

# Development and integration of an autonomous UAV into an urban security system

Hendrik G. Brand

20653301

Dissertation submitted in fulfilment of the requirements for the degree Magister in Mechanical  
Engineering at the Potchefstroom campus of the North-West University

Supervisor: Professor L. Liebenberg

Co-supervisor: Professor E. H. Mathews

November 2011

# Abstract

---

The aim of this project is to develop an integrated security system making use of a UAV. The majority of unmanned aerial vehicles (UAVs) have been developed for military applications. The regulation for flying in the national airspace system (NAS) is being finalised which has shifted the focus to the development of UAVs for civilian applications. One of these applications is security systems. The problem, however, is that UAVs have not been effectively integrated with the human systems. A fixed-wing miniature (mini) UAV is developed for the purpose of this project.

Typical security systems, including South African systems, are analysed. Silver Lakes Golf Estate — that already has a well managed and effective ground security system — is the reference urban security site. The role of a mini UAV in this type of security system is determined. Consequently, it is possible to determine the requirements for full integration into this security system.

Investigation is done into the most suitable UAV for application in the security system. A UAV is subsequently developed to fulfil these requirements. The UAV is then compared to presently available UAVs in terms of cost and features. This gives the benchmark for the requirements of a successful UAV.

The UAV is tested to measure the extent to which it fulfils the requirements of a UAV. A flight-test procedure is developed to do the initial flight-testing. The requirements of the UAV flight are: stable flight, accurate waypoint tracking and height control. The requirements for integration into an urban security system are subsequently tested which include: testing the range, endurance, ground control station (GCS) and video feed. To ensure that these results are correct, the sensor data are validated. The sensors that were tested are the pressure sensor, global positioning system (GPS) sensor and airspeed sensor.

It is found that the UAV was capable of stable flight while accurately following waypoints and maintaining the preset height. The range of the telemetry systems and the endurance of the UAV are sufficient to monitor the reference urban security site. The UAV is able to stream live video and all the necessary information is displayed on the GCS.

Further research is required for analysing video feed software. The UAV also requires autonomous take-off and landing capabilities to simplify operation. The unmanned aircraft system (UAS) needs to be tested by implementing it in an urban environment. Finally, obstacle-avoidance capabilities can be incorporated for avoiding crashes and consequently increasing safety.

# Acknowledgements

---

I thank TEMM International (Pty) Ltd and MCI (Pty) Ltd who sponsored my studies. I would like to thank Professor Leon Liebenberg for his technical assistance. Your patience and insight are deeply appreciated. My deepest appreciation goes out to Professor Eddie Mathews for assisting by piloting the UAV at short notice and his guidance throughout the project. I would also like to thank the other pilots (Waldo Bornman and Ian Mathews), Dougie Velleman for his technical assistance and Jan Botha and his team at *Online Intelligence* for their insights into security systems. Finally, thanks go to my friends, family and loving girlfriend for their support in everything I do. Without everyone involved, this project would not have been possible.

# Table of contents

---

|                                                                |     |
|----------------------------------------------------------------|-----|
| Abstract .....                                                 | i   |
| Acknowledgements .....                                         | iii |
| Table of figures .....                                         | vi  |
| List of tables.....                                            | ix  |
| List of symbols.....                                           | x   |
| List of abbreviations .....                                    | xi  |
| Glossary .....                                                 | xii |
| 1. Introduction.....                                           | 1   |
| 1.1. Background.....                                           | 2   |
| 1.2. Purpose of study .....                                    | 3   |
| 1.3. Scope of study.....                                       | 3   |
| 2. State-of-the-art UAV technology.....                        | 6   |
| 2.1. Introduction.....                                         | 7   |
| 2.2. What is a UAV?.....                                       | 7   |
| 2.3. Classification .....                                      | 8   |
| 2.4. Applications.....                                         | 10  |
| 2.5. Regulation.....                                           | 11  |
| 2.6. Present UAVs suited for civilian applications.....        | 13  |
| 2.7. Sensors and control .....                                 | 16  |
| 2.8. Typical UAS layout .....                                  | 17  |
| 2.9. General urban security systems .....                      | 18  |
| 2.10. South African urban security system.....                 | 20  |
| 2.11. Alternative security considerations.....                 | 24  |
| 2.12. UAVs in security applications .....                      | 25  |
| 2.13. Integration requirements .....                           | 29  |
| 2.14. Determining the need for the development of the UAV..... | 31  |
| 2.15. Conclusion .....                                         | 32  |
| 3. Assembly and integration .....                              | 33  |
| 3.1. Introduction.....                                         | 34  |
| 3.2. Choosing the airframe.....                                | 34  |
| 3.3. Motor selection .....                                     | 35  |
| 3.4. Autopilot selection.....                                  | 38  |
| 3.5. Camera selection.....                                     | 46  |

|                                                                                                                     |     |
|---------------------------------------------------------------------------------------------------------------------|-----|
| 3.6. Telemetry equipment .....                                                                                      | 47  |
| 3.7. Battery selection .....                                                                                        | 49  |
| 3.8. Assembling the autopilot .....                                                                                 | 50  |
| 3.9. Sensor measurement .....                                                                                       | 56  |
| 3.10. Problems experienced .....                                                                                    | 58  |
| 3.11. Discussion of final product .....                                                                             | 60  |
| 3.12. Integration into an urban security system .....                                                               | 62  |
| 3.13. Conclusion .....                                                                                              | 66  |
| 4. Verification and validation .....                                                                                | 67  |
| 4.1. Introduction .....                                                                                             | 68  |
| 4.2. Flight-testing procedure .....                                                                                 | 68  |
| 4.3. Verification of fly-by-wire mode .....                                                                         | 71  |
| 4.4. Verification of GPS waypoint tracking .....                                                                    | 73  |
| 4.5. Verification of height maintenance .....                                                                       | 77  |
| 4.6. Verification of hand-launch capability .....                                                                   | 80  |
| 4.7. Verification of maximum height .....                                                                           | 81  |
| 4.8. GCS verification .....                                                                                         | 82  |
| 4.9. Flight range verification .....                                                                                | 85  |
| 4.10. Endurance verification .....                                                                                  | 87  |
| 4.11. Data validation .....                                                                                         | 87  |
| 4.12. Conclusion .....                                                                                              | 93  |
| 5. Conclusion .....                                                                                                 | 95  |
| 5.1. Review of study goals .....                                                                                    | 96  |
| 5.2. Addressing goals .....                                                                                         | 97  |
| 5.3. Project conclusions .....                                                                                      | 98  |
| 5.4. Recommendations for future study .....                                                                         | 99  |
| 6. References .....                                                                                                 | 100 |
| Appendix A - Project cost breakdown .....                                                                           | 107 |
| Appendix B – Raw data sample .....                                                                                  | 108 |
| Appendix C – Issues raised and proposed operational system received from <i>Online Intelligence (Pty) Ltd</i> ..... | 111 |

# Table of figures

---

|                                                                                                                                                        |    |
|--------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figure 1: AeroVironment RQ-11 Raven being hand-launched (Defense Industry Daily, 2011) .....                                                           | 8  |
| Figure 2: Mavionics' Capolo P50 (RUVSA, 2011) .....                                                                                                    | 13 |
| Figure 3: Cyber Aerospace's CyberBug Micro with GCS and other equipment (Cyber Aerospace, 2011) .....                                                  | 14 |
| Figure 4: ATE's Kiwit flying low (ATE-Group, 2010) .....                                                                                               | 15 |
| Figure 5: Three axis of rotation used to describe aircraft orientation (Bozkurt & Aslan, 2009) .....                                                   | 16 |
| Figure 6: Typical unmanned aircraft system layout (Adapted from Müller, 2008).....                                                                     | 18 |
| Figure 7: Typical urban security system layout (Dufrene, 2005) .....                                                                                   | 19 |
| Figure 8: A recorded video of a parking area (Shah <i>et al.</i> , 2007).....                                                                          | 21 |
| Figure 9: Biometric capabilities applied to Figure 8 (Shah <i>et al.</i> , 2007).....                                                                  | 21 |
| Figure 10: Silver Lakes Golf Estate map — the red oval denotes where flight testing was conducted (Silver Lakes, 2009) .....                           | 22 |
| Figure 11: Silver Lakes Golf Estate control room .....                                                                                                 | 23 |
| Figure 12: Cost of mission per flight hour per kilogram (Russel, 2007).....                                                                            | 26 |
| Figure 13: Mass of UAVs and CPAs (Clothier <i>et al.</i> , 2011) .....                                                                                 | 26 |
| Figure 14: Typical UAV mission cycle (Russel, 2007).....                                                                                               | 27 |
| Figure 15: CPA and UAV mishap rates per flight hours (adapted from Cambone <i>et al.</i> , 2005) .....                                                 | 28 |
| Figure 16: Multiplex Twinstar II (Multiplex, 2011).....                                                                                                | 35 |
| Figure 17: Pulso X2008/20 electric motor.....                                                                                                          | 38 |
| Figure 18: LEGO® MINDSTORMS® NXT 2.0 with included sensors and servos (Bozkurt & Aslan, 2009).....                                                     | 40 |
| Figure 19: MicroPilot MP1028 <sup>g</sup> autopilot (MicroPilot, 2011) .....                                                                           | 40 |
| Figure 20: Arduino ArduPilot Mega (Arduino, 2011).....                                                                                                 | 41 |
| Figure 21: PID position-controller schematic (adapted from Sinnathamby <i>et al.</i> , 2010).....                                                      | 43 |
| Figure 22: Effect of gain $K_p$ on the PID output with the other gains at 0.1, based on a step input (Adapted from Franklin <i>et al.</i> , 1986)..... | 44 |
| Figure 23: Effect of gain $K_i$ on the PID output with the other gains at 0.1, based on a step input (Adapted from Franklin <i>et al.</i> , 1986)..... | 44 |

|                                                                                                                                                        |    |
|--------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figure 24: Effect of gain $K_d$ on the PID output with the other gains at 0.1, based on a step input (Adapted from Franklin <i>et al.</i> , 1986)..... | 45 |
| Figure 25: KX191 camera (Rangevideo, 2011).....                                                                                                        | 46 |
| Figure 26: Camera mounted beneath the UAV.....                                                                                                         | 47 |
| Figure 27: Spektrum RC package .....                                                                                                                   | 48 |
| Figure 28: XBee modules before assembly .....                                                                                                          | 48 |
| Figure 29: Video transmitter (Rangevideo, 2011) .....                                                                                                  | 49 |
| Figure 30: Video receiver (Rangevideo, 2011).....                                                                                                      | 49 |
| Figure 31: XPower LiPo battery .....                                                                                                                   | 50 |
| Figure 32: Autopilot with headers (DIY Drones, 2011) .....                                                                                             | 50 |
| Figure 33: IMU shield with headers (DIY Drones, 2011).....                                                                                             | 51 |
| Figure 34: Fully assembled autopilot (DIY Drones, 2011) .....                                                                                          | 51 |
| Figure 35: Autopilot connections (DIY Drones, 2011) .....                                                                                              | 52 |
| Figure 36: Autopilot built into the airframe.....                                                                                                      | 55 |
| Figure 37: HappyKillmore ground control station with Google Earth™ flight path .....                                                                   | 56 |
| Figure 38: Airspeed sensor output .....                                                                                                                | 57 |
| Figure 39: Gyrometer output .....                                                                                                                      | 58 |
| Figure 40: Final mini UAV.....                                                                                                                         | 62 |
| Figure 41: Video position GPS correction.....                                                                                                          | 63 |
| Figure 42: Elevation graph from ArduPilot Mega Planner.....                                                                                            | 65 |
| Figure 43: Testing fly-by-wire mode.....                                                                                                               | 72 |
| Figure 44: Roll and pitch angles of UAV during test. ....                                                                                              | 73 |
| Figure 45: Waypoint following with a constant waypoint radius.....                                                                                     | 74 |
| Figure 46: Flight path during waypoint-following test.....                                                                                             | 75 |
| Figure 47: Distance from waypoint during the autopilot flight .....                                                                                    | 76 |
| Figure 48: Height-test flight path.....                                                                                                                | 77 |
| Figure 49: Desired and maintained flight height .....                                                                                                  | 78 |
| Figure 50: Desired and maintained flight height after compensation adjustment .....                                                                    | 79 |
| Figure 51: Flight path during hand-launch testing.....                                                                                                 | 80 |
| Figure 52: Climb ratio for different wind scenarios .....                                                                                              | 81 |
| Figure 53: Flight path during maximum height test .....                                                                                                | 82 |
| Figure 54: GCS testing.....                                                                                                                            | 83 |

|                                                                    |    |
|--------------------------------------------------------------------|----|
| Figure 55: ArduPilot Mega Planner screenshot .....                 | 84 |
| Figure 56: Flight path during UAS range verification testing ..... | 86 |

# List of tables

---

|                                                                                |    |
|--------------------------------------------------------------------------------|----|
| Table 1: Classification of UAVs (de Fatima Bento, 2008).....                   | 9  |
| Table 2: Petrol engine and electric motor comparison .....                     | 37 |
| Table 3: Pulso X2008/20 specifications .....                                   | 38 |
| Table 4: Autopilot initial comparison (Adapted from Arnott, 2007).....         | 39 |
| Table 5: Autopilot feature comparison .....                                    | 42 |
| Table 6: Ziegler-Nichols PID tuning method (Mallesham <i>et al.</i> 2009)..... | 45 |
| Table 7: UAV comparison .....                                                  | 61 |
| Table 8: GPS coordinate correction.....                                        | 64 |
| Table 9: Flight checklist.....                                                 | 71 |
| Table 10: Range determination.....                                             | 86 |
| Table 11: Roll validation data.....                                            | 89 |
| Table 12: Pitch validation data .....                                          | 89 |
| Table 13: GPS error validation .....                                           | 90 |
| Table 14: Pressure sensor validation.....                                      | 92 |
| Table 15: Wind speed sensor validation .....                                   | 93 |
| Table 16: Comparison between commercial UAVs and project UAV .....             | 97 |
| Table 17: Security requirements .....                                          | 98 |

# List of symbols

| Symbol     | Description                                 | Unit                             |
|------------|---------------------------------------------|----------------------------------|
| E          | Output error                                |                                  |
| g          | Gravitational acceleration                  | $\text{m.s}^{-2}$                |
| H          | Height above ground                         | m                                |
| I          | Current needed to power the UAV for an hour | Ah                               |
| $K_d$      | Derivative gain constant                    |                                  |
| $K_i$      | Integral gain constant                      |                                  |
| $K_p$      | Proportional gain constant                  |                                  |
| p          | Absolute pressure                           | Pa                               |
| $p_d$      | Dynamic pressure                            | Pa                               |
| R          | Ideal gas law constant for air              | $\text{J.kg}^{-1}.\text{K}^{-1}$ |
| T          | Temperature                                 | $^{\circ}\text{C}$               |
| t          | Time                                        | s                                |
| u          | Output                                      |                                  |
| x          | Roll adjustment                             | m                                |
| y          | Pitch adjustment                            | m                                |
| $\Delta H$ | Height change                               | m                                |
| $\Delta p$ | Pressure change                             | kPa                              |
| $\theta$   | Angle                                       | $^{\circ}$                       |
| $\theta_p$ | Pitch angle                                 | $^{\circ}$                       |
| $\theta_r$ | Roll angle                                  | $^{\circ}$                       |
| $\theta_y$ | Yaw angle                                   | $^{\circ}$                       |
| v          | Velocity of air                             | $\text{m.s}^{-1}$                |
| $\rho$     | Density of air                              | $\text{kg.m}^{-3}$               |
| $\tau$     | Differential time                           | s                                |
| $\omega$   | Rotational speed                            | $^{\circ}.\text{s}^{-1}$         |

# List of abbreviations

---

|                  |                                                     |
|------------------|-----------------------------------------------------|
| AMSL             | Above Mean Sea Level                                |
| ATC              | Air Traffic Control                                 |
| ATE              | Advanced Technologies and Engineering               |
| BEC              | Battery Eliminator Circuit                          |
| CCD              | Charge-coupled Device                               |
| CCTV             | Closed-circuit Television                           |
| CLI              | Command Line Interpreter                            |
| COTS             | Commercially Off-the-shelf                          |
| CPA              | Conventionally Piloted Aircraft                     |
| DIP              | Dual In-line Package                                |
| DIY              | Do-it-yourself                                      |
| EEPROM           | Electrically Erasable Programmable Read-only Memory |
| ESC              | Electronic Speed Control                            |
| EW               | East-west Axis                                      |
| F-H              | Flight Hour                                         |
| GCS              | Ground Control Station                              |
| GPS              | Global Positioning System                           |
| HAD              | Hole Accumulation Diode                             |
| HFACS            | Human Factor Analysis and Classification System     |
| HIL              | Hardware-in-the-loop                                |
| I/O              | Input/Output                                        |
| I <sup>2</sup> C | Inter-integrated Circuit                            |
| IDE              | Integrated Development Environment                  |
| IMU              | Inertial Measurement Unit                           |
| LiPo             | Lithium-polymer                                     |
| LOS              | Line-of-sight                                       |
| MALE             | Medium Altitude Long Endurance                      |
| MTOW             | Maximum Take-off Weight                             |
| NAS              | National Airspace                                   |
| PID              | Proportional-integral-derivative                    |
| RAM              | Random Access Memory                                |
| RC               | Radio Controlled                                    |
| RUVSA            | Russian Unmanned Vehicle Systems Association        |
| SR               | Short Range                                         |
| SSRR             | Safety, Security and Rescue Robotics                |
| UAS              | Unmanned Aircraft System                            |
| UAV              | Unmanned Aerial Vehicle                             |
| USB              | Universal Serial Bus                                |
| VTOL             | Vertical Take-off and Landing                       |
| μ                | Micro                                               |

# Glossary

---

## **EEPROM**

EEPROM is flash memory that uses floating gate transistors to store information. Each transistor gate can be programmed to be either a high or low (Kuo, 1992). This allows the EEPROM to store digital information.

## **Navigation gain**

This value determines how the servo control varies from the required correction (DIY Drones, 2011). It is the proportional gain of the PID loop controlling the GPS navigation system as described in Par. 3.4.4.

## **Pitot tube**

This comprises two small tubes that are mounted facing the direction the UAV is travelling. The one measures static pressure with holes in the side. The other measures total pressure with a hole in the front. The difference between the two pressures is the dynamic pressure (DIY Drones, 2011).

## **Battery eliminator circuit (BEC)**

This is a function of the electronic speed control (ESC). It supplies the radio control (RC) equipment with power used to power the motor. This eliminates the need for a second battery.

## **Infrared (camera)**

Infrared light has a wavelength slightly longer than visible light. The frequency of infrared light ranges between 1 THz and 400 THz. It is detected using a charge-coupled device (CCD) sensor.

CCD light-imaging sensors can pick up infrared light at a television resolution (Hau, 2009). This sensor absorbs photons that can be converted to a digital value. This allows cameras to have night vision and thermal radiation imaging.

### **Hole accumulation diode (HAD)**

HAD is a technique developed by Sony Corporation for reducing noise in signals from a CCD sensor.

### **ELAPOR<sup>®</sup>**

ELAPOR<sup>®</sup> is proprietary foam that is strong, durable and easy to repair (Cameron *et al.*, 2011).

### **Flight stability**

“The property of an aircraft or missile to maintain its attitude and to resist displacement, and, if displaced, to tend to restore itself to the original attitude” (Parker, 2002).

### **Random access memory (RAM)**

RAM consists of millions of small capacitors and transistors. If the capacitor on the gate of the transistor has a charge, it is digitally registered as a 1. If it does not have a charge, it is registered as a 0. In this manner digital information is stored. Any one of these cells can be accessed at any time if the column and row of the cell is known (How Stuff Works, 2011).

### **Universal serial bus (USB)**

USB is the standard connection between auxiliary devices and personal computers. It was developed by Intel<sup>®</sup> and other companies (Intel, 2011).

**UAV health**

UAV health monitoring is when the ground control station allows the operator to monitor irregular behaviour or system failures (Qiang *et al.*, 2009).

**Absolute altitude**

Absolute altitude is the vertical distance an object is above sea level.

**Flight simulator**

A simulation of aircraft flight that takes into account effects like air density, wind, etc.

**Hardware-in-the-loop (HIL) simulation**

HIL simulations are when the autopilot is used to give commands to a flight simulator in order to test its functionality.

**Biometric capabilities**

The capability to recognise humans based on behaviour and motion.

**Urban area**

An urban area refers to a region with a high population density.

## References used in Glossary

Kuo, C., Weidner, M., Toms, T., Choe, H., Chang, K., Harwood, A., Jelemensky, J. & Smith, P. (1992). A 512-kb flash EEPROM embedded in a 32-b microcontroller. *IEEE Journal of Solid-State Circuits*, 27(4):574-582.

Hau, L., Shi-chao, Z., Choa, H., Ming, Z. & Xiao-feng, M. (2009). *A near infrared imaging detection system based on Davinci platform*, 4-154-4-159. Conference proceedings of the 9<sup>th</sup> International Conference on Electronic Measurement & Instruments held in Beijing, China.

DIY Drones (2011). Homepage of Do-It-Yourself Drones. Available from: [www.DIYDrones.com](http://www.DIYDrones.com) (Accessed 13 March 2011).

Cameron, T., Colton, K. & Childers, D. (2011). *AETHER*. Bachelor's dissertation. California, USA: California Polytechnic State University, School of Electrical Engineering.

How Stuff Works. (2011). Homepage of How Stuff Works. Available from: [www.howstuffworks.com](http://www.howstuffworks.com) (Accessed 11 October 2011).

Parker, S. P. (2002). *McGraw-Hill Dictionary of Scientific and Technical Terms*. 6<sup>th</sup> edition. New York: The McGraw-Hill Companies.

Qiang, M., Sasa, M., Lianghua, X. & Yabin, W. (2009). *Research on security monitoring and health management system of medium-range UAV*, 854-857. Conference Proceedings of the 8<sup>th</sup> International Conference on Reliability, Maintainability and Safety held in Chengdu, China.

---

# 1. Introduction

---

---

## Summary

In this chapter project background of UAV technology and security systems are given.

---

## 1.1. Background

Most research presently addresses the development of unmanned aerial vehicles (UAVs) for military use (de Fatima Bento, 2008; Cameron, 2011). Military UAVs have developed at a faster pace than UAVs for civilian application (Ro *et al.*, 2007). This is largely because of the generous military funding made available (Gheorghe & Ancel, 2008). With regulations for flying UAVs in civil airspace in the process of being developed, the focus has shifted from military to civilian applications (Jovanovic, 2006). “There is now a sense of urgency to integrate UAVs into South African civilian airspace” (Ingham *et al.* 2006)

It is expected that UAVs will eventually routinely monitor cities and wildlife (Schwager *et al.*, 2011). For this to be successful a fleet of UAVs should be able to monitor the same site without colliding or simultaneously capturing images of the exact same location. Each UAV would monitor a section of the location. Mosaicking is the process of using multiple cameras and combining the images to obtain a complete view of the entire location (Schwager *et al.*, 2011).

The different military applications are overshadowed by the various civilian applications UAVs are suited for. Many of the smaller UAVs used in military applications can be adapted for use in civilian applications. One of the best-suited civilian applications of UAVs is security and surveillance (DeGarmo, 2004). There is, however, still a great deal of technology that needs to be integrated into UAVs for safe use in the national airspace system (NAS).

UAVs are generally powered by small high energy batteries. Energy density has increased with advances in batteries. This, and material advances, has caused a rapid increase in the development of UAVs (Michael & Kumar, 2011). High energy density batteries allow UAVs to be small (as small as 100 mm in length), and light (as light as 100 g) making transporting, initialising and storing the UAV easier. UAVs that are this small are not suited for security applications, as they cannot carry the necessary payload. UAVs for urban security applications are generally miniature UAVs, or mini UAVs.

The term mini UAV, refers to a UAV that has a maximum take-off weight (MTOW) of less than 30 kg and a range of less than 10 km. Mini UAVs also have a maximum altitude of less than 300 m and endurance of less than 2 hours.

In South Africa the highest population density occurs in the South African urban areas: Cape Town, Johannesburg, Pretoria and Durban. For this reason UAV testing and the urban security system analysis will be done in Pretoria, Gauteng.

## **1.2. Purpose of study**

The goal of this study is to develop a fully integrated urban security system using UAVs for patrolling the perimeter of Silver Lakes Golf Estate. This should be a complete system; ready for application in the security field. This dissertation investigates the integration of a mini unmanned aircraft system (UAS) into the NAS. This investigation is necessary because integration into an urban security system cannot be done without considering integration into the NAS (Gheorghe & Ancel, 2008; DeGarmo, 2004).

## **1.3. Scope of study**

This study focuses on the urban security application of UAVs. This necessitated the development of a mini UAV specifically for urban security purposes. The urban security system is analysed for shortcomings in the security site that was chosen. After identifying the requirements and limitations the focus shifted to integrating the mini UAV into an urban security system.

A fully autonomous mini UAV is developed that can send and receive telemetry from a ground control station (GCS) in order to relay its attitude, position and a live video feed. The system needs an operator for monitoring the video feed. Future study will, however, need to focus on software analysis of the video feed which could remove the human factor to a large extent. The system requires two people to launch it. Thereafter, it will require only one person to monitor the video feed.

The UAV can have automatic take-off and landing capabilities, but this has not been incorporated into this study. The UAV developed for this study is launched by hand in manual mode and piloted to the preset operational altitude. It is then switched to autopilot mode when it will automatically and independently follow waypoints at the predetermined altitude. The pilot is then required to land the UAV in manual mode.

A video camera is incorporated into the unmanned system in order to stream video for real-time decision-making and monitoring. Video-streaming is done from the autopilot using a separate telemetry system. The video camera is able to switch to night vision automatically for night-time perimeter monitoring.

To reduce costs, a short-range (2 km) mini UAV was developed. Building a long-range (longer than 200 km) UAV requires a satellite link to compensate for the curvature of the earth. Thus, the line-of-sight control of the UAV is negated. A long-range UAV also requires considerably more time to develop and is exorbitantly expensive. With a telemetry range of 2 km the UAV is able to monitor sites up to 4 km in diameter, with a perimeter of 12.56 km. This telemetry range is adequate for the purpose of doing perimeter-based security surveillance.

A conventional off-the-shelf airframe is used since it is able to fly at slower speeds and still remain stable. This will expedite the development of the UAV. Further development of the UAV can ultimately be done on other airframes that are more stable at higher speeds. This option will, however, not be investigated in this study. A summary of this dissertation is listed below:

- **State-of-the-art UAV technology.** This chapter reports on published work to determine the role and limitations of UAVs in an urban security environment.
- **Assembly and integration.** In this chapter the UAV building and security system integration process are discussed.
- **Verification and validation.** In this chapter the UAV capabilities and integration requirements are verified. The measurements are then confirmed by validating the UAV sensor data.

- **Conclusion.** In this chapter the study goals are reviewed and compared with the attained results. The project is concluded and recommendations for future study are made.

---

## 2. State-of-the-art UAV technology

---

---

### Summary

In this chapter a background is provided into the state-of-the-art technology applied during the design of mini UAVs. Mini UAVs are defined and classified. Different commercial mini UAVs are studied to determine their operational requirements. The chapter also investigates the regulations governing UAV flight in the NAS. This specifies the benchmark requirements for the development of the mini UAV. It focuses on the shortcomings in urban security systems and what the role of UAVs is. This allows the specification of integration requirements of the UAV into an urban security system.

---

## **2.1. Introduction**

The purpose of this chapter is to determine the state-of-the-art technology used in the design of mini UAVs. The first step is to define what UAVs are, and how they are classified. This is done to determine which type of UAV will be most suited to security applications. The study also investigates current UAVs that are suitable for security applications. This provides a good benchmark on the performance that can be expected from the UAV.

A study is done into the typical layout of the UAS and how the regulation affects its operation. Attention is also given to the urban security system and exactly what the role of the UAV will be. Special consideration is given to the specifications and requirements of the reference security system.

## **2.2. What is a UAV?**

According to the United States Department of Defense a UAV can be defined as follows: “An aircraft or balloon that does not carry a human operator and is capable of flight under remote control or autonomous programming” (United States of America Department of Defense, 2001). Originally, autopilots were only used for aircraft stabilisation and later evolved into full flight control (Ribeiro & Oliveira, 2010).

UAVs usually have dedicated airframes specifically designed for autonomous flight. However, conventionally piloted aircraft (CPA) have also been converted to become UAVs (Tikanmäki *et al.*, 2011). UAVs have been in use from as early as the 1950s for reconnaissance purposes (Willy, 2003).

The minimum electronic sensors required to operate UAVs satisfactory measure pitch, roll and the global position. These inputs are then used to generate the desired response such as Global Positioning System (GPS) coordinate tracking (Chao *et al.*, 2007). Determining the yaw is not required since successive GPS coordinates determine the direction the UAV is travelling. A picture of a typical military mini UAV is shown in Figure 1.



**Figure 1: AeroVironment RQ-11 Raven being hand-launched (Defense Industry Daily, 2011)**

A typical UAS includes a ground control station, telemetry system, UAV and the facilities for launching, maintaining and transporting the UAV (Russel, 2007). If one of these elements malfunctions the system cannot function properly. General requirements for civilian use are that UAVs should at least be safe, economical, easy to use, credible, fast, reliable and able to document information in real-time (Tikanmäki *et al.*, 2011).

There are four ways of launching the UAV if rapid deployment is desired: rocket-launching, mechanical launching, hand-launching and air-deployment (dropped from another aircraft). Of these launching methods, hand-launch is the most economical; air deployment from another aircraft is the fastest (Cheng, 2007).

### **2.3. Classification**

UAV's may be categorised according to their size, endurance and altitude. These categories are labelled micro ( $\mu$ ), mini, short-range (SR) and medium altitude long endurance (MALE), to name but a few. Some of these UAVs also have vertical take-off and landing (VTOL)

capabilities. VTOL gives UAVs a great advantage since they do not need a runway to land or take off (Tikanmäki *et al.*, 2011). The typical classification of UAVs is shown in Table 1.

**Table 1: Classification of UAVs (de Fatima Bento, 2008)**

|                   | Category (acronym)                     | Maximum Take Off Weight (kg) | Maximum Flight Altitude (m) | Endurance (hours) | Data Link Range (Km) | Example                                                                                                                        |                                                                                                                                           |
|-------------------|----------------------------------------|------------------------------|-----------------------------|-------------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
|                   |                                        |                              |                             |                   |                      | Missions                                                                                                                       | Systems                                                                                                                                   |
| Micro/Mini UAVs   | Micro (MAV)                            | 0.10                         | 250                         | 1                 | < 10                 | Scouting, NBC sampling, surveillance inside buildings                                                                          | Black Widow, MicroStar, Microbat, FanCopter, QuattroCopter, Mosquito, Hornet, Mite                                                        |
|                   | Mini                                   | < 30                         | 150-300                     | < 2               | < 10                 | Film and broadcast industries, agriculture, pollution measurements, surveillance inside buildings, communications relay and EW | Mikado, Aladin, Tracker, DragonEye, Raven, Pointer II, Carolo C40/P50, Skorpio, R-Max and R-50, RoboCopter, YH-300SL                      |
| Tactical UAVs     | Close Range (CR)                       | 150                          | 3.000                       | 2-4               | 10-30                | RSTA, mine detection, search & rescue, EW                                                                                      | Observer I, Phantom, Copter 4, Mikado, RoboCopter 300, Pointer, Camcopter, Aerial and Agricultural RMax                                   |
|                   | Short Range (SR)                       | 200                          | 3.000                       | 3-6               | 30-70                | BDA, RSTA, EW, mine detection                                                                                                  | Scorpi 6/30, Luna, SilverFox, EyeView, Firebird, R-Max Agri/Photo, Hornet, Raven, phantom, GoldenEye 100, Flyrt, Neptune                  |
|                   | Medium Range (MR)                      | 150-500                      | 3.000-5.000                 | 6-10              | 70-200               | BDA, RSTA, EW, mine detection, NBC sampling                                                                                    | Hunter B, Mücke, Aerostar, Sniper, Falco, Armor X2, Smart UAV, UCAR, Eagle Eye+, Alice, Extender, Shadow 200/400                          |
|                   | Long Range (LR)                        | -                            | 5.000                       | 6-13              | 200-500              | RSTA, BDA, communications relay                                                                                                | Hunter, Vigilante 502                                                                                                                     |
|                   | Endurance (EN)                         | 500-1.500                    | 5.000-8.000                 | 12-24             | > 500                | BDA, RSTA, EW, communications relay, NBC sampling                                                                              | Aerosonde, Vulture II Exp, Shadow 600, Searcher II, Hermes 450S/450T/700                                                                  |
|                   | Medium Altitude, Long Endurance (MALE) | 1.000-1.500                  | 5.000-8.000                 | 24-48             | > 500                | BDA, RSTA, EW weapons delivery, communications relay, NBC sampling                                                             | Skyforce, Hermes 1500, Heron TP, MQ-1 Predator, Predator-IT, Eagle-1/2, Darkstar, E-Hunter, Dominator                                     |
| Strategic UAVs    | High Altitude, Long Endurance (HALE)   | 2.500-12.500                 | 15.000-20.000               | 24-48             | > 2.000              | BDA, RSTA, EW, communications relay, boost phase intercept launch vehicle, airport security                                    | Global Hawk, Raptor, Condor, Theseus, Helios, Predator B/C, Libellule, EuroHawk, Mercator, SensorCraft, Global Observer, Pathfinder Plus, |
| Special Task UAVs | Lethal (LET)                           | 250                          | 3.000-4.000                 | 3-4               | 300                  | Anti-radar, anti-ship, anti-aircraft, anti-infrastructure                                                                      | MALI, Harpy, Lark, Marula                                                                                                                 |
|                   | Decoys (DEC)                           | 250                          | 50-5.000                    | < 4               | 0-500                | Aerial and naval deception                                                                                                     | Flyrt, MALD, Nuika, ITALD, Chukar                                                                                                         |
|                   | Stratospheric (Strato)                 | TBD                          | 20.000-30.000               | > 48              | > 2.000              | -                                                                                                                              | Pegasus                                                                                                                                   |
|                   | Exo-stratospheric (EXO)                | TBD                          | > 30.000                    | TBD               | TBD                  | -                                                                                                                              | MarsFlyer, MAC-1                                                                                                                          |

## 2.4. Applications

UAVs were originally designed and developed for military applications and UAV deployment is expected to reduce the risk of human life loss (Willy, 2003; Dufrene, 2005). Examples of military uses of UAVs are border patrol, search-and-rescue, bombing, reconnaissance and target acquisition (Tikanmäki *et al.*, 2011). Recently, however, great interest has been shown in using UAVs in civilian applications (Ribeiro & Oliveira, 2010). Possible civilian applications of UAVs are listed below (Tikanmäki *et al.*, 2011):

- Natural disaster monitoring
- Humanitarian relief
- Environmental monitoring
- Weather and storm tracking
- Agricultural applications, such as crop monitoring
- Cargo transport
- Wireless communication
- Wildlife monitoring
- Security surveillance
- Traffic and accident monitoring
- Area mapping
- Emergency supply delivery
- Law enforcement
- Riot control
- Aerial photography

It can be seen that there are many civilian applications for UAVs, mainly because they can substitute the “dull, dirty and dangerous” CPA missions (Shawn & Weed, 2002). It seems as if there are numerous gaps in the market that can be filled by UAVs when the regulation is finalised (DeGarmo, 2004).

## 2.5. Regulation

Present aviation restrictions limit UAVs from entering restricted and populated areas. No official civil UAS regulation has been established (Ingham *et al.* 2006). The biggest concern regarding UAVs is collisions with passenger aircraft (Clothier *et al.*, 2011). Existing regulation only tries to operate the UAV safely and is limited to larger aircraft (Beainy & Mai, 2009). Large UAV flights are considered to be once-off events and authorisation is needed for every flight (Gheorghe & Ancel, 2008). Regulation will need to specify — among others — the following (DeGarmo, 2004):

- **A dedicated frequency for UAV telemetry.** International UAV flights will ideally use one frequency when crossing borders. For this reason an international UAV frequency should be established.
- **Flight zones.** The NAS is subdivided into flight zones according to their entry requirements, altitude, location and required communication capabilities.
- **The skill level and medical requirements of the UAV operator.** Since this regulation has not been specified, assumptions are made regarding the skill requirements. These requirements will probably be dependent on the UAV size, operational flight zones, UAV capabilities and the amount of UAVs controlled simultaneously.
- **Air traffic control (ATC) operations.** This refers to communication between ATC and the UAV or UAV operator.
- **Emergency procedures.** Procedures in case of UAV mishaps must be specified.
- **Requirements for airworthiness of a UAV.** This refers to safety standards, engineering requirements, communication capabilities, etc.

If the regulation in the United Kingdom can be used as a benchmark, mini UAVs will be ideally suited for urban security applications. Radio-controlled (RC) aircraft can be classified as either a model aircraft or a UAV, of which only the latter is strictly regulated. Maintaining model status will reduce operational cost as a result. An RC aircraft is classified as a model airplane if it does

not carry live cargo, does not weigh more than 35 kg and is operated in line-of-sight (LOS) (Sinnathamby *et al.*, 2010).

South Africa is not at the forefront of UAV technology and regulation development. It is likely that the South African UAV regulation will be adopted from the United Kingdom, the United States of America or Australia (Ingham *et al.* 2006). Until the regulation for flying UAVs has been developed, the priority in South Africa will probably remain the development of safe UAVs. Some of the features that will need to be incorporated into successful larger UASs according to presently proposed Australian regulation are listed below (Russel, 2007):

- The ability to avoid obstacles (visual collision avoidance).
- The filing of flight plans.
- Over the horizon communication.
- Situational awareness.
- ATC voice recognition.
- Improved link recovery technology.
- Autonomous take-off and landing.
- Procedures for safe flight.

If these issues are addressed, and the system is tested to the regulating authorities' satisfaction, UAVs may be allowed to fly in the NAS. Some of these features are advanced and will most probably only be enforced on UAVs that have UAV status — as opposed to model status. The main considerations when integrating UAVs into the NAS are listed below (DeGarmo, 2004):

- Safety
- Security
- Regulation
- Control
- Society

UAVs will be required to conform to existing regulations rather than create new regulations specifically for UAVs (DeGarmo, 2004). For the purpose of this study the UAV will be classified as an RC model in order to avoid incorporating these advanced features.

## **2.6. Present UAVs suited for civilian applications**

Mini UAVs seem to be well-suited for civilian applications since they are relatively cheap to develop. UAVs are also expected to be less likely to cause serious injuries and death in the event of system malfunction (Dufrene, 2005). Research into present mini UAVs will give insight into the state-of-the-art technology used during the design of civilian UAVs. The following is a selection of popular mini UAVs presently suited for use in civilian applications:

### **2.6.1. Carolo P50**

The Carolo P50 was developed by Mavionics in Germany. It is a mini UAV with a wingspan of 0.5 m and a length of 0.4 m. It is presently under development for use in reconnaissance and surveillance applications. Police applications might also be possible. The Carolo P50 is capable of speeds up to  $74 \text{ km.h}^{-1}$  and has an endurance of 30 min. It can reach altitudes in excess of 457 m (RUVSA, 2011). An image of the Carolo P50 is shown in Figure 2.



**Figure 2: Mavionics' Capolo P50 (RUVSA, 2011)**

This UAV communicates with a GCS, relaying its position and attitude while allowing the operator to give the UAV commands. The UAV autonomously tracks waypoints while streaming

live video through a separate data link. It is launched by hand and lands by skidding on the ground (RUVSA, 2011). The Carolo P50 was still in development when this document was written and consequently the cost is unknown.

### **2.6.2. CyberBug Micro**

The CyberBug Micro was developed by the company Cyber Aerospace located in Oklahoma, USA. It has a wingspan of 0.76 m and can be assembled in three minutes. It was developed for use in reconnaissance, surveillance and search-and-rescue missions. The MTOW is 1.18 kg and it has an endurance of 40 minutes. The maximum height it can reach is 3,050 m. An image of the CyberBug Micro is shown in Figure 3.



**Figure 3: Cyber Aerospace's CyberBug Micro with GCS and other equipment (Cyber Aerospace, 2011)**

The UAV communicates with a GCS relaying its position and attitude over a maximum distance of 5 km. It also streams live video to the GCS or a hand-held video receiver. The CyberBug Micro is launched by hand and costs R272,300 (Cyber Aerospace, 2011); exchange rate of R7 to US\$1 assumed.

### **2.6.3. Kiwit**

The Kiwit was developed by the company ATE Aerospace located in Pretoria, South Africa. The Kiwit was developed for policing applications, site surveillance, aerial photography, wildlife

monitoring and disaster surveillance. The UAV can be hand-launched since it weighs only 4 kg. It has an endurance of 45-60 minutes. Assembly takes only 10 min. An image of the Kiwit is shown in Figure 4.



**Figure 4: ATE's Kiwit flying low (ATE-Group, 2010)**

The Kiwit is operated from a GCS which displays its position and attitude. The pre programmed flight plan can also be changed from the GCS while the UAV is airborne. Live video is streamed to a portable video receiver and the coordinates of the centre of the image is displayed (ATE-Group, 2010). The price of the Kiwit is undisclosed.

#### **2.6.4. Summary of popular mini UAVs**

The aforementioned mini UAVs are suitable for civilian security applications since they were developed for reconnaissance and surveillance purposes. They can all be hand-launched because of their small size and light mass. It is concluded that they have a mass of less than 5 kg and a wingspan of less than 2 m.

The basic requirements for successful UAVs are LOS telemetry link of approximately 5 km with an endurance of slightly less than an hour. These goals are all achievable for this study. The UAVs that are currently available are expensive. This provides the opportunity for developing a low-cost system with similar performance.

## 2.7. Sensors and control

A UAV airframe is usually controlled by three servomechanisms (“servos”), and the speed throttle. The three servos are used for controlling the elevator, rudder and ailerons respectively. The elevator is situated on the horizontal tail wing and controls the pitch of the airplane. Ailerons are located on the main wings of the airplane and control the roll of the airframe by moving the flaps on the wings in opposite directions.

Finally, the rudder is situated on the rear end of the fuselage and controls the yaw of the airplane and can assist with the roll of the airframe since it is not symmetrical around the centreline (Müller, 2008). The three axes of rotation are shown in Figure 5.

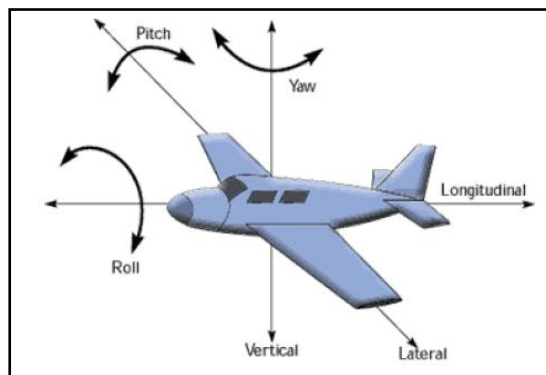


**Figure 5: Three axis of rotation used to describe aircraft orientation (Bozkurt & Aslan, 2009)**

The last method of controlling the UAV is the throttle that controls the speed of the electric motor driving the propeller (Müller, 2008). These controls are utilized to orientate the airframe in any desired direction.

The attitude and the three-dimensional position of the airplane have to be known to control the aircraft efficiently. The three-dimensional position is usually determined with a GPS although absolute pressure sensors might be used in combination with the GPS for determining the height (Chao, 2007). The attitude of the airframe is determined using one of three methods:

The first method uses a micro-inertial guidance system with an added GPS. This system has a gyrometer and an accelerometer, collectively known as an inertial measurement unit (IMU). The gyrometer is a device that measures the angular orientation. The accelerometer measures the acceleration in the direction of the three axes (Chao, 2007).

The second method for determining the attitude of the UAV is to use infrared sensors. This system has four infrared sensors which are paired on two axes. Two sensors on the same axis measure the heat difference between them to determine attitude. The infrared emitted from the earth is higher than the infrared from the sky (Chao, 2007).

The last method for determining the attitude of the UAV is by using video-based navigation. This method still requires thorough research and is currently combined with IMU sensors. The sensors add more stability to the autopilot navigation (Chao, 2007). It is used for low-flying applications since it is the most accurate way to determine the position of houses and trees (Barrows, 2002).

For airframes that can hover, a magnetometer is often added to determine the orientation while hovering. Autopilots can require the relative airspeed to control the throttle setting. This measurement is taken using a pitot tube because of its accuracy and low cost (DIY Drones, 2011).

## **2.8. Typical UAS layout**

A typical system comprises sensors measuring the attitude and position of the airframe. The microcontroller interprets these sensor measurements and controls the UAV via the servo and throttle controls (Chao, 2007). The attitude and position are relayed to a GCS via wireless modules. The modules not only allow the operator to see the airframe position; but can also stream live video. Some GCSs even allow the user to transmit commands to the airframe.

The airframe can receive RC inputs from either the GCS or a handheld RC. The RC inputs override the autopilot control (Müller, 2008). A typical layout is shown in Figure 6.

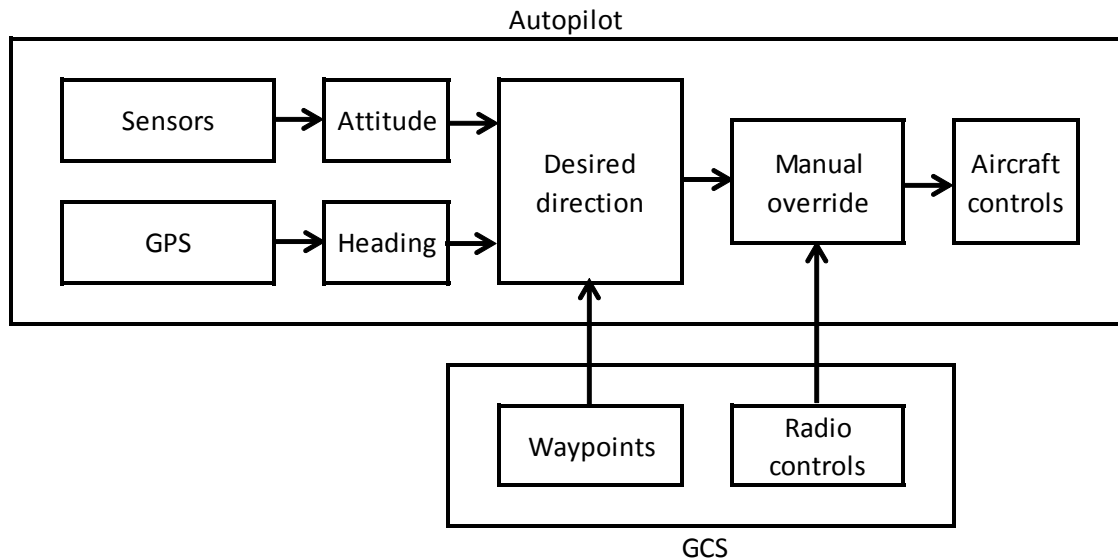


**Figure 6: Typical unmanned aircraft system layout (Adapted from Müller, 2008)**

It can clearly be seen that the UAV is only one part of the many needed for successful UAV missions. The UAS includes a GCS for displaying and storing information, a telemetry link for transmitting and receiving information, a system for maintaining the UAV, and transport for getting the UAV to desired launch locations (Russel 2007).

## 2.9. General urban security systems

The urban security system can be subdivided into six main categories, namely: information security, environment, personnel, physical, technology and policy (Dufrene, 2005). All of these categories must be addressed for an urban security system to be effective.

Information refers to the information gathered from the physical equipment, such as cameras and personnel. For information to be secure it needs to be confidential, nonreputable, authentic,

available, of high integrity, and someone (or something) must be held accountable (Dufrene, 2005). The typical urban security system layout is illustrated in Figure 7.

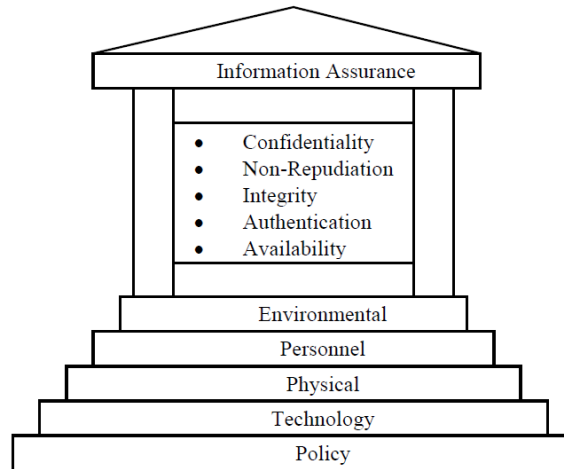


**Figure 7: Typical urban security system layout (Dufrene, 2005)**

A security environment can include any situation where someone or something needs to be protected or monitored. It includes border patrol, perimeter surveillance, crop and wildlife monitoring environments, to name but a few. In typical urban security systems personnel can be separated into guards that patrol the premises, regulate the entrance to the premises and monitor video data.

New technology must be integrated with urban security systems on a regular basis. This is where UAVs feature, since it represents new technology that has not been properly integrated into human urban security systems (Tvaryanas *et al.*, 2005). Transferring new technology into existing urban security systems has largely been unsuccessful, because new technology can be expensive and people also tend to view it with suspicion (Pavlidis *et al.*, 2001). Suspicion can be overcome by making the public aware of the benefits of using UAVs (DeGarmo, 2004).

## **2.10. South African urban security system**

Typically, South African urban security systems have stationary closed-circuit television (CCTV) cameras. Humans patrol the perimeter and monitor the video feed from the CCTV cameras in a control room (Lipton *et al.*, 2002). This system relies heavily on humans for detection, assessment and reaction to threats (Pavlidis *et al.*, 2001). The effectiveness depends on human vigilance rather than equipment (Shah *et al.*, 2007).

In South Africa, urban security systems are data intensive. At some sites cameras monitor more than 1,000,000 events daily which are stored as more than 1,000 Gb of data. Manually reviewing this data is expensive and time-consuming (Dhlamini *et al.*, 2007). This data is often not recorded and then relies solely on human monitoring (Pavlidis *et al.*, 2001).

This problem is, however, not limited to South Africa. There are numerous cameras in the United Kingdom and not enough people to continuously monitor the video feeds. For this reason, most data is only monitored after a breach in security. This security system is in need of an automated video surveillance system that can warn guards of a security breach (Dick & Brooks, 2003). Using artificial intelligence can give security systems this capability (Dhlamini *et al.*, 2007).

Biometric capabilities can be added to identify trespassers (Dhlamini *et al.*, 2007). This allows the security cameras to track individuals and identify suspicious behaviour. It also allows real-time decision-making based on analysis of the video feed (Dick & Brooks, 2003). Biometrics can differentiate between background and foreground parts of an image. This is illustrated in Figure 8 and Figure 9.



**Figure 8: A recorded video of a parking area (Shah *et al.*, 2007).**



**Figure 9: Biometric capabilities applied to Figure 8 (Shah *et al.*, 2007).**

The testing of the urban security features will be carried out at Silver Lakes Golf Estate which is situated in Pretoria East. The estate considered to be a state-of-the-art South African security estate and was voted the most secure estate in South Africa in a 2009 independent survey (Silver Lakes, 2009). Silver Lakes Golf Estate was chosen because it is a relatively large urban environment compared to other residential areas and has a high population density (1200 houses). It also has a diverse range of living environments, including houses, complexes, a retirement village, a golf course and a game reserve. A map of Silver Lakes Golf Estate is shown in Figure 10.



Figure 10: Silver Lakes Golf Estate map — the red oval denotes where flight testing was conducted (Silver Lakes, 2009)

Silver Lakes Golf Estate is 3,132 m at its widest with a perimeter of 8,030 m. It is assumed that Silver Lakes Golf Estate is the largest area that the mini UAV will be required to monitor. As a result the UAS will need a range of half the diameter — which is 1,566 m — and will require sufficient endurance to circle the perimeter at least once.

Silver Lakes has electric fencing around its perimeter with security guards monitoring the entrance. Fingerprints are used to enter the estate. CCTV cameras monitor the activity at the entrance gates and around the perimeter. The Silver Lakes security control room is shown in Figure 11.



**Figure 11: Silver Lakes Golf Estate control room**

Guards also patrol the inside of the residential area on a regular basis. The problem with this specific urban security system is that the perimeter is lined with houses, and the area is

surrounded by other enclosed residential areas. This limits the patrol guards' ability to monitor the perimeter since access is difficult. Monitoring the perimeter is very effective for security systems but can lead to claims of invasion of privacy (Pavlidis *et al.*, 2001; Ro *et al.*, 2007).

### **2.11. Alternative security considerations**

Human operators are unable to effectively monitor multiple screens. This is because it is difficult to give their full attention to more than one thing simultaneously (Lipton *et al.*, 2002). It is also likely that the human patrol will either sleep on duty, or refrain from doing some of the tasks unless there is a system to monitor them.

Humans are responsible for 17% of UAS failures (DeGarmo, 2004). It is apparent that the human factor needs to be minimised for urban security systems to be effective. This can be achieved by using UAVs with artificial intelligence software to monitor the video feed.

One of the main concerns in South Africa, as well as the rest of the world, is safety; especially where surveillance of urban security sites are concerned. UAVs must at least be as safe as CPAs for integration into the NAS (DeGarmo, 2004). In order to establish its safety, it is important to conduct extensive testing of the UAV prior to releasing it for general use. After commissioning the UAV it is very important to ensure that it is maintained (Qiang *et al.*, 2009).

Another safety concern is collisions with other aircraft or uncontrolled flight into the ground. For this reason collision avoidance capabilities are essential (DeGarmo, 2004). This requires onboard video processing. Collision avoidance is done using a typical procedure described below:

- Identify the danger.
- Determine whether it is stationary or moving.
- Analyse the trajectory of the obstacle.
- Avoid colliding with it.

Monitoring the UAV during flights is important. It allows the operator to ensure that the UAV is functioning correctly. The information must be stored for analysis of the UAV health after the flight, and can also allow for preventative maintenance (Qiang *et al.*, 2009). Transmitting the battery life of the UAV to the GCS allows the operator to avoid crashing due to power loss.

### **2.12. UAVs in security applications**

Recently, there has been great interest in using UAVs for security applications. UAVs can be used in civilian security applications because of the monotonous nature of the tasks (Shawn & Weed, 2002). Unlike manned aircraft, UAVs can loiter in a specific location, or terrain, for extended periods without getting tired or bored. Autopilots also do not take risks (DeGarmo, 2004).

The Predator, for example, has endurance in excess of 24 hours while the piloted Blackhawk helicopter has endurance of 2 hours and 18 minutes. UAVs are also cheaper than piloted aircraft with similar capabilities — the Predator costs US\$4.5 million while the Blackhawk costs US\$8.6 million (Bolkcom, 2004). Using autopilots instead of pilots also improves the collision avoidance capability of the aircraft, since UAVs can have a 360° viewing angle (DeGarmo, 2004).

When UAVs are compared by costs per flight hour, per kilogram, it costs between ten and a hundred times more to conduct UAV missions than piloted aircraft missions (Russel, 2007) as illustrated in Figure 12. This cost excludes the pilot training costs which could be as high as 90% of the total cost. UAV pilot training costs are significantly lower (Ingham *et al.* 2006). This ensures that UAV operational costs are lower than the cost of piloted aircraft.

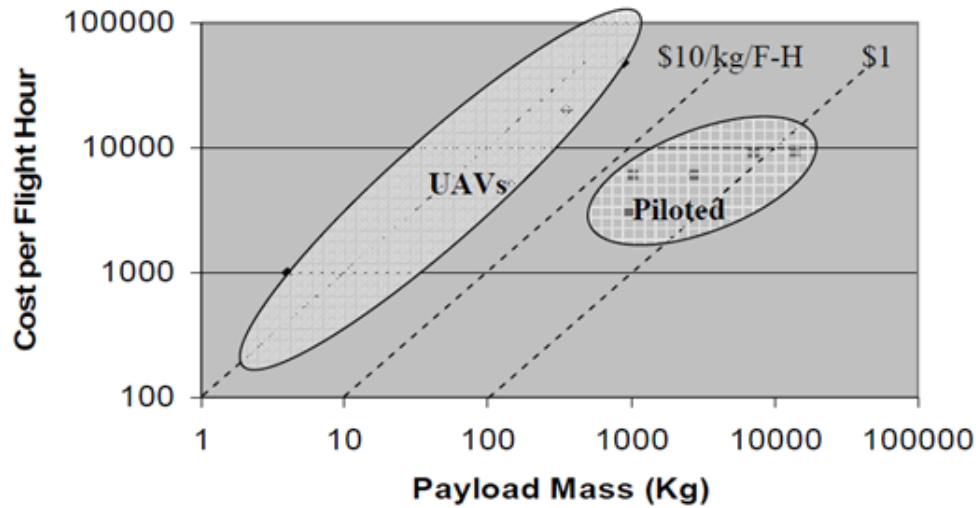


Figure 12: Cost of mission per flight hour per kilogram (Russel, 2007)

The camera required for basic surveillance has a mass of 50 g (Rangevideo, 2011) and the mass of the autopilot, relative to a human pilot, is negligible (DIY Drones, 2011). Thus, it allows mini UAVs for surveillance application to have a MTOW as light as 100 g (Clothier *et al.*, 2007). The mass difference between UAVs and CPAs is shown in Figure 13.

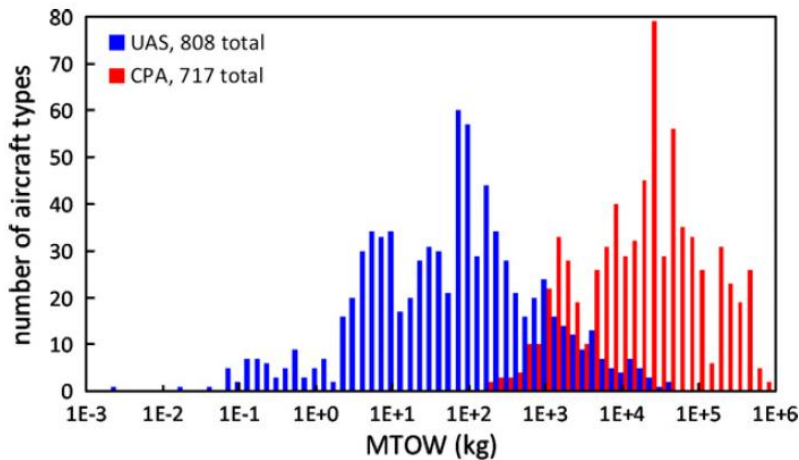


Figure 13: Mass of UAVs and CPAs (Clothier *et al.*, 2011)

UASs are also superior to stationary cameras since they can cover larger areas (Shawn & Weed, 2002) and have the ability to track trespassers. Stationary cameras do, however, prevent crime effectively since people are less likely to commit crimes when they think that they are recorded

on camera (Lipton *et al.*, 2002). Adding notice boards specifying that an area is monitored by UAV can give the UAS the same advantage.

Other functions of UASs that can be incorporated for security applications are information analysis methods and applicable software for control and interpretation (Dufrene, 2005). These can add significant costs to a project. A typical UAV mission timeline is shown in Figure 14.

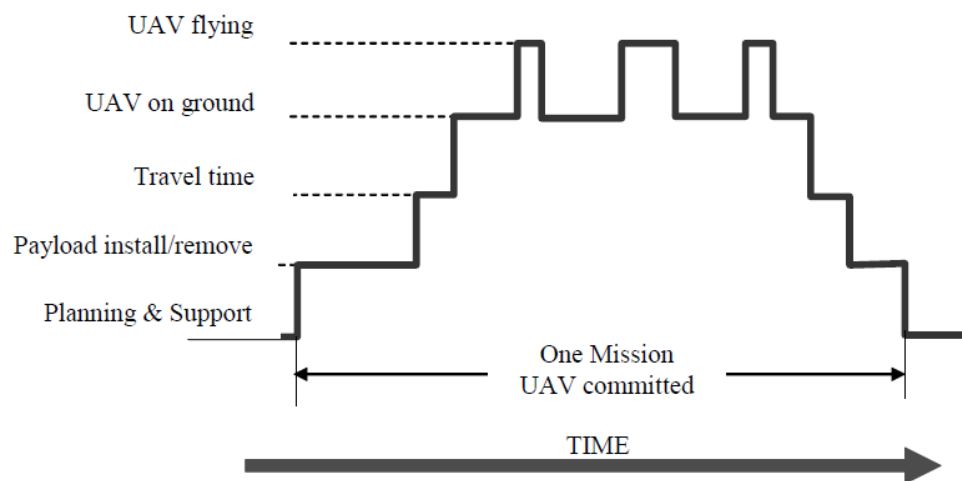


**Figure 14: Typical UAV mission cycle (Russel, 2007)**

The mission timeline emphasises the importance of having extended flight endurance, since a longer flight time makes missions more economical. With increasing experience the tasks of payload installation and uploading the code will become more efficient and thus shorter (Russel, 2007).

Piloted aircraft rely heavily on the pilot's sight while UAVs can be guided using high-resolution and infrared cameras. Thus, UAVs have a huge advantage when flying where visibility is impaired. UAVs can also react faster to turbulence, which makes them more stable for taking photos and video (Sinnathamby *et al.*, 2010).

The mishap rates for UAVs are significantly higher than those of piloted aircraft (DeGarmo, 2004). Less redundancy has been built into UAVs (Bolkcom, 2004) because of the reduced risk of human deaths. This is especially true for mini UAVs such as the Pioneer. The more capable UAVs — such as the Global Hawk and Predator — compare favourably with the mishap rates for piloted aircraft such as the Lockheed U-2 (Russel, 2007). This is clearly shown in Figure 15.

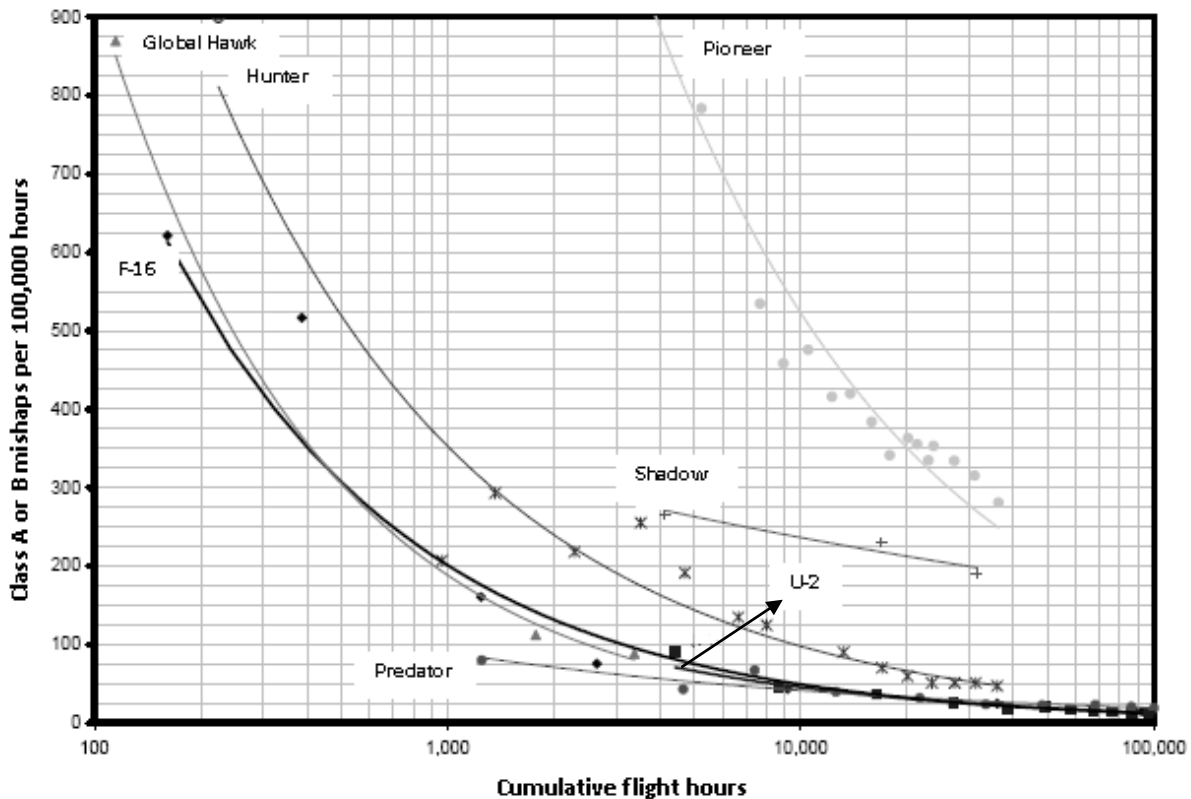


**Figure 15: CPA and UAV mishap rates per flight hours (adapted from Cambone *et al.*, 2005)**  
**A class A mishap refers to fatalities, permanent disability, midair collisions or damage in excess of US\$1 million. A class B mishap refers to partial disability or damage of US\$200,000 – 1,000,000.**

More than half of UAV mishaps are caused because UASs have not been integrated sufficiently with human systems (Tvryanans *et al.*, 2005), and due to a lack of situational awareness. Situational awareness can be improved by supplying the operator with a three-dimensional overlay of the airframe on a globe such as Google Earth™ (Drury *et al.*, 2006).

During a UAV collision few parts are re-usable. Thus, the importance of proper ground testing and using low-cost parts cannot be over-emphasised. The risks of collision can be mitigated by doing hardware-in-the-loop (HIL) simulations with a flight simulator such as X-plane (Ribeiro & Oliveira, 2010).

Consideration must be given to whether it is more economical to buy multiple smaller UAVs, or to buy expensive UAVs that are less prone to accidents. When opting to buy cheaper UAVs the chances of losing some of them in accidents are greater (Russel, 2007).

Mini UAVs can track more targets simultaneously, since numerous UAVs can be deployed for the same cost of larger UAVs. Mini UAVs are more agile than larger UAVs and are so small that they are almost invisible to humans observing them from the ground. Larger UAVs require more personnel and equipment for transportation. Also, larger UAVs will probably require a landing strip to take off and land (Shawn & Weed, 2002). Larger UAVs do, however, have better stability in strong winds. From this information it can be concluded that mini UAVs are more economical in terms of functionality per costs (Shawn & Weed, 2002).

This study was done on military-standard UAVs that require more capabilities. Mini UAVs will be even more suited to surveillance applications since more expensive UAVs have excessive capabilities. It should, however, be noted that UAV operators for security applications will probably be less educated than their military-trained counterparts. Mini UAVs are thus favoured, since they are easier to operate and cheaper to replace. It is therefore acceptable to assume that mini UAVs will be better suited to certain security applications.

### **2.13. Integration requirements**

Using the aforementioned information the requirements for integration of the UAV into an urban security system can be defined. The first requirement is that the UAV must be small enough to be hand-launched, which also restricts the mass of the UAV. The size requirement was necessary

since residential areas and buildings such as Silver Lakes Golf Estate usually do not have their own landing strips.

The second requirement is that the UAV must have live video-streaming. If there is any delay in the information the trespasser will be able to escape before being arrested. The operator must also know the coordinates of the video that is displayed. The UAV coordinates are usually not the same as the video coordinates. Any pitch or roll will cause the coordinates to differ; this effect is increased with an increase in height. Knowing where the video camera is pointing is essential for quick response.

The next requirement is that the UAV must be able to fly high enough for it to avoid detection — the UAV should not be seen or heard. In addition, it is a requirement that the UAV position must be overlaid on Google Earth™ to help the operator with visualisation of the flight path.

All the relevant information must be shown on a single screen, which will simplify the operator's task. This information must at least be transmitted over a distance of 1566 m, which will allow for effective monitoring of the Silver Lakes Golf Estate. The UAV must have sufficient endurance to circle the entire perimeter at least once, regardless of the wind velocity.

The information that is displayed must be secure. If the information is interruptible, trespassers can block the video feed to the security operator. By intercepting the video feed, information can also be changed or replaced. This can allow terrorists to use UAVs as weapons if they are flown in urban environments (DeGarmo, 2004).

The next requirement is that the UAV needs to be safe. Ground collisions must be avoided in residential areas. It must also fly low enough to avoid crashing into CPAs and other UAVs. For this reason collision avoidance capabilities are essential. Ground collision avoidance can be improved by maintaining a safe flight height. The GCS can assist by incorporating a warning system when the UAV is close to the ground (Torun, 1999). For avoiding in-air collisions video processing is required.

UAVs — especially mini UAVs — are susceptible to strong winds. To avoid crashing the UAV as a result of sudden weather changes, it is important to monitor the weather from inside the UAV. This allows preventative action to be taken. It is specified in par. 2.5 that one of the features that will probably be expected from large UAVs, is lost link recovery. This is addressed in this study even though the UAV has model status.

#### **2.14. Determining the need for the development of the UAV**

Establishing the shortcomings of the urban security systems and the capabilities of UAVs has shown that urban security systems will benefit greatly from incorporating UAVs into their systems. This is possible since UAV flight regulation is being developed. The development of UAVs has mostly taken place independently of urban security systems. This is because UAVs are developed and built to be independent systems. As a result the need arose to design a UAV that is fully integrated with urban security systems.

The system needs to be as small as possible since it might be necessary to transport it from one location to the next, with as little specialised equipment as possible. It will also help with storing the UAV during inactive periods which makes a mini UAV ideal for urban security applications.

The mini UAV would ultimately need to be handled by operators that might be undereducated (Pavlidis *et al.*, 2001). This means that the system needs to be autonomous for each of the take-off, flight and landing procedures. Autonomous take-off and landing will, however, be left for future study. Thus, changing the flight plan, operating the GCS, and interpreting the information displayed needs to be uncomplicated.

The system will need to be operated during the night, as well. This means that it will require either appropriate night vision or an infrared camera. Infrared cameras are able to locate concealed heat transmitting humans and will be the preferred option if it is within budget. If night vision is chosen the airframe needs to be inaudible and be able to fly high enough to avoid detection. If the UAV is detected the trespasser can hide and avoid detection.

The airframe should also be weather-resistant to allow for all weather operation. This requires a waterproof airframe and the capability of operating in strong winds. The security post of an urban security system should ideally be in the centre of the security site. If this is not the case the UAS needs to have a sufficient range for patrolling the perimeter without being overly expensive.

In this study commercially available products will be used for the autopilot, airframe, GCS, video-streaming and all telemetry. This expedites the development and also saves money. The dimensions and the capabilities of typical UASs have been determined. The airframe needs to have a wingspan of less than 2 m, a telemetry range of at least 1,566 m and be light enough for hand-launch capabilities. Finally, the UAV must be capable of circling the perimeter at least once on a single battery charge.

## **2.15. Conclusion**

In this chapter the state-of-the-art technology applied to mini UAVs was investigated. The most suitable UAV for security applications was identified. Regulations pertaining to UAVs were specified; including regulation that was expected to be enforced in the future. After studying UAVs that are presently available it was also apparent what the typical performance is. This gave a benchmark for products used in the development of the UAV.

Urban security systems were analysed and deficiencies were identified. The role of a mini UAV in the urban security system was also investigated. The need to develop and integrate a UAV into an urban security system was stated. The critical information for developing the UAV is now available and the building process can start.

---

## 3. Assembly and integration

---

---

### Summary

This chapter investigates the different commercially off-the-shelf (COTS) autopilots for building the mini UAV. It also discusses the airframe, camera, motor, battery and telemetry system chosen for the UAV. The building and customising process is presented and different problems experienced are investigated and how they were resolved. Finally, the integration process is discussed and the methods for reading the airspeed and gyrometer sensors are briefly described.

---

### **3.1. Introduction**

This chapter motivates the decision for selecting the different commercially off-the-shelf (COTS) products. Different autopilots were compared to find the best autopilot for the urban security application. Production was expedited by eliminating the need for designing specialised products.

After the products were selected, the UAV building process commenced. This process included steps that were necessary to customise the autopilot for the specific airframe. Problems encountered during the building process are also discussed. The focus of the study then shifted to integrating the UAV into an urban security system.

### **3.2. Choosing the airframe**

The Multiplex Twinstar II airframe was chosen because it is a cost-effective and stable airframe. It consists of a conventional airframe with four-channel control. The airframe is made from ELAPOR<sup>®</sup> which is relatively strong compared to balsa wood. It has a wingspan of 1.42 m with a flying mass of 1.45 kg excluding the autopilot, sensors and photographic equipment (Multiplex, 2011). An image of the airframe is shown in Figure 16.



Figure 16: Multiplex Twinstar II (Multiplex, 2011)

The aircraft is able to carry the necessary payload while still being small enough for easy transport and hand-launch capabilities. It has a good glide ratio of 10:1 (Da Silva *et al.*, 2008). Other airframes can give improved flight stability in strong winds compared to conventional airframes. A conventional airframe can, however, provide better glide ratios and be more stable.

Using a conventional airframe gives the pilot more time to react when there is a component failure, or any other system error since it can fly slowly without stalling. The stall speed on the Multiplex Twinstar II is  $40 \text{ km.h}^{-1}$  (Da Silva *et al.*, 2008).

### 3.3. Motor selection

The main consideration was whether to select a petrol engine or an electric engine. The advantages of each motor type is summarised below:

#### Electric motors

- Low cost
- Non-flammable

- Easy to use
- No emissions
- Low heat
- Easy maintenance
- Low noise

### **Petrol engines**

- Longer endurance than electric motors
- Ability to regulate temperature by adjusting air/fuel mixture
- Refuelling is fast

Depending on manufacturer and distributor, the cost of a petrol engine is about double that of an electric motor. This excludes the cost of the fuel tank, carburettor, fuel pump, engine mountings, etc. These add considerable costs and mass. Using an electric motor only requires an additional battery and electronic speed control (ECS). Electric motors are considered to be cheaper and lighter than petrol engines.

If a component malfunctions and there is a collision with the ground or other aircraft, having fuel onboard can be dangerous. UAVs using petrol engines are bigger and heavier than electric UAVs, which also makes for bigger impacts during crashes. The main advantages of using petrol engines are the endurance and refuelling time. This allows the UAV to be in the air a large percentage of the time. However, modular electric-motored UAVs may save time by merely replacing discharged battery packs with fully charged battery packs. However, recharging battery packs is time consuming.

Operating a UAV using an electric motor requires the user to charge the battery and connect it to the aircraft. Operating a petrol engine requires regular cleaning and maintenance (adding new glow plugs, etc), refuelling and starting the motor. This makes operating an electric UAV easy and cheap. Petrol engines heat up during use, which makes it necessary to mount it on aluminium

engine mountings, since plastic will melt. As a result the airframe for petrol motors is heavier and more expensive. Electric UAVs can be manufactured from polystyrene without any disadvantages.

The UAV for this project will serve as a proof of concept and as a result, development cost and safety are the main concerns. For this reason an electric motor was chosen at the expense of having considerable endurance. A comparison between an electric motor and a petrol engine is shown in Table 2.

**Table 2: Petrol engine and electric motor comparison**

These values are assigned by the author using his own judgement. The values range from zero to one in increments of 0.33.

|                           | Petrol engine | Electric motor |
|---------------------------|---------------|----------------|
| Cost                      | 0.66          | 1              |
| Endurance                 | 1             | 0.33           |
| Safety                    | 0.33          | 1              |
| Ease of use               | 0.33          | 1              |
| Refueling/recharging time | 1             | 0.66           |
|                           | 3.33          | 4              |

The electric motor used for this project is the Pulso X2008/20, the standard brushless out-runner-type supplied with the Mutiplex Twinstar II. The motor is shown in Figure 17 and the specifications are given in Table 3. Two of these motors are used to power the UAV. They might seem underpowered since each of the motors is recommended for an aircraft of 450 g. This recommended model mass is specified for aerobatic flight, whereas this model will only do level flight.



Figure 17: Pulso X2008/20 electric motor

Table 3: Pulso X2008/20 specifications

| Pulso X2008/20 specifications     |     |
|-----------------------------------|-----|
| Internal resistance [ $m\Omega$ ] | 99  |
| No-load current [A]               | 0.9 |
| Mass [g]                          | 45  |
| Recommended model mass [g]        | 450 |
| Electronic speed control [A]      | 20  |

### 3.4. Autopilot selection

One of the main concerns was autopilot selection. The main requirement was that the UAV is able to conduct fully autonomous flights as cheaply, and quickly as possible. There are various COTS autopilots available for application in mini UAVs. They mainly consist of autopilots measuring attitude using IMU sensors or infrared sensors and are listed below (Arnott, 2007; Chao *et al.*, 2007; Bozkurt & Aslan, 2009):

- MicroPilot (MP1028<sup>g</sup>)
- Procerus Technologies (Kestrel<sup>TM</sup>)
- Cloud Cap Technology (Piccolo LT)
- Arduino (ArduPilot Mega)
- LEGO<sup>®</sup> MINDSTORMS<sup>®</sup> (NXT 2.0)

- UNAV (PICOPILOT)
- Athena Technologies Incorporated (GuideStar 111)
- AUAV (EZI-NAV)
- CrossBow (MNAV)
- weControl (wePilot)

Only five of these autopilots were considered for this project and are listed in Table 4. The most important consideration is the cost of the autopilot. Thus, the Kestrel™ and Piccolo LT autopilots are excluded from further considerations. The three autopilots that are considered further are the LEGO® MINDSTORMS® NXT 2.0, the MicroPilot MP1028<sup>g</sup> and the Arduino ArduPilot Mega.

**Table 4: Autopilot initial comparison (Adapted from Arnott, 2007)**

| Brand                 | Model               | Price [R] | Dimensions [mm] | Weight [g] |
|-----------------------|---------------------|-----------|-----------------|------------|
| MicroPilot            | MP1028 <sup>g</sup> | 12,053    | 100 × 40 × 15   | 28         |
| Procerus Technologies | Kestrel™            | 35,000    | 53 × 35 × 12    | 17         |
| Cloud Cap Technology  | Piccolo LT          | 55,267    | 102 × 51 × 20   | 109        |
| Arduino               | ArduPilot Mega      | 1,750     | 80 × 40 × 17    | 45         |
| LEGO® MINDSTORMS®     | NXT 2.0             | 1,750     | 145 × 97 × 61   | 249        |

### **3.4.1. LEGO® MINDSTORMS® NXT 2.0**

The LEGO® MINDSTORMS® NXT 2.0 has a 32-bit microcontroller with 64 kb of random access memory (RAM) and can store the code on 256 kb of memory. It has four inputs of which one is an inter-integrated circuit (I<sup>2</sup>C) and can control the airframe using three independent outputs. It features Bluetooth® communication and the controller comes at a cost of R1,750, which includes touch, light and colour sensors (Bozkurt & Aslan, 2009). The product is shown in Figure 18.



**Figure 18: LEGO® MINDSTORMS® NXT 2.0 with included sensors and servos (Bozkurt & Aslan, 2009)**

### **3.4.2. MicroPilot MP1028<sup>g</sup>**

MicroPilot MP1028<sup>g</sup> autopilot has a low mass of only 28 g. It has eight servos, or relay outputs, and already has all the sensors required for autonomous flight. The autopilot can take off and fly autonomously while still allowing the pilot on the ground to override the control with an RC input (MicroPilot, 2011). The attitude and position information can be logged; and comes at a relatively high cost of R7,350. The MicroPilot MP1028<sup>g</sup> autopilot is shown in Figure 19.



**Figure 19: MicroPilot MP1028<sup>g</sup> autopilot (MicroPilot, 2011)**

### 3.4.3. Arduino ArduPilot Mega

The Arduino ArduPilot Mega can be programmed to have full autonomous flight, including take-off and landing. It can log and display its flights on a GCS. The microprocessor has a clock speed of 16 MHz, RAM of 8 kb and can store the code on 128 kb of flash memory. It has 54 input/output (I/O) pins of which 14 are pulse-width modulation outputs that can control servos. All the sensors necessary for autonomous flight are included and the system costs R1,750 (Arduino, 2011). The Arduino ArduPilot Mega, the microprocessor used for the autopilot, is shown in Figure 20.



Figure 20: Arduino ArduPilot Mega (Arduino, 2011)

### 3.4.4. Selection of Autopilot

LEGO® MINDSTORMS® NXT 2.0 was one of the cheaper autopilots and could be programmed to be fully autonomous. One of the problems with the LEGO® MINDSTORMS® NXT 2.0 was that it is the heaviest of the three autopilots. The autopilot also lacks the sensors required for autonomous flight. Each of these sensors would add additional mass and cost to the project. The code for each sensor would have to be written, which is not impossible but would be time-consuming.

MicroPilot MP1028<sup>g</sup> is a complete autopilot with all the sensors included. It is light, but it is the most expensive option. As a result, it falls outside the project budget. Arduino ArduPilot Mega is the cheapest autopilot that has full autonomous control, take-off and landing capabilities. The software required to program the autopilot is freeware that is constantly updated and has an

excellent support website. It has enough inputs to support all the sensors by connecting the IMU sensor shield. It also has enough outputs to support all the control servos and throttle.

The shield adds an IMU sensor with six degrees of freedom, absolute pressure sensor and a temperature sensor. It has all the necessary control switches and buttons to customise the autopilot. Also, it has connections for the XBee communication module, the airspeed sensor and the GPS. A comparison between the different autopilots is shown in Table 5.

**Table 5: Autopilot feature comparison**

These values are assigned by the author using his own judgement. The values range from zero to one in increments of 0.33.

| Feature                | LEGO® MINDSTORMS® NXT 2.0 | MicroPilot MP1028 <sup>g</sup> | Arduino ArduPilot Mega |
|------------------------|---------------------------|--------------------------------|------------------------|
| Full autonomous flight | 1                         | 1                              | 1                      |
| Take-off and landing   | 0.33                      | 1                              | 0.66                   |
| Affordable price       | 0.66                      | 0                              | 1                      |
| Radio control          | 0.66                      | 1                              | 1                      |
| Flight logging         | 0.33                      | 1                              | 1                      |
| All required sensors   | 0                         | 1                              | 1                      |
| Four-channel control   | 0                         | 1                              | 1                      |
| Small size             | 0                         | 1                              | 1                      |
| Light weight           | 0.33                      | 1                              | 1                      |
| <b>Total</b>           | <b>3.33</b>               | <b>8</b>                       | <b>8.66</b>            |

From this comparison it can clearly be seen that the Arduino ArduPilot Mega scored the highest when the autopilot features were compared. The most important features considered were cost and full autonomy. Here the Arduino ArduPilot Mega scored the highest as well, since the MicroPilot MP1028<sup>g</sup> and LEGO® MINDSTORMS® NXT 2.0 are expensive.

The Arduino ArduPilot Mega was eventually chosen since it included all the sensors required for autonomous flight at a comparatively low cost. The Arduino ArduPilot Mega consists of a proportional-integral-derivative (PID) controller which means that the autopilot will measure the present state and position of the aircraft. The microprocessor will then calculate an error value when compared to the desired direction (Sinnathamby *et al.*, 2010). This error will then be

converted to outputs which will control the servos and throttle. This will allow the UAV to alter its direction and speed. A schematic of a PID position-controller is shown in Figure 21.

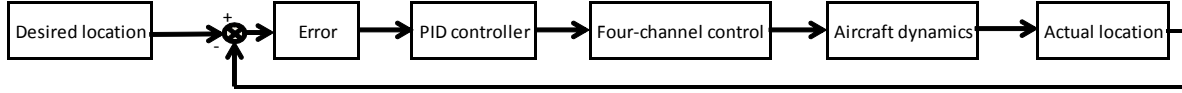


**Figure 21: PID position-controller schematic (adapted from Sinnathamby *et al.*, 2010)**

The P term depends on the present error, the I term on the past errors and the D term on expected future errors. A weighted sum of these values is used to determine the controls to the UAV outputs (Visioli, 2006). The output can be calculated using Equation (1) (Visioli, 2006):

$$u(t) = K_p E(t) + K_i \int_0^t E(\tau) d\tau + K_d \frac{d}{dt} E(t) \quad (1)$$

$K_p$ ,  $K_i$  and  $K_d$  are the gain constants of the proportional, integral and derivative controls, respectively. By tuning these values the desired response of the PID loop can be achieved (Visioli, 2006). An autopilot using a PID controller was chosen because it is simple to understand, requires little processing resources and can be extended with other control algorithms. The only disadvantage is that tuning the parameters requires skill and is time consuming (Chao *et al.*, 2007).

$K_p$  in the PID loop sets the rate at which the output tends to the required output. This is shown in Figure 22.  $K_i$  in the PID loop accelerates the rate at which the output tends to the required output. This is shown in Figure 23. And finally,  $K_d$  in the PID loop limits the change in the output value. This is shown in Figure 24.

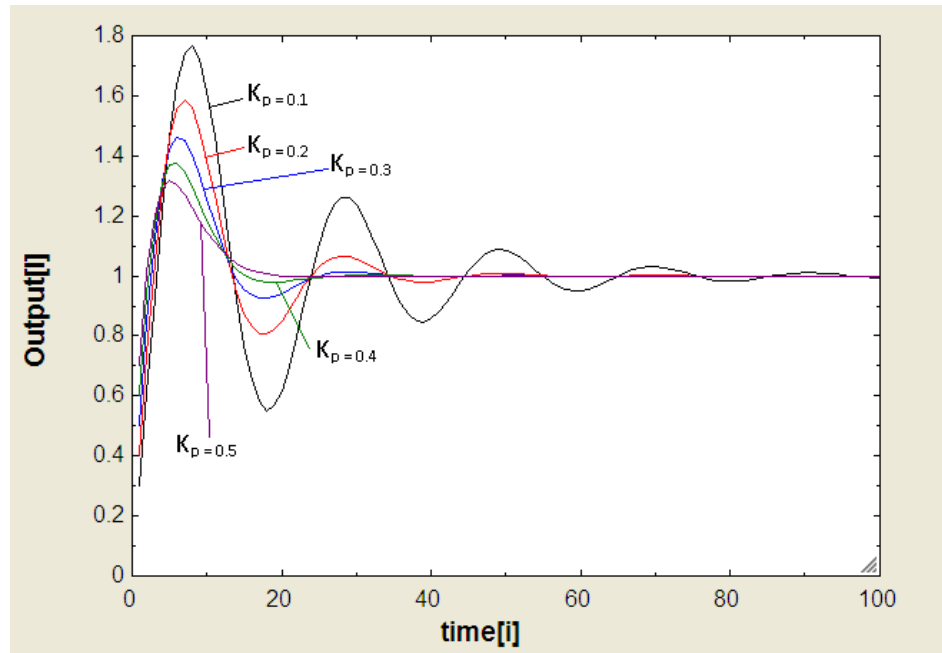


Figure 22: Effect of gain  $K_p$  on the PID output with the other gains at 0.1, based on a step input (Adapted from Franklin *et al.*, 1986).

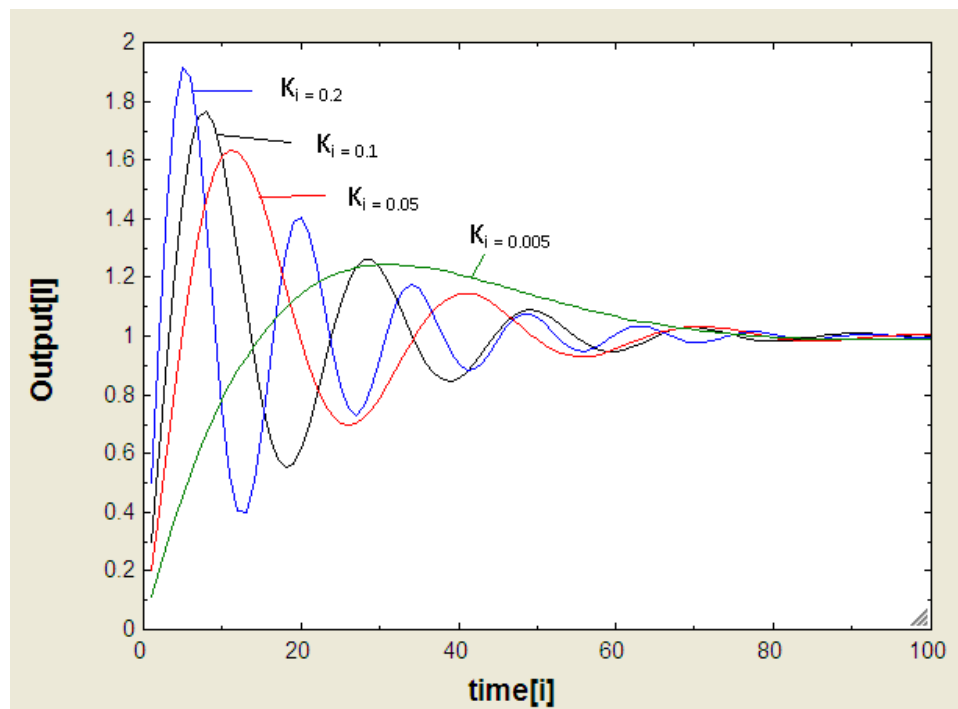


Figure 23: Effect of gain  $K_i$  on the PID output with the other gains at 0.1, based on a step input (Adapted from Franklin *et al.*, 1986).

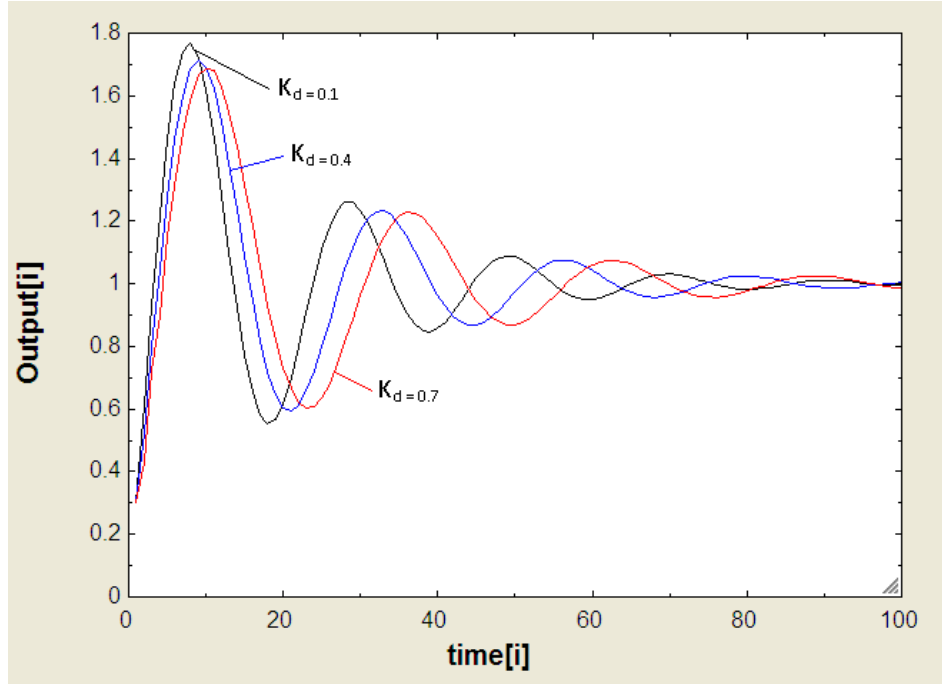


Figure 24: Effect of gain  $K_d$  on the PID output with the other gains at 0.1, based on a step input (Adapted from Franklin *et al.*, 1986).

These PID gains can be tuned using trial and error or using the Ziegler-Nichols parameter tuning method. This is done by setting  $K_i$  and  $K_d$  to zero and adjusting  $K_p$  until the output varies at a constant altitude. This value of  $K_p$  is known as  $K_u$ . The period in which a cycle is repeated is known as  $T_u$  (Mallesham *et al.* 2009). Using these values the gain of  $K_p$ ,  $K_i$  and  $K_d$  can be determined using Table 6.

Table 6: Ziegler-Nichols PID tuning method (Mallesham *et al.* 2009)

| Controller | $K_p$      | $K_i$         | $K_d$   |
|------------|------------|---------------|---------|
| P          | $0.5 K_u$  | -             | -       |
| PI         | $0.45 K_u$ | $1.2 K_p/T_u$ |         |
| PID        | $0.6K_u$   | $2/T_u$       | $T_u/8$ |

### 3.5. Camera selection

Cameras can either be stationary, or be able to pan and tilt (Schwager *et al.*, 2011). Gimballed cameras can be acquired, or two servos can be used on a stationary camera to enable it to pan and tilt. Controlling the camera with two servos would be the cheaper option. Programming the cameras to pan and tilt with the UAV attitude would require too much time. As a result a stationary camera was used.

The KX191 camera, which has a resolution of  $512 \times 582$  pixels, was selected. It weighs 46 g and has a size of  $35 \times 35$  mm. The lens on the camera is a Sony Super hole accumulation diode (HAD) charge-coupled device (CCD). It is sensitive to near-infrared light and allows it to record video in low light and night-time situations.

The camera automatically switches from normal video to night vision which makes monitoring the video feed easier for the operator. It has all the necessary capabilities and comes at a low cost of R735. A picture of the camera is shown in Figure 25. The camera is mounted underneath the fuselage as shown in Figure 26.



Figure 25: KX191 camera (Rangevideo, 2011)

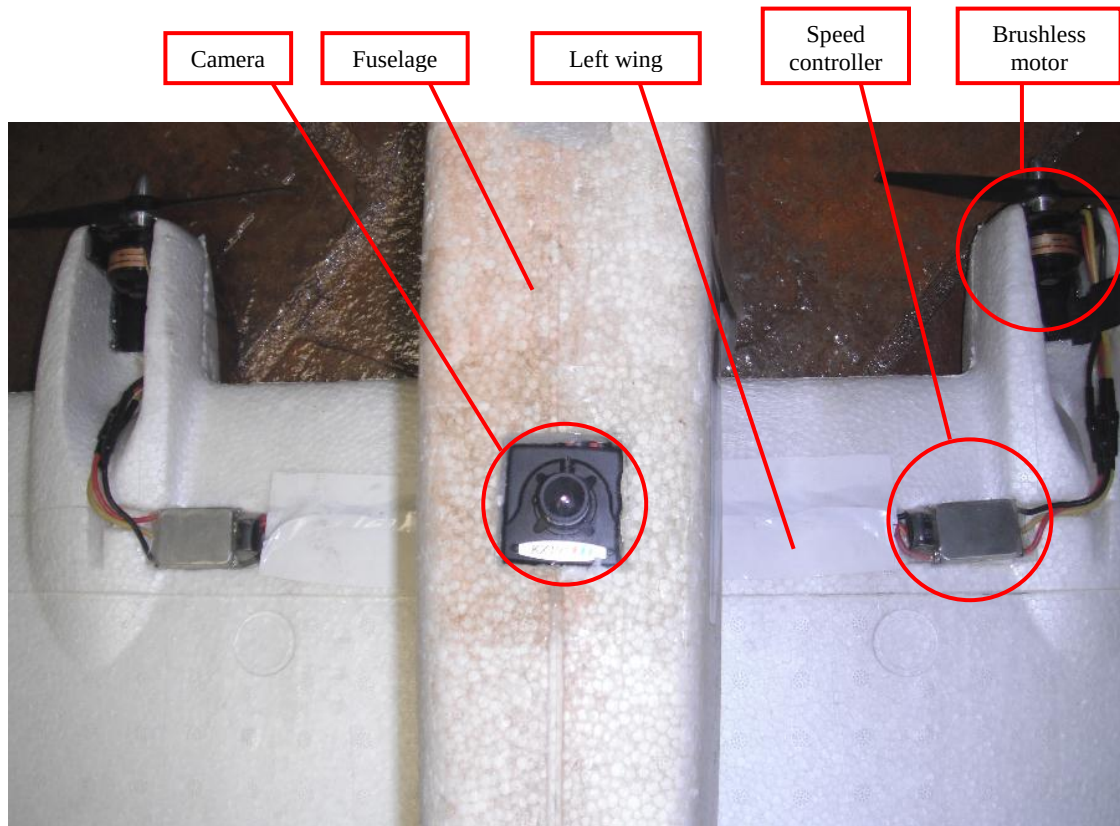


**Figure 26: Camera mounted beneath the UAV**

### **3.6. Telemetry equipment**

For RC the Spektrum DX7 was chosen since it was determined that the autopilot would require at least six channels (DIY Drones, 2011). Four of the channels are necessary for RC, while the other two channels are used to switch between modes. The Spektrum DSM2 receiver and RC communicate at a frequency of 2.4 GHz and have a range of 3 km. This was determined to be sufficient since the RCs will only be used during take-off, landing and the testing phases. The Spektrum DX7 is shown in Figure 27.



**Figure 27: Spektrum RC package**

The autopilot microcontroller needed to communicate with the GCS and a telemetry range of 1,566 m was determined to be sufficient. This is below the typical range for telemetry systems communicating at a frequency of 900 MHz (DIY Drones, 2011). The XBee 900 telemetry modules were chosen, which fulfilled all the specified requirements. The XBee modules are shown in Figure 28.



**Figure 28: XBee modules before assembly**

Some telemetry link systems for autopilots allow the video-streaming to be transmitted through the same modules. The XBee telemetry system could, however, not stream autopilot telemetry and video simultaneously. The video-streaming could also not be done at the same frequency as the autopilot telemetry due to interference between two signals at the same frequency. As a result the video is streamed at 1.3 GHz.

The range for the video-streaming is expected to be greater than 2 km but this will need to be tested. The video telemetry system is shown in Figure 29 and Figure 30.

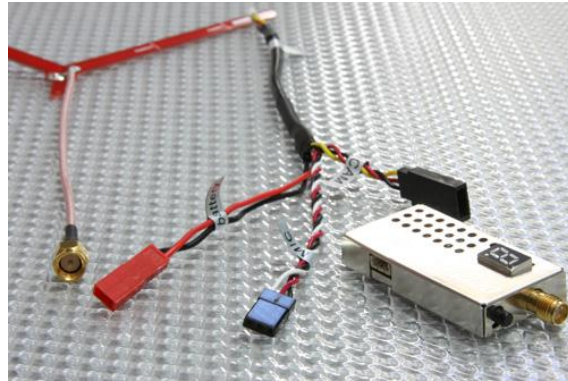


**Figure 29: Video transmitter (Rangevideo, 2011)**



**Figure 30: Video receiver (Rangevideo, 2011)**

### **3.7. Battery selection**

Lithium-polymer (LiPo) batteries have the highest energy density of presently available small, rechargeable batteries. As a result they are preferred for use in mini UAVs where mass, endurance and size are the main considerations. Mass production has also reduced the price of LiPo batteries, making them more affordable (Cosyn & Vierendeels, 2007). The battery that was

chosen for this project is a XPower 2.2 Ah, 11.1 V battery. It weighs 175 g and an image of the battery is shown in Figure 31.



Figure 31: XPower LiPo battery

### 3.8. Assembling the autopilot

The first step in assembling the autopilot was to solder the headers onto the autopilot microcontroller and the IMU shield. This allowed the shield to be connected to the microcontroller so that all the sensors could be read for processing (DIY Drones, 2011). The headers on the IMU shield and the microprocessor are shown in Figure 32 and Figure 33. This allows the two to be connected as is shown in Figure 34.

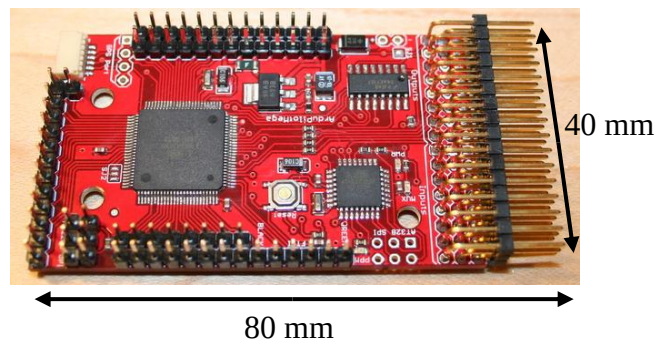


Figure 32: Autopilot with headers (DIY Drones, 2011)

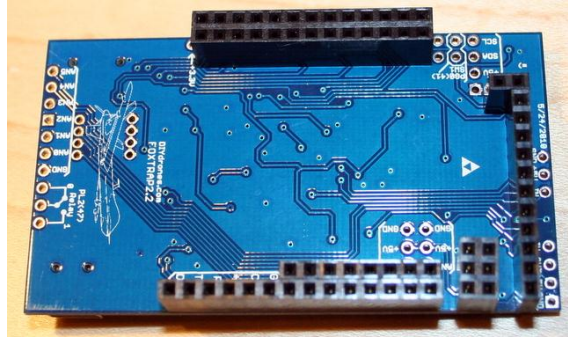


**Figure 33: IMU shield with headers (DIY Drones, 2011)**

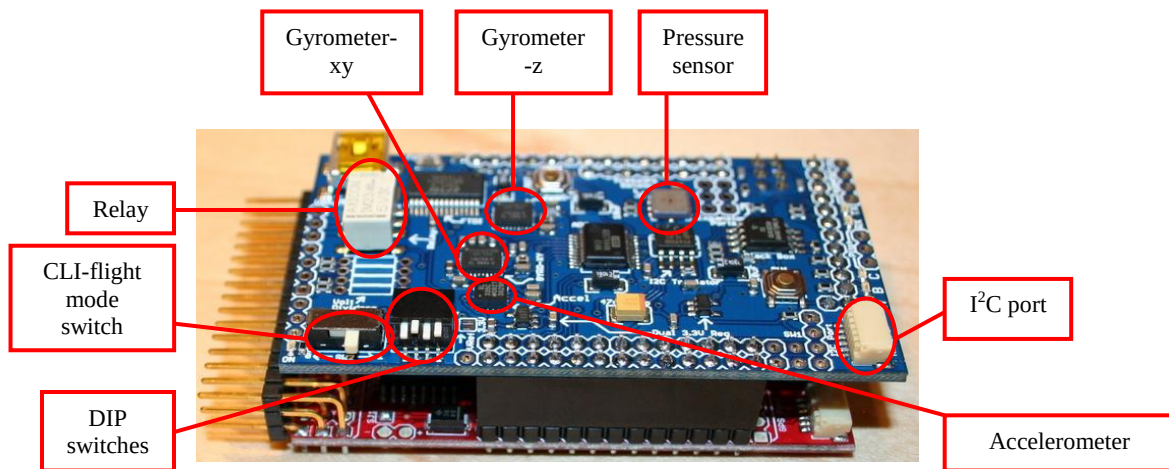


**Figure 34: Fully assembled autopilot (DIY Drones, 2011)**

After the autopilot and IMU shield were connected, the remaining sensors and RC equipment were connected. The sensors are connected as shown in Figure 35.

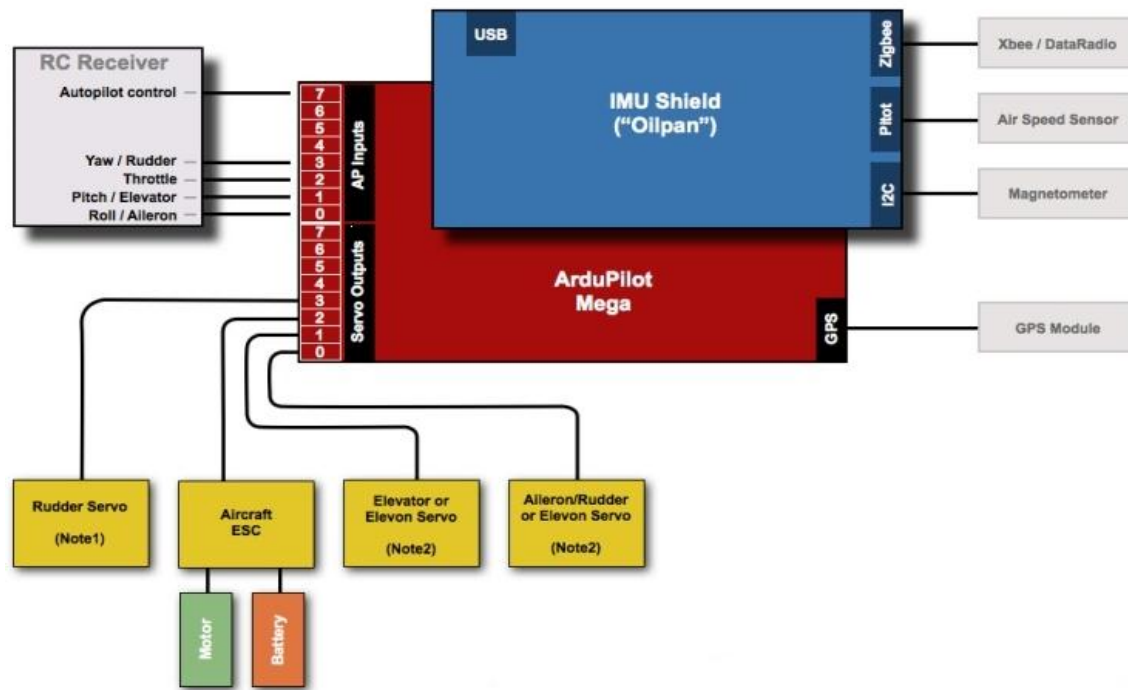


Figure 35: Autopilot connections (DIY Drones, 2011)

All the manual RCs are relayed through the autopilot to allow the RC to override the autopilot control. The XBee modules formed the wireless system that allowed the autopilot to communicate with the GCS. There was no magnetometer connected since it could be assumed that the aircraft is pointing in the direction of travel in a fixed-wing airframe.

The airspeed sensor was connected to the IMU shield which allowed the autopilot to control the flight speed according to the airspeed (DIY Drones, 2011). The airspeed sensor was installed facing forward on the front end of fuselage. It was installed 3 cm from the body to ensure that the boundary layer of the air flowing over the airframe did not interfere with the airspeed reading.

The microcontroller was programmed through the Arduino integrated development environment (IDE). It is a Java-based language and can be extended with C++ libraries (Bin & Justice, 2009). The code is freeware and can be downloaded from the DIY Drones website and then uploaded to the Arduino ArduPilot Mega microcontroller. It is loaded onto the autopilot via a miniature

universal serial bus (USB) cable. It is updated constantly by freelance programmers and users (Bin & Justice, 2009).

After the code was loaded, the sensors were tested using the Command Line Interpreter (CLI) that forms part of the Arduino IDE. It is essential that each sensor operates properly and is transmitting the correct information. The sensors that were tested were the IMU, airspeed, temperature, GPS and absolute pressure sensors. These essential parameters are validated in the following chapter.

After testing the sensors the RC was set up by entering the radio setup mode in the CLI and moving all the switches to their respective extremes. The flight mode setup was then done. The different modes available are “manual”, “stabilise”, “fly-by-wire”, “return-to-launch”, “autopilot” and “loiter”. The other modes that can be customised are the take-off and landing modes (DIY Drones, 2011). The flight modes are described below:

- **Manual** — The pilot has full control of the UAV.
- **Stabilise** — The pilot has full control of the UAV, but when the UAV does not get any input from the pilot it levels itself.
- **Fly-by-wire** — The pilot has full control but the UAV limits the roll and pitch angles to preset values.
- **Return-to-launch** — In this mode the autopilot has full control of the UAV and navigates its way to the launch position.
- **Autopilot** — The autopilot has full control of the flight direction and speed while navigating between the waypoints.
- **Loiter** — In this mode the autopilot has full control and the UAV will circle in its current position.

The autopilot was then connected to the GCS by connecting the two XBee modules to the autopilot and a computer — functioning as the GCS — via USB. The code was then customised

to exchange data through the modules rather than the USB port. This was done by adding the following lines in the configuration file of the code:

- `#define SERIAL3_BAUD 57600`
- `#define GCS_PORT 3`

The code defined the telemetry port of the modules and set the baud rate to 57,600. The autopilot built into the airframe can be seen in Figure 36. Additional code that needed to be changed for this specific project is listed below:

- `#define GCS_PROTOCOL` `GCS_PROTOCOL_MAVLINK`
- `#define GCS_PORT` `3`
- `#define MAV_SYSTEM_ID` `1`
- `#define GPS_PROTOCOL` `GPS_PROTOCOL_UBLOX`
- `#define AIRSPEED_SENSOR` `ENABLED`
- `#define THROTTLE_MIN` `35`
- `#define THROTTLE_CRUISE` `52`
- `#define THROTTLE_MAX` `62`
- `#define AOA` `50`
- `#define AIRSPEED_CRUISE` `13`
- `#define NAV_ROLL_P` `0.35`
- `#define XTRACK_GAIN` `0.01`
- `#define XTRACK_ENTRY_ANGLE` `15`
- `#define ARSPD_OFFSET` `0`

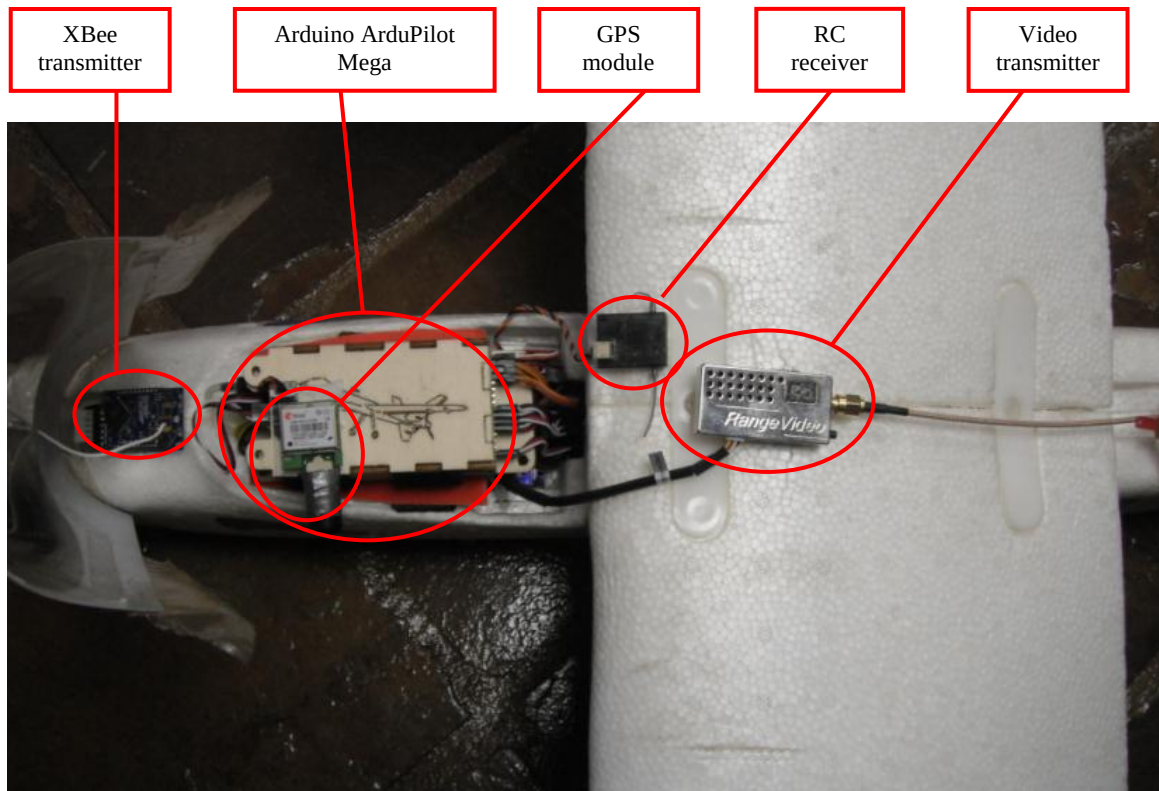


**Figure 36: Autopilot built into the airframe**

The freeware GCS was downloaded and used to communicate with the autopilot. The attitude and position was then relayed wirelessly by the UAV and displayed on the GCS. The HappyKillmore GCS was also used to upload waypoints to the electrically erasable programmable read-only memory (EEPROM) of the microcontroller. Waypoints were written using point-and-click on Google Earth™. The flight path was also overlaid on Google Earth™ as the mini UAV flew (DIY Drones, 2011). A screenshot of the HappyKillmore GCS is shown in Figure 37.

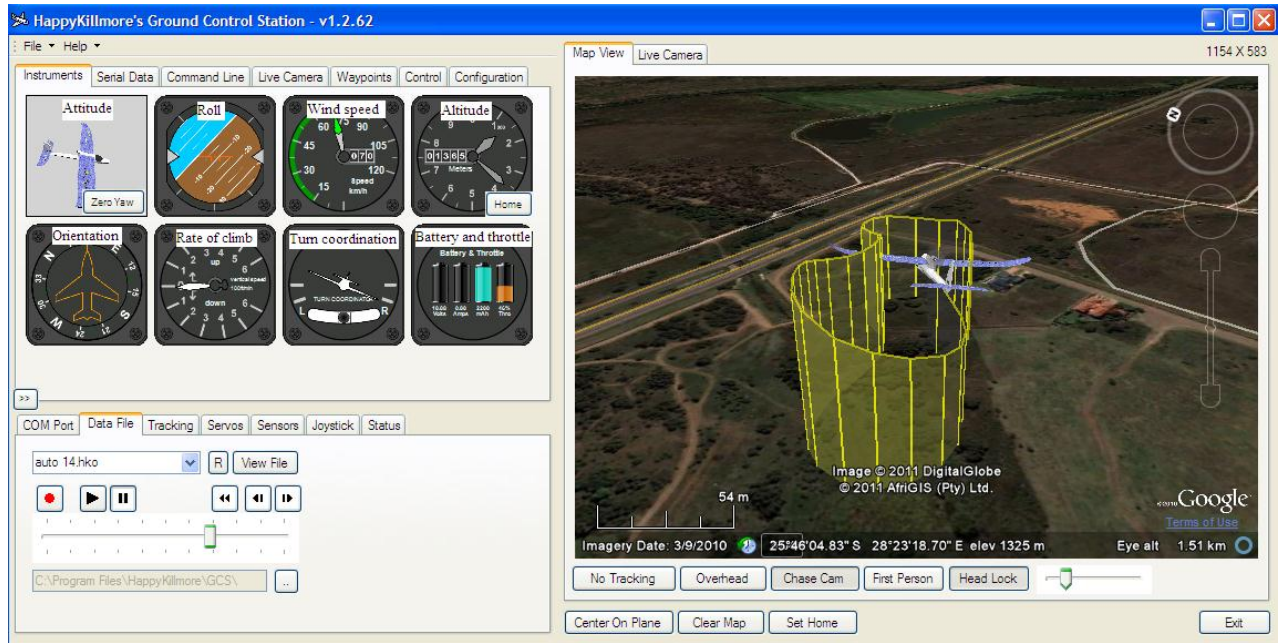


Figure 37: HappyKillmore ground control station with Google Earth™ flight path

In this figure the HappyKillmore GCS can be seen. The labelled instrument panels can be seen in the top-left corner. In the bottom-left corner flight data can be stored and loaded for reviewing. The flight path can be seen on the right. The top of the two-dimensional flight path is the height of the mini UAV while the bottom is the position on Google Earth™. The UAV model used in the GCS is that of a Multiplex Easy Star since it is the only model available.

### 3.9. Sensor measurement

#### 3.9.1. Airspeed sensor

Pitot tubes are required to measure airspeed. This is done by taking a measurement of the static pressure and the total pressure of the air during flight. The difference between these two measurements is the dynamic pressure. This dynamic pressure can be converted to airspeed if the air density is known by applying Equation (2). The density is calculated using the temperature and pressure of the air. This calculation is shown in Equation (3).

$$p_d = \frac{1}{2} \rho v^2 \quad (2)$$

$$\rho = p/RT \quad (3)$$

These calculations are done onboard the microprocessor of Arduino ArduPilot Mega. The differential pressure sensor used is the MPXV7002DP. This sensor can measure pressure differences of  $\pm 2$  kPa. It requires a constant supply of 5 V and has a sensitivity of  $1 \text{ V.kPa}^{-1}$ . The output of this sensor is a voltage ranging from 0.5 V to 4.5 V depending on the pressure difference. This voltage reading is then converted to airspeed. The measured output of this sensor is shown in Figure 38.

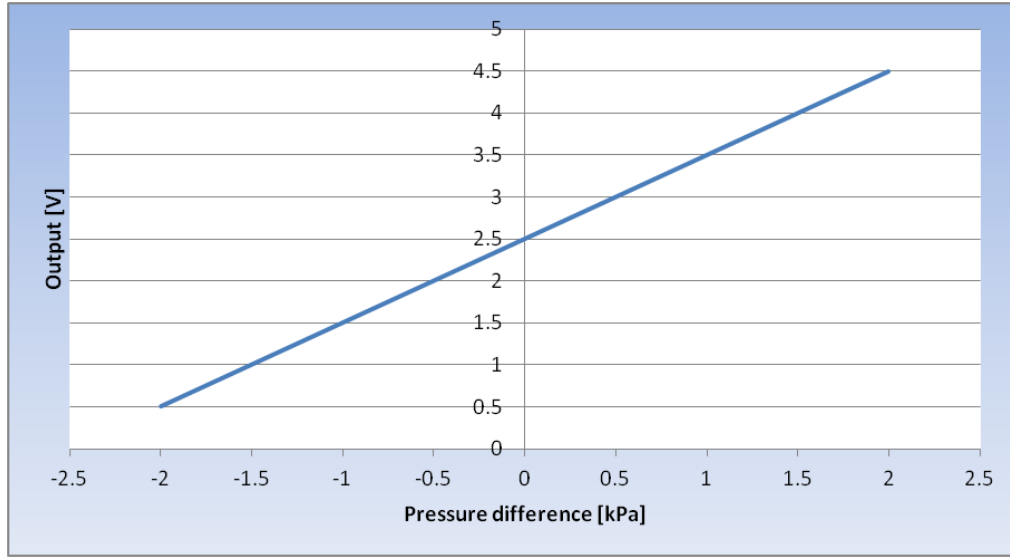


Figure 38: Airspeed sensor output

### 3.9.2. Gyrometer sensor

The gyrometer measures the rate of rotation (angular velocity) around the three axes. This angular velocity can be converted to a change in angle as shown in equation (4). The gyrometer requires a 3 V supply and has a sensitivity of  $2 \text{ mV.}(\text{°.s})^{-1}$ . When calibrating this unit, the gyrometer assumes that it is level and any change in angle will be added to give present orientation. If there is no change in angle the gyrometer output is 1.35 V and increases or decreases with a change in rotation. The output is shown in Figure 39.

$$\theta = \omega t \quad (4)$$

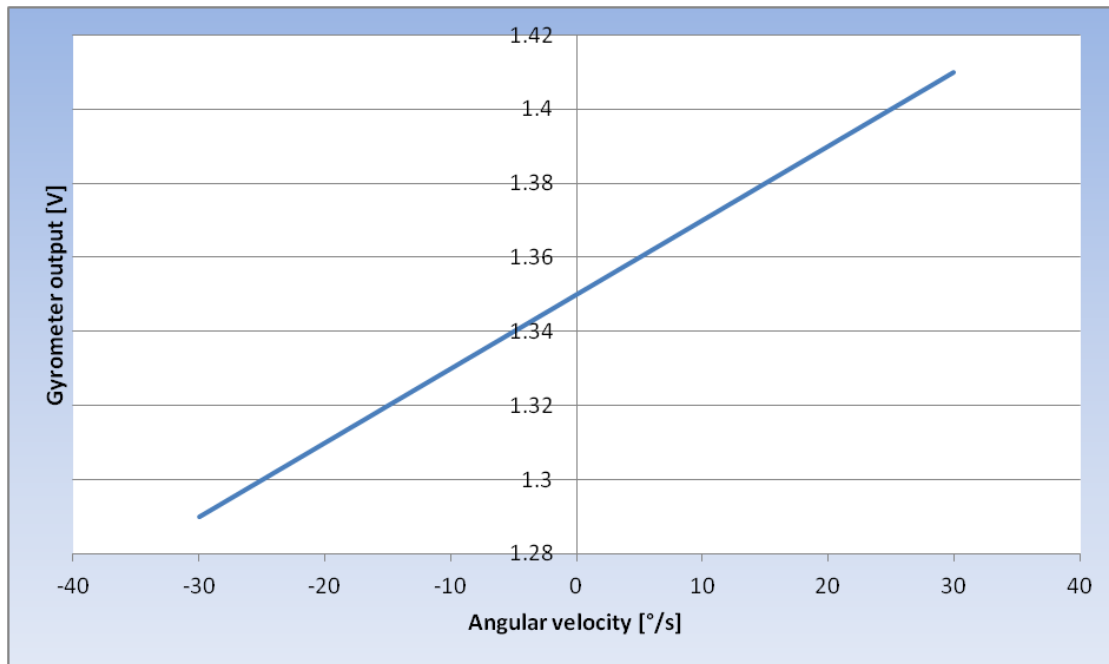


Figure 39: Gyrometer output

### 3.10. Problems experienced

#### 3.10.1. Autopilot resetting

While the autopilot is in manual mode the pilot has full control. If the autopilot is switched into any other mode the GCS reports an autopilot telemetry link loss. As a result no information is available on the GCS and the UAV flies independently. The pilot is able to switch into manual mode again, but the UAV does not report linking with the UAV again.

It was originally thought to be a problem with the baud rate or package size used to communicate with the UAV. The information that needs to be sent during manual mode is less than the information sent during autopilot mode. If the baud rate is set too low the GCS would report no telemetry in autopilot. The original baud rate was set to 38,400 and re-adjusted to 57,600. This did not solve the problem and further research was done.

While testing the autopilot on the ground it was noted that the autopilot resets itself when it was switched to any other mode than manual. The problem was the ESC that was not supplying enough current for the autopilot through its battery eliminator circuit (BEC) causing the telemetry link loss.

The autopilot required 15 s to recalibrate all the sensors during start-up. Thus, if it stayed in autopilot mode it would probably have collided before being able to complete the calibration cycle. The UAV was always able to switch back to manual mode. This allowed the pilot to resume manual control and avoid collisions.

The speed control was switched from the standard 20 A to a 40 A speed control. Changing the speed control allowed the ESC to supply adequate power for both the autopilot and brushless out-runner motors. This kept the autopilot from resetting itself.

### **3.10.2. Maintaining airspeed in level flight**

With Arduino ArduPilot Mega the throttle is adjusted to maintain a constant airspeed. The user must specify an upper-, lower- and cruise throttle setting, as well as the required cruising airspeed ( $13 \text{ m.s}^{-1}$  for this project). The autopilot would then regulate the airspeed by applying the maximum specified throttle percentage (62% for this project) until the mini UAV reaches the required airspeed. It would then lower the airspeed by setting the throttle to the cruise throttle position (52% for this project). If the speed exceeds the required airspeed it would lower the throttle setting to a minimum, namely the lower throttle setting (35% for this project).

Some autopilots regulate the throttle by using a climb-, cruise- and descend setting. Using speed feedback allows the UAV to maintain its airspeed more accurately than when compared to the three different throttle settings (Ahmed & Samar, 2007). An airframe with a low stalling speed was essential when the throttle is adjusted too low by the autopilot.

The UAV was flown in manual mode to determine the throttle settings. The throttle percentage required to maintain a steady flight speed was then noted by the flight operator. The cruise

throttle was set to this percentage. The maximum throttle was adjusted to 10% above the cruise throttle setting which prevented the autopilot from setting the throttle too high.

### **3.10.3. Maintaining height**

The UAV lost height during the test flight. It followed the waypoints successively, but went through each waypoint at a lower height. It was originally thought to be an electronic navigation gain problem. Adjusting the navigation gain on the elevator servo only reduced the rate of descend over each waypoint.

The first problem was that the waypoint altitude was specified above mean sea level (AMSL). Thus, it was tracking the waypoints at a height of 50 m above sea level rather than 50 m above ground altitude. This was corrected by downloading another GCS called ArduPilot Mega Planner. It allowed the default altitude to be set at a height above ground altitude. This setting was retained when using HappyKillmore GCS.

Another problem was that the UAV was not supplying enough up-elevator deflection during the turns which caused the UAV to lose height while turning. This was corrected by adjusting the pitch compensation value which adjusts the amount of up-elevator that aids in maintaining a constant altitude turn. The pitch compensation value is the proportional gain of the PID loop that coordinates the turns.

### **3.11. Discussion of final product**

When considering the basic requirements set for urban use of the UAV it can be seen that the UAV fulfils most of the requirements. The specifications for the final product are compared to the commercial UAVs in Table 7. The cost breakdown for developing the UAV is given in Appendix A.

**Table 7: UAV comparison**

|                   | Carolo P50 | CyberBug Micro | Kiwit | Project UAV |
|-------------------|------------|----------------|-------|-------------|
| Endurance [min]   | 30         | 40             | 60    | 16          |
| Range [km]        | *          | 5              | 5     | 2.065       |
| Wingspan [m]      | 0.5        | 0.76           | 1.8   | 1.4         |
| Speed [km/h]      | 74         | *              | 50    | 60          |
| Altitude [m]      | 457        | 3050           | *     | 110         |
| Mass [kg]         | *          | 1.18           | 4     | 2.3         |
| Cost [R]          | *          | 272,300        | *     | 11,141      |
| * = not specified |            |                |       |             |

This then shows that the final product is comparable with the mini UAVs listed in the second chapter. The endurance, altitude and range are less than the reference UAVs. This is, however, justifiable since the cost of developing the UAV is significantly less than the other UAVs, because COTS products are used for the project.

The UAV follows waypoints while maintaining the preset height. It also streams back all the necessary information for effective control, navigation and battery monitoring. The video-streaming allows the operator to monitor the perimeter during night- and daytime flights. The autopilot was now built and the requirements for safely operating the UAV in the NAS identified and integration of the autopilot and the urban security system could commence. The final UAV is shown in Figure 40.

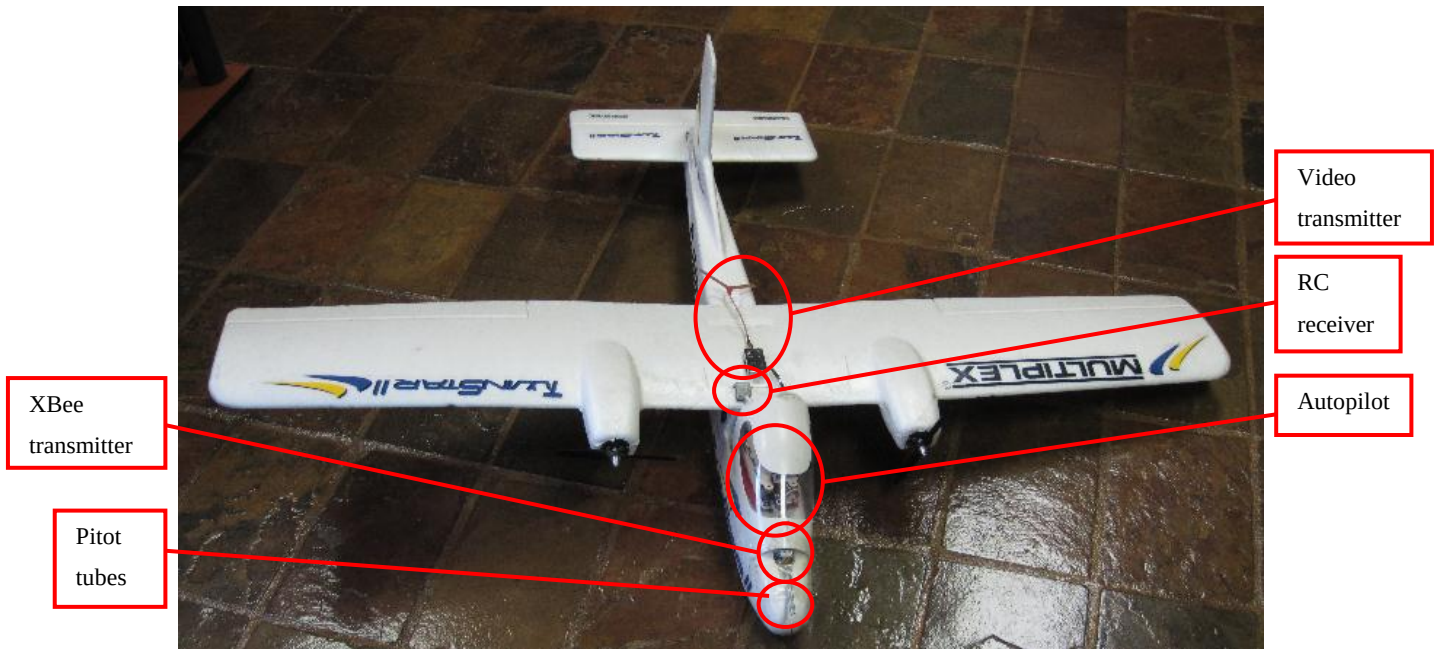


Figure 40: Final mini UAV

### 3.12. Integration into an urban security system

The specific requirements for operation of UAVs in an urban environment are:

- Hand-launch capabilities.
- Video-streaming.
- Coordinates of the centre of the image.
- It must be hard to detect — inaudible and invisible to the naked eye.
- All the information needs to be shown on one screen.
- The UAV in this study needs enough endurance to circle Silver Lakes Golf Estate at least once.
- It needs to have a telemetry range greater than 1,566 m.
- The flight path must be overlain on Google Earth™.
- The information must be secure.
- Collision avoidance capabilities.
- Weather monitoring.

- Lost link recovery system.
- Low flight height warning system.

A Microsoft Excel<sup>®</sup> model was developed to calculate the coordinates of the centre of the image. This was possible since the camera was mounted stationary. If the pitch angle, roll angle, yaw orientation, height and GPS coordinates are known, the coordinates of the video feed can be determined. The pitch adjustment was calculated using Equation (5) and the roll adjustment was calculated using Equation (6). This situation is shown in Figure 41.

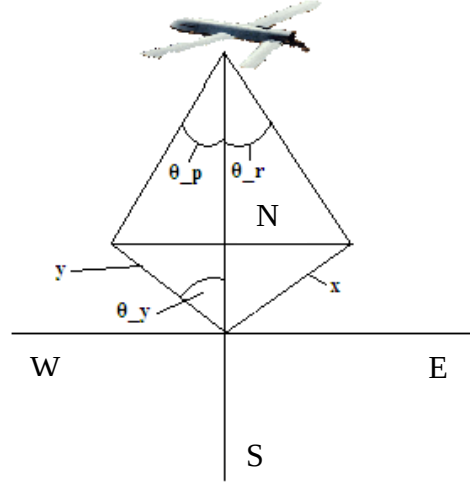


**Figure 41: Video position GPS correction**

$$y = H \sin \theta_p \quad (5)$$

$$x = H \sin \theta_r \quad (6)$$

With the yaw orientation known, the component of the pitch and roll adjustment in the north, south, east and west directions can be determined. The components were determined using Equations (7) and (8).

$$NS = y \cos \theta_y + x \sin \theta_y \quad (7)$$

$$EW = x \cos \theta_y + y \sin \theta_y \quad (8)$$

The UAV GPS coordinates are known which means that by adding the components calculated in Equation (7) and (8) to the GPS coordinates the centre of the video feed could be calculated. The Microsoft Excel<sup>®</sup> model is shown in Table 8.

**Table 8: GPS coordinate correction**

|                  |          |     |                          |
|------------------|----------|-----|--------------------------|
| Pitch            | 20       | [°] | Negative if down         |
| Roll             | 10       | [°] | Negative if rolling left |
| Yaw              | 30       | [°] | Clockwise from North     |
| Height           | 100      | [m] | From home altitude       |
| Latitude (lat)   | 25.46119 | [°] | Negative if south        |
| Longitude (long) | 28.21812 | [°] | Negative if west         |
| Video Lat        | 25.46153 | [°] |                          |
| Video Long       | 28.21841 | [°] |                          |

The height the UAV can still be seen at was determined since the UAV needs to fly in line of sight to retain model status. The endurance and range of the autopilot was also tested. HappyKillmore GCS can overlay the UAV flight path on Google Earth<sup>™</sup>. This allows the operator to focus his attention on the video feed and flight path. The results of the requirement testing are given in Chapter 4.

Making the information secure would have increased the development costs significantly because military-grade telemetry systems would be required. By repeatedly testing the UAV, safety could be established. This was done by completing 41 test flights. Some of these flights were used to provide test data which is given in the following chapter.

The other safety concern is airborne or ground collision avoidance. Ground collisions can be avoided by maintaining a specified minimum height above ground level. This UAS unfortunately

only has the ability to maintain a preset height above the initial launch altitude. The problem with maintaining an altitude above the initial launch altitude is the fact that protrusions above the initial altitude can cause the UAV to crash into the ground. This situation is shown in Figure 42. The flight elevation above the launch altitude is 25 m. The change in *en route* altitude is however larger than 25 m which will cause the UAV to collide with the ground.

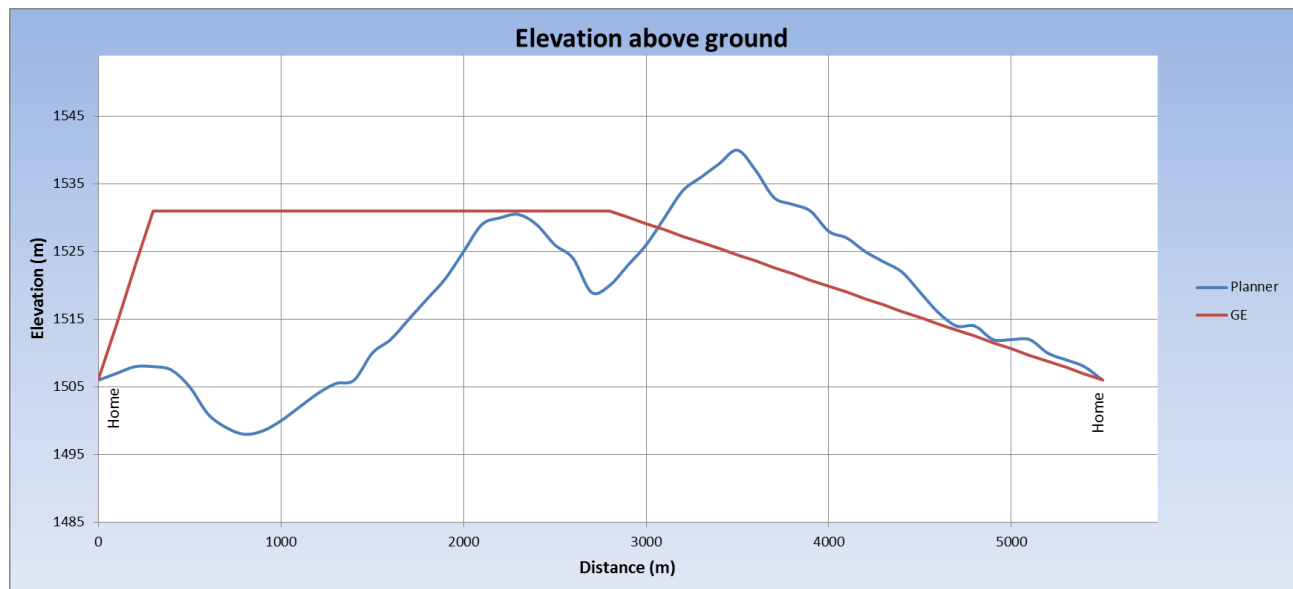


**Figure 42: Elevation graph from ArduPilot Mega Planner.**  
GE is the ground elevation.

When doing test flights the flight route must be plotted on the GCS and the elevation of the *en route* ground surface must be retrieved from Google Earth™. The waypoint heights must then be set to ensure that the UAV flies high enough to avoid crashing. From the ground elevation graph above it can be seen that the maximum ground elevation AMSL is 1,540 m while the launch altitude is 1,506 m AMSL. This is a difference of 34 m. For a minimum flight height of 50 m it is then required to set the flight height to 84 m to compensate for changes in the ground elevation.

When the UAV loses telemetry link the operator is unable to control the UAV. This is a concern, since it can be the cause of crashes when large urban areas are monitored. Although the UAV does not have this capability, an alternative solution was found. When the UAV loses telemetry

link the UAV will automatically switch to return-to-launch mode. The UAV will fly to the home location where the operator will be able to revert to manual control so as to land the UAV and then to reconnect to the UAV.

### **3.13. Conclusion**

In this chapter the COTS autopilot selection process was discussed. The specifications of the other products were also given. The UAV was then assembled and customised. Some of the problems while assembling the UAV were discussed and the solutions were found. With the UAV built, the integration process was initiated. With the UAS integrated into the urban security system the testing procedure could begin.

---

## 4. Verification and validation

---

---

### Summary

In this chapter the UAV and the urban security system requirements are tested to verify expected results. A specific flight-test procedure is developed to minimize the possibility of accidents. For verification of the UAV capabilities the autonomous flight, navigation accuracy and height control accuracy, were tested. Urban security system verification included range, endurance, maximum height, hand-launch capability, video feed and GCS evaluation. The chapter concludes with validation of the data that was recorded with the GCS.

---

## 4.1. Introduction

The UAV capabilities were tested to ensure that it fulfilled all the requirements of autonomous flight. First the flight-testing procedure will be discussed which included ground and air testing. The first verification test was done to determine whether the autopilot prevented the UAV from exceeding preset roll and pitch angles. Next the autopilot function was tested to determine how accurately waypoints were tracked. The UAV's ability to maintain the preset height within acceptable limits was also tested as part of UAV verification.

The integration with the urban security system was verified. This phase started by testing whether the UAV could be hand-launched in a close confinement. The video-feed function, maximum flight height and the functionality of the GCS were all tested. The range and endurance of the UAV were determined. The data relating to the position and attitude was then validated to establish that the measurements were correct.

## 4.2. Flight-testing procedure

### 4.2.1. Ground testing

The autopilot calibrates its gyrometer, accelerometer and onboard pressure sensor each time it powers up. During the calibration procedure these sensor data are measured. The gyrometer and accelerometer position are assumed to be the neutral UAV attitude. The onboard pressure sensor reading is assumed to be that of the launching pressure, which is converted to a launching altitude using Equation (9) and sea level as reference height. The GPS coordinates are measured and assumed to be the initial launching coordinates.

$$\Delta p = -\rho g \Delta H \quad (9)$$

The calibration of the sensors takes approximately 15 s. During calibration the UAV must remain stationary. The operator must ensure that the autopilot is level during power-up. This was

done by powering the UAV on a horizontal, level surface and ensuring that the wings are level. This was done because the autopilot would maintain the UAV's calibration attitude while flying.

After calibrating the autopilot, it was set to manual and procedures followed to ensure that the RC had full control of the UAV. The UAV was set to stabilise mode and the airframe was rolled from side to side. It was ensured that the autopilot moved the ailerons in the right direction to counteract the roll. If the movement of the ailerons was in the wrong direction, the corresponding dual in-line package (DIP) switch was reversed. A similar procedure was followed to ensure the elevator control was in the correct direction by pitching UAV.

#### **4.2.2. First flight**

All flight tests were conducted inside Silver Lakes Golf Estate in order to obtain realistic flight results in an urban environment. The first flight was completed using manual, stabilise and fly-by-wire modes. The UAV was launched in manual mode and piloted to a height of 80 m. It was then switched to stabilise mode while still giving manual inputs. All the controls were in neutral position to ensure that the UAV levelled itself. In manual mode and at 52% throttle the 2.2 Ah batteries allowed 14 minutes endurance.

The procedure was repeated three times before switching over to fly-by-wire mode. The UAV was then given manual controls to ensure that it did not roll or pitch past the preset boundaries. This tested the UAV's ability to level itself and correct its flying attitude after manual inputs had been given. It also helped to determine what the required pitch compensation value is that allows for constant height turns.

#### **4.2.3. Second flight**

The next step was to set a home point using the GCS. The UAV was initiated in manual mode and piloted to a height of 70 m above the home point altitude. GPS lock was confirmed and when the UAV was 100 m from the launch coordinates it was switched to return-to-launch mode. The UAV successfully navigated its way towards the launch point.

When it reached the home location the UAV circled the waypoint while still maintaining the required height. This confirmed the autopilot's ability to navigate to a GPS coordinate — that was set by the GPS — while maintaining a preset height. This flight lasted only five minutes.

#### **4.2.4. Third flight**

The third flight was used to test the autopilot mode. Waypoints were uploaded to the UAV from the GCS. The UAV was then launched in manual mode and piloted to the height of the waypoints. The UAV was then switched to autopilot mode and allowed to navigate the waypoints while maintaining a constant altitude of 70 m above the launch altitude. The lateral distances between the waypoints were approximately 100 m. This process was repeated until the autopilot flight was correct.

If the UAV was losing height during turns the pitch compensation value was adjusted. If the UAV was snaking the navigation gain was lowered and if it was not following the desired waypoints the navigation gain was increased. Each of the flights lasted approximately eight minutes.

A checklist was created and reviewed to ensure that everything had been attended to, and included before each flight. The flight checklist used for flying the mini UAV is shown in Table 9. Using the checklist helped avoid unnecessary accidents since failing to check servo control direction is the number one cause of accidents (DIY Drones, 2011).

The waypoints for testing the autopilot were in LOS and no further than the RC's range. Some error margin was allowed since it is not always immediately apparent whether the UAV was navigating in the correct direction. The UAV was flown in a range of 100 m from the launching point even though it was possible to control the UAV from 300 m.

**Table 9: Flight checklist**

|    | UAV Checking Procedure                                                          | Check |
|----|---------------------------------------------------------------------------------|-------|
| 1  | Was the autopilot reset after code was loaded?                                  |       |
| 2  | Were the modes set up correctly?                                                |       |
| 3  | Was the radio calibrated?                                                       |       |
| 4  | Is the battery fully charged?                                                   |       |
| 5  | Was the radio on when the autopilot was started?                                |       |
| 6  | Was the airframe level during startup?                                          |       |
| 7  | Is the ground control station set on 38,400 baud rate?                          |       |
| 8  | Does the ground control station receive information through the right COM port? |       |
| 9  | Is the autopilot connected to the ground control station?                       |       |
| 10 | Were the waypoints read successfully?                                           |       |
| 11 | Were new waypoints loaded after resetting the autopilot?                        |       |
| 12 | Is the home point at current location?                                          |       |
| 13 | Are the waypoints in line of sight?                                             |       |
| 14 | Are the waypoints at the correct height?                                        |       |
| 15 | Does the autopilot have GPS lock?                                               |       |
| 16 | Does the autopilot register that it is on the ground?                           |       |
| 17 | Is the airspeed sensor reading the correct speed?                               |       |
| 18 | Does the autopilot register that the plane is level?                            |       |
| 19 | Are the manual controls working?                                                |       |
| 20 | Are the controls in the right direction?                                        |       |
| 21 | Can the throttle go to maximum?                                                 |       |
| 22 | Does the autopilot autocorrect in stabilise mode?                               |       |
| 23 | Is the ground control station set up correctly?                                 |       |
| 24 | Is the ground control station recording the flight?                             |       |
| 25 | Is the video transmitter powered-up?                                            |       |
| 26 | Is the video receiver powered-up?                                               |       |
| 27 | Is the video receiver connected to the GCS?                                     |       |
| 28 | Does the GCS receive telemetry?                                                 |       |
| 29 | Is the video feed logging?                                                      |       |

### **4.3. Verification of fly-by-wire mode**

During the first flight the UAV's ability to keep from rolling or pitching past the preset angles was tested using the fly-by-wire mode. In fly-by-wire mode the control is manual but the roll and pitch of the UAV is limited to specific adjustable values. The pitch was limited to 20° above the

horizontal position and  $15^\circ$  below the horizontal. The bank angle was set to  $45^\circ$  in the lateral direction from the straight and level flight position. The fly-by-wire flight path is shown in Figure 43.



**Figure 43: Testing fly-by-wire mode**  
The flight trajectory and height can be seen from the yellow-coloured flight path. The coordinates and the absolute altitude can be seen in the bottom-centre of the picture.

The pilot tried unsuccessfully to invert the mini UAV by rolling it from side to side. The UAV was then pitched up and down to try to invert it while supplying it with enough power to prevent a stall. The result is shown in Figure 44.

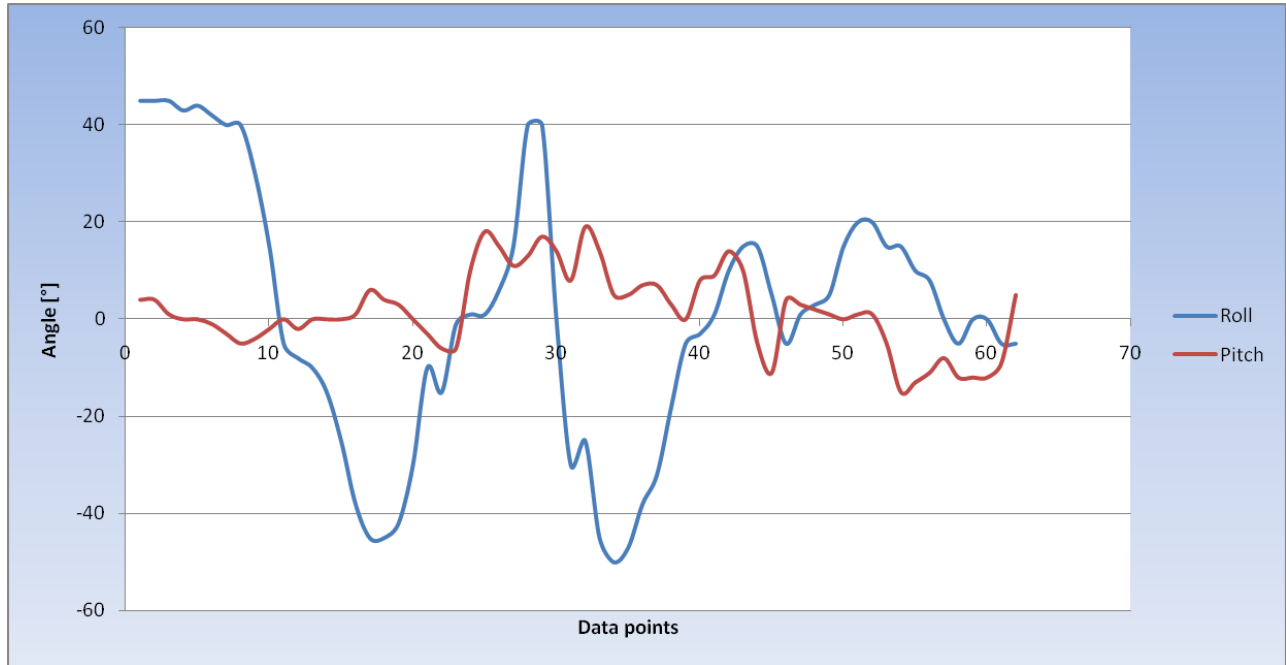


**Figure 44: Roll and pitch angles of UAV during test.**  
**Right bank angles are positive while left bank angles are negative. Pitch up angles are positive while pitch down angles are negative.**

From these results it can be seen that the maximum roll angles were  $45^\circ$  and  $48^\circ$  to the right and left respectively. These are close to the preset values. The small margin by which the left-roll angle was exceeded could be attributed to wind, turbulence or measurement errors. The wind speed was  $13 \text{ m.s}^{-1}$  from a south-west direction. The maximum values for the pitch angles are  $19^\circ$  up and  $15^\circ$  down. This was close enough to the expected pitch values to assume that it was correct.

#### **4.4. Verification of GPS waypoint tracking**

For autopilot flight the UAV needed to navigate specific GPS waypoints. Waypoints were written onto the autopilot's EEPROM as GPS coordinates with specific heights and radiuses. The first autopilot test determined whether it followed the GPS waypoints. An example of the telemetry received from the UAV is given in Appendix B.

Three waypoints were loaded onto the EEPROM for the first autopilot flight. They were all set at the same height of 70 m above the launch level. The distances between the waypoints were approximately 120 m. The radius of each waypoint was 20 m. This means that if the UAV navigated to any GPS coordinate within 20 m of the waypoint coordinates it would register that it had reached the waypoint. In Figure 45 it can be seen what the ideal flight path would typically look like while following three waypoints.

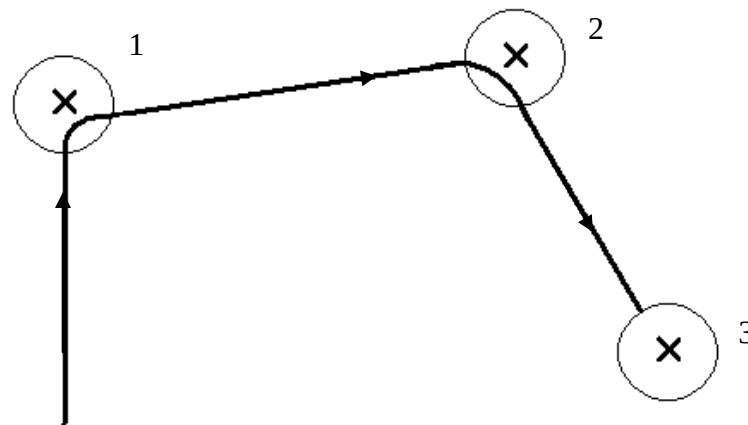


**Figure 45: Waypoint following with a constant waypoint radius.**

This radius was determined by trial and error. If the waypoint radius is too small the UAV tends to miss waypoints and then would have to turn back. A waypoint is missed when the UAV flies past a waypoint at a distance greater than the waypoint radius. If the waypoint radius is too large the UAV does not follow the waypoints accurately enough. The typical flight path is shown in Figure 46.



**Figure 46: Flight path during waypoint-following test**

The flight started off-centre to the right of the figure. It can clearly be seen that the UAV then started following the waypoints numbered from 1 to 3. The snapshot of the flight was taken just after it reached its third and final waypoint. This was a simple flight path that only tested the autopilot's ability to follow waypoints. The result is shown in Figure 47.

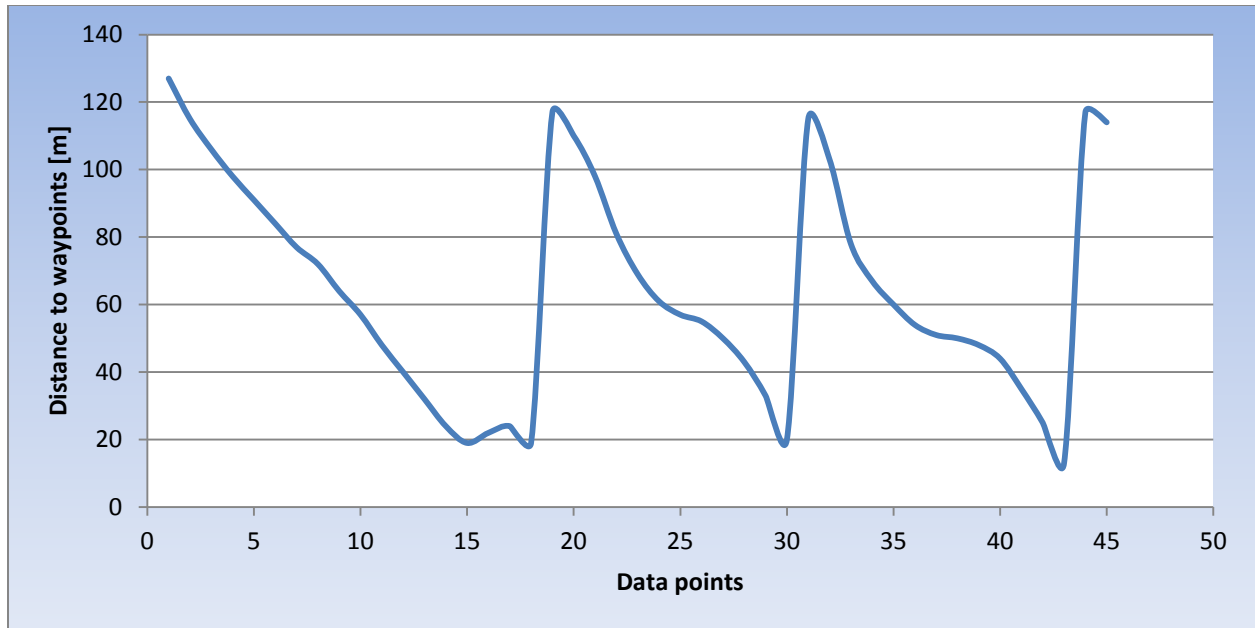


**Figure 47: Distance from waypoint during the autopilot flight**

From Figure 47 it can clearly be seen that the UAV followed its waypoints. It started out 126 m from its first waypoint. The UAV then navigated its way to the first waypoint. When it got to within 20 m of the waypoint it correctly registered that it had reached the first waypoint. It then started navigating its way to the next waypoint. Since the first and second waypoints were approximately 120 m apart it registered that it was about 120 m from the next waypoint. This process repeated until the UAV had finished its mission.

The UAV missed the first waypoint and then compensated by turning sharply towards it. The missed waypoint was most probably because of the crosswind of  $8 \text{ m.s}^{-1}$  from a western direction blowing the UAV off course. Turning sharply caused the UAV to lose a significant amount of height as is discussed in the next section. After it had followed all its waypoints it registered that it was 120 m from its home location. The distance to the home location is the distance it relays when not following waypoints.

#### 4.5. Verification of height maintenance

This test was conducted to determine the accuracy to which the UAV maintains the preset height. During this test the UAV was given four waypoints to navigate, all at an altitude of 50 m above ground level. The average height AMSL of the ground was 1,320 m with a deviation of  $\pm 3$  m. This means that the average height that the autopilot needed to maintain was 1,370 m AMSL. The waypoint radius was 20 m and the flight path is shown in Figure 48.



Figure 48: Height-test flight path

After launching, the UAV started following the waypoints numbered from 1 to 4. The waypoints were chosen in these specific positions to test the ability of the autopilot to maintain its height while banking. It can clearly be seen that the UAV needed to make a turn of approximately  $180^\circ$  every time it got to a waypoint. This tested the autopilot's ability to maintain flight height during

the sharpest turn it will ever be expected to negotiate. A graph of the flight height is shown in Figure 49.

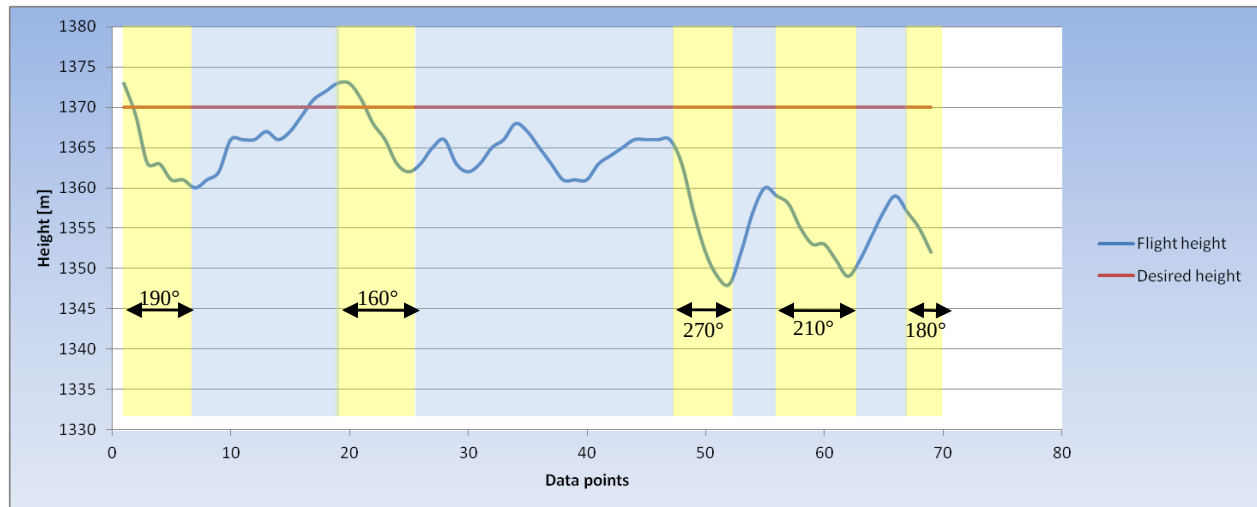


**Figure 49: Desired and maintained flight height**

The yellow shaded areas represent the five major turns that the UAV had to make. From this graph it can clearly be seen that the UAV lost height while turning. The wings are designed to produce lift. While turning, the UAV rolled into the turn which caused a portion of the lift to be directed onto the horizontal plane. Consequently the vertical lift was reduced and the UAV lost height.

The grey shaded areas are the portions of the flight that the UAV was level. It can be seen that most of the height lost during turns was regained while in level flight because the UAV had more lift. The waypoints were too close to each other. Consequently, the UAV did not have enough time to fully regain the lost height.

The UAV started out higher than the desired altitude, which explains the first time the maintained altitude was higher than the desired altitude. At data point 20 the ground altitude was 1,323 m. Thus, for the UAV to be 50 m above ground level it needed to be at 1,373 m. This explains why it went above the average desired altitude.

From data points 25 to 47 the UAV was flying in the direction of the wind. When travelling in the direction of strong winds the UAV struggles to maintain airspeed, reducing its ability to climb. It should also be noted that if the waypoints were further apart the UAV would have reached the required preset altitude between waypoints. From these results it can be seen that the UAV made the correct adjustments to maintain the preset height. The lowest altitude during the flight was 24 m below the desired height.

The pitch compensation value was adjusted to 0.4 which allowed the UAV to make level turns by applying up-elevator during turns. Consequently, the UAV lost less height during turns. The same flight path was flown after the pitch compensation value was adjusted. The results are shown in Figure 50. The absolute height that the UAV was required to maintain was 1,415 m which was 95 m above the launch altitude.

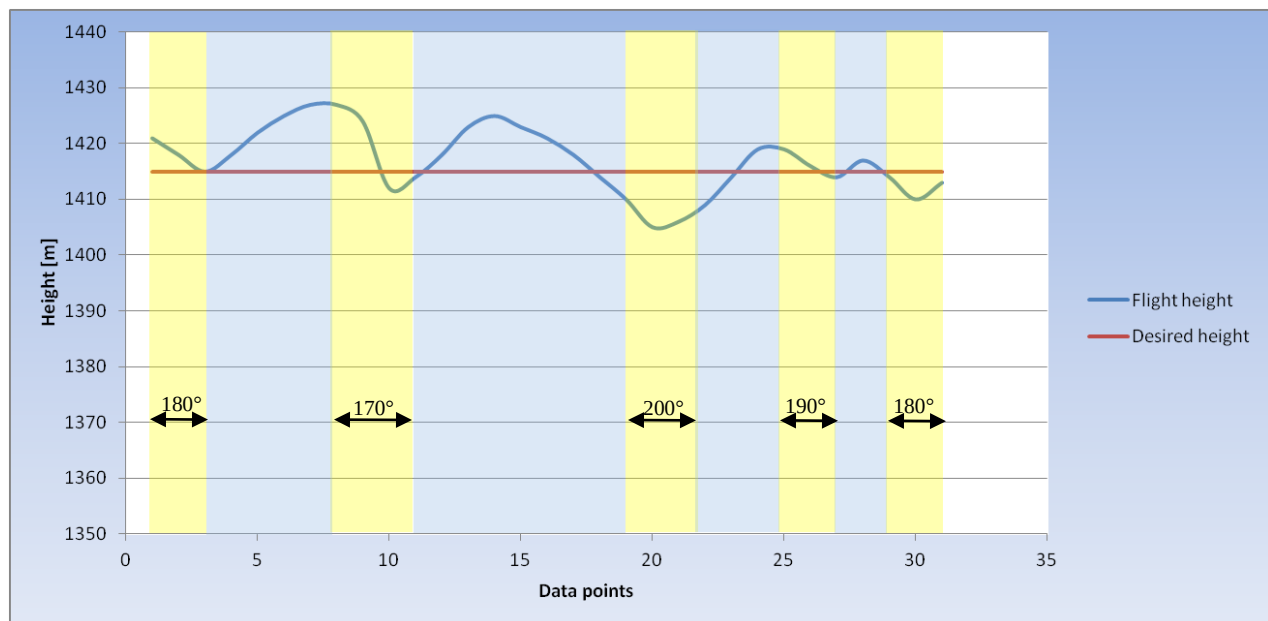


**Figure 50: Desired and maintained flight height after compensation adjustment**

From these results it can be seen that the UAV was able to maintain height more accurately while turning. It also lost less height while turning and it can be seen that the height is more accurately maintained when the turns are closer together. The lowest the UAV went during the

flight was 10 m below the preset height. This is an improvement from the 24 m below the desired height during the initial flight.

#### **4.6. Verification of hand-launch capability**

The hand-launch testing was done in manual mode to determine the minimum distance the UAV would need to reach an altitude of 15 m. The highest house in Silver Lakes Golf Estate is a three-storey house with a roof height of 12 m. It is assumed that the UAV would be able to safely fly over a three-storey house at an altitude of 15 m. The flight test is shown in Figure 51.



**Figure 51: Flight path during hand-launch testing**

Figure 51 demonstrates that the UAV can climb steeply. In this launch the UAV climbed 18 m over a distance of 14 m which is a climb ratio of 1.3:1. This means that the UAV can be launched in a space as small as 11.6 m in diameter if a 15 m height is required. Based on experience 20 m is sufficient distance for safely launching the UAV. The UAV can be launched in a 20 m strip of road without trees. The hand-launch capability is thus sufficient.

The UAV always needs to be launched into the wind to attain maximum airspeed and maximum climb ratio. The UAV has a maximum climb ratio relative to the distance travelled in air. If there is no wind the climb ratio relative to the ground will be the same. With wind, however, this climb ratio relative to the ground will change. A head wind will increase the climb ratio since the relative airspeed is higher, which increases the lift. In the same manner the climb ratio is reduced when a tail wind blows. This is shown in Figure 52.

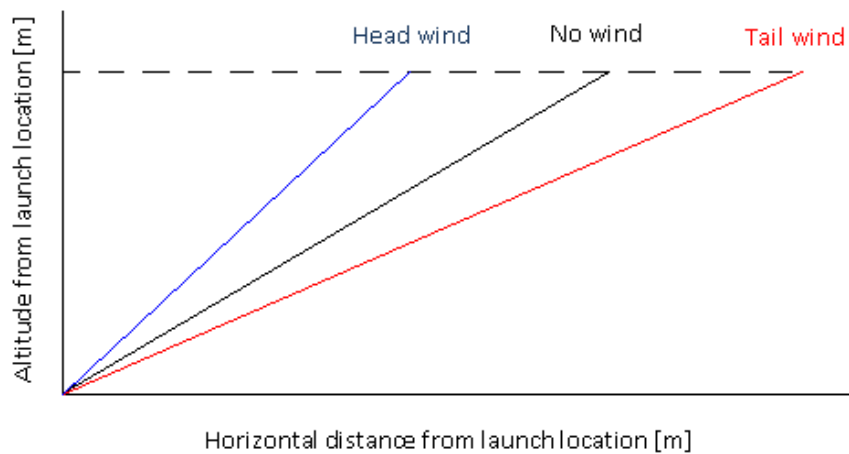


Figure 52: Climb ratio for different wind scenarios

#### 4.7. Verification of maximum height

During this test the height at which the UAV could be safely flown was determined. The UAV was flown in manual mode until it was difficult to determine the UAV's orientation. The height was at an altitude of 1,432 m AMSL which is 110 m above the launch altitude. The flight path for this test is shown in Figure 53.

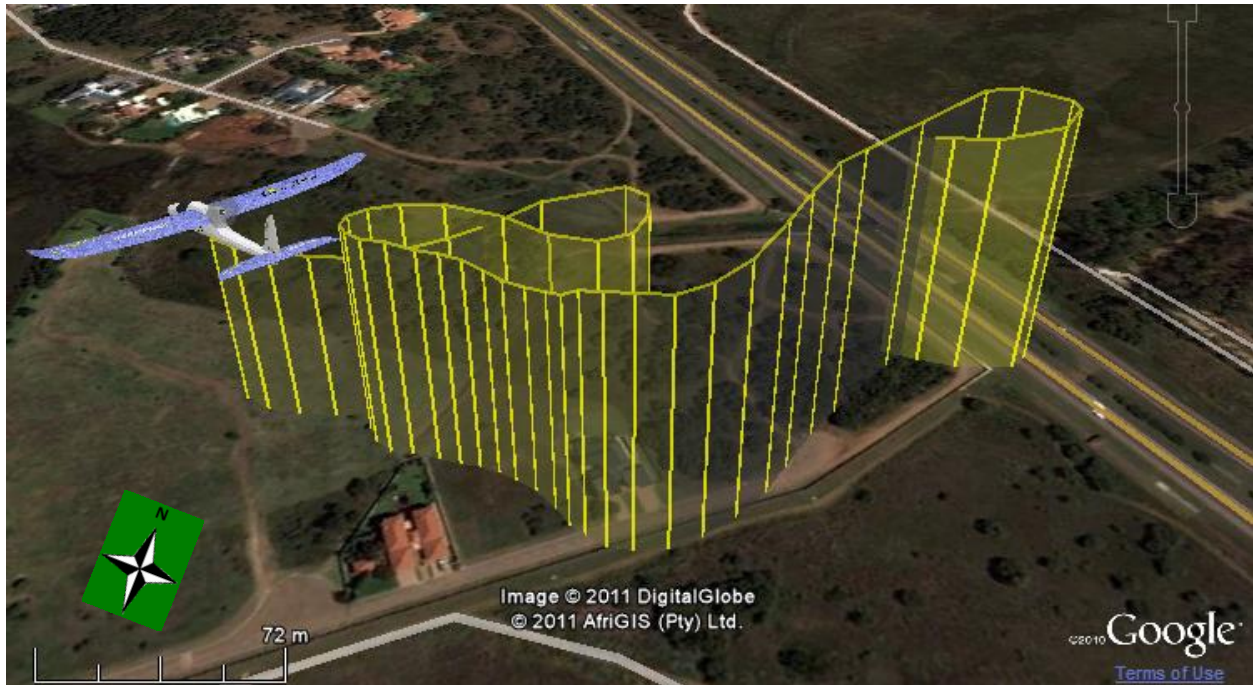


**Figure 53: Flight path during maximum height test**

At this height the UAV could still be clearly seen from the ground. Thus, observers on the ground would also be able to see it. It can, however, not be flown higher since it will then lose its model status — as line of sight is required for model status. At this height the noise is inaudible, which means that it fulfils its requirements partly.

#### **4.8. GCS verification**

The GCS is required to relay the attitude and the position of the UAV. It must show an overlay plot of the UAV on Google Earth™. Finally, all the information needs to be available on a single screen. A screenshot of the ground control station is shown in Figure 54.

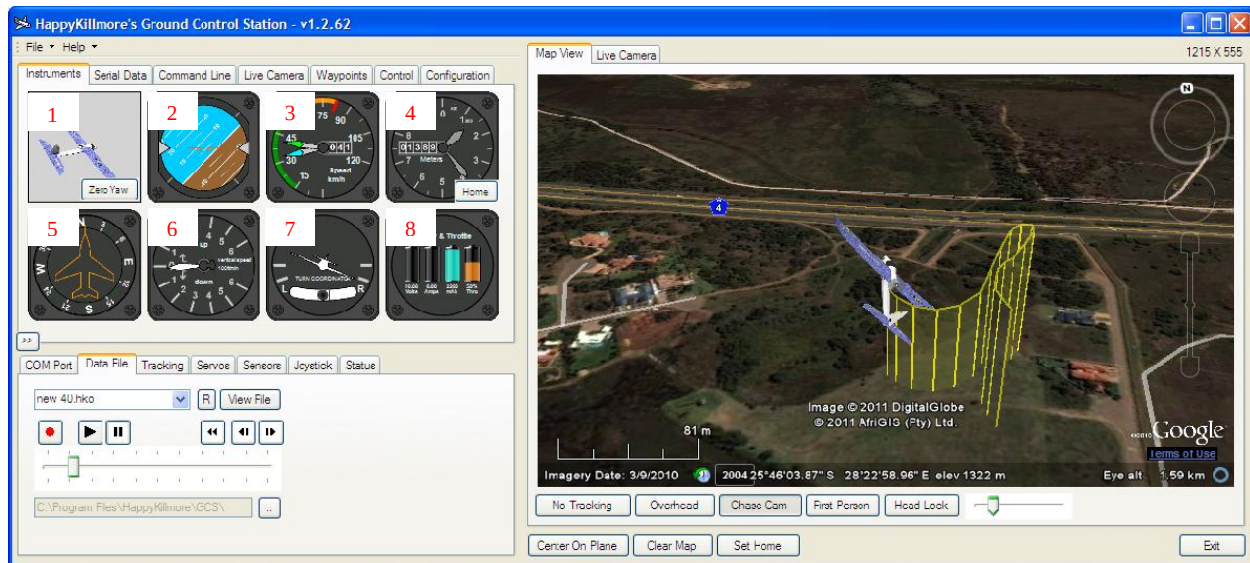


**Figure 54: GCS testing**

Figure 54 displays the attitude in the icons numbered 1 and 2 while 3 and 4 provide airspeed and height AMSL. The position of the UAV is overlaid on Google Earth™. The measured coordinates can also be seen at the bottom of the overlay plot. The true aircraft heading is shown in icon number 5 of the instrument section. The true heading is determined using successive GPS coordinates.

The only problem, however, is the fact that the user needs to toggle between the overlay plot and the live video feed in order to either track the present location of the UAV or monitor the video feed. Further investigation was carried out to acquire a GCS that would allow the operator to monitor the video feed and the position of the UAV simultaneously on the same screen. For this reason ArduPilot Mega Planner was chosen and used for the remainder of the project. It had all the capabilities of the other GCS.

This verified the video feed capability. The camera was mounted stationary — facing directly downwards. The lines on the video feed show the orientation of the UAV relative to the horizon. Please see electronic Appendix attached for an example of the video feed. A screenshot of the new GCS — with video feed enabled — is shown in Figure 55.

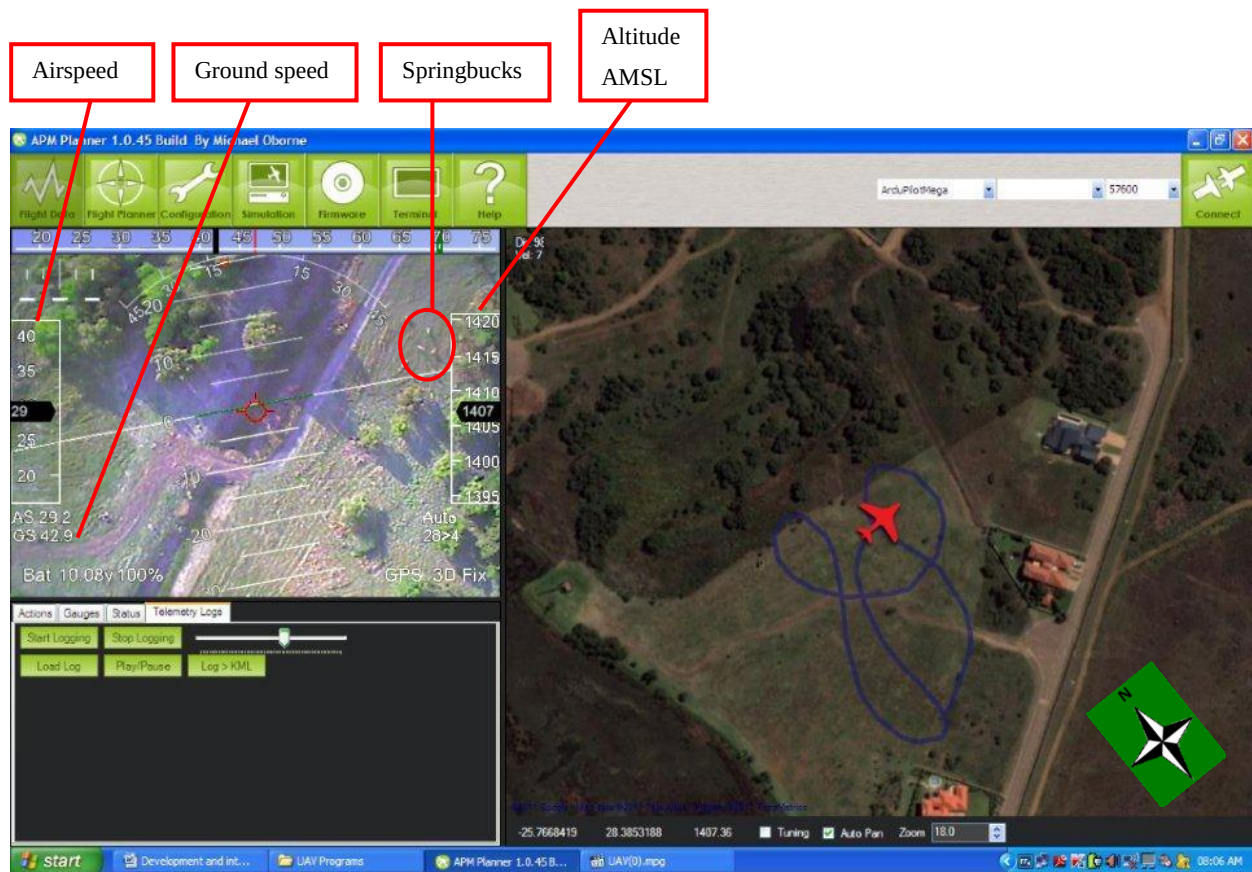


**Figure 55: ArduPilot Mega Planner screenshot**

The video feed is shown in the left of the picture. The Google Earth™ overlay plot is shown on the right. The airspeed is shown on the left of the video feed section. The absolute altitude is shown on the right of the video feed section.

It can clearly be seen that this GCS has the ability to show all attitude and position data required for monitoring the UAV. The video feed and the Google Earth™ overlay is shown on the same screen. The airspeed and groundspeed of the UAV is calculated by the GCS. This allows the user to calculate the wind velocity in the direction the UAV is travelling. The wind velocity is the difference between the airspeed and groundspeed.

Springbucks are indicated on the video feed. This confirms that the camera resolution is sufficient to detect a human on the ground since they are of comparable size. The ground control station has a warning function that tells the operator when the UAV is flying too low. The GCS also informs the operator with voice outputs of the altitude, waypoint, airspeed and battery level during flight.

#### **4.9. Flight range verification**

Another security requirement is that the flight range should be at least 1,566 m. A test was conducted to establish whether the telemetry range exceeded 1,566 m which would ensure that the UAV would be able to monitor the reference urban environment — Silver Lakes Golf Estate. The flight-test procedure is described below:

1. Power the UAV.
2. Connect the UAV to the GCS.
3. Write waypoints to the UAV EEPROM.
4. Allow the UAV to follow the waypoints.
5. Note the coordinates where the UAV loses telemetry link.

These waypoints were all written to coincide with positions above a dirt road that had no obstacles obstructing the pilot's view. For safety reasons, it also needed to be far away from any areas with high population density. The operator remained at the launch position. The pilot was driven along the same path for which the flight path was programmed. This ensured that emergency action could be taken if there were any deviations from the flight path. The flight path can be seen in Figure 56. The start and end coordinates are shown in Table 10.



**Figure 56: Flight path during UAS range verification testing**

**Table 10: Range determination**

| Start coordinates [°] | End coordinates [°]   | Distance between coordinates [m] |
|-----------------------|-----------------------|----------------------------------|
| -25.81898<br>28.51651 | -25.81608<br>28.49561 | 2065.1                           |

The starting coordinates is where the operator monitored the telemetry while the end coordinates is the position where the GCS lost telemetry link. From these results it can clearly be seen that the telemetry range of the UAV is 2,065 m which is well above the required telemetry range of 1,566 m. The video streaming range was determined to be 1.65 km which is above the required range as well. As a result the UAS range is sufficient for monitoring the relatively large Silver Lakes Golf Estate.

#### 4.10. Endurance verification

By studying UAVs with similar applications, it was determined that the UAV required endurance of 30 minutes. The UAV for this project only needed to have enough endurance to circle Silver Lakes Golf Estate at least once. It is assumed that there is no head wind. If a head wind was blowing while the UAV flew along the perimeter, the groundspeed would decrease since the airspeed remains constant. If the groundspeed is lower, the UAV would travel less distance on a single battery charge.

The batteries used are 2.2 Ah LiPo batteries. The UAV needs 52% throttle for maintaining the preset height and airspeed during flight. At this throttle setting the current drawn per motor is 5 A. The UAV has two of these motors. Thus, for the UAV to stay in the air for an hour it would require a 10 Ah battery. Consequently, the flight time can be calculated as in Equation (10).

$$\begin{aligned} t &= Ah60/I \\ t &= \frac{2.2}{10} 60 \\ t &= 13.2 \text{ min} \end{aligned} \tag{10}$$

The cruise speed of the mini UAV was set to 60 km.h<sup>-1</sup>. If the site perimeter is 8030 km, the endurance of the mini UAV must consequently be at least 8.03 min to patrol the perimeter. The maximum flight time of the UAV was measured to be 16.5 min which differed from the calculated theoretical value. It could be attributed to the fact that the UAV reduced throttle while descending which reduced electricity consumption. As a result, the UAV would be able to circle Silver Lakes Golf Estate twice, which is sufficient.

#### 4.11. Data validation

The flight results could not be assumed to be correct if the data was not properly validated. The measurements were validated by comparing the GCS data with manually measured data. The most critical data were those relating to attitude and position. Therefore the gyrometer, GPS and

air pressure data needed to be validated. Additionally, to assure that the airspeed is correct the pitot tube would have to be tested.

#### **4.11.1. Gyrometer data validation**

More specifically, the gyrometer data refers to the roll, pitch and yaw values. The yaw of the UAV is estimated by using the GPS coordinates to determine the direction the UAV is heading. The yaw of the UAV is not essential to autonomous operation. In this section the roll and pitch measurements were validated. The roll and pitch validation procedure is summarised below:

1. The UAV was placed on an eye-level surface.
2. The roll and pitch angles of the fuselage were measured to ensure that they were zero.
3. The UAV was powered to calibrate its sensors.
4. The UAV was connected to the GCS.
5. The UAV was tilted to different angles.
6. The roll and pitch values were measured with a protractor at different angles.
7. These values were compared with the values displayed by the GCS.

The roll limit is  $45^\circ$  in both directions. Thus, if the roll was contained to a value of below  $50^\circ$  it would be sufficient, since it would not invert. This allowed for an average error of 11.1%. An average error measurement of less than 10% would consequently be ideal. The same argument could be followed for the pitch value. The pitch values were  $15^\circ$  and  $20^\circ$  respectively in the down and up positions. A 10% error would contain the UAV pitch to  $16.5^\circ$  and  $22^\circ$ . This was assumed to be sufficiently accurate. The roll data is shown in Table 11 with the pitch data shown in Table 12.

**Table 11: Roll validation data**

The error values are formatted conditionally. The cell colour varies from green to red with an increase in the error value.

| Measured [°] | GCS [°] | Error |
|--------------|---------|-------|
| -35          | -34.6   | 1.2%  |
| -10          | -10     | 0.0%  |
| -6           | -6      | 0.0%  |
| 0            | 0       |       |
| 4            | 4.3     | 7.0%  |
| 12           | 13      | 7.7%  |
| 15           | 17.1    | 12.3% |
| 29           | 30.9    | 6.1%  |
|              | Average | 4.9%  |

**Table 12: Pitch validation data**

The error values are formatted conditionally. The cell colour varies from green to red with an increase in the error value.

| Measured [°] | GCS [°] | Error |
|--------------|---------|-------|
| -23          | -23.4   | 1.7%  |
| -9           | -10.1   | 12.2% |
| -4           | -4.5    | 12.5% |
| 0            | 0       |       |
| 4            | 3.9     | 2.5%  |
| 9            | 10.1    | 10.9% |
| 18           | 17.9    | 0.6%  |
|              | Average | 6.7%  |

The average roll error of 4.9% was well below the maximum assumed value of 10%. The average pitch error of 6.7% was also below 10%. The measurement error was also worsened by the fact that visual measurement of the angles can only be accurate within one degree when determined by protractor. Therefore, it is assumed that both values were sufficiently accurate and that the roll and pitch angles that were displayed were reliable.

#### **4.11.2. GPS data validation**

The GPS coordinates refers to the latitudinal and longitudinal position on the surface of the earth. The GPS can also be used to determine the height of the UAV. Although the autopilot has this

function, it is not used. The height is determined using only the pressure sensor data. The validation procedure is described below and the results are shown in Table 13.

1. Six points, that were well-defined and easy to locate, were marked on Google Earth™.
2. The GPS coordinates of these points were noted.
3. The UAV was powered to calibrate the sensors.
4. It was then connected to the GCS.
5. The UAV was then taken to each of the points mentioned in step 1.
6. The Google Earth™ GPS coordinates were then compared to the values displayed on the GCS.

**Table 13: GPS error validation**

| Google Earth™ coordinate [°] | UAV coordinate [°]    | Distance between coordinates [m] |
|------------------------------|-----------------------|----------------------------------|
| -25.46075<br>28.21477        | -25.46093<br>28.21448 | 33.7                             |
| -25.46075<br>28.21474        | -25.46098<br>28.21445 | 36.7                             |
| -25.46077<br>28.21470        | -25.46100<br>28.21442 | 35.9                             |
| -25.46090<br>28.21462        | -25.46107<br>28.21498 | 39.2                             |
| -25.46089<br>28.21473        | -25.46081<br>28.21435 | 38.1                             |
| -25.46081<br>28.21476        | -25.46069<br>28.21439 | 38.2                             |
| Average error [m]            |                       | 37.0                             |

The average accuracy of the GPS was 37 m. The camera had a viewing angle of 30° to either side which meant that at a height of 100 m the distance that could be seen on the ground was 115.5 m. If the UAV was then able to fly within 37 m accuracy, it would definitely show the desired waypoint. The navigation was consequently accurate enough for this application. It should also be mentioned that Google Earth™ only gives approximate coordinates. The exact accuracy of Google Earth™ cannot be determined since it varies from 0.1 m to 15 m resolution.

#### **4.11.3. Pressure sensor data validation**

The height of the UAV can be determined using the onboard pressure sensor, GPS sensor or a combination of the two. The onboard pressure sensor was used to determine the height during this project. The atmospheric pressure is inversely related to the height above sea level (see Equation (9)) which allows the autopilot to calculate its height above sea level by measuring the atmospheric pressure. By programming the autopilot to maintain a certain height it actually flies at a predetermined, constant pressure. The results are shown in Table 14 and the validation procedure is described below:

1. The UAV was powered to calibrate the sensors.
2. It was then connected to the GCS.
3. The UAV was flown in autopilot mode.
4. The height determined by the pressure sensor was noted.
5. The height determined by the GPS sensor was noted.
6. These two values were compared for validating the pressure sensor.

From Table 14 it can be seen that the average error from these values was 7.4 m. This is an error of 0.52% when compared to height AMSL and an error of 7.8% when compared to height above launch location. The autopilot flight height was 95 m in this test. If the error is factored in, the height would be between 87.6 m and 102.4 m. The lowest the UAV would fly is 80 m since the maximum error value is 15 m as seen in Table 14. This was high enough to avoid crashing into buildings. The flight height could also be adjusted to allow for sensor error when planning a flight. As a result this accuracy was assumed to be satisfactory.

**Table 14: Pressure sensor validation**

| Pressure sensor | GPS sensor | Error [m] | Error [%] |
|-----------------|------------|-----------|-----------|
| 1421            | 1409       | 12.0      | 0.84      |
| 1418            | 1408       | 10.0      | 0.71      |
| 1415            | 1406       | 9.0       | 0.64      |
| 1418            | 1410       | 8.0       | 0.56      |
| 1422            | 1412       | 10.0      | 0.70      |
| 1425            | 1413       | 12.0      | 0.84      |
| 1427            | 1414       | 13.0      | 0.91      |
| 1424            | 1409       | 15.0      | 1.05      |
| 1412            | 1409       | 3.0       | 0.21      |
| 1418            | 1409       | 9.0       | 0.63      |
| 1423            | 1412       | 11.0      | 0.77      |
| 1421            | 1409       | 12.0      | 0.84      |
| 1414            | 1407       | 7.0       | 0.50      |
| 1410            | 1407       | 3.0       | 0.21      |
| 1405            | 1407       | 2.0       | 0.14      |
| 1406            | 1409       | 3.0       | 0.21      |
| 1409            | 1409       | 0.0       | 0.00      |
| 1414            | 1411       | 3.0       | 0.21      |
| 1419            | 1410       | 9.0       | 0.63      |
| 1419            | 1409       | 10.0      | 0.70      |
| 1416            | 1407       | 9.0       | 0.64      |
| 1414            | 1408       | 6.0       | 0.42      |
| 1417            | 1409       | 8.0       | 0.56      |
| 1414            | 1408       | 6.0       | 0.42      |
| 1410            | 1408       | 2.0       | 0.14      |
| 1410            | 1409       | 1.0       | 0.07      |
| Average error   |            | 7.4       | 0.52      |

#### 4.11.4. Pitot tube data validation

The airspeed sensor was also tested. The airspeed reading was used to regulate the speed at which the UAV flew. The sensor was tested by mounting an anemometer and the pitot tubes to a moving vehicle. This was done since a wind tunnel was not available at the time the test was

done. Both these measurements were then noted and compared to determine the accuracy of the pitot tubes. The measurements are shown in Table 15.

**Table 15: Wind speed sensor validation**

The error values are formatted conditionally. The cell colour varies from green to red with an increase in the error value.

| Anemometer [m/s]  | Pitot tube [m/s] | Error [%] |
|-------------------|------------------|-----------|
| 6.75              | 8                | 16.2      |
| 8.22              | 10.55            | 22.1      |
| 10.76             | 12.5             | 13.9      |
| Average Error [%] |                  | 17.4      |

From the measurements taken it was established that the average error was 17.4%. The UAV flies at a maximum speed of  $60 \text{ km.h}^{-1}$  ( $16.67 \text{ m.s}^{-1}$ ). If the speed of the mini UAV was regulated at  $60 \text{ km.h}^{-1}$  a 17.4% accuracy would ensure a wind speed of  $49.6 - 70.4 \text{ km.h}^{-1}$ . This accuracy was determined to be sufficient since any speed within these boundaries will ensure stable flight without stalling.

#### 4.12. Conclusion

In this chapter a flight procedure for testing the autopilot was described. It included the ground and air testing. A flight checklist was compiled to minimize any operator errors. The UAV was then tested and it was found that the flight was stable and the UAV did not roll or pitch past the preset limits.

The UAV followed the waypoints accurately and maintained the preset height within acceptable limits. Excess height-loss only occurred during sharp turns, which was regained during level flight. Greater elevator deflection resulted in improved level flight performance during extended turns. This ensured that the UAV height was maintain more accurately.

The requirements for integrating the UAV into an urban security system were also tested. This included testing the hand-launch capabilities. The UAV could be hand-launched over a distance

of 20 m with sufficient obstacle clearance capabilities. Next the maximum manual controllable flight height for the airframe was tested and found to be 110 m. This was sufficient for the airplane to be inaudible, although it could still be observed visually. It was also observed that the GCS was sufficient with the exception of the one-screen display functionality. This was corrected by changing GCSs.

Finally, the sensor data were validated to ensure that the results were reliable. It was found that all the sensors were acceptably accurate. This confirmed that the flight results were accurate. From these results it could be assumed that the UAV has fully autonomous flight capabilities, with the exception of automatic take-off and landing capabilities, which is not part of this project.

---

## 5. Conclusion

---

---

### Summary

In this chapter the study goals are reviewed. It is then discussed how the goals were addressed and recommendations for future study are made.

---

### **5.1. Review of study goals**

The goal of this study was to develop a low-cost, fully autonomous mini UAV. The UAS needed to be simple to use and be allowed to fly in the NAS. By analysing UAVs presently suited for security applications, basic UAV requirements could be identified. These requirements are:

- The UAV must accurately follow predefined waypoints.
- The flight must be stable enough for video-streaming.
- Accurate height control.

While building and testing the mini UAV the abovementioned requirements were the main considerations. The requirements for operation in the NAS were not considered since the UAV has model status and these requirements do not apply. The UAV also needed to be integrated into an urban security system. By testing the autonomous flight capabilities and the integration requirements the effectiveness of the system could be measured. The integration requirements that were identified are listed below:

- Hand-launch capabilities.
- Video-streaming.
- Coordinates of the centre of the image.
- It must be hard to detect.
- All the information needs to be shown on one screen.
- The flight path must be overlain on Google Earth<sup>TM</sup>.
- It needs to have a telemetry range greater than 1,566 m.
- Collision avoidance capabilities.
- Lost link recovery system.
- Low flight height warning system.

## 5.2. Addressing goals

Research was done to determine exactly how UAVs are classified. UAVs currently available were then researched to determine what technology is available. The research gave good insight into what can realistically be expected of the UAV that needed to be developed.

A study was then done to assess basic urban security systems to identify weaknesses a UAV could improve on. Next, the security applications of UAVs were investigated. Consequently requirements of UAVs in security applications could be identified.

A fully functional UAV was developed using the Arduino ArduPilot Mega autopilot and other carefully selected COTS products. The UAV was customised for optimum performance and integrated with a GCS. This allowed the user to see the attitude and position of the UAV, give it commands and stream live video. The integration with the urban security system was completed according to the specified requirements.

Different functions of the UAV were tested including testing the stability, waypoint-tracking accuracy and height-maintaining capabilities. The final product specifications, and how these specifications compare to commercially available UAVs, are shown in Table 16.

**Table 16: Comparison between commercial UAVs and project UAV**

|                        | Commercial UAV specifications | Project UAV specifications |
|------------------------|-------------------------------|----------------------------|
| Wingspan [m]           | 2                             | 1.4                        |
| Endurance [min]        | 60                            | 16.5                       |
| Range [km]             | 5                             | 2.1                        |
| Mass [kg]              | 5                             | 2.3                        |
| Hand-launch capability | Yes                           | Yes                        |
| Cost [R]               | 272,300                       | 11,141                     |
| Assembly time [min]    | 10                            | 10                         |
| Speed [km/h]           | 74                            | 60                         |
| Video streaming        | Yes                           | Yes                        |

Finally, the security features were tested extensively. As a result it could be seen how successfully the UAV was integrated with the urban security system. In Table 17 the security requirements are summarised.

**Table 17: Security requirements**

| Requirement                   | Has requirement been satisfied | Comment                       |
|-------------------------------|--------------------------------|-------------------------------|
| Hand-launch                   | Yes                            | 20 m launch distance          |
| Video-streaming               | Yes                            | -                             |
| Image coordinates             | Yes                            | -                             |
| Difficult to detect           | Yes                            | Inaudible, but can be seen    |
| One screen display            | Yes                            | Attitude and position display |
| Endurance to circle perimeter | Yes                            | -                             |
| 1.566 km range                | Yes                            | -                             |
| Google Earth™ flight path     | Yes                            | -                             |
| Secure information            | No                             | Cost too high                 |

### 5.3. Project conclusions

The UAV that was developed has fully autonomous flight capabilities. It has also been successfully integrated with urban security systems. It can navigate its waypoints successfully as well as maintain its height. The UAV can be hand-launched and maintains a sufficient height to fly safely. All the relevant information necessary for monitoring the flight is relayed to the operator.

The UAV is also small enough for car transport. It has a flight range long enough to monitor perimeters of 16.5 km and the endurance of the UAV is 16.5 min. It is concluded that the UAV is effectively integrated with an urban security system. One of the drawbacks of the system is that it is difficult to operate since it does not yet have autonomous take-off and landing capabilities. Lastly, the height it is capable of flying at is not high enough to avoid detection.

#### 5.4. Recommendations for future study

- It is recommended that the autopilot be built into a delta-wing airframe. This will give the UAV increased stability at high speeds.
- The UAV needs autonomous take-off and landing capabilities which would make controlling the UAV simpler for less educated operators.
- Software for the analysis of the video feed needs to be acquired or developed which would allow the system to function without human operators.
- The video feed telemetry can be integrated into the autopilot telemetry which would simplify the system and save production cost.
- Test the UAS with real security operators which could give further insight into integration requirements.
- Research objects detection and collision avoidance systems which could include in-flight collision avoidance with other airplanes, UAVs and birds as well as stationary objects like buildings and masts.
- Research how multiple UAVs should be positioned to give the optimal view of an area (Schwager *et al.*, 2011).
- Testing the UAV with disturbances such as wind.
- A backup navigation system — that uses ground based equipment — needs to be created for when the UAV does not have GPS lock (DeGarmo, 2004).
- A UAS system needs to be developed that not only incorporates multiple UAVs, but can also be controlled from more than one GCS (DeGarmo, 2004).
- Alter the code to maintain a preset height above ground level, rather than launch altitude.
- Wind tunnel testing must be done to accurately determine the stall speed.
- Install a camera that can pan and tilt.
- Refer to Appendix C for further recommendations from *Online Intelligence* (a South African security firm that is, among others, responsible for the security in Silver Lakes Golf Estate).

---

## 6. References

---

Ahmed, S. & Samar, R. (2007). *Comparative study of speed control algorithms for UAV applications*, 68-72. Conference proceedings of International Bhurban Conference on Applied Sciences and Technology held in Islamabad, Pakistan.

Arduino. (2011). Homepage of Arduino. Available from: [www.arduino.cc](http://www.arduino.cc) (accessed 13 March 2011).

Arnott, R. (2007). *The selection and implementation of an autopilot for a mini-unmanned aerial vehicle*. B.Eng. dissertation. Pretoria, South Africa: University of Pretoria.

ATE-Group. (2010). Homepage of Advanced Technology and Engineering. Available from: [www.ate-group.com](http://www.ate-group.com) (accessed 15 August 2011).

Barrows, G. L. (2002). *Future visual micro sensors for mini/micro UAV applications*, 7:498-506. Proceedings of the IEEE International Workshop on Cellular Neural Networks and their Applications held in Frankfurt, Germany.

Beainy, F. & Mai, A. (2009). *Unmanned aerial vehicles operational requirements and fault-tolerant robust control in level flight*, 17:700-705. Conference proceedings at the Mediterranean Conference on Control and Automation held in Thessaloniki, Greece.

Bin, H. & Justice, A. (2009). The design of an unmanned aerial vehicle based on the Ardupilot. *Indian Journal of Science and Technology* 2(4):12-15.

Bolkcom, C. (2005). *Homeland security: Unmanned aerial vehicles and border security*. CRS Report for Congress. Available from: <http://www.fas.org/sgp/crs/homesecc/RS21698.pdf>

Bozkurt, U. & Aslan, M. (2009). *Assembly of an unmanned aerial vehicle (UAV)*. Bachelor's dissertation. Halmstad, Sweden: Halmstad University, School of Electrical Engineering.

Cambone, S., Krieg, K., Pace, P. & Wells II, L. (2005). *Unmanned aircraft systems roadmap 2005-2030*. Office of the Secretary of Defense, US Department of Defense. Available from: <http://uav.navair.navy.mil/roadmap05/USRoadmapAug%2005.pdf>

Cameron, T., Colton, K. & Childers, D. (2011). *AETHER*. Bachelor's dissertation. California, USA: California Polytechnic State University, School of Electrical Engineering.

Chao, H., Cao, Y. & Chen, Y. (2007). *Autopilots for small fixed-wing unmanned air vehicles: A survey*, 3144-3149. Conference proceedings of the IEEE International Conference on Mechatronics and Automation held in Harbin, China.

Cheng, S. (2008). *Rapid deployment UAV*, 1-8. Conference proceedings of the Aerospace Conference held in Big Sky, Montana, USA.

Clothier, R. A., Palmer, J. L., Walker, R. A. & Fulton, N. L. (2011). Definition of an airworthiness certification framework for civil unmanned aircraft systems. *Safety Science*, 49:871-885.

Cosyn, P. & Vierendeels, J. (2007). Design of fixed wing micro air vehicles. *The Aeronautical Journal*.

Cyber Aerospace. (2009). Homepage of Cyber Aerospace. Available from: [www.cyberaerospace.com](http://www.cyberaerospace.com) (Accessed 15 August 2011).

Da Silva, B., Demers, É., Gédéon, A., Jobin, S., Lévesque, M., Mignault, C. & Vidal, C. (2008). AUVSI 2008 Student UAS Competition: VAMUdeS Project. Québec, Canada: Université de Sherbrooke.

de Fatima Bento, M. (2008). Unmanned aerial vehicles: An overview. *Inside GNSS Magazine*, January/February Print Edition, 54-61.

Defense Industry Daily. (2011). Homepage of Defense Industry Daily. Available from: [www.defenseindustrydaily.com](http://www.defenseindustrydaily.com) (Accessed 17 October 2011).

DeGarmo, M. T. (2004). *Issues concerning integration of unmanned aerial vehicles in civil airspace*. MITRE, Center for Advanced Aviation System Development, McLean Virginia, USA.

Dhlamini, M., Kachienga, M. O. & Marwala, T. (2007). Artificial intelligence as an aide in management of security technology. *IEEE Africon*, 200:1-5.

Dick, A. R. & Brooks, M. J. (2003). *Issues in automated visual surveillance*, 195-204. Conference proceedings of the 7<sup>th</sup> Digital Image Computing: Techniques and Applications held in Sydney, Australia.

DIY Drones (2011). Homepage of Do-It-Yourself Drones. Available from: [www.DIYDrones.com](http://www.DIYDrones.com) (Accessed 13 March 2011).

Drury, J. L., Richer, J., Rackliffe, N. & Goodrich, M. A. (2006). *Comparing situation awareness for two unmanned aerial vehicle human interface approaches*. Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR) held in Gaithersburg, Maryland, USA.

Dufrene, W. R. Jr. (2005). *Mobile military security with concentration on unmanned aerial vehicles*, 2:1-8. Conference proceedings of the 24<sup>th</sup> Digital Avionics Systems Conference held in Washington, DC, USA.

Franklin, G. F., Powell, J. D., Emami-Naeini, A., (1986). *Feedback control of dynamic systems*. Reading, Massachusetts: Addison-Wesley Publ. Co.

Gheorghe, A. V. & Ancel, E. (2008). *Unmanned aerial systems integration to national airspace system*, 1-5. Conference proceedings of the 1<sup>st</sup> International Conference on infrastructure Systems and Services: Building Networks for a Brighter Future held in Rotterdam.

Hau, L., Shi-chao, Z., Choa, H., Ming, Z. & Xiao-feng, M. (2009). *A near infrared imaging detection system based on Davinci platform*, 4-154-4-159. Conference proceedings of the 9<sup>th</sup> International Conference on Electronic Measurement & Instruments held in Beijing.

How Stuff Works. (2011). Homepage of How Stuff Works. Available from: [www.howstuffworks.com](http://www.howstuffworks.com) (Accessed 11 October 2011).

Ingham, L. A., Jones, T., Maneschijn, A. (2006). Certification of unmanned aerial vehicles in South African airspace. *Research and Development Journal* 22(1):21-27.

Intel. (2011). Homepage of Intel. Available from: [www.intel.com](http://www.intel.com) (Accessed 11 October 2011).

Jovanovic, O. (2006). *Unmanned aerial vehicles development of navigation and mission planning*. Bachelor's dissertation in Digital Systems. Melbourne, Australia: Monash University, Clayton Campus.

Kuo, C., Weidner, M., Toms, T., Choe, H., Chang, K., Harwood, A., Jelemensky, J. & Smith, P. (1992). A 512-kb flash EEPROM embedded in a 32-b microcontroller. *IEEE Journal of Solid-State Circuits*, 27(4):574-582.

Lipton, A. J., Heartwell, C. H., Haering, N. & Madden, D. (2002). *Critical asset protection, perimeter monitoring, and threat detection using automated video surveillance*. Unpublished manuscript. Available from: <http://www.objectvideo.com/products/onboard/whitepapers>

Mallesham, G., Mishra, S., Jha, A. N. (2009). *Maiden application of Ziegler-Nichols method to AGC of distributed generation system*. Power Systems Conference and Exposition held in Seattle, Washington.

Michael, N. & Kumar, V. (2011). Control of ensembles of aerial robots. *Proceedings of the IEEE*; 99(9):1587-1602.

MicroPilot. (2011). Homepage of MicroPilot. Available from: [www.micropilot.com](http://www.micropilot.com) (Accessed 16 August 2011).

Müller, M. W. (2008). *UAV autopilot implementation: Longitudinal control*. Bachelor's dissertation. Pretoria, South Africa: University of Pretoria, School of Mechanical Engineering.

Multiplex. (n.d.). Homepage of Multiplex USA. Available from: [www.multiplexusa.com](http://www.multiplexusa.com) (Accessed 24 April 2011).

Parker, S. P. (2002). *McGraw-Hill Dictionary of Scientific and Technical Terms*. 6<sup>th</sup> edition. New York: The McGraw-Hill Companies.

Pavlidis, I., Morellas, V., Tsiamyrtzis, P. & Harp, S. (2001). Urban surveillance systems: From the laboratory to the commercial world. *Proceedings of the IEEE*, 89(10):1478-1497.

Qiang, M., Sasa, M., Lianghua, X. & Yabin, W. (2009). *Research on security monitoring and health management system of medium-range UAV*, 854-857. Conference proceedings of the 8<sup>th</sup> International Conference on Reliability, Maintainability and Safety held in Chengdu, China.

Rangevideo. (2011). Homepage of Range Video. Available from: [www.rangevideo.com](http://www.rangevideo.com) (Accessed 23 May 2011).

Ribeiro, L. R. & Oliveira, N. M. F. (2010). *UAV autopilot controllers test platform using Matlab/Simulink and X-Plane*. Conference proceedings at the Frontiers in Education Conference held in Washington, DC, USA.

Ro, K., Oh, J. & Dong, L. (2007). *Lessons learned: Application of small UAV for urban highway traffic monitoring*. 45<sup>th</sup> AIAA Aerospace Sciences Meeting and Exhibit: Reno, Nevada.

Russel, S. (2007). The potential for using UAVs in Australian security applications. *Recent Advances in Security Technology*, 388-397.

RUVSA. (2011). Homepage of Russian Unmanned Vehicle Systems Association. Available from: <http://en.ruvs.com/> (Accessed 15 August 2011).

Schwager, M., Julian, B. J., Angermann, M. & Rus, D. (2011). Eyes in the sky: Decentralized control for the deployment of robotic camera networks. *Proceedings of the IEEE*, 99(9):1541-1561.

Shawn, M. A. J. & Weed, C. (2002). *The quality of quantity: Mini-UAVs as an alternative UAV acquisition strategy at the army brigade level*. Leavenworth, USA: School of Advanced Military Studies.

Silver Lakes. (2009). Homepage of Silver Lakes Golf Estate. Available from [www.silverlakes.co.za](http://www.silverlakes.co.za) (Accessed on 18 October 2011).

Sinnathamby, G., Wisnarama, S. & Sotodeh, S. (2011). *Design and implementation of a stabilization system for a remote controlled, fixed-wing, model unmanned aerial vehicle*. Bachelor's dissertation. Toronto, Canada: York University.

Tikanmäki, I., Touhima, T. & Rajamäki, J. (2011). How and why unmanned aircraft vehicles can improve real-time awareness. *International Journal of Circuits, Systems and Signal Processing*, 5(5):469-477.

Torun, E. (1999). *UAV requirements and design consideration*. RTO SCI Symposium on “Warfare Automation: Procedures and Techniques for Unmanned Vehicles” held in Ankara, Turkey.

Tvaryanas, A. P., Thompson, B. T. & Constable, S. H. (2005). *US Military unmanned aerial vehicle mishaps: Assessment of the role of human factors using human factor analysis and classification system (HFACS)*. Brooks City-Base, USA: 311th Performance Enhancement Directorate.

United States of America Department of Defense. (2001). *Dictionary of Military and Associated Terms*. Darby, Pennsylvania: Diane Publishing Co.

Visioli, A. (2006). *Practical PID control*. London, England: Springer.

Willy, L. W. L. (2003). *Effects of radio wave propagation in urbanized areas on UAV-GCS command and control*. Master’s thesis. Monterey, California: Naval Postgraduate School.

---

## Appendix A - Project cost breakdown

---

| Product                                                   | Quantity | Cost per component [R] | Cost [R] |
|-----------------------------------------------------------|----------|------------------------|----------|
| Autopilot, IMU shield and GPS sensor                      | 1        | 1750                   | 1750     |
| Xbee telemetry kit                                        | 1        | 1050                   | 1050     |
| KX 191 Camera                                             | 1        | 735                    | 735      |
| Airspeed Kit with MPXV7002DP differential pressure sensor | 1        | 175                    | 175      |
| X Power 2200 mAh battery                                  | 2        | 275                    | 550      |
| Ardupilot Mega Case                                       | 1        | 91                     | 91       |
| 1.3 GHz 800mW video transmitter                           | 1        | 455                    | 455      |
| 1.3 GHz deluxe receiver                                   | 1        | 455                    | 455      |
| Spectrum DX-7 Radio                                       | 1        | 3500                   | 3500     |
| Spectrum AR9010 receiver                                  | 1        | 1372                   | 1372     |
| Multiplex Twinstar II                                     | 1        | 1008                   | 1008     |
| Total cost [R]                                            |          |                        | 11,141   |

The following is sample data collected from the GCS. It is the raw data which is sent to the GCS by the UAV. This telemetry data is converted to usable attitude and position data by HappyKillmore GCS. This attitude and position is also overlain on Google Earth™.

108

109

110

---

## Appendix C – Issues raised and proposed operational system received from *Online Intelligence (Pty) Ltd*

---

- **Establish central control room for UAV operators.**

*Online Intelligence* proposes that the UAV operator must be in a remote location where all the security sites' UAVs can be operated from a central location. The video feed is then streamed to the onsite control room where the information can be analysed and stored. This allows qualified personnel to be used in multiple locations.

- **Automatic take-off and landing.**

Because the UAV operator will be off-site it is very important that the UAV can be operated autonomously.

- **Fixed surveillance schedule with UAV able to deploy if an alarm signal is received.**

It is proposed that the UAV be launched at fixed intervals to do a complete scan of the perimeter. If an alarm signal is received the UAV must be immediately deployed to the location of the threat. If the UAV is in the air or there are more than one UAV in the air the closest UAV must change course to the location of the threat following the shortest route.

- **Incorporate a handheld receiver.**

When an alarm signal is received, the UAV is deployed. When the UAV reaches the required position the video feed is used to determine whether there is a security threat. The UAV operator then follows the trespasser while the patrol guards are dispatched. The video feed then needs to be relayed to the patrol guards in order for them to intercept the trespasser. For this reason a handheld video receiver is proposed.

- **Incorporate infrared camera.**

It is proposed that serious consideration be given to incorporating infrared cameras into the UAV. Spotting trespassers that are concealed during night-time is a high priority.

- **On board video logging and estimate of last location.**

When the UAV malfunctions or is operated by an unknown source, deciding where to start searching for the UAV is important. Since it is possible that the UAV might fly out of range of the GCS it is advised that on board data logging is incorporated.

- **Build a servomechanism to point camera down.**

*Online Intelligence* proposed building a servo mechanism to manoeuvre the video camera to point down. This would ensure that the desired location is shown while following waypoints.

- **Telemetry identification.**

If more than one UAV is used at a specific site and they are all transmitting at the same frequency it is important that each UAV has telemetry identification. This will allow the UAVs to be controlled individually.

- **Video streaming over secure internet protocol address.**

It is advised that the video be streamed to the central control room over a secure internet protocol address. This will allow for advanced decision making at a remote location.

- **Rugged design for repeated use.**

The UAV will be used repeatedly and is required to be a rugged design.

- **Rather deliver the UAV service than distribute complete packages to identify commissioning issues.**

Initially the UAS will have typical commissioning issues that will need to be resolved. It is therefore advisable to deliver a service initially to identify and resolve potential problems with the system. There will however be customers requiring to run and maintain the UAS themselves and therefore a product will eventually be required.

- **Complete UAV electronics package able to be transferred between airframes.**

The system will require interchangeable parts. If a component malfunctions it must be replaced immediately without requiring specialised tools. For this reason it is advised that the autopilot with all its sensors be a complete system that can be transferred between airframes.