drones in indoor environments. [online] Available at: https://www.researchgate.net/publication/328643432_Autonomous_flight_of_small_drones_in_indoor_environments [Accessed 13 Jul. 2020].

De Croon, G. and De Wagter, C. (2018). Challenges of Autonomous Flight in Indoor Environments. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/8593704 [Accessed 27 Apr. 2020].

El-Rabbany, A. (2002, 121 pp). Introduction to GPS : the Global Positioning System. Boston, Ma: Artech House. ISBN 1-58053-183-1

Espinosa, G. and Rubenstein, M. (2018). Towards mixed reality system with quadrotor : Autonomous drone positioning in real and virtual reality space. [online] Available at: https://robotics.northwestern.edu/documents/publications/Towards_mixed_reality_system_with_quadrotor_Autonomous_drone_positioning_in_real_and_virtual_reality_space_-_Final_.pdf [Accessed 15 Sep. 2020].

Essentials of Satellite Navigation Compendium. 2007. [online] Available at: http://zogg-jm.ch/Dateien/GPS_Compendium(GPS-X-02007).pdf. [Accessed 2 July. 2020].

Eynard, D., Demonceaux, C., Vasseur, P., Fremont, V., Demonceaux, C. and Frémont, V. (2010). UAV Altitude Estimation by Mixed Stereoscopic Vision. [online] Available at: https://hal.archives-ouvertes.fr/hal-00522124/document [Accessed 6 Jul. 2020].

Faa.gov. (2019). Federal Aviation Administration. [online] Available at: https://faa.custhelp.com/app/answers/detail/a_id/738/kw/indoor [Accessed 7 Sep. 2020].

FierceElectronics, 2009. Compensating for Tilt, Hard-Iron, and Soft-Iron Effects. [online] Available at: https://www.fierceelectronics.com/components/compensating-for-tilt-hard-iron-and-soft-iron-effects [Accessed 13 Jul. 2020].

Gade, K. (2016). The Seven Ways to Find Heading. Journal of Navigation, [online] 69(5), pp.955–970. Available at: https://www.navlab.net/Publications/The_Seven_Ways_to_Find_Heading.pdf [Accessed 15 Jul. 2020].

Gageik, Nils & Strohmeier, Michael & Montenegro, Sergio. (2013). An Autonomous UAV with an Optical Flow Sensor for Positioning and Navigation. International Journal of Advanced Robotic Systems. [online] Available at: https://www.researchgate.net/publication/276185024_An_Autonomous_UAV_with_an_Optical_Flow_Sensor_for_Positioning_and_Navigation [Accessed 8 Sep. 2020].

Hentschke, Matheus & Pignaton de Freitas, Edison & Hennig, Carlos & Caike, Igor. (2018). Evaluation of Altitude Sensors for a Crop Spraying Drone. [online] Available at: https://www.researchgate.net/publication/326759217_Evaluation_of_Altitude_Sensors_for_a_Crop_Spraying_Drone [Accessed 6 July. 2020].

HIROKAWA, R. and EBINUMA, T. (2009). A Low-Cost Tightly Coupled GPS/INS for Small UAVs Augmented with Multiple GPS Antennas. Navigation, 56(1), pp.35–44.

Infiniti Electro-Optics, 2016. NIR (Near Infrared) Imaging for Long-Range Surveillance Cameras. [online] Available at: https://www.infinitioptics.com/technology/nir-near-infrared [Accessed 1 Sep. 2020].

Jin, Y.-H., Ko, K.-W. and Lee, W.-H. (2018). An Indoor Location-Based Positioning System Using Stereo Vision with the Drone Camera. [online] Mobile Information Systems. Available at: https://www.hindawi.com/journals/misy/2018/5160543/ [Accessed 20 Apr. 2020].

Jung-Sup Um (2019, 33 – 35, 143 – 176 pp). Drones as Cyber-Physical Systems : Concepts and Applications for the Fourth Industrial Revolution. Singapore: Springer Singapore

Kashyap Vyas (2018). A Brief History of Drones: The Remote Controlled Unmanned Aerial Vehicles (UAVs) [online] Interestingengineering.com. Available at: https://interestingengineering.com/a-brief-history-of-drones-the-remote-controlled-unmanned-aerial-vehicles-uavs.

manualsdir.com. (1999). Leica Geosystems GPS Basics User Manual | 64 pages. [online] Available at: https://www.manualsdir.com/manuals/651926/leica-geosystems-gps-basics.html [Accessed 6 Sep. 2020].

Masiero, A.; Fissore, F.; Vettore, A. (2017). A Low Cost UWB Based Solution for Direct Georeferencing UAV Photogrammetry. Remote Sens, no. 5, pp. 414.

matlabacademy.mathworks.com. (2020). MATLAB Course. [online] Available at: https://matlabacademy.mathworks.com/R2019b/portal.html?course=deeplearning [Accessed 17 Jun. 2020].

Medium, W. (2019). Wing to launch early access air delivery program in Finland in June. [online] Medium. Available at: https://medium.com/wing-aviation/wing-to-launch-early-access-air-delivery-program-in-finland-in-june-13f8c1537ff2.

Nelson, R.C. (1998). Flight stability and automatic control. Boston, Mass.: Mcgraw Hill.

OptiTrack, 2020a. General FAQs. [online] Available at: https://optitrack.com/support/faq/general.html [Accessed 2 Apr. 2020].

OptiTrack, 2020b. Motion Capture for Robotics. [online] Available at: https://optitrack.com/motion-capture-robotics/ [Accessed 2 Apr. 2020].

Outline, (2020). ET 250-3D | Outline. [online] Available at: https://outline.it/outline-products/measurement-systems/et-250-3d/ [Accessed 21 May 2020].

Patrik, A., Utama, G., Gunawan, A.A.S., Chowanda, A., Suroso, J.S., Shofiyanti, R. and Budiharto, W. (2019). GNSS-based navigation systems of autonomous drone for delivering items. Journal of Big Data, 6(1).

Pfeifer, R., Blumberg, B., Meyer, J.-A. and Wilson, S.W. (1998). Modeling Ant Navigation with an Autonomous Agent. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/6278757 [Accessed 9 Aug. 2020].

pilotfriend.com, 2018. gyroscopic systems and instruments. [online] Available at: http://www.pilotfriend.com/training/flight_training/fxd_wing/gyro.htm [Accessed 3 Jul. 2020]. [Accessed 3 July. 2020].

Poza-Luján, J.-L., Posadas-Yagüe, J.-L., Cristóbal, A. and Rosa, M. (2018). Indoor Drones for the Creative Industries: Distinctive Features/Opportunities in Safety Navigation. Drones and the Creative Industry, pp.129–141.

Pozyx NV, 2020a. page_faq | Pozyx. [online] Available at: https://www.pozyx.io/faq [Accessed 20 Jul. 2020].

Pozyx NV, 2020b. layers | Pozyx. [online] Available at: https://docs.pozyx.io/enterprise/latest/visualization/layers [Accessed 1 Sep. 2020].

PyImageSearch. (2019). Keras, Regression, and CNNs. [online] Available at: https://www.pyimagesearch.com/2019/01/28/keras-regression-and-cnns/

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. Neural Networks, 61, pp.85–117.

Schmidt, G. (2011). INS/GPS Technology Trends. [online] Available at: https://pdfs.semanticscholar.org/eaa0/53cc86d68b7b8610c376d985169a30c4c0d7.pdf [Accessed 6 Apr. 2020].

Sporttechie.com, 2019. The Drone Racing League Turned to Aerodynamics Research and Motion Capture to Build a Lifelike Simulator. [online] Available at: https://www.sporttechie.com/drone-racing-league-aerodynamics-motion-capture-video-game/ [Accessed 2 Apr. 2020].

Steven M. LaValle, 2019. Virtual Reality, chapter 9. [Online] Available at: http://vr.cs.uiuc.edu/ [Accessed 2 Apr. 2020].

tech.fb.com, 2019. From the lab to the living room: The story behind Facebook's Oculus Insight technology and a new era of consumer VR. [online] Available at: https://tech.fb.com/the-story-behind-oculus-insight-technology/ [Accessed 2 Apr. 2020].

TechCrunch, 2019. A first look at Amazon's new delivery drone. [online] Available at: https://techcrunch.com/2019/06/05/a-first-look-at-amazons-new-delivery-drone/.

TerrisGPS. (2015). GNSS / GPS Technology Differences | TerrisGPS. [online] Available at: http://www.terrisgps.com/gnss-gps-differences-explained/ [Accessed 17 Jul. 2020].

Tiemann, J., Schweikowski, F. and Wietfeld, C. (2015). Design of an UWB indoor-positioning system for UAV navigation in GNSS-denied environments. 2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN). [online] Available at: https://www.kn.e-technik.tu-dortmund.de/.cni-bibliography/cnidoc/Tiemann2015.pdf [Accessed 14 Jul. 2020].

Tsouros, D.C., Bibi, S. and Sarigiannidis, P.G. (2019). A Review on UAV-Based Applications for Precision Agriculture. Information, [online] 10(11), p.349. Available at: https://www.mdpi.com/2078-2489/10/11/349 [Accessed 6 Apr. 2020].

uavnavigation.com, 2018. UAV Navigation in depth: Magnetometers, why are they critical for UAV navigation? | UAV Navigation. [online] Available at: https://www.uavnavigation.com/company/blog/uav-navigation-depth-magnetometers-why-are-they-critical-uav-navigation [Accessed 13 Jul. 2020].

Vetrella, A., Fasano, G., Accardo, D. and Moccia, A. (2016). Differential GNSS and Vision-Based Tracking to Improve Navigation Performance in Cooperative Multi-UAV Systems. Sensors, 16(12), pp .2164.

Wang, J.-H. and Gao, Y. (2005). A new magnetic compass calibration algorithm using neural networks. Measurement Science and Technology, 17(1), pp.153–160.

Wikipedia Contributors (2019). Accelerometer. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Accelerometer [Accessed 2 July. 2020].

Wikipedia. (2020). Dead reckoning. [online] Available at: https://en.wikipedia.org/wiki/Dead_reckoning [Accessed 21 Apr. 2020].

William, J. and K Munson (1977). Jane's Pocket Book of Remotely Piloted Vehicles: Robot Aircraft Today; Collier Books: New York, NY, USA

zf-laser.com. (n.d.). ZF-Laser - Z+F IMAGER® 5016, 3D Laser Scanner. [online] Available at: https://www.zf-laser.com/Z-F-IMAGER-R-5016.184.0.html?&L=1 [Accessed 11 Aug. 2020].