Investigation of Fisheye Lenses for Small UAV Aerial Photography

Masters by Research
in
Electrical Engineering

Alex Gurtner
Ingenieur FH Elektrotechnik
Berner Fachhochschule Burgdorf
Schweiz, 2006

Australian Research Center for Aerospace Automation
Built Environment and Engineering
Engineering Systems
Queensland University of Technology
Brisbane, Australia

Principal Supervisors:
Professor Rodney Walker
Associate Professor Wageeh Boles

November 24, 2008, Brisbane

Statement of Original Authorship

The work contained in this thesis has not been previously submitted to meet requirements for an award at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

Signature sig Alex Gurtner

Date Payerne, 24.11.2008

Abstract

Aerial photography obtained by UAVs (Unmanned Aerial Vehicles) is an emerging market for civil applications. Small UAVs are believed to close gaps in niche markets, such as acquiring airborne image data for remote sensing purposes. Small UAVs will be able to fly at low altitudes, in dangerous environments and over long periods of time. However, the small lightweight constructions of these UAVs lead to new problems, such as higher agility leading to more susceptibility to turbulence and limitations in space and payload for sensor systems. This research investigates the use of low-cost fisheye lenses to overcome such problems which theoretically makes the airborne imaging less sensitive to turbulence. The fisheye lens has the benefit of a large observation area (large field of view) and doesn't add additional weight to the aircraft, like traditional mechanical stabilizing systems. This research presents the implementation of a fisheye lens for aerial photography and mapping purposes, including theoretical background of fisheye lenses. Based on the unique feature of the distortion being a function of the viewing angle, methods used to derive the fisheye lens distortion are presented. The lens distortion is used to rectify the fisheye images before these images can be used in aerial photography. A detailed investigation into the inner orientation of the camera and inertial sensor is given, as well as the registration of airborne collected images. It was found that the attitude estimation is critical towards accurate mapping using low-quality sensors. A loosely coupled EKF filter applied to the GPS and inertial sensor data estimated the attitude to an accuracy of 3–5 ° (1-sigma) using low-cost sensors typically found in small UAVs. However, the use of image stitching techniques may improve the outcome. On the other hand, lens distortion caused by the fisheye lens can be addressed by rectification techniques and removed to a sub-pixel level. Results of the process present image sequences gathered from a piloted aircraft demonstrating the achieved performance and potential applications towards UAVs. Further, an unforeseen issue with a vibrating part in the lens lead to the need for vibration compensation. The vibration could be estimated to ±1 pixel in 75 % of the cases by applying an extended Hough transform to the fisheye images.

Contents

List of Abbreviations **XI**

Nomenclature **XV**

List of Figures **XIX**

List of Tables **XXIII**

1 Introduction **1**

2 Literature Review **5**

2.1 UAV Aerial Photography 6
2.2 Fisheye Lenses 7
2.3 Fisheye Image Rectification 9
2.4 Image Registration and Mapping 13
2.5 Collision Avoidance 14
2.6 Identified Benefits 15
2.7 Summary 15

3 Approach to the Research **17**

3.1 Fisheye Lens Calibration 18
3.1.1 Fisheye Lens Parameter Estimation 18
3.2 Image Rectification 29
3.2.1 Extraction and Rectification 29
3.2.2 Further use of Image Extraction 32
3.3 Image Registration 32
3.3.1 Attitude Estimation 33
3.3.2 DEM Data Preparation 36
3.3.3 Image Transformation 37
3.4 Mapping 42

3.5 Summary 43

4 Image Acquisition **45**
4.1 Airborne Data Acquisition 45
4.2 Sensor Installation 47
4.3 Data Synchronization 49
4.3.1 Laptop Interrupt Delay 50
4.3.2 Synchronizing Image Data 54
4.3.3 Synchronizing Inertial Sensor Data 55
4.4 Summary 57

5 Vibration Compensation **59**
5.1 The Problem 59
5.2 Requirements 61
5.3 Approach to the Vibration Compensation 62
5.3.1 Speeding up the Hough Transform 66
5.4 Selection of the Fisheye Circle Boundary 66
5.4.1 Factors Influencing the Circle Boundary 67
5.4.2 The Canny Edge Detector 70
5.5 Results and Analysis 72
5.5.1 Results of Single Images 73
5.5.2 Results based on Image Sequences 77
5.5.3 Refined Results 78
5.6 Summary 80

6 Results and Analysis **83**
6.1 Fisheye Lens Calibration 83
6.1.1 Estimated Parameters for ORIFL190-3 83
6.1.2 Estimated Parameters for Fujinon Lens 85
6.2 Image Rectification 88
6.2.1 Calibration Setup 88
6.3 Image Registration and Mapping 94
6.3.1 Mosaic Samples 95
6.4 Summary 102

7 Discussion **103**
7.1 Answering the Research Questions 103
7.1.1 Question 2 103
7.1.2 Question 3 104

7.1.3 Question 4 . . . 107
7.1.4 Question 5 . . . 109
7.1.5 Question 1 — Summary . . . 110
7.2 Research Significance . . . 111
7.3 Summary . . . 112

8 Conclusion and Recommendations 113
8.1 Research Outcome . . . 113
8.2 Recommendations and Future Work . . . 115
8.3 Summary . . . 115

9 Acknowledgment 117

Appendices 119
A.1 Fujinon Fisheye Lens . . . 119
A.1.1 Measured Lens Points . . . 119
A.1.2 Lens Functions . . . 120
A.2 Measured Registration Errors . . . 121
A.2.1 Registration Errors for Statistical Approach . . . 121
A.2.2 Registration Errors . . . 123
A.3 Image Rectification Algorithm . . . 126
A.4 Satellite Systems . . . 127
A.4.1 Satellite Systems References . . . 127
A.4.2 Satellite Systems Table . . . 127
A.5 Aerial Photography Systems . . . 129
A.5.1 Aerial Photography References . . . 129
A.5.2 Aerial Photography Table . . . 129
A.6 Specification Sheets . . . 131
A.6.1 Omnitech Robotics ORIFL190-3 . . . 132
A.6.2 Fujinon YV2.2X1.4A-2 . . . 136

Bibliography 137

List of Abbreviations

AGL	Above Ground Level
AMSL	Above Mean Sea Level
AOI	Area of Interest
ARCAA	Australian Research Center for Aerospace Automation
BRDF	Bidirectional Reflectance Distribution Function
CASA	Civil Aviation Safety Authority
CCD	Charge Coupled Device
CNC	Computer Numerical Control
CPU	Central Processing Unit
CSIRO	Australian Commonwealth Scientific and Research Organization
DEM	Digital Elevation Map
ECEF	Earth Centered Earth Fixed
EKF	Extended Kalman Filter
FOV	Field of View

FPGA	Field Programmable Gate-Array
GIS	Geographical Information System
GPS	Global Positioning System
GPU	Graphics Processing Unit
GSD	Ground Sample Distance
HydroSHEDS	Hydrological data and maps based on Shuttle Elevation Derivatives at multiple Scales
INS	Inertial Navigation System
KML	Keyhole Markup Language
LLH	Longitude-Latitude-Height
MATLAB	Matrix Laboratory (technical computing software)
MEMS	Micro Electro Mechanical Sensor
MTOW	Maximum Take-Off Weight
NaN	Not a Number (MATLAB)
NED	North-East-Down
PNG	Portable Network Graphic
PPS	GPS Pulse Per Second
QUT	Queensland University of Technology
RAM	Random Access Memory

RPM	Revolutions Per Minute
SAR	Search and Rescue
SLAM	Simultaneous Localization and Mapping
SLR	Single Lens Reflex
SRTM	Shuttle Radar Topography Mission
SRTM-3	SRTM data at 3 arc-seconds
UAV	Unmanned Aerial Vehicle
USB	Universal Serial Port
USGS	United States Geological Survey
WGS84	World Geodetic System 1984

Nomenclature

Constants

f_{ORIFL}	Focal length of ORIFL190-3 fisheye lens ($f_{ORIFL} = 1.24\,\text{mm}$)	$^{\text{m}}/_{\text{pixel}}$
PS	Pixel size of CCD sensor ($PS_{Flea} = 4.65\,^{\mu\text{m}}/_{\text{pixel}}$)	$^{\text{m}}/_{\text{pixel}}$

Pseudonyms

Bayer Layer Color coding array; e. g. on CCD sensors, where RGB is represented by 2 green, 1 blue and 1 red sub pixels. (also called Bayer Filter or Bayer Array).

BestXYZ GPS position log from Novatel GPS receiver.

Collimator A light source, which produces parallel light rays, due to filters.

Conventional lens Lens corrected for distortions producing a rectilinear image (opposite to fisheye lenses, which don't provide a rectilinear image).

CrossBow MicroNav Low-cost inertial sensor and GPS receiver.

Flea Flea camera produced by Point Grey Research.

Fujinon Manufacturer of YV2.2x1.4A-2 fisheye lens: Fujinon (www.fujinon.com).

Gimbal system A mechanical system to allow for rotational movements (in almost any direction) from a single mounting point.

Goniometer An instrument to measure angles, such as a protractor.

Google Earth Earth terrain visualizer developed by Google (see also World Wind).

KML-file Google Earth file format to display geographical data.

Metadata Additional information provided for a data sample.

Mosaicing A process of using the image information of overlapping images and align the images with each others by feature recognition.

Nadir A point vertically underneath the observer (opposite of zenith).

Novatel Propak-G4 GPS receiver used in for the image acquisition (position and attitude estimation).

Omnitech Robotics Manufacturer of ORIFL190-3 fisheye lens: Omnitech Robotics International (www.omnitech.com).

PNG-file Portable Network Graphic, a graphics file format.

PSB632 Photogrammetry coursework for surveying at QUT.

Stitching An image processing algorithm to further improve the overlapping of two following images after the registration.

World Wind Earth terrain visualizer developed by NASA and using the Landsat image set (see also Google Earth).

Fisheye lens boundary The circle appearing on fisheye images separating the image from the lens case.

Symbols/Functions

$2Drms$ 2 dimensional RMS error

$atan2(y, x)$ Four quadrant arctangent in MATLAB

G Accumulated pixels $g_{i,j}$

$g_{i,j}$ Pixel at coordinate $(i; j)$ for the extraction grid (rectification)

$griddata()$ Grid data interpolation in MATLAB (Qhull)

$HT()$ Hough transform

LF1–LF8 Lens functions for the Fujinon fisheye lens

Units

$^\circ$ Degrees

lp/mm Line pairs per millimeter

fps frames per second ($1\,\text{fps} = 1\,\text{Hz}$)

kt	knots (aviation) ($1\,\mathrm{kt} = 1.852\,\mathrm{km/h}$)	
MP	mega-pixels (1024^2 pixels)	

Variables

$\mathbf{R}$	Rotation matrix	
$\mathbf{R_{90°}}$	2D rotation matrix for rotating an image by 90 °	
κ	Rotation angle z-axis	°
λ	Wavelength	m
ω	Rotation angle x-axis	°
ϕ	Rotation angle y-axis	°
ρ	Resolution of a digital image	pixels/°
Θ	The extraction FOV of a rectified image	°
θ	Elevation angle / viewing angle to fisheye lens	°
φ	Azimuthal angle of fisheye lens	°
f	Camera focal length	m
H	Flying height in AGL or AMSL	m or ft
pd	Principal distance	m
$ppmm$	Pixels per millimeter	pixels/mm

List of Figures

1.1 Comparison of roll behavior . . . 2
1.2 Comparison of pitch behavior . . . 3
1.3 Images from different lenses . . . 4

2.1 Illustration of mapping from small UAVs with fisheye lenses . . . 8
2.2 Spherical fisheye lens model . . . 10
2.3 Raytracing method using collimators . . . 11
2.4 Calibration disk with target pointer . . . 12
2.5 Available fisheye lens functions . . . 13

3.1 Fisheye lenses used in this research . . . 19
3.2 The perspective camera . . . 20
3.3 Non-linearity of the fisheye lens . . . 21
3.4 Setup for focal length and lens length evaluation . . . 22
3.5 Lens length measurements with a ± 1 pixel error . . . 24
3.6 Target pointer . . . 26
3.7 Calibration disk setup with target pointer . . . 27
3.8 Target pointer edge selection . . . 27
3.9 Calibration images used to derive the lens functions LF1–LF8 . . . 28
3.10 Illustration of the rectification process . . . 31
3.11 Extraction of the fisheye image horizon . . . 32
3.12 Roll gyroscope measurement . . . 35
3.13 Aerial image registration . . . 38
3.14 Camera installation on the Cessna 172 . . . 39
3.15 From camera mounting to map coordinates . . . 40
3.16 Using relief displacement to approximate point A . . . 41
3.17 Bounding box around registered image . . . 42

4.1 Illustration of the data acquisition system . . . 46

4.2 Flight path in the southern area of Brisbane 47
4.3 Camera installation underneath the wing of a Cessna 172 48
4.4 Cessna 172 plan 49
4.5 Installation of the data acquisition system in the Cessna 172 50
4.6 Computer interrupt delay 51
4.7 Measuring errors on ground features 52
4.8 Normal distributions of relative and absolute errors 53
4.9 Camera and image transmission timing 54
4.10 MicroNav clock timedrift 55
4.11 Zoomed illustration of MicroNav delays 56

5.1 Loose lens part as source of vibration 60
5.2 Translational or rotational vibration component 60
5.3 Translational shift verification test 61
5.4 Hough transform illustration on circle detection 63
5.5 Accumulator space for each of the tested radii 64
5.6 Different illuminated images show expanded borders 67
5.7 Sun-glare appearance as a vertical line through the image 68
5.8 Climbing aircraft merges horizon with circle boundary 69
5.9 Mounting and camera specific influences 70
5.10 Illustration of circle boundary selection within the radius range R . . . 72
5.11 Applying the Canny edge detector to an image 73
5.12 Accumulator spaces for all tested radii (left) and for a single radius (right) 74
5.13 Typical radius distribution before and after AND operation 75
5.14 Examples of the Hough transform 76
5.15 Vibration magnitudes of Δr, Δxm and Δym 77
5.16 Spread matrix showing the certainties of radii over a sequence of images 78

6.1 Verification of the measurements from the two diagrams 85
6.2 Proposed function approximation by Kannala 86
6.3 The 8 lens functions and $r_{general}$ 86
6.4 Deviations of the lens functions 88
6.5 Calibration images showing the grid pattern 89
6.6 Close-up of figure 6.8 90
6.7 Absolute error distribution 90
6.8 Grid points and rectified grid point overlaid over the rectified image . 91
6.9 A vector map showing the directions and magnitudes of the errors . . 92
6.10 Combination of vectors and absolute errors 93

6.11 Lens center displacement shown with the vector map 93
6.12 Taxiing on ground (white line) overlaid on Google Earth maps 94
6.13 Close-up of a scene (Woodridge, Logan Central) 95
6.14 Mosaic map over Redland Bay 97
6.15 Mosaic map close to Tingalpa Reservoir 97
6.16 Identifying objects to measure a relative across-track error 98
6.17 Close-up of figure 6.16: Across-track error of a ground feature 98
6.18 Mapping during climbing and turning after taking-off at Archerfield . . 100
6.19 Example image sequence during a turn at 30 ° bank at Beenleigh . . . 101

7.1 Relative and absolute error diagrams 106
7.2 Rectified image with a FOV of 160 ° 108

List of Tables

3.1 Measured pixels per mm at the specific height z 23
3.2 Lens center displacement 25

4.1 Mean and standard deviation of measured errors 54

5.1 Overview of differences of the left- and right-hand side 79

6.1 Measured points from datasheet diagram 84
6.2 Direct comparison of diagram F13 and F14 measurements 87

7.1 Mean and standard deviation of measured errors 105

A.1 Measured lens function points of the Fujinon fisheye lens 119
A.4 List of state-of-the-art satellites for remote sensing and earth observation 128
A.5 List of state-of-the-art aerial photography systems and services 130

In der Welt geschieht nichts, worin man nicht den Sinn eines bestimmten Maximums oder Minimums erkennen könnte.

Leonhard Euler, Swiss Mathematician, 1707–1783

Chapter 1

Introduction

The UAV (Unmanned Aerial Vehicle) market is in the process of emerging from a niche market into a widespread contributor to the field of surveillance. Although major use of UAVs has been in the military sector, the civil market is now experiencing steady growth [14, 83]. The growth of UAVs into civil markets has been slow due to a lack of compelling business cases to support their use and the increased regulatory restrictions placed on civil airspace operations. However, the Aerosonde UAV program represents a contrast to this with its civil remote sensing missions into cyclone activity (http://www.aerosonde.com/).

As UAVs are competing with equivalent piloted aircraft missions, they need to offer significant benefits over these businesses. One such benefit of UAVs includes their long endurance capabilities (>12 hours), which makes UAVs feasible for aerial photography missions. Another benefit, especially in the military sector, is health and safety. As UAVs are unmanned, the number of personnel exposed to risk is limited. A low flying UAV (<50 m AGL (Above Ground Level (Aviation))) can provide high resolution aerial imagery over a long time period without refueling. The crew operating the UAV from a ground station can be replaced at any time, without the need of aborting the mission. Additionally, multiple UAVs can fly concurrently in the same area to provide maps at shorter revisit times. With the steadily growing military market in such areas, more competing UAVs will become available in the civil market.

One problem of adopting off-the-shelf components into UAVs is their light construction method. Although UAVs as large as manned aircraft have been constructed, many new competitors are building small scale UAVs. Typically, a small UAV <30 kg MTOW (Maximum Take-Off Weight) is more susceptible to turbulence than a manned aircraft. Figures 1.1 and 1.2 compare the roll and pitch angle variation recorded by commercially available UAV autopilots by a QUT (Queensland University of Technology) project [reference confidential and therefore not open to the public. Please contact the authors for more details]. One was fitted to a small UAV, whilst the other

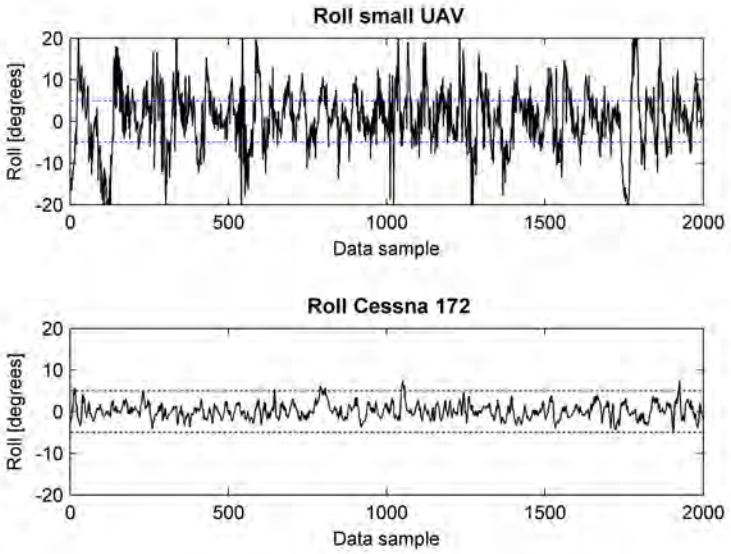


Figure 1.1: *Comparison of roll behavior*

was fitted to a larger and more stable Cessna 172. Both data sets were collected on good flying days and on periods when the aircraft was commanded for straight and level flight.

As can be seen from these figures, the small UAV is obviously more susceptible to turbulence in roll and pitch behavior. The axes in both figures show the same range. The dashed blue lines mark the maximum amplitude of roll and pitch of the Cessna 172 and are overlaid in the UAVs diagram. It can clearly be seen that the amplitudes of the UAV are 2–3 times larger than those of the Cessna. Even though small UAVs are cheap and have a very long endurance, they face challenges. Small UAVs have limited stability due to less mass and inertia than their manned counterparts. This stability, limited size and weight impact their suitability for aerial photography, due to their sensors being of a smaller size and reduced weight. Further, UAVs are often connected with real-time data-links and don't store collected data on-board. This is in contrast to the requested high quality acquisition of aerial images.

UAVs can provide a new range of airborne data acquisition, for example from low altitude, in dangerous environments or even inside buildings. As a UAV is unmanned, no personnel are exposed to risk and their small and lightweight construction decreases fatal hazards. The opportunity of providing low-altitude images can benefit from fisheye lenses, as the lens provides easier target tracking and situational awareness. High resolution image data from a selected target could be acquired by flying over it using the high resolution image center provided by the fisheye lenses. Such a target

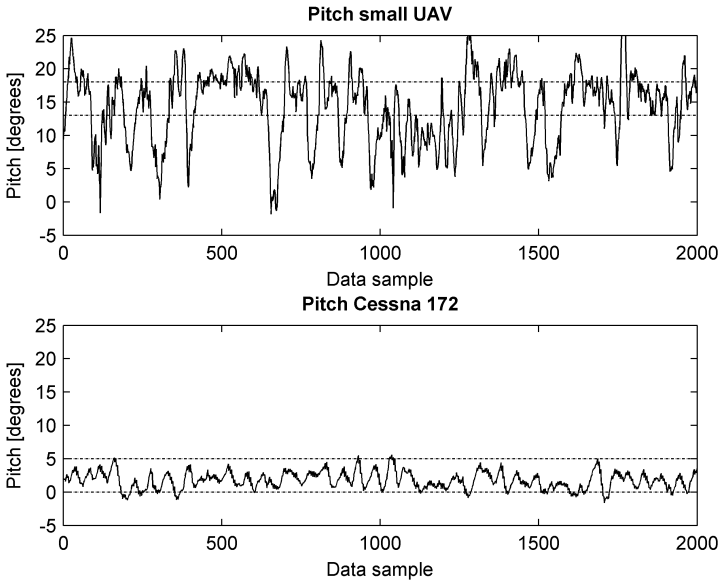


Figure 1.2: *Comparison of pitch behavior*

can be seen from the front, top and rear by looking through a fisheye lens whereas a narrow FOV (Field of View) camera would require a gimbal system. Increased situational awareness is seen in the large FOV where the horizon would always be partially visible and help support obstacle detection in front of the aircraft, potentially allowing attitude estimation using image processing techniques.

Especially narrow FOV cameras showed significant problems in providing any useful imagery for analysis purposes. The small UAV was used by the QUT project [reference confidential] to capture a video sequence of features on the ground. However, the video sequence shows significant problems of keeping the selected features within the FOV of the camera. This research therefore investigates the use of a fisheye lens as an optical sensor to acquire airborne imagery. A fisheye lens image and an image from a conventional camera is shown in figure 1.3. The fisheye lens provides a hemispherical image from the ground, including the horizon, whereas the conventional lens only shows a small area of the ground.

Fisheye lenses are facing issues of the trade-off between the large FOV and spatial resolution. As such, do the above benefits of fisheye lenses provide enough value to justify their use? The spatial resolution of fisheye lenses is smaller than narrow FOV lenses, but this can be adjusted by flying at lower altitudes. As such, the visible range decreases with a narrow FOV camera. However, this stays approximately the same with a fisheye lens, as the fisheye lens' FOV doesn't decrease linearly.

The hypothesis is that a fisheye lens in aerial photography could overcome some of

(a) Fisheye lens image (b) Conventional lens image

Figure 1.3: *Images from different lenses*

the limitations mentioned above, due to the large FOV of the fisheye lens. Although this video wasn't available at the beginning of this research, the fisheye lens has been seen to overcome issues of focusing on the area of interest. Previously problems of using narrow FOV cameras were reported by the Australian Coastwatch Trials [97], where in certain cases boats couldn't be found on the ocean. Hence a fisheye lens may improve the capabilities of existing and future UAV systems. In this regard, there are research questions that must first be answered:

1. What are the quantitative benefits of using a fisheye lens in small UAV applications?
2. To what accuracy can a fisheye lens image be rectified to represent an image taken by a conventional perspective camera?
3. What accuracy does an airborne fisheye lens image provide for registration to the ground using typical small UAV sensors?
4. What limitations are fisheye lenses exposed to and whether conventional lenses can outperform a fisheye lens?
5. Is there a justifiable trade-off between FOV and spatial resolution on-board small UAVs for aerial photography purposes?

Prior to discussing the methodology and results, it is pertinent to undertake a literature review to examine research previously undertaken in this area.

Chapter 2

Literature Review

In the introduction I stated that this research considers fisheye lenses in UAV aerial photography applications. I now establish the theoretical and historical background this area, image rectification, UAVs and image geo-referencing. This literature review is a high level approach to this research. Relevant publications are discussed in this chapter and referred to in chapter 3.

Although each field has been intensively researched, a fisheye lens for aerial photography purposes hasn't appeared in the literature this far. The literature review shows that a fisheye lens can potentially be used for aerial photography purposes, although there are limitations. These limitations are investigated throughout this thesis.

Please note that the vibration compensation in chapter 5 isn't mentioned in the literature review, because at the time the literature review was conducted this was an unforeseen problem.

Aerial photography with UAVs must integrate with two major fields of remote sensing:

Satellites: Satellites like GeoEye-1, Quickbird-2 or WorldView-1 provide image data at resolutions of up to 0.41 m GSD (Ground Sample Distance). Although satellites have high initial costs and lead times >5 years, they can provide remotely sensed image data in a cost effective manner. Limitations are the revisit times (re-observing the same location) and atmospheric influences. Revisit times are published as best cases in the range of 2–5 days, but will be larger due to the limited maneuver ability of satellites. The atmosphere limits certain light-bands constantly, while other atmospheric effects are dynamic and require compensation models.

Aerial Photography Systems: Leica Geosystems ADS40 or Jena-Optronics JAS 150s are state-of-the-art airborne imaging systems that provide high resolution image data at <10 cm GSD, compensate for the forward motion and capture data at rectangular color bands (see appendix A.5.2 for more details). However,

airborne manned photography systems require long lead times in bookings (typically 6 months) and costs are considerably higher than satellite systems (>5000 $ per flight hour). Further, manned aircrafts put personnel at risk, which can become an increasing insurance problem due to health and safety standards.

2.1 UAV Aerial Photography

Numerous studies refer to the use of UAVs for mapping applications [76, 103] or archaeological sites [8, 12] and for remote sensing applications [38, 68]. These reviews leave the reader with little doubt that aerial photography and remote sensing of high spatial resolution data collected from UAVs will form an important component of the future remote sensing industry.

UAVs aren't considered to compete against satellites and airborne manned aerial photography, but establish their own niche market. UAVs used, or proposed to be used, in aerial photography and mapping applications that can't be covered by the above systems, for example:

- Forest fire perimeter monitoring with small, low-altitude short-endurance UAVs, as presented by Casbeer et al. [19] or Zhou et al. [108]. These UAVs are proposed to support fire fighters with essential information at low risk.

- Flying into tropical cyclones for weather observation reported by Beven and Cobb [10].

- Jones and Earp [59] who remotely gather image data of ground objects, which is planned to replace high risk human observations by helicopters flying close to the ground and obstacles.

One problem of all these applications is the expectation to have a replaceable platform and therefore limit the costs per platform. Small UAVs are cheaper than the above options. However, as mentioned in the introduction, small UAVs are more susceptible to turbulence. This turbulence requires active compensation, such as a gyro stabilizer in combination with a gimbal system, to keep the camera tracking the object of interest.

The conclusion is that since UAVs are limited in payload and costs, such active stabilizers are out of range and other methods need to be considered as compensation for turbulence. A potential solution is the use of large FOV cameras providing a larger image of the surrounding area of an object of interest. Although the object of interest may move all over the image, motion compensation could keep it in a steady position.

At this stage fisheye lenses were believed to deliver such large FOV images at lower cost and weight than conventional large FOV lenses.

2.2 Fisheye Lenses

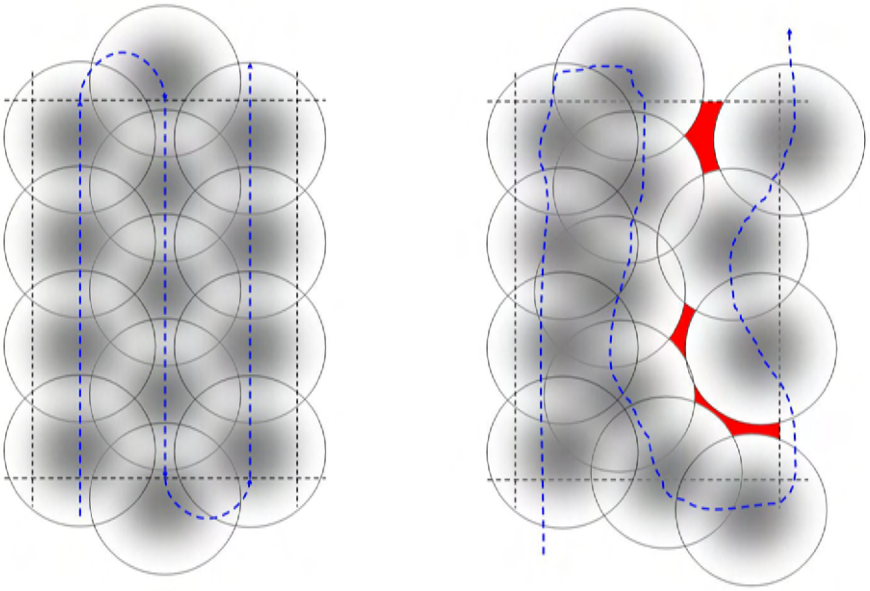
Fisheye lenses have been identified as an innovative approach to overcome limitations in aerial photography applications. Therefore, this research uses sensors typically mounted in small UAVs for aerial photography. A fisheye lens mounted as a downward looking camera on a UAV shows the entire horizon, including parts of the sky. The lens provides a forward, side and backward look at the same time.

However, since the 1950s the use of fisheye lenses has been traditionally limited to scenic, panoramic and artistic photography. From a technical perspective attempts have been made to use the fisheye lens in robotics applications [26, 70, 107], with only few attempts to use the fisheye lens in airborne imaging sensors. An airborne application using fisheye lenses has been shown by Hrabar et al. [54], where stereo cameras with fisheye lenses are used to navigate an autonomous helicopter through an urban canyon. Although the system can estimate distances of approaching objects and avoid them, there were no attempts of using these sensors to register images and create maps.

Demonceaux et al. [32] used a catadioptric camera to perform an attitude estimation in UAVs. A catadioptric camera uses, similar to a fisheye lens, a large FOV, but projects the image to a mirror instead of using a lens. An image acquired by a catadioptric camera can provide a similar image to a fisheye lens, but shows the camera in the center (or parts) of the image and can therefore not replace a fisheye lens for mapping purposes. Regrettably, Demonceaux et al. don't provide any performance of his attitude estimation, due to the unavailability of a ground truth, even though he estimated approximate errors of 3–5 °.

Although the fisheye lens can now be seen as a way to overcome the low stability of small UAVs, the distortion and spatial resolution of the lens requires further investigation. The spatial resolution will be significantly lower for fisheye lenses, compared to narrow FOV cameras. For example, the angular resolution of a fisheye lens is approximately 4.78 pixels per degree, compared to 51 pixels per degree for a 20 ° FOV lens, mounted on a 1024×768 pixel sensor. Further, a distortion in the image requires compensation if the images are to be used for aerial photography purposes.

These drawbacks can be somewhat overcome through the use of a fisheye lens. That is, the overlaying of images on a map of a region of interest, such as in figure 2.1, can leave gaps in the data due to turbulence (red areas). In this example the UAV is given a mission plan (figure 2.1(a)) to fly 3 straight strips along the dashed blue



(a) Mission plan (b) Reality

Figure 2.1: *Illustration of mapping from small UAVs with fisheye lenses*

line and collect images at the center of the circles to cover the entire area. The small UAV is exposed to turbulence while it performs maneuvers and this mission might look like figure 2.1(b). Taking the same amount of pictures at respective positions leads to uncovered areas (red) by the aerial photography. A narrow FOV camera wouldn't be able to retrieve image information from the uncovered areas whereas the fisheye lens provides such information, although at the cost of lower spatial resolution. The extraction FOV for the rectification process would be increased and image information from uncovered areas is made available.

As stated in the introduction, a small UAV can be flown close to the ground to adjust the spatial resolution on an application specific basis. The fisheye lens has a non-linear behavior in the case of the relationship between the altitude and spatial resolution and therefore profits from features like the horizon staying visible in the fisheye image. The fisheye lens' FOV stays almost constant compared to a narrow FOV lens, where the FOV Θ is proportional to the altitude H of the aircraft: $\Theta \sim H$. Furthermore, the fisheye lens provides additional information of the surrounding area of the UAV, even though it is flying at low altitudes. This increased situational awareness is a benefit no conventional lens can provide. Rectilinear ultra-wide angle lenses are technically limited to typical FOVs $<120\,^\circ$ [56]. This approach of using a low flying UAV instead of installing a conventional lens on a gimballed system allows a saving of

payload, space and cost, which are critical in small UAV applications.

Another benefit of fisheye lenses is their ability to provide aerial images even when the aircraft performs turns. Although the lens center with the highest spatial resolution will point outside the area of interest, the nadir position of the aircraft remains in the FOV of the camera. A conventional gimballed camera-lens system would have to turn the camera and using the aircraft's energy supply.

In conclusion, the fisheye lens has limitations in spatial resolution which can be overcome by adjusting the flying altitude to increase the spatial resolution. The fisheye lens can therefore be adjusted to similar spatial resolutions as conventional lenses. In return for this drawback, the fisheye lens always provides a hemispherical FOV of $>180\,°$. On a downward looking fisheye lens used for aerial photography the lens shows the entire horizon, including parts of the sky. Investigations will show that these image features can provide important situational awareness that no other lens system can provide at the same weight, space and cost.

2.3 Fisheye Image Rectification

Using a fisheye lens for aerial photography requires precise knowledge about the lens model and the lens distortion. Kannala and Brandt [60] presented a generic camera model for conventional and wide angle lenses, including fisheye lenses. It provides the lens function as a function of the incoming viewing angle θ, illustrated in figure 2.2. The model is based on a sphere, which isn't the model itself, but allows the determination of the incoming viewing angle as an elevation angle. The lens model proposed by Kannala and Brandt is shown in figure 2.2, with the object plane (world) on the bottom and the image plane (photo negative) on top. The point P_w represents an object which is then projected to the image plane in P_i by using the lens function $\mathrm{r}(\theta)$. The incoming viewing angle θ is the angle (elevation) an observer would see the object in point P_w. The lens model assumes that the azimuthal angle φ isn't affected by the lens distortion. However, this was found to be incorrect. The lens distortion function should instead be a function of θ and φ, where the current assumed image radius $r(\theta)$ is replaced by $r(\theta, \varphi)$. Investigations towards this have been taken by using multiple lens functions instead of one lens function predicted by the lens model, as will be presented in section 3.1.1.

The lens function $\mathrm{r}(\theta)$ can be freely chosen for this model. Kannala and Brandt identified the lens function in equation (2.1) as a unified valid function for any lens. However, the least square curve fitting had problems in fitting the lens distortion functions. Therefore, the lens function was replaced by a polynomial (2.2) instead. Other lens distortion approximations as presented by Devernay and Faugeras [33], Basu

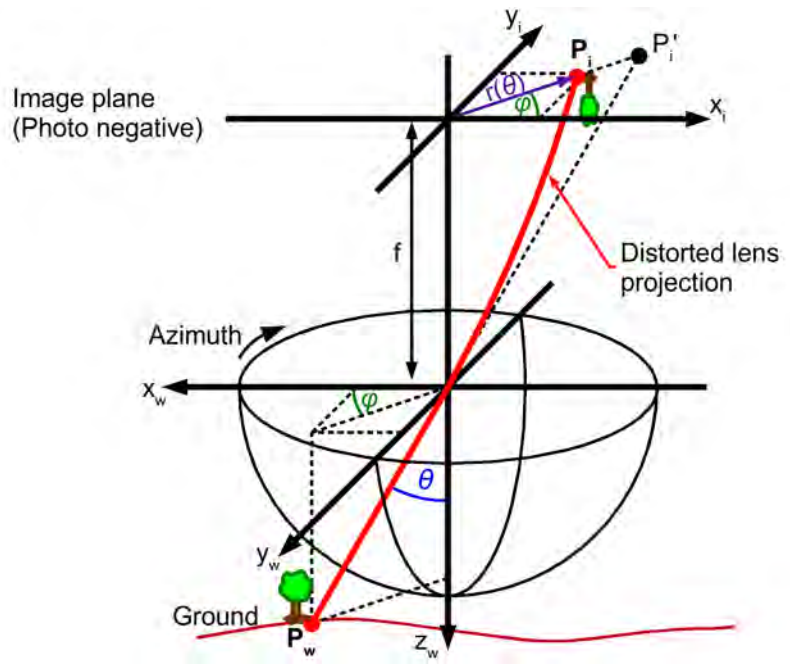


Figure 2.2: *Spherical fisheye lens model*

and Licardie [7] or Hansen [47] aren't appropriate for aerial photography purposes, because image points aren't accurately projected to coordinates or the lens model isn't appropriate for a fisheye lens modelling.

$$r_{Kannala}(\theta) = k_1 \cdot \theta + k_2 \cdot \theta^3 + k_3 \cdot \theta^5 + k_4 \cdot \theta^7 + k_5 \cdot \theta^9 + \ldots \tag{2.1}$$

$$r_{poly}(\theta) = p_1 \cdot \theta^4 + p_2 \cdot \theta^3 + p_3 \cdot \theta^2 + p_4 \cdot \theta + p_5 \tag{2.2}$$

Zhao and Aggarwal [106] presented a scene reconstruction from synthetic fisheye images. They used stereo vision to estimate depth in the fisheye images and applied a rectification process to remove the lens distortion. Recently Nishimoto and Yamaguchi [82] published a 3 dimensional measurement from fisheye images. They demonstrated the height detection of passing objects, such as cars or bikes, in images obtained from a fisheye lens camera mounted above a street. Unfortunately, they don't present accuracies achieved by using the simplified lens function $r = f \cdot \theta$. I found that $r = f \cdot \theta$ can approximate the lens function to 1 % at small angles (<30 °), but shows errors of >10 % for larger angles (for more details, see section section 6.1.1). Zhao and Aggarwal used a theoretical approach with $r = c \cdot \theta$, where c is a constant, derived from the focal length. The conclusion for both publications is that they show possible ways to measure distances and depth in fisheye images. However, both lack a practical

application [106] or don't show precise results [82]. With the available lenses, however, I was confronted with further problems.

The lens function for the Omnitech Robotics ORIFL190-3 lens is available from the manufacturer, whereas the lens function for the the Fujinon lens wasn't published. Although the lens function of the ORIFL190-3 is only given in diagrams, it was possible to derive it and verify it with a second diagram, presented in section 6.1.1. However, the lens function for the Fujinon lens was required to be derived from the lens itself. Techniques to derive lens functions are described by Clarke and Fryer [24] and Heikkila [50]. Heikkila uses an image processing approach with precise calibration setups, which was identified to be too cost intensive for this research of using low-cost components. Although Clarke and Fryer's raytracing method requires even more precise equipment, I decided to follow this approach using a simplified version of it. This raytracing method is illustrated in figure 2.3, where collimators arranged in a circular manner send out a light ray towards the lens front. On the rear, the light rays are displayed on a focal plane. The idea is to measure the collimators position in relationship to the light ray appearing on the focal plane, which can then be used to derive the lens distortion.

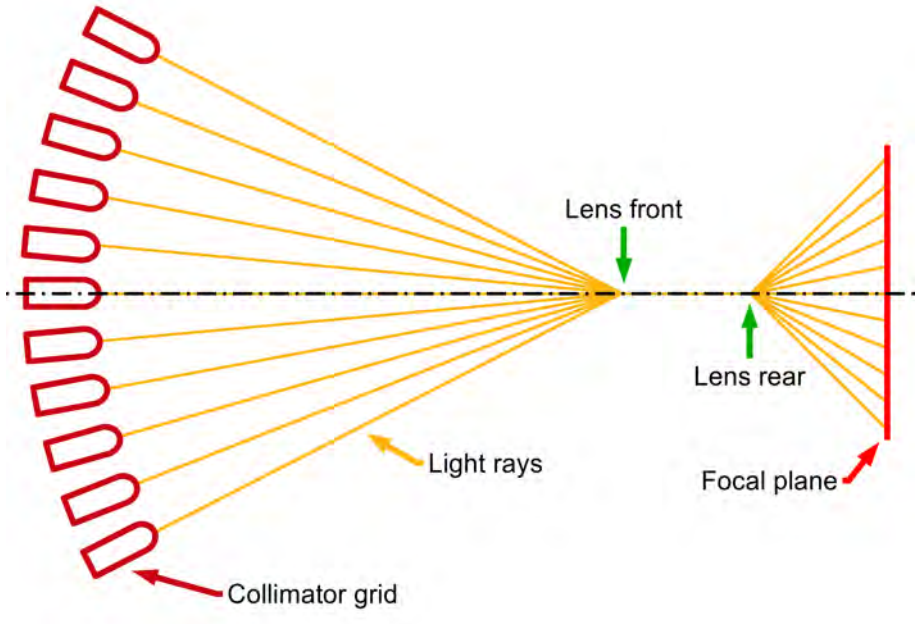


Figure 2.3: *Raytracing method using collimators*

The simplified method used is replacing the collimators with a target pointer and mounting the camera to the lens, instead of using a focal plane with precise measuring equipment. The target pointer should then be seen in the cameras obtained images

at the locations a collimator would be placed to instead, as illustrated in figure 2.4. This approach using a target pointer in a circular arrangement also correlates with the spherical lens model from above, which allows for angles $\theta > 90°$ as encountered in fisheye lenses. The calibration disk would provide position holes at every 5 °, to ensure the target pointer is set precisely. A conventional calibration, where a flat ruler is placed opposite the lens, would only allow to test for angles <90 ° and has been identified as not suitable to derive a fisheye lens distortion function.

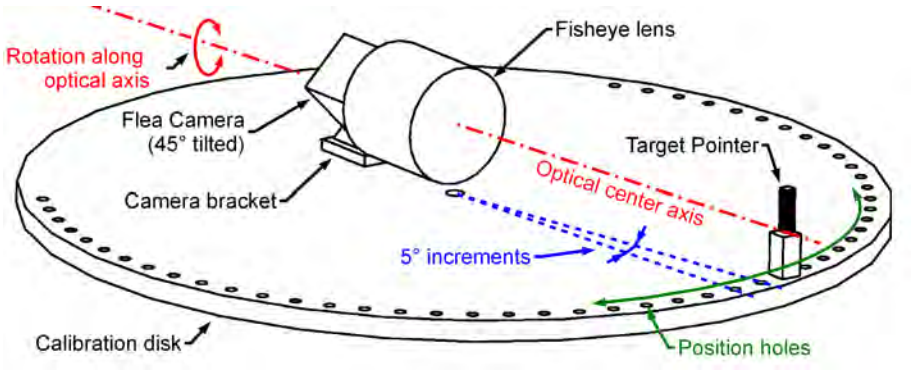
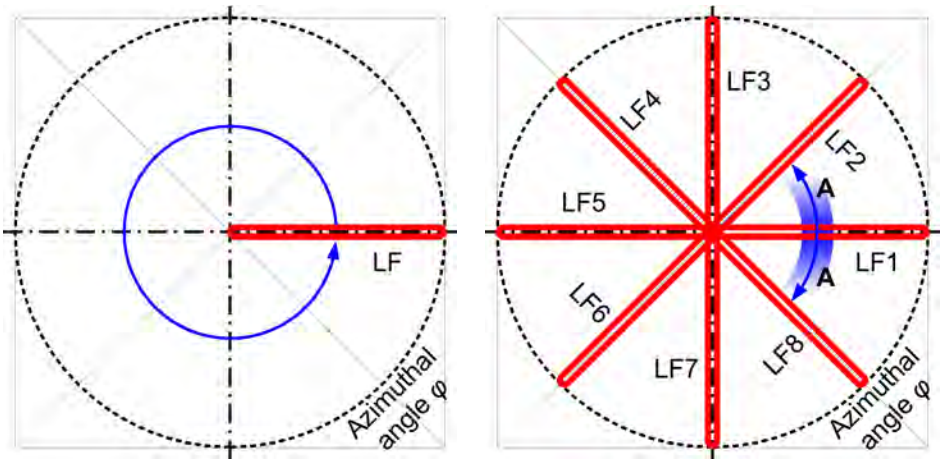


Figure 2.4: *Calibration disk with target pointer*

As mentioned in the previous paragraph, multiple lens functions were derived to account for azimuthal distortions. The calibration disk itself doesn't allow for this, but a special bracket was made to turn the camera along the optical center axis in steps of 45 °, which resulted in 8 different lens functions LF1...LF8, illustrated in figure 2.5. The results presented in section 6.1.2, show that the azimuthal divergences of the lens function are 5 pixels at a FOV of 40 ° and 20 pixels at 80 °.

It was found that Kannala and Brandt lens model, in combination with the lens distortion as a function of θ and φ, provides an accurate projection of image points to coordinates for even low quality lenses, such as the Fujinon lens. However, the Omnitech Robotics ORIFL190-3 lens is of better quality and could therefore be used with just a single lens function given by the manufacturer. An accurate lens model is essential in aerial photography, where deriving coordinates together with an accurately rectified image is required. Further, the quality of measuring a lens distortion is a function of the available equipment.

A further discussion of methods to derive lens parameters is given in section 3.1. Section 3.1 presents the estimation of the focal length, lens center and lens functions in more details. Further, it explains why the lens length couldn't be determined by using the available equipment, as the Flea camera doesn't provide a high enough angular resolution.



(a) 1 lens function (Omnitech Robotics) (b) 8 lens functions (Fujinon)

Figure 2.5: *Available fisheye lens functions*

2.4 Image Registration and Mapping

Image registration is one of the most researched areas since it is under permanent competition of geo-referencing companies, such as Leica Geosystems, Jena-Optronik, Microsoft Vexcel or Intergraph. The registration in this thesis therefore follows the instructions by Wolf [102], which provide a complete description of photogrammetry in airborne geo-referencing applications.

Wolf's book provides information about image registration, photography and cameras in sections "Principles of Photography and Imaging", "Cameras and Other Imaging Devices" and "Image Measurements and Refinements". "Aerotriangulation" builds the basis of the image registration, which uses the theory of relief displacement to estimate the accurate positioning of the image corners, while considering the terrain elevation.

However, the inner image orientation is unique to each aerial photography system and must be considered as a new problem each time. Textbooks like Wolf [102] provide general information about this coordinate transformation and will be explained in detail in section 3.3.3.

The registration itself is a simple approach once the rotation angles and the inner orientation are available. Issues caused by the rotation angles, which depend on the accuracy of the inertial sensors, were found to be limited to an accuracy of approximately 3.1° (1-sigma) in roll and pitch and 5.8° (1-sigma) in yaw (see section 3.3.1). Little research has been conducted in this area regarding small aircrafts or UAV mapping applications where attempts were made to overcome such problems. An outstanding exception is the publications by Lin et al. [74, 75], where they use underlaying Google

Earth maps to enhance the registration of aerial photos collected by UAVs. Since there are already existing maps from most places available (especially in Google Earth or World Wind), this was an excellent idea.

Further improvement in the image alignment is achieved by using mosaicing. Mosaicing is a process of using the image information of overlapping images and align the images with each other by feature recognition. Ramachandran and Chellappa [86] demonstrated mosaicing of airborne images from a low-cost camera, similar to this research, where the images were initially registered using inertial sensor data. They then refined the mosaicing process by using image processing techniques, such as phase correlation and optic flow, to further align the images with each other.

Interestingly, Lin and Medioni's demonstration also used the Google Earth system to register their images. This was also identified during this research, as Google Earth is a readily available platform for presenting geo-referenced data.

2.5 Collision Avoidance

Using the fisheye images in collision avoidance for aircrafts was briefly investigated, due to previous research by a colleague. Collision avoidance has been previously researched by Carnie et al. [18], using a narrow FOV camera. He tried to achieve a similar traffic detection performance as a human pilot of potentially colliding aircrafts. An aircraft detection has been introduced based on a pixel-size basis using raw Bayer layer image data. The images were captured at 30 fps and the traffic was tracked over several frames to improve the detection certainty. A comparison was undertaken with the estimated performance of a fisheye lens. The conclusion was that a fisheye lens provides an angular resolution of 4.78 $^{\text{pixels}}/_{\circ}$ on a 1024×768 pixel (0.8 MP) sensor, which is approximately 10 times worse than the narrow FOV camera ($FOV = 20\,^{\circ}$,51 $^{\text{pixels}}/_{\circ}$). Considering a higher resolution sensor would require a CCD (Charge Coupled Device) sensor of >78 MP. However, the near future won't be able to provide such imaging sensor at affordable prices. State-of-the-art CCD sensors reach 38 MP, such as from DALSA Sensor Solutions (www.dalsa.com), but no SLR (Single Lens Reflex) or compact cameras potentially be used in small UAVs use such sensors.

The conclusion is that the fisheye lens mounted on an off-the-shelf low-cost camera isn't capable of providing an angular resolution high enough to detect other aircrafts at the required time.

2.6 Identified Benefits

The fisheye lens was identified as a feasible approach towards aerial photography and providing benefits compared to conventional lenses. The current CASA (Civil Aviation Safety Authority) regulations in Australia (CASR Part 101 [3]) encourage the use of small UAVs (<150 kg) to operate at altitudes below 400 ft without the need for certification. However, turbulence also increases closer to the ground [67], and this is further increased due to the light construction of small UAVs, as shown in the introduction. Therefore, small UAVs require new approaches towards sensor systems, such as the identified fisheye lens. The fisheye lens can potentially provide benefits over conventional lenses through:

- Resistance to turbulence due to a larger FOV without the need for a gimballed camera system.
- Using the same imaging processing hardware for multiple tasks, such as horizon monitoring, target tracking and aerial photography.
- Potentially providing more information in images due to forward, sideward and backward looking capabilities.

This is a promising outlook and further investigation into fisheye lenses and their performance in aerial photography applications, was undertaken.

2.7 Summary

This literature review shows that the use of a fisheye lens for aerial photography and mapping is possible. A fisheye lens image can be projected to coordinates, which on their part can be used to register a rectified fisheye image. Application measuring distances within fisheye images have been previously presented [1, 82], although none of them presented real applications with corresponding accuracy achievements. Based on the identified potential benefits of using a fisheye lens on a small UAV and given that this topic hasn't already been investigated, the decision was to proceed with this investigation. Risks with this research had been identified in obtaining accurate lens distortion functions and registering images with low quality inertial sensor data. Although the low quality of attitude estimation from inertial sensors has been identified, there is only one published way to approach to address this issue by using image processing steps to improve mosaicing. Since this area has been extensively researched, this thesis doesn't apply any of these techniques, but rather investigates the performance of the registration with inertial sensor data only.

Chapter 3

Approach to the Research

As concluded in the previous chapter, this research investigates the use of fisheye lenses in aerial photography. It was decided to approach it from the bottom up, meaning the fisheye lens had to be understood before being used for further purposes. This research is approached by evaluating basic parameters fisheye lenses provide and determining whether they can be used in similar ways as conventional lenses. The findings are that a rectified fisheye image can be considered equal to an image taken from a conventional lens, with the exception that virtual lens parameters describe the scale of the image.

The lenses used for this purpose were an Omnitech Robotics and a Fujinon fisheye lens. Due to the cost factor impacting this research, I used an Omnitech Robotics lens that was available from CSIRO (Australian Commonwealth Scientific and Research Organization). CSIRO is Australia's Commonwealth Scientific and Industrial Research Organisation and collaborates with QUT. At a later stage, this lens was no longer available and a Fujinon lens was acquired instead. The structure and operation both lenses were thoroughly investigated.

Using a fisheye lens distorts the image projection, which needs to be corrected for aerial photography. The first step is to understand the lens itself and derive a lens model. The model requires certain lens parameters, which are measured in section 3.1. The proposed methods allow the estimation of the most important lens parameters with your own measurements, without having the lens specifications available. These parameters are then used in the lens model and for the rectification process described in section 3.2.

After the fisheye image has been rectified and extracted it can be seen as a conventional image taken by a conventional lens. This rectified image is then registered in such a way that multiple pictures can be stitched together to create a map. The registration process described in section 3.3 transforms the image according to the aircraft's attitude and position to an image with a known scale and correct projection. The corrected projection is as the observed place would be seen by a camera installed

vertically above the ground (in the zenith).

This chapter therefore describes the technical aspects of this research. It describes the algorithms used to perform image rectification, registration and mapping. A more analytical perspective will be given in the discussion in chapter 7.

3.1 Fisheye Lens Calibration

Image registration requires precise knowledge of the underlying hardware, such as the optical lens. An optical lens in photography is a complex internal structure to refract light to create a virtual image from a real world scene. The complex internal structure simplifies the external physical description, known as the perspective camera model. Photogrammetry is based on the perspective camera model. Fisheye lenses, however, don't conform to the idea of the perspective camera model. In this section, I present the transformation of a fisheye image, including calibration methods, fisheye lens models and fisheye image rectification. Although the rectification of the distorted fisheye image looks like it loses image quality, the quality loss is actually due to a spreading of concentrated image information to a larger area.

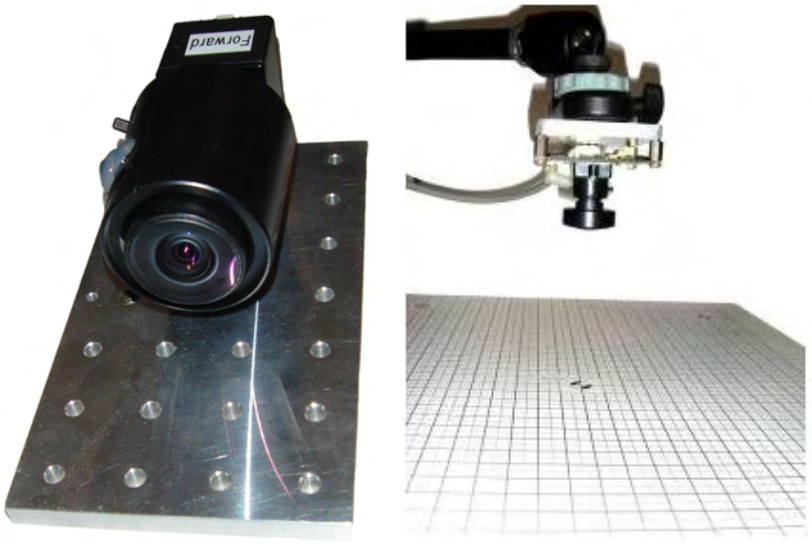
The fisheye lenses used for this research are shown in figure 3.1. These lenses were used due to availability and are typical low-cost lenses used in small UAV applications. Both lenses provide a FOV of larger than 180 ° and fit on the camera (Point Grey Research: Flea camera) available in our laboratory. The data sheets of the lenses are attached in appendix A.6.

The following sections describe in detail the lens parameters and how these parameters were validated. Following this, section 3.1.1 describes the method used to derive the lens functions for the Fujinon lens with the calibration disk using the spherical lens model, as mentioned in the literature review (section 2.3). The rectification in section 3.2 will use these lens functions to rectify parts of the distorted images gathered with the fisheye lens.

3.1.1 Fisheye Lens Parameter Estimation

Measuring lens parameters is required to obtain parameters of the lens or to confirm parameters given by the manufacturer. The use of two different lenses during this research showed wide differences in quality and the amount of given parameters. The rectification process in section 3.2 requires at least the focal length to determine real coordinates from the fisheye images.

The fisheye lens provided from Omnitech Robotics has given parameters, which are confirmed in section 6.1. This Omnitech fisheye lens is of good quality and the



(a) Fujinon YV2.2X1.4A-2 (b) Omnitech Robotics ORIFL190-3

Figure 3.1: *Fisheye lenses used in this research*

measured parameters confirm the manufacturer's specifications. In contrast, the lens provided from Fujinon doesn't always match the data sheet specifications. Less information is available from the Fujinon as for the Omnitech Robotics lens. Information about the lens distortion is also missing. A considerable amount of time was spent in measuring and confirming these parameters, as they influence all further work based on these lenses. The following sections describe the methods used to measure the fisheye lens parameters.

Focal Length

The focal length is the basic parameter of every lens and describes the distance between the lens and the sensor plane. Lenses only focus at the distance of this focal length, which represents a projection of the scene that can be interpreted by a human eye. In literature, the focal length is also known as the principal distance. The focal length is determined by using the perspective camera equation:

$$\frac{f}{d} = \frac{f'}{d'} = \frac{H}{D} \tag{3.1}$$

with $f = f'$ being the focal length, H the distance of the lens to the object, D the real object size and $d = d'$ the image object size, as illustrated in figure 3.2.

This approach is used for conventional lenses and was found to work on fisheye lenses as well, even though they distort the image. To overcome this problem, it was

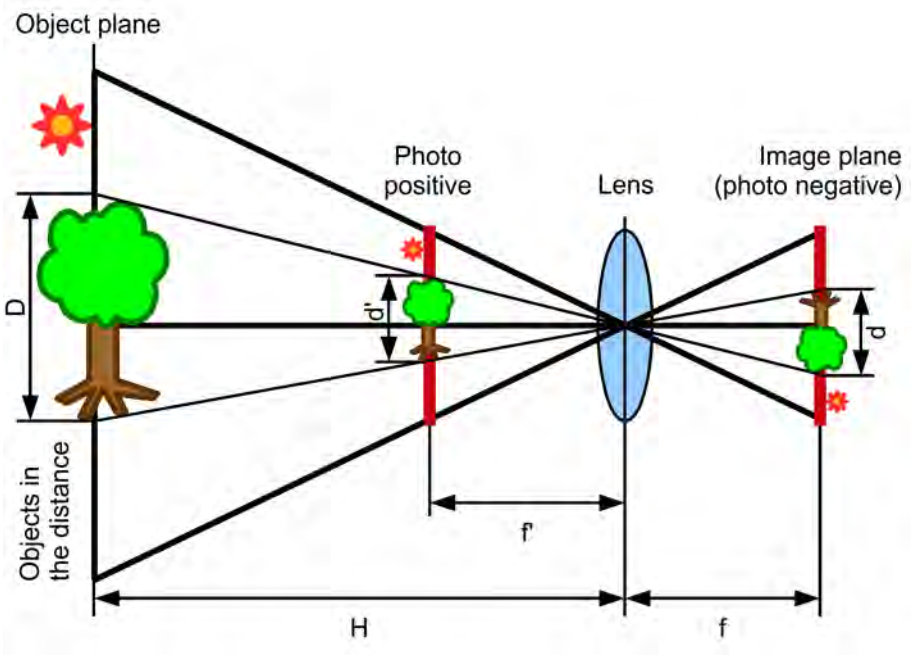


Figure 3.2: *The perspective camera*

assumed that the lens distortion is linear close to the center of the lens. As shown in figure 3.3, the lens shows a linear behavior between ±20 ° from the center. Although the figure looks more like a curve, the lens distortion varies only between 29 to 30.5 pixels per 5 degrees within ±20 °. Therefore, only a small amount of the full range of the CCD sensor is used. This approach is legitimate considering that involving the lens distortion would be problematic as well, since the distortion function isn't available at all or only at a low precision (see section 3.1.1).

The camera with the fisheye lens was installed opposite a ruler pattern and all distances were measured manually. Problems using this method are the accuracy of the measurements. Since the sensor is of a "digital" format (in terms of image pixels), the measuring accuracy on the image senor is limited to the pixel size. To get an accurate measurement, the distance between the camera and the ruler should be at least 100–1000 times higher than the estimated focal length (inside the camera), to ensure that measurement errors of the distance H don't influence the result. If the distance is smaller the resulting measurement errors on the image (which stay constant) would result in an inaccurate focal length estimation.

Furthermore, the lens length (or distance between the principal distance (see figure 3.4 for more details)) for the Omnitech Robotics ORIFL190-3 influences the measurement at small distances of H. The lens length is given by the manufacturer, but it is unknown if this distance is the lens case length or the distance between the inner

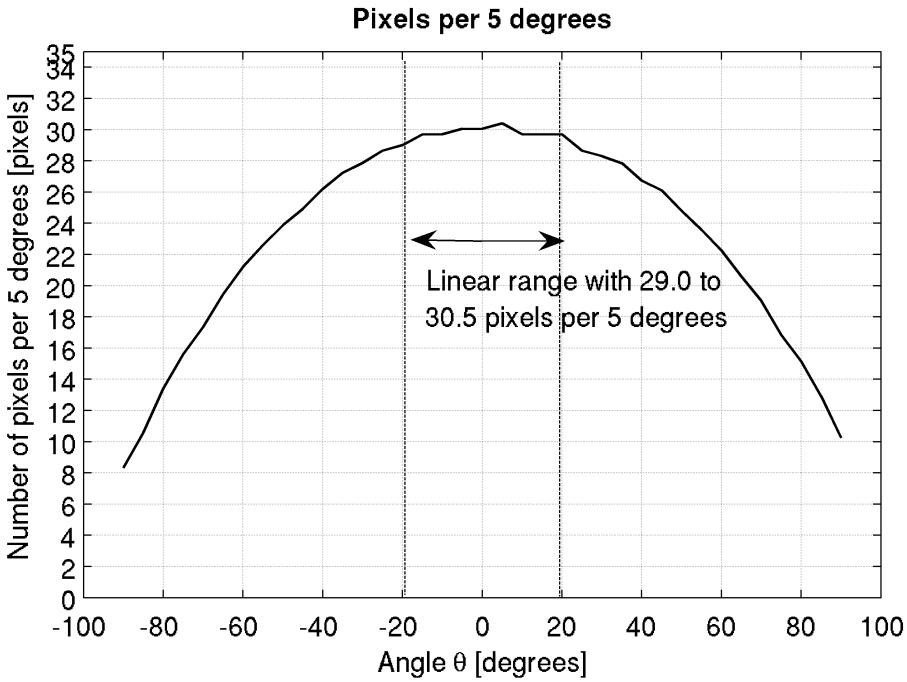


Figure 3.3: *Non-linearity of the fisheye lens*

and outer principal distances. The lens length couldn't be verified, as described in the following section. Figure 3.4 shows an illustration of the calibration setup, with an exaggeration of the lens length, as it was used to determine the focal length.

Two measurements at different distances of H were conducted to estimate the focal length, finding that the focal length is between 1.2411 mm and 1.26 mm. The measurement was conducted according to figure 3.4, using the Flea camera as the focal plane and a ruler placed at distances of 3.5 m and 6.4 m. These measurements are within an accuracy of 1.6 %, considering that a 1 pixel error would result in an error of >2 %. Further, the focal length of the lens also depends on how the lens is screwed into the thread. In the case of the ORIFL190-3 lens, the lens is screwed in by hand until the image appears in focus. However, the lens appears to be in focus over a range of a 1/4 turn of an ISO 12 mm×0.5 mm thread, resulting in an error of 0.125 mm or 10 % of the focal length.

In conclusion, the focal length can be determined by using conventional lens methods. However, the lens thread is a concern, as the user has to manually screw in the lens until focused. A proper lens thread, such as C- or CS-mount, removes uncertainties introduced by the user, by providing a limit stop. The ORIFL190-3 lens provided only a manual thread, while the Fujinon lens provides a CS-mount thread.

Lens Length

In this section I attempted to determine the lens length to confirm the manufacturer's specifications. It was found that this isn't possible by just using a low resolution camera, like the Flea camera used in this research.

The principal distance is equivalent to the focal length and is measured from the inner principal point to the focal plane. As shown in figure 3.4, a lens has two principal points, an inner and an outer principal point. The lens length is measured as the distance between these two principal points, where PP_{inner} and PP_{outer} mark the inner and outer principal distance:

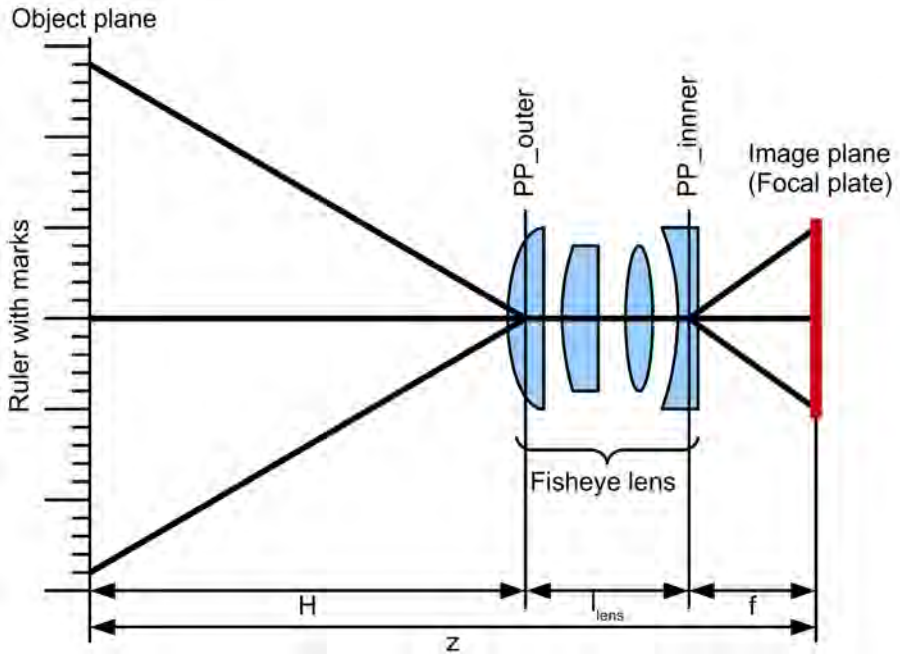


Figure 3.4: *Setup for focal length and lens length evaluation*

Often this distance is ignored if $H \gg l_{lens}$. Since the lens length l_{lens} itself is given in the data sheet, I attempted to estimate this distance as well. The lens length can be calculated as:

$$l_{lens} = PP_{inner} - PP_{outer} = z - (H + f) \tag{3.2}$$

Combining equation (3.1) with equation (3.2) we get equation (3.3), which is used to determine the lens length. D is set to $D = 1\,\text{mm}$ and d is determined using equation (3.4) with the values listed in table 3.1. The pixel size is taken from the data sheet as $PS_{Flea} = 4.65\,{}^{\mu\text{m}}/_{\text{pixel}}$. The table shows the resolution ($ppmm \mathrel{\widehat{=}}$ pixels per millimeter) measured at different camera heights z, with an expected decrease in the resolution

Table 3.1: *Measured pixels per mm at the specific height z*

Resolution *ppmm* [Pixels/mm]	Height z [mm]
19.20	38
7.70	58
5.45	75
4.25	88
3.70	98
3.55	102
3.45	105
3.30	108
2.45	133

with increasing camera height.

$$l_{lens} = z - f \cdot \left(\frac{D}{d} + 1 \right) \tag{3.3}$$

$$d = ppmm \cdot PS_{Flea} \tag{3.4}$$

Figure 3.5 shows the results from table 3.1 in a diagram, with the resulting estimates for the lens length l_{lens} as a function of the camera height z. The bold black **x**'s mark the measured values with a potential measurement error of $\pm$ 1 pixels shown as a vertical range in red. As a comparison, the lens length given by Omnitech Robotics is shown with a horizontal dashed blue line at $l_{lens} = 25.88\,\text{mm}$.

It was concluded as a result of this test that it wasn't possible to precisely estimate the lens length and it couldn't be confirmed with the manufacturer's specification. The issue is that the lens length requires a more accurate method of measurement. These measurements aren't possible with a low resolution CCD sensor (1024×768 pixels), as used with the Flea camera.

Lens Center Displacement

This section describes the estimation of the optical center axis of fisheye lenses. Three methods were identified, but only one is recommended to estimate the lens center displacement. A lens will always be displaced to the center of the sensor, but in case of conventional lenses this is often irrelevant, as the lens provides a rectilinear image. The displacement becomes relevant in combination with fisheye lenses, as the rectification will use a function $r(\theta)$ to determine the lens distortion. The lens function

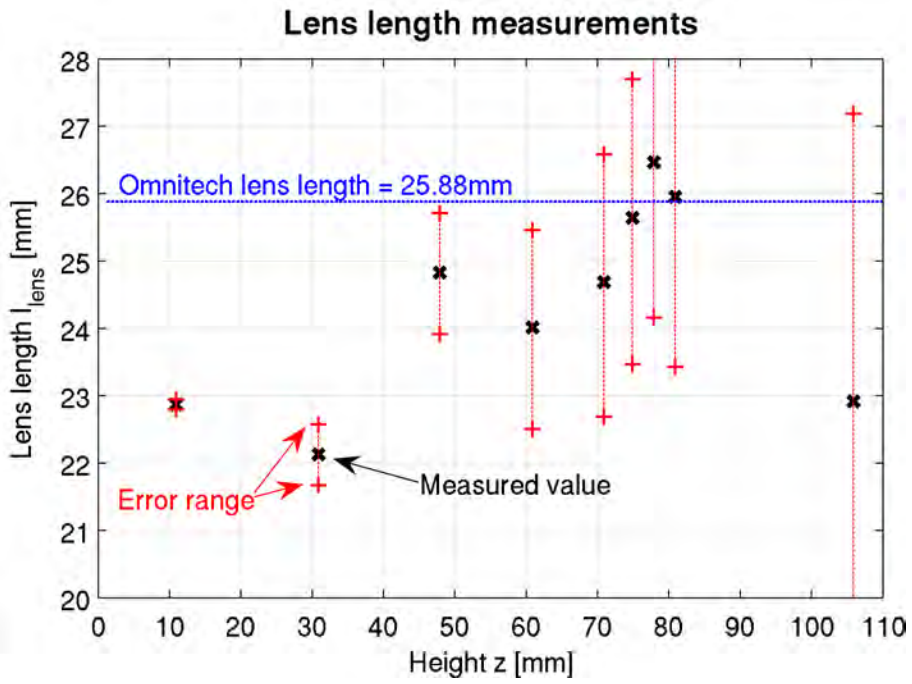


Figure 3.5: *Lens length measurements with a ±1 pixel error*

$r(\theta)$ requires to be placed accurately to the lens' optical center.

The three methods are as follows:

1. The center of the lens is measured by using the fisheye images and placing a circle along the fisheye image boundary. This is the circular edge dividing the fisheye image and the black areas (see section 5, for more details).

2. The lens distortion function is used to determine the center of the lens. The center would be derived from the slope of the distortion, where a slope of 0 would show the center. For example, the lens center could be derived from figure 3.3. However, this method isn't recommended as the lens distortion doesn't necessarily have to align with the optical center (see figure 3.3), and measuring the lens distortion itself requires more accurate measuring methods and equipment than was used for this research.

3. Another potential method for center estimation is the use of a method like Aggarwal and Ahuja [1] describes, where a checker board in front of the camera is used to estimate the lens center displacement. Aggarwal and Ahuja don't mention fisheye lenses (or lenses with a non-linear behavior). As such, this would have to be researched before this method could be used for fisheye lenses.

Table 3.2: *Lens center displacement*

Fisheye lens	x-axis [pixels]	y-axis [pixels]
Omnitech Robotics ORIFL190-3	−3.0	−1.0
Fujinon	+15.5	−25.5

The lens center was measured using method one and it was found that the lens centers are displaced according to the following table, where the x- and y-axis are in MATLAB specifications. Negative values show a displacement in the opposite direction of the axis.

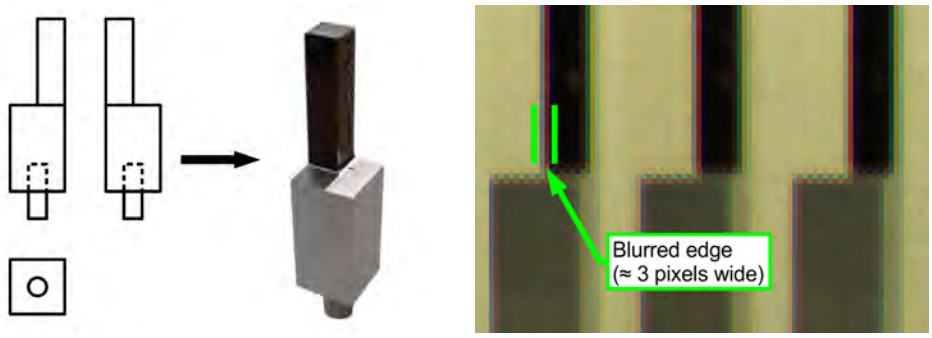
It is confirmed that the fisheye lens from Omnitech Robotics is a better quality lens compared to the Fujinon fisheye lens. In the case of the Fujinon lens, the displacement was actually refined after the vibration compensation, as it showed an offset of 1.2 pixels. This offset is now included in table 3.2, rounded to a precision of 0.5 pixels.

Lens Distortion Function Omnitech Lens

Omnitech Robotics gives the lens distortion function in the form of a diagram and a polynomial. As stated before, the polynomial coefficients aren't precisely given. I worked around this by measuring data points from the r-θ-diagram and fit a polynomial to these points. Afterwards the non-linearity-diagram can be used as a backup to measure the accuracy of the polynomial approximation. Details about this approach are given in section 6.1.1.

Lens Distortion Function Fujinon Lens

The Fujinon lens doesn't provide any specification for the lens distortion. This could be due to lack of interest from the manufacturer or the low quality of the lens, as the distortion function would differ for each lens. Further, in previous work [45], it was stated that other ways of retrieving lens functions must be addressed in case the manufacturer doesn't give any details. Therefore, methods of measuring lens functions and distortions were investigated. The method is inspired from the work of Clarke and Fryer [24] (for details see literature review section 2.3), where they used an angular movement of the light source to measure the lenses refraction. Clarke and Fryer used a collimator (light source) mounted on a goniometer (protractor) to send a light ray through the testing lens with a sensor placed behind the lens. Although this method is often used for high precision instruments the costs of adopting such a system to this



(a) Target pointer close-up (b) Target pointer from camera view

Figure 3.6: *Target pointer*

research was out of range.

I therefore decided to use a simplified approach: The light source is replaced by a target pointer (replacing the collimator) and the angles are fixed to increments of 5 ° (replacing an accurate goniometer). The measurement disk, as illustrated in figure 3.7, was manufactured in the university's workshop. The size of the disk was limited to a diameter of 400 mm due to the available CNC (Computer Numerical Control) manufacturing machine.

The disk holds the camera in a fixed position in the center of the disk. The target pointer illustrated in figure 3.6(a) is placed into the position holes around the edge of the disk. A picture of the target pointer is taken at each position hole to retrieve the lens functions in steps of 5 °. The idea is to overlay all these images in a way such that the target pointer becomes visible at each position as illustrated in figure 3.9. At the end the lens distortion can be measured from this image with the known positions of the target pointer.

One drawback of not using a collimator is that the light source isn't of a defined color, which causes problem, due to chromatic aberrations, where edges appear blurred due to different non-linear refraction of different wave lengths λ of light. The target pointer has a one-sided face (figure 3.6(a), black face) and therefore requires photography from both sides to average this error. Photographing the target pointer from two sides (left and right) and taking the mean distance of both images averages this color shift and reduces the error. Figure 3.6(b) illustrates the chromatic aberration from the target pointer of up to 3 pixels.

The target pointer was photographed under the same light conditions for all images to detect the edges under similar conditions. After images have been taken from either side of the target pointer, over all position holes from $-110\,°$ to $+110\,°$ and for each of the 4 lens positions, the images were stitched together. Afterwards the

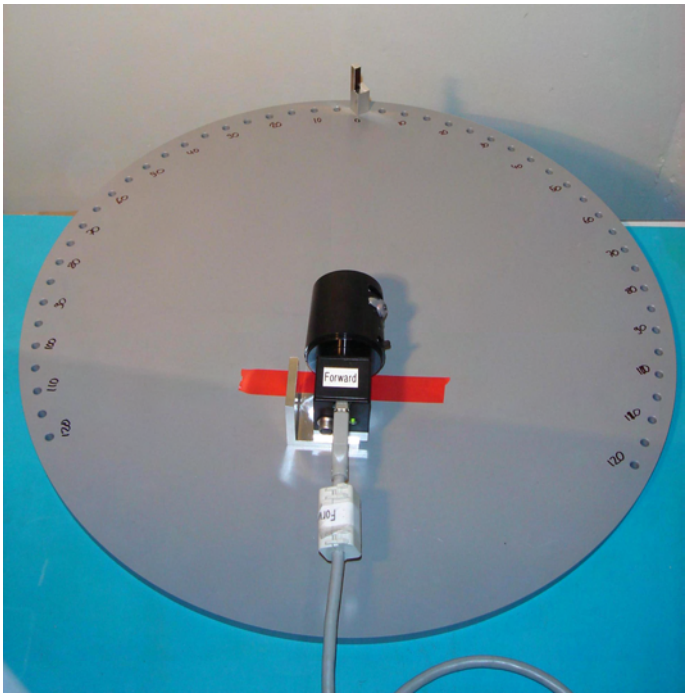
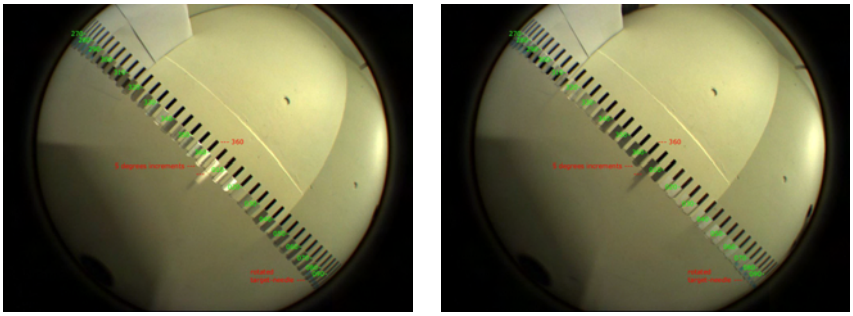


Figure 3.7: *Calibration disk setup with target pointer*

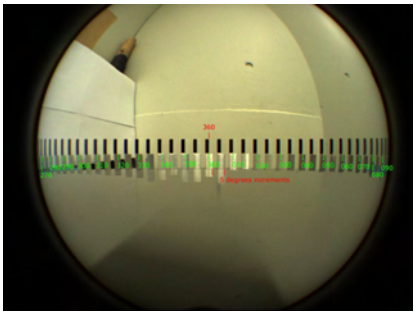
Figure 3.8: *Target pointer edge selection*

edges of the target pointers were automatically selected by a Canny [17] edge detector (MATLAB function). The edge detection uses constant parameters for the thresholds (low=0.1 and high=0.2), as the light in the laboratory is constant and not to influence the edge detection by an adaptive behavior. Figure 3.8 shows the stitched together target pointers and the applied Canny edge detector. The edges would now have been measured by a hand method at the respective positions in combination with the other side of the the target pointers edges. The results are presented in section 6.1.2, and the numerical tables and lens functions are listed in table A.1.

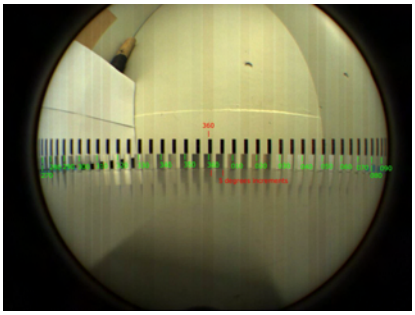


(a) Calibration image 1 left

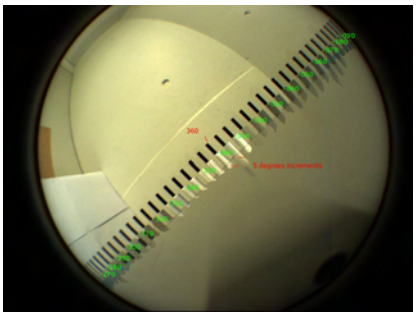
(b) Calibration image 1 right



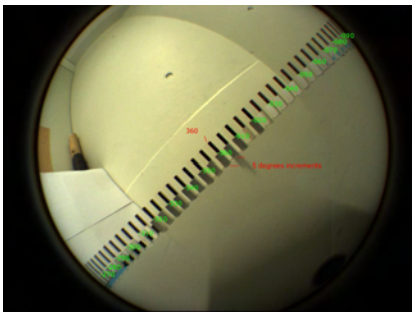
(c) Calibration image 2 left



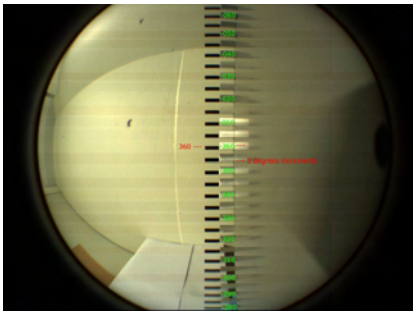
(d) Calibration image 2 right



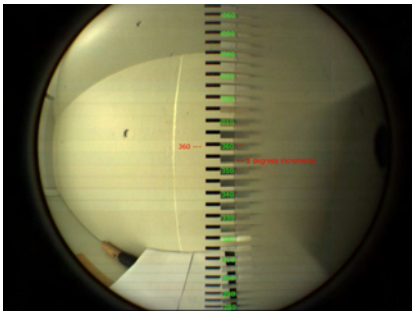
(e) Calibration image 3 left



(f) Calibration image 3 right



(g) Calibration image 4 left



(h) Calibration image 4 right

Figure 3.9: *Calibration images used to derive the lens functions LF1–LF8*

3.2 Image Rectification

It is now possible to remove the distortion from parts of the image with the spherical lens model in figure 2.2. The resulting lens functions will be presented in section 6.1 and describe the distortion as a function of $r(\theta)$. To rectify a fisheye image the inverse distortion must be considered, as described in the following sections. The rectification described uses a grid, which represents the inverse distortion, overlaid over the fisheye image. Extracting and aligning the grid in a rectangular manner removes the distortion.

The rectification method presented here is only one of many, as stated in the literature review in chapter 2. The method chosen must consider the use for photogrammetry where the rectified points can be related to position coordinates. Depending on the AOI (Area of Interest) the extracted area can vary not only in size but also in "position". Position means that an AOI can be selected anywhere over the image, to represent any view within the hemispherical FOV of the fisheye image.

3.2.1 Extraction and Rectification

The common method of image extraction is to extract a rectangular shape from the center of the fisheye image. Such an extracted and rectified image can then be used for mapping, where the image has the same behaviors as an image taken by a perspective camera. This extraction and rectification requires the lens distortion function, as described in section 3.1.1.

First, the lens distortion function is overlaid over the distorted fisheye image as shown in figure 3.10. The grid with the representation of the lens distortion (bent rectangle over fisheye image) is overlaid over the fisheye image. The bent rectangle contains the reverse fisheye lens distortion, as it is a rectangular extraction grid and distorted by the lenses distortion. The spherical lens model in figure 2.2 is applied to each grid point of the bent rectangle.

The lens model uses the underlaying sphere because the basic parameter of the fisheye lens distortion function is the incoming angle θ. recalling the literature review, the sphere itself isn't the lens model and is only used to determine the angle θ. An observer at the position of the camera would see a point P_w at this angle. A mathematical description of the rectification is derived from this model in equation (3.5), with $g_{i,j}$ being an image pixel of the rectified image:

$$g_{i,j} = \begin{pmatrix} g_i \\ g_j \end{pmatrix} = r(\theta_{i,j}) \begin{bmatrix} cos(\varphi_{i,j}) \\ sin(\varphi_{i,j}) \end{bmatrix} \tag{3.5}$$

where θ is the angle the observer would see the world point $P_w = (x_w; y_w; z_w)$, shown in equation (3.6). The indices i and j are the indices of a digital image of size $i \times j$ pixels and the angle φ is the azimuthal angle from the spherical lens model.

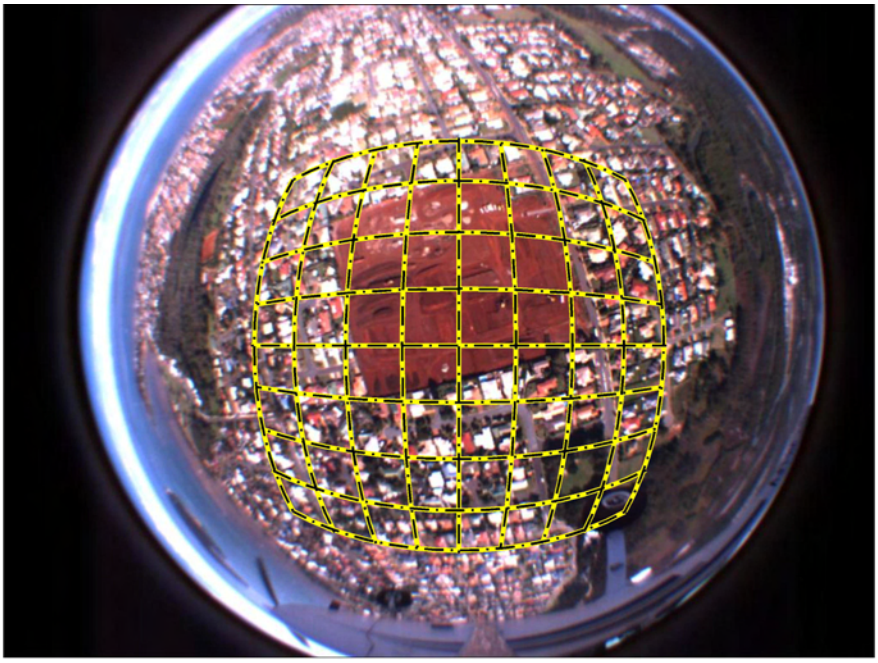
$$\theta = acos\left(\frac{z_w}{\sqrt{x_w^2 + y_w^2 + z_w^2}}\right) \tag{3.6}$$

The world points used to setup the grid matrix $\mathbf{G}$ in (3.7) are virtual points. These virtual points form a rectangular grid at an arbitrary altitude (e. g. 100m). It is arbitrary, because the scale of the rectified image isn't influenced at this step, but it can be by correctly using the altitude arguments. It was decided not to use the scale at this step, but rather to implement the scale to the projective transformation in section 3.3.3. Equation (3.7) shows the cumulative matrix G, containing all g's, representing the bent grid of the extraction area.

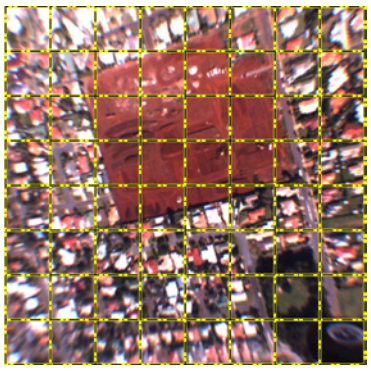
$$\mathbf{G} = \begin{bmatrix} g_{1,1} & g_{2,1} & \cdots & g_{i,1} \\ g_{1,2} & g_{2,2} & & \\ \vdots & & \ddots & \\ g_{1,j} & & & g_{i,j} \end{bmatrix} \tag{3.7}$$

This method limits the extraction area to a maximum θ of $<90\,°$, which would be a FOV of $<180\,°$. For each grid intersection the color value of the underlying fisheye image gets interpolated with a cubic interpolation to create a pixel in the new rectified image. The distorted grid is then aligned rectilinear, which results in a rectified image as shown in figure 3.10(b). For completeness, figure 3.10(c) shows the registered image, which is the transformed rectified image.

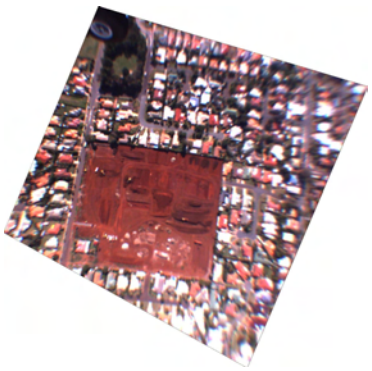
The algorithm using the above mathematical description of the image rectification is given in appendix A.3. The next section describes that the image extraction from fisheye lens images isn't limited to the center of the image or to angles $\theta < 90\,°$.



(a) Fisheye image with overlaid extraction grid



(b) Extracted AOI aligned in a rectilinear grid



(c) Rectified image after registration

Figure 3.10: *Illustration of the rectification process*

3.2.2 Further use of Image Extraction

The image extraction isn't limited to the center of the fisheye lens. Figure 3.11 shows an example of an extraction of the horizon. Top left is the extraction matrix (green), which is a rectangular grid of extraction points bent into the shape of a partial annulus. The zoomed in grid in the red rectangle shows the underlying grid. This partial annulus is used to extract (interpolate) the image pixels from the top left corner of the fisheye image. Aligning the grid in a rectangular manner results in a rectilinear image shown underneath.

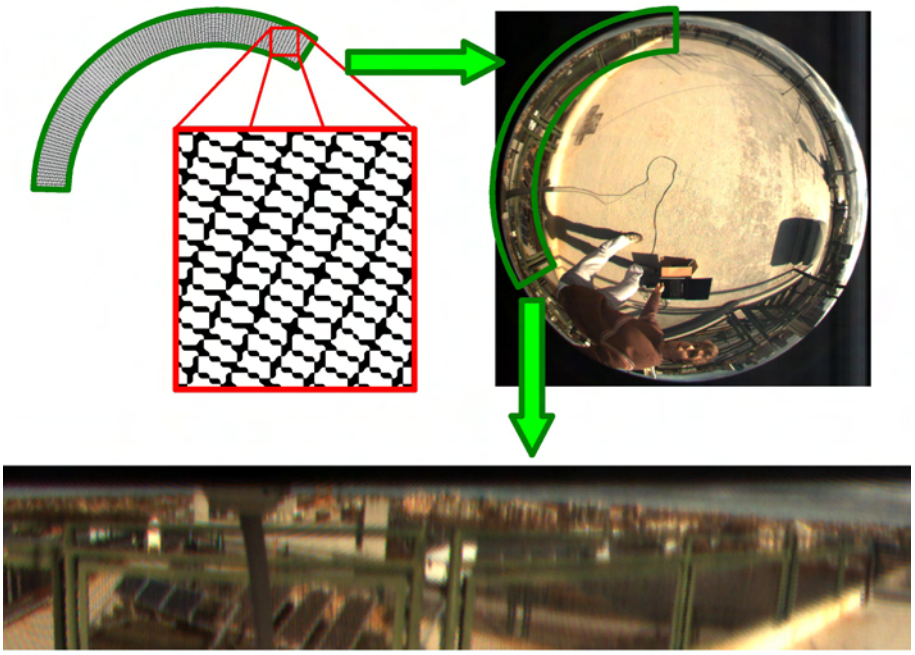


Figure 3.11: *Extraction of the fisheye image horizon*

This demonstration shows a benefit of the fisheye lens providing image mapping and other feature extraction from the same image and sensor. Such a horizon extraction could be used to track the attitude of the aircraft, detect targets in front and retrieve image information from the front or rear of an object (house) on the ground. The following image registration however will only use extracted and rectified images according to the previous section 3.2.

3.3 Image Registration

Registration of images in aerial photography adds positioning metadata to the rectified image. In other words, the registration orientates the image absolutely and adds the

additional information to the image.

Since new generation sensor systems are available, the entire registering process has changed for the end-user perspective. The traditional method used a 3-step approach to register images:

- Interior orientation: Film orientation, distortion correction, centering of film, proper principal distance estimation.
- Exterior/Relative orientation: Orientation of each image and images to each other.
- Absolute orientation: Scaling to world, orientation to datum, Aerial triangulation.

New generation systems, such as the Leica Geosystems ADS40, are using direct geo-referencing. Direct geo-referencing requires GPS (Global Positioning System) and INS (Inertial Navigation System) sensors, which are used to register data from the line scanner in real-time [98].

The fisheye lens system is one of the new generation systems. Whilst direct geo-referencing is meant for the end-user, the interior and exterior orientation is still required to be implemented. The fisheye imaging sensor uses GPS and INS sensors to provide geo-referenced images, although this isn't in real-time. The registering process for the fisheye lens is divided into the following steps:

1. Lens parameter definition (see section 3.1.1)
2. Image rectification (see section 3.2)
3. Attitude estimation using GPS/INS
4. Data synchronization (see section 4.3)
5. DEM (Digital Elevation Map) data preparation
6. Image transformation for absolute orientation

The latter 4 items are now described in more detail in the following sections.

3.3.1 Attitude Estimation

This section "Attitude Estimation" originated from a collaborative publication with Duncan Greer (QUT).

The inertial sensor and GPS data is processed in a loosely coupled GPS-INS navigation filter. The filter is an EKF (Extended Kalman Filter) meaning that the system is linearized about the current state estimate.

Sensors

The GPS sensor used was a dual frequency Novatel Propak-G4 configured to output the BestXYZ data. This data is the best estimate of aircraft position and velocity given in ECEF (Earth Centered Earth Fixed) coordinates. No differential corrections were used and hence the position and velocity solution is a stand-alone dual frequency solution. Typical position and velocity accuracy during the test was 6.0 m and 0.72 $^{m}/_{s}$ respectively (these estimates are made in real-time by the receiver).

The inertial sensor employed was a CrossBow MicroNav. This sensor comprises low-cost MEMS (Micro Electro Mechanical Sensor) gyroscopes and accelerometers from Analog Devices as well as a barometric altitude sensor and a three-axis magnetometer. The MicroNav also includes a GPS receiver and airspeed sensor. However these were not used in this experiment. All of these sensors are housed in a compact package. Since the magnetometer cannot be mounted remotely from other electronic equipment and the aircraft structure, its measurements are considered unreliable. Similarly, the pressure altimeter could not be connected to a calibrated static source and hence includes some airspeed dependent error due to the cockpit pressure.

A major issue with this class of sensor in this application is the bandwidth of the gyroscopes. The bandwidth of the gyroscopes in the MicroNav is 40 Hz (3 dB cutoff). This frequency corresponds directly with the vibrations induced by the cruise RPM (Revolutions Per Minute) of the engine and propeller (also at 2400 RPM or 40 Hz). Furthermore, it would be expected that there would be higher order vibrations (for example 80 Hz due to the engine power strokes or higher order vibrations due to component resonances). Clearly this is far above the frequencies which can be adequately measured by the gyroscopes, which results in aliasing of the gyroscope measurements. This is somewhat reduced by anti-vibration foam used between the aircraft structure and the MicroNav, but nevertheless some residual errors exist. The amount of noise is clearly seen in figure 3.12 which shows engine startup at times of 100 and 115 minutes into the test. Typical aircraft roll rates are in the order of 5 degrees per second, well below the noise floor during flight.

Process Model

The principal filter states are position error, velocity error and attitude (tilt) error. These states are augmented by six gyroscope and accelerometer bias terms. Thus the total state vector has 15 states. The state transition matrix is calculated from the standard INS error model equations, found for example in Titterton and Weston [96]. Since the data is post-processed in MATLAB, the discrete time transition matrix is found accurately from the state matrix using MATLAB's matrix exponential function

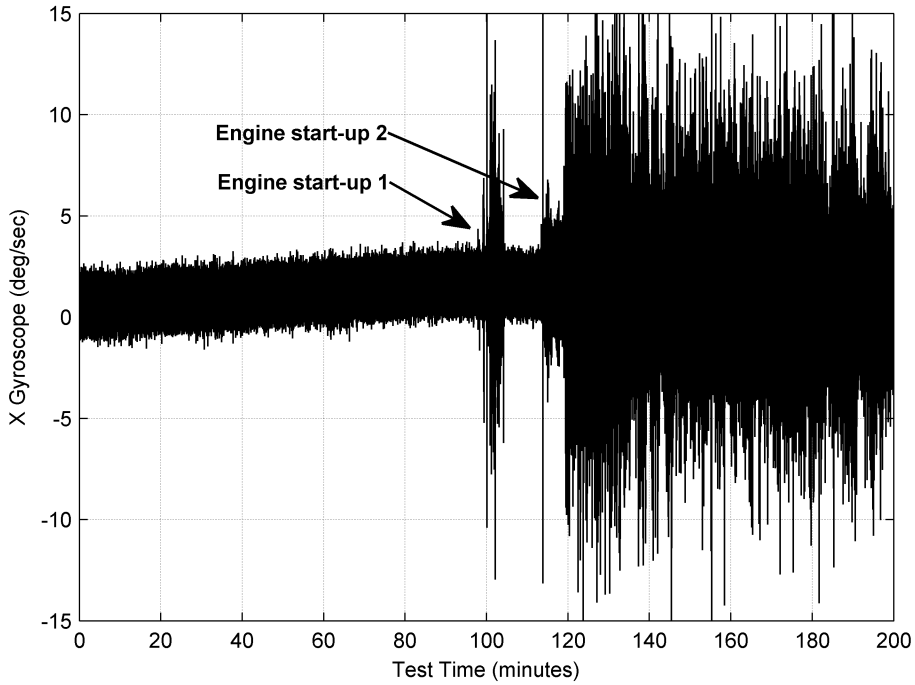


Figure 3.12: *Roll gyroscope measurement*

(expm) rather than using an approximation. The input connection matrix is used to connect the augmented states (gyroscope and accelerometer bias) to the inertial error states.

Process Noise

The continuous time process noise is found from the power spectral density and correlation time of the gyroscope and accelerometer noise. There are no cross correlation terms. The discrete time process noise is calculated as per Gelb [41].

Measurement Model

The filter measurements are the GPS position and velocity given in Latitude, Longitude, Height (LLH) and North East Down (NED) Velocities respectively as well as magnetic heading provided by the MicroNav's magnetometer measurements. The addition of the heading measurement improves the observability matrix rank from 12 to 13 (out of 15 for full rank), however has the potential to introduce heading error through magnetic anomalies or improper calibration. The magnetometer is ultimately required to provide attitude estimation stability due to the high noise and poor quality of the sensors. The measurement noise matrix R values are taken from the GPS receiver's estimates of position and velocity accuracy, as well as estimates of the magnetic

heading uncertainty.

Results

Although no truth data was available for the attitude accuracy, the following observations were made: The attitude accuracy achieved by the loosely coupled filter was approximately 3.2 ° (1-sigma) in roll and pitch and 5.8 ° (1-sigma) in yaw which is estimated through covariance analysis. This pitch and roll accuracy corresponds to a ground distance of 15.7 m at 300 m AGL. This is consistent with observations made from the alignment of registered images, which align on average to about $<$15 m (1-sigma), as shown in section 7.1.2.

3.3.2 DEM Data Preparation

The height map used in the registering process is the DEM available from USGS (United States Geological Survey). USGS provides SRTM-3 (SRTM (Shuttle Radar Topography Mission)) data for free from its SRTM. The SRTM-3 data has an underlying grid of 3 arc-seconds, which correspond to about a 90 m grid. SRTM-3 data has accuracy limitations due to occurrences of missing data points (so called "`NO_DATA_VALUE`"). Other sources of DEM data are listed below, but none of them was used. It was planned to replace the current data with data available from HydroSHEDS.

HydroSHEDS: Improved SRTM data by USGS. Information on their webpage `http://hydrosheds.cr.usgs.gov/hydro.php` stated that DEM data of Australia will be available in December 2007, which is now post-poned to February 2008. It was planned to replace the current data with HydroSHEDS data.

ViewFinderPanoramas: Only a few selected areas available, mainly mountain areas. A combination of this with SRTM-3 could improve the overall coverage, especially in mountain areas.

CGIAR-CSI: Data has the same problems as USGS with no data points. The release of version 3 of the data was planned for 2006, but still only version 2 is available. Further, download is restricted to GeoTIFF- and ArcInfo-format.

The SRTM-3 data was used in respect to the expectation of replacing the SRTM-3 data with the HydroSHEDS data. Further, the few missing data points in the Brisbane area could be interpolated using a MATLAB built-in Qhull-interpolation method (see MATLAB-help: griddata). The interpolation is required to avoid the possibility of

catching a "NO_DATA_VALUE", which is set to −9999 m AMSL (Above Mean Sea Level (Aviation)) by USGS.

The problem with the interpolation method is the limited size of matrix that can be worked with. Therefore, DEM data area is split into smaller matrices of 400–2500 data points. Each matrix is then separately processed and concatenated to the interpolated DEM matrix. The boundary points between two matrices are only interpolated by either matrix. The problem of the interpolation behavior between concatenated matrices isn't investigated because it is a minor issue and the plan is to replace the data with the HydroSHEDS data set, once available.

The pre-processed elevation data (DEM data) is now used to transform the rectified images to the ground, with the respective projection from the rotation matrices. The following section describes this process in more detail.

3.3.3 Image Transformation

At this stage the rectified image with its virtual image parameters is available. As stated in section 3.1, this image can now be treated as an image from a conventional lens. As illustrated in figure 3.13, image coordinates are taken in the 4 corners (A–D) of the image and the center of the image (M). First, the camera must be orientated correctly within the aircrafts assembly, then the coordinates must be setup and the image scale approximated to transformation the image. As shown in figure 3.13, the aerial image (points a–d) is ortho-rectified (registered) at the image plane (points a'–d'), which corrects the image for ground elevations. As explained further down in this section, the ortho-rectification only uses the 4 corner points, ignoring any elevation between them. This ground approximation was considered reasonable for aerial photography over non-mountainous areas, according with the general Australian terrain. An ortho-rectified image is assigned a scale, with which the now registered image can be scaled to a map, for instance as a Google Earth ground overlay.

Camera orientation, more exactly the CCD sensor orientation, depends on the installation to the aircraft. As shown in figure 3.14, the camera was placed under the wing of a Cessna 172 (see chapter 4 for further information about the camera installation and data acquisition system). In terms of coordinates this means that the camera's x-axis is opposite the flight vector, whereas the y-axis is directed away from the aircraft body (illustrated in step **1** and **2** in figure 3.15). In the image processing software (MATLAB Image Processing Toolbox) image coordinates are registered differently. Step **3** shows the MATLAB coordinates, where the y-axis is turned upside down. Equation (3.8) describes this coordinate flipping and shifting from steps 1 to 3. $ImSize$ is the size of the image in pixels (either x- or y-size), vector $(x_{cam}; y_{cam})$ the

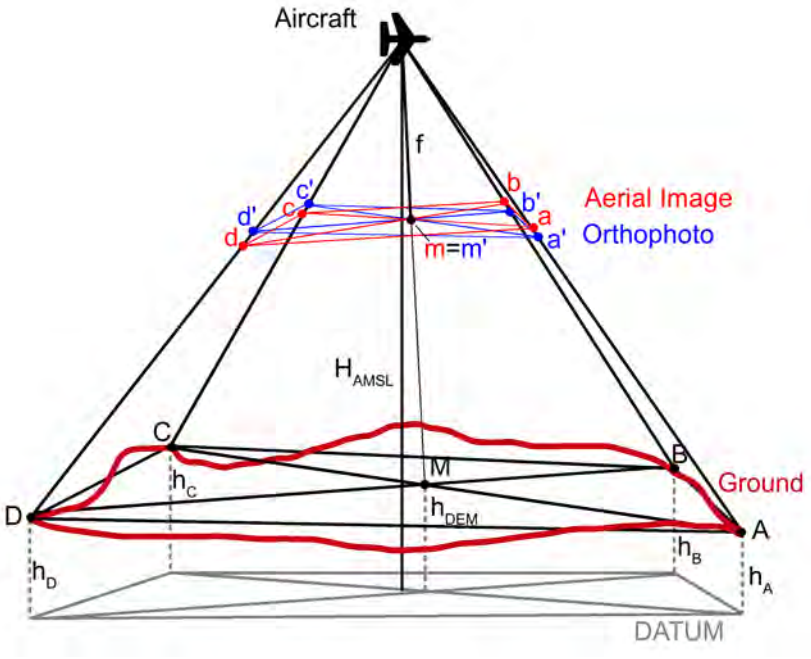


Figure 3.13: *Aerial image registration*

camera coordinates and $(x_{matlab}; y_{matlab})$ the image coordinates.

$$\begin{pmatrix} x_{matlab} \\ y_{matlab} \end{pmatrix} = \begin{pmatrix} x_{cam} \\ y_{cam} \end{pmatrix} \begin{pmatrix} -1 \\ -1 \end{pmatrix} + \frac{1}{2} \begin{pmatrix} ImSize \\ ImSize \end{pmatrix} \tag{3.8}$$

The ground coordinates can be calculated without considering the coordinate order. However, to register the correct image coordinate to the corresponding ground coordinate the image must be placed correctly in the world, otherwise the image would be twisted, flipped or rotated in the wrong manner. Equation (3.9) explains the setup of the transformation edge points $(x_T; y_T)$, which are, due to a peculiarity in MATLAB, the flipped coordinates $(x'_T; y'_T)$. The transformation edge points flip the coordinates from equation (3.8) then scale and shift them to a ± 1 range, as shown in figure 3.15, step 4. Lastly, the coordinates are rotated by $R_{90\circ} = 90\,^\circ$ counterclockwise, due to the axis alignment between the inertial sensor and the aircraft.

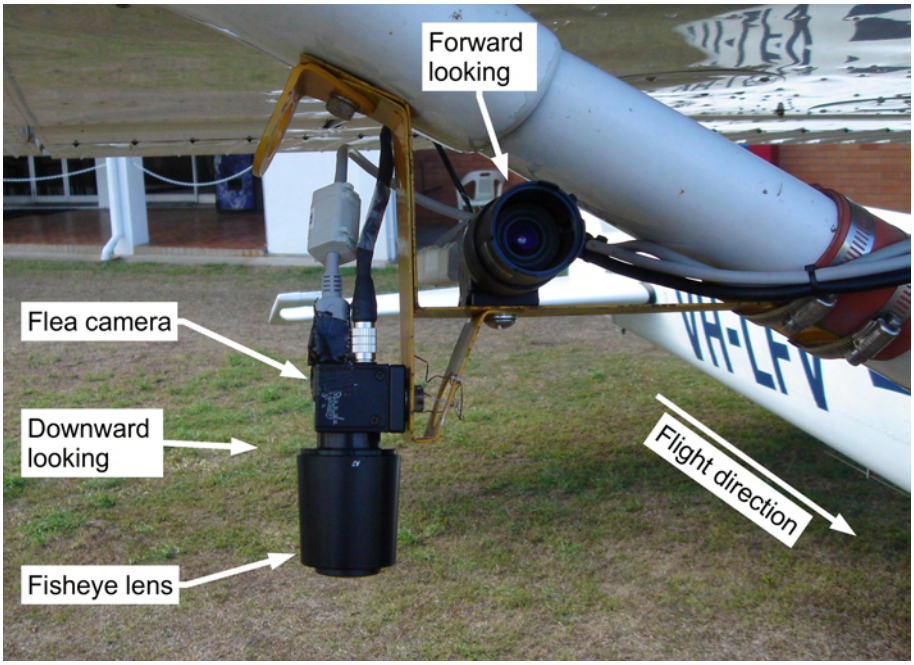


Figure 3.14: *Camera installation on the Cessna 172*

$$\begin{pmatrix} x_T \\ y_T \end{pmatrix} = \begin{pmatrix} -1 \\ 1 \end{pmatrix} \begin{pmatrix} x'_T \\ y'_T \end{pmatrix} \tag{3.9}$$

$$\begin{pmatrix} x_T \\ y_T \end{pmatrix} = \mathbf{R}_{90^\circ} \left[\begin{pmatrix} x_{matlab} \\ y_{matlab} \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \begin{pmatrix} \frac{2}{ImSize} \\ \frac{2}{ImSize} \end{pmatrix} + \begin{pmatrix} -1 \\ -1 \end{pmatrix} \right] \tag{3.10}$$

Step 5 transforms the local camera coordinates to world coordinates. The transformed base points $(x_B; y_B)$ are the coordinates scaled down to the ground. $(x_B; y_B)$ takes the result from equation (3.10), scales and rotates these coordinates according to the synchronized GPS and inertial sensor values (see section 4.3), as illustrated in equation (3.11). PS is again the pixel size, and in the case of the Flea camera $PS_{Flea} = 4.65\,\mu$m. $\mathbf{R}$ is the rotation matrix for x-,y- and z-axis, fed with the rotation angles ω, ϕ and κ from the inertial sensor in radians.

$$\begin{pmatrix} x_B \\ y_B \end{pmatrix} = \mathbf{R} \left[\begin{pmatrix} x_T \\ y_T \end{pmatrix} \begin{pmatrix} ImSize \\ ImSize \end{pmatrix} \cdot PS \cdot \frac{1}{scale} \right] \tag{3.11}$$

The *scale* is evaluated as in equation (3.12), where H and f are the camera height and focal length (see figure 3.2). Since the scale of the image depends on the ground

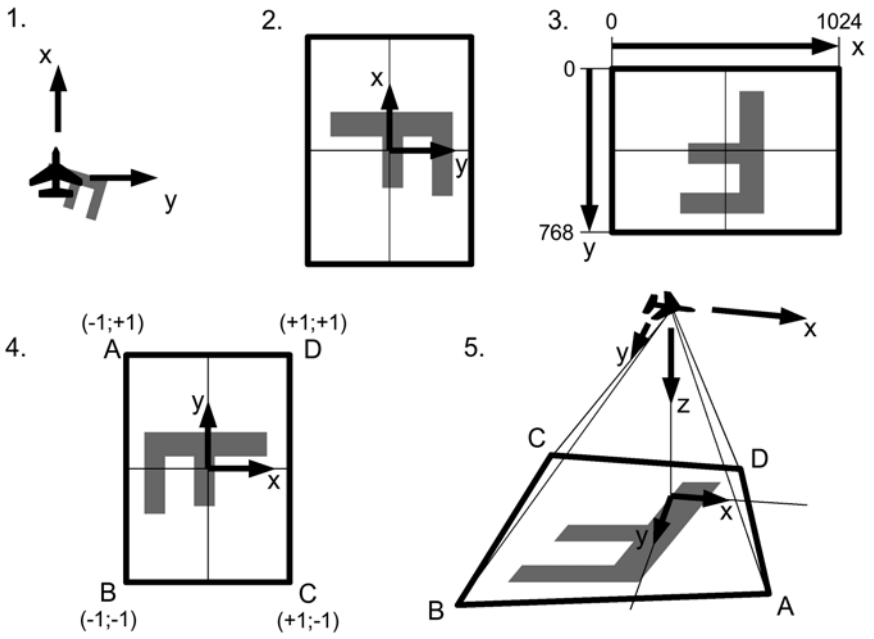


Figure 3.15: *From camera mounting to map coordinates*

elevation H is replaced by H_{AMSL} (the aicraft's altitude) and h_{DEM} (the ground elevation at the aircrafts nadir point).

$$\begin{aligned} scale &= 1 : \left(\frac{H}{f}\right) \\ &= 1 : \left(\frac{H_{AMSL} - h_{DEM}}{f_v}\right) \end{aligned} \tag{3.12}$$

The scale of the corner points A–D is calculated from the relief displacement using an iterative approach. Figure 3.16 illustrated this iteration problem, where the correct position of a point A can't be calculated directly since the position of A depends on the elevation h_A, which itself depends on the position of A. Therefore, A' gets approximated by equation (3.13) using the *scale* of the nadir point M. After this first iteration the *scale* is replaced with the elevation at A' and equation (3.13) is repeated until the changes in A' become negligible.

$$\underline{X_{A,approx}} = scale \cdot \mathbf{R} \cdot \underline{\mathbf{x_a}} \tag{3.13}$$

These coordinates are then transformed into the respective position of the aircraft at the local coordinate and scaled to the earth's surface. Therefore the perspective camera projection must be used with respect to the relief displacement. The aircraft's attitude is integrated by applying the rotation matrix, as in equation (3.14):

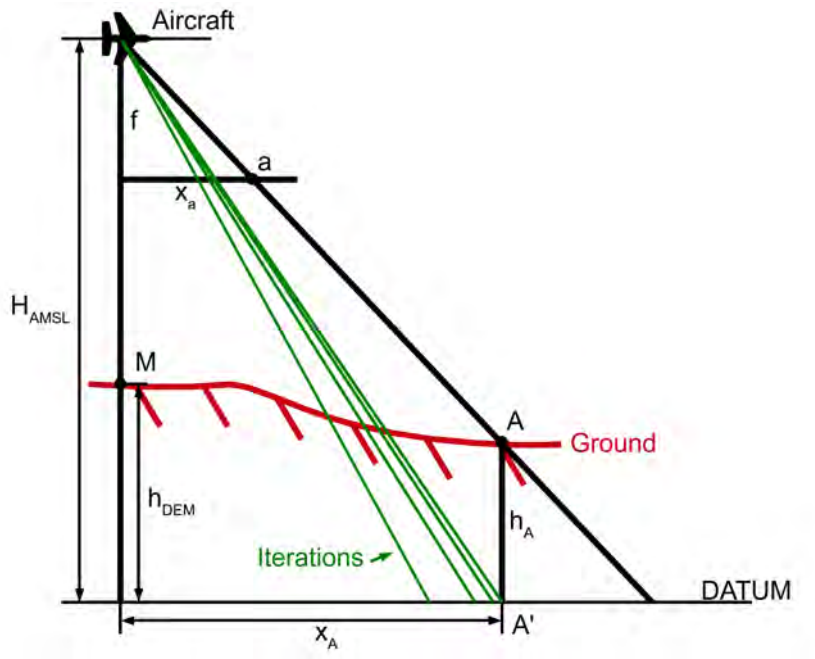


Figure 3.16: *Using relief displacement to approximate point A*

$$\mathbf{R} = \mathbf{R_z} \bullet \mathbf{R_y} \bullet \mathbf{R_x} \tag{3.14}$$

where the complete rotation matrix $\mathbf{R}$ is as follows:

$$\mathbf{R} = \begin{bmatrix} cos(\kappa)cos(\phi) & -sin(\kappa)cos(\omega) + cos(\kappa)sin(\phi)sin(\omega) & sin(\kappa)sin(\omega) + cos(\kappa)sin(\phi)cos(\omega) \\ sin(\kappa)cos(\phi) & cos(\kappa)cos(\omega) + sin(\kappa)sin(\phi)sin(\omega) & -cos(\kappa)sin(\omega) + sin(\kappa)sin(\phi)cos(\omega) \\ -sin(\phi) & cos(\phi)sin(\omega) & cos(\phi)cos(\omega) \end{bmatrix} \tag{3.15}$$

As the rotation matrices can cause confusion, the order of rotations of the rotation matrix is important because the matrix multiplications aren't commutative. In the case of this research (and aviation generally) the rotation matrices are usually ordered as in equation (3.14), as used in the CrossBow MicroNav and described by Nelson [81]. However, photogrammetry publications sometimes reverse the rotation matrices, for example [9, 40, 102]. There is no correct description of this as the rotation matrices are system dependent and depend on the underlaying hard- or software.

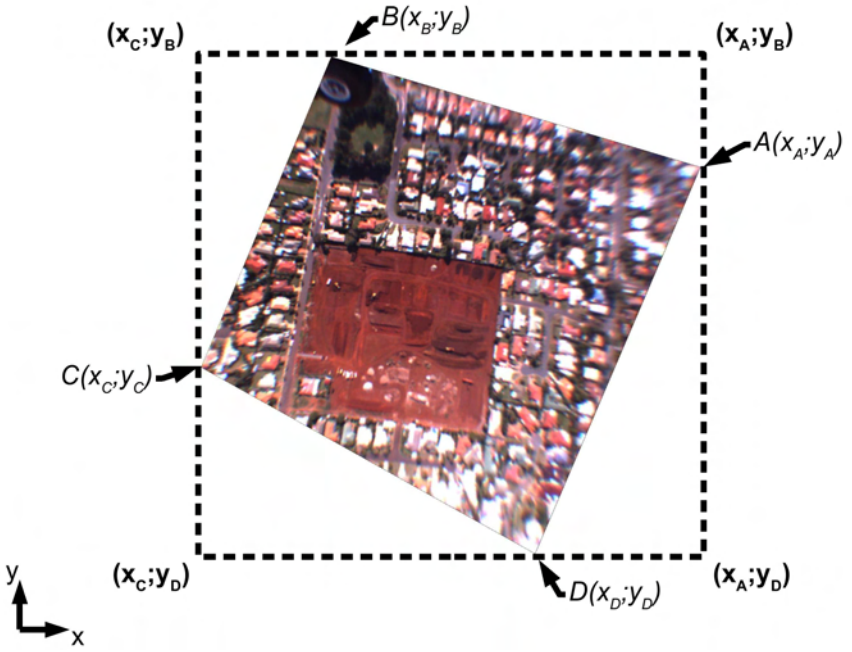


Figure 3.17: *Bounding box around registered image*

3.4 Mapping

The registered (and ortho-rectified) images are assigned a scale and an image size. With these numbers available longitude and latitude coordinates can be derived using a NED to LLH coordinates transformation [11]. The coordinates of the edge points are saved in a KML-file (KML (Keyhole Markup Language)) for Google Earth, according to the KML 2.1 Reference [43].

For each image a separate KML-file is created, which can be loaded into Google Earth. Google Earth places these images according to their longitude and latitude coordinate, stacking each image on top of previously loaded images. The longitude and latitude coordinates are the bounding box coordinates of the rectangular image surrounding a registered image, as illustrated in figure 3.17. The bounding box coordinates are derived from the registered image corner coordinates. The white spaces within the bounding box not filled by the registered image are set to be transparent. This was implemented by saving the images as PNG-files (PNG (Portable Network Graphic)), allowing transparency to be set in the alpha-channel. Stacked images in Google Earth therefore appear as the registered images only, without the white spaces from the bounding box (see figure 4.7).

3.5 Summary

This approach established the basis to understand fisheye lenses, rectify and registering fisheye images. The fisheye lens distortion can be approximated using the spherical lens model according Kannala and Brandt [60]. If the lens function isn't available it can be derived using the calibration disk presented in figure 3.7. However, it was concluded that a fisheye lens distortion varies over the lens (see section 6.1.2 for more details). Therefore, the derivation of the lens distortion is planned to be taken over multiple azimuthal angles φ.

The image rectification uses the lens function $r(\theta)$ or $r(\theta, \varphi)$ to overlay a grid over the fisheye lens, containing the lenses distortion. By aligning this grid in a rectilinear manner parts of the fisheye image get extracted (interpolated) and rectified. Section 6.1.1 will show that the rectification performance is approximately 1.1 %.

These rectified images are based on a virtual parameters defining its FOV, focal length and size, as it would be an image taken by a conventional lens. The registering process transforms these images according the inner and outer orientation to world coordinates. The registered images are used in Google Earth to demonstrate the mosaicing performance of image sequences presented in section 6.3. These results are presented in chapter 6.

The airborne data acquisition in the following chapter presents the process of acquiring aerial photos with the fisheye lens. Presented is the installation of the fisheye lens camera to a Cessna 172 and the data acquisition system consisting of a GPS and inertial sensor. Timing issues of synchronizing the image and sensor data are analyzed in detail as they are critical to the registering process. An unforeseen vibration of a loose lens part lead to a vibration compensation using the Hough transform applied to the fisheye images presented in chapter 5.

Chapter 4

Image Acquisition

After setting up the theoretical background of rectifying and registering fisheye images, an airborne image acquisition was conducted to collect real data. The airborne image acquisition was conducted with a manned aircraft, equipped with a downward looking fisheye lens camera, in June 2007. Data was collected with the data acquisition system. This is a selfmade system developed by students within the research group (ARCAA) to acquire imagery data, GPS data and inertial sensor data at high update rates, synchronized to each other. After the flight large amounts of the data collected was processed and the results are shown in chapter 6. However, the images also showed some major problems in the acquired data, such as vibration detected in the images after the flight. This problem is discussed in detail in chapter 5.

4.1 Airborne Data Acquisition

The airborne data acquisition was prepared by mounting the entire data acquisition system into a car for trials due to the complexity of the system. Since the system is selfmade at a very low cost, it lacks user interfaces and requires trained people to operate it. Once ready, it was decided to go for an airborne data acquisition "mission", which included installing the system into a Cessna 172, as shown in figures 4.3 and 4.5.

Figure 4.1 is a technical drawing of the data acquisition system. It consists of:

- GPS antenna and receiver (Novatel Propak)
- Inertial sensor (CrossBow MicroNav)
- Downward looking camera (fisheye lens camera) and a forward looking camera
- Two laptop computers to control the 2 cameras
- External storage (hard disks) to store the image and sensor data from the cameras, inertial sensor and GPS

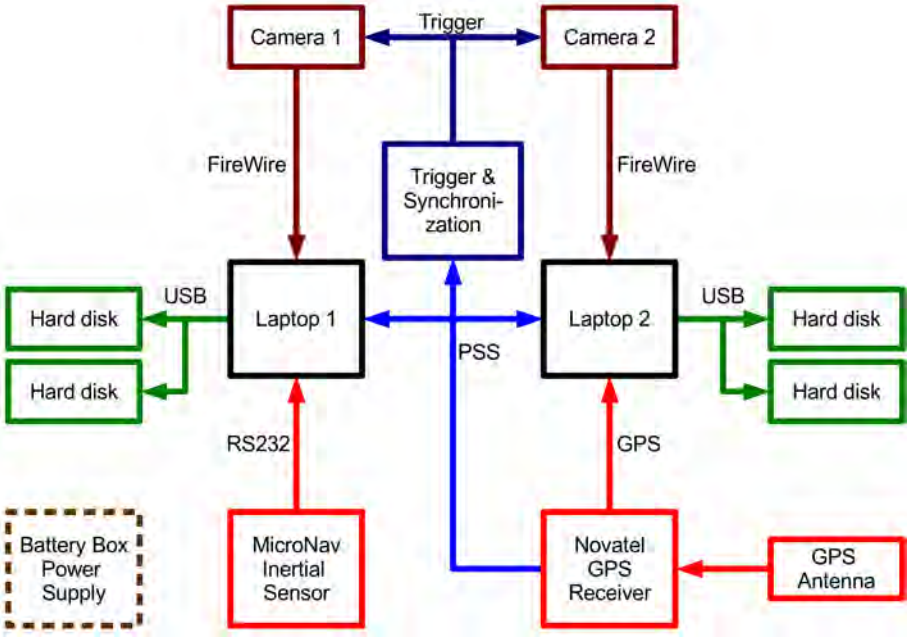


Figure 4.1: *Illustration of the data acquisition system*

- Battery box power supply

The entire data acquisition system weighs approximately 45 kg on its own (including the battery box to supply the system for >6 h) which is close to the Cessna 172 weight and balance limits.

A Cessna 172 was chosen instead of a UAV because of several reasons. Primarily, there was no UAV available at the time of the image acquisition. Further, a UAV image acquisition would have been complex to organize, due to restrictions such as CASR Part 101 [3] in Australia. A UAV would also have restrictions in payload and space which would have made the available data acquisition system unusable. Last, the data acquisition system has been flown in a similar setup previously in 2005. Therefore, a manned aircraft was chosen due again to the availability and comfort. The flight took place in the southern Brisbane area between Archerfield Airport and the Gold Coast. The flight path (white line) drawn on a Google Earth map is shown in figure 4.2. The percentages next to the flight path represent the recorded time into the flight test, with the flight starting at approximately 54 % and ending at 100 %. It was planned to fly over the areas selected to provide different kinds of image data (marked with the green letters). The selected places of interest included:

A: A lake (Tingalpa Reservoir)

B: Populated coastal area (Redland Bay)

C: An airport (Southport)

D: High-rise buildings (Surfers Paradise)

E: Low density housing area (Coomera)

F: Farmland (Beenleigh)

G: High density housing area (Logan)

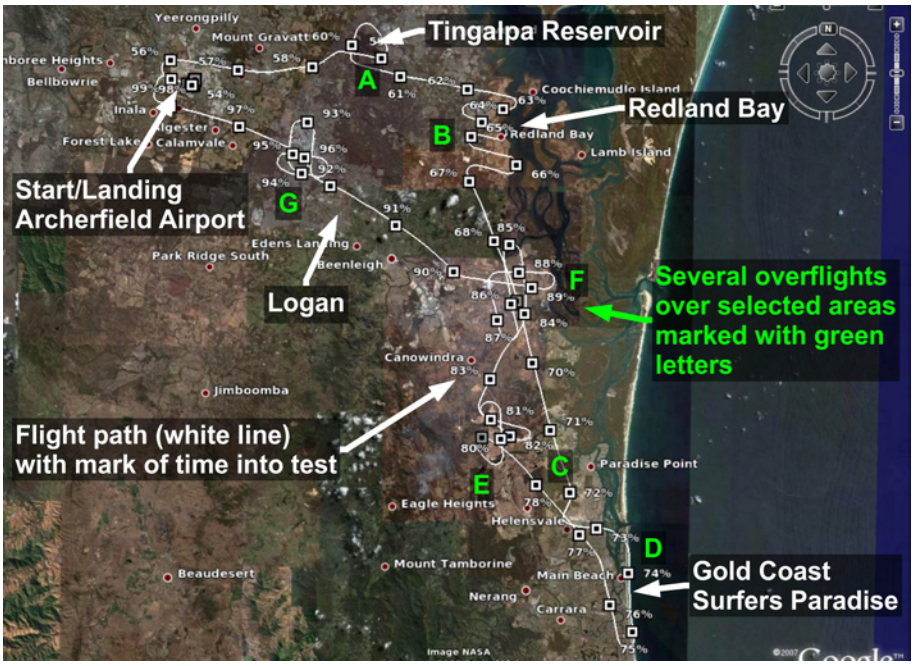


Figure 4.2: *Flight path in the southern area of Brisbane*

The system was setup to collect data at 15 fps from the downward looking camera. Images were collected as raw data (Bayer layer). There were ≈83000 images collected during the flight or ≈100 GB worth of data. Image, GPS and inertial sensor data are synchronized to each other using timestamps which are processed after the flight. The following sections give an overview of this time synchronization, with a detailed analysis that shows no significant problems with the data synchronization.

4.2 Sensor Installation

The fisheye lens was installed as a downward looking camera underneath the wing of a Cessna 172, as shown in figure 4.3. The camera was installed to the bracket, so that the camera x-axis was opposite to the flight vector and the y-axis was directed away

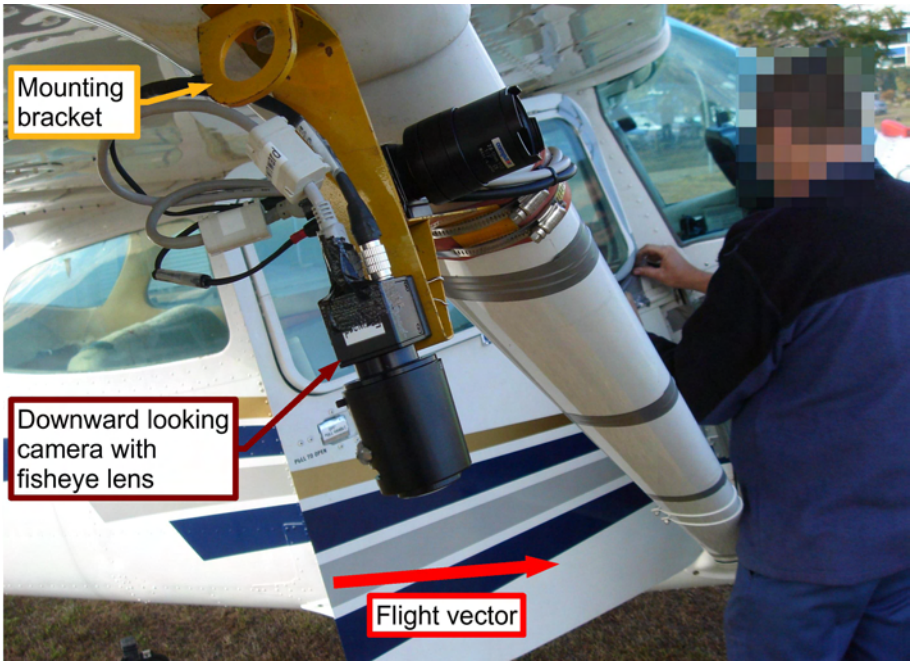


Figure 4.3: *Camera installation underneath the wing of a Cessna 172*

from the fuselage. The forward looking camera was installed as well, but wasn't used for this research. The cameras were mounted onto a certified bracket which required an Engineering Order stating that an aircraft engineer is required to mount the bracket onto the aircraft.

Unfortunately, the camera and the inertial sensor are required to be mounted in two different places, as illustrated in figure 4.4. The inertial sensor was placed into the baggage storage which is behind the the rear seats and is accessible through a door on the side of the aircraft, as shown in figure 4.5.

The distance between the sensors was measured as illustrated in figure 4.4. This offset between the sensor was also included in the image registration in section 3.3.3. However, this separate installation isn't ideal, as it can be assumed to have vibration of the wing relative to the sensor, which adds additional errors to the image registration. Further, the measurement of the distances is difficult and an accurate angle estimation is nearly impossible. The problem is that there are no good reference points available, except for two level means (according to Pilot's Operating Handbook Cessna 172 [20], p. 6-4). One level mean is placed at the upper door sill, which the camera was compared to. The aircraft's engineer installing the cameras estimated the camera tilt to 3 ° backwards, which he assumed to be horizontal in flight according to the payload. Here, a significant improvement must be made to estimate this important piece of the inner orientation of the camera more accurately. Although this may be difficult without

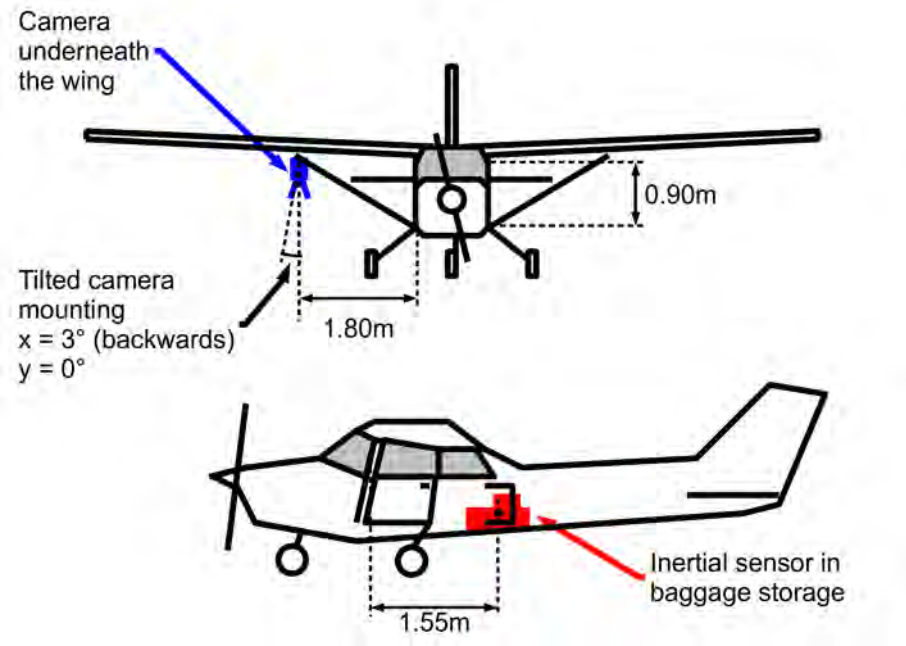


Figure 4.4: *Cessna 172 plan*

owning the aircraft and placing reference points to bracket installation. At this stage it is planned to acquire an aircraft for research purposes within ARCAA (Australian Research Center for Aerospace Automation) where camera brackets could then be permanently mounted to the aircraft.

The GPS antenna was placed on the right-side of the dashboard, away from the pilot, where it doesn't influence the pilot's view. The GPS antenna cable was laid along the door and floor of the aircraft to the receiver in the baggage storage.

4.3 Data Synchronization

The general approach for managing the time synchronization for this experiment was to use the GPS time as the common timing source. The GPS receiver provides a PPS (GPS Pulse Per Second) signal which was used for this purpose. All sensor and image data was timestamped with GPS time and the timing differences were analyzed with great care. It turned out that a part of the synchronization timestamping failed between the synchronization of the GPS time and the computer clock, however this could be solved as explained in section 4.3.1.

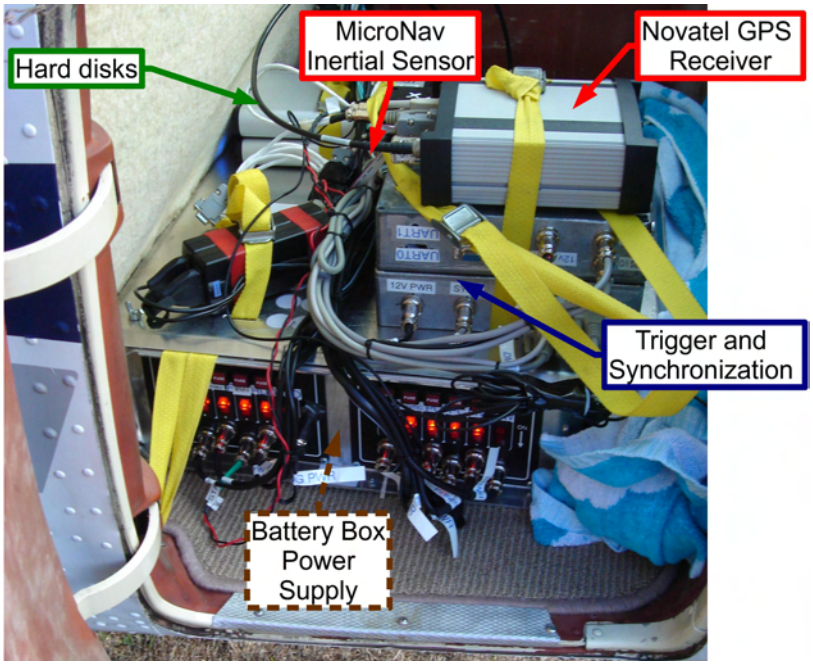


Figure 4.5: *Installation of the data acquisition system in the Cessna 172*

4.3.1 Laptop Interrupt Delay

First, an analysis of the time drift of the computer interrupt delay was conducted. The reference signal for this was the PPS signal from a Novatel GPS receiver, measured at the parallel port of the computer. It was expected to see interrupts within the time frame of the image triggers so that no images were dropped by the system. The expected interrupt delay time is <25 ms (see timing diagram in figure 4.9) if the images are taken at 15 fps. Each time the PC received the PPS signal an interrupt was generated that records the PC clock. The results of this test are shown in figure 4.6.

The x-axis is the time into the experiment in seconds, with the first 6000 s representing setup time and operations on ground. The y-axis shows the epoch by epoch difference between the PC clock second (interrupt) and the GPS receiver PPS. It can be seen that during ground operations, when the data acquisition system was not receiving image data, the relative delay was consistent. Under load, when the processor is busy servicing other interrupt routines, the time synchronization interrupt subroutine is delayed, resulting in greater uncertainty in the timing. The interrupt delay increases, as expected, due to limited CPU (Central Processing Unit) time in the non-real-time Linux operating system. However, for >99 % of the measurements the time delays are within 25 ms, which can be considered as "on time" following the

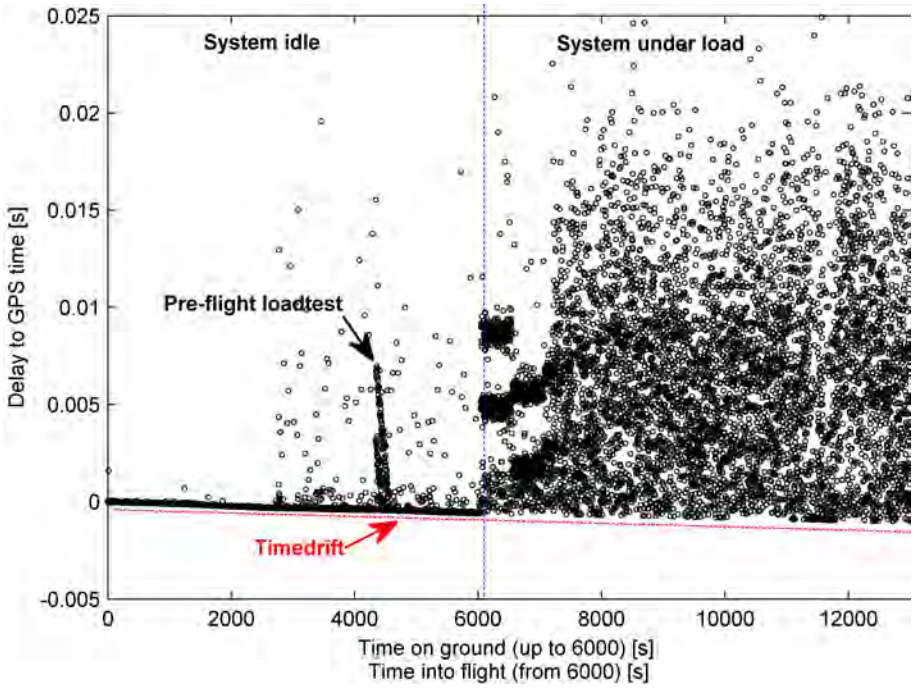


Figure 4.6: *Computer interrupt delay*

description in section 4.3.2. This interrupt delay is also shown in figure 4.9, which illustrates the entire camera timing, from the camera trigger pulse to the timestamp of the image. If all interrupts had been processed with zero delay all the points would lie on the dashed red line.

This dashed red line represents the native PC clock drift relative to GPS time. In this case the computer clock is running faster than the GPS time ($\approx 0.6\,^{\mathrm{ms}}/_{\mathrm{h}}$), therefore the computer clock seconds are shorter than GPS time seconds, which results in the negative slope of the curve. With this slope of the dashed red line we can correct the computer clock for the time drift even though this drift isn't significant.

However, a part of the synchronization hardware failed leading to a required time not being recorded. The problem related to which PPS (GPS second) corresponded with which computer second, with this information being unknown, and the hardware would record this in the normal case. The missing number is an integer offset between the GPS and computer time. The work around to this problem was to initially guess the offset to register images in Google Earth (see section 6.3) and measure their relative and absolute errors. This initial guess was made using the correlation between the inertial sensor and GPS data at engine start-up or in a taxi turn. This allowed a "guesstimate" of the time within a time frame of approximately $\pm 5\,\mathrm{s}$ and allows an estimate of the attitude and position of the aircraft in straight and level flight, as the

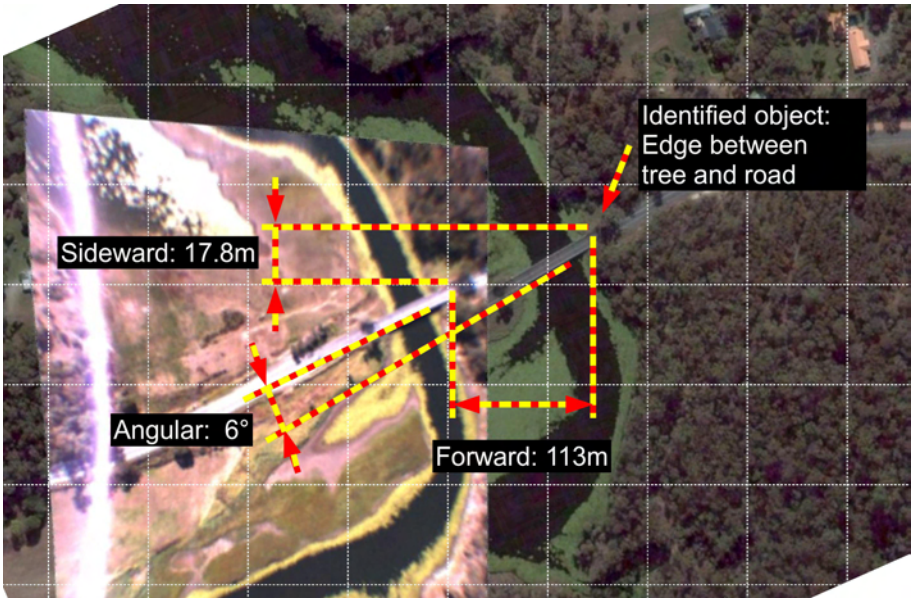


Figure 4.7: *Measuring errors on ground features*

GPS only updates the inertial sensor from drifting away. In straight and level flight this offset will register the images either in the past or the future leading to a larger number of absolute along-track errors than across-track or relative errors.
A set of 111 images was analyzed in detail measuring absolute and relative errors along-track and across-track, as illustrated in section 6.3.1 and shown in figure 4.7. Ground features had to be carefully chosen not to introduce further errors. Measured image errors (relative along-track/across-track and absolute along-track/across-track) are listed in the appendix in section A.2.1. The statistical analysis in figure 4.8 and table 4.1 show that the relative errors and the absolute across-track error have a mean at approximately 0 m, except for the absolute along-track error. The mean of the absolute along-track error is 108 m but as expected the approximated error of 11.8 m (1-sigma) is similar to the other diagrams. This lead to the conclusion that the absolute along-track error could be shifted by the mean towards zero, as the along-track error should show a similar distribution around zero, as in the other diagrams. Further, the aircraft speed is approximately 110 kt (=56.6 m/s) in straight and level flight. The offset is an integer value of the second and therefore either a mean-offset of 56.6 m, 113.2 m or 169.8 m. Therefore the offset was estimated to be 2 s in the past as the images of the future are registered in the past. By adjusting the computer clock by this offset, images are registered at the correct positions.

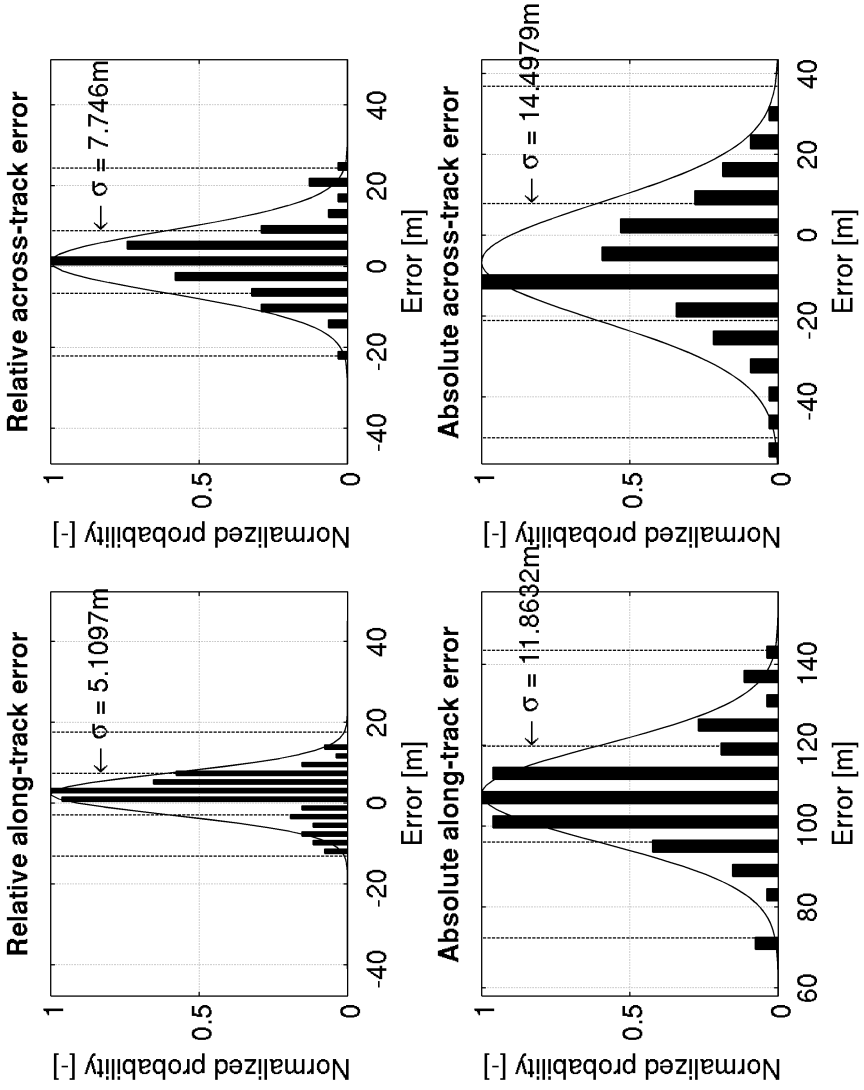


Figure 4.8: *Normal distributions of relative and absolute errors*

Table 4.1: *Mean and standard deviation of measured errors*

Error	mean [m]	1-sigma [m]
Relative along-track	2.2275	5.1097
Relative across-track	1.1198	7.7460
Absolute along-track	107.8874	11.8632
Absolute across-track	-6.6410	14.4979

4.3.2 Synchronizing Image Data

Another important synchronization problem is that associated with the images. Again, the PPS signal from the GPS was used as a trigger to the camera to collect the image. The image was then saved onto the PC using the Coriander IEEE camera software. As illustrated in figure 4.9, there are several delays that play into the time stamping of the images.

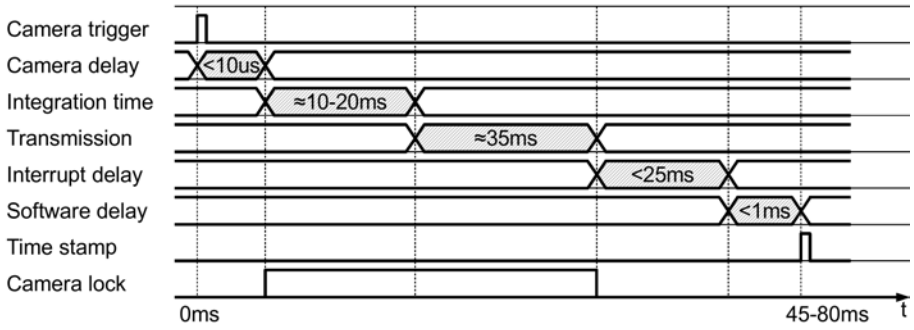


Figure 4.9: *Camera and image transmission timing*

The first delay is the camera delay, which is the time taken for the camera to receive the trigger signal and to start integrating the image. This time is less than 10 us, and according to Point Grey Research article 255: "Measuring the time from trigger input to start of shutter (trigger latency)". The integration time is set by the camera automatically and depends on the illumination. Based on our measurements it is typically 10–20 ms on a cloudy day. The transmission of the image data over the FireWire 400 bus requires approximately 35 ms.

Then there are the delays within the computer. The chosen operating system is Linux, which isn't a real-time operating system. The latency to process an interrupt can be assumed to be $<$25 ms based on previous experimental results. The delay of the Coriander software to collect and save the image is assumed to be small ($<$1 ms).

This results in a total latency of approximately 80 ms from the time the PPS signal triggers the camera to the time the image is saved onto the PC. Notice also, that the camera is blocked while it is processing an image. Blocked means that it won't process any other trigger signal until the transmission of the last image is complete.

The image timestamp from the Coriander software is therefore delayed by the delays illustrated in figure 4.9. An image is delayed by 45–80 ms where the interrupt delay time can be estimated from figure 4.6, although this interrupt time is for the parallel port and not for the FireWire bus. The interrupt delay from the parallel port is still a good indicator since it is an indicator of the systems load. Further, all the images and data are processed in time, as the system buffers a constant amount of data. This allows us to correlate the time the images were collected with GPS time to an accuracy of ≈10 ms.

4.3.3 Synchronizing Inertial Sensor Data

The CrossBow MicroNav was the chosen inertial sensor for use with the experiments. It was chosen since it is representative of the quality of sensor that is typically found in small UAV systems. The MicroNav has its own internal clock which is synchronized in a similar manner to the PC clock. Figure 4.10 shows the clock drift of the MicroNav compared to the GPS PPS signal.

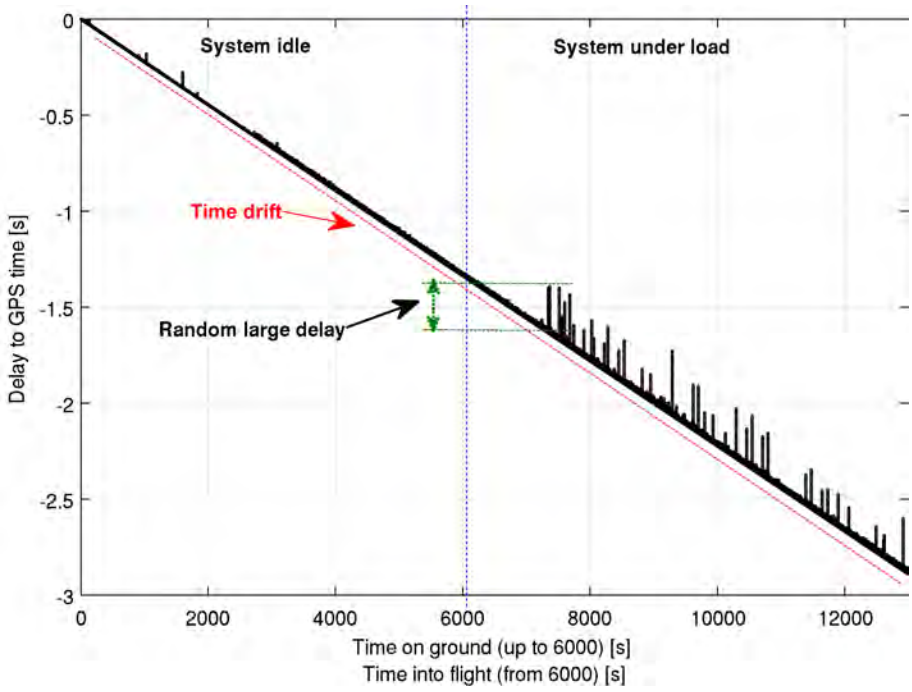


Figure 4.10: *MicroNav clock timedrift*

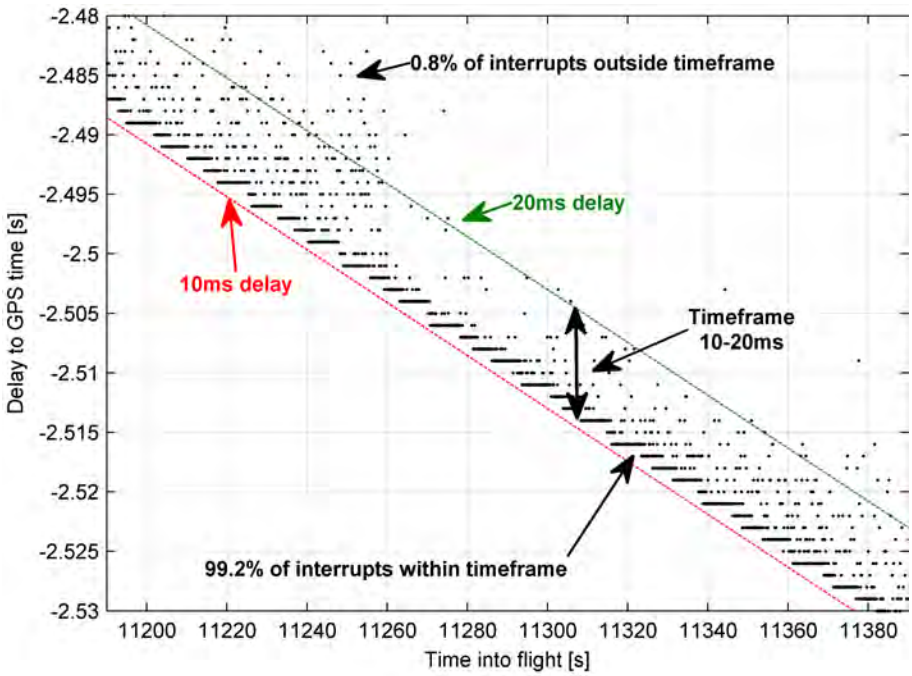


Figure 4.11: *Zoomed illustration of MicroNav delays*

Figure 4.10 shows that the drift of the MicroNav internal clock is significantly worse than the PC clock (gradient of the red dashed line). This drift can be ignored since the internal MicroNav clock is not relied on the incoming MicroNav packets are timestamped against the PC clock and adjusted for the estimated delay time between the MicroNav sensor measurement and the reception at the PC.

By analyzing the transmission delays further (as shown in figure 4.11), it was determined that approximately 1/3 of the delays are smaller than 10 ms. About 2/3 are delayed more than 10 ms. However, on average the delay time is about 10 ms, which was also verified with previous experiments. To work around this problem the average transmission time of 10 ms is delayed to the interrupt time, to synchronize the inertial sensor measurements to GPS time.

One observation from the experiment was the random large delay, as illustrated with the green dotted lines in figure 4.10. Since these random delays are rare (approximately 100 images out of 80000 images), only a few images during the entire flight are affected, which could be removed from processing. The cause for these delays is believed to be the USB (Universal Serial Port) load of the PC under full load. The PC has to serve data to 2 external hard disks and read GPS data and inertial sensor data which are all connected to the USB ports that can lead to bottlenecks.

4.4 Summary

Great care was taken to understand the timing problems associated with the images, sensors and interrupt delays. Whilst this methodology isn't perfect, it represents a good starting point for a system that is suitable for use in small UAVs. Although the inertial sensor isn't triggered by the GPS PPS it still allows estimation to an accuracy of <10 ms (see section 4.3.3). The cameras are triggered to the GPS PPS, but are timestamped by software in the computer. These delays were analyzed concluding that the images can be synchronized to an accuracy of ≈ 10 ms to GPS time.

The failure of synchronization hardware could be overcome using a statistical approach to the registered images. Removing the mean offset from the absolute along-track component allowed an estimation of the integer time offset between the GPS time and the computer clock. However, even though this hardware failure occurred, the decision was made not to repeat this cost expensive flight since the registered images allowed a work around.

Errors in geo-referencing resulting from the above uncertainties would be related to attitude and position uncertainties. A position error of 0.55 m can result from an aircraft flying at a speed of 110 kt (typical cruise speed of a Cessna 172). The position error from an angular error would be approximately 5.25 m, assuming an altitude of 300 mAGL and a roll rate of 100 °/s. These errors show that uncertainties in the data synchronization influence the registration results presented in section 6.3.1.

The airborne data acquisition itself was successful, although a vibration in the image data was detected afterwards. This vibration caused by a loose lens part will be compensated by applying a Hough transform for circles to the fisheye images. This process is explained in the next chapter following a detailed analysis of the vibration and the fisheye images.

Chapter 5

Vibration Compensation

A vibration within the fisheye images was detected soon after analyzing the first sequences of the airborne acquired fisheye images. It showed that the vibration was caused by the lens case and had an amplitude of approximately ± 4 pixels, which is larger than errors introduced from a non-perfect image rectification and therefore required to be removed. It is expected to have a similar or larger vibration in a small UAV due to its light construction, close sensor placement to the engine and vibrations from turbulence. Further, a low flying UAV will be exposed to more turbulence as turbulence increase towards the ground (Kuehr [67]). The vibration is estimated by applying a Hough transform to the fisheye image to detect circles to estimate the center of the circle, which should correlate with the vibration. The following description of removing this vibration is based on the image data only, as no reference data of the vibration itself is available.

5.1 The Problem

The vibration can be seen easily by looking at image sequences and monitoring the fisheye circle boundary. With the fisheye circle boundary the circular edge between the fisheye projection and the black surrounding is understood. The detected vibration is a relative movement of the fisheye circle boundary to the image plane. Although a vibration of the entire camera lens system is possible, it is irrelevant for the fisheye circle movement. If the entire camera moved, this would be visible as a blur effect without moving the circle around the image. The visible vibration is therefore a movement of the lens relative to the camera, as illustrated in figures 5.1 and 5.2.

The main problem with the vibration is that it can't completely be determined if the movement is translational or has components of a rotational movement. Analyzing the lens mounted to the Flea camera showed that the lens case itself is fixed to the camera but the most outer lens is loose. However, it couldn't be determined if any internal lenses were mounted in a loose manner. A translational shift, as shown in figure

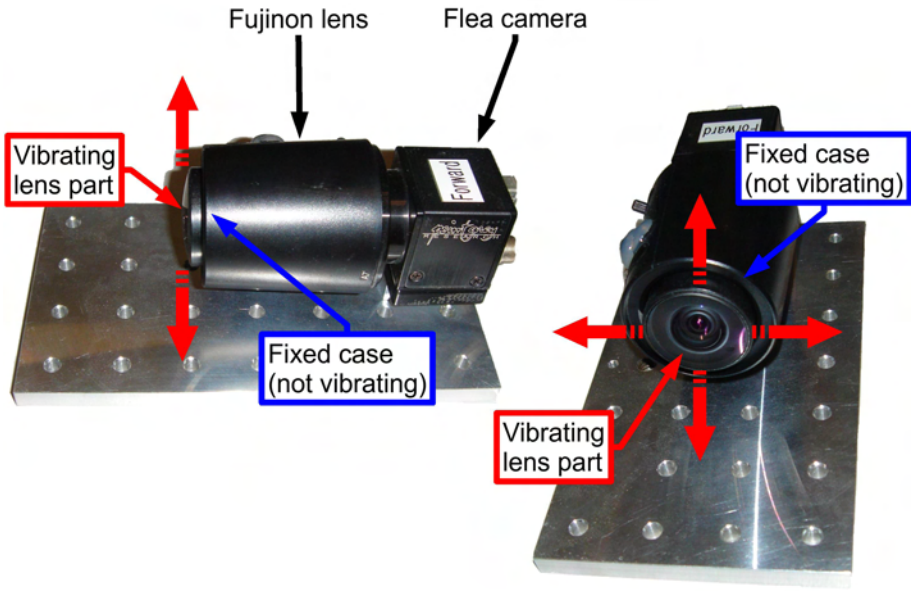


Figure 5.1: *Loose lens part as source of vibration*

5.2, would only move the fisheye lens distortion over the image, causing larger shifts towards the edges of the images. On the other hand, a rotational vibration component would distort the entire fisheye projection and show a distorted fisheye image and circle boundary when the lens rotates (vibrates) relative to the image plane. Although a rotational vibration component can't be excluded, there was no evidence found of such. Therefore, the vibration was approximated to be translational only.

A detailed analysis of moving the outer lens relative to the camera (and the lens case) shows the same effects as can be seen from the vibrating image samples. For this purpose, the loose part of the lens was moved manually in a laboratory test. The

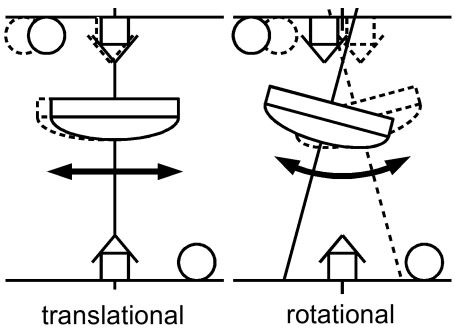
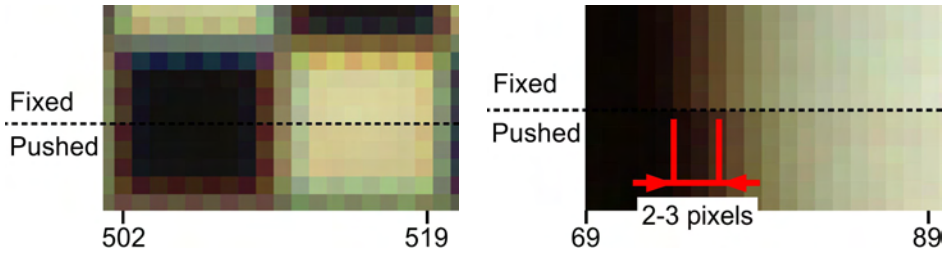


Figure 5.2: *Translational or rotational vibration component*



(a) Center of fisheye image (b) Boundary of fisheye image

Figure 5.3: *Translational shift verification test*

images in figure 5.3 show two test images, with two overlaid pictures each. The upper image shows the lens in resting position, while the lower image shows the lens pushed to one side by hand. Figure 5.3(a) shows the fisheye image center, with a translational shift of 0 pixels. Figure 5.3(b) shows the border of the fisheye image with a shift of 2–3 pixels. These tests showed that only the boundary of the fisheye lens moves by pushing the loose lens part.

The distortion due to the vibration is non-linear and increases away from the center of the fisheye image. As such, the intention of using this fisheye lens for mapping, especially with a large FOV, is influenced by this vibration as it moves the lens distortion function relative to the fisheye image.

5.2 Requirements

The vibration shifts the distortion in a translational manner which introduces uncertainties in the rectification process. If the vibration could be estimated to a 1 pixel accuracy the lens distortion functions could be placed as accurately as they would be placed in the case of a normal center estimation.

The vibration is assumed to be within ±10 pixels which also covers larger vibrations. The vibration compensation operates on a 0.5 pixel accuracy allowing testing for sub-pixel solutions. The radius of the circle boundary is believed not to change more than a few pixels in size, but due to the adaptive behavior of the circle boundary selection, the radius may vary more than expected. The radius range is therefore set to ±10 pixels as well, with 0.5 pixel increment steps. However, results show no large variations of the radius size between following images and images in short sequences, as the sun influence wouldn't change a lot.

The inertial sensor data was also analyzed to see, if any frequencies showed a similar behavior as detected by the lens vibration. There were no correlations found which could be related to the low sampling rate of 100 Hz and the low quality of the sensor

(CrossBow MicroNav).

5.3 Approach to the Vibration Compensation

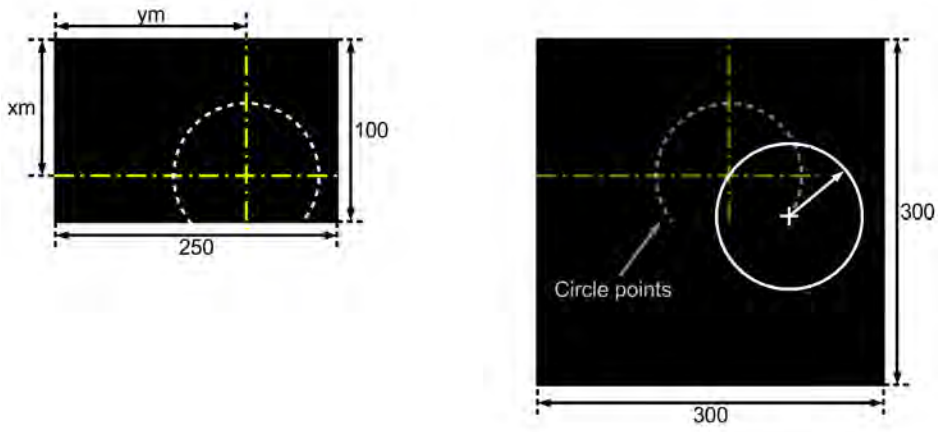
The classical Hough Transform used today was introduced in 1972 by Duda and Hart [35] (derived from the initial idea by Hough [53]) to detect lines and curves in images. Since then the Hough transform evolved to detect other shapes, such as circles. The Hough transform used is a circle Hough transform, as described by Rhody and Carlson [87]. Many other authors [21, 62, 80] take a similar approach, all using the general Hough equation for circles as given in equation (5.1). The Hough transform is a computative intensive process. Real-time applications of the Hough transform have been shown using a FPGA (Field Programmable Gate-Array) or GPU (Graphics Processing Unit) [77, 105] and could be used to implement the presented Hough transform as it is highly parallel processing.

The extended Hough transform to detect circles is extending the classical Hough transform in another dimension. The circle is described by the radius (r_{circle}) and the two circle center coordinates xm and ym, with the actual circle points x_{point} and y_{point}:

$$r_{circle} = \sqrt{(x_{point} - xm)^2 + (y_{point} - ym)^2} \tag{5.1}$$

The variables x_{point} and y_{point} are used as the input points of the input image containing the object to detect with the Hough transform.

The classical Hough transform adds all potential lines through each detected point in an image to the accumulator space. In the case of a circle shape, the accumulator adds circles instead of lines for each detected point, illustrated in figure 5.4. The image within figure 5.4(a) shows detected points of a circle like shape which will now be attempted to be detected by the circle Hough transform. It then draws a circle (or circles) for each testing point, as illustrated in figure 5.4(b) (these testing points are the detected circle boundary points, as described in section 5.4.2) and adds the results to the accumulator space. Figure 5.4(c) finally shows the accumulator space after drawing a circle (of known radius) for each test point. Peaks in the accumulator space show potential centers of circles with the specific radius. In this accumulator space there is only one peak visible, which provides the center coordinates xm and ym of the circle shape from the binary image. Applying the found coordinates with the known radius (figure 5.4(d)) gives the same coordinates as they were given from the input image (figure 5.4(a)) and a circle with the same radius is drawn.



(a) Binary image showing a circle-like shape (b) Accumulator space after first circle

(c) Accumulator space with all circles (d) Matching circle after interpretation

Figure 5.4: *Hough transform illustration on circle detection*

If, however, the radius is unknown as well, the accumulator space becomes a collection of accumulator spaces, as illustrated in figure 5.5. Each accumulator space represents one radius. Further, each of those accumulator spaces will generate one or multiple peaks, each defining a circle center with the specific radius for this accumulator. The matching circle which best represents the shape of the detected points will produce the highest peak. Further, not matching radii often produce multiple peaks allocated around the true circle center (e. g. figure 5.5, first accumulator plane, where the radius chosen was too large).

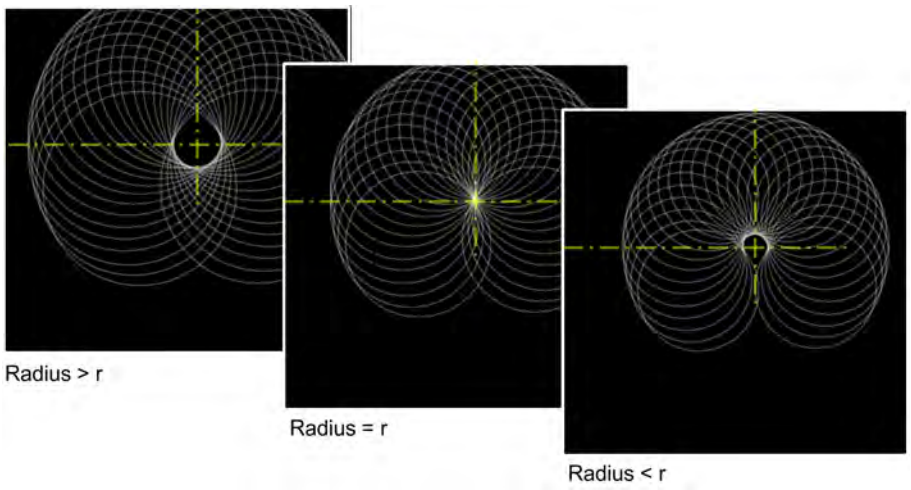


Figure 5.5: *Accumulator space for each of the tested radii*

The algorithm in appendix 5.3 describes the extended Hough transform to detect circles. Variables with i_{xm},i_{ym} and i_{xy} are counting variables of the for-loops (iteration loops). The first two loops wander through the search area for the center coordinate. This area can be the same, smaller or larger as the input image (see following paragraphs for more details). The third loop wanders through all the detected points from the binary image (given as a coordinate $i_{xy} = (x; y)$). The Hough transform is calculated for each point, but only valid points are then registered in the accumulator H. Valid points are given by the range of radii testing (e. g. $r_v = r \pm 10\,\text{pixels}$)

Extended Hough Transform

```
foreach i_xm                          % x-axis pixel
   foreach i_ym                       % y-axis pixel
      foreach i_xy                    % Detected circle point coordinate (x; y)

         r ⇐ HT(i_xm, i_ym, i_xy)     % Hough transform for current position (i_xm; i_xm) ...
                                      % and coordinate i_xy

         if r ∈ r_v                   % When r is a valid radius ...
            H(r_v(r)) ⇐ +1            % increment accumulator by 1
         end

      end
   end
end
```

With 3 now unknown parameters the extended Hough transform becomes a computational intensive problem. For example, a binary image of 1024×768 pixels resolution, 500 detected points and an estimated circle radius within ±100 pixels would result in either time consuming for-loops or massive matrices:

- For-loops: $1024 \cdot 768 \cdot 201 \cdot 500 \approx 80$ billion times calculating equation (5.1) and adding the result to the correct accumulator spaces.
- Matrices: $1024 \cdot 768 \cdot 201 \cdot 500 \cdot 4\,\text{byte} \approx 316\,\text{GB}$ per matrix, where 4 matrices would be required.

The problem can be solved using combinations of matrices and for-loops to reduce the amount of RAM (Random Access Memory) and for-loop iterations. The solution in this case was to use the fisheye image features. Most of the fisheye images are likely to be within a certain range of vibration, which was defined to be ±10 pixels in x- and y-direction at 0.5 pixels increments. Further, the radius is assumed to stay within a ±10 pixel range at 0.5 pixels increments. These assumptions considerably reduce the computational complexity of the Hough transform. With these assumptions and 500 detected points, the problem is reduced by a factor of 1000 to:

- For-loops: $41 \cdot 41 \cdot 41 \cdot 500 \approx 34$ million times calculating equation (5.1) and adding the result to the correct accumulator spaces.

- Matrices: $41 \cdot 41 \cdot 41 \cdot 500 \cdot 4\,\text{byte} \approx 131\,\text{MB}$ per matrix, where 4 matrices would be required.

After establishing the algorithmic background, a fast implementation of the Hough transform for MATLAB is described. The implementation minimizes the computational time by using Matlab's matrix calculation feature.

5.3.1 Speeding up the Hough Transform

Attempts at improving the speed were taken by implementing the Hough transform for the Matlab matrix processing strength. Matlab has built-in processor specific algorithms for faster processing of parallel calculations, such as those in the Hough transform. Instead of implementing a common for-loop nesting the for-loops were removed and large matrices used instead. Implementing this algorithm into single matrices to remove the for-loops is done by creating an array entry for each possible solution from each input. Inputs are the potential circle points with x,y-coordinates, potential radii and the the estimated center coordinates $(xm; ym)$.

The Hough transform can then be calculated as highly parallel operations, which are built into MATLAB. The resulting computational time is variable due to the variable amount of detected points (see table 5.1). However, the general speed increase with the matrix-form instead of for-loops is a factor of 4.

5.4 Selection of the Fisheye Circle Boundary

The circle Hough transform can estimate the vibration in fisheye images in theory by using the fisheye circle boundary. Although this is at the cost of intensive processing. The fisheye circle boundary is the circular edge between the image and the black surroundings, as seen in typical fisheye images (see figure 5.9 page 70). However, the Hough transform faces problems of external influences by adding noise that decreases the quality of the estimation.

The main issue with the fisheye circle boundary selection is influences caused by the sun or the camera mounting. Since sun influences can vary drastically the edge detector is required to be adaptive to different light effects. In the common case where the edge detector fails to provide a clean circle boundary, the Hough transform and the respective analyzing technique is capturing most of these errors. However, the better the selection of this circle boundary is in the first place the better the entire algorithm is. Some of the major influencing factors are described in the following paragraphs, supported with images from the flight.

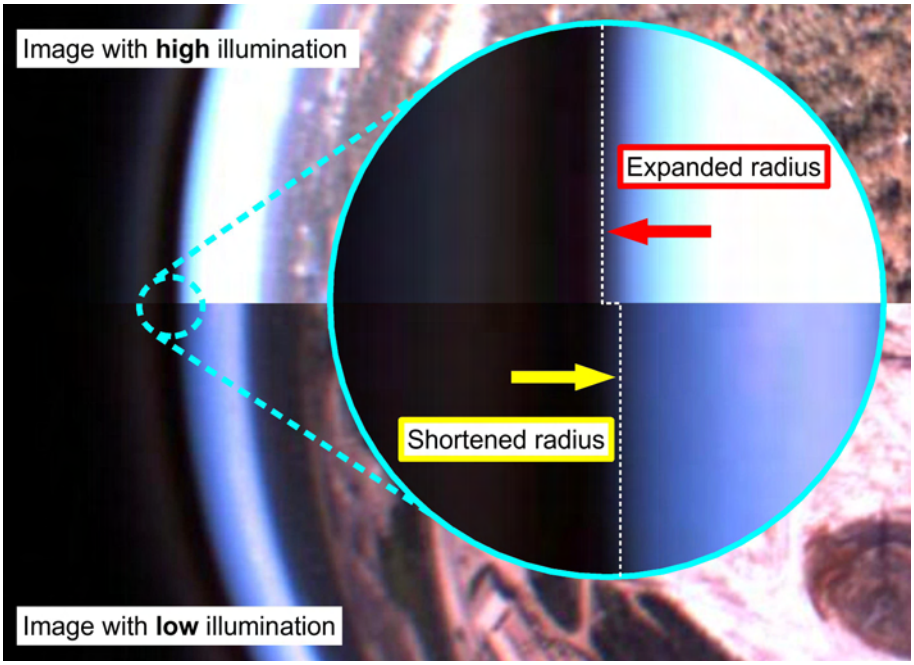


Figure 5.6: *Different illuminated images show expanded borders*

5.4.1 Factors Influencing the Circle Boundary

There are several factors increasing the issue of different illumination. These are illumination, sun glare or the effects of a climbing, descending and turning aircraft, and also the projection of the aircraft hull onto the image itself. Illumination adds errors to the edge detection, such that it can increase or decrease the radius of the fisheye circle. It is also possible that it shifts the circle towards one side of the lens or influences the edge detection. Therefore, it is important to use a circle like matching pattern (such as the Hough transform for circles) to cover some of these influences. Examples of different influences are given in the following paragraphs.

Illumination

An example of the effect of illumination is given in figure 5.6, where it expands one side of the circle and would result in errors. However, the flight was conducted during the time where the sun had reached its zenith, minimizing this influence.

Sun Glare

Sun glare is sun reflections (or direct sunlight) captured in images, such as from water, snow, roof or car surfaces. An example of sun glare creating a white line over a fisheye image is shown in figure 5.7. The white line is of high intensity and expands the

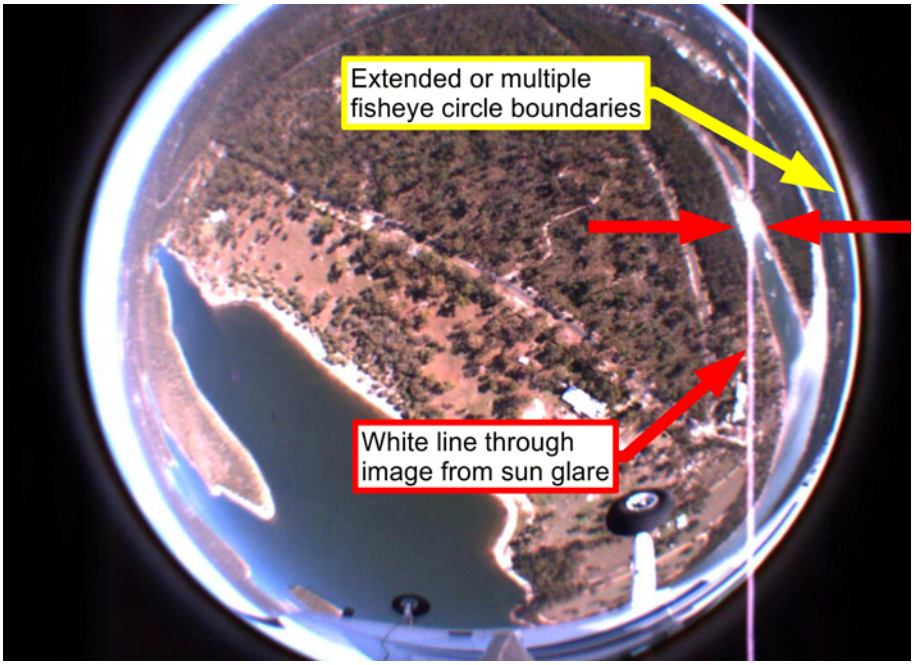


Figure 5.7: *Sun-glare appearance as a vertical line through the image*

fisheye image boundary or shows multiple borders in just one direction (one part of the image), as shown in figure 5.7. Sun glare therefore often only influences the edge detection in one direction, adding errors to the vibration compensation. Sun glare can be detected easily and such areas (or the entire image) can be removed from the sequences of images used for further processing.

Wrong Edge Followed

A climbing, descending or turning aircraft doesn't directly affect the fisheye image circle itself, but it may disturb the edge detection result. Figure 5.8 shows an example of a climbing aircraft, where the horizon merges together with the fisheye circle boundary. The edge detection can then fail in separating the horizon and the boundary from each other. The edge detection may capture wrong edges or fail completely. Images containing forest, mountains or high buildings can cause major problems.

Aircraft in Image

Further influences include uncertainties in the image that are of internal matter. First, there is the projection of the aircraft in the image. The aircraft, as expected, vibrates in a relative manner to the camera, which was mounted underneath the wing. Due to features on the aircraft's fuselage that are often detected by the edge detector, the

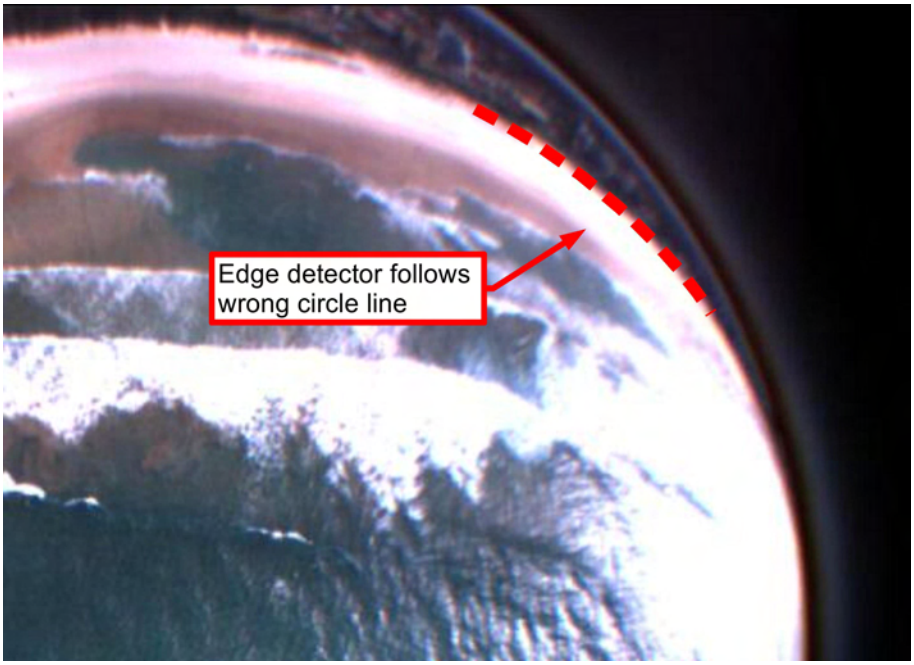


Figure 5.8: *Climbing aircraft merges horizon with circle boundary*

area around the aircraft isn't processed. Figure 5.9 illustrates the positions of these areas marked with red rectangles.

Second, on the upper part of the image, where the circle boundary extends outside the image, the circle boundary becomes unsharp. As illustrated in figure 5.9 with yellow parallelograms, these areas are also cut-off. The remaining area for the circle boundary detection is marked with green dashed lines.

The conclusion out of these influences is that the uncertainty increases over water (due to sun glare and higher illumination of the picture) and during the time the aircraft is making turns. Due to the fact that this research is using a fisheye lens to perform mapping even during turns of the aircraft, these disturbances play a significant role in contributing to the uncertainties in terrain mapping applications and can't be ignored. Further, because large areas of the images are removed from processing, the vibration compensation is almost only along the x-axis, where the y-axis should show less vibration. Since a true fisheye lens would provide a complete projection of the fisheye circle (see figure 6.5(a)), the influences described in the previous paragraph are less. A true fisheye image would not only improve the circle boundary detection, but also improve the overall points that are available to the Hough transform. Knowing the influences of the circle boundary detection, the technique used to actually detect the circle boundary must now be proceeded to.

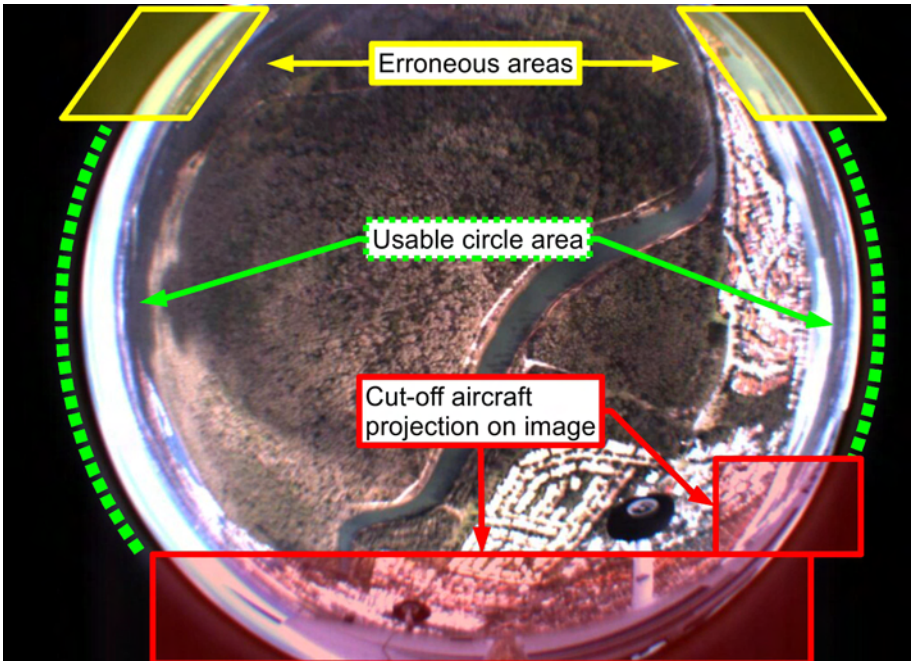


Figure 5.9: *Mounting and camera specific influences*

5.4.2 The Canny Edge Detector

This section presents the implementation of the edge detection. First, the analytical approach to the edge detection is discussed. Prior to presenting the edge detection in detail a discussion of why a previous approach has been dropped and a simplified approach has been used instead takes place.

Analysis

To achieve an edge detection for a broad set of pictures a random set of 80 images was chosen from the flight test images. These images should contain almost all the above discussed cases of different light exposure, but also different ground texture. Further, some of the images also show merging horizons with the fisheye circle boundary.

First, the images were analyzed in different color spaces, such as RGB (red, green, blue) and HSV (hue, saturation, value). There was no single color space found that would improve the edge detection in all pictures. However, it was noticed, that the HSV-value plane sometimes increases the contrast of the fisheye circle boundary. However, in pictures taken over the ocean or forest, the HSV-value doesn't show any improvement. Again, a higher contrast of the circle boundary could be found in the RGB-red plane, but only in images with clouds in the sky. In blue sky, the circle boundary and the sky isn't separable. Therefore, the edge detector is applied on the

grayscale image, which is the converted RGB image to gray scale.

The only improvement that was implemented adjusts the histogram of the image. Spreading the histogram gives an increased contrast to the circle boundary. However, spreading the histogram may also expand or shorten the radius. As long as the expansion is uniform the center coordinate of the circle will not be affected.

The Canny edge detector was used as an edge detecting method. As Shin et al. [91] presented, Canny performs best under unknown threshold conditions. This is exactly is required as the threshold setting should be automatic for the various different types of images.

Edge Detection

The Canny [17] edge detection method was chosen due to its specific behavior in the case of fragmented lines. The Canny edge detector uses a two step approach (two thresholds) to detect edges. First, strong edges are detected, followed by the second detection of weaker edges. However, the Canny edge detector validates the weaker edges by verifying if the weaker edges are ongoing edges from the strong edges. In the case of this fisheye boundary detection, this behavior is desirable to completely detect the circle boundary without too much fragmentation.

A previously presented approach (see Gurtner et al. [46]) of selecting the circle boundary in a 2 step approach has been dropped with the ongoing research into vibration compensation. The reason that a 2 step approach was chosen was to minimize the chances of multiple boundary selections. However, with refining the processing technique and limiting the area to search for a circle within the Hough transform, this 2 step approach has been dropped, without notably worsen the overall outcome.

Before the edges are detected the image requires pre-processing. First, the features in the image are limited to the area of interest, which is the expected radius r and a range dr (figure 5.11(b)). The remaining image features containing the fisheye circle boundary are then converted to a grayscale image. The grayscale image is then allowed to adjust its histogram by expanding the color range of 0.25–0.75 to 0–1 (with black and white being 0 and 1), as illustrated in figure 5.11(c).

This grayscale image containing the circle boundary in a circle ring is then filtered by the Canny edge detector. Due to the fact that the search area was previously limited to $R = r \pm dr$, the edge detector often selects these circles as well. These circles are removed by limiting dr by another pixel to $R = r \pm (dr - 1)$. Figure 5.11(d) shows a filtered image containing all potential fisheye circle boundaries. The Canny edge detector used is a built-in function in MATLAB that allows the user to set thresholds and search behavior.

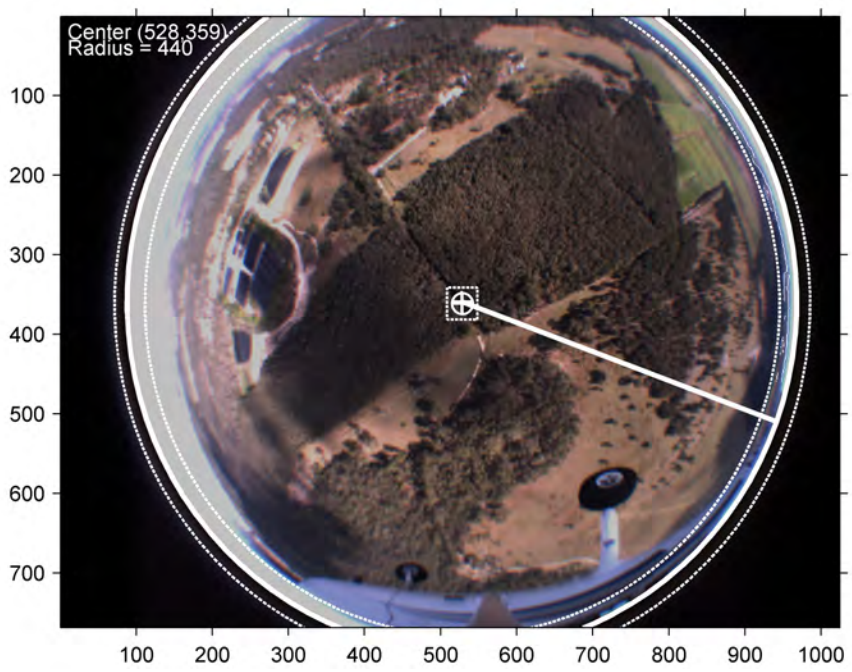


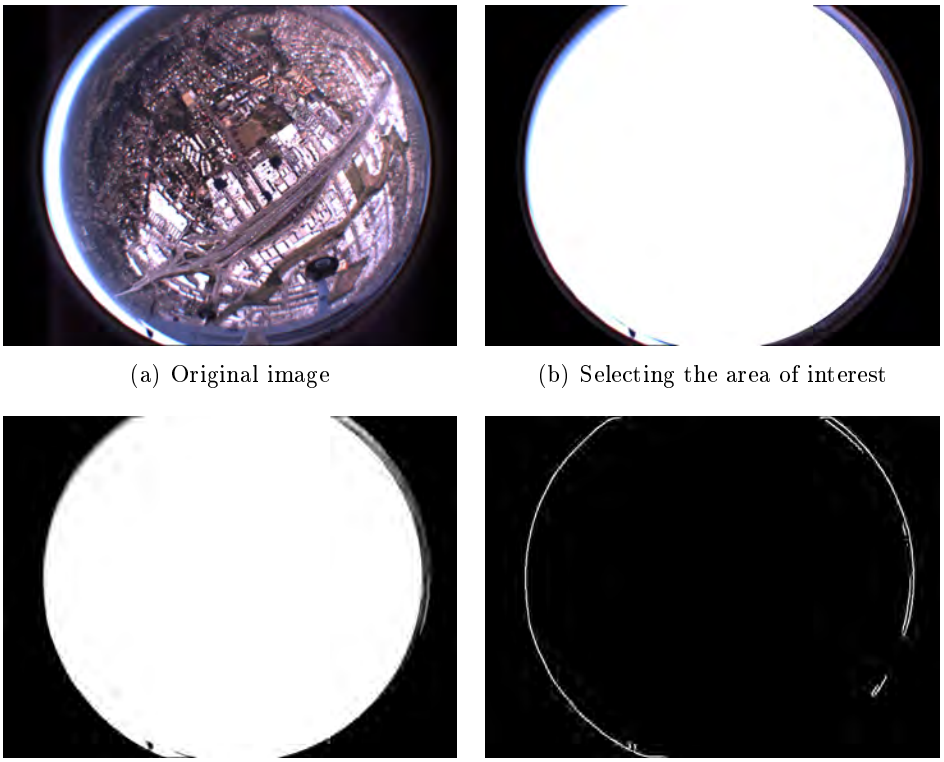
Figure 5.10: *Illustration of circle boundary selection within the radius range R*

The settings for the Canny edge detector were derived by applying the filter to a random set of images. The performance for different settings was then analyzed and the settings which came closest to the expectations of selecting the fisheye circle boundary were chosen. The Canny edge detector was found to achieve a better performance by a setting of $\sigma = 3$ and leaving the thresholds to be set by its internal routine. A σ greater than 1 was chosen because the circle boundary is a smooth feature whereas the ground is usually rougher. A Gaussian filter with $\sigma = 3$ smooths these rough surfaces away while the already smooth circle boundary remains unaffected.

The finding of leaving the threshold setting to the algorithm correlates to the previously mentioned strong differences in image light exposures. Therefore, the Canny edge detector provides an adaptive selection of the circle boundary.

5.5 Results and Analysis

As stated in the introduction, the vibration compensation doesn't have any data for proofing its performance. Indicators to proof its correctness can still be found, such as observing the vibration directions and magnitude by hand.



(a) Original image

(b) Selecting the area of interest

(c) Grayscale image with expanded histogram

(d) After the Canny edge detector

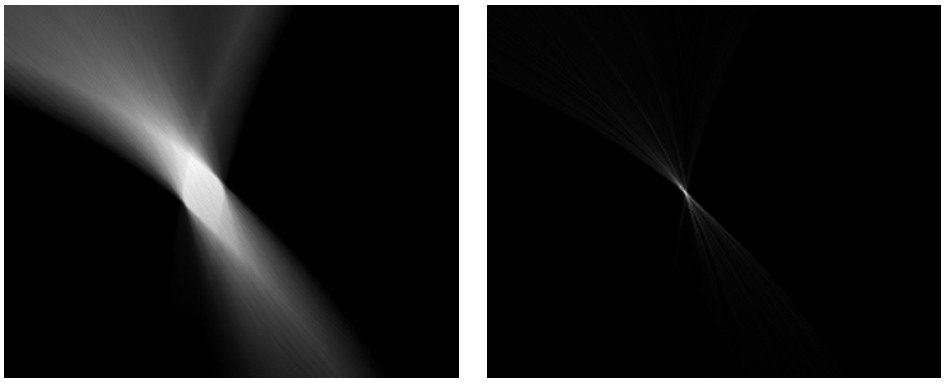
Figure 5.11: *Applying the Canny edge detector to an image*

5.5.1 Results of Single Images

The resulting accumulator spaces, as explained in section 5.3, can be imagined as a set of overlaid images (a set of 2D arrays), with each accumulator space containing a potential solution. The idea is that the accumulator space containing the maximum peak value is the matching circle with its correspondent radius r and center coordinates $(x_m; y_m)$.

However, in practice, multiple possible circles or circles with different radii, but similar maxima are found. Figure 5.13 shows the distribution of matches per radius (solid line with circles) over a range of radii. The maxima are reached at r's between 433–442, and 2 peaks at $r = 434$ and $r = 441$ are shown. There is no clear separation of one matching radius visible, due to strong noise on the signal. The signal noise's source is from a non-circular fisheye circle boundary or misleading pixels. A non-circular fisheye circle boundary is like a blur effect over the Hough transform solution, if the circle is actually circular. This noise can also be visualized by accumulating each accumulator space to a single cell, as shown in figure 5.12. Figure 5.12 shows

the accumulator space for 60 tested radii, shown as white areas. The white area turns sharply to black, because there is no data available for these other radii.



(a) Accumulated spaces (b) Single space

Figure 5.12: *Accumulator spaces for all tested radii (left) and for a single radius (right)*

As Ryde and Hu [89] mentioned, further processing to interpret the Hough transform is required. By applying an AND operation (details in the following section) for each cell of the accumulator space an identifiable separation of one peak (radius) becomes visible (dashed line in figure 5.13). Further examples, including the original images are given in figure 5.14 to illustrate different performances on a variety of images.

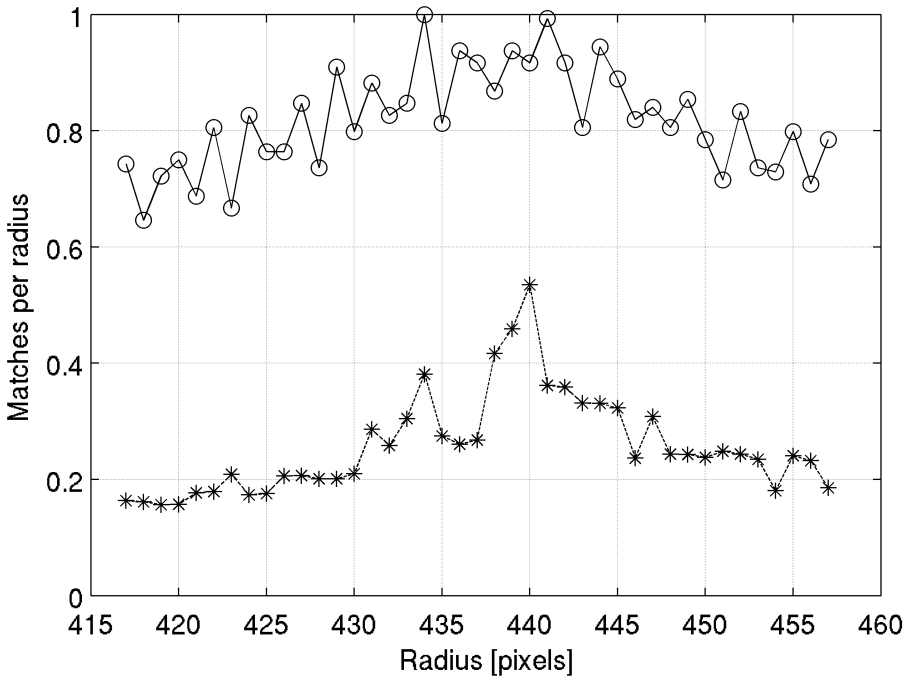
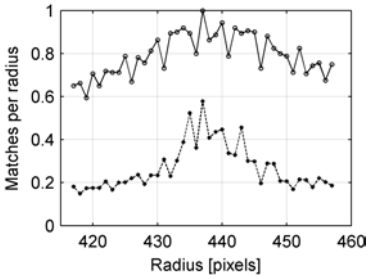
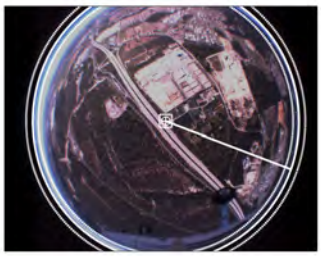


Figure 5.13: *Typical radius distribution before and after AND operation*

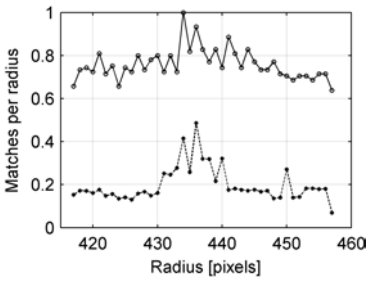
(a) Original image with detected circle



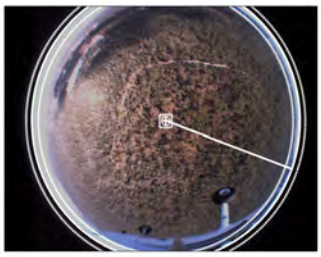
(b) Hough transform results



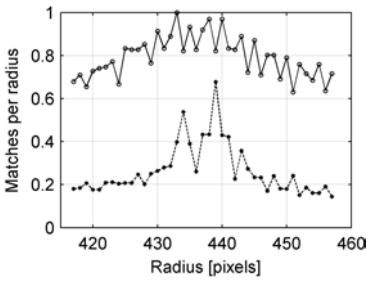
(c) Original image with detected circle



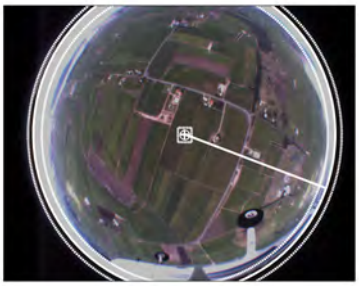
(d) Hough transform results



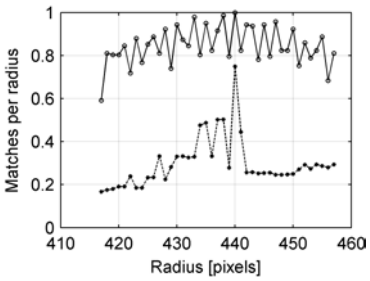
(e) Original image with detected circle



(f) Hough transform results



(g) Original image with detected circle



(h) Hough transform results

Figure 5.14: *Examples of the Hough transform*

5.5.2 Results based on Image Sequences

Following the vibration compensation over a sequence of images shows some of the problems described in the previous sections. For example, figure 5.15 shows the relative vibration (differences) of the radius r and the center coordinates $(xm; ym)$ over 1000 images. As expected, the vibration of the radius changes only minor whereas the x-axis shows larger amplitudes than the y-axis.

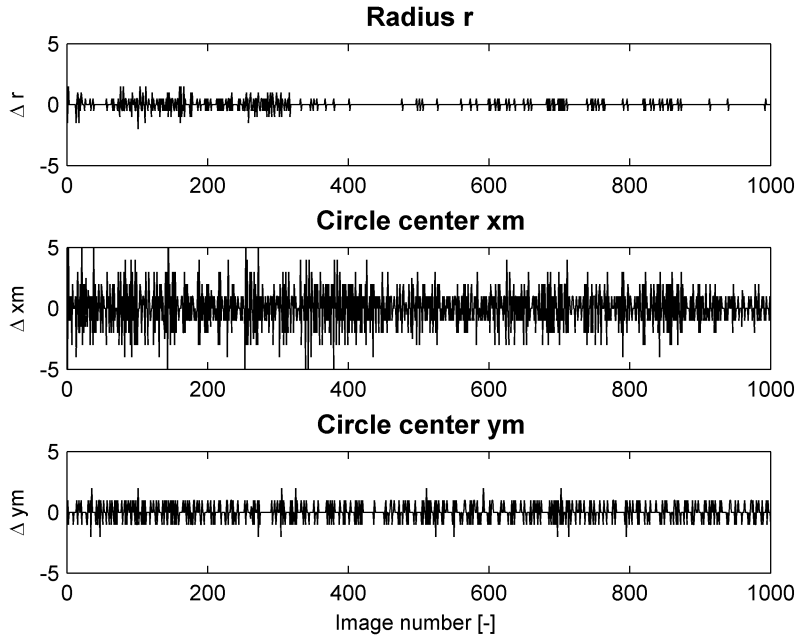


Figure 5.15: *Vibration magnitudes of Δr, Δxm and Δym*

In the first diagram it can be seen that the radius stays within ± 2 pixels, with slightly larger values up to image sample 400. In this regard the problem lies with detecting clear peak probability matches of the radii, shown in figure 5.16 indicating a "nuisance" (radii detection problem). Figure 5.16 shows the radii matches over the same sequence of images showing the probability of each radius per image. The radii matches are the simplified output from the accumulator space multiplied by the number of radius pixels matching the fisheye circle boundary on each side. A strong certainty of the vibration estimation is found in dark red areas whereas yellow to blue areas show a weak correlation between the Hough transform solution and the detected circle boundary. Within the nuisance no clear detection of the circle boundary is possible.

Except for this nuisance before image sample 400 the vibration compensation works according to our estimates. These nuisances are errors from the edge detection (illumination, sun glare) and couldn't be removed by the Hough transform alone. The result

presented in this diagram is therefore the refined data from the Hough transform by using the AND operation explained in the following section.

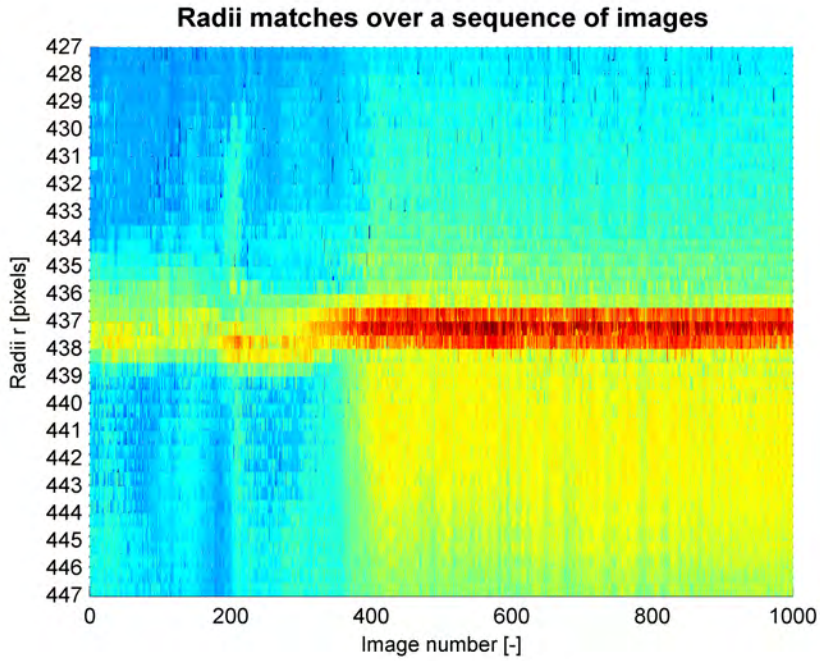


Figure 5.16: *Spread matrix showing the certainties of radii over a sequence of images*

5.5.3 Refined Results

The AND operation is used to measure the certainty of the Hough transform solution. The AND operation takes the result from each accumulator space and "compares" (AND operation) it with the detected pixels from the circle boundary detection. This is illustrated as *pleft* and *pright* in the algorithm on page 80. The more pixels that match, the more certain a solution is a true solution. This is necessary since the Hough transform itself is lacking a small signal to noise ratio in our case. This low signal to noise ratio is graphically illustrated in figure 5.13 page 75 where the original signal is on top with the edges marked with circles. However, this AND implementation faces some problems.

One problem is the amount of the overall pixels detected from the circle boundary detection, which varies between 210 to 1435 pixels. Another problem is the side on which the pixels are detected. The area on the left-hand side is slightly bigger than the right-hand side. On average there are 32 % more points detected on the left-hand side. A matching circle would more likely match the left-hand side which isn't appropriate.

The implementation of this AND operation must therefore take into account this uneven problem. Table 5.1 gives an overview of the uneven distribution of the detection points. First, the detected points are counted on both sides (left and right). There is a minimum of 100 points that are required on each side, otherwise it is likely that the Hough transform will fail completely. As shown in table 5.1, there are 234 (547) cases where 100 points couldn't be detected on one side.

The solution from each radius of the Hough transform is then taken and compared to the detected pixels on each side separately, since there are on average 32 % more points detected on the left-hand side. This ensures no advanteage to the left-hand side. As shown in figure 5.16, the certainty drops if the left-hand or right-hand side is having problems matching the Hough transform solution. Low certainties can now be detected but so far no mechanism is implemented, to improve the result acordingly.

Table 5.1: *Overview of differences of the left- and right-hand side*

Observation	left	right
Not enough circle boundary points detected (<100 points):	243	547
More than twice as many detected points on other side:	1.3 %	4.4 %
Average amount of detected points per side:	100 %	76 %

The algorithm **AND Operation** on page 80 describes this entire process again in a high-level point of view. The image file is read and *DetectCircleBoundary* is applied. *DetectCircleBoundary* is the process described in the previous sections (section 5.1 and 5.3). It applies the described circle boundary detection to the image and results in a binary image containing potential edge points of the fisheye image boundary. Afterwards, the AND operation is applied to both sides of the circle (*pleft* and *pright*). The resulting solution is a "filtered" radius distribution signal, as it was illustrated in figure 5.13.

AND Operation

```
I ⇐ ReadImageFile                    % Read image file

Ibw ⇐ DetectCircleBoundary(I)        % Apply the circle boundary detector to image
Ileft ⇐ CropLeft(Ibw)                % Crop
Iright ⇐ CropRight(Ibw)              %

H ⇐ HoughTransform(Ibw)              %

foreach r                            % Radius of the Hough transform H
    pleft(r) = Ileft&H(r)            % Probability for the left-hand side
    pright(r) = Iright&H(r)          % Probability for the right-hand side
    p(r) = left(r) · right(r)        % Probability for both sides
end

maximum(p)                           % most probable would be the value with ...
                                     % the highest p-value
```

5.6 Summary

The results presented show that a vibration compensation with a Hough transform is feasible. However, the influences of the sun and the non-perfect circle projection of the fisheye lens decreases chances of detecting a matching circle boundary. Further, since there is no truth data available, the vibration compensation relies on the certainty of the results presented by the vibration compensation.

Further improvements of the vibration compensation can be seen in the edge detection and analysis of the Hough transform solutions. The edge detection could be improved by actually detecting the edge according to the color change. The edge often changes from sky colors to black (lens case) which could potentially improve the edge detection in straight and level flight. Further, currently only global peaks are tested for from the Hough transform ignoring any local peaks. There might be an improvement testing for these local peaks as well, but it will also increase the computation time by large amounts. Last, the circle shape could also be replaced by an ellipsis, by adding another dimension to the Hough transform. An ellipsis is believed to match the slightly distorted circle shape in a better way and can produce higher certainties

with the combination of the AND operation. Further, the presented Hough transform isn't ideally weighted to the x- and y-axis. It shows a higher x-axis sensitivity due to limited fisheye boundary information for the y-axis.

The presented Hough transform provides the required center coordinates compensated for the vibration. The rectification process places the lens distortion accordingly. The detected vibration shows problems that occur using typical sensors for small UAV applications. The demonstrated algorithm can be applied to any images containing circle-like shapes from a fisheye image by changing the specific parameters. The following chapter presents the derived fisheye lens functions and the registration and mapping results using the airborne acquired images.

Chapter 6

Results and Analysis

The approach to this research was presented in section 3, where fisheye lenses were analyzed in detail. Remember that the spherical lens model (figure 2.2) provides a description of the lens distortion $r(\theta)$ as a function of the viewing angle θ. The verification of the lens function for the Omnitech Robotics ORIFL190-3 lens is given in section 6.1.1. Multiple lens functions for the Fujinon lens were derived using the calibration disk described in section 3.1.1, and the results of this calibration are now presented section 6.1.2.

The images acquired during the airborne data acquisition are used to create mosaic maps, as presented in section 6.3. The results are analyzed for performance and are now shown in pictures previously mentioned with problems of non-calibrated and low-quality sensors (see chapter 4). The results presented in this section will be further discussed in chapter 7, which also answers the research questions from the introduction in section 1.

6.1 Fisheye Lens Calibration

6.1.1 Estimated Parameters for ORIFL190-3

The fisheye lens ORIFL190-3 from Omnitech Robotics provides optical lens parameters and diagrams of the lens distortion. However, the lens distortion function is given in two diagrams (F13 and F14 in the ORIFL190-3 datasheet in appendix A.6.1), but polynomial coefficients given couldn't be related to those diagrams. This is likely because the numbers are given at an unsuitable precision. The lens function must therefore be derived by a hand method.

The lens function is derived using diagram F14. The measured points are listed in table 6.1, where factor $f_{non-linear}$ stands for the non-linearity of the lens distortion to the general lens function $r_{general}(\theta)$:

$$f_{non-linear}(\theta) = \frac{r_{ORIFL190-3}(\theta)}{r_{general}(\theta)} \tag{6.1}$$

and the resulting lens function:

$$r_{ORIFL190-3}(\theta) = \\ -6.2030 \cdot 10^{-10} \cdot \theta^3 + 2.5623 \cdot 10^{-8} \cdot \theta^2 + 2.1082 \cdot 10^{-5} \cdot \theta + 2.6903 \cdot 10^{-6} \tag{6.2}$$

Table 6.1: *Measured points from datasheet diagram*

Angle θ [°]	$f_{non-linear}(\theta)$ [-]
0	1.000
10	0.999
20	0.996
30	0.988
40	0.978
50	0.963
60	0.943
70	0.919
80	0.888
90	0.850

The derived lens function from diagram F14 was then verified with the measurements from diagram F13. The derived $r(\theta)$-function in figure 6.1 makes use of the values from table 6.1, marked as red crosses. The measured points from diagram F13 are marked with blue stars and listed as $F13_{measured}$ in table 6.2. Figure 6.1 shows the approximation of the measured lens points from the diagram F14 using the polynomial approximation discussed in the literature review, section 2.3. The linear dashed line represents the lens approximation function $r = f \cdot \theta$ which approximates the lens function to 1 % at $\theta < 30\,°$. However, at larger angles the error of this function increases to >10 % and is therefore not suitable for accurate rectification purposes.

The errors given in table 6.2 are calculated in the following equations, where the pixel cell size on the CCD sensor array is $PS_{Flea} = 4.65\,\mu$m. The measured values from the diagrams F13 and F14 are listed and related to angles θ. The last column ($\Delta error_{pixels}$) shows the measurement error in pixels, between the values from diagram F13 and F14, which were measured by hand. It shows that the errors are of sub-pixel size except for one value being larger than 1 pixel.

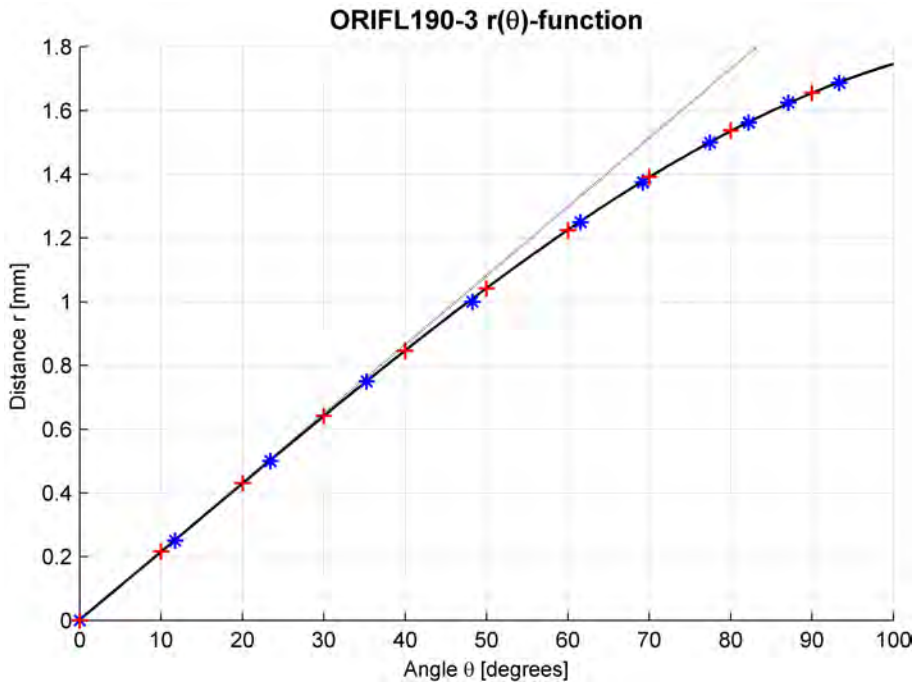


Figure 6.1: *Verification of the measurements from the two diagrams*

$$\Delta error_{\mu m} = F13_{measured} - F14_{approximated} \tag{6.3}$$

$$\Delta error_{pixels} = \frac{\Delta error_{\mu m}}{PS_{Flea}} \tag{6.4}$$

Further, the general lens function approximation suggested by Kannala (see equation (2.1)) failed to approximate the given points accurately. Figure 6.2 shows Kannala's equation (2.1) compared to the polynomial (6.2) approximation. A 3rd-order polynomial approximates the given lens points by $<1\,\%$. Therefore, a polynomial approximation of the fisheye lens function is used for both lenses.

6.1.2 Estimated Parameters for Fujinon Lens

In section 3.1.1 another approach to determine the lenses distortion in the case of the Fujinon fisheye lens was explained. Instead, there were 8 lens functions taken over the lens, as illustrated in figure 2.5. The measured points for each of the 8 functions are listed in appendix A.1. The lens functions are shown in figure 6.3. Since there are 8 different lens functions, only one lens function is listed in equation (6.5). All other lens functions are listed in appendix A.1.2.

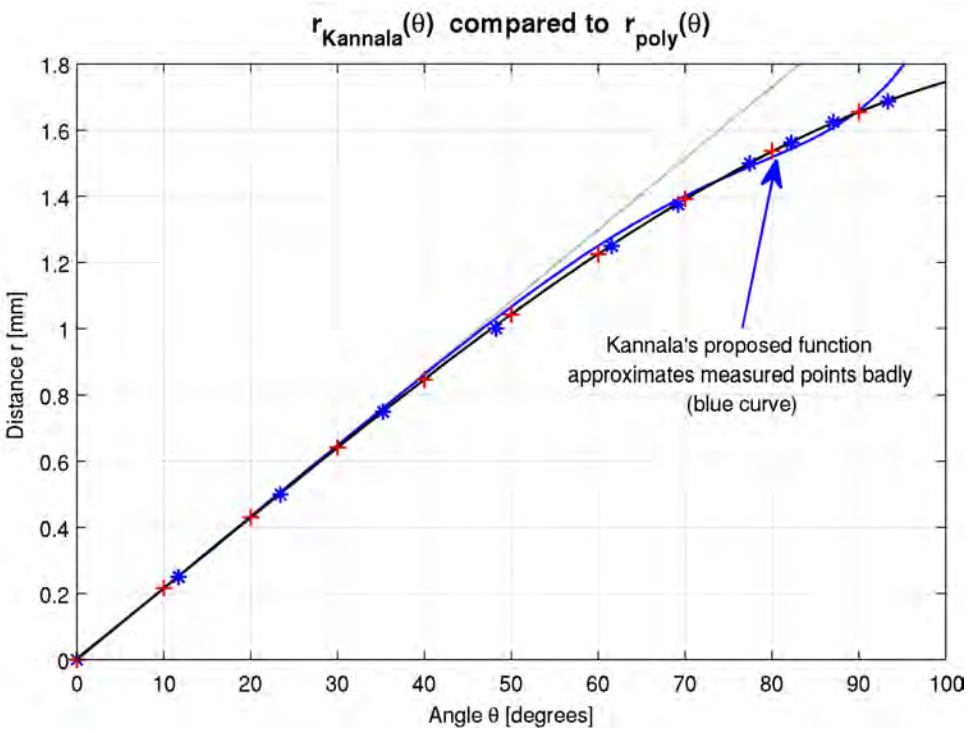


Figure 6.2: *Proposed function approximation by Kannala*

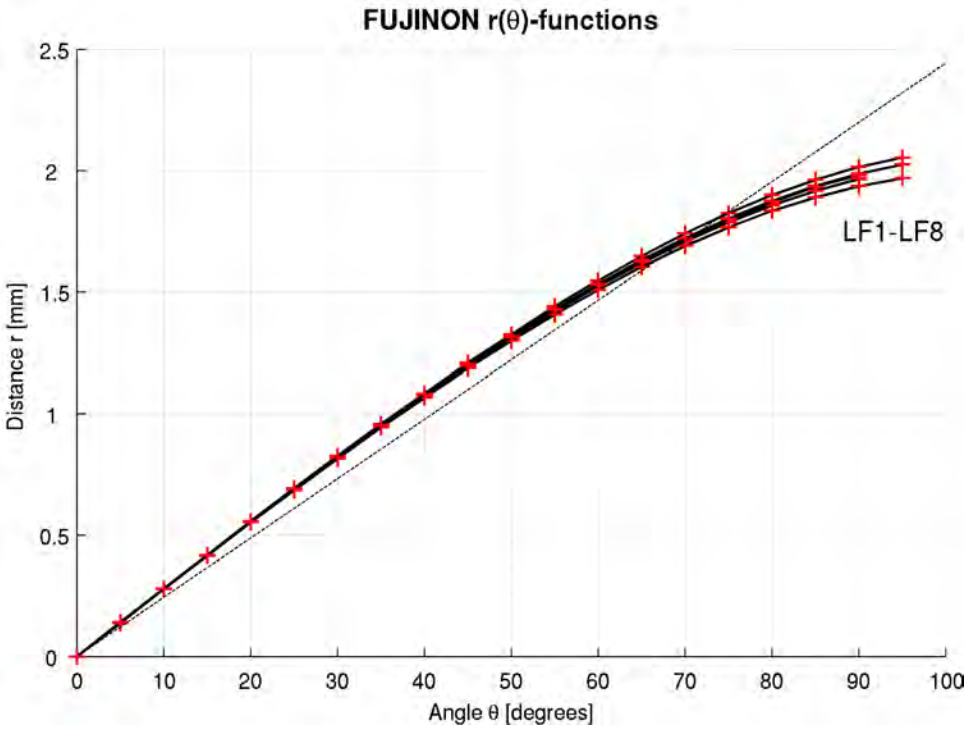


Figure 6.3: *The 8 lens functions and* $r_{general}$

Table 6.2: *Direct comparison of diagram F13 and F14 measurements*

Angle θ [°]	F13$_{measured}$ [mm]	F14$_{approximated}$ [mm]	$\Delta error_{\mu m}$ [μm]	$\Delta error_{pixels}$ [pixels]
0.0000	0.0000	0.0000	0.0	0.0000
11.7180	0.2500	0.2532	−3.2	−0.6882
23.4359	0.5000	0.5038	−3.8	−0.8172
35.2532	0.7500	0.7503	−0.3	−0.0645
48.2622	1.0000	1.0090	−9.0	−1.9355
61.5690	1.2500	1.2525	−2.5	−0.5376
69.2155	1.3750	1.3792	−4.2	−0.9032
77.4578	1.5000	1.5022	−2.2	−0.4731
82.2244	1.5625	1.5658	−3.3	−0.7097
87.0904	1.6250	1.6244	0.6	0.1290
93.3466	1.6875	1.6894	−1.9	−0.4086

$$r_{FUJINON_LF1}(\theta) = \\ -9.1791 \cdot 10^{-10} \cdot \theta^3 + 2.1463 \cdot 10^{-8} \cdot \theta^2 + 2.7567 \cdot 10^{-5} \cdot \theta + 1.8291 \cdot 10^{-6} \quad (6.5)$$

The first finding was that the lens distortion functions vary over the lens. In figure 6.4, the lens functions LF2–LF8 are compared to LF1, showing that the spread of the functions is >10 pixels. Some of the functions deviate by more than 2 pixels if the lens is used with a FOV of ≥35 °, which was the maximum error of the ORIFL190-3 lens at a FOV of ≥63.5 °. This shows that a single lens function wouldn't have been appropriate for the Fujinon lens. Further, figure 6.4 confirms that the spherical lens model should account for azimuthal errors by using r as a function of θ and φ.

The second finding was that there were issues of the projection of the image onto the CCD sensor. The lens specifications from the manufacturer state that the lens should provide a "true" fisheye image (the entire image projection within the the CCD sensor). This isn't the case because the diameter and the picture on the datasheet are wrong. The diameter is closer to 4 mm (twice the maximum distance r in figure 6.3), instead of 3.45 mm from the datasheet. Even if there was an error in the measured lens functions in figure 6.3, this effect is also seen in any fisheye image acquired by the Fujinon lens (see figure 5.10 page 72), where the fisheye image circle is larger than the CCD sensor plane.

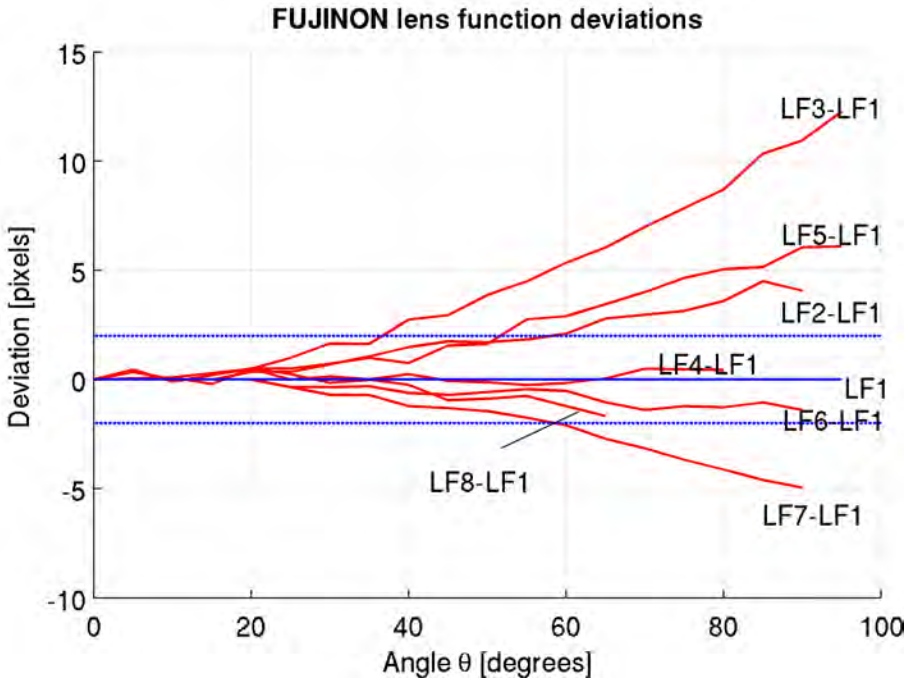


Figure 6.4: *Deviations of the lens functions*

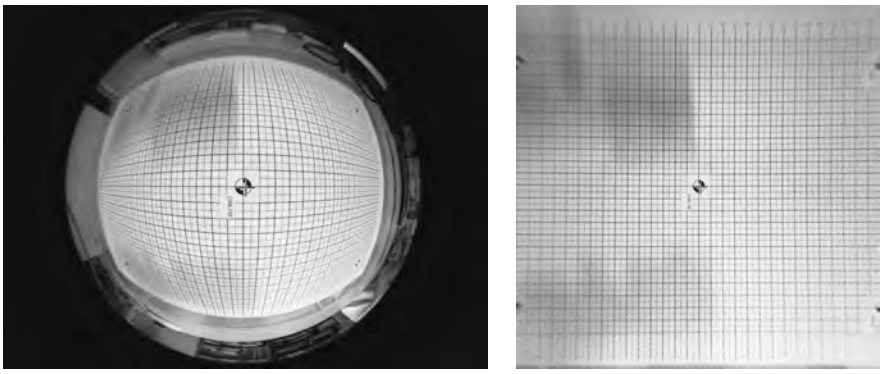
6.2 Image Rectification

Remember that the overall goal of this rectification process is to produce an image that can be treated as an image from a conventional camera (perspective camera). The rectification process extracts a certain part of the fisheye image as illustrated in figure 3.10. The rectification can't remove the entire distortion, as it is limited to the precision of the lens distortion functions. This section provides a detailed analysis of the Omnitech Robotics fisheye lens.

6.2.1 Calibration Setup

The fisheye lens is mounted as a downward looking camera. A sheet of paper with a 5×5 mm minor grid pattern (10×10 mm major grid pattern) is placed on the ground, on top of a glass plate to ensure a flat placing. The camera is placed horizontally using a spirit level. Figure 6.5(a) shows the fisheye image taken for the calibration purpose. This fisheye image is then rectified as shown in figure 6.5(b).

As mentioned previously, the rectification process itself can be used to "register" an image. It allows scaling of the image to a certain FOV, but it can't perform a projective transformation. The camera is placed parallel to the ground plate and therefore doesn't require a projective transformation. The camera is setup is with the



(a) Fisheye image (b) Rectified fisheye image

Figure 6.5: *Calibration images showing the grid pattern*

following parameters:

Resolution rectified image:	1536×1536 pixels
Field of View (FOV):	300 mm (126.9 °)
Camera height z:	102 mm

The rectified image shows an extraction of a FOV of 300 mm (the round marks on the left and right of the image) like the mark in the center of the images. To measure the rectification performance grid points are measured in the rectified image and compared with known positions. The known positions are found by dividing the FOV equally into the grid size.

Rectification Performance

The performance is measured as the difference between the known positions and the grid points in the rectified image. A selection of 165 grid points were measured by hand, selecting a rectified grid point at each intersection of the major grid. Figure 6.8 shows the measured points. A close-up of this figure 6.8 is shown in figure 6.6.

Black crosses ('**x**') mark the measured rectified grid points. Known grid points are marked with a blue plus sign ('+') from where a blue arrow indicates the direction of correction. This indicator can be used to place a misplaced grid point at its true position. The red lines throughout the image indicate the absolute error of the misplacement in millimeters.

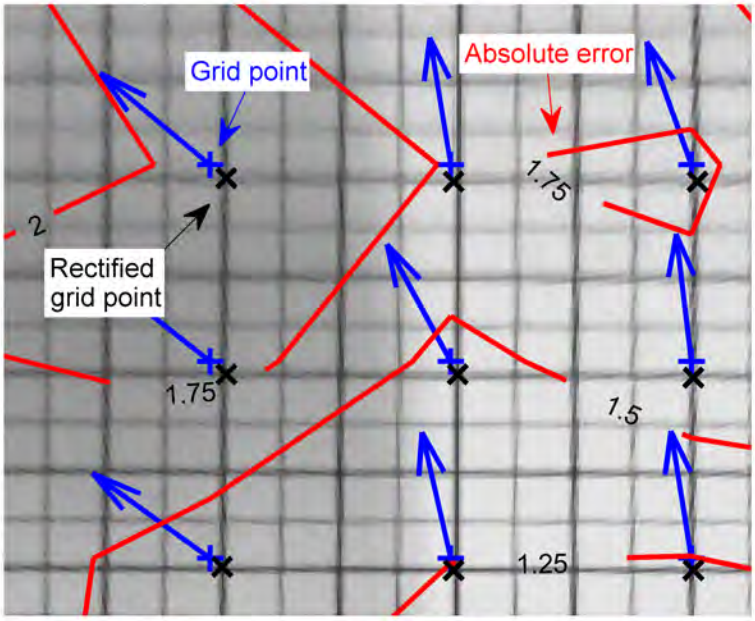


Figure 6.6: *Close-up of figure 6.8*

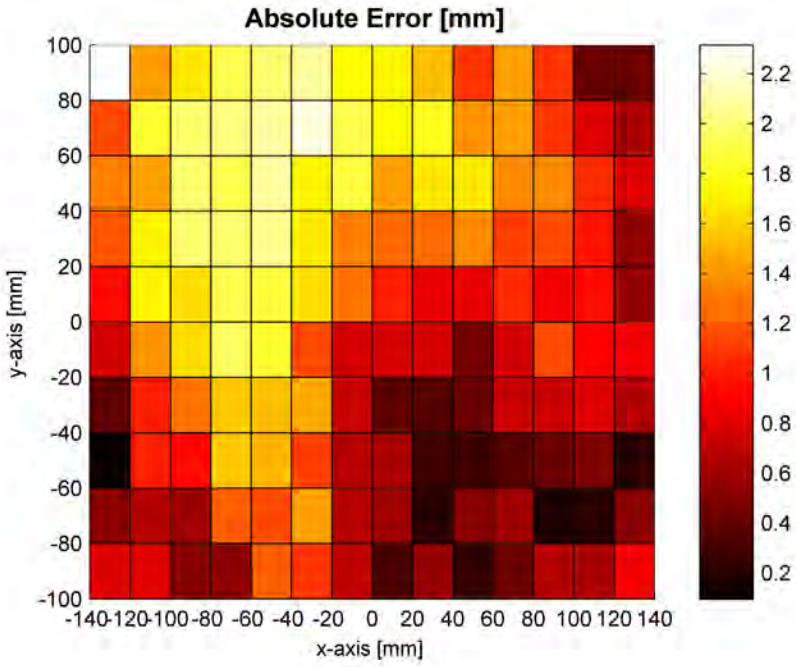


Figure 6.7: *Absolute error distribution*

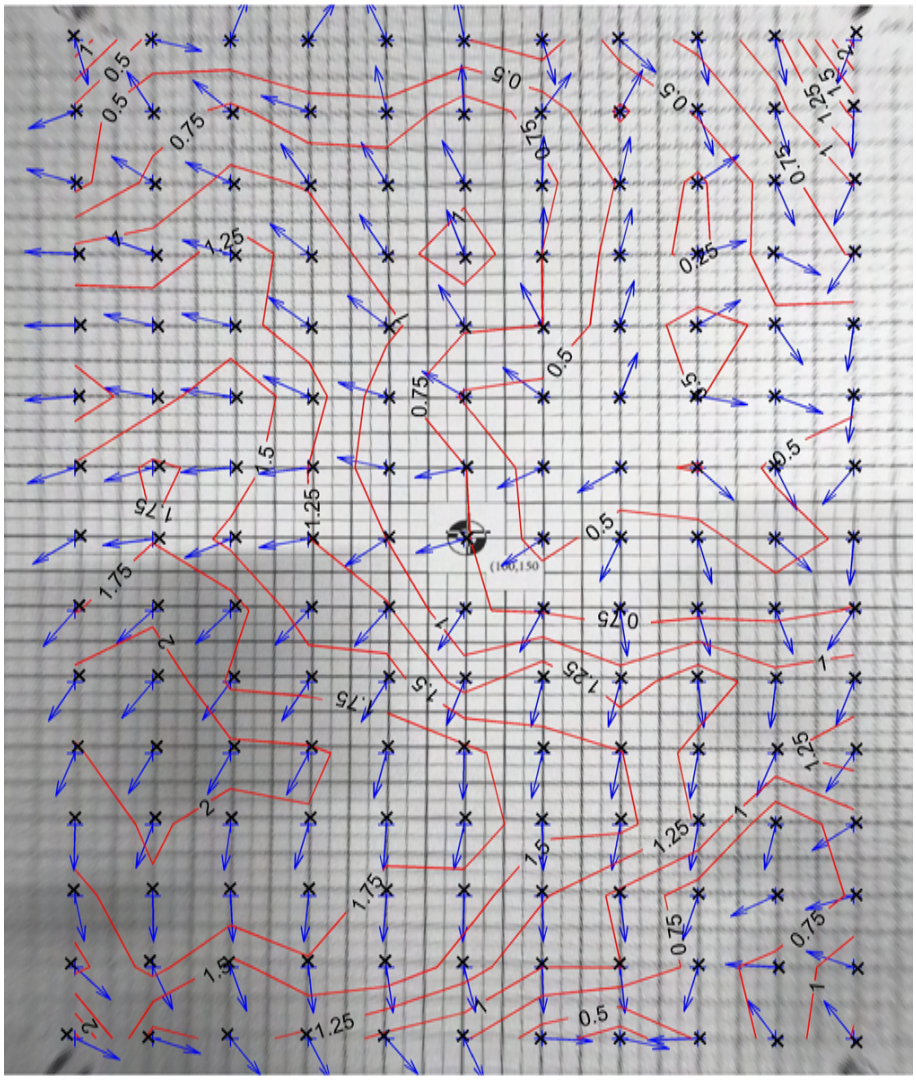


Figure 6.8: *Grid points and rectified grid point overlaid over the rectified image*

The absolute error is the difference between the true position and the rectified position of a grid point, given as a two dimensional error err_{2D} as in equation (6.6). The delta errors dx and dy are the delta error in the x- and y-axis for each grid point (x,y). Figure 6.7 (page 90) shows the distribution of this absolute error err_{2D} for each grid point. The light colors indicate large errors, while dark colors indicate small errors. The maximum absolute error is 2.3 mm or 1.15 %. For example, in the achieved accuracies of approximately 100 m from an image taken at 300 m AGL and a FOV of 60 % this error would contribute an absolute positioning error of 3.5 %.

$$err_{2D} = \sqrt{dx_{x,y}^2 + dy_{x,y}^2} \tag{6.6}$$

The lens distribution of the absolute error direction is shown in figure 6.7. Interesting to see is that the error is nearly zero close to (**A**). The absolute error then grows in a radial manner towards the upper half (**B**) of the image, with the exception of area (**C**). In area (**C**), an irregularity of the lens is seen compared to the lens model. An overlay of vectors and absolute errors in figure 6.10 can be used to refine the image data in the background.

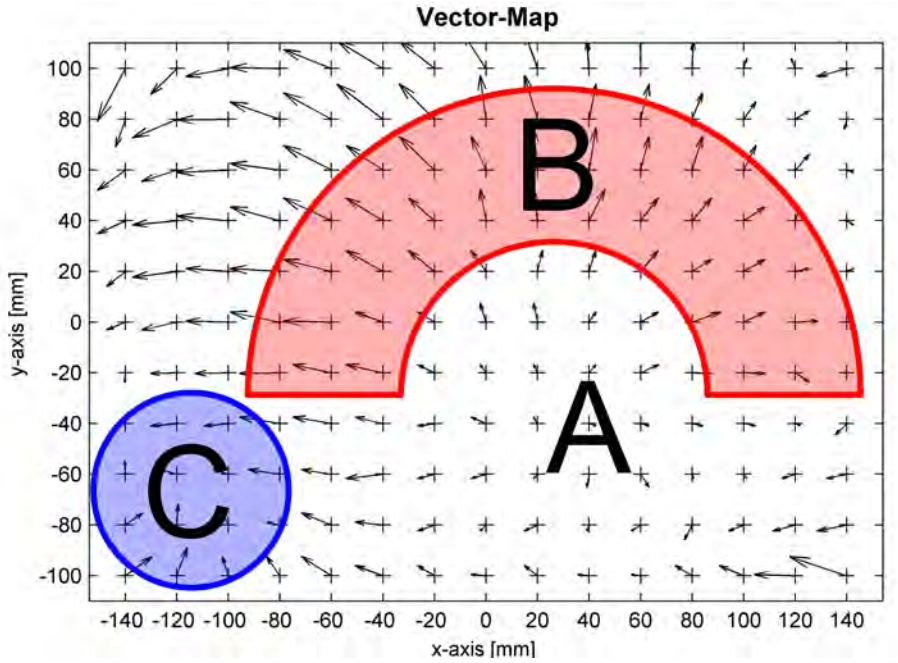


Figure 6.9: *A vector map showing the directions and magnitudes of the errors*

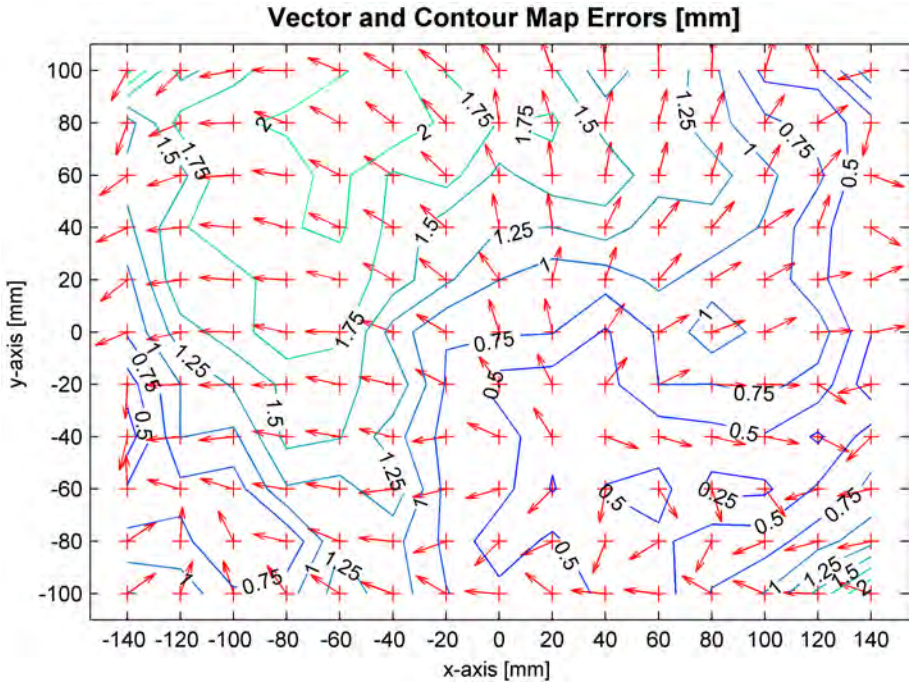
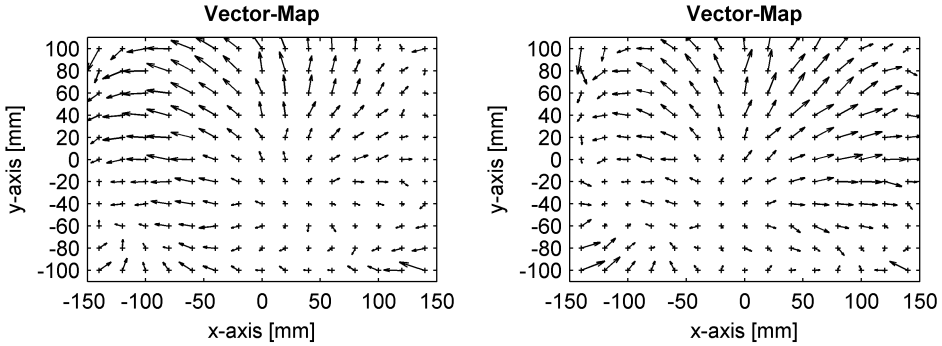


Figure 6.10: *Combination of vectors and absolute errors*

Center Displacement

As mentioned in section 3.1.1, the lens center is another variable that needed to be estimated by measuring the fisheye circle boundary. However, the lens center can also be found using the vector maps, as they indicate areas with only minor remaining distortions. As an example, the estimated lens center is displaced by 1 pixel along the x-axis. Figure 6.11 shows the lens center displacement with and without displacement, indicating that the center is between these two values.



(a) Without displacement (b) With 1 pixel displacement

Figure 6.11: *Lens center displacement shown with the vector map*

6.3 Image Registration and Mapping

Registering single images is the absolute orientation of the image data according to the GPS position and attitude estimation. Where there are image sequences available each registered image can be used to create a mosaic map — a map of a set (sequence) of images. With this information available, the images can be mosaiced in any software capable of using registered images (e.g. Google Earth, FreeGIS, GRASS). For convenience Google Earth was used as a visualization tool, as this software is readily available, easy to use and free.

Using Google Earth

A short investigation was conducted, to determine the precision of the existing Google Earth maps (which make use of WGS84 (World Geodetic System 1984)). Figure 6.12 shows the ground track of the aircraft operating on the ground before take-off and after landing, where the GPS positions from taxiing follow the taxiway lines very accurately. The accuracy was measured to be within ±5 m. To provide an indication of the precision the taxiway tracks after landing were also analyzed with similar results.

(a) Before take-off (b) After landing

Figure 6.12: *Taxiing on ground (white line) overlaid on Google Earth maps*

However, since the GPS is inaccurate over short time periods, this error my be larger than expected. During the setup time the GPS was recording the same position over a period of 2 hours, where it shows a 2Drms error of 4.33 m during this time. The conclusion is that the Google Earth software, for our experiments, has an acceptable level of accuracy of up to <10 m. However, the precision of the GPS and Google Earth couldn't be evaluated as there were no known reference points available.

6.3.1 Mosaic Samples

The image data collected during the flight is processed and registered by the methods described in section 3.3. The following images show results from the data acquisition flight, with image sequences at different altitudes and places. These image samples demonstrate how rectified fisheye images can be used for mapping purposes. Image sequences were taken at various altitudes and processed with different FOVs. Therefore, the GSD changes for each sequence and especially when using larger FOVs the rectified fisheye images show blurr effects towards the edges. Quantitative results are given within this section and are further discussed in chapter 7.

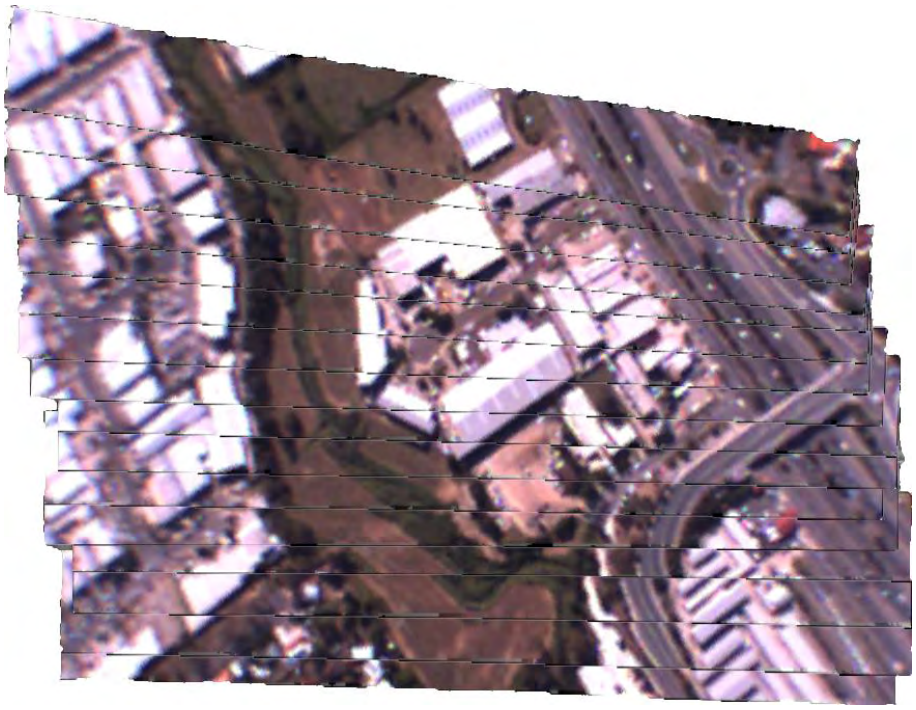


Figure 6.13: *Close-up of a scene (Woodridge, Logan Central)*

The image sequence in figure 6.13 shows a close-up scene from 2000 ft AGL, with images extracted at a FOV of 10×80 ° (or ≈100×1000 m in ground distances). This FOV was chosen because in the image sequences shown in the following images are only showing every 10^{th} to 20^{th} image captured, compared to every 2^{nd} image in this close-up. The fisheye rectification provides approximately 26 pixels per 5 degrees which corresponds to an approximated GSD of <2.4 m. The GSD can be approximated using the lens' non-linearity function (see figure 3.3). Equation (6.7) evaluates the approximated GSD, where Θ is the FOV, ρ the resolution and D the ground distance of the extracted and scaled image:

$$
\begin{aligned}
GSD &= (\Theta \cdot \rho)^{-1} \cdot D \\
&= \left(80° \cdot \frac{26\,\text{pixels}}{5°}\right)^{-1} \cdot 1000\,\text{m} \\
&= 2.4\,\text{m}
\end{aligned}
\tag{6.7}
$$

The GSD is therefore <2.4 m as it increases towards the center of the image (correct is $<2.4\,^{\text{m}}/_{\text{pixel}}$, but the unit "pixel" is irrelevant for the GSD, as it is always based on a unit size). The smallest objects that can be separated from the background are cars on the road or single trees in the field.

The image sequence shows effects of non-alignment of consecutive images, mainly due to errors in the attitude estimation. Further errors could be caused by a wrong altitude estimation by the GPS, the underlying DEM or GPS position errors. The edges of the sequence show that not all errors are attitude estimation errors as some images are smaller than the consecutive one. Further, remember that the rectification performance is 1.1 % or 11.5 m for this image sequence (section 6.2).

The image sequence in figure 6.14 shows a mosaic map taken at 1000 ft AGL over Redland Bay. The white line through the image is the flight path of the aircraft. The images are rectified with a FOV of 80×80 ° (500×500 m) and, due to large FOV, show the aircraft wheel in the images. The GSD of this sequence is ≈1.20 m. Figure 6.15 is a mosaic map with a FOV of 60 ° taken at 1000 ft AGL close to the Tingalpa Reservoir. Both images show divergences of 20–100 m in x- and y-direction between the alignment of the images, depending on the sequence, altitude and axis direction. Further, this image sequence shows a sun-glare effect (see section 5.4.1). White lines appear in images due to sun reflections on the water surface.

Registered images were measured for performance evaluation for relative and absolute errors in flight direction (along-track) and 90 ° to the side (across-track). For this purpose, images were loaded into Google Earth and the measurement tools provided were used. Figure 6.16 shows a sequence of registered images in Google Earth. The red and yellow circle marks an identified ground object that can be used as a measuring feature and the underlying white grid aligns with the flight vector. A close up of this scene is given in figure 6.17 showing the relative across-track error. Another example is given in figure 4.7 showing the large along-track error from the synchronization issue as explained in section 4.3.1.

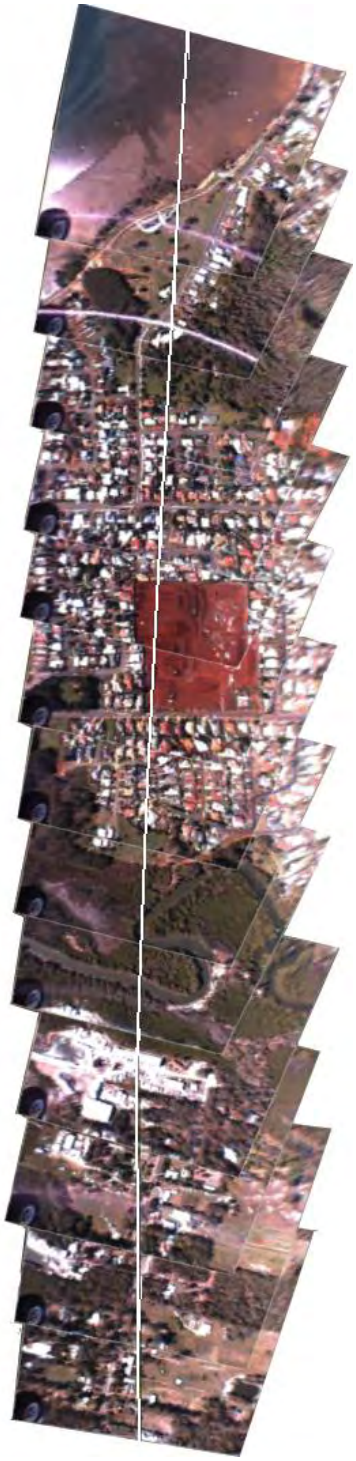


Figure 6.14: *Mosaic map over Redland Bay*

Figure 6.15: *Mosaic map close to Tingalpa Reservoir*

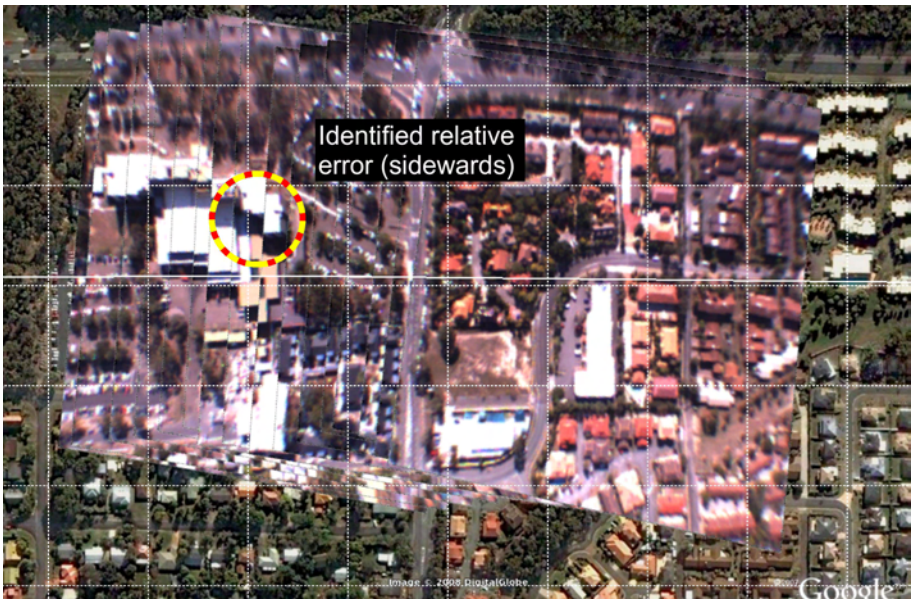


Figure 6.16: *Identifying objects to measure a relative across-track error*

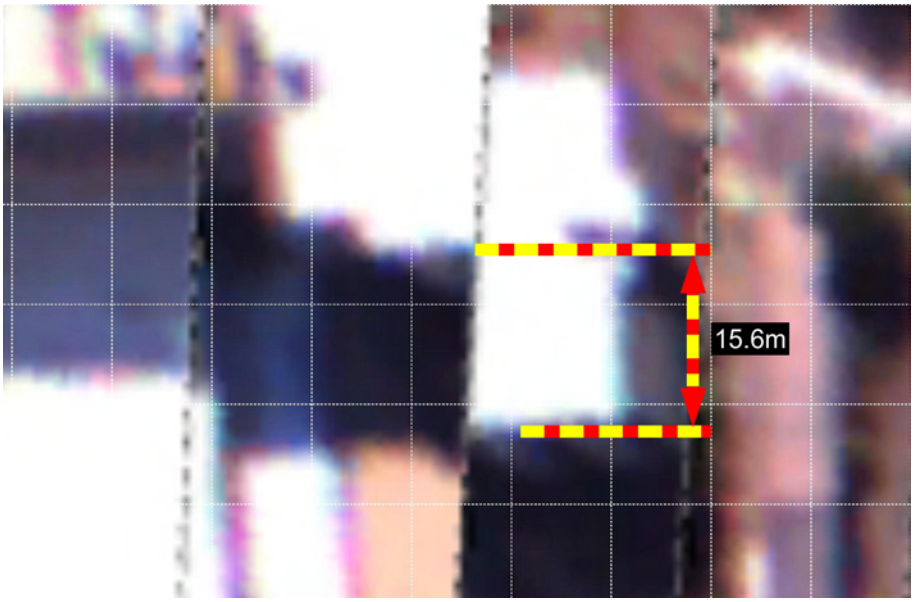


Figure 6.17: *Close-up of figure 6.16: Across-track error of a ground feature*

The image sequences shown in figures 6.18 and 6.19 were both taken when the aircraft was performing turns. Figure 6.18 shows a mosaic map from images taken soon after take-off when the aircraft is performing a turn while climbing. The trapezium shape of the single images is clearly visible and an indication showing the aircraft is in climb and performing a turn. The altitude is approximately 400 ft AGL and the images were extracted at a FOV of 60×60 ° (70×70 m), with a GSD of 0.4 m.

Figure 6.19 shows one of the major benefits of using a fisheye lens as the aircraft performs a 30 ° bank turn. Due to the large FOV (extraction FOV of 90 °) this bank isn't showing significant effects to the overall coverage of the scene in the nadir point of the aircraft (B). Towards the outer side (A) the images loose on quality from two effects, the decreasing spatial resolution and the decreasing angular resolution. Latter is due to the angular dependency of the fisheye lens. The fisheye lens provides an even larger FOV and can "look" inside the turn, although at the costs of spatial resolution. Conventional lenses are limited in the FOV and would require a gimballed camera to "look" inside the turn. The flying altitude of this sequence is 2000 ft AGL and a FOV of 90×90 °.

A more detailed performance discussion is given in the following chapter.

Figure 6.18: *Mapping during climbing and turning after taking-off at Archerfield*

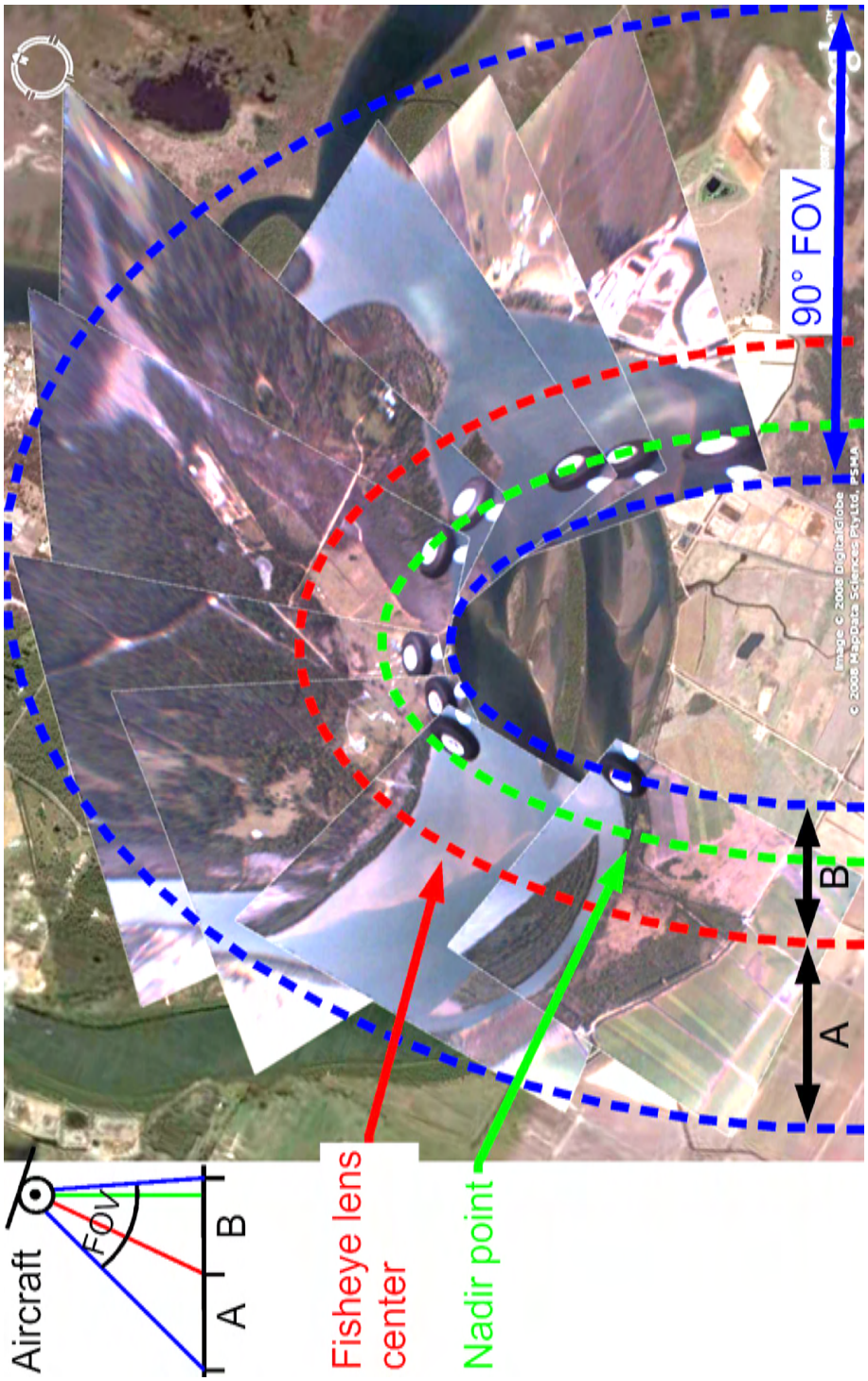


Figure 6.19: *Example image sequence during a turn at 30° bank at Beenleigh*

6.4 Summary

This chapter presented the results of part of the planning of this research. The results from the fisheye lens function derivation were also presented. It was concluded that the lens distortion is a function of the θ and φ (see spherical lens model in figure 2.2), as the rectification performance showed effects of varying lens function over the azimuthal axis. Therefore, multiple lens functions were derived for the second lens. The lens functions were used for the image extraction and rectification showing a performance of 1.15 %. The results of the unforeseen vibration compensation were presented previously in chapter 5. The lens distortion function is positioned according to this vibration compensation in the rectification process. The rectified fisheye images were registered and overlaid over Google Earth maps. The observation of such image sequences shows that consecutive images have alignment errors. Although these errors can be expected by only using a low-cost inertial sensor, further performance analysis is given in the following discussion in chapter 7. The conclusion is that the images are registered in a way from where image stitching processes can be applied to further improve the alignment of the images, although this doesn't necessarily improve the accuracy of the maps.

Chapter 7

Discussion

Section 6 presented results achieved with the images collected during the airborne data acquisition and analyzed their quality in a quantitative way. In this discussion, the research questions listed in the introduction in section 1 are now presented and a critical discussion of the outcomes of this research is undertaken.

7.1 Answering the Research Questions

Question 1 is answered last, as it is seen as a summary of questions 2–5.

7.1.1 Question 2

To what accuracy can a fisheye lens image be rectified to represent an image taken by a conventional perspective camera?

A fisheye lens images can be rectified to an image taken from a conventional lens, and pixel positions in the fisheye lens image can be allocated to real world coordinates. However, the performance depends on certain factors. With the presented spherical lens model in section 2.3, the rectification is limited to $<180\,°$. In practice, a rectification FOV will always be smaller, otherwise the distances in the rectified image will approach infinity. Further, the rectification accuracy is a function of the FOV, with a decreasing accuracy with larger FOVs, due to a less accurate lens function towards large $\theta's$, as shown in figure 6.9. The measured accuracy at a FOV of $126.9\,°$ is $1.1\,\%$ (see section 6.2), but shows non-linear distributions of errors over the lens. This is mainly due to the lens function being a function of θ and φ, as it varies along the azimuth. A single lens distortion function had to be used over the entire azimuth, in the case of the Omnitech Robotics ORIFL190-3 lens. More lens functions, as used for the Fujinon lens, should improve the lens distortion model, but will also involve more work and effort. Using a rectification error-vector map (see figure 6.10) can further improve

the accuracies in a rectified image, but will require another iteration of refining pixel coordinates.

In conclusion, a lens function $r(\theta, \varphi)$ models the lens distortion more accurately. However, if an error in the range of 1 % is acceptable or if the error-vector map is used to refine coordinates, the presented rectification using a single lens function shows a simple way to rectify fisheye images.

7.1.2 Question 3

What accuracy does an airborne fisheye lens image provide for registration to the ground using typical small UAV sensors?

Mosaicing examples were previously presented in section 6.3.1 showing examples in different flight conditions. Further, figures 4.7, 6.16 and 6.17 demonstrated how registration errors were measured in Google Earth using the implemented measuring tools. After solving the synchronization issue in section 4.3.1, another 62 images were analyzed in detail for relative and absolute errors (listed in section A.2.2). The statistical diagrams are shown in figure 7.1 showing mean values close to zero and errors of 6.1–16.7 m (1-sigma) for 1000 ft AGL. The absolute along-track error shows the largest mean due to the forward movement of the aircraft. Even though the absolute along-track diagram shows the least correlation to the normal distribution, a concentration of errors between 0 to 20 m is visible.

Probable sources of these errors include the problem of measuring the angle of the camera installation, the inertial sensor calibration, sensor noises and finally in Google Earth itself. The tilt of the camera installation angle was measured as stated in section 4.2 and compared to the inertial sensors positioning. It can be assumed that a measurement error of 5 ° is possible, since it wasn't possible to determine this tilt to good reference points. An error of 5 ° would result in a 26 m displacement in the images, taken at 1000 ft AGL. However, this error is only partially responsible for the displacement. Other sources of errors are in the attitude estimation, Google Earth errors and an inertial sensor offset.

The inertial sensor offset is especially causing problems. The yaw angle, for example, is corrected by 30 ° counterclockwise. The source of this offset, and the offsets for pitch and roll, originate from the uncalibrated MicroNav inertial sensor. Such a calibration would require the senors to be installed in the aircraft while leveling the aircraft. Measuring the angular differences between the inertial sensor's pitch and roll with the leveled aircraft and the compass with the aircraft alignment would be necessary to conduct further airborne data acquisition. However, since the aircraft was only available for 6 hours during the airborne data acquisition, it wasn't possible

Table 7.1: *Mean and standard deviation of measured errors*

Error	mean [m]	1-sigma [m]
Relative along-track	2.8	6.1
Relative across-track	0.7	8.0
Absolute along-track	10.9	14.8
Absolute across-track	-0.1	16.7

to calibrate the sensors. Further, the noise on the inertial sensor signals are believed to originate from vibrations. However, it is also possible that interferences with the aircraft's avionics can cause noise, but this is less likely due to strict electromagnetic compatibility regulations in avionics. The conclusion is that there are performance increases possible by using calibrated sensors and by understanding the noise sources better. Investigations towards this calibration are planned for the future, as ARCAA plans to install the inertial sensors permanently into their own aircraft. However, the best results would be achieved by mounting the inertial sensor onto the camera, where it would measure inertia at the point the images are taken.

To answer the research question, the absolute orientation accuracies achieved using uncalibrated inertial sensors are approximately 17 m (1-sigma) at a flying altitude of 300 m AGL. This error correlates with the attitude estimation error of 3 ° for roll and pitch, which is 15 m ground displacement at this altitude. The relative errors between the images are smaller with relative accuracies of approximately 8 m (1-sigma) at a flying altitude of 300 m AGL The relative orientation shows a performance from which further alignment of the images should be possible using the techniques described in the literature review [74, 75, 86] (see section 2.4).

The conclusion is that small UAVs are facing attitude estimation problems using only low-cost inertial sensors and GPS. Further, remember that a small aircraft is more susceptible to turbulence (see introduction section 1), which will further reduce the accuracies achieved in the Cessna 172 unless significant improvements in low-cost inertial sensors are achieved. Coupling multiple sensors, such as the GPS and the inertial sensor in case of this research have already improved the sensor signals to 3.2–5.8 ° (1-sigma) (see attitude estimation in section 3.3.1). Otherwise, the inertial sensor measurements would be useless.

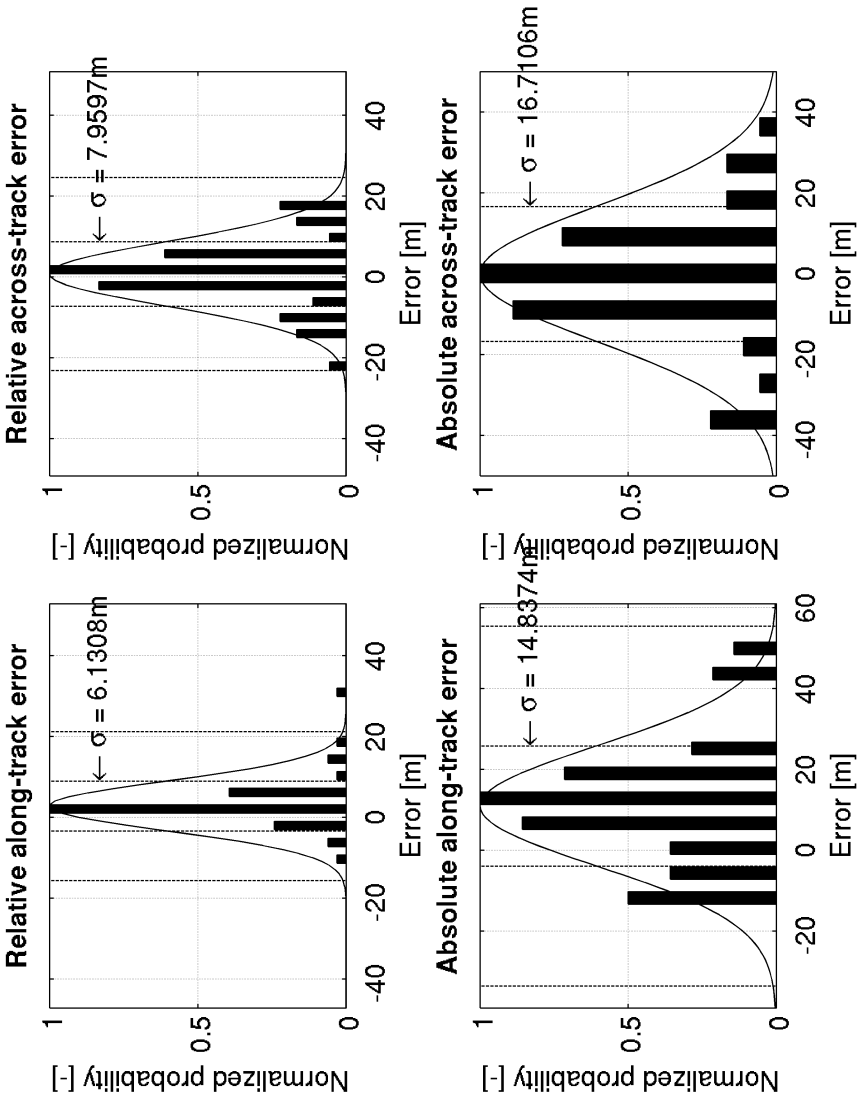


Figure 7.1: Relative and absolute error diagrams

7.1.3 Question 4

What limitations are fisheye lenses exposed to and whether conventional lenses can outperform a fisheye lens?

Compared to conventional lenses, the fisheye image rectification and the angular resolution are limited in fisheye lenses. The rectification performance is limited by the available lens distortion model (lens functions), which can be measured in an arbitrary accuracy depending on the available measurement equipment and effort (for more details, see question 2, section 7.1.1).

The rectification showed that good quality images can only be achieved by using FOV extraction angles $<130\,°$. Larger angles lead to blurry images, as less pixels per degree are available but need to cover larger areas. An example of a rectified fisheye image with a FOV of $160\,°$ is given in figure 7.2, where only the center of the image seems to be usable. The black hyperbolic curve-like shapes show that the fisheye image doesn't provide any information at these angles. There is no black shape on the left-side of the image, because the 8 lens distortion functions vary over the lens (see section 6.1.2).

The angular resolution of a fisheye lens image is approximately 10 times smaller than in an image acquired by a conventional lens with a FOV of $20\,°$. Further, the spatial resolution of a fisheye image decreases with increasing viewing angles θ, because less pixels per degree are available (shown in figure 3.3).

Another challenge is seen in lighting conditions, with increasing influence in larger FOVs. Lighting influences from the sun and canopy overlay reflections (generally known as BRDF (Bidirectional Reflectance Distribution Function)) are well researched by the satellite community [65, 73, 100, 101]. Any airborne collected images will have to correct for this influence, but the large FOV of the fisheye lens may cause a bigger challenge than narrow FOV cameras.

The conclusion is that a conventional lens will always outperform a fisheye lens in respect to spatial resolution. Further, a fisheye lens would always require an initial rectification before being used in mapping applications, requiring more computing effort. However, a conventional lens also contains lens distortions similar to a rectified fisheye image. Refining image coordinates would therefore require further investigation into remaining distortions for both lenses. Further, achieving the same FOV with a conventional lens would either require mounting the camera onto a gimbal system or installing a number of cameras to cover larger FOVs, resulting in increased costs, payload and space. Further, an image from a conventional lens may not have to be corrected for different lighting conditions, whereas the fisheye lens images show larger

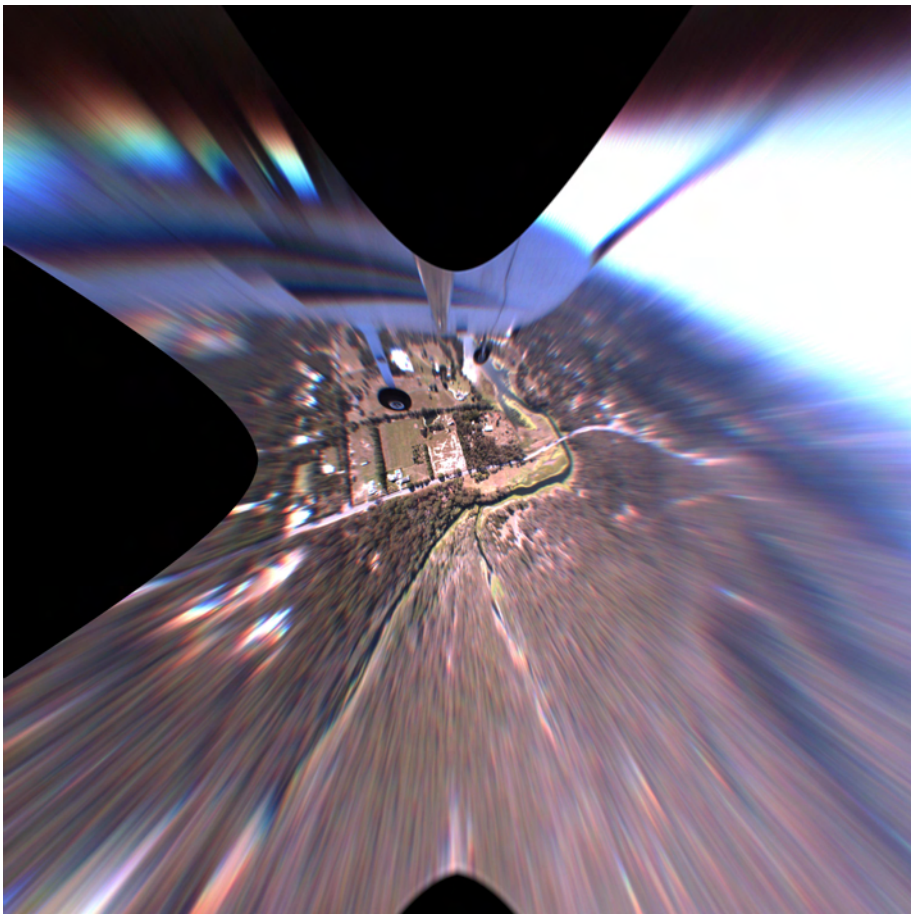


Figure 7.2: *Rectified image with a FOV of 160°*

effects towards lighting.

7.1.4 Question 5

Is there a justifiable trade-off between FOV and spatial resolution onboard small UAVs for aerial photography purposes?

Figure 7.2 shows an extracted rectified fisheye image at a FOV of 160 °. Although the center of the image is shown at a high resolution, the spatial resolution decreases towards the outer parts of the image, appearing blurry. Whether this variable spatial resolution in fisheye images is a problem or a benefit, is a philosophical question.

It is a benefit in the case that an object on the ground (e. g. a house) can be seen from different angles by the same camera-lens system. An approaching house could be seen from the front first, then from the top, while the aircraft is flying over the house, and finally from the back, allowing registration of image information from different angles, while images provided from conventional narrow FOV cameras can not. On the other hand, a gimballed narrow FOV camera can provide information, of the house for example, at higher resolutions and without distortion in the images. However, this comes at the cost of keeping the FOV limited to the house while tracking it, whereas the fisheye lens would provide the 180×360 ° FOV at any time.

In a fisheye image the house is seen at the highest resolution as it moves through the center of the image. To overcome the spatial resolution problem the idea is to use small UAVs, as they can fly close to the ground (and are encouraged by to do so, as it was stated in the literature review in section 2.6). As the angular resolution of a fisheye lens is approximately 10 times smaller compared to a conventional lens with a FOV of 20 °, a UAV could fly 10 times closer to the ground instead and still benefit from the large FOV provided from the fisheye lens.

Further, there is no need to actually rectify a fisheye image, as target tracking can use a fisheye image. The fisheye lens image from a downward looking camera sees the entire horizon including parts of the sky. A low flying UAV is exposed to ground obstacles which can be detected in the fisheye image while the same image is being used in a mosaic map.

Another benefit is seen in SAR (Search and Rescue) applications. A narrow FOV camera may provide a higher resolution image and identify a target from larger distances. On the other hand, to find a target, large areas must be searched and scanned within limited time. This limits the benefits of a narrow FOV camera, but makes a fisheye lens more attractive. The fisheye lens provides a broad overview of a scene, while specific targets can be observed by either flying over them or flying to lower altitudes and using the high resolution parts of the fisheye lens. A combination of

a fisheye lens and a conventional lens camera is also conceivable where the fisheye lens camera would be used for large area scanning and the conventional narrow FOV camera observing specific targets at higher resolution found in the fisheye images.

Further, as stated in the introduction, since small UAVs are more susceptible to turbulence, maneuvers of the aircraft would be compensated using extracted images from different parts of the fisheye image. A higher resolution camera would also decrease the blur-effects shown in figure 7.2 and provide more useful image information at larger extraction angles θ.

Concluding from this, the fisheye lens shows a justifiable trade-off depending on the application. Nevertheless, small UAVs are limited with costs, payload and space, making the use of a fisheye lens more attractive, especially since small UAVs are likely to fly at low altitudes and exposed to obstacles. Although a conventional lens in the same quality class costs approximately the same as a fisheye lens, the conventional lens isn't able to provide the benefits of a fisheye lens. Conventional lenses are physically limited to a FOV of approximately $<120\,^\circ$ and can therefore not compete with the larger FOV of fisheye lenses.

Finally, question 1 will be answered, as it can be considered a summary of the research questions.

7.1.5 Question 1 — Summary

What are the quantitative benefits of using a fisheye lens in small UAV applications?

The quantitative benefits of fisheye lenses in small UAV applications include:

- The fisheye lens can provide images at the same spatial resolutions as conventional lenses by adjusting the flying height of the small UAV. Benefits are savings in cost, payload and space, which will be limited in small UAV applications.
- Registering images using low-cost GPS and inertial sensors will benefit users in absolute accuracies with low-flying UAVs. However, the relative accuracy will remain the same, or even increase, due to the higher susceptibility of small aircrafts to turbulence. Turbulence is known as increasing with decreasing altitude [67].
- A fisheye lens provides similar benefits to a camera mounted on a gimbal system, without the need of additional complexity and payload of the gimbal system.
- Fisheye lenses are less expensive than their conventional lens counterparts. Conventional lenses with large FOV become expensive because it is a physical chal-

lenge to correct for distortions in lenses with large FOV. Ultra-wide-angle lenses with a FOV of 110 ° are 4–10 times more expensive than fisheye lenses with a 180 ° FOV.

This listing outlines that small UAVs can benefit from fisheye lenses, if the limits and drawbacks compared to conventional lenses are considered. Further, as computing power becomes less of an issue today fisheye image rectification isn't an issue and can provide images at accuracies achieved by conventional lenses. The biggest benefit, however, is seen in multi-purpose applications where the fisheye lens can replace multiple conventional lenses and gimbal systems.

7.2 Research Significance

In the literature review in section 2, the mosaicing presented by Lin and Medioni [74], where they use underlaying maps to enhance the registration of aerial photos collected by UAVs, is mentioned. Even though this paper doesn't mention the used UAV system, sensors used or attitude estimates, they mention the ability to place images to an accuracy of 1.35 to 3.17 pixels. Therefore, it is difficult to compare these registration accuracies to the ones achieved during this research, which would be approximately 5 times better than the accuracy achieved by only using the attitude estimates in this thesis (see questions 2, section 7.1.2).

The mapping process presented in this thesis is far from being finished as it was designed for a case study. Future work to improve the maps just using the attitude estimates can still be seen in placing the images according to image features to each other, after the images have been roughly placed by the registration process. Lin and Medioni's work is seen as one step ahead of this research and uses secondary features (image features) to register the images. A significant issue is mentioned by Lin and Medioni, where the feature matching between the image to register and the underlaying image is exposed to different lighting conditions and resolutions. Therefore, poor feature matching can be expected. Although this paper shows accurate results, an adaptation to fisheye lens images is seen as challenging, especially due to the mentioned lighting conditions (BRDF), which change more in an image with a large FOV, than in conventional lenses.

Further, it was found that the low-cost inertial sensor typically used in small UAVs causes the largest uncertainties. The inertial sensor signal has a low signal to noise ratio and must be coupled with a GPS to be used for an accurate attitude estimation. The problem is due to noise caused by vibration, which are of higher frequencies than the sampling rate of the inertial sensor and therefore appears as aliasing.

7.3 Summary

The fisheye lens is a unique way to perform aerial photography and shows benefits in a larger FOV, situational awareness and adjustable extraction FOVs. However, limitations are the angular resolution decrease towards the outer parts of the fisheye image and the lower spatial resolution compared to conventional lenses.

The fisheye images could be rectified to an accuracy of 1.15 % at a FOV of 126 °. These rectified images were registered and presented in Google Earth. The registration errors are 8 m relative errors (1-sigma) and 17 m absolute errors (1-sigma), which was found to correlate with the attitude estimation errors of 3 ° (1-sigma) for roll and pitch.

Improvements are identified, such as to calibrate the inertial sensor within the aircraft or using image processing techniques to further align registered images. These results were achieved using low-cost inertial sensors that can potentially be used in small UAVs. It shows that a fisheye lens can overcome turbulence issues by adjusting the extraction FOV for the registration images.

Chapter 8

Conclusion and Recommendations

The use of UAVs for aerial photography purposes is challenging in many ways. Small UAVs are limited in payload and space and therefore typically use low-quality GPS and inertial sensors. Further, they are more susceptible to turbulence as it was shown in the introduction in (section 1). A fisheye lens was identified to potentially overcome some of these problems, confirmed in this research in the case of limited payload and low-quality inertial sensors. Fisheye lenses for aerial photography applications haven't been researched previously and are a novel approach towards small UAV mapping applications.

Typical low-quality inertial sensors are limited to accuracies an order of magnitude worse than what can be expected from an aerial imaging system. However, UAVs show benefits in being able to acquire airborne images from low-altitudes without exposing personnel to risk. Further, the spatial resolution loss can be overcome by flying at lower altitudes as it was discussed in the literature review in section 7.1.4. This research shows that a fisheye lens provides benefits, such as aerial imaging in turns, without restricting the image quality and area of interest to affect the acquired images, such as from fixed narrow FOV cameras. Further, this research shows an approach of gathering lens specifications of fisheye lenses, rectifying fisheye images, registering images from low-quality sensors and mapping the images as Google Earth overlays.

8.1 Research Outcome

It was found that the distortion caused by the lens is the smallest problem, whereas the attitude estimation is seen as the biggest issue. This research identified following outcomes:

- It was demonstrated that fisheye images can be rectified to an accuracy of 1.15 % and is limited by the available measurement equipment. However, a further analysis showed that the rectification errors can be reduced to sub-pixel errors

by using the error maps in section 6.1.1.

- The mapping results presented in Google Earth show registered images at accuracies of 6.1–17 m (1-sigma) at altitudes of 300 m AGL by using the attitude estimation from a low-cost inertial sensor (CrossBow MicroNav) typically found in small UAVs (section 7.1.3).

- It was found that the lower angular resolution of fisheye lenses compared to conventional lenses can be adjusted by flying at lower altitudes. Uncertified small UAVs are currently restricted and encouraged to fly below 400 ft within Australia and therefore ideally suited for low altitude aerial photography.

- A failure in the time synchronization hardware could be overcome by using registered images and estimating the relative and absolute along-track/across-track errors. It was found that the absolute along-track error is shifted by an integer offset of 2 s which could be corrected using a statistical approach presented in section 4.3.1.

- A Hough transform applied to the vibrating fisheye image was able to compensation for this vibration to an accuracy of ± 1 pixel in 75 % of the cases. However, this compensation is influenced by various effects, such as lighting effects and limited circle boundary detection areas, as it was identified in section 5.4.1. Due to a low signal to noise ratio of the Hough transform specific attempts were conducted to improve the detection certainty of the fisheye circle, such as implementing the AND operation presented in section 5.5.3. The detection certainty could be estimated and attempts towards this compensation have been made. However, a main improvement to this vibration compensation is seen in using a elliptical shape instead of a circle to match with the fisheye image boundary.

- The research questions could be answered thoroughly with only few exceptions. Question 3 could be investigated further towards an error analysis at different altitudes. Further, question 5 is a philosophical question and depends on the specific application, if the fisheye lens can provide a justifiable trade-off between spatial resolution and FOV.

- Small UAVs are limited in payload and require novel approaches such as the demonstrated fisheye lens mapping to overcome limitations of gimballed narrow FOV cameras.

8.2 Recommendations and Future Work

This research identified following recommendations and further research investigation:

- The fisheye lens provides a hemispherical view and can therefore track objects on the ground without using a gimballed camera system. Depending on the altitude, 3D reconstruction of ground features is possible [82, 106]. However, there are so far no relevant publications on 3D measurement accuracies from images obtained by fisheye lenses.
- The downward looking fisheye lens shows the horizon as a circular shape. Dusha et al. [36] detects the horizon in conventional lenses and uses it for attitude estimation in combination of a forward and downward looking conventional lens camera. To what accuracy can s fisheye lens be used to detect the horizon and estimate the attitude of the aircraft?
- To further increase the quality of the mosaic maps the attitude estimation can profit from a more precise calibration process, as it was identified in chapter 4.
- The mosaic maps could be improved using image processing techniques as described by [74, 75, 94] to further align registered images.
- The image registration can be further improved by using a better elevation map (such as HydroSHEDS, section 3.3.2), mounting the inertial sensor to the camera and correct for remaining image rectification errors (section 6.1.1).
- The vibration compensation may be improved using an elliptical shape instead of a circle to match with the fisheye circle boundary in the Hough transform.

8.3 Summary

This investigation of using fish-eye lenses for small UAV aerial photography has not been previously researched and is therefore a novel contribution to this exciting new avenue of aerial photography. Fisheye lenses provide benefits towards such low-cost UAVs and are believed to compete with conventional camera-lens systems to reduce weight and used spaced. Fisheye lenses have their limitations and drawbacks but outperform conventional lenses in other areas, such as a larger FOV, adjustable extraction area and a $180 \times 360\,^{\circ}$ FOV at any time. The major benefit is seen in situational awareness, where the same imaging sensor can be used to track objects on the ground while flying over them, without the need for a gimballed camera and provide image information of the horizon and the surrounding area. Following this thesis, further research in

this area is expected to prove proposed benefits in real applications and demonstrate the capabilities of fisheye lenses.

Chapter 9

Acknowledgment

Nach dieser intensiven und lehrreichen Phase dieses Masters Studium ist es an der Zeit, einigen Leuten meinen Dank auszusprechen:

Meinen Eltern, Elsbeth und Kurt Gurtner, für ihre Unterstützung während den zwei Jahren in Australien. Oma und Chef, für ihre regelmässigen Briefe, Karten und Päckli.

Christine (merci für d'Tastatur!), Jürg (dä wo dahei zum rächte gluegt het, obwou er nie dahei sig), sowie am Storch (ja, de kenne si mittlerwile ou hie) . . . merci viu mau!

At this stage of my Masters Degree, I would like to thank certain people:

Todd Garrigus, who helped to make it possible for me to study abroad. Thanks for all the reviews of documents he did during my studies.

Duncan Greer, for his collaboration work for the attitude estimation.

Rod Walker, Wageeh Boles, Richard Glassock, Troy Bruggemann, Damien Dusha, Ryan Carnie, Lennon Cork and Reece Clothier (all QUT) for their constructive contribution to this research.

Steven Mills, John Lai, Paul Westall, Xi Liu and Pillar Eng (all QUT) for their help getting the airborne data collection done.

Members of ARCAA, that share a dream together and are working hard to make it happen.

My Aussie family, Elaine, Alan, Amanda, Tim and Emma, for the many invitations, the great time and Aussie traditions they shared with me.

Jazz, for the bestest moments Down Under.

Appendices

A.1 Fujinon Fisheye Lens

A.1.1 Measured Lens Points

Table A.1: *Measured lens function points of the Fujinon fisheye lens*

Angle θ [°]	LF1 [pixels]	LF2 [pixels]	LF3 [pixels]	LF4 [pixels]	LF5 [pixels]	LF6 [pixels]	LF7 [pixels]	LF8 [pixels]
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	30.05	29.70	30.00	30.00	30.40	30.05	30.50	30.50
10	60.10	60.10	60.00	60.00	60.10	60.10	60.00	60.00
15	89.80	89.80	89.50	90.00	89.80	89.80	90.00	90.00
20	119.50	119.15	119.00	119.00	119.50	119.50	120.00	120.00
25	148.50	148.15	147.50	148.50	148.15	148.15	149.00	148.75
30	177.15	176.45	175.50	177.00	176.45	176.80	177.88	177.00
35	205.01	203.95	203.38	205.00	204.30	204.70	206.00	205.00
40	232.25	230.76	229.50	232.00	231.03	231.64	233.00	232.00
45	258.45	256.70	255.50	258.50	257.14	257.75	260.00	257.50
50	283.38	281.66	279.50	283.50	281.93	282.80	285.00	282.50
55	307.25	305.41	302.75	307.50	305.50	306.81	310.00	306.50
60	329.85	327.75	324.50	330.00	327.75	329.33	332.75	328.63
65	351.05	348.25	345.00	351.00	348.34	350.00	354.50	349.38
70	370.50	367.53	363.50	370.00	367.35	369.10	374.50	NaN
75	387.85	384.70	380.00	387.38	384.18	386.63	392.50	NaN
80	403.45	399.85	394.75	403.00	399.33	402.18	408.50	NaN
85	416.85	412.34	406.50	NaN	412.25	415.80	422.00	NaN
90	427.45	423.38	416.50	NaN	422.50	426.05	433.50	NaN
95	435.78	NaN	423.50	NaN	NaN	NaN	441.88	NaN
100	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
105	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
110	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

A.1.2 Lens Functions

$$r_{FUJINON_LF1}(\theta) = -9.1791 \cdot 10^{-10} \cdot \theta^3 + 2.1463 \cdot 10^{-8} \cdot \theta^2 + 2.7567 \cdot 10^{-5} \cdot \theta + 1.8291 \cdot 10^{-6} \quad \text{(A.1)}$$

$$r_{FUJINON_LF2}(\theta) = -7.8155 \cdot 10^{-10} \cdot \theta^3 + 1.9318 \cdot 10^{-9} \cdot \theta^2 + 2.8099 \cdot 10^{-5} \cdot \theta + -2.0761 \cdot 10^{-6} \quad \text{(A.2)}$$

$$r_{FUJINON_LF3}(\theta) = -8.9031 \cdot 10^{-10} \cdot \theta^3 + 1.2023 \cdot 10^{-8} \cdot \theta^2 + 2.7628 \cdot 10^{-5} \cdot \theta + 1.4609 \cdot 10^{-6} \quad \text{(A.3)}$$

$$r_{FUJINON_LF4}(\theta) = -9.5451 \cdot 10^{-10} \cdot \theta^3 + 2.5307 \cdot 10^{-8} \cdot \theta^2 + 2.7462 \cdot 10^{-5} \cdot \theta + 2.1109 \cdot 10^{-6} \quad \text{(A.4)}$$

$$r_{FUJINON_LF5}(\theta) = -9.8556 \cdot 10^{-10} \cdot \theta^3 + 2.7737 \cdot 10^{-8} \cdot \theta^2 + 2.7222 \cdot 10^{-5} \cdot \theta + 5.0286 \cdot 10^{-6} \quad \text{(A.5)}$$

$$r_{FUJINON_LF6}(\theta) = -8.9585 \cdot 10^{-10} \cdot \theta^3 + 1.8404 \cdot 10^{-8} \cdot \theta^2 + 2.7604 \cdot 10^{-5} \cdot \theta + 1.6977 \cdot 10^{-6} \quad \text{(A.6)}$$

$$r_{FUJINON_LF7}(\theta) = -1.0328 \cdot 10^{-9} \cdot \theta^3 + 4.0758 \cdot 10^{-8} \cdot \theta^2 + 2.6995 \cdot 10^{-5} \cdot \theta + 6.3683 \cdot 10^{-6} \quad \text{(A.7)}$$

$$r_{FUJINON_LF8}(\theta) = -1.3107 \cdot 10^{-9} \cdot \theta^3 + 6.0193 \cdot 10^{-8} \cdot \theta^2 + 2.6469 \cdot 10^{-5} \cdot \theta + 9.0517 \cdot 10^{-6} \quad \text{(A.8)}$$

A.2 Measured Registration Errors

A.2.1 Registration Errors for Statistical Approach

Sample	Image number	Relative error		Absolute error	
		along-track [m]	across-track [m]	along-track [m]	across-track [m]
1	114836	5.00	0.00	102.00	-3.00
2	114003	0.90	4.40	104.00	-10.30
3	113726	1.20	6.50	115.00	-8.60
4	114110	2.20	0.00	73.00	-13.70
5	114116	5.00	-23.90	68.00	-9.70
6	113952	8.40	6.00	146.00	18.40
7	114019	4.90	4.90	110.00	-20.50
8	114759	1.00	-8.00	107.00	-4.00
9	113935	8.20	7.60	123.00	-16.60
10	114715	4.00	20.00	106.00	-29.90
11	113718	1.00	-2.60	106.00	3.00
12	114140	6.20	0.00	100.00	3.30
13	113708	-3.00	-14.70	135.00	-6.10
14	114718	1.70	3.00	107.00	-33.50
15	114204	3.80	0.00	116.00	-26.90
16	114017	3.50	-0.50	109.00	-9.35
17	114756	-13.00	19.20	125.00	-7.80
18	114601	6.80	-5.30	94.00	-16.90
19	113905	0.00	-9.20	113.00	9.00
20	114847	2.40	19.00	115.00	-5.00
21	114211	0.00	-3.30	105.00	-4.50
22	114817	13.60	0.00	96.00	-12.00
23	114729	0.00	4.90	100.00	7.00
24	114728	3.60	6.00	106.00	-3.20
25	114125	10.30	-9.70	135.00	-38.10
26	114723	7.40	-10.10	101.00	-7.80
27	113919	9.90	9.70	123.00	-3.80
28	113845	2.40	1.80	119.00	-3.60
29	113805	4.90	-2.80	102.00	-9.10
30	114651	7.60	-3.00	109.00	13.80
31	114644	6.80	-3.00	92.00	-11.50
32	114826	3.00	3.20	93.00	2.80
33	114608	5.30	1.90	108.00	-14.70
34	114619	0.00	-2.40	99.00	-2.00
35	114816	3.00	7.10	83.00	-9.50
36	113837	-1.10	4.20	113.00	-14.00
37	114615	14.90	-9.10	108.00	-9.50

... continued on next page.

Sample	Image number	Relative error		Absolute error	
		along-track [m]	across-track [m]	along-track [m]	across-track [m]
38	113722	4.90	3.00	116.00	-7.80
39	113957	5.80	0.00	103.00	-16.10
40	114738	3.30	5.40	105.00	-10.80
41	113809	-9.80	8.50	100.00	25.00
42	114738	2.70	3.50	104.00	-10.30
43	114209	2.50	-7.30	90.00	6.50
44	114739	4.30	2.10	108.00	-4.00
45	114153	6.60	-5.00	103.00	-19.80
46	113928	1.45	13.00	116.00	5.00
47	113655	7.20	-4.90	102.00	0.00
48	114007	6.70	11.20	134.00	-26.20
49	114742	-9.00	-11.20	108.00	18.60
50	113832	-1.40	5.80	100.00	5.70
51	113722	2.30	0.00	115.00	-9.30
52	113754	3.60	-11.30	108.00	-10.40
53	114737	-7.00	0.00	116.00	-25.00
54	113840	1.00	1.20	119.00	-26.00
55	114206	0.00	0.00	98.00	-15.30
56	114605	2.90	4.50	96.00	-11.30
57	113933	4.50	4.90	122.00	-9.50
58	114205	5.50	4.50	105.00	23.90
59	114803	0.00	8.70	105.00	7.00
60	113703	0.00	-4.30	104.00	6.60
61	114544	3.70	0.00	103.00	23.00
62	113804	3.20	6.60	107.00	-23.50
63	114559	6.10	-4.00	98.00	-8.70
64	113704	-7.50	0.00	103.00	3.60
65	113931	8.00	1.00	128.00	0.00
66	114127	2.80	-10.90	101.00	-26.50
67	114753	-4.00	-5.80	102.00	7.30
68	113712	9.10	-1.50	97.00	-9.20
69	113923	-2.80	8.70	113.00	18.30
70	114057	1.00	8.10	99.00	-21.80
71	114040	4.40	26.70	135.00	-14.80
72	113904	-1.60	-1.20	106.00	13.50
73	113927	-6.00	18.90	114.00	-6.20
74	114154	0.00	0.00	93.00	-11.00
75	114721	8.00	5.80	102.00	3.50
76	113806	-2.90	-2.00	101.00	-11.30
77	114833	1.50	-5.00	105.00	-14.30
78	114835	4.00	8.90	113.00	-17.20
79	114303	9.40	4.50	128.00	-48.00

. . . continued on next page.

Sample	Image number	Relative error		Absolute error	
		along-track [m]	across-track [m]	along-track [m]	across-track [m]
80	113702	3.90	-4.00	112.00	-7.00
81	114013	6.30	-12.90	114.00	-5.60
82	113840	1.80	4.40	111.00	-19.00
83	114200	5.00	-4.90	114.00	3.80
84	121832	-9.70	-10.70	100.00	-56.60
85	114605	0.00	2.90	92.00	-13.80
86	114159	2.80	6.70	92.00	3.40
87	113824	-4.70	-10.80	105.00	-5.30
88	114543	2.30	0.00	96.00	11.90
89	113835	-8.10	-3.30	114.00	0.00
90	113734	6.70	-7.50	115.00	-9.30
91	113656	8.20	3.10	119.00	-16.50
92	113748	0.00	-4.50	110.00	4.40
93	113705	0.00	7.80	114.00	5.90
94	114647	0.00	0.00	112.00	0.00
95	114851	0.00	1.60	107.50	-4.50
96	113901	-11.60	10.70	114.00	-11.00
97	114726	0.00	-2.80	114.00	33.50
98	114638	7.00	19.30	94.00	-33.00
99	114805	-4.00	2.20	103.00	-8.50
100	114836	3.30	2.20	114.00	-8.40
101	114819	10.80	0.00	121.00	3.00
102	113723	0.00	4.10	111.00	-10.50
103	114612	2.40	0.00	94.00	-4.10
104	114648	7.10	-1.00	104.00	2.70
105	113840	-8.60	5.20	123.00	-24.00
106	114812	5.00	4.80	109.00	17.60
107	113812	6.00	-4.40	124.00	2.30
108	114833	2.20	-4.00	110.00	-20.00
109	113735	-1.60	6.00	114.00	-14.30
110	113747	-4.50	3.20	109.00	-8.80
111	114732	3.00	-2.00	99.00	9.20

A.2.2 Registration Errors

Sample	Image number	Relative error		Absolute error	
		along-track [m]	across-track [m]	along-track [m]	across-track [m]
1	113845	-2.20	0.70	4.80	30.20
2	113814	0.30	-1.40	0.50	-4.60

... continued on next page.

Sample	Image number	Relative error		Absolute error	
		along-track [m]	across-track [m]	along-track [m]	across-track [m]
3	114020	-2.30	-24.00	17.50	-14.40
4	113925	2.10	9.10	5.30	2.70
5	113844	15.20	-2.40	4.80	0.00
6	113661	1.00	-1.50	-3.80	-4.20
7	113727	0.00	0.50	-9.00	10.80
8	113903	6.10	-1.20	8.50	-7.00
9	113827	0.10	3.50	27.10	4.00
10	114930	1.90	3.20	16.50	-8.70
11	113716	4.60	-3.00	24.00	-4.80
12	113712	2.40	0.00	4.60	6.60
13	123812	1.10	2.50	14.50	20.20
14	113710	2.00	1.00	-3.00	6.30
15	114019	0.00	-13.30	-2.00	11.40
16	114611	5.60	5.00	-8.90	-32.90
17	121213	-3.00	16.30	-14.80	-7.50
18	115610	0.00	-2.90	22.40	-22.50
19	113842	-4.00	4.80	-12.00	9.00
20	113926	1.90	4.90	-10.00	4.50
21	113813	5.00	5.10	17.00	-32.80
22	113911	-5.40	11.70	12.60	-1.40
23	115610	-0.70	5.60	12.20	9.20
24	113840	0.50	12.20	11.80	-39.80
25	113940	-8.20	-0.70	-4.80	0.00
26	114609	3.10	-9.80	16.70	-10.80
27	115432	-0.10	-8.70	26.40	59.00
28	113921	1.80	-3.50	42.70	-9.60
29	114609	0.00	-9.50	12.20	-40.70
30	113735	1.50	-4.00	13.40	7.50
31	123813	8.20	17.90	20.50	36.00
32	114526	33.00	-6.50	49.00	-30.80
33	113732	3.50	0.30	-7.90	2.20
34	113755	6.70	1.40	15.60	3.40
35	113703	0.50	4.70	12.40	-0.50
36	113723	3.30	1.80	-14.90	4.30
37	114531	7.20	-15.20	11.45	25.00
38	113706	2.10	0.00	2.80	-5.00
39	114933	2.00	4.10	8.40	10.30
40	113909	5.90	1.10	12.30	-6.60
41	113934	0.00	-3.00	21.20	-11.90
42	113657	7.70	3.00	9.30	0.00
43	113821	6.00	0.80	13.50	-0.80
44	113829	0.70	-11.50	12.20	5.80

... continued on next page.

Sample	Image number	Relative error		Absolute error	
		along-track [m]	across-track [m]	along-track [m]	across-track [m]
45	113743	2.30	0.70	17.00	14.00
46	113812	13.50	6.10	9.10	5.40
47	113929	3.20	19.60	15.00	-8.30
48	113711	-2.70	-3.20	3.00	-8.10
49	114610	0.00	14.70	8.00	7.90
50	113749	0.50	-3.80	53.00	7.10
51	113815	1.00	1.30	46.00	-6.30
52	113922	-0.60	5.40	0.30	2.70
53	113847	4.80	-3.60	16.40	-13.40
54	113937	-12.40	16.00	7.50	3.00
55	113841	4.70	3.80	17.10	18.90
56	113723	3.70	-0.70	-13.60	5.00
57	113840	8.30	-7.10	41.00	-2.80
58	113839	3.80	6.32	7.80	23.00
59	113759	1.70	1.20	-3.90	-8.70
60	113851	17.55	-1.00	20.90	-3.40
61	113830	4.80	-12.90	8.20	-1.50
62	113727	0.60	0.00	12.40	-10.00

A.3 Image Rectification Algorithm

Fisheye Image Rectification

% Parameter Defintion
$Virt_Alt$ % A virtual altitude required to not to influnce the...
% image scale in the rectificaton process (is set to 100 m)
FOV % The field of view of the AOI to extract.
Off_X % Center offset positions of lens distortion function...
Off_Y % given by the vibration compensation

% Image Rectification
$G \Leftarrow grid(sizeof(AOI))$ % Create a grid of size of the AOI. The grids ...
% resolution is defined by the the required ...
% extraction resoltion.
$G \Leftarrow shift(Off_X, Off_Y)$ % Shift grid points by the offset.
foreach i % x-axis grid point $G.x$
  foreach j % y-axis grid point $G.y$
    $\varphi \Leftarrow atan2(G(i,j))$ % Angles phi for each grid point G. The ...
    % $atan2()$-function is the 4 quadrant arctangent.

    $\theta \Leftarrow atan(\sqrt{G.x^2 + G.y^2}/Virt_Alt)$
    % Angles theta for each grid point G, where...
    % $Virt_Alt$ is used to calculate the angle, but not ...
    % influencing the image scale.
    $R \Leftarrow r(G, \varphi, \theta)$ % Lens function for each grid point G.

    $xx(i,j) \Leftarrow rr(i,j) \cdot cos(\varphi(i,j))$
    $yy(i,j) \Leftarrow rr(i,j) \cdot sin(\varphi(i,j))$
    % x- and y-coordinate for each grid point G, ...
    % projected over the fisheye image

  end
end
$I_{rect} \Leftarrow interpolation(I_{fisheye}(xx, yy))$
% Interpolate each rectified image pixel I_{rect} of the ...
% fisheye image $I_{fisheye}$ at coordinates $(xx; yy)$.
% Interpolaiton method: Cubic.

A.4 Satellite Systems

A.4.1 Satellite Systems References

Quickbird-2:
http://www.digitalglobe.com/about/quickbird.html
Iknonos-2:
http://www.geoeye.com/products/imagery/ikonos/default.htm
Landsat-7:
http://landsat.gsfc.nasa.gov/about/L7_td.html
Cartosat-2:
http://www.isro.org/pslv-c7/pg7.html
Spot-5:
http://events.eoportal.org/pres_SPOT5.html
Formosat-2:
http://www.spotimage.fr/web/en/979-formosat-2-imagery-features.php
WorldView-1:
http://www.geovar.com/wv-1.htm
WorldView-2:
http://www.geovar.com/wv-2.htm
Kompsat-2:
http://www.spotimage.fr/web/en/1155-kompsat-2-images.php
Eros-B:
http://www.eschel.co.il/dui/directory/erosB.htm
Resurs-DK:
http://directory.eoportal.org/pres_ResursDK1ResursHighResolution1.html
GeoEye-1:
http://www.satimagingcorp.com/satellite-sensors/geoeye-1.html
Satellite image costs:
http://www.sopac.org/tiki/tiki-index.php?page=Satellite+Image+Availability

A.4.2 Satellite Systems Table

Table A.4: *List of state-of-the-art satellites for remote sensing and earth observation*

Name	GeoEye-1	Quick-bird-2	Ikonos-2	Land-sat-7	World-View-1	World-View-2	Spot-5	Formo-sat-2	Komp-sat-2	Eros-B	Resurs-DK	Carto-sat-2
Nation	US	US	US	US	US	US	France	Taiwan	Korea	Israel	Russia	India
Launch	(2007)	08/10/01	24/09/99	15/04/99	18/09/07	(2008)	04/05/02	20/05/04	28/07/06	25/04/06	15/06/06	10/01/07
Company	Digital Globe	Digital Globe	GeoEye	USGS (NASA)	Digital Globe	Digital Globe	AMSAT	National Space Organi-zation	Aero-space Research Institute	Israel Aircraft Industries	Sovin-form-sputnik	Space Research Organi-zation
Spatial resolution PAN/MS	0.41m/ 1.65m	0.61m/ 2.44m	0.82m/ 3.2m	15m/ 30m	0.45m/ -	0.48m/ 1.8m (Non-US max. 0.5m)	5m/ 10m (or 2.5m dual im-ages)	2m/ 8m	1m/ 4m	1m/ - (with over-sampling)	1m/ 2.5m (0.5m at lower or-bit)	1m/ -
Horizontal accuracy	(est. 15m)	23m	n/a	30m	7.6m	10.5m	50m	-	-	16.5m	100m	(220m)
Swath with	15.2km	16.5km	11km	183km	16km	16.4km	60km	24km	15km	7km	28.3km	9.6km
Revisit time	<3 days	3–7 days	3–5 days	16 days	1.7–5.9 days	1.1–3.7 days	2–3days	(daily)	3 days	3 days	5–7 days	4 days
Quantiza-tion bits	11	11	11	8	11	11	8	12	10	10	10	10
Image bands	R G B PAN IR-A	R G B PAN IR-A	R G B PAN IR-A	R G B PAN IR-A 2×IR-B IR-C	PAN	R G B PAN IR-A 3 special colors	R G B PAN IR-B	R G B PAN IR-A	R G B PAN IR-A	1 Color band IR	R G PAN IR-A (no blue-band)	PAN
Pricing (USD)	n/a	22.00/km²	13.50/km²	Free	n/a	n/a	1.88/km²	n/a	n/a	14.25/km²	13.51/km²	2.75/km²

A.5 Aerial Photography Systems

A.5.1 Aerial Photography References

Canada scanned photos:
http://airphotos.nrcan.gc.ca/
Leica Geosystems:
http://www.leica-geosystems.com/corporate/en/ndef/lgs_57627.htm
Jena-Optronik:
http://www.jena-optronik.com/cps/rde/xchg/SID-26EE34DB-FB681631/optronik/hs.xsl/3919.htm
Intergraph:
http://www.intergraph.com/dmc/Key_Components.asp
Vexcel/Microsoft:
http://www.microsoft.com/ultracam/downloads/default.mspx
Furhter:
Franz Leberl [39]

A.5.2 Aerial Photography Table

Table A.5: *List of state-of-the-art aerial photography systems and services*

Name	**ADS40 (SH52)**	**JAS 150s**	**UltraCamD** and **UltraCamX**	**Z/I Imaging DMC**	**RC30**	**Scanned film photos**	**getmapping.com**
Company	Leica Geosystems	Jena-Optronik	Microsoft Vexcel	Intergraph	Leica Geosystems	National Resources Canada	Getmapping Inc
Number of imaging systems	1	1	2	1	n/a	n/a	n/a
Resolution	$GSD \approx \frac{H}{10000}$ [Hull2007]	$GSD \approx \frac{H}{20000}$	$GSD \approx \frac{H}{14000}$	n/a	$<10\,\mu m$ (on 230 mm film)	$42\,\mu m$ (on 230 mm film)	$GSD \approx 25$ cm
Accuracy	0.25 m	± 1 pixels	± 0.28 pixels	n/a	n/a	n/a	± 2 m
Swath (width or angle)	64 °	14.9 °	55x37 °	n/a	80 °	n/a	1x1 km
Sensor Type	digital line-scanner stereo	digital line scanner	136 MP (14450x 9420 pixels)	n/a	analog film	analog film scanned	JPEG Image
Quantization bits	12	12	12	n/a	n/a	grayscale 8 / color 24	n/a
Image bands	RGB PAN IR-A	RGB PAN IR-A	RGB PAN IR-A	n/a	color film	color film	color image
Pricing	n/a	n/a	n/a	n/a	n/a	USD 41.25 / image (600 dpi, 250 mm film)	USD 74 / km^2
Remarks					Discontinued as of end of 2007	Scanned photo service	Online image data

A.6 Specification Sheets

Omnitech Robotics ORIFL190-3:
http://www.omnitech.com/pdf/ORIFL190_3%20rev%208.pdf

Fujinon YV2.2X1.4A-2:
http://www.rmassa.com/specsheets/YV22x14A-21.pdf

This appendix is not available online.
Please consult the hardcopy thesis
available from the QUT Library

Bibliography

[1] M. Aggarwal and N. Ahuja. Camera center estimation. In N. Ahuja, editor, *Proc. 15th International Conference on Pattern Recognition*, volume 1, pages 876–880 vol.1, 2000.

[2] C. Australia. Caution: Wake turbulence. Flight safety magazine (p. 34–37), Flying Operations CASA, 2000.

[3] C. Australia. CASR part 101: Unmanned aircraft and rocket operations. Regulations CASR Part 101, CASA, 2002.

[4] H. Bakstein and T. Pajdla. Panoramic mosaicing with a 180/spl deg/ field of view lens. *Omnidirectional Vision, 2002. Proceedings. Third Workshop on*, pages 60–67, 2002.

[5] E. Baltsavias. Image quality, radiometric analysis, preprocessing. In *ISPRS Technical Commission IV Symposium "Geospatial Databases for Sustainable Development"*, 2006.

[6] E. Baltsavias. Introduction (definition of HR, current HR sensors, main characteristics, technological alternatives). In *ISPRS Technical Commission IV Symposium "Geospatial Databases for Sustainable Development"*, 2006.

[7] A. Basu and S. Licardie. Alternative models for fish-eye lenses. *Pattern Recogn. Lett.*, 16(4):433–441, 1995. ISSN 0167-8655.

[8] H. Bendeaa, F. Chiabrandoa, F. G. Tonolob, and D. Marenchinoa. Mapping of archaeological areas using a low-cost UAV. *XXI International CIPA Symposium*, Athens, 2007.

[9] J. Bethel. *Derivation of 3D Rotation Matrix by Sequential Rotations*. Purdue University - School of Civil Engineering, 2005.

[10] J. Beven and H. D. Cobb. Tropical cyclone report hurricane ophelia. Technical report, National Hurricane Center, 2006.

[11] J. Bijker. Development of an attitude heading reference system for an airship. Master's thesis, Stellenbosch University, 2008.

[12] G. Bitelli, V. A. Girelli, M. A. Tini, and L. Vittuari. Low-height aerial imagery and digital photogrammetrical processing for archeological mapping. *International Society for Photogrammetry and Remote Sensing*, Istanbul, 2004.

[13] C. Brauer-Burchardt and K. Voss. A new algorithm to correct fish-eye- and strong wide-angle-lens-distortion from single images. *Image Processing, 2001. Proceedings. 2001 International Conference on*, 1:225–228 vol.1, 2001.

[14] W. Brokx. Where is the business? *3rd UAVNET Meeting*, 2002.

[15] T. S. Bruggemann, D. G. Greer, and R. A. Walker. GPS integrity monitoring with an aerodynamic model and low quality INS. *In Proceedings International Global Navigation Satellite Systems Society IGNSS Symposium 2007, Sydney, Australia*, 2007.

[16] M. Bryson and S. Sukkarieh. Active airborne localisation and exploration in unknown environments using inertial SLAM. *Aerospace Conference, 2006 IEEE*, pages 13 pp.–, 4-11 March 2006.

[17] J. F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8:769–798, 1986.

[18] R. Carnie, R. Walker, and P. Corke. Image processing algorithms for UAV sense and avoid. *ICRA Robotics and Automation - IEEE International Conference Proceedings on*, IEEE, 2006.

[19] D. Casbeer, R. Beard, T. McLain, S.-M. Li, and R. Mehra. Forest fire monitoring with multiple small UAVs. *American Control Conference, 2005. Proceedings of the 2005*, 5: 3530–3535, 8-10 June 2005. ISSN 0743-1619.

[20] *Pilot's Operating Handbook - Cessna 172 Skyhawk.* Cessna Aircraft Company, 1977 edition, 1993.

[21] R. Chan. New parallel hough transform for circles. *Computers and Digital Techniques, IEEE Proceedings* -, 138(5):335–344, Sep 1991. ISSN 1350-2387.

[22] L. Chapman, J. Thornes, J.-P. Muller, and S. McMuldroch. Potential applications of thermal fisheye imagery in urban environments. *Geoscience and Remote Sensing Letters, IEEE*, 4(1):56–59, Jan. 2007. ISSN 1545-598X.

[23] C. H. Chen. *Frontiers of remote sensing information processing.* World Scientific, 2003.

[24] T. A. Clarke and J. G. Fryer. The development of camera calibration methods and models. In *Photogrammetric Record*, 1998.

[25] R. A. Clothier and R. A. Walker. Determination and evaluation of UAV safety objectives. *In Proceedings 21st International Unmanned Air Vehicle Systems Conference*, Bristol: 18.1–18.16, 2006.

[26] J. Courbon, Y. Mezouar, L. Eckt, and P. Martinet. A generic fisheye camera model for robotic applications. *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1683–1688, Oct. 29 2007-Nov. 2 2007.

[27] T. H. Cox, I. Somers, and D. J. Fratello. Earth observations and the role of UAVs. *Civil UAV Assessment Team*, 1.1:49, 2006.

[28] CRC. Project 6.7 — the ergon project. Technical report, CRC for Spatial Information, 2007.

[29] CSIRO. Training course on the use of remote sensing in hydrological and agrometeorological applications. In D. de Vries, editor, *Proceedings of the fifth UN/FAO/WMO/ESA*, page 236. CSIRO Office of Space Science and Applications, 1991.

[30] F. Cuccoli and L. Facheris. Normalized differential spectral attenuation (NDSA): a novel approach to estimate atmospheric water vapor along a LEO-LEO satellite link in the ku/k bands. *Geoscience and Remote Sensing, IEEE Transactions on*, 44(6):1493–1503, June 2006. ISSN 0196-2892.

[31] D. Cunningham. Image motion deblurring. Technical report, School of Informatics, Appleton Tower, Crichton Street, Edinburgh, 2002.

[32] C. Demonceaux, P. Vasseur, and C. Pegard. Omnidirectional vision on UAV for attitude computation. *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2842–2847, May 15-19, 2006. ISSN 1050-4729.

[33] F. Devernay and O. Faugeras. Straight lines have to be straight. *Machine Vision and Applications*, 13:14–24, 2001.

[34] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *Robotics and Automation, IEEE Transactions on*, 17(3):229–241, Jun 2001. ISSN 1042-296X.

[35] R. Duda and P. Hart. Use of the hough transform to detect lines and curves in pictures. *Communication of the ACM*, 15, 1972.

[36] D. Dusha, W. Boles, and R. Walker. Attitude estimation for a fixed-wing aircraft using horizon detection and optical flow. *Digital Image Computing Techniques and Applications, 9th Biennial Conference of the Australian Pattern Recognition Society on*, pages 485–492, 3-5 Dec. 2007.

[37] S. Erturk. Digital image stabilization with sub-image phase correlation based global motion estimation. *Consumer Electronics, IEEE Transactions on*, 49(4):1320–1325, Nov. 2003. ISSN 0098-3063.

[38] F. Esposito, G. Rufino, A. Moccia, P. Donnarumma, M. Esposito, and V. Magliulo. An integrated electro-optical payload system for forest fires monitoring from airborne platform. *Aerospace Conference, 2007 IEEE*, pages 1–13, 3-10 March 2007. ISSN 1095-323X.

[39] M. G. Franz Leberl. Flying the new large format digital aerial camera ultracam. In D. Fritsch, editor, *Photogrammetrische Woche 2003*, 2003.

[40] C.-S. Fuh. *Analytic Photogrammetry - Chapter 14.* National Taiwan University, 2006. p. 20.

[41] A. Gelb. *Applied optimal estimation.* M.I.T. Press, 1974.

[42] R. C. Gonzalez and R. E. Woods. *Digital image processing.* Prentice Hall, 2nd ed. edition, 2002.

[43] *KML 2.1 Reference - Google Earth.* Google, 2.1 edition, 2007.

[44] D. G. Greer, T. S. Bruggemann, and R. A. Walker. Sigma point kalman filters for GPS navigation with integrity in aviation. *In Dempster, Andrew, Eds. Proceedings IGNSS 2007, Sydney, Australia.*, 2007.

[45] A. Gurtner, R. Walker, and W. Boles. A performance evaluation of using fish-eye lenses in low-altitude UAV mapping applications. In *Conference Proceedings.* Australian International Aerospace Congress (AIAC), 2007.

[46] A. Gurtner, R. Walker, and W. Boles. Vibration compensation for fisheye lenses in UAV applications. *In Proceedings Digital Image Computing: Techniques and Applications (DICTA) 2007*, pages pp. 1–8, 2007.

[47] P. Hansen. Fisheye lens image characterization, June 2006. Personal Notes.

[48] R. Harrill and McGeer. Aerosonde robotic airplane completes historic trans-atlantic flight. Technical Report 3221, University of Washington Office of News and Information, 1998.

[49] R. Hartley and S. B. Kang. Parameter-free radial distortion correction with center of distortion estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(8):1309–1321, Aug. 2007. ISSN 0162-8828.

[50] J. Heikkila. Geometric camera calibration using circular control points. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1066–1077, Oct 2000. ISSN 0162-8828.

[51] D. Holland. Use of high resolution satellite imagery for mapping, landscape change detection and map update. In *ISPRS Technical Commission IV Symposium Geospatial Databases for Sustainable Developmen"*, 2006.

[52] Z. Hongying, Y. Lei, G. Pengqi, and L. Shuqiang. The real time image merge method for the remote sensing image acquired from the UAV. *Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International*, 3:1757–1759, 25-29 July 2005.

[53] P. V. C. Hough. Machine analysis of bubble chamber pictures. 1959.

[54] S. Hrabar, G. Sukhatme, P. Corke, K. Usher, and J. Roberts. Combined optic-flow and stereo-based navigation of urban canyons for a UAV. *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3309–3316, 2-6 Aug. 2005.

[55] D. Hull. Leica geosystems ADS40 airborne digital sensor. In *ASPRS Potomac Region Chapter Large Format Digital Camera Symposium*, 2007.

[56] G. il Kweon. New wide-angle lens produces pictures without distortion. *Optical Society of America*, ScienceDaily:1, 2006.

[57] B. Jaehne. *Digital image processing.* Springer, 2002.

[58] C. F. Joanne Poon, Lee Smith. Orthoimage resolution and quality standards. Technical Report Project Number 2.3 Final Report, CRC for Spatial Information, October 2006.

[59] D. I. Jones and G. K. Earp. Requirements for aerial inspection of overhead power lines. *Proceedings of the 12th International Conference on Remotely Piloted Vehicles*, 1996.

[60] J. Kannala and S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *Transactions on Pattern Analysis and Machine Intelligence*, 28(8):1335–1340, 2006. ISSN 0162-8828.

[61] M. Kayton and W. R. Fried. *Avionics Navigation Systems.* 2nd ed. edition, 1997.

[62] D. Kerbyson and T. Atherton. Circle detection using hough transform filters. *Image Processing and its Applications, 1995., Fifth International Conference on*, pages 370–374, 1995.

[63] J.-H. Kim and S. Sukkarieh. Airborne simultaneous localisation and map building. *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, 1:406–411, 14-19 Sept. 2003. ISSN 1050-4729.

[64] K. Koehler. Aerosonde UAV completes first operational flights at NASA wallops. Technical Report 04-03, NASA Wallops Press Releases, 2004.

[65] H. J. Kramer. *Observation of the earth and its environment: survey of missions and sensors*. Springer, 2002.

[66] G. Krishna. Orthoimage generation. In *ISPRS Technical Commission IV Symposium Geospatial Databases for Sustainable Development"*, 2006.

[67] W. Kuehr. *Grundlagen der Flugwetterkunde*, volume Band 2. 2005.

[68] S. Kurnaz and R. B. Rustamov. Earth observation remote sensing and GIS services for monitoring of integration systems. *Recent Advances in Space Technologies, 2007. RAST '07. 3rd International Conference on*, pages 268–270, 14-16 June 2007.

[69] M. F. Land and D.-E. Nilsson. *Animal eyes*. Oxford University Press, 2002.

[70] S. Li. Monitoring around a vehicle by a spherical image sensor. *Intelligent Transportation Systems, IEEE Transactions on*, 7(4):541–550, Dec. 2006. ISSN 1524-9050.

[71] S. Li, M. Nakano, and N. Chiba. Acquisition of spherical image by fish-eye conversion lens. *Virtual Reality, 2004. Proceedings. IEEE*, pages 235–236, 27-31 March 2004. ISSN 1087-8270.

[72] E. B. Li Zhang. Automated and semi-automated object extraction (mainly roads and buildings). In *ISPRS Technical Commission IV Symposium Geospatial Databases for Sustainable Development"*, 2006.

[73] S. Liang. *Quantitative remote sensing of land surfaces*. Wiley-Interscience, 2004.

[74] Y. Lin and G. Medioni. Map-enhanced UAV image sequence registration and synchronization of multiple image sequences. *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–7, 17-22 June 2007.

[75] Y. Lin, Q. Yu, and G. Medioni. Map-enhanced UAV image sequence registration. *Applications of Computer Vision, 2007. WACV '07. IEEE Workshop on*, pages 15–15, Feb. 2007. ISSN 1550-5790.

[76] H. Madjidi, S. Negahdaripour, and E. Bandari. Vision-based positioning and terrain mapping by global alignment for UAVs. *Proceedings. IEEE Conference on Advanced Video and Signal Based Surveillance, 2003.*, pages 305–312, 21-22 July 2003.

[77] T. Maruyama. *Lecture Notes in Computer Science*, volume Volume 3203/2004. Springer Berlin / Heidelberg, 2004.

[78] I. A. McManus. *A multidisciplinary approach to highly autonomous UAV mission planning and piloting for civilian airspace*. Phd thesis, QUT, 2005.

[79] B. Micusik and T. Pajdla. Structure from motion with wide circular field of view cameras. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(7): 1135–1149, July 2006. ISSN 0162-8828.

[80] H. Muammar and M. Nixon. Approaches to extending the hough transform. *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on*, 3: 1556–1559, 1989.

[81] R. C. Nelson. *Flight Stability and Automatic Control*. Boston, Mass.: WCB/McGraw Hill, 1998.

[82] T. Nishimoto and J. Yamaguchi. Three dimensional measurement using fisheye stereo vision. *SICE 2007 Annual Conference*, pages 2008–2012, 17-20 Sept. 2007.

[83] OSD. Unmanned aircraft systems roadmap 2005-2030. *Office of the Secretary Defense*, -(A581734):213, 2005.

[84] P. Peursum, G. West, and S. Venkatesh. Combining image regions and human activity for indirect object recognition in indoor wide-angle views. *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 1:82–89 Vol. 1, 17-21 Oct. 2005. ISSN 1550-5499.

[85] Z. Qin, M. Zhang, T. Christensen, W. Li, and H. Tang. Remote sensing analysis of rice disease stresses for farm pest management using wide-band airborne data. *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03. Proceedings. 2003 IEEE International*, 4:2215–2217, 21-25 July 2003.

[86] M. Ramachandran and R. Chellappa. Stabilization and mosaicing of airborne videos. *Image Processing, 2006 IEEE International Conference on*, pages 345–348, 8-11 Oct. 2006. ISSN 1522-4880.

[87] H. Rhody and C. F. Carlson. Hough circle transform. Lecture, Center for Imaging Science, Rochester Institute of Technology, 2005.

[88] J. C. Russ. *The image processing handbook*. CRC Press, 4th ed. edition, 2002.

[89] J. Ryde and H. Hu. Fast circular landmark detection for cooperative localisation and mapping. *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2745–2750, 2005.

[90] G. Schuster and A. Katsaggelos. Robust circle detection using a weighted MSE estimator. *Image Processing, 2004. ICIP '04. 2004 International Conference on*, 3:2111–2114, 24-27 Oct. 2004. ISSN 1522-4880.

[91] M. Shin, D. Goldgof, K. Bowyer, and S. Nikiforou. Comparison of edge detection algorithms using a structure from motion task. *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, 31(4):589–601, 2001. ISSN 1083-4419.

[92] N. M. Short. The remote sensing tutorial. Technical report, NASA, 2007.

[93] S. Srinivasan and R. Chellappa. Image stabilization and mosaicking using the overlapped basis optical flow field. In *ICIP '97: Proceedings of the 1997 International Conference on Image Processing (ICIP '97) 3-Volume Set-Volume 3*, page 356, Washington, DC, USA, 1997. IEEE Computer Society. ISBN 0-8186-8183-7.

[94] R. Szeliski. Image alignment and stitching: A tutorial. Technical report, Microsoft Research, 2004. pp. 71.

[95] T. Taylor, W. W. Boles, and S. Geva. Map building using cheap digital cameras. *Digital Image Computing Techniques and Applications, 9th Biennial Conference of the Australian Pattern Recognition Society on*, pages 130–137, 3-5 Dec. 2007.

[96] D. H. Titterton and J. L. Weston. *Strapdown inertial navigation technology*. Institution of Electrical Engineers, 2nd ed. edition, 2004.

[97] S. Troeth. Coastwatch trials: Customs coastwatch trials new 'eye in the sky' technology. Technical report, Aerosonde, 2004.

[98] D. Tsoulis. GPS/INS integration for direct georeferencing of airborne imaging sensors. In *International Summer School "Digital Recording and 3D Modeling",*, page 30, 2006.

[99] H. White, L. Sun, C. Champagne, K. Staenz, and S. Leblanc. BRDF normalization of hyperspectral image data. *Geoscience and Remote Sensing Symposium, 2002. IGARSS '02. 2002 IEEE International*, 4:2572–2574 vol.4, 24-28 June 2002.

[100] J. Williams. *GIS processing of geocoded satellite data.* Springer, 2001.

[101] J. Williams. *Geographic information from space: processing and applications of geocoded satellite images.* Wiley, 1995.

[102] P. R. Wolf. *Elements of photogrammetry: with applications in GIS.* McGraw-Hill, 2000.

[103] J. Wu and G. Zhou. High-resolution planimetric mapping from UAV video for quick-response to natural disaster. *Geoscience and Remote Sensing Symposium, 2006. IGARSS 2006. IEEE International Conference on*, pages 3333–3336, July 31 2006-Aug. 4 2006.

[104] W. Yu and Y. Chung. A calibration-free lens distortion correction method for low cost digital imaging. *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, 1:1–813, 14-17 Sept. 2003. ISSN 1522-4880.

[105] S.-C. Zhang and Z.-Q. Liu. A new algorithm for real-time ellipse detection. *Machine Learning and Cybernetics, 2003 International Conference on*, 1:602–607, 2003.

[106] H. Zhao and J. Aggarwal. 3d reconstruction of an urban scene from synthetic fish-eye images. *Image Analysis and Interpretation, 2000. Proceedings. 4th IEEE Southwest Symposium*, pages 219–223, 2000.

[107] J. Zheng, Y. Zhou, and P. Mili. Scanning scene tunnel for city traversing. *Visualization and Computer Graphics, IEEE Transactions on*, 12(2):155–167, March-April 2006. ISSN 1077-2626.

[108] G. Zhou, C. Li, and P. Cheng. Unmanned aerial vehicle (UAV) real-time video registration for forest fire monitoring. *Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International*, 3:1803–1806, 25-29 July 2005.