

**HYPER-SPECTRAL IMAGING FOR AIRBORNE METEORITE
DETECTION**

DAVID MOORHOUSE

Date: 29th of October 2014.

Part i. Abstract

Meteorites are sought after by both scientists and enthusiasts due to their unique characteristics and the window they provide to the broader universe. Current meteorite collection methods are labour and resource intensive and return only relatively few finds in the context of the investment. The basis of this project was to investigate whether meteorites can be identified through a hyper-spectral camera which would be ultimately fitted to an unmanned aerial vehicle (UAV). Such an approach would allow greater geographic coverage of search areas, less human resources and potentially due to these factors, a greater return on investment. While work has been undertaken on identifying the spectral signatures of meteorites and on the use of hyper-spectral imaging in detection and identification, a search of the literature reveals that no earlier work on the use of hyper-spectral imaging for the identification and detection of meteorites. This project therefore builds on the more general work undertaken to apply hyper-spectral imaging to meteorite detection and identification.

A key component of this project was the design and construction of a low cost hyper-spectral camera, which involved the development of two prototypes. Collection of hyper-spectral data, including of meteorites and known and unknown terrestrial rocks, was performed. This was then analysed for the presence of meteorites. The analysis and interpretation of this data required the research and development of a system to analyse the data to determine the presence and location of objects of interest. Ultimately this has produced a system that analyses hyper-spectral data to determine the the presence of particular types of meteorites under full sun lit conditions. The software that produces these results also logs the presence of the meteorites against the frame number and location of the find.

The findings of the project indicate that hyper-spectral imaging is an appropriate way to detect and identify meteorites both at a pure spectral level and practically with imperfect equipment that relies upon reflections of sunlight off the sample materials. The project identifies further work which would allow meteorite detection from an aerial vehicle. While, the software which enables the meteorite detection system to perform hyper-spectral analysis and meteorite detection on board an aerial vehicle has been written, the hardware requires further work. The hardware (that is, the hyper-spectral camera) requires refinement to support its use on an aerial vehicle, including ensuring an appropriate level of robustness to support its use on an aerial vehicle in remote areas.

Part ii. Disclaimer

University of Southern Queensland Faculty of Health, Engineering and Sciences ENG4111/ENG4112 Research Project

Limitations of Use

The Council of the University of Southern Queensland, its Faculty of Health, Engineering & Sciences, and the staff of the University of Southern Queensland, do not accept any responsibility for the truth, accuracy or completeness of material contained within or associated with this dissertation. Persons using all or any part of this material do so at their own risk, and not at the risk of the Council of the University of Southern Queensland, its Faculty of Health, Engineering & Sciences or the staff of the University of Southern Queensland. This dissertation reports an educational exercise and has no purpose or validity beyond this exercise. The sole purpose of the course pair entitled "Research Project" is to contribute to the overall education within the student's chosen degree program. This document, the associated hardware, software, drawings, and other material set out in the associated appendices should not be used for any other purpose: if they are so used, it is entirely at the risk of the user.

Part iii. Candidates Certification

University of Southern Queensland Faculty of Health, Engineering and Sciences ENG4111/ENG4112 Research Project

Certification of Dissertation

I certify that the ideas, designs and experimental work, results, analyses and conclusions set out in this dissertation are entirely my own effort, except where otherwise indicated and acknowledged. I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

David Moorhouse 0061016633

Part iv. Acknowledgements

I would like to thank my colleague Fergus McCracken for providing technical advice on the project as well as access to his rapid prototyping resources. In addition, Ashley Norris and Nicholas West assisted greatly through contributing their knowledge and interest in meteorites and geology. My employer, Seeing Machines Ltd. has also made valuable contributions in the form of loaning parts of the optical hardware that made building the system possible. Dr. Tobias Low, who has provided his supervision of the project has contributed extremely useful feedback as the project has developed. I would also like to extend special thanks to my lovely wife Caitlin Moorhouse who has been extremely supportive of my research and has provided invaluable advice on academic writing.

CONTENTS

Part i. Abstract	2
Part ii. Disclaimer	3
Part iii. Candidates Certification	4
Part iv. Acknowledgements	5
List of Figures	10
List of Tables	12
Nomenclature	13
Part 1. Introduction	14
1. Summary of Achievements	15
2. Objectives	16
2.1. Technical Objectives	16
2.1.1. Technical Tasks Beyond Outside of Scope	16
2.2. Research objectives	16
Part 2. Literature Review	18
Part 3. Project Context	22
3. Meteorites	22
3.1. Reasons for Collection	22
3.2. Cataloguing of Meteorites	22
3.3. Types of Meteorites	22
3.4. Meteorite Weight	24
3.5. Meteorite Collection over Time	25
3.6. Meteorite Collection Programs	26
3.7. Meteorite Collection Locations	28
3.8. Meteorite Spectra	29
4. Hyper-spectral Imaging	32
4.1. Technology	32
5. Unmanned Aerial Vehicles (UAVs)	35
5.1. Definition	35
5.2. Types	35
5.3. History	35
5.4. Modern Utility	35
Part 4. Project Methodology	36
6. Methodology Introduction	36
7. Prevalence of Meteorite Types	38
8. Characteristics of Meteorite Spectra by Type	38
9. Target Meteorite Types	39
10. Weight of Target Meteorite Types	39
11. Volume of Target Meteorite Types	40
12. Area of Target Meteorite Types as Seen From the Air	41

13. Aircraft Altitude & Front-End Optics	42
13.1. Field of View	42
13.2. Focus	43
14. UAV Coverage & Speed	43
15. Hyper-Spectral Camera Development	44
15.1. Introduction	44
15.2. First Prototype Hyper-Spectral Camera	44
15.3. Second Prototype Hyper-Spectral Camera	45
15.4. Ximea API Python Connector	49
15.5. Hyper-spectral Image Capture	49
16. Target Meteorite Reference Spectra	50
16.1. Samples	51
17. Spectral Reference Capture	53
18. Spectral Matching Technique	54
19. Hyper-spectral Pre-processing	57
19.1. Spectral Adjustment	57
19.2. Region of Interest	57
19.3. Image Rotation	58
19.4. Camera Exposure Control	58
19.5. Noise Reduction	59
19.6. Distortion Removal	59
20. Meteorite Presence Detection	59
21. Trial Meteorite Search	60
22. Report Location of Detected Meteorites	60
23. Location Confirmation	61
24. Prototype UAV	61
25. Final UAV	61
26. Final Hyper-Spectral Camera	62
27. Final Meteorite Search	62
Part 5. Resource Requirements	63
28. Hardware	63
28.1. Hyper-spectral Camera	63
28.1.1. Optics	63
28.1.2. Video Camera	63
28.2. GNSS	64
28.3. Unmanned Aerial Vehicle	65
28.3.1. Wing Configuration	65
28.3.2. Flight time	65
28.3.3. Energy Storage	65
28.3.4. Automation	65
28.3.5. On-board devices	65
28.3.6. Payload	66
28.4. Meteorite Samples	66
29. Software	66
29.1. Mathematical Modelling	66
29.2. 3D Modelling	67
29.3. Databases	67

29.4. Version Control	67
Part 6. Safety, The Law & Consequential Effects	69
30. Legislation	69
30.1. UAVs - Australia	69
30.2. Meteorites	69
31. Safety	70
31.1. UAV Construction	70
31.2. Lithium-Polymer Batteries	70
31.3. Flight	70
31.4. Eye Safety	70
31.5. Electricity	71
31.6. Soldering	71
Part 7. Results	72
32. Introduction	72
33. Meteorite Detection - Laboratory Spectroscopy	72
33.1. Context	72
33.2. Distance Results	73
34. Meteorite Detection - Solar Reflectance Spectra	77
34.1. Context	77
34.2. Error in Measured Spectra	77
34.3. Distance Results	80
35. Aerial Advantage	82
36. Summary of Results	82
Part 8. Time-line	83
Part 9. Key Outcomes	83
Part 10. Conclusions	84
Part 11. Further Work	84
References	86
Appendices	91
Appendix A. Project Specification	91
A.1. Introduction	91
A.2. Project Aim	91
A.3. Programme	91
Appendix B. Hyper-spectral Camera Geometry Calculations - Python Code	92
Appendix C. Meteoritical Bulletin Analysis - Python Code	95
Appendix D. Gaffey Spectra Analysis - Python Code	112
Appendix E. Image Sensor Spectral Response - Python Code	118
Appendix F. Front-End Optics - Python Code	122
Appendix G. xipyAPI - Ximea Camera API Python Connector - Python Code	123

Appendix H.	Hyper-Spectral Camera Controls - Python Code	126
Appendix I.	Hyper-Spectral Camera Main - Python Code	130
Appendix J.	Hyper-Spectral Matching - Python Code	133
Appendix K.	USGS & Gaffey Matching - Python Code	142

LIST OF FIGURES

2.1 Airborne Meteorite Detection Overview: UAV image from UAV Robotics Australia[UAV-Robotics-Australia, 2014]. Meteorite Image from Meteorites Australia[Meteorites-Australia, 2014]	16
2.2 ANSMET Team 2013-14 - Systematic Searching	18
2.3 Ordinary Chondrite Spectral Reflectance [Gaffey, 2001]	20
3.1 Meteorite Classification [Wiesberg et al., 2006]	23
3.2 Catalogued Meteorite Types [The-Meteoritical-Society, 2014]	24
3.3 Catalogued Meteorites Ordered by Weight[The-Meteoritical-Society, 2014]	25
3.4 Meteorites Catalogued by Decade of Discovery[The-Meteoritical-Society, 2014]	26
3.5 Programs Responsible for Collection of Catalogued Meteorites [The-Meteoritical-Society, 2014]	28
3.6 Collection Location of Catalogued Meteorites [The-Meteoritical-Society, 2014]	29
3.7 Spectral Reflectance of Samples of Ordinary Chondrites, Normalised at 710nm	30
3.8 Ordinary Chondrite Reference Spectra	30
3.9 Solar Spectral Irradiance	31
3.10 Match to Spectral Reference of Ordinary Chondrite Sample Spectra	32
4.1 Hyper-spectral Camera Dispersion	33
4.2 Hyper-spectral Cube: Meteorite Image of “Northwest Africa 5393”[Meteorites-Australia, 2014]	34
4.3 Line Scan Camera	34
6.1 Overall Methodology Flowchart	37
7.1 Catalogued Meteorites by Gaffey Spectral Group[The-Meteoritical-Society, 2014, Gaffey, 1976]	38
8.1 Catalogued Meteorites by Spectral Characteristics [The-Meteoritical-Society, 2014]	39
10.1 Ordinary Chondrite Weights [The-Meteoritical-Society, 2014]	40
11.1 Ordinary Chondrite Volumes	41
15.1 First Prototype Hyper-spectral Camera	45
15.2 Second Prototype Hyper-spectral Camera with Front-End Optics attached	46
15.3 3D Transparent Camera Assembly	47
15.4 Second Prototype Hyper-spectral Camera without Front-End Optics attached	48
15.5 Image Sensor Spectral Response Comparison	49
15.6 Hyper-spectral image captured from NWA869 L4-6 Chondrite (from solar reflection)	50
16.1 Identified Terrestrial Rock Samples	51
16.2 Unidentified Terrestrial Rock Samples	52

HYPER-SPECTRAL IMAGING FOR AIRBORNE METEORITE DETECTION	11
16.3 Meteorite Samples	53
17.1 Configuration of Data Collection	54
18.1 Distance Plot - Cosine Distance - Sample 15 as Reference	56
19.1 Un-processed Hyper-spectral image captured from NWA869 L4-6 Chondrite (solar reflection)	58
24.1 Prototype UAV	61
25.1 Final UAV: Unassembled (Pen on wing for reference size)	62
28.1 Rolling Shutter (left) compared to Global Shutter (right): Images from Point Grey Research[Point-Grey-Research, 2014]	64
29.1 Spyder Screenshot	67
29.2 SVN Commits in the week of 26th of April	68
31.1 Lithium Polymer Fire-Proof Bag[Hobbyking, 2014]	70
33.1 USGS and Gaffey Spectra, Correlation Distance Results from Gaffey Reference 34	73
33.2 USGS and Gaffey Spectra, Correlation Distance Results from Gaffey Reference 35	75
33.3 USGS and Gaffey Spectra, Correlation Distance Results from Gaffey Reference 70	76
34.1 Consistent Spectral Measurements	78
34.2 Inconsistent Spectral Measurements	79
34.3 Distance Plot - Correlation Distance - Sample 10 as Reference	81

LIST OF TABLES

1	Meteorite Spectral Reflectance Characteristics from 350nm to 2500nm[Gaffey, 1976]	19
2	Vineyard Hyper-spectral Camera Specifications[Zarco-Tejada et al., 2013]	21
3	Vineyard Hyper-spectral UAV Specifications[Zarco-Tejada et al., 2013]	21
4	Front-end Lens Candidates	42
5	Lens Geometry for Focus	43
6	Bill of Materials - First Prototype Hyper-Spectral Camera	45
7	Bill of Materials - Second Prototype Hyper-Spectral Camera	49
8	Meteorite Samples	52
9	Results Log File Example	60
10	On-Board Devices	65
11	Python Libraries	66
12	Relevant Databases	67
13	USGS and Gaffey Spectra Differentiation, Gaffey Reference 34	74
14	USGS and Gaffey Spectra Differentiation, Gaffey Reference 35	75
15	USGS and Gaffey Spectra Differentiation, Gaffey Reference 70	76
16	Detection Results - Correlation Distance - Sample 10 as Reference	82
17	Detection Results - Correlation Distance - Sample 10 as Reference	82
18	Project Milestones	83

NOMENCLATURE

AGL	Above ground level - Used for altitude measurements
ANSMET	The Antarctic Search for Meteorites (USA)
ANSMET	US Government funded Antarctic Search for Meteorites program
CARE	Chinese Antarctic Research Expedition (PRC)
CCD	Charge Coupled Device
IDE	Integrated Development Environment
NIPR	Japan's National Institute of Polar Research
NIPR	National Institute of Polar Research (Japan)
RAMS	Robotic Antarctic Meteorite Search
SAM	Spectral Angle Mapper
SVN	Subversion - A version control system
UAV	Unmanned Aerial Vehicle

Part 1. Introduction

Meteorites are small parts of other worlds that are sought by scientists and collectors alike to help unlock the mysteries of the universe or to simply own a piece of it. Large scale meteorite collection today is mostly isolated to a few places on earth and is performed in a rather basic and manual manner mostly without modern tools. If the process of meteorite detection could be automated with relatively low cost equipment that was more conducive to detection in a greater number of locations, the rates of collection could increase rapidly.

This research project investigated the use of hyper-spectral imaging from an unmanned aerial vehicle (UAV) to search for meteorites. This involved research into meteorites already discovered, meteorite spectra and spectral matching techniques. It also involved the development of hyper-spectral imaging equipment suitable to be used with an unmanned aircraft.

A review of literature relevant to this project was undertaken to identify similarities in existing research and relevant literature and opportunities to build upon it. This is detailed at Part 2.

1. SUMMARY OF ACHIEVEMENTS

All of the mandatory items in the project specification in Appendix A have been achieved or have been considered to be unnecessary after further research. In addition to these achievements, item six in the programme, which was considered an optional has been partially achieved.

The design and construction of a low cost hyper-spectral line scan camera, as listed as point 1 in the project programme has been completed. The details of this achievement are located in sub-sections 15.2 and 15.3 for the first and second prototypes of the hyper-spectral imaging systems respectively.

Although a logging system to record frame number, speed and location has been developed, it has not been applied to the spatial reconstruction of the line scan images. This is due to the research that was completed as part of this project that has found that it was not necessary to spatially reconstruct the hyper-spectral images in order meet the objectives. It was found that the analysis of the spectrum of samples was sufficient.

Simulated aerial collection of hyper-spectral data has been completed and the data has been analysed for the presence of meteorites. This is an achievement outlined in item three of the project programme.

Research and development has been completed as in item four of the project programme, that has produced a system that analyses hyper-spectral data to determine the the presence of particular types of meteorites under full sun lighting conditions. The software that produces these results also logs the presence of the meteorites against the frame number and location of the find.

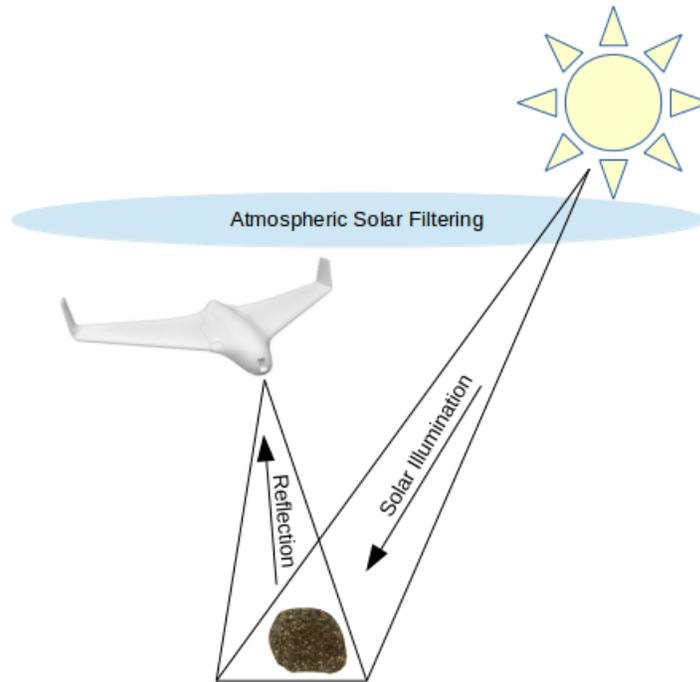
The effectiveness of the system that detects meteorites has been analysed and compared to existing meteorite collection techniques which has satisfied the achievement outlined in item five of the program.

Item six of the project programme was an optional achievement that could be completed if time allowed it. This has been partially completed in that the software has been written that enables the meteorite detection system to perform hyper-spectral analysis and meteorite detection on board an aerial vehicle. The hardware however has not yet reached the point where it is appropriate to be used on-board an aerial vehicle. The further work that would be required to complete this achievement is outlined in part 11.

2. OBJECTIVES

2.1. Technical Objectives. This project involved the design and construction of a hyper-spectral line scan camera adapted from a pre-existing machine vision video camera, with the aim of enabling the measurement of the spectral signatures of potential meteorites. The project also required the development of an image and position logging system for the purpose of determining potential meteorite locations. While the UAV aspects are out of scope of this research project, the constraints associated with the UAV have been considered in the design and construction of the hyper-spectral camera.

FIGURE 2.1. Airborne Meteorite Detection Overview: UAV image from UAV Robotics Australia[UAV-Robotics-Australia, 2014]. Meteorite Image from Meteorites Australia[Meteorites-Australia, 2014]



2.1.1. Technical Tasks Beyond Outside of Scope. Once the imaging system has been created it will need to be attached to an unmanned aerial vehicle in order to retrieve aerial hyper-spectral images of potential meteorite sites. Expeditions to appropriate sites to make flights and potentially to collect meteorite samples will also be undertaken. These expeditions, including the design and construction of the aerial vehicle, will be undertaken in collaboration with a third party and are not included within the scope of this research project.

2.2. Research objectives. The major research objective of this project was to determine if it is possible to use hyper-spectral imaging to detect meteorites from

aerial vehicles. To meet this research objective an image analysis system that can detect meteorites from their hyper-spectral signature and report their location will need to be developed. Once these hyper-spectral images have been collected this data will need to be analysed for the presence of meteorites. This will involve computer vision analysis of hyper-spectral images and the correlation of spectral results to predetermined meteorite characteristics and the reporting of the coordinates of objects of interest for collection. Once objects are collected, they will be assessed to determine if they are meteorites.

Part 2. Literature Review

This research project considers issues relating to current methods for the detection of meteorites, spectral reflectivity data for meteorites and hyper-spectral imaging. The below literature review examines information sources across these areas.

ANSMET is the organisation responsible for the collection of over a third of the approved meteorites catalogued by the Meteoritical Society and recorded in their Meteoritical Bulletin Database. As such, a significant proportion of the literature available focuses on the ANSMET approach and on Antarctic meteorites. It is important to note however that meteorites are just as likely to fall elsewhere on the planet as they are in Antarctica but are more easily detected there, (with current methods) due to high visual contrast between meteorites and the background, low levels of terrestrial rock and low moisture[Love and Harvey, 2014, Harvey, 2003].

The meteorite search strategies employed by the ANSMET, as noted by Harvey in 2003[Harvey, 2003] and in 2014 [Love and Harvey, 2014], typically follow a transect sampling model whereby a line is formed by the search team with a distance of tens of meters between each person. The team then moves forward together and visually detect meteorites on the ground with the naked eye. Harvey also states that “in spite of the availability of highly technological sensors, currently (2003) the most effective meteorite detector for Antarctic meteorite searches is the human eye”[Harvey, 2003]. Harvey attributes this to his observation that “no currently available electronic system can match the human visual system’s amazing capacity to rapidly differentiate a scene into its key elements and recognise those elements that are unique or out of place”[Harvey, 2003]. While this approach may be appropriate in areas where there is little terrestrial rock visible, such as parts of Antarctica, this method becomes significantly less fruitful when there is more background terrestrial rock[Harvey, 2003]. When practical, satellite imaging including multi-spectral imaging has been used to aid in the search for meteorites in Antarctica. This approach has however been limited to locating geography that is better suited to the manual search for meteorites rather than direct meteorite detection[Love and Harvey, 2014, Mardon, 2009, Lucchitta et al., 1987]. It is worth noting that since Harvey’s 2003 work, visual imaging systems and associated technologies have improved significantly.

FIGURE 2.2. ANSMET Team 2013-14 - Systematic Searching



A number of attempts have also been made to automate the search for meteorites using robotics. A notable and early example is the use of the Nomad rover

by the Robotic Antarctic Meteorite Search (RAMS) to find and classify meteorites autonomously in Antarctica. Nomad is an autonomous four-wheeled, harsh-environment rover that was equipped with a 24 bit colour camera and a 300-1100 nm fibre optic reflection spectrometer. The colour camera was used to find objects of interest and the spectrometer was used to collect hyper-spectral images that could confirm that the objects were or were not meteorites and to obtain a meteorite classification if appropriate [Apostolopoulos et al., 2001]. In January 2000 Nomad made the first autonomous classification of a meteorite by a robot [Pedersen et al., 2001]. The project was considered successful but the search was noted as being relatively slow due to mechanical and operational limitations [Apostolopoulos et al., 2001].

Airborne hyper-spectral imaging is used in a variety of applications and industries including agriculture and surveying [Kokaly et al., 2011, Zarco-Tejada et al., 2013, Calderón et al., 2013]. From my review of the available literature on this topic, airborne hyper-spectral imaging does not appear to have been used to detect meteorites. However, for surveying purposes, airborne imaging spectrometers are frequently used to allow the detection of materials and the mapping of their distributions across the landscape [Kokaly et al., 2011]. For example, in agriculture, aerial hyper-spectral images are regularly used to gather data on crops such as plant health and the presence of diseases [Calderón et al., 2013, Zarco-Tejada et al., 2013].

Meteorites have specific characteristics which are not shared by known rocks of Earth, making it possible to identify them even if they have not been seen to fall to Earth [Smithsonian, 2014]. Of particular relevance to this research project is the significant work that has been performed to collect the spectral reflectivity data of a wide range of rocks, minerals and more specifically, meteorites [Baldrige et al., 2009, Clark et al., 2007, CLOUTIS et al., 2010, Gaffey, 1976, 2001, Johnson and Fanale, 1973]. As part of this work by Gaffey, it was identified that each meteorite type representing a particular mineral assemblage and metamorphic grade has a characteristic spectral reflectance within 350 to 2500nm [Gaffey, 1976]. Gaffey categorised meteorite types into strong, weak and featureless spectral characteristics. The results of this categorisation are shown in 1.

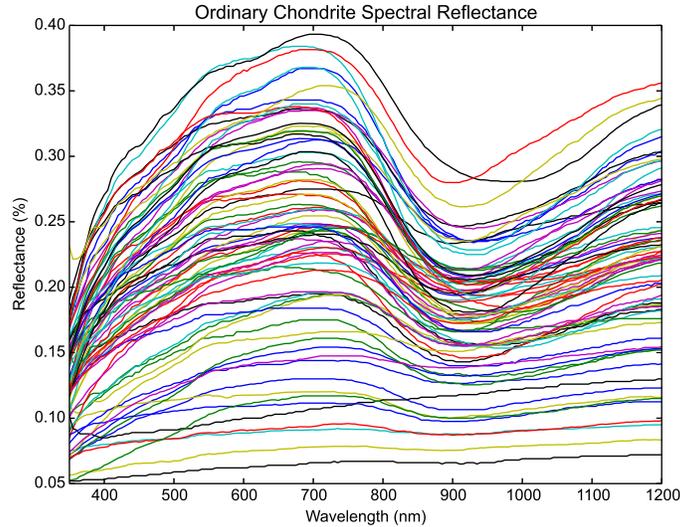
TABLE 1. Meteorite Spectral Reflectance Characteristics from 350nm to 2500nm [Gaffey, 1976]

Strong	Weak	Featureless
Ordinary Chondrites	Ureilites	Iron meteorites
Basaltic Achondrites (HED)	Black Chondrites	Enstatite Chondrites
Nakhlites	Stony Irons	Enstatite Achondrites
Angrites	Some Carbonaceous Chondrites	Some Carbonaceous Chondrites
Chassignites		

Meteorite spectral data from Gaffey's 1976 study has been archived and made available by NASA's Planetary Data System. The spectra of the 77 ordinary chondrites has been extracted from this data and plotted in figure 2.3. The characteristics of these ordinary chondrite spectral responses identified by Gaffey [Gaffey, 1976] are consistent with an investigation on the spectral responses of 18 ordinary

chondrite samples performed by Pentikäinen et al. and published in 2014 [Pentikäinen et al., 2014]. As such Gaffey’s findings will be utilised in this research project.

FIGURE 2.3. Ordinary Chondrite Spectral Reflectance [Gaffey, 2001]



More generally, hyper-spectral imaging is used for a range of quality control and identification and detection purposes. Significant work has been undertaken on hyper-spectral imaging in food quality and control (Sun 2010). It is also worth noting that the Commonwealth Scientific and Industrial Research Organisation (CSIRO) has developed hyper-spectral imaging software for minerals exploration, called The Spectral Assistant (TSA) [Berman, 2005]. The TSA supports the identification of potential new mine sites for commercial use.

Preprocessing of hyper-spectral images is a key component of their use for identification and detection purposes. The preprocessing of raw hyper-spectral images is almost always necessary [Vidal and Amigo, 2012]. Preprocessing can be used to remove unwanted features such as spikes, dead pixels, other noise and spatial data beyond the region of interest [Vidal and Amigo, 2012].

There are a range of methods for the matching of preprocessed hyper-spectral target images to known reference spectra [Kumar et al. [2010], Kang et al. [2013], Padma and Sanjeevi [2014], Drake et al. [1999]]. A simple method is to normalise the reflectance of the target image to that of the reference spectra at a given wavelength and to spectrally match the two sets of data with a distance function [Drake et al., 1999]. This would then yield a metric that describes the overall difference between the target and reference data. More complex methods include the Jeffries-Matusita (JM) approach and the Spectral Angle Mapper (SAM) approach. A combination of these two methods has shown to provide more accurate spectral matching results than either method in isolation [Padma and Sanjeevi, 2014], and as such will be considered as the approach used in this project.

As noted elsewhere in this paper, it is necessary to have knowledge of the spectral signature of the object that is the subject of the identification or detection process.

The work of Gaffey, discussed in some detail above, is therefore critical to this project. However, it is worth noting that there are several libraries of reflectance spectra of natural and man-made materials available for public use. These libraries are primarily based in the USA and provide a source of reference spectra that can aid the interpretation of hyper-spectral and multi-spectral images. The ASTER Spectral Library is one of these and is run by NASA, as part of its Advanced Spaceborne Thermal Emission and Reflection Radiometer (AS-TER) imaging instrument program. The library holds spectral reflectance data for over 2000 items, including minerals, rocks, soils, man-made materials, water, and snow [Smith, 2012].

Research completed by Zarco-Tejada et al. and published in 2013 involved airborne campaigns from 2009 to 2010 which collected hyper-spectral images of vineyards from two fixed wing unmanned aerial vehicles (UAVs). The aim of the collection of these hyper-spectral images was to measure the carotenoid content in vineyards. This was achieved with errors below 9.7 per cent. Table 2 summarises the specifications of one of the hyper-spectral video camera used while 3 summarises the specifications of one of the UAV platforms used[Zarco-Tejada et al., 2013].

TABLE 2. Vineyard Hyper-spectral Camera Specifications[Zarco-Tejada et al., 2013]

Characteristic	Value
Spectral Bands	260
Spectral Density	1.85 nm/pixel
Spectrum	400-885nm
Bit Depth	12
Frames per second	50
Sensor Integration Time	18ms
Slit size	25 μ m
Full-Width at Half Maximum	6.4nm
Front End Optics Focal Length	8mm
Field of View	50°

TABLE 3. Vineyard Hyper-spectral UAV Specifications[Zarco-Tejada et al., 2013]

Characteristic	Value
Wing Configuration	Fixed
Wing Span	5m
Flight Time	3hr
Take-off Weight	13.5kg
Ground Resolution	53cm x 42cm
Altitude	575m AGL
Ground Speed	75km/h

Part 3. Project Context

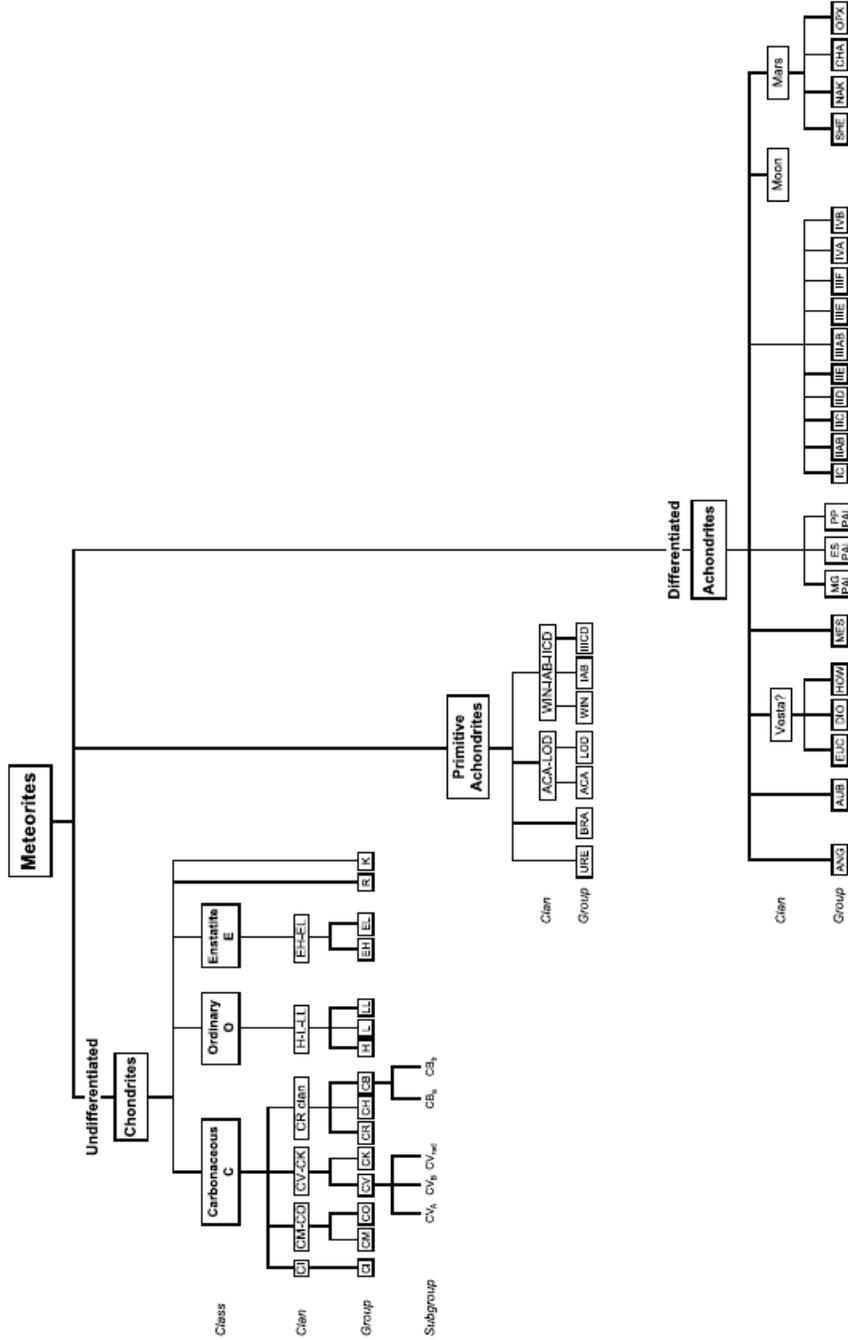
3. METEORITES

3.1. Reasons for Collection. Available evidence and research indicates that most meteorites are fragments of asteroids - samples from minor planets. As meteorites have extremely old formation ages many meteorites preserve chemical and physical properties that were established 4.5 billion years ago, during the earliest history of the solar system, and thus provide some of the best clues to the nature of the events that occurred in that remote time [Smithsonian, 2014]. Meteorites therefore provide a unique source of information about the early evolution of the Solar System [WA_Museum, 2013]. The study of collected meteorites is valuable for scientific research particularly in relation to planet formation, composition and the origin and sustainability of life[Véronique et al., 2012]. Meteorites also have commercial value to collectors and much of their trade is carried out through e-commerce[The-Meteorite-Market, 2014, Aerolite, 2014, Bathurst_Observatory, 2014, Meteorites-For-Sale, 2014, Meteorites-Australia, 2014].

3.2. Cataloguing of Meteorites. A database of all known meteorites is maintained by the Meteoritical Society, an international not-for-profit scholarly organisation. As at 14 May 2014 there were 48,973 total approved meteorites recorded by the Society [The-Meteoritical-Society, 2014]. The Society is responsible for the naming of known meteorites and publishes these in the database, the Meteoritical Bulletin. Meteorite statistics used in this report are derived from data extracted from the Meteoritical Bulletin as at 14 May 2014. This research project utilises statistics derived from the Meteoritical Society's Meteoritical Bulletin.

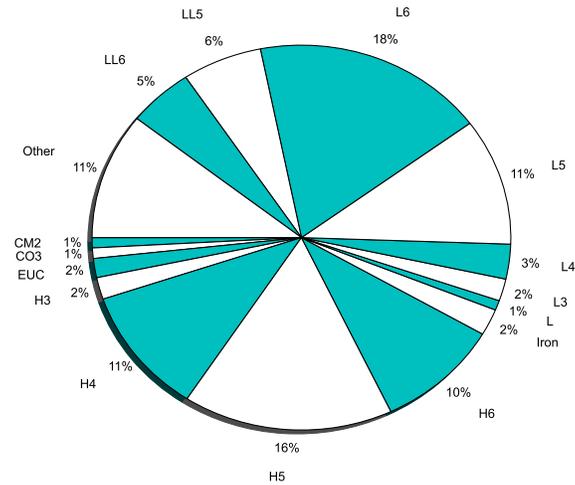
3.3. Types of Meteorites. There are a range of methods for meteorite classification that can be used in order to to group them by different attributes[Wiesberg et al., 2006]. The most prevalently used is that developed by Wiesberg et al. In this classification scheme, meteorites are most broadly divided into chondrites, primitive achondrites, and achondrites[Wiesberg et al., 2006]. The graphic shown in figure 3.1 which has been taken from the work of Weisberg et al. [Wiesberg et al., 2006], shows a classification method for meteorites that will be used throughout this document.

FIGURE 3.1. Meteorite Classification [Wiesberg et al., 2006]



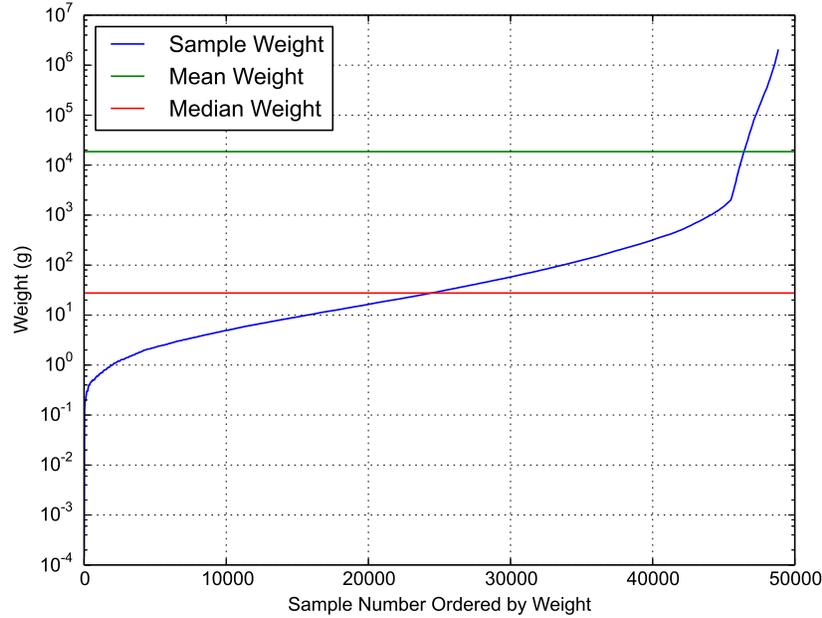
The distribution of the groups shown in figure 3.1 as a proportion of the Meteoritical Bulletin approved meteorite population is shown in figure 3.2.

FIGURE 3.2. Catalogued Meteorite Types [The-Meteoritical-Society, 2014]



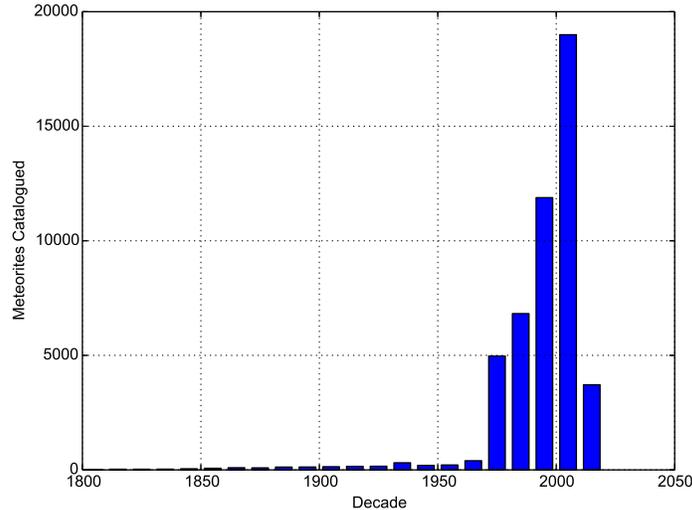
3.4. Meteorite Weight. There is a great variation in the weight of catalogued meteorites with the lightest weighing $150\mu g$ and the heaviest weighing $1998kg$. The distribution of the weights of the catalogued meteorites is shown in figure 3.3, ordered by weight and displayed don a log-linear scale. This data will be useful in approximating the sizes of meteorites to search for.

FIGURE 3.3. Catalogued Meteorites Ordered by Weight [The-Meteoritical-Society, 2014]



3.5. Meteorite Collection over Time. The number of meteorites that are collected and catalogued each decade has been rapidly growing since the 1970s [The-Meteoritical-Society, 2014]. Figure 3.4 displays this data including the incomplete decade of 2010 to 2020. Bevan (1992) notes that the discovery of major caches of meteorites preserved in the ice-fields of Antarctica, and the extension of dedicated meteorite collection programs in hot desert regions may also contain concentrations of meteorites that have accumulated with time, has led to the discovery of a large number of new meteorites [Bevan, 1992]. In part, this dramatic increase could be due to the beginning of the ANSMET program in 1976.

FIGURE 3.4. Meteorites Catalogued by Decade of Discovery [The-Meteoritical-Society, 2014]



3.6. Meteorite Collection Programs. There are a range of meteorite collection programs currently under way. These are primarily government sponsored and focus on Antarctica. A high profile and long established program is the US Government funded Antarctic Search for Meteorites program (ANSMET). ANSMET is a field-based science project that recovers meteorite specimens from Antarctica, with more than 20,000 recovered since 1976. A key finding from the ANSMET program is that some meteorite specimens are planetary materials, including from the Moon and Mars. This finding, coupled with the ongoing recovery of meteorites, provides scientists with access to planetary materials at a much lower cost and risk than a space mission. The ANSMET approach to recovering meteorites is discussed further in this paper's Literature Review, but is essentially a ground search by groups of scientists. It is resource intensive and time consuming, and is not without physical risks due to the extreme Antarctic weather conditions and remoteness [ANSMET, 2014].

In addition to ANSMET, there are a number of countries with emerging meteorite collection programs in Antarctica. Japan's National Institute of Polar Research (NIPR) has a research program involving meteorite collection. The focus of Japan's research using meteorites is discovering more about the history of the universe and changes in the Earth's crust and sea surface [NIPR, 2014]. In February 2014 South Korea opened their second year round station serve as a base camp for continental research including meteorite studies [Xinhua, 2014].

It is also worth noting that there are a number of private collectors of meteorites.

Figure 3.5 identifies the main programs responsible for the collection of catalogued meteorites. These are:

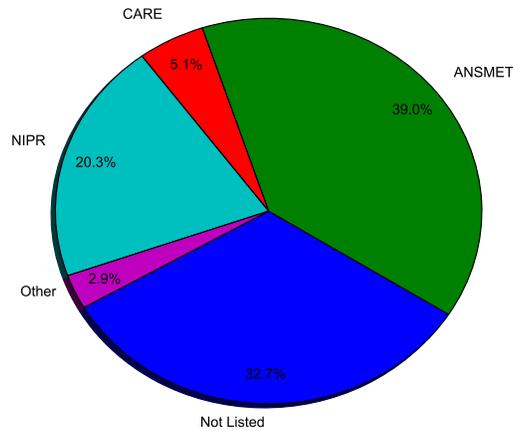
- - The Antarctic Search for Meteorites (USA)
- - National Institute of Polar Research (Japan)

- - Chinese Antarctic Research Expedition (PRC)

The first recorded meteorite discovered in Australia was found in 1854 near Cranbourne, Victoria [Bevan, 1992]. Australia has a low ratio of falls to finds (1:20) compared with other countries (e.g. USA 1:7) which is due to Australia's relatively small population and large land mass. However, Bevan (1992) notes that once normalised to population density, the rate of recovery of meteorites (falls and finds) in Australia exceeds that of most other countries of similar size and range of climatic conditions. In Australia, the Western Australian Museum holds one of the most significant meteorite collections in the southern hemisphere, with around 14,000 specimens from 750 distinct and described meteorites [WA_Museum, 2013]. The Museum also holds a number of meteorites collected from the Nullarbor Region. The Nullarbor Region is an area of generally treeless, limestone desert and as noted in the Western Australian Museum's website, the semi-arid to arid climate of the Nullarbor is conducive to the preservation of meteorites [WA_Museum, 2014]. With the exception of Antarctica, the Nullarbor Region is one of the most prolific areas in the world for meteorite finds. Bevan (1992) notes that as in Antarctica, the frequency of meteorite types in the population of meteorites collected to date from the Nullarbor Region is depleted in irons, and may differ from that in the rest of the world [Bevan, 1992]. The dating of the Nullarbor stony meteorites indicates that meteorites have been accumulating there on a stable surface for at least 35,000 years [WA_Museum, 2014].

A large number of additional meteorites from the Western Australian Nullarbor collected since the early 1980s (more than 500) are yet to be examined, so the exact population currently in collections is unknown, and may be twice the number currently identified. As at the time of writing these were yet to be classified [WA_Museum, 2014].

FIGURE 3.5. Programs Responsible for Collection of Catalogued Meteorites [The-Meteoritical-Society, 2014]

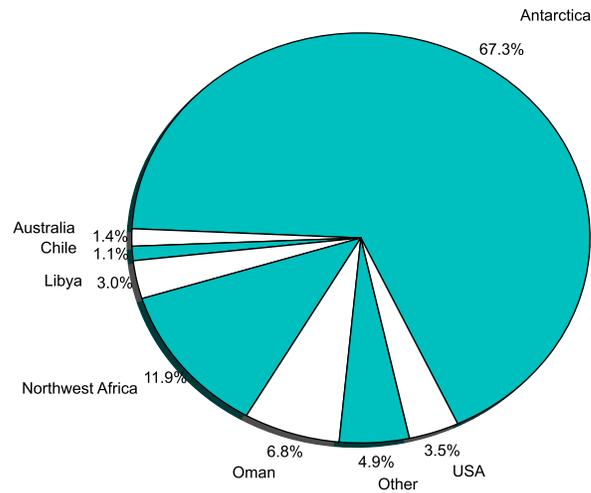


3.7. Meteorite Collection Locations. As noted in Section 3.6, the majority (67.3 per cent) of all catalogued meteorites have been located in Antarctica. This may be due to the lack of other terrestrial rock in the landscape which increases the visibility of meteorites [Harvey, 2003]. This may also explain why significant quantities of meteorites have been found in desert regions across the world. Further, the climactic conditions in both Antarctica and desert regions assist in preventing chemical weathering of fallen meteorites (through a lack of water in desert regions and the low temperatures in Antarctica [Imperial_College_London, 2014]. These climactic conditions mean that meteorites which fall in deserts can survive for long periods - up to 50,000 years. Over time as more samples fall a meteorite accumulation may result [Imperial_College_London, 2014]. Significant sites for meteorite location in hot desert regions include Northwest Africa (11.9 per cent) and Oman (6.8 per cent). Just over one per cent (1.4 per cent) of the world's catalogued meteorites were found in Australia [The-Meteoritical-Society, 2014].

While meteorites in hot deserts may survive for significant periods, the cold desert climate of Antarctica is even better for preserving meteorites, with some in Antarctica recovered with terrestrial residence times of greater than 2 million years [Imperial_College_London, 2014]. As such, a large accumulation can result over these time-scales. In addition, environmental changes which resulted in movement of the ice over significant stretches of time (tens of thousands of years) have resulted in high concentrations of meteorites in some locations - as high as one per square meter [ANSMET, 2014], resulting in a rich searching ground.

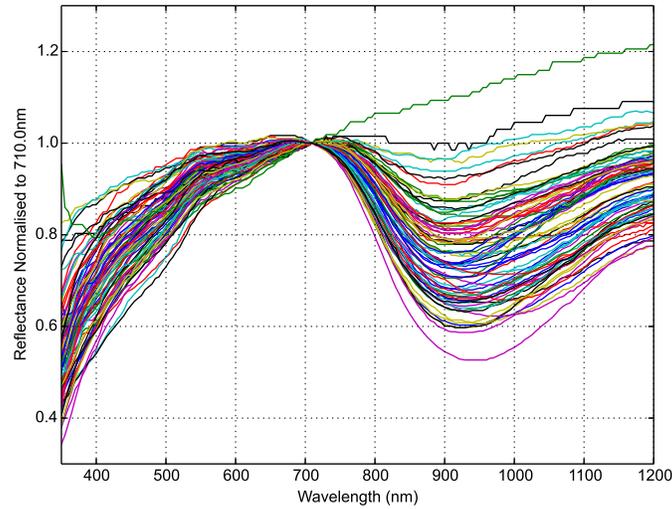
Figure 3.6 provides a further breakdown of meteorite collection locations.

FIGURE 3.6. Collection Location of Catalogued Meteorites [The-Meteoritical-Society, 2014]



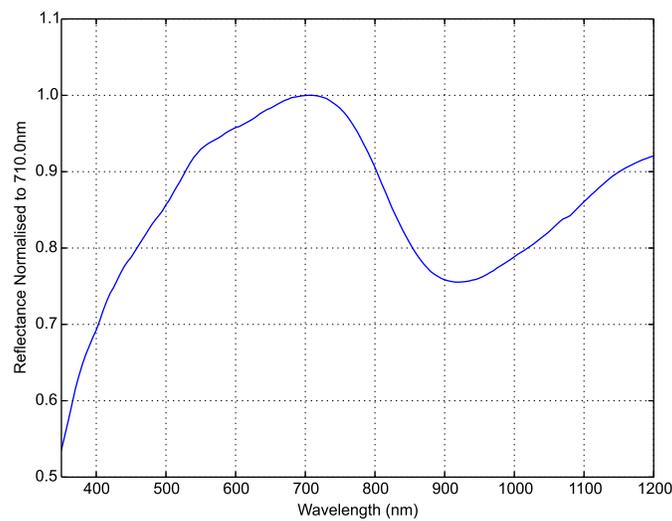
3.8. Meteorite Spectra. By taking the Gaffey spectral data of ordinary chondrites shown in figure 2.3 and normalising it to the wavelength at which the maximum reflectance is achieved (710nm), the plot in figure 3.7 is produced. This method allows for different brightness conditions to be compensated for whilst maintaining the spectral signature of the meteorites. For example if there are different lighting conditions or distances from the camera to the potential meteorite, the absolute brightness of the image will be changed but the relative spectral response will be maintained.

FIGURE 3.7. Spectral Reflectance of Samples of Ordinary Chondrites, Normalised at 710nm



If we take the mean of the normalised spectral response of the Gaffey ordinary chondrite samples that are shown in figure 3.7, we are left with a single normalised reflectance value for each wavelength and a general reference spectral response for all ordinary chondrites. This data is plotted in figure 3.8.

FIGURE 3.8. Ordinary Chondrite Reference Spectra

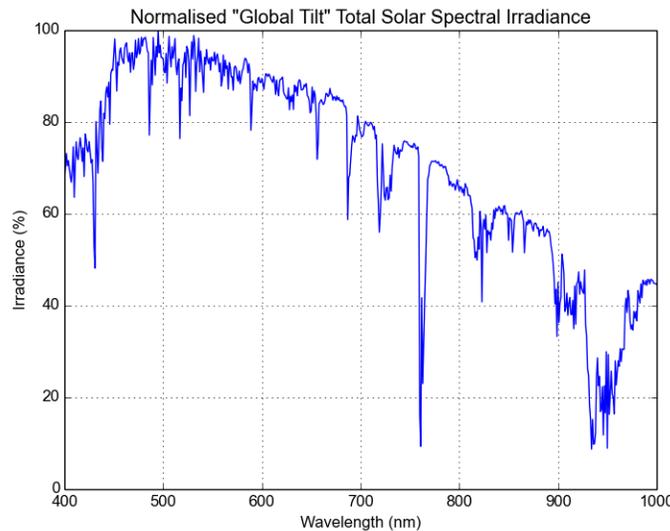


These meteorite spectra should resemble the ordinary chondrite spectra that is recorded as part of this research project. There will however be differences due to at least three factors:

- The spectral irradiance of sunlight after it has passed through the atmosphere
- The spectral response of the image sensor used in the hyper-spectral camera
- The accuracy of the wavelength measured by the hyper-spectral camera

The spectral irradiance of the sunlight after it passes through the Earth's atmosphere as shown in Figure 3.9, is non-uniform. There are several spectral troughs where the atmosphere has filtered out light over these bands. Due to this phenomenon, these attenuated bands will not be able to be reflected off meteorites with at the same level that neighbouring bands that are not attenuated as much are.

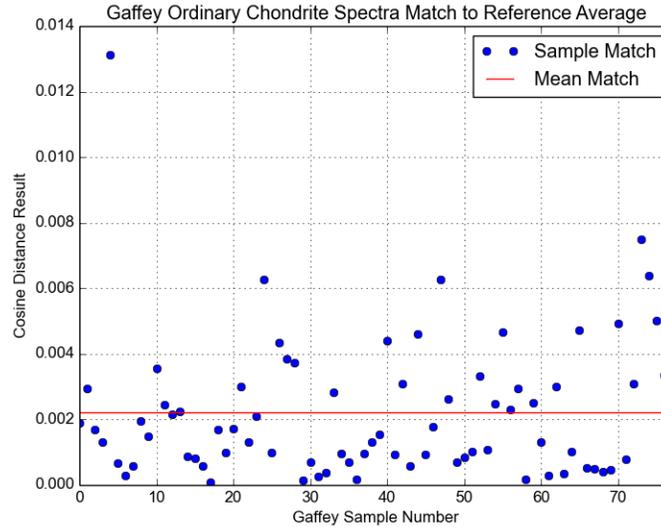
FIGURE 3.9. Solar Spectral Irradiance



The spectral response of the image sensor will have a similar effect as the image sensor is not evenly sensitive across its spectrum. This is discussed in more detail in Part 15.

The example in figure 3.10 uses the Canberra distance method to match the spectra in figure 3.7 with the reference spectrum of 3.8. By doing this we can quantify the variation in the spectra of Ordinary Chondrites obtained from Gaffey. This allows for the qualification of the method of using a single spectrum to match to all potential Ordinary Chondrites. This method applied to this data yields a distance result of between 0.97 and 16.13 with a mean distance result of 5.67.

FIGURE 3.10. Match to Spectral Reference of Ordinary Chondrite Sample Spectra



4. HYPER-SPECTRAL IMAGING

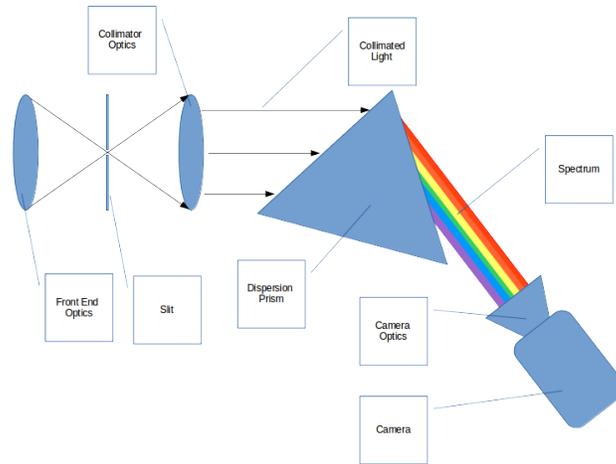
4.1. Technology. “Hyper-spectral imaging is a non-contact sensing technique for obtaining both spectral (or chemical) and spatial information about a sample” [Berman, 2005]. A hyper-spectral image captures the brightness of many bands of the electromagnetic spectrum for a given spatial picture.

Hyper-spectral images are produced by instruments called imaging spectrometers. As noted in the Literature Review, Harvey (2003) considered that machine vision was less effective than the human eye at identifying meteorites. Smith (2012) notes that hyper-spectral imaging was developed to address the limitations of regular machine vision and spectroscopic techniques.

Hyper-spectral imaging is used for a wide variety of purposes, including, of particular relevance to this research project, the detection and identification of geological features such as possible mine sites. There are also a number of emerging uses, including the use of hyper-spectral imaging to capture the biochemistry in tissue, either as a diagnostic of disease or as a bio-marker of specific metabolic processes [Berman, 2007]. The CSIRO is also developing hyper-spectral imaging solutions for production line quality assurance during the manufacture of pharmaceutical tablets. Hyper-spectral imaging allows manufacturers to confirm that active ingredients are present in correct concentrations and are distributed evenly through the tablet, and that no contaminants are present [Berman, 2005].

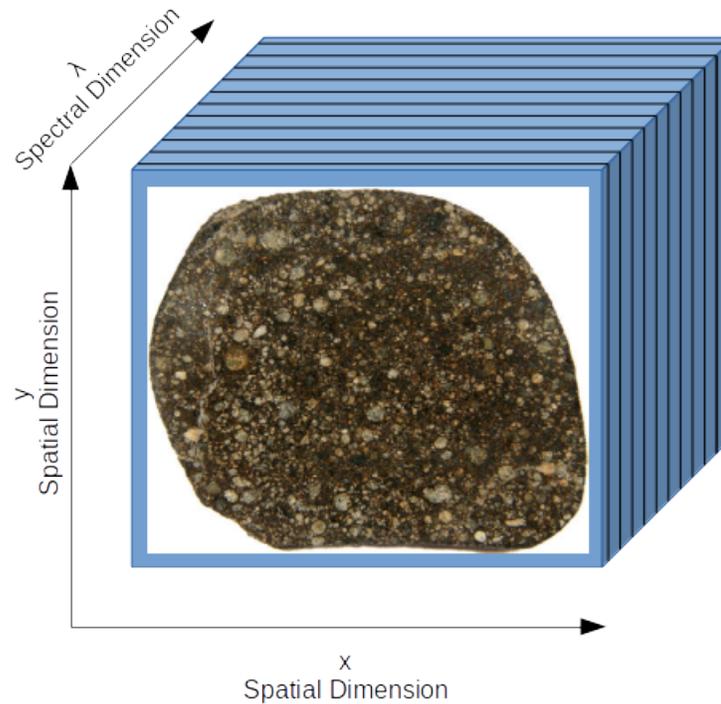
Hyper-spectral imaging is loosely distinct from a multi-spectral image by having a greater number of spectral bands (approximately more than 1000). This is achieved by spatially separating the component wavelengths of an image so that they can be recorded with a camera. The hyper-spectral camera dispersion process is shown in figure 4.1.

FIGURE 4.1. Hyper-spectral Camera Dispersion



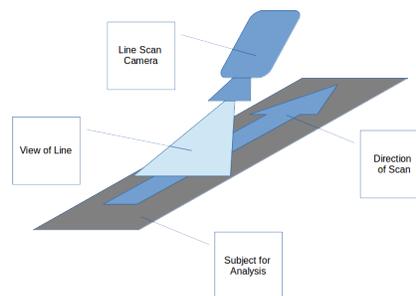
This third dimension of the image can be modelled with the hyper-spectral cube as seen in figure 4.2. This shows two dimensions of spatial resolution (x and y dimensions) and one of spectral resolution (denoted by λ).

FIGURE 4.2. Hyper-spectral Cube: Meteorite Image of “Northwest Africa 5393”[Meteorites-Australia, 2014]



As image sensors can generally only collect two dimensions, a third dimension is collected by scanning spatially. At any point in time, the hyper-spectral camera will capture spectral data for one of the two spatial dimensions. This will provide all of the spectral data for a single line of spatial information. As the camera is moved relative to the subject, spectral data for the the second spatial dimension is captured, completing the hyper-spectral cube. Figure 4.3 provides an overview of the line scan process.

FIGURE 4.3. Line Scan Camera



5. UNMANNED AERIAL VEHICLES (UAVs)

5.1. Definition. UAVs are aircraft that are not controlled by an on-board pilot. They can be autonomously controlled with on-board devices or they can be controlled remotely by a human pilot.

5.2. Types. UAVs differ greatly in form and function based on their wing configuration. These configurations include fixed wing, multi-rotor and helicopter. This project will design a camera suitable for use on a fixed wing model. Fixed wing UAVs are appropriate for the purposes of aerial mapping and terrain modelling larger areas and undertaking topographic surveys. This is due to the greater efficiency over distance that the configuration allows when compared to multi-rotor UAVs.

5.3. History. Modern UAVs emerged in the 1960s, with their usage by the USA reported during the Vietnam War [Garamone, 2002]. UAVs have historically been remote piloting, however there is an increasing shift to autonomous. Power storage improvements, innovative motor design and design and optimisation techniques have enhanced the civilian use of UAVs for a range of purposes [Logan et al., 2007]. Indeed UAVs are the fastest growing sector of the aviation industry with sales expected to top \$15 billion by 2014 [RMIT, 2014].

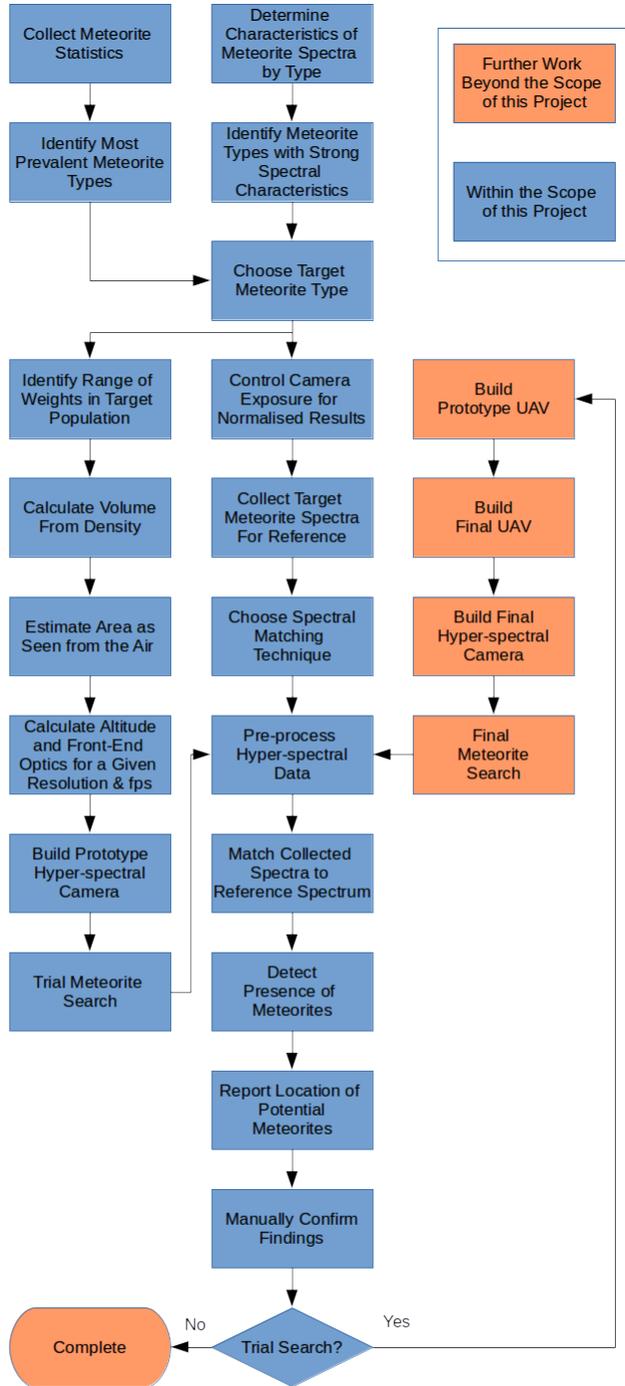
5.4. Modern Utility. UAVs are used for a variety of purposes in both the military and civilian domains. Military usage includes for the purposes of providing battlefield intelligence and for high risk combat operations [Blom, 2010]. They are also used to deliver cargo and for other types of logistic exercises. Civilian uses include agricultural surveying, film-making, search and rescue operations and logistic exercises including the delivery of supplies to remote areas [Mortimer, 2014].

Part 4. Project Methodology

6. METHODOLOGY INTRODUCTION

This section of the report outlines the proposed methodology for the project. Figure 6.1 is a flowchart which represents the relationships between aspects of the project methodology. The proposed methodology allowed for the timely adaptation of necessary changes as they arise whilst endeavouring to best predict the possible outcomes.

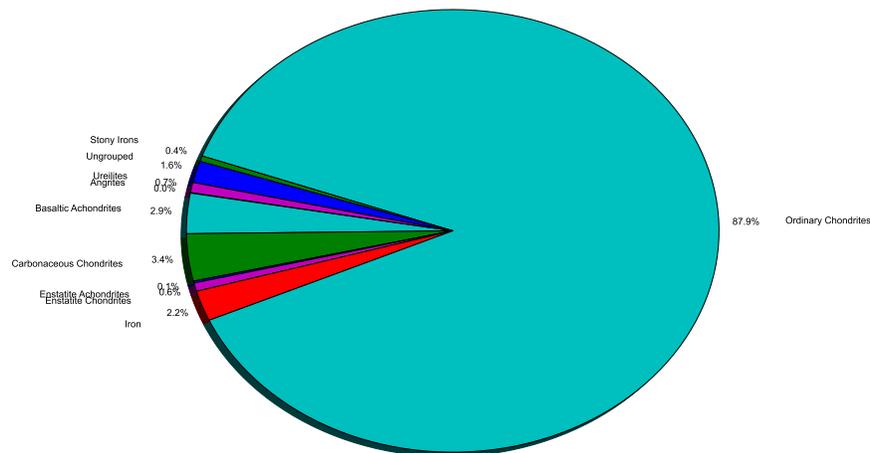
FIGURE 6.1. Overall Methodology Flowchart



7. PREVALENCE OF METEORITE TYPES

By applying the meteorite classification scheme shown in Table 1 to the Meteoritical Bulletin, it is possible to illustrate how these meteorite categories are reflected in the population of catalogued meteorites with the results shown in Figure 7.1. As is clearly seen in the figure below, the vast majority of meteorites that have been catalogued are ordinary chondrites.

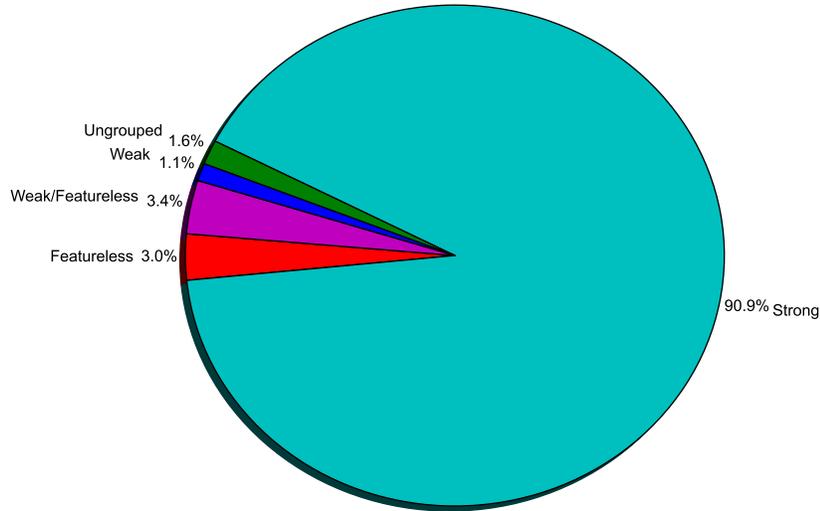
FIGURE 7.1. Catalogued Meteorites by Gaffey Spectral Group [The-Meteoritical-Society, 2014, Gaffey, 1976]



8. CHARACTERISTICS OF METEORITE SPECTRA BY TYPE

By further binning this data into the spectral characteristics identified by Gaffey in Table 1, we can illustrate the strength of the spectral characteristics of the catalogued meteorite population. This is shown in Figure 8.1.

FIGURE 8.1. Catalogued Meteorites by Spectral Characteristics [The-Meteoritical-Society, 2014]



Upon examination of Figures 1 and 8.1, it can be seen that the vast majority (90.9%) of catalogued meteorites exhibit strong spectral characteristics. This group of strong spectral character meteorites is mostly comprised of ordinary chondrites which account for 87.9% of the total population of catalogued meteorites.

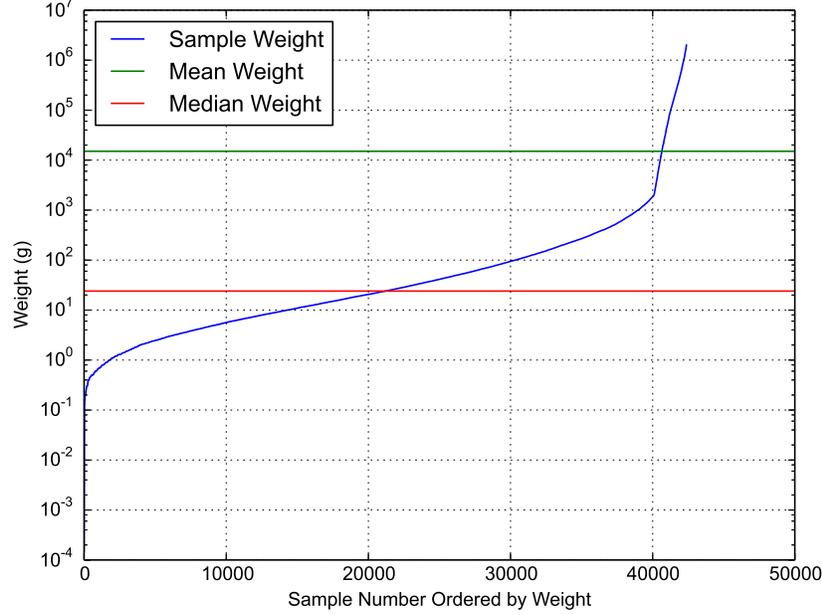
9. TARGET METEORITE TYPES

Based on the prevalence of ordinary chondrites as shown in figure 7.1, the strength of their spectral characteristics[Gaffey, 1976] and the simplicity and consistency of their spectral characteristics as shown in figures 3.7 and 3.8, this project will seek only to detect ordinary chondrites. This single reference spectrum will simplify the matching and detection process without placing too broad a limitation on the population of meteorites that will be covered.

10. WEIGHT OF TARGET METEORITE TYPES

The weight of ordinary chondrites catalogued in the Meteoritical Bulletin Database [The-Meteoritical-Society, 2014] range from $150\mu g$ to $1998kg$. They have a mean weight of $13.05kg$ and a median weight of $15.90g$. Figure 10.1 displays this data ordering meteorites by weight on a log-linear scale.

FIGURE 10.1. Ordinary Chondrite Weights [The-Meteoritical-Society, 2014]



11. VOLUME OF TARGET METEORITE TYPES

The average density of ordinary chondrites is $3.42g/cm^3$ for the H Group, $3.36g/cm^3$ for the L Group and $3.22g/cm^3$ for the LL Group [Consolmagno et al., 2008]. Applying these values to the weight values in the Meteoritical Bulletin [The-Meteoritical-Society, 2014] we can determine the approximate volume of catalogued ordinary chondrites using equation 11.1 where $\rho = \text{density}$, $m_{median} = \text{mass} = 15.90g$ and $m_{mean} = \text{mass} = 13050g$.

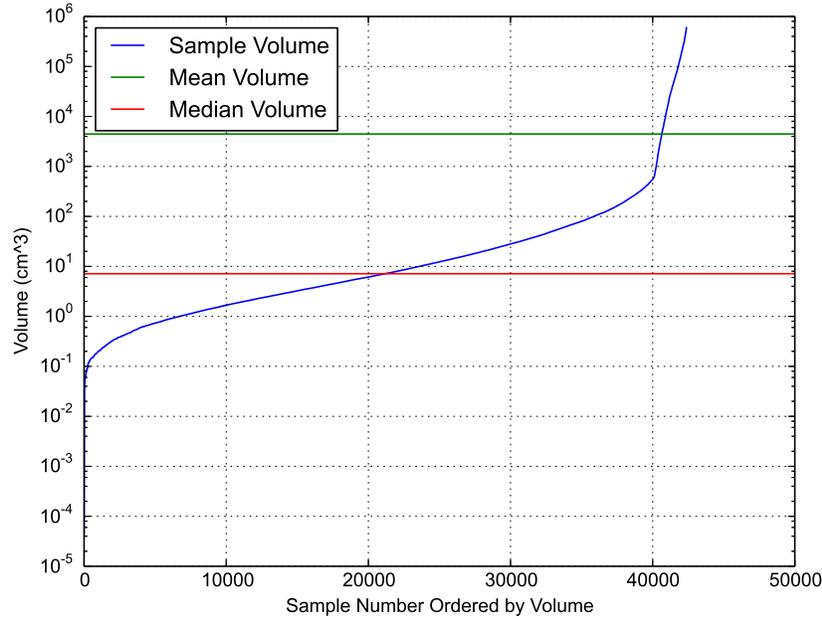
$$(11.1) \quad V = \frac{m}{\rho}$$

By applying these densities to their corresponding ordinary chondrite groups using a python program (listed in appendix C) acting on the meteoritical database of informations, the following volumes are yielded:

A mean volume of $4453.47cm^3$ and a median volume of $7.14cm^3$.

These are displayed, along with the volumes of the entire catalogued ordinary chondrite population in figure 11.1.

FIGURE 11.1. Ordinary Chondrite Volumes



12. AREA OF TARGET METEORITE TYPES AS SEEN FROM THE AIR

If we make the worst case assumption that the volume of the target meteorite is distributed evenly throughout a sphere, we can determine an approximate worst case area of the target object as viewed from the air. This assumption is based on the further assumption that a meteorite that does not have infinite axes of symmetry will most likely come to rest with a larger area viewable from the air than that of a sphere. This is done by applying equation 12.2, which yields the radius of the sphere as 1.20cm for the median volume of 7.14cm^3 (weight, density and volume are explored in previous sections). The other metric that will need to be extracted from this data is a minimum area detectable as viewed from the air. This will reduce the detectable population of meteorites but will allow for a lower ground resolution (measured in ground area per pixel) which will lead to a greater area being covered. This information can be used to tune the front-end optics and altitude of the UAV. By applying the equation for the area of a circle, as shown in equation 12.2, we are left with an area of the median ordinary chondrite as viewed from the are of 4.48cm^2 .

$$(12.1) \quad r = \sqrt[3]{\frac{3V}{4\pi}}$$

$$(12.2) \quad A = \pi r^2$$

13. AIRCRAFT ALTITUDE & FRONT-END OPTICS

13.1. Field of View. The altitude above ground and the focal length of the front-end optics will allow for the tuning of the ground resolution. This has been chosen based on the minimum detectable meteorite area as viewed from the air of 4.48cm^2 . This number has been taken from an approximation of the median ordinary chondrite volume evenly distributed throughout a sphere. By setting the spatial width viewed by a single pixel of the hyper-spectral camera to the diameter of this area, which is to 2.39cm , we have a starting point for calculating the requirements of the front end optics of the camera. Given that there are 680 spatial pixels of resolution in the hyper-spectral camera, the diameter of the field of view is given by the product of this spatial resolution and the diameter to be viewed by a single pixel, The result of this is 1624.59cm or 1.63m .

Four lenses of different focal length have been considered for use as the front end optics of the hyper-spectral camera. These lenses are manufactured by Edmund Optics and are part of their ‘‘Techspec’’ range. They have been chosen for consideration for this project due to their apparent performance in the visible to near infra-red range, their fixed focal length and their ability to operate with the size of sensor being used in this project. The specifications of the four lenses in this series that are being considered for this project are shown in the first three columns of table 4[Edmund-Optics, 2014].

By taking the known values of lens field of view, field of view diameter, equation 13.1 can be used to generate the altitude of the aerial vehicle above the ground.

$$(13.1) \quad d = \frac{FOV_{dia}}{2} / \tan \frac{FOV_{lens}}{2}$$

Where d is the distance of the aerial vehicle above ground level, FOV_{dia} is the diameter of the field of view (previously calculated to be 1.63m) and FOV_{lens} is the angular field of view of the lens. By using the python script listed in appendix F, the values of altitude for each of the lenses that have been considered have been populated in the fourth column of table 4.

TABLE 4. Front-end Lens Candidates

Part Number	Focal Length (mm)	Field of View, $1/2^\circ$ Sensor ($^\circ$)	Altitude for 1.63m Field of View (m)
67-714	16	22.7	40.47
67-715	25	14.5	63.85
67-716	35	10.4	89.26
67-717	50	6.2	149.87

In Australia, as discussed in section 30, unmanned aircraft are permitted to fly only up to 400 feet which is equivalent to 121.9 metres above ground level. This rules out the use of the 50mm focal length lens which would require the altitude above ground level to be 149.87 metres. The other three lenses however are within this legislated limit.

13.2. **Focus.** Now that candidate front end lenses have been identified along with their focal lengths and corresponding altitudes for appropriate field of view, the distance between the lens and the optical slit on the hyper-spectral camera can be determined. This distance is what will cause the image to be in focus for a given lens focal length and altitude above ground level. The relationship between these three parameters is approximated using the thin lens approximation shown in equation 13.2.

$$(13.2) \quad \frac{1}{f} = \frac{1}{u} + \frac{1}{v}$$

Where f is the focal length of the lens, u is the altitude of the aerial vehicle above ground level and v is the distance between the front end lens and the optical slit of the camera. Transposing for the distance between the front end lens and the optical slit as the subject yields equation 13.3.

$$(13.3) \quad v = 1 / \left(\frac{1}{f} - \frac{1}{u} \right)$$

A python script was used to solve this equation for each of the three remaining lens candidates. This program is listed in appendix F. The results of this are listed in table 5.

TABLE 5. Lens Geometry for Focus

Part Number	Focal Length (mm)	Altitude for 1.63m Field of View (m)	Distance between lens and optical slit (mm)
67-714	16	40.47	16.006
67-715	25	63.85	25.010
67-716	35	89.26	35.014

Although any of these three lenses would be appropriate, the 35mm lens has been selected for use aboard an aerial vehicle due to its altitude above ground level. this additional height above the other options provides an increased buffer between the UAV and geological features as well as helps to account for drift in measured altitude accuracy. The additional length of 35mm that will be added to the overall length of the hyper-spectral camera of 150mm (not including the front end lens) is of little consequence.

14. UAV COVERAGE & SPEED

From the calculations completed in section 13, the diameter of the spatial field of view for a single spatial pixel is $2.39cm$. If this much ground is covered with every frame and with a frame rate of $60fps$ then the minimum ground speed of the aerial vehicle that will not cause any overlap in covered area is $143.4cm$ per second or $0.1434m/s$ or $0.51624km/h$. This result is given by equation 14.1.

$$(14.1) \quad V = FOV_{pixdia} \cdot fps$$

Where V is velocity, FOV_{pixdia} is the diameter of the field of view of one pixel and fps is the frame rate of the image sensor. The rate of covered area is given by equation 14.2.

$$(14.2) \quad Coverage_{rate} = FOV_{pixdia} \cdot FOV_{dia} \cdot fps$$

Where FOV_{dia} is the full diameter of the field of view as opposed to just one pixels worth. This formula yields $23.3m^2/s$ which is approximately enough to cover the area of a rugby field in 7 minutes and 12 seconds.

As the speed is increased beyond $0.51624km/h$ the rate of coverage does not decrease but instead the area that is covered is spread out over a larger area, with gaps of uncovered area in-between covered areas. The purpose of this system is to detect meteorites for collection as opposed to determining if a particular area contains meteorites or not, therefore this apparent shortcoming of the system does not effect the goals of the project.

If the frame rate of the image sensor was to be increased, the coverage rate and minimum ground speed for no overlap would increase linearly.

15. HYPER-SPECTRAL CAMERA DEVELOPMENT

15.1. Introduction. One of the goals of this project was to develop a low cost hyper-spectral camera from an existing machine vision digital video camera. This process has been completed after building two prototype models.

15.2. First Prototype Hyper-Spectral Camera. The first prototype of the hyper-spectral camera is a fully functional spectrograph and video camera, but has not been used to record spectral data from rock samples and was not intended to be mounted in a UAV at any stage. The purpose of this piece of equipment is to develop and verify the geometry of the hyper-spectral camera before it is placed into an environment with more stringent requirements. The open-frame design with moveable lenses make the prototype easy to change the focus of the lenses and to exchange slits. The angle and position of the prism and camera relative to one another and the spectroscope are also easily adjustable.

The major disadvantage of this open frame design is that because of the flexibility of the components it is also easy to change the geometry of the device inadvertently which can cause radically different results in the spectral analysis and contaminate the data. Although the open-frame design allows for access by the user, it also allows uncontrolled light to enter the device at many stages. This is avoided by operating the system with a large box covering all of the components and with a single entry point for light into the front end of the device.

FIGURE 15.1. First Prototype Hyper-spectral Camera



TABLE 6. Bill of Materials - First Prototype Hyper-Spectral Camera

Description	Usage	Part Number	Manufacturer	Cost (qty:1)
Spectrograph body	1	Custom 3D print	–	AUD45.75
Optical Slit	1	Custom Cardboard	–	AUD0.02
Collimator Lens	1	32721	Edmund Optics	AUD12.50
Front End Lens	1	32721	Edmund Optics	AUD12.50
Dispersion Prism	1	47277	Edmund Optics	AUD100
Camera Lens	1	52160	Tamron	USD129 - No Cost Loan
Camera	1	USB-CAM-R0.5	Nimble Embedded	AUD180 - No Cost Loan
Base Board	1	DRBD45	Kia Ora	AUD15
				Total: AUD493.50*

*Based on USD:AUD of 1 USD = 1.15300 AUD as at 5/10/2014 [XE, 2014]

15.3. Second Prototype Hyper-Spectral Camera. The second prototype of hyper-spectral camera was built once the geometry of the optics was tested in the initial prototyped and confirmed. This prototype differs from the first in that it has a closed frame design that does not allow access to the internal optics and fixes them in place. At the same time it also blocks external light from entering the device from anywhere except the front-end. The body is constructed using a different method of 3D printing that produces a less accurate yet more durable result.

FIGURE 15.2. Second Prototype Hyper-spectral Camera with Front-End Optics attached



The camera is configurable to have the front-end optics fitted or removed depending on the distance to the subject to be analysed.

FIGURE 15.3. 3D Transparent Camera Assembly

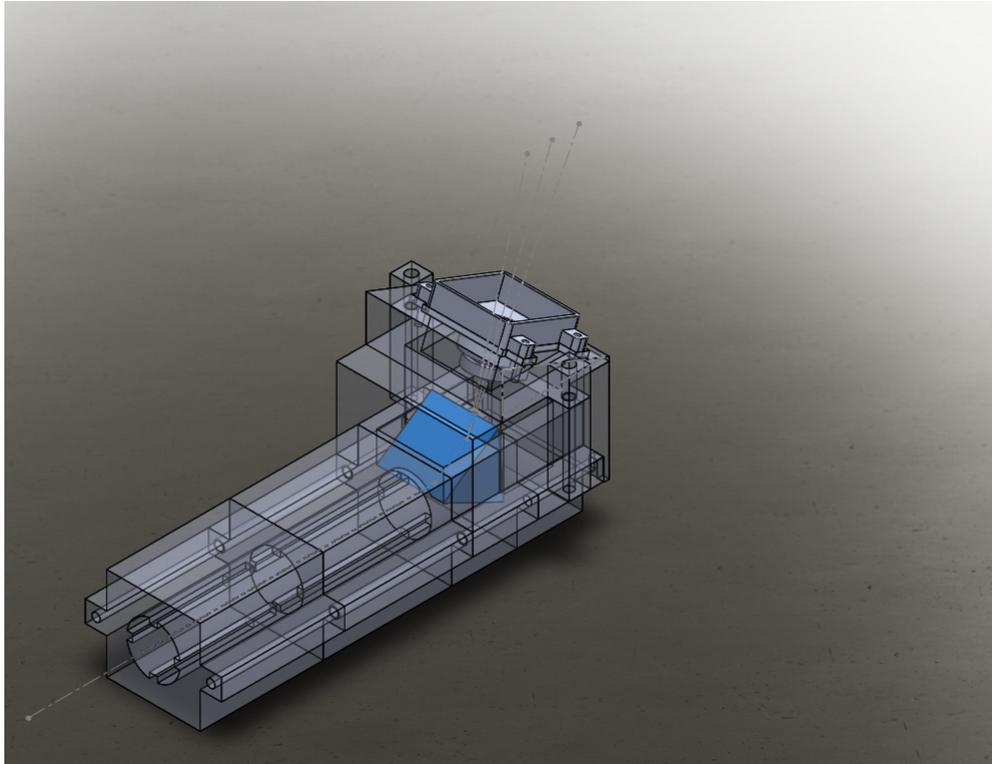
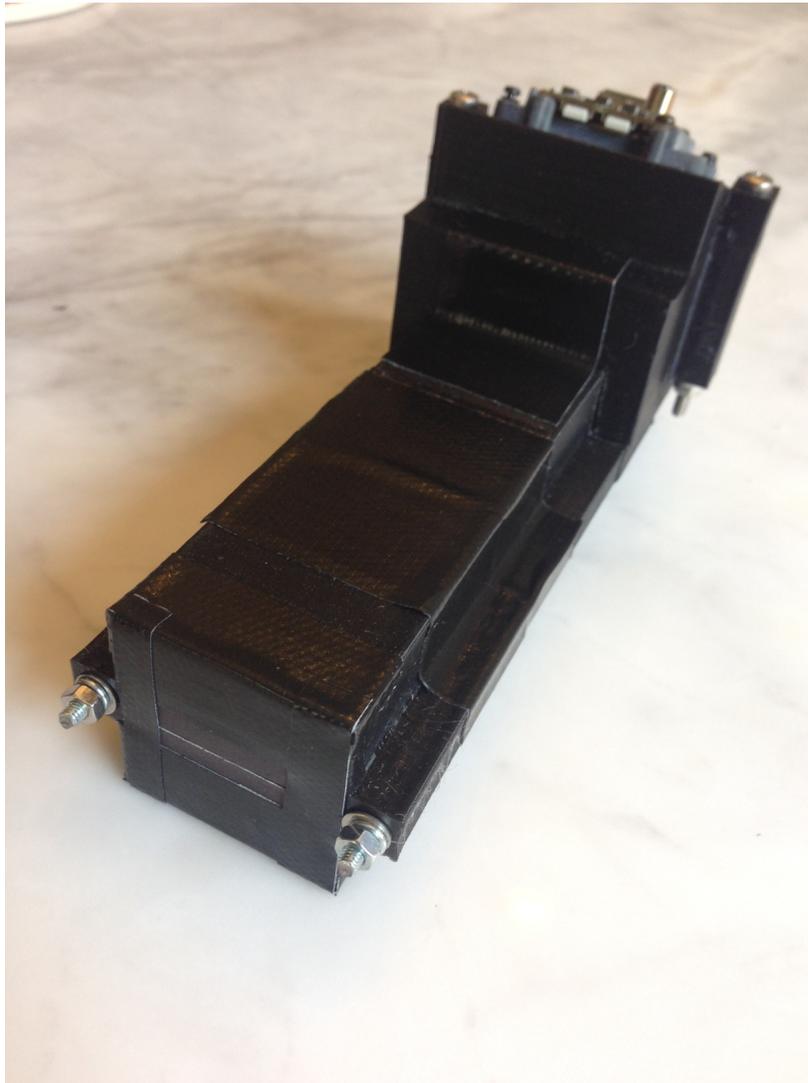


Figure 15.3 shows the assembly of the 3D model which includes 6 individual extruded 3D printed pieces as well as a dispersion prism and camera housing. The dotted lines show the path of light as it enters the camera assembly at the front and is dispersed by the prism and ending up on the camera lens. This model was used to verify the geometry calculations of the optics. It was also used to define the geometry of the sections of camera body to be 3D printed.

FIGURE 15.4. Second Prototype Hyper-spectral Camera without Front-End Optics attached



The second prototype also saw a change to a different camera. The image sensor that will be used for the construction of this hyper-spectral camera will be the e2v EV76C661 and is part of a camera made by the camera manufacturer Ximea. This sensor has been chosen for its global shutter, good response across the VIS-NIR range, its chrominance (monochrome) and its availability to the researcher. A plot of its spectral response is shown in green in figure 15.5

A comparison of the spectral responses of the image sensors used in the first and second hyper-spectral camera prototypes can be seen in Figure 15.5. The blue curve represents the e2v EV76C560 which is the image sensor used in the camera from the first prototype whereas the green curve represents the e2v EV76C661 which

is the image sensor used in the camera in the second prototype of hyper-spectral camera.

FIGURE 15.5. Image Sensor Spectral Response Comparison

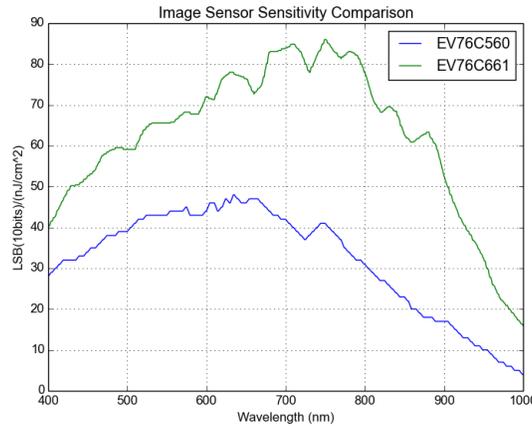


TABLE 7. Bill of Materials - Second Prototype Hyper-Spectral Camera

Description	Usage	Part Number	Manufacturer	Cost (qty:1)
Spectrograph body	1	Custom 3D print	–	0
Optical Slit	1	Custom Cardboard	–	AUD0.02
Collimator Lens	1	32721	Edmund Optics	AUD12.50
Front End Lens	1	52160	Tamron	USD129 - No Cost Loan
Dispersion Prism	1	47277	Edmund Optics	AUD100
Camera Lens	1	B3M16018	Lensation	EUR60
Camera	1	MQ013CG-E2-BRD	Ximea	AUD619- No Cost Loan
				Total: AUD818.073*

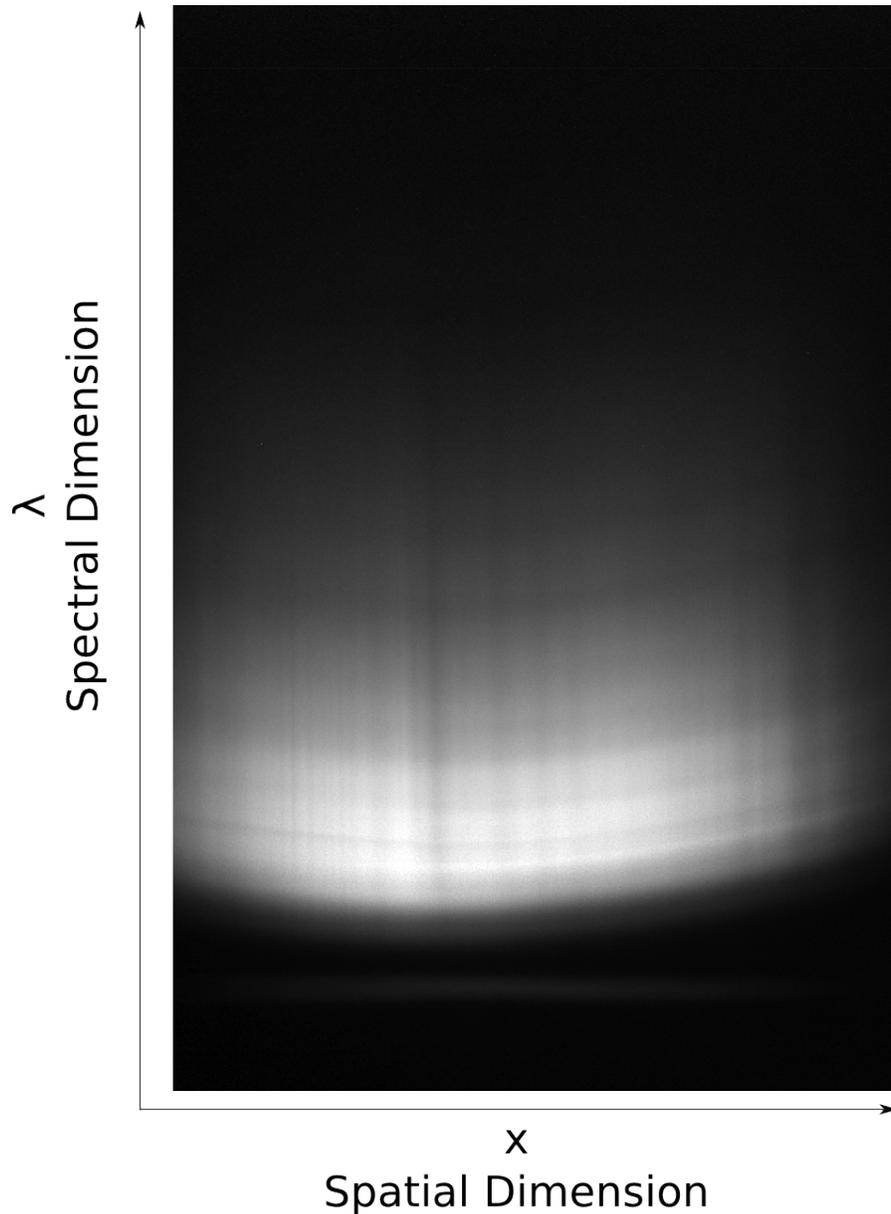
*Based on EUR:AUD of 1 EUR = 1.44284 AUD as at 5/10/2014 [XE, 2014]

15.4. Ximea API Python Connector. The Ximea Camera that was used in the second prototype of the hyper-spectral camera is controlled through a proprietary API that is written in C programming language. The program code for the hyper-spectral camera however is written in the python programming language. In order to communicate to the camera using the python language, a connector was written. This connector, named “xipyAPI” proved functional for the purposes of this research project and has since been open sourced so that it can be utilised or advanced by others without cost. It is hosted on Github at <http://github.com/DavidHenryM/xipyAPI>.

15.5. Hyper-spectral Image Capture. The hyper-spectral camera that has been built for this project is configured as a line-scan camera. This means that for a given still image frame, there will be one line of spatial dimension, and many lines of spectral dimension. The second prototype hyper-spectral camera allows for 680 units of spatial dimension and 1024 units of spectral dimension. As discussed in

sub-section 19.2, for the purposes of this project, the spatial resolution has been reduced to one pixel.

FIGURE 15.6. Hyper-spectral image captured from NWA869 L4-6 Chondrite (from solar reflection)



16. TARGET METEORITE REFERENCE SPECTRA

The target meteorite reference spectra will be taken from the spectroscopy work performed by Gaffey on 77 different ordinary chondrites of varying taxonomical

groups[Gaffey, 2001, 1976]. These spectra are displayed in figure 2.3 in Part 2 of this document.

16.1. **Samples.** In order to develop a system that could differentiate between meteorites and terrestrial rocks, several samples of each were acquired. For this exercise, 5 identified terrestrial rocks were acquired along with 8 unidentified terrestrial rocks. The terrestrial rocks, shale, granite, limestone, sandstone and basalt and were numbered as samples 17 to 21 respectively. These samples are pictured in Figure 16.1.

FIGURE 16.1. Identified Terrestrial Rock Samples



The unidentified terrestrial rocks were collected from a suburban garden and are pictured in Figure 16.2.

FIGURE 16.2. Unidentified Terrestrial Rock Samples



The meteorites used in this research project were all ordinary chondrites that were purchased from the online meteorite marketplace, “The Meteorite Market”. The meteorite samples were comprised of ordinary chondrites of different meteorite classification groups, different weathering grades and both cleanly cut and polished faces and crusts. These characteristics for the different samples are shown in Table 8.

TABLE 8. Meteorite Samples

Sample#	Name	Type	Weathering Grade	Surface
9	NWA 869	L4-6	-	Crust
10	NWA 869	L4-6	-	Cut
11	NWA 774	H4	2	Crust
12	NWA 774	H4	2	Cut
13	NWA 775	L6	3	Cut
14	Gao-Guenie	H5	-	Crust
15	NWA 791	L6	-	Cut
16	Kharabali	H5	-	Crust

Meteorites NWA869 and NWA774 were each sampled twice. This was due to them each having both a “crust” and a “cut” side. This is the reason for the absence of meteorite samples shown in position 10 and position 12 of Figure 16.3.

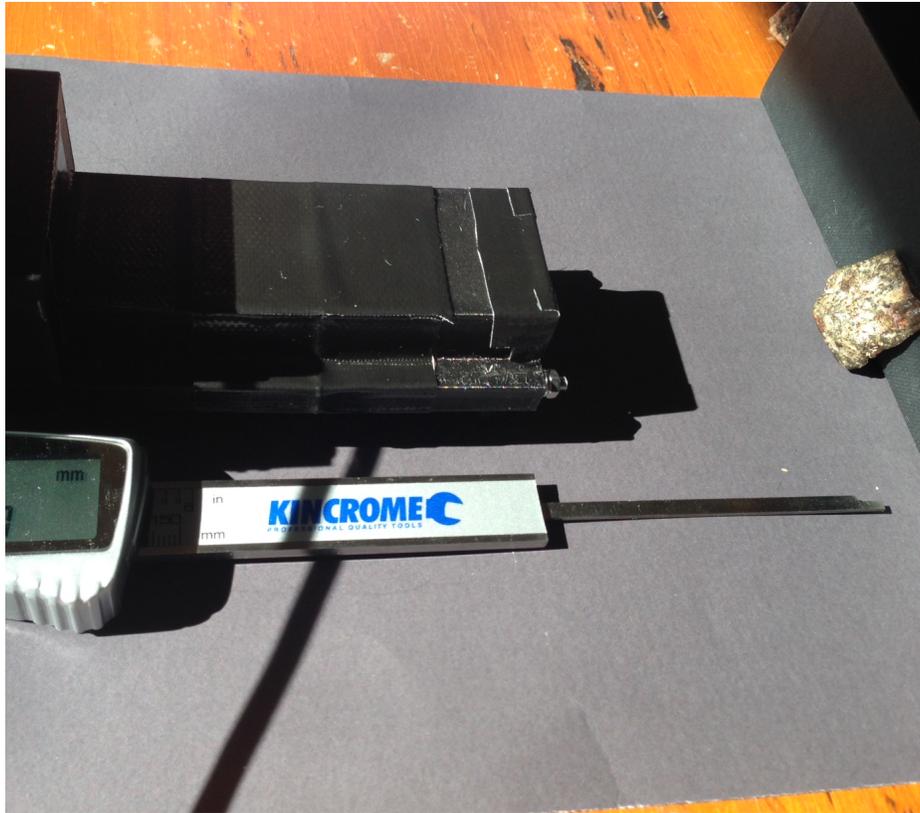
FIGURE 16.3. Meteorite Samples



17. SPECTRAL REFERENCE CAPTURE

Using hyper-spectral imaging for the purposes of detection requires prior knowledge of the spectral signatures of the items likely to be present in the scene which is to be scanned [Chang, 2003]. To be able to detect meteorites with a hyper-spectral camera, information on their spectral properties must first be recorded. The spectral reflectance data was recorded three times for each of the 21 samples on three different days. This data was collected using the second prototype hyper-spectral camera. This was done with both the hyper-spectral camera and the samples in a stationary position. The illumination of the samples was from direct sunlight with zero visible cloud cover on all three occasions. Before recording a single frame for each sample, an exposure control algorithm running on the computer, as opposed to the camera, adjusts the exposure to within a window of acceptable maximum pixel intensity and minimum number of saturated pixels. The distance between the hyper-spectral camera and the surface of the sample was 50mm.

FIGURE 17.1. Configuration of Data Collection



After this process was completed, a library of 63 spectra was recorded for further analysis.

18. SPECTRAL MATCHING TECHNIQUE

There are multiple spectral matching techniques that are available to be used as outlined in Part 2.

One of the methods that could be used is the method of taking the ordinary chondrite meteorite spectra obtained in the Gaffey research and outlined in Part 2 and sub-section 3.8 and matching potential meteorite samples to it. This process introduces two major difficulties. The first is that the spectrum of the reference will need to be aligned very accurately with the spectrum of the sample. This means that the output of the hyper-spectral camera would have to be calibrated to a high degree of accuracy in order to match up to the reference spectra without significant error.

There is extra processing that would need to occur if this method was to be utilised. This is due to the images that are captured being reflections from the sun off the sample. As discussed in sub-section 3.8, the spectrum of the sun after it is filtered by passing through the Earth's atmosphere is not uniform and has multiple bands that are attenuated to a much greater degree than others. The response of

the image sensor used in the hyper-spectral camera would also need to be taken into account if this method was used. This is because it has different levels of sensitivity across its spectrum. The raw hyper-spectral image would then have to be processed to normalise for the solar irradiance and image sensor sensitivity. This process of normalisation increases the processing time of an image and also reduces the depth of the resulting data.

Another possible spectral matching technique, which was the spectral matching technique that was used in this research project, is to match the sample spectra to a reference spectrum captured by the same hyper-spectral camera under the same conditions. This method eliminates any differences that may be present in a reference taken from a remote source, (such as the Gaffey spectra) and the captured spectra of the samples. It also eliminates the requirement to accurately calibrate the spectral axis of the hyper-spectral images. It does however introduce the additional task of compiling a reference spectrum from meteorite samples. Due to the simplicity of this method, it has been selected for use in this project.

The process of selecting a distance method, to determine the difference between a sample spectrum and the reference spectrum, has required significant evaluation to identify the method best suited to this application. As such multiple distance algorithms were trialled to find the one with the greatest ability to differentiate between the samples of terrestrial and extra-terrestrial rock.

The distance algorithms that were trialled were:

- Bray-Curtis distance

$$(18.1) \quad d(u, v) = \frac{\sum |u_i - v_i|}{\sum |u_i + v_i|}$$

- Canberra distance

$$(18.2) \quad d(u, v) = \sum_i \frac{|u_i - v_i|}{|u_i| + |v_i|}$$

- Chebyshev distance

$$(18.3) \quad d(u, v) = \max_i |u_i - v_i|$$

- Manhattan (City Block) distance

$$(18.4) \quad d(u, v) = \sum_i |u_i - v_i|$$

- Correlation distance

$$(18.5) \quad d(u, v) = 1 - \frac{(u - \bar{u}) \cdot (v - \bar{v})}{\|(u - \bar{u})\|_2 \|(v - \bar{v})\|_2}$$

where \bar{u} is the mean of the elements of u and $(u - \bar{u}) \cdot (v - \bar{v})$ is the dot product of the two terms in parenthesis.

- Cosine distance

$$(18.6) \quad d(u, v) = 1 - \frac{u \cdot v}{\|u\|_2 \|v\|_2}$$

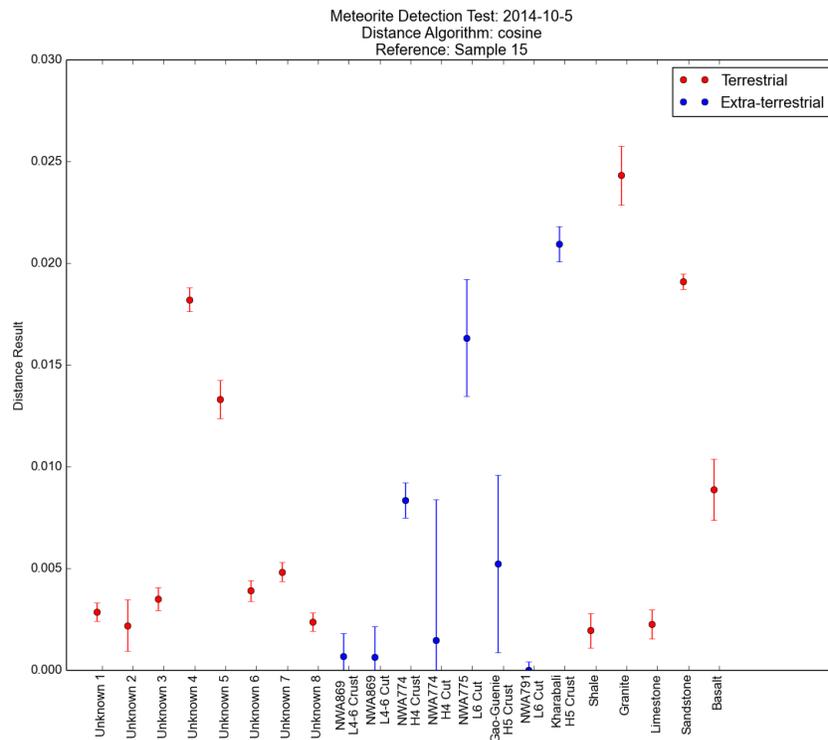
where $u \cdot v$ is the dot product of the two elements

- Euclidean distance

$$(18.7) \quad d(u, v) = \|u - v\|_2$$

In order to compare the performance of each of these a software program was written in the python language. This program is listed in Appendix J. The program operates by iterating through the previously collected spectra of the meteorite samples, for a given day, and setting each one as the reference. Once the reference is chosen, it is compared against the spectra of all of the samples for that day, of both terrestrial rocks and meteorites. This is done for each of the aforementioned distance functions and plotted, yielding 56 difference plots, one for each meteorite as a reference and each distance method. This process was performed with some differences found in the spectra of some samples over the different days. This is discussed further in section 34.2. The difference plots allow for this to be visualised with the addition of error bars. An example of one of these 56 distance plots is shown in Figure 18.1. This particular distance plot uses the cosine distance function and Sample 15 - NWA791, an L6 ordinary chondrite as the reference meteorite.

FIGURE 18.1. Distance Plot - Cosine Distance - Sample 15 as Reference



To determine the most suitable distance function to match the reference spectrum to the samples being analysed, each of these 56 plots were analysed for the most desirable true and false positive results. These results are detailed in subsection 34.3.

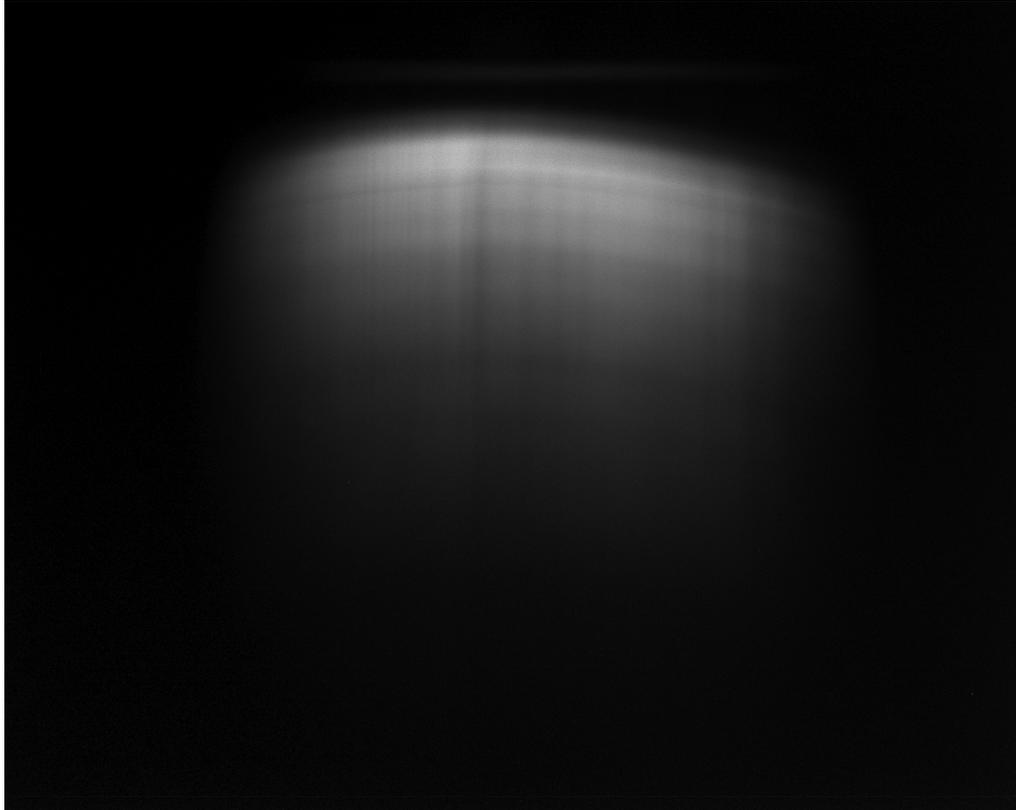
19. HYPER-SPECTRAL PRE-PROCESSING

19.1. Spectral Adjustment. If a reference spectrum was used that was captured from a different piece of equipment and/or under different lighting conditions, then it would be important to adjust the image as part of the pre-processing of the raw hyper-spectral data captured from the camera. This has not occurred in this project and is therefore not a necessary part of the image pre-processing.

19.2. Region of Interest. One of the operations that is applied to the image is defining a region of interest (ROI). This is the process of defining an area of pixels that is smaller than the full resolution of the image sensor of the camera, that can then be processed further. More specifically, for this project, before the region of interest was applied, pixels in the spatial dimension on the image were being captured that contained no data. This can be visualised by examining the black bands running down the sides of the raw hyper-spectral image shown in figure 19.1. The region of interest can be achieved at different points, either in software after the full resolution image has been captured or by defining an ROI on the image sensor of the camera. In this project, the region of interest operation has been performed on the camera by passing coordinates to the camera through the xipyAPI connector. These operations are shown in the program listing in appendices G, H and I. This has been done in order to allow for increased frame-rate of the camera. This operation was initially performed in software but was moved onto the camera and therefore limited the amount of data the image sensor has to retrieve in a given frame and limited the amount of data that was transmitted across the USB interface for a given frame.

Due to the lens distortion present in the raw hyper-spectral image, as discussed in sub-section 19.6, the captured data shown in this document has a region of interest that has been limited to a single spatial line at full spectral resolution. This method significantly reduces the spatial resolution of the camera but negates the effects of lens distortion.

FIGURE 19.1. Un-processed Hyper-spectral image captured from NWA869 L4-6 Chondrite (solar reflection)



19.3. Image Rotation. The rotation of the image is an operation that is performed in software. During this operation the image is rotated 180 degrees. This is performed so that when this information is entered into an array, the spectral dimension (the Y axis) will be ordered by ascending wavelength. This simplifies the visualisation of the data.

19.4. Camera Exposure Control. Before a hyper-spectral image is analysed, the exposure is set to achieve comparable spectra between the reference spectrum and the captured spectra or in the case of the initial spectral reference capture, between each of the captured spectra. For this application, it was decided to control the maximum pixel intensity within a window. This window is defined in Equation 19.1 and Equation 19.2.

$$(19.1) \quad \textit{Max Pixel Intensity} > 235$$

$$(19.2) \quad \textit{Saturated Pixel Count} < 10$$

The control is achieved by first recognising the direct proportional relationship that occurs between the exposure setting of the camera that was used in the second

prototype and pixel intensity. This method that exploits this relationship is shown in Equation 19.3.

$$(19.3) \quad \text{New Exposure} = \frac{\text{Desired Max Pixel Intensity}}{\text{Current Max Pixel Intensity}} \cdot \text{Current Exposure}$$

While Equation 19.3 is functional to control for maximum pixel intensity above a threshold, it is limited when pixels are saturated beyond a predefined acceptable level. This is due to the relationship between the saturated pixels light intensity and the light intensity of the non-saturated pixels being unknown. The exposure control handles this by dividing the exposure value by a set value until it is within acceptable limits.

The exposure control is implemented in the main program code on the computer as a python function.

19.5. Noise Reduction. Pre-processing can also be performed to remove unwanted noise from the raw images. In this project the noise created by “bad pixels” has been limited. this has been done by turning on the bad pixel correction function on the Ximea camera. Without this in place, multiple “bad” pixels are displayed incorrectly as having the maximum possible pixel value. Due to the nature of these pixels to remain consistently bad throughout all captured images, their effects on capturing and matching hyper-spectral images should be minimal. The bad pixels have been eliminated in this project in order to allow for a more realistic spectrum to be displayed.

19.6. Distortion Removal. As can be seen in figure 19.1, there is a warping of the image which is due to distortion introduced by the various optics in the hyper-spectral camera design. What this means is that the spectrum is slightly shifted to a different position at different point on the spatial axis. For example at the centre of the spatial dimension, the wavelength at a given point may be 800nm but the wavelength at the pixel the left may be 805nm with the error continuing to grow as the position from the centre increases.

For this project the distortion has not been removed as part of the pre-processing or at any stage. The alternate approach that has been taken has been to only capture hyper-spectral images for a single spatial pixel (and the entire spectral resolution). This method doesn’t include different spatial spectra with offset spectral dimensions simply because it only includes one spatial dimension. While this method allows for the collection and analysis of hyper-spectral images, it is limited by the range of spatial resolution that it can collect to analyse the spectrum for. For example, for a single still image, this method allows for the spectral analysis of a single spatial unit of resolution instead of the full resolution of 680. Although this will need to be resolved before such a system is used in a UAV for meteorite collection, it is more appropriate for the proof of concept covered in this project due to it’s simplicity.

20. METEORITE PRESENCE DETECTION

As shown in the example spectral matching plot of figure 3.10, different meteorite samples have different match results. In order to pass the maximum amount of real meteorites and reject the maximum amount of non-meteorites, it was necessary to determine an appropriate threshold for meteorite approval. In determining this, it

was necessary to decide whether it is more important to detect the location of every possible meteorite or to reject the false detection of non-meteorites.

21. TRIAL METEORITE SEARCH

The purpose of a trial meteorite search was to test the concept of aerial hyper-spectral meteorite detection in a controlled environment. This was done using the second prototype hyper-spectral camera, and known classified ordinary chondrite meteorite sample. This trial was not conducted with the use of an aircraft but rather using a simulated flight by moving the hyper-spectral camera across an area with known meteorites and known non-meteorites (colloquially referred to as meteor-wrongs). The area in question was in full sun on all occasions. This process that was used for tuning the hyper-spectral camera and algorithm design and was repeated several times to refine the tuning of the hyper-spectral camera and algorithm design.

22. REPORT LOCATION OF DETECTED METEORITES

Once a potential detection has been classified as a meteorite, the location will need to be reported so that the sample can be collected for further investigation. This means that the location information which is captured by the GNSS receiver, will need to be correlated to an adequate spatial resolution. Upon classification, the geographic coordinates are then reported as a meteorite location.

This exercise is managed by the main hyper-spectral camera python program listed in appendix I. It does this by creating and saving a log file which contains the following information for each frame:

- UTC time stamp
- Latitude
- Longitude
- Altitude above sea level (not ground level)
- Speed
- Distance to reference result

An example of this log file is shown in table 9.

TABLE 9. Results Log File Example

UTC Time stamp	Latitude (°)	Longitude (°)	Altitude (m)	Speed (km/h)	Distance to Reference Result
2014-10-24T23:54:24.000Z	-35.280832205	149.141945203	582.604	0.0	0.335541788601
2014-10-24T23:54:24.000Z	-35.280832205	149.141945203	582.604	0.0	0.336105788655
2014-10-24T23:54:25.000Z	-35.280832205	149.141945203	582.604	0.0	0.337192373934
2014-10-24T23:54:26.000Z	-35.280832205	149.141945203	582.604	0.0	0.002334593971
2014-10-24T23:54:27.000Z	-35.280832205	149.141945203	582.604	0.0	0.336037667251
2014-10-24T23:54:28.000Z	-35.280832205	149.141945203	582.604	0.0	0.336187912558

By analysing this log file, the location of meteorites can be determined and investigated further.

23. LOCATION CONFIRMATION

After the receipt of geographic coordinates, the location will have to be reached manually for collection of the sample. Once the sample has been collected, it can then be further analysed. The results of this process will be used as a measure of the effectiveness of the overall system.

24. PROTOTYPE UAV

The prototype UAV will be a smaller, cheaper, slower fixed wing craft than the final platform. The intention of this prototype is to tune the design and learn how to manually and autonomously fly a fixed wing UAV on a lower risk platform. This UAV is not intended to be fitted with a hyper-spectral camera. An example of an appropriate craft is based on the Bixler 2 airframe shown in Figure 24.1.

FIGURE 24.1. Prototype UAV



25. FINAL UAV

Drawing on the lessons learned from the prototype UAV, the larger, more costly and faster fixed wing UAV will be constructed. This UAV will also have a larger payload bay to fit greater quantities of batteries for longer flight time and also to fit the hyper-spectral camera. This could be the final UAV design that could be used to search for meteorites in later projects, building on the research performed in this project. An example of an appropriate airframe is the Skywalker X8, shown in Figure 25.1.

FIGURE 25.1. Final UAV: Unassembled (Pen on wing for reference size)



26. FINAL HYPER-SPECTRAL CAMERA

The final revision of the hyper-spectral camera will be one that is both functional and fit for operation in the final revision of the UAV. This will require all components to be secured so that they are not dislodged from place, on take-off, manoeuvring and landing. Care will also have to be taken to ensure that the vibration resistance of the system is adequate. This could be the subject of further work drawing on the research completed in this project.

27. FINAL METEORITE SEARCH

The final meteorite search will involve the final revision of the UAV equipped with the final revision of the hyper-spectral camera searching for meteorites in a location with an unknown quantity of them. Although this particular task is beyond the scope of this project, it could be performed as later work that builds upon the findings made as part of this research project.

Part 5. Resource Requirements

28. HARDWARE

28.1. Hyper-spectral Camera. The hyper-spectral camera is the piece of equipment that is used to capture the spectral information of an object and convert it into digital data that can be processed by a computer. It is one of the technical requirements to design and build this component.

28.1.1. *Optics.*

- Front End Focusing Optics

This is the lens that captures and focuses the light reflected from the sun off potential meteorites. The correct focal length will have to be chosen so that meteorites of the target size can be captured at adequate resolution and focus to classify. This will have to be tuned to the resolution of the image sensor and the altitude of the aircraft.

- Collimator

Collimating is the process of changing the focal distance of light so that it becomes infinite. For this project, collimation will occur only in the spatial dimension not in the spectral dimension. This requires the use of a fine slit followed by a lens placed at its focal length away from the slit.

- Dispersion prism

A dispersion prism is used to split the collimated white light into its component spectra. This is achieved by light passing between media of different refractive indexes causing different wavelengths of light to be refracted by different amounts.

- Camera focusing optics

Once the single line of spatial resolution has been split into its spectral components, this light will need to be refocused onto the image sensor of the camera with a lens. This will need to be tuned, along with the alignment of the image sensor and prism to produce the desired spectral range and to ensure that the available spectral resolution is captured by the image sensor.

28.1.2. *Video Camera.*

- Chrominance

The video camera image sensor has needed to be monochrome rather than colour. Colour image sensors generally use a Bayer filter to achieve a colour image. This involves the use of a filter in front of the image sensor that bins incoming light into either red, green or blue. The colour on the image sensor in this case is determined by its spatial position on one axis and the use of a three-bin colour filter would limit the ability of the sensor to detect many different spectral bands. Another factor is that image sensors will often be more sensitive to incident brightness if they are monochrome rather than colour.

- Resolution

The resolution of the image sensor determines both the spatial and spectral resolution of the system. The spatial resolution can be tuned by adjusting the altitude of the aircraft, however resolution of the image sensor and spectral range are the only variables that will determine the spectral resolution of the system.

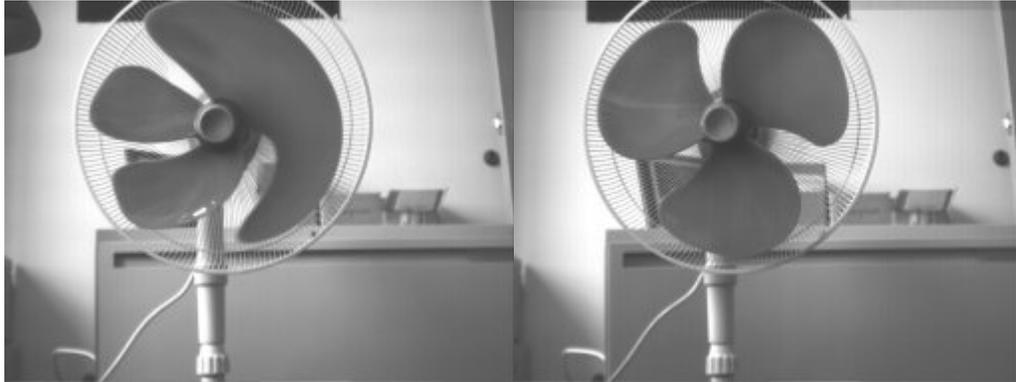
- Frame rate

The frame rate of the image sensor must be high enough (given a set altitude, speed and focal length of the front-end optics) to capture all of the data incident to it. If the images incident to the image sensor are changing more quickly than the image sensor is able to capture data then data will be lost and there will be sections of the search area that will not have been effectively covered.

- Shutter

A global shutter of an image sensor captures the light incident to the image sensor all at the same time for any given frame. A rolling shutter on the other hand will capture light for any given frame at different points in time for different parts of the image sensors resolution. If a rolling shutter was used for this application, there may have been either spectral or spatial distortion occurring because of the rapidly moving image. This could cause incorrect hyper-spectral data to enter the spectral matching algorithms and hinder the detection of meteorites. Due to this, a global shutter image sensor has been used. An example of motion blur occurring with a rolling shutter camera is shown in figure 28.1.

FIGURE 28.1. Rolling Shutter (left) compared to Global Shutter (right): Images from Point Grey Research[Point-Grey-Research, 2014]



- Computer Connection

In order to be able to transport the required quantity of data being produced by the hyper-spectral camera, the interface to the computer must be capable of an adequate band-width. This can only be determined once the bit-depth, resolution and frame rate of the image sensor has been chosen and can done using equation 28.1.

$$(28.1) \quad \textit{Band Width} = \textit{Bit Depth} \cdot \textit{Resolution} \cdot \textit{Frame Rate}$$

28.2. **GNSS.** In order to report the position of potential meteorites, a position signal has been used to log each spatial section of hyper-spectral data. A GNSS receiver such as a GPS has been used for this purpose.

28.3. Unmanned Aerial Vehicle. The UAV is the vehicle that carries the hyper-spectral camera and GNSS over the search area to collect hyper-spectral data. There are several characteristics of the vehicle that will need to be chosen to best suit the application.

28.3.1. *Wing Configuration.* There are different configurations of UAV that each have their own strengths and weaknesses. The most common configurations are:

- Fixed Wing
- Multi-rotor
- Helicopter

Fixed wing aircraft are the best suited of these three for travelling longer distances efficiently. They do lack the ability to take off vertically and to rapidly change direction however these are characteristics that will not be required for this project. Therefore a fixed wing configuration has been chosen.

28.3.2. *Flight time.* The UAV must have a long enough flight time to be able to cover an adequate search area and collect enough data before refuelling/recharging. The flight time will be a function of mainly the type of energy storage used (liquid fuel/batteries) and the weight of the craft.

28.3.3. *Energy Storage.* The decision between the use of batteries or liquid fuels for energy storage will contribute to the cost, reliability, construction complexity and flight time of the system. Liquid fuel systems produce a greater flight time but are generally more expensive, less reliable and more complex to construct. Due to the time and financing restraints on this project, battery storage will be used. At a time when the overall concept of hyper-spectral imaging from UAVs to detect meteorites has been proven, a liquid fuel based system could be used to extend the flight time of the UAV.

28.3.4. *Automation.* An autonomous UAV will be used due to the ability to pre-plan a flight mission and reduce human error.

28.3.5. *On-board devices.* An autonomous electric UAV will require the on-board devices shown in table 10.

TABLE 10. On-Board Devices

Device	Function
Motor Driver	Control the speed of the propeller
GNSS Receiver	Report location of UAV
Control Surface Servos	Move the control surfaces to steer the UAV
Servo Drivers	Control servos
Air-speed Sensor	Measure air-speed
Altitude Sensor	Measure Altitude
Acceleration Sensor	Measure Acceleration
Telemetry Radio	Report flight data to the ground station
Autopilot	Control the other devices for autonomous flight
Embedded Computer	Record video and location information

28.3.6. *Payload.* The UAV must be able to carry the on-board devices and hyper-spectral imaging equipment for the minimum duration required for a search mission.

28.4. **Meteorite Samples.** In order to show the effectiveness of the final system, having a search area with a known meteorite sample has been necessary. This has required the procurement of several meteorite samples. This has been done commercially through one of a number of internet based meteorite markets “The Meteorite Exchange” (<http://www.meteorites-for-sale.com/>). It has been necessary to establish the type of meteorites purchased as well as weathering characteristics and crust level. The samples have been chosen to reflect the target meteorite types.

29. SOFTWARE

29.1. **Mathematical Modelling.** Mathematical modelling software will be used for the following tasks:

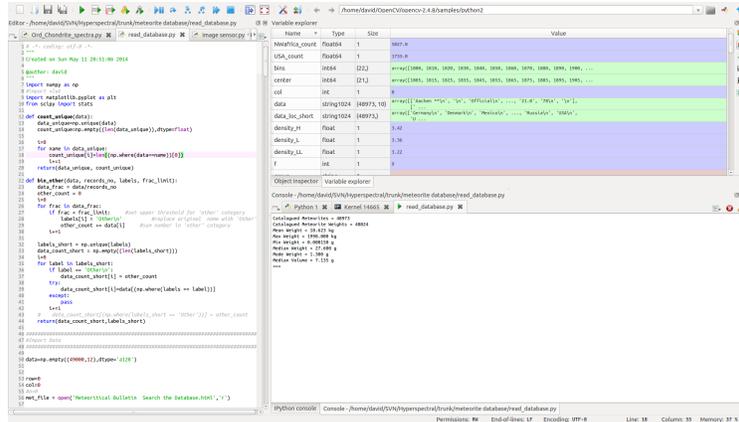
- Optics geometry calculations
- Statistical analysis and plotting of meteorite data
- Analysis and manipulation of meteorite reference spectra library
- Pre-processing of hyper-spectral data
- Spectral matching
- Location detection logging

The above tasks will be performed using Python 2.7 and programmed with Spyder IDE (Scientific PYthon Development EnviRonment) which is free open source software. A screen shot of the Spyder IDE is shown in figure 29.1. Additional Python libraries will be used to aid in the above tasks. The most noteworthy of these are shown in table 11.

TABLE 11. Python Libraries

Library	Function
NumPy	Arrays, matrices and high-level mathematics
SciPy	Scientific computing, statistics
Matplotlib	Plotting
Hyperspy	Hyper-spectral specific tools
OpenCV	Computer vision library for manipulating images

FIGURE 29.1. Spyder Screenshot



29.2. 3D Modelling. In order to physically model components of the hardware in 3D space a 3D CAD program has been used. Solidworks is the program that has been chosen for this purpose. 3D modelling will aid in the construction of the hyper-spectral camera housing and is especially useful when used to generate models to be 3D printed. 3D printing allows for mechanical prototypes to be made quickly and at low cost.

29.3. Databases. Having access to databases that contain pertinent information of meteorite statistics, spectral responses of target meteorites and spectral responses of potential background spectra, will be vital to the success of this project. These databases have been identified in table 12.

TABLE 12. Relevant Databases

Name	Maintained by	Function
Meteoritical Bulletin	The Meteoritical Society	Catalogue of all known meteorites
Gaffey Meteorite Spectra	NASA Planetary Data System	Spectra of samples of meteorites
USGS Digital Spectral Library	U.S. Geological Survey	Spectra of geographical features
Reference Solar Spectral Irradiance	National Renewable Energy Laboratory	Spectra of sunlight at ground level

29.4. Version Control. All files associated with this project are kept under version control using a cloud based Subversion (SVN) server. This allows for data backup and provides the ability to roll back changes if needed. It also allows for the generation of statistics on work completed by number of commits to the repository.

FIGURE 29.2. SVN Commits in the week of 26th of April



Part 6. Safety, The Law & Consequential Effects

30. LEGISLATION

30.1. UAVs - Australia. Civilian use of UAVs is a relatively new practice and the legal framework for this is still emerging in different jurisdictions around the world today. In Australia the government body responsible for the regulation of UAVs is the Civil Aviation Safety Authority (CASA) under the Civil Aviation Safety Regulations 1998 Part 101. As at June 2014, CASA discriminates between model aircraft and what it refers to as remotely piloted aircraft (RPA) on the basis of the application of the aircraft. If the aircraft is used for commercial, government or research purposes, then it is considered to be an RPA by CASA. Model aircraft on the other hand are used for sport and/or recreation. CASA states that “Unmanned aircraft activities are approved for operations over unpopulated areas up to 400 feet AGL (above ground level) (120 metres), or higher with special approvals” [CASA, 2014]. Unmanned aerial vehicle (UAV) controller’s certificates and an unmanned operator’s certificates (UOC) are only required if the RPA is operated for commercial gain[CASA, 2014].

Interestingly, in addition to safety, drones and UAVs also raise issues around privacy. In early 2014 the House of Representatives’ Standing Committee on Social Policy and Legal Affairs released a report “Eyes in the Sky” which made six recommendations in total which fall into two broad categories - safety and privacy. The Government is still considering its response to the recommendations [Woodley, 2014].

In the context of taking forward this project, the aforementioned certificates will not be required for Australian operations because of the absence of commercial gain. The regulations will also mean that UAV operations must be performed in unpopulated areas and at altitudes above ground level of no more than 120 meters. These restrictions will have implications on the design of the aircraft and imaging system, particularly with respect to tuning the optical focal length with the aircraft altitude and camera spatial resolution to the target object (potential meteorites) are as viewed from the air. The search area will also have to be carefully researched to avoid encroaching on populated areas and being non-compliant with the regulations.

30.2. Meteorites. There are different laws governing the ownership and collection of meteorites in different jurisdictions. For example, all meteorites that fall to earth in South Australia are the property of the state’s Museum Board [The_Parliament_of_South_Australia, 1976]. This is also the case in Western Australia [WA_Museum, 2013]. It is also illegal under the Commonwealth Protection of Movable Cultural Heritage Act (1986) to export any meteorite from Australia, without permission the authority for which currently rests with the Ministry of the Arts within the Attorney-General’s Department[of Legislative-Drafting and Attorney-General’s Department, Canberra, 1987]. It is also illegal to export meteorites from Canada, South Africa and Namibia without a permit [NHM, 2014].

In Antarctica meteorites are regularly collected by scientific bodies for scientific research and have been for some time[Harvey, 2003]. However at the fourth meeting of the Committee for Environmental Protection (which was established under the Antarctic Treaty), a resolution was adopted to urge parties to the Protocol on Environmental Protection to the Antarctic Treaty to take measures to ensure

the preservation of Antarctic meteorites “so that they are collected and curated according to accepted scientific standards, and are made available for scientific purposes”[Secretariate_of_the_Antarctic_Treaty, 2001].

These laws have implications for this project, particularly for the selection of a meteorite search site.

31. SAFETY

31.1. UAV Construction. The construction of the UAV has involved the use of glues and hand tools. The appropriate safety measures have been taken when using these tools. The workshop where the work is carried out has also be fitted with a fire extinguisher.

31.2. Lithium-Polymer Batteries. Lithium Polymer (Li-Po) batteries are preferential for use in UAVs due to their high specific energy, allowing greater flight times and/or payloads. They are however potentially dangerous under several conditions. Li-Po batteries can catch fire and explode if they are overcharged, over-discharged or subject to shock. Care must always be taken when dealing with Li-Po batteries especially when charging them. Li-Po specific chargers should always be used and the batteries should be placed in a fire proof bag when recharging. An example of a fire proof bag used for Li-Po batteries is shown in figure 31.1.

FIGURE 31.1. Lithium Polymer Fire-Proof Bag[Hobbyking, 2014]



31.3. Flight. UAVs can travel at high speeds and can have exposed propellers rotating at high rotational rates. This can expose risks from collision with a moving craft and collision with spinning propeller blades. The UAV should not be launched by hand and should be kept well clear of whilst it is armed. Landings and take-offs should far clear from people and where possible the operator(s) should have cover that can be used to take refuge from an approaching aircraft.

31.4. Eye Safety. Eyes can be damaged by the sun. Particular care has been taken to avoid exposing eyes to optics that reflect or focus the sun's light. Given that this project is dealing with spectra in the VIS-NIR range the spectrograph optics will need to be configured for this. That will mean that the prism will be

refracting non-visible infra-red light from the sun. For the purposes of safety, it has not been assumed that if the light can not be seen then it is not present.

31.5. Electricity. The electrical circuits present in this project will be of a low voltage. Short circuited wires at low voltage can cause dangerous levels of heating and be a fire danger. Where appropriate, circuit protection has been used.

31.6. Soldering. Soldering irons are useful tools in the construction of electronic circuits. For them to perform their utility they are required to reach temperatures over $300^{\circ}C$. Care has been taken to avoid exposure of skin to hot irons and to avoid fire risks by returning the iron to it's stand and clearing the surrounding area.

Part 7. Results

32. INTRODUCTION

The results that are contained within this part of this research project will determine the suitability of hyper-spectral imaging from an airborne vehicle to detect meteorites.

These results attempt to quantify the ability of hyper-spectral imaging to differentiate between Ordinary Chondrite meteorites and terrestrial background that may be encountered from an aerial vehicle. This is done in two ways. The first is in a theoretical sense, by analysing and comparing spectra of Ordinary Chondrite meteorites as well as spectra of other minerals and terrestrial background materials that have been collected by other parties under laboratory conditions. Secondly the second prototype of the hyper-spectral camera, that has been developed as part of this research project, is used to collect hyper-spectral images of materials from their reflection of sunlight. These materials include physical samples of Ordinary Chondrite meteorites as well as terrestrial rocks both identified and unidentified. The hyper-spectral images of these materials have then been analysed for their ability to differentiate between Ordinary Chondrite meteorites and terrestrial background rocks.

In both approaches scores have been given to the quality of the results as they relate to the application.

33. METEORITE DETECTION - LABORATORY SPECTROSCOPY

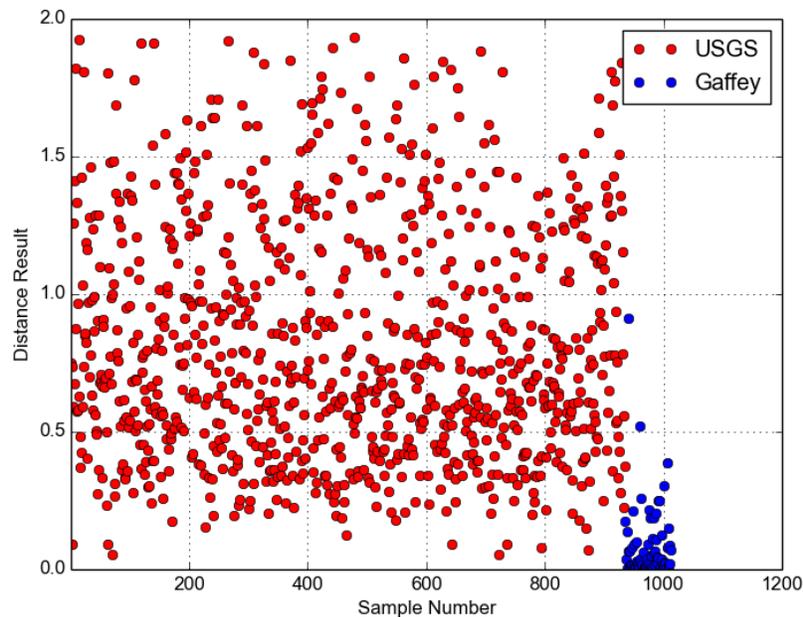
33.1. Context. The United States Geological Survey published their most recent digital spectral library in 2007. This is known as Version 06 and is a library of 1365 different materials[Clark et al., 2007], 935 of which have been considered in this project due to data being either fully or partially available in the band of 400 to 1000nm wavelength. The materials in this library have been included with remote sensing applications in mind and therefore contain the spectra of many materials that may be commonly encountered in remote sensing applications[Clark et al., 2007]. Some of these materials include vegetation such as trees, grass and flowers. A large number of minerals are included and also soils and snow from a variety of areas within the United States. The library also includes man made materials such as fibreglass.

This library provides many spectra that can be used as a general reference for background spectra that may be encountered when capturing aerial hyper-spectral images and is therefore useful in determining if the background spectra are differentiable from meteorite spectra. It is worth noting that spectra in the library were recorded in a laboratory environment using band appropriate spectrometers and do not reflect hyper-spectral camera image sensitivity or sunlight irradiance at different wavelengths. This differs from solar reflectance spectroscopy in that the reflectance of all wavelengths considered can be measured equally. In solar reflectance spectroscopy, there are significant troughs in the spectral irradiance that are attenuated from passing through the Earth's atmosphere (as shown in Figure 3.9). This means that spectral information that is available in the laboratory spectra is not necessarily available in solar reflectance spectra. This may lead to better performing results when using the laboratory spectra.

In this section of the project report, the Gaffey spectra of 77 different Ordinary Chondrite meteorites (as discussed in the literature review of Part 2 and sub-section 3.8) has been super-imposed onto the United States Geological Survey digital spectral library version 06. This approach has been performed in order to use a difference algorithm to attempt to differentiate the Gaffey meteorite spectra from the USGS background spectra. The success of this process will add weight to the plausibility of using aerial hyper-spectral images to detect ordinary chondrites.

33.2. Distance Results. From the 77 Gaffey Ordinary Chondrite spectra, each one was individually used as a reference to be matched against the others with a correlation distance function. The three best performing of these are shown here. Figure 33.1 shows the first of these three with the red points being the background spectra from the USGS spectral library and the blue points representing the Gaffey Ordinary Chondrite meteorite spectra. The lower the correlation distance result (represented by the Y axis) the closer each spectrum is to the reference spectrum. This chart can then be used to differentiate between meteorites and background materials by drawing a horizontal threshold line whereby every marker above it is rejected as not being a meteorite and every marker below it is accepted as being a meteorite.

FIGURE 33.1. USGS and Gaffey Spectra, Correlation Distance Results from Gaffey Reference 34



The performance of each of these three reference spectra can be measured in different ways depending on the aim and the willingness to accept false positives or false negatives. This is done by placing the threshold at different levels of distance

result. As can be seen in the results table of Table 33.1, three different threshold levels have been chosen as options. These have been chosen, for all three of the best performing reference spectra, as the level that produces no false positives (accepts no non-meteorites as meteorites), the level that produces no false negatives (detects all meteorites as meteorites) and a level in between these values that reaches a balance in these results to maximise the meteorites accepted and the background material rejected.

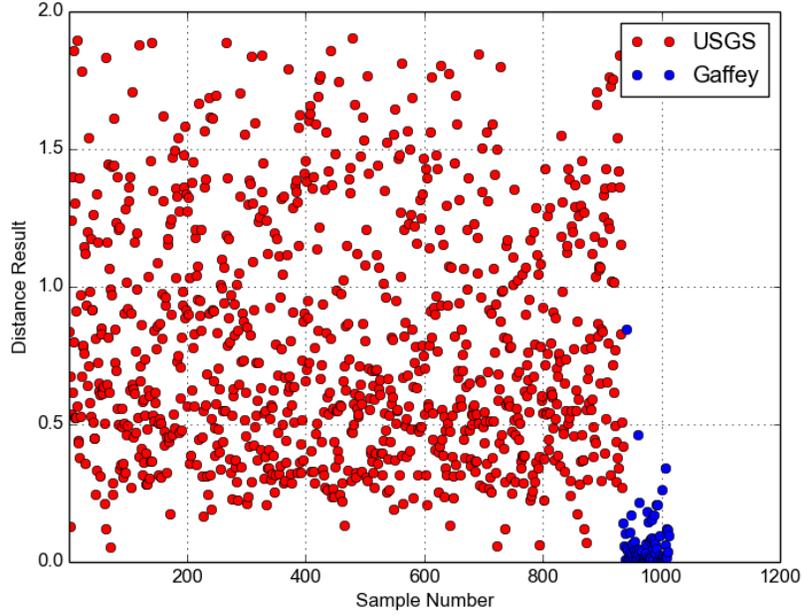
The results in Table 33.1 show that if all of the background spectra is rejected then 61 per cent of the 77 meteorites will be accepted. Including all of the meteorites however leads to only 34 per cent of the background being rejected. The compromise between these two rejects over 99 per cent of the background spectra and accepts 79 per cent of the meteorites as meteorites.

TABLE 13. USGS and Gaffey Spectra Differentiation, Gaffey Reference 34

Threshold	False Positives	False Negatives	True Positives	True Negatives	Meteorites Accepted	Terrestrial Rejected
0.052	0	30	47	934	61.0%	100.0%
0.120	8	16	61	927	79.2%	99.1%
0.913	615	0	77	320	100.0%	34.2%

The plot in Figure 33.2 shows the distance results for the second of the three best performing Gaffey meteorite reference spectra.

FIGURE 33.2. USGS and Gaffey Spectra, Correlation Distance Results from Gaffey Reference 35



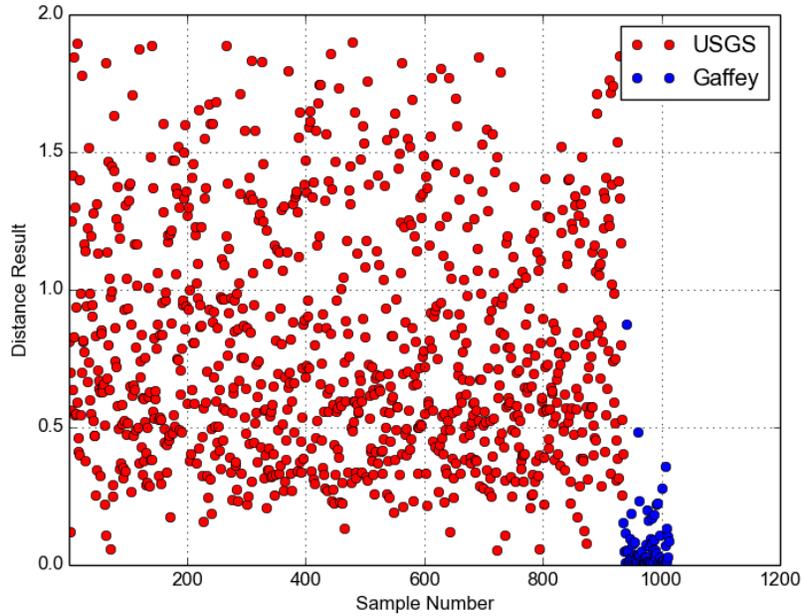
The results table of Table 14 produces very similar results to Table 13 although performs slightly better in both meteorite acceptance and background rejection in at all three threshold levels.

TABLE 14. USGS and Gaffey Spectra Differentiation, Gaffey Reference 35

Threshold	False Positives	False Negatives	True Positives	True Negatives	Meteorites Accepted	Terrestrial Rejected
0.052	0	29	48	934	62.3%	100.0%
0.120	5	14	63	930	81.8%	99.5%
0.848	583	0	77	352	100.0%	37.6%

The final of the three distance plots is shown in Figure 33.3.

FIGURE 33.3. USGS and Gaffey Spectra, Correlation Distance Results from Gaffey Reference 70



The results table from the final of the three best performing references is shown in Table 15. Using this reference provides the best meteorite acceptance results when all background is rejected but is out performed by the previous reference at the balanced threshold and for background elimination with the balanced threshold.

TABLE 15. USGS and Gaffey Spectra Differentiation, Gaffey Reference 70

Threshold	False Positives	False Negatives	True Positives	True Negatives	Meteorites Accepted	Terrestrial Rejected
0.055	0	27	50	934	64.9%	100.0%
0.120	7	14	63	923	81.8%	99.3%
0.875	594	0	77	341	100.0%	36.5%

Out of all of these the most desirable model to be used for the purposes of aerial hyper-spectral imaging for meteorite detection is the balanced threshold of 0.12 from reference spectrum 35 as shown in Table 14. This provides a very broad rejection of background materials (99.5 per cent) and a broad acceptance of meteorites (81.8 per cent). The balance of meteorite acceptance to background rejection is preferred to be skewed in the favour of increasing background rejection simply due to the logistical difficulties that could be involved in collecting potential samples after they have been identified.

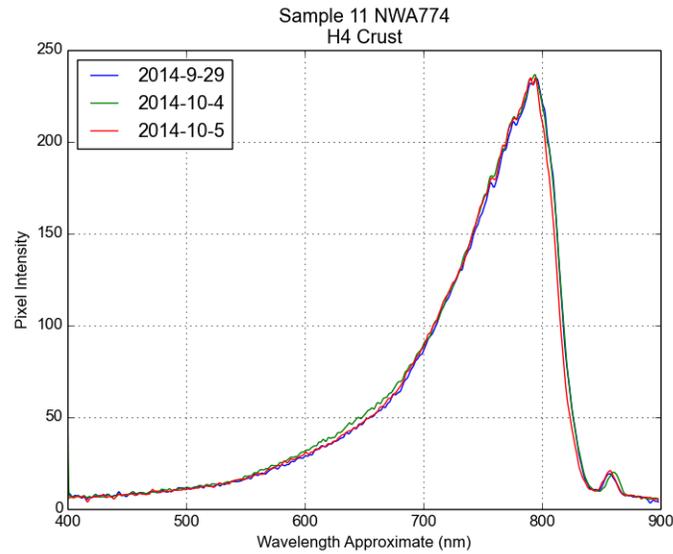
These results show that without the influences of hyper-spectral camera image sensor sensitivity and the varying spectrum of sunlight after being filtered by the atmosphere, it is theoretically possible to differentiate the spectra of Ordinary Chondrites from other minerals and background commonly encountered in aerial hyper-spectral imaging.

34. METEORITE DETECTION - SOLAR REFLECTANCE SPECTRA

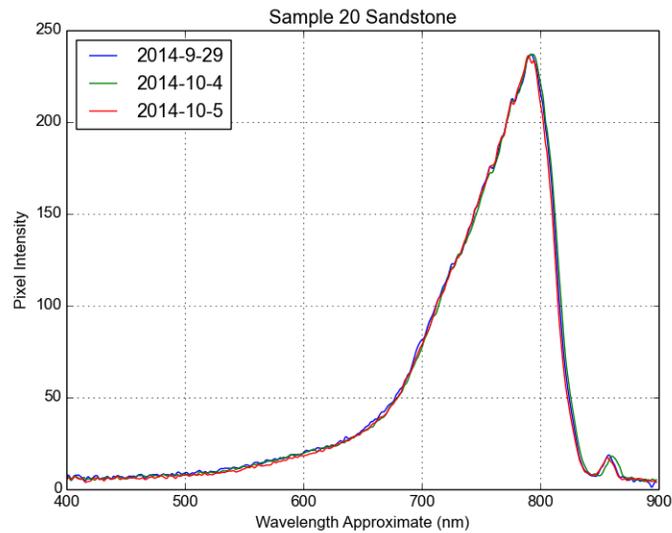
34.1. Context. To practically apply the theoretical work performed on the laboratory acquired spectra of Section 33 similar original spectra were captured from samples of Ordinary Chondrite meteorites and terrestrial rocks using the hyper-spectral camera developed as part of this project. These spectra were captured using reflectance of direct sunlight and were not adjusted for image sensor sensitivity. This means that unlike the results of Section 33, the spectra captured in this section are not the “true” reflectance spectra of the materials but are the reflectance of the spectrum of sunlight that is present after atmospheric filtering and sensed at different levels for different wavelengths based on the sensitivity of the image sensor. This is an important difference in aerial hyper-spectral imaging because the only light source available is sunlight.

34.2. Error in Measured Spectra. Sample spectra were collected at the same time in the same conditions on three different days in Canberra, Australian Capital Territory. Many of the recorded spectra of the same sample were found to be very similar, although not completely identical, across the different days, as in the Figure 34.1, which shows the recorded spectra for the crust side of the H4 meteorite NWA774 and Sample 20 - sandstone.

FIGURE 34.1. Consistent Spectral Measurements



(A) Sample 11 - NWA774 H4 Crust Spectral Reflectance Measurements

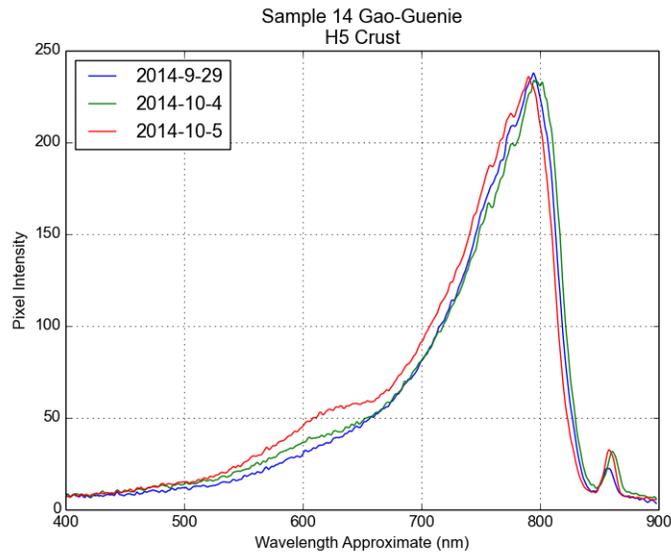


(B) Sample 20 - Sandstone

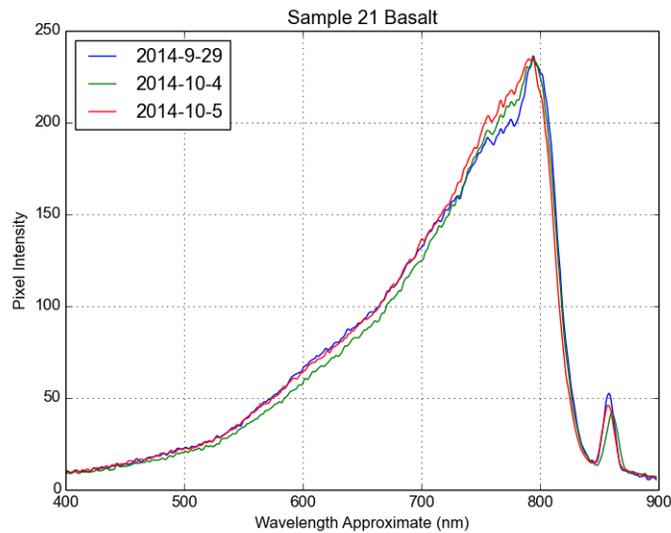
Some of the samples yielded spectral measurements that were significantly different over the three different days. An example of this is in Figure 34.2 which shows the three recorded spectra of sample 14, an H5 Gao-Guinea meteorite and

the results for Sample 21 - Basalt. A discussion of why this may have occurred is detailed below.

FIGURE 34.2. Inconsistent Spectral Measurements



(A) Sample 14 - Gao-Guinea H5 Crust Spectral Reflectance Measurements



(B) Sample 21 - Basalt Spectral Reflectance Measurements

Although the exact reason for these errors in measurements is not precisely known, it is possible that measurements were taken from slightly different positions

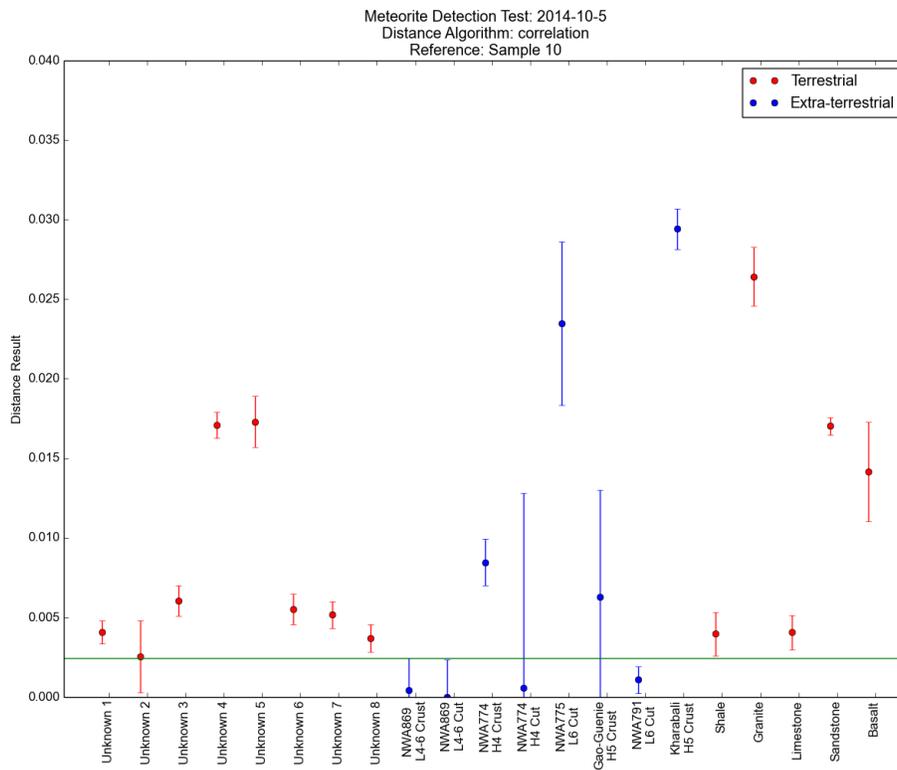
on the face of the sample which could produce different reflectance spectra due to differences in the sample across the face, possibly in some circumstances caused by weathering.

When generating the distance plots that are shown later in this document, error bars have been used to visualise these inconsistencies across the different days. This is done by measuring the maximum distance between the most different two of the three spectra of a sample and adding this value as an error both above and below the distance measurement. This helps to address the issue of inconsistent measurements by providing a visual representation of the error and allowing for evaluations to be made around not just the centre measurement of a data point but the extremes of its outlying results.

34.3. Distance Results. Seven distance functions were used independently with each of the eight meteorite samples as a reference to measure the difference between the spectra of the samples and the spectrum of the reference. This produced 56 distance plots in total. These were each qualified based on a high number true positives (meteorites that have been correctly identified as meteorites) and a low number of false positives (terrestrial rocks that have been incorrectly identified as meteorites).

The highest performing combination of these was found to be the Correlation Distance algorithm using sample 10, which is the cut face of the L4-6 meteorite, NWA 869. The distance plot using these parameters is shown in Figure 34.3.

FIGURE 34.3. Distance Plot - Correlation Distance - Sample 10 as Reference



By choosing a threshold of 0.00246 (which is shown in Figure 34.3 as a green horizontal line), the meteorite detection results from Table 16 were able to be generated. Taking the error of measurements on different days into account, there are three samples of meteorite that should always be detectable. There is a further one that's centre is below the threshold and one further still that has part of it's error falling below the threshold. This is a total of five of the eight meteorite samples that are possibly detectable with this method. Table 16 also shows one terrestrial sample (Unknown 2) that has part of it's error falling below the threshold. This means that out of there is one sample of terrestrial rock that it is possible will be

detected as a meteorite (a false positive) and 13 terrestrial rock samples that will be rejected as not being meteorites.

TABLE 16. Detection Results - Correlation Distance - Sample 10 as Reference

Threshold = 0.00246

Meteorite Total Error Below Threshold	Meteorite Centre Below Threshold	Meteorite Partial Error below Threshold	Terrestrial Total Error Below Threshold	Terrestrial Centre Below Threshold	Terrestrial Partial Error Below Threshold
3	4	5	0	0	1

If this results are simplified to discount the error bars, it becomes simpler to directly compare the results of the data practically collected with the hyper-spectral camera developed for this project, to the performance results given in Section 33. These results can be seen in Table 17.

TABLE 17. Detection Results - Correlation Distance - Sample 10 as Reference

Threshold	False Positives	False Negatives	True Positives	True Negatives	Meteorites Accepted	Terrestrial Rejected
0.00246	0	4	4	13	50%	100%
0.032	13	0	8	0	100%	0%

35. AERIAL ADVANTAGE

The current methods that are used to detect meteorites are overwhelmingly land-based as explored in Part 2. Even attempts to use more “technological” approaches have tended to be focussed on improving methods of ground searches. However, aerial search methods offer considerable advantages particularly given the rugged and remote terrain where meteorite searches tend to be focussed. Low cost modern civilian fixed-wing unmanned aerial vehicles are not limited by harsh terrain in the same way that ground based vehicles are and can also be scaled up with comparatively lower costs and human resources.

36. SUMMARY OF RESULTS

When compared to the results given in Section 33, the practical system can be seen to be of a lower performance. This is to be expected because of the reduced levels of some bands of the reflectance spectrum that results from using the sun as a reflectance source rather than a broad band controlled light source.

In spite of the lower performance of the practical system relative to the spectra collected by other parties in Section 33, the system was able to correctly classify 37.5 per cent of meteorites as meteorites at all levels of measured error. It was also able to correctly classify a further 12.5 per cent of the total population of meteorites

examined for greater than half of their error values and a further 12.5 per cent for less than half of their error values. This totals to be a worst case meteorite detection rate of 37.5 per cent and a best case of 62.5 per cent. For the chosen threshold there was also only one terrestrial rock that had any part of its error levels falling below this threshold and in that case it was less than half. This leads to a terrestrial rock rejection rate at worst case of 92.3 per cent and of best case at 100 per cent.

Assuming that these results correlate to what would be found in the field, aerial searches for meteorites could expect to result in a very low rate of false positives. This would mean that there would be a low chance of a report of a meteorite that in fact was not a meteorite which would reduce the amount of collection expeditions that yielded no meteorites thus resulting in a more efficient overall meteorite collection system than what would have otherwise been the case.

Part 8. Time-line

The project planning of this project was scheduled to key deliverables referred to as milestones. These have all been met as per the status section of the table below.

TABLE 18. Project Milestones

Milestone	Completion Date	Status
Project Proposal	12th March 2014	Complete
Project Specification	19th March 2014	Complete
Spectrograph Prototype Built	22nd March 2014	Complete
UAV Prototype Built	25th March 2014	Complete
Meteorite Statistics Analysed	28th May 2014	Complete
Target Meteorite Type(s) Chosen	29th May 2014	Complete
Reference Spectra Compiled	30th May 2014	Complete
Literature Review Draft Complete	1st June 2014	Complete
Preliminary Report	4th of June 2014	Complete
Hyper-Spectral Camera First Prototype Built	8th of June 2014	Complete
Hyper-Spectral Camera Geometry Determined	16th of June 2014	Complete
Hyper-Spectral Camera Second Prototype Built	19th July 2014	Complete
Target Meteorite Samples Procured	22nd of July 2014	Complete
Spectral Matching Technique Chosen	7th of September 2014	Complete
Detection Algorithms Complete	20th September 2014	Complete
Final Dissertation Complete	30th of October 2014	Complete

Part 9. Key Outcomes

A low cost hyper-spectral line scan camera has been built and has been found to be functional after going through several prototype stages. Software has been written that captures, processes and analyses hyper-spectral images from the camera. This software also measures the correlation between the captured spectrum and a pre-defined target and will map global coordinates to the result of the match.

Part 10. Conclusions

Hyper-spectral imaging can be used to differentiate between materials and has been applied to meteorites specifically.

After comparing the laboratory collected “Gaffey” spectra of Ordinary Chondrites to the laboratory collected United States Geological Survey spectra of terrestrial background materials, it has been found that it is possible to differentiate between most Ordinary Chondrites and other minerals including commonly encountered materials viewed from aerial imaging. This is possible with very low rates of false positives as noted in the results of Section 33.

From the small group of samples of Ordinary Chondrite meteorites and terrestrial rocks that have been acquired and studied as part of this research project, it has been found and detailed in Section 34, that it is largely possible to differentiate between the two groups using a low cost, 3D printed body, hyper-spectral camera with solar reflections, to a degree where it is plausible to use this approach for aerial identification.

After researching current methods that are used to detect and collect meteorites, it has been concluded that aerial meteorite detection is more efficient than methods currently used and theoretical investigations show that this process should be achievable from a UAV as noted in Section 35.

The ability of Hyper-spectral imaging to differentiate between Ordinary Chondrite meteorites and terrestrial materials has the potential to allow for the broadening of the number of collection locations, including to locations that are more easily accessible than those that are commonly used at present. This in itself has the potential to vastly increase the collection rates of meteorites.

The use of aerial vehicles allows for detection to occur over rough terrain and more specifically the use of UAVs allows for the automation of the meteorite detection process which has the potential to allow for detection to be scaled upwards at a comparatively lower cost than currently employed methods.

Part 11. Further Work

There are multiple packages of additional work that could build upon the outcomes achieved in this research project.

Several aspects of the hyper-spectral camera prototype could be improved to enhance its effectiveness at detecting meteorites. By adding software that compensates for lens distortion, the entire spatial range of the camera would be made available to be analysed. This would allow for areas to be analysed more quickly and/or more thoroughly. The software could also be tuned to allow for a higher camera frame rate. At present this is limited by the speed in which the image processing is performed and not by the limitations of the image sensor or the camera interface.

The development of a fixed wing UAV platform is another step that could be taken to further this project. The UAV would have to be capable of travelling at the speeds and altitudes that have been identified in this research project and be capable of carrying the hyper-spectral camera and companion computer. Additional work to integrate the camera system to a UAV so that the technology could be practically applied would be a valuable addition to the project.

Once the aforementioned tasks are completed, the system could be used to collect meteorites practically.

This project has focused on the collection of the Ordinary Chondrites meteorite group only. Further research could be performed to analyse the effectiveness of this system to detect other types of meteorite, expanding the population that could be collected.

REFERENCES

- Aerolite. Meteorites for sale - buy a genuine meteorite, 2014. URL <http://www.aerolite.org/meteorites-for-sale.htm>.
- ANSMET. The antarctic search for meteorites -, 2014. URL <http://artscilabs.case.edu/ansmet/page/3/>.
- D.S. Apostolopoulos, L. Pedersen, B.N. Shamah, K. Shillcutt, M.D. Wagner, and W.L. Whittaker. Robotic antarctic meteorite search: outcomes. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 4, pages 4174–4179 vol.4, 2001. doi: 10.1109/ROBOT.2001.933270.
- A. M. Baldridge, S. J. Hook, C. I. Grove, and G. Rivera. The ASTER spectral library version 2.0. *Remote Sensing of Environment*, 113(4):711–715, 2009. ISSN 0034-4257. doi: <http://dx.doi.org/10.1016/j.rse.2008.11.007>. URL <http://www.sciencedirect.com/science/article/pii/S0034425708003441>.
- Bathurst_Observatory. mets, 2014. URL <http://www.bathurstobservatory.com.au/mets.htm>.
- Mark Berman. Hyperspectral imaging for tablet inspection, April 2005.
- Mark Berman. Hyperspectral imaging for tissue analysis, November 2007.
- A.W.R. Bevan. Australian meteorites, 1992.
- John David Blom. Unmanned aerial systems: A historical perspective. *Combat Studies Institute Press US Army Combined Arms Center*, page 108, September 2010. ISSN 978-0-9823283-0-9.
- R. Calderón, J. A. Navas-Cortés, C. Lucena, and P. J. Zarco-Tejada. High-resolution airborne hyperspectral and thermal imagery for early detection of verticillium wilt of olive using fluorescence, temperature and narrow-band spectral indices. *Remote Sensing of Environment*, 139(0):231 – 245, 2013. ISSN 0034-4257. doi: <http://dx.doi.org/10.1016/j.rse.2013.07.031>. URL <http://www.sciencedirect.com/science/article/pii/S0034425713002435>.
- CASA. Civil aviation safety authority - model aircraft and RPA, 2014. URL http://www.casa.gov.au/scripts/nc.dll?WCMS:STANDARD::pc=PC_100375.
- Chein-I Chang. *Hyperspectral Imaging: Techniques for Spectral Detection and Classification - Chein-I Chang*, volume 1. Kluwer Academic, 2003. URL http://books.google.com.au/books?hl=en&lr=&id=JhBbXwFaA6sC&oi=fnd&pg=PA1&dq=hyperspectral+imaging+techniques+chang&ots=r1dIw_J1sU&sig=pYRd0vNRShzyvq80xiRUVhVdZTE#v=onepage&q=hyperspectral%20imaging%20techniques%20chang&f=false.
- R.N. Clark, G.A. Swayze, R. Wise, E. Livo, T Hoefen, R Kokaly, and S.J. Sutley. USGS digital spectral library splib06a: U.s. geological survey, digital data series 231, 2007. URL <http://speclab.cr.usgs.gov/spectral.lib06/ds231/index.html>.
- Edward A. CLOUTIS, Paul S. HARDERSEN, David L. BISH, Daniel T. BALLEY, Michael J. GAFFEY, and Michael A. CRAIG. Reflectance spectra of iron meteorites: Implications for spectral identification of their parent bodies. *Meteoritics & Planetary Science*, 45(2):304–332, 2010. ISSN 1945-5100. doi: 10.1111/j.1945-5100.2010.01033.x. URL <http://dx.doi.org/10.1111/j.1945-5100.2010.01033.x>.
- G. J. Consolmagno, D. T. Britt, and R. J. Macke. The significance of meteorite density and porosity. *Chemie der Erde - Geochemistry*, 68(1):1 – 29, 2008. ISSN 0009-2819. doi: <http://dx.doi.org/10.1016/j.chemer.2008.01.003>. URL <http://>

- [//www.sciencedirect.com/science/article/pii/S0009281908000044](http://www.sciencedirect.com/science/article/pii/S0009281908000044).
- Nick A. Drake, Steve Mackin, and Jeff J. Settle. Mapping vegetation, soils, and geology in semiarid shrublands using spectral matching and mixture modeling of {SWIR} {AVIRIS} imagery. *Remote Sensing of Environment*, 68(1):12 – 25, 1999. ISSN 0034-4257. doi: [http://dx.doi.org/10.1016/S0034-4257\(98\)00097-2](http://dx.doi.org/10.1016/S0034-4257(98)00097-2). URL <http://www.sciencedirect.com/science/article/pii/S0034425798000972>.
- Edmund-Optics. Compact VIS-NIR fixed focal length lenses | edmund optics, 2014. URL <http://www.edmundoptics.com/imaging/imaging-lenses/techspec-lenses/compact-vis-nir-fixed-focal-length-lenses/3338>.
- Michael J. Gaffey. Spectral reflectance characteristics of the meteorite classes. *Journal of Geophysical Research*, 81(5):905–920, 1976. ISSN 2156-2202. doi: [10.1029/JB081i005p00905](http://dx.doi.org/10.1029/JB081i005p00905). URL <http://dx.doi.org/10.1029/JB081i005p00905>.
- Michael J. Gaffey. Meteorite spectra. EAR-a-3-RDR-METEORITE-SPECTRA-v2.0. NASA planetary data system, 2001. URL <http://sbn.psi.edu/pds/resource/gaffey.html>.
- Jim Garamone. Defense.gov news article: From u.s. civil war to afghanistan: A short history of UAVs, April 2002. URL <http://www.defense.gov/news/newsarticle.aspx?id=44164>.
- Ralph Harvey. The origin and significance of antarctic meteorites. *Chemie der Erde - Geochemistry*, 63(2):93 – 147, 2003. ISSN 0009-2819. doi: <http://dx.doi.org/10.1078/0009-2819-00031>. URL <http://www.sciencedirect.com/science/article/pii/S0009281904700215>.
- Hobbyking. Lithium polymer charge pack 18x22cm sack, 2014. URL http://www.hobbyking.com/hobbyking/store/__4134__lithium_polymer_charge_pack_18x22cm_sack.html.
- Imperial_College_London. Searching for meteorites, 2014. URL <http://www3.imperial.ac.uk/earthscienceandengineering/research/iarc/collection/searchingformeteorites>.
- Torrence V. Johnson and Fraser P. Fanale. Optical properties of carbonaceous chondrites and their relationship to asteroids. *Journal of Geophysical Research*, 78(35):8507–8518, 1973. ISSN 2156-2202. doi: [10.1029/JB078i035p08507](http://dx.doi.org/10.1029/JB078i035p08507). URL <http://dx.doi.org/10.1029/JB078i035p08507>.
- U. Kang, Martial Hebert, and Soonyong Park. Fast and scalable approximate spectral graph matching for correspondence problems. *Information Sciences*, 220(0):306 – 318, 2013. ISSN 0020-0255. doi: <http://dx.doi.org/10.1016/j.ins.2012.07.008>. URL <http://www.sciencedirect.com/science/article/pii/S0020025512004720>. Online Fuzzy Machine Learning and Data Mining.
- Raymond F Kokaly, Trude V.V. King, and Todd M Hoefen. Mapping the distribution of materials in hyperspectral data using the USGS material identification and characterization algorithm (MICA). *Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International*, July 2011. ISSN 2153-6996.
- A. Senthil Kumar, V. Keerthi, A. S. Manjunath, Harald van der Werff, and Freek van der Meer. Hyperspectral image classification by a variable interval spectral average and spectral curve matching combined algorithm. *International Journal of Applied Earth Observation and Geoinformation*, 12(4):261 – 269, 2010. ISSN 0303-2434. doi: <http://dx.doi.org/10.1016/j.jag.2010.03.004>. URL <http://www.sciencedirect.com/science/article/pii/S0303243410000310>.

- Michael J Logan, Julio Chu, Mark A Motter, Dennis L Carter, Michael Ol, and Cale Zeune. Small UAV research and evolution in long endurance electric powered vehicles. In *AIAA Infotech@ Aerospace 2007 Conference and Exhibit, AIAA Paper*, volume 2730, pages 7–10, 2007.
- Stanley G. Love and Ralph P. Harvey. Crew autonomy for deep space exploration: Lessons from the antarctic search for meteorites. *Acta Astronautica*, 94(1):83 – 92, 2014. ISSN 0094-5765. doi: <http://dx.doi.org/10.1016/j.actaastro.2013.08.001>. URL <http://www.sciencedirect.com/science/article/pii/S0094576513002968>.
- Baerbel Lucchitta, Jo-ann Bowell, Kathleen Edwards, Eric Eliason, and Holly Ferguson. Multispectral landsat images of antarctica. *U.S. Geological Survey Bulletin*, 1696, 1987.
- Austin Mardon. *The use of geographic remote sensing, mapping and aerial photography to aid in the recovery of blue ice surficial meteorites in Antarctica*. 2009. ISBN 978-1-897472-35-4.
- Meteorites-Australia. Meteorites for sale - meteorites australia, 2014. URL <http://www.meteorites.com.au/sale.html>.
- Meteorites-For-Sale. Meteorites for sale, 2014. URL <http://www.meteorites-for-sale.com/meteorites-for-sale.html>.
- Gary Mortimer. DHL drone delivers medicine across the rhine | sUAS news, January 2014. URL <http://www.suasnews.com/2014/01/26865/dhl-drone-delivers-medicine-across-the-rhine/>.
- NHM. Meteorite FAQs | natural history museum, 2014. URL <http://www.nhm.ac.uk/nature-online/space/meteorites-dust/meteorite-faq/>.
- NIPR. National institute of polar research, 2014. URL <http://www.nipr.ac.jp/english/>.
- Office of Legislative-Drafting and Attorney-General’s Department, Canberra. Protection of movable cultural heritage regulations 1987 statutory rules 1987 no. 149 as amended made under the protection of movable cultural heritage act 1986, 1987. URL <http://www.comlaw.gov.au/Details/F2004C00105>.
- S. Padma and S. Sanjeevi. Jeffries matusita based mixed-measure for improved spectral matching in hyperspectral image analysis. *International Journal of Applied Earth Observation and Geoinformation*, 32(0):138 – 151, 2014. ISSN 0303-2434. doi: <http://dx.doi.org/10.1016/j.jag.2014.04.001>. URL <http://www.sciencedirect.com/science/article/pii/S0303243414000853>.
- L. Pedersen, M. Wagner, D. Apostolopoulos, and W.R. Whittaker. Autonomous robotic meteorite identification in antarctica. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 4, pages 4158–4165 vol.4, 2001. doi: 10.1109/ROBOT.2001.933268.
- Hanna Pentikäinen, Antti Penttilä, Karri Muinonen, and Jouni Peltoniemi. Spectroscopic investigations of meteorites. *Journal of Quantitative Spectroscopy and Radiative Transfer*, (0):–, 2014. ISSN 0022-4073. doi: <http://dx.doi.org/10.1016/j.jqsrt.2014.02.007>. URL <http://www.sciencedirect.com/science/article/pii/S0022407314000648>.
- Point-Grey-Research. Key differences between rolling shutter and frame (global) shutter., 2014. URL <http://www.ptgrey.com/support/kb/index.asp?a=4&q=115>.

- RMIT. Unmanned aircraft systems (UAS) - RMIT university, 2014. URL <http://www.rmit.edu.au/research/institutes/platformtechnologies/uav>.
- Secretariate_of_the_Antarctic_Treaty. Resolution 3 (2001) ATCM XXIV CEP IV, st. petersburg, July 2001.
- Randall B Smith. Introduction to hyperspectral imaging, May 2012.
- Smithsonian. Mineral sciences - meteorite collection, 2014. URL <http://mineralsciences.si.edu/collections/meteorites.htm>.
- The-Meteorite-Market. The meteorite market, 2014. URL <http://www.meteoritemarket.com/>.
- The-Meteoritical-Society. Meteoritical bulletin, May 2014. URL <http://www.lpi.usra.edu/meteor/metbull.php>.
- The_Parliament_of_South_Australia. South australian museum act 1976 1.2.2010 to 11.5.2011, 1976.
- UAV-Robotics-Australia. Product information - x8 skywalker flying wing (FPV & UAV), 2014. URL <http://www.uavrobotics.com.au/skywalker-flying-wing-p-129.html>.
- Dehant Véronique, Breuer Doris, Claeys Philippe, Debaille Vinciane, De Keyser Johan, Javaux Emmanuelle, Goderis Steven, Karatekin Özgür, Spohn Tilman, Vandaele Ann Carine, Vanhaecke Frank, Van Hoolst Tim, and Wilquet Valérie. From meteorites to evolution and habitability of planets. *Planetary and Space Science*, 72(1):3 – 17, 2012. ISSN 0032-0633. doi: <http://dx.doi.org/10.1016/j.pss.2012.05.018>. URL <http://www.sciencedirect.com/science/article/pii/S0032063312001316>. Mars Habitability.
- Maidier Vidal and José Manuel Amigo. Pre-processing of hyperspectral images. essential steps before image analysis. *Chemometrics and Intelligent Laboratory Systems*, 117(0):138 – 148, 2012. ISSN 0169-7439. doi: <http://dx.doi.org/10.1016/j.chemolab.2012.05.009>. URL <http://www.sciencedirect.com/science/article/pii/S0169743912001220>. Special Issue Section: Selected Papers from the 1st African-European Conference on Chemometrics, Rabat, Morocco, September 2010 Special Issue Section: Preprocessing methods Special Issue Section: Spectroscopic imaging.
- WA_Museum. Meteorite collection | western australian museum, 2013. URL <http://museum.wa.gov.au/research/collections/cosmochemistry/meteorite-collection>.
- WA_Museum. Meteorites from the nullarbor region | western australian museum, 2014. URL <http://museum.wa.gov.au/research/collections/earth-and-planetary-sciences/meteorite-collection/meteorites-nullarbor-region>.
- M Wiesberg, T McCoy, and A Krot. Systematics and evaluation of meteorite classification. In *Meteorites and the Early Solar System II*, volume Two. University of Arizona Press, 2006.
- Naomi Woodley. Stronger privacy laws needed to protect public from drones, parliamentary committee says - ABC news (australian broadcasting corporation), July 2014. URL <http://www.abc.net.au/news/2014-07-14/call-for-stronger-privacy-laws-to-protect-from-drones/5594950>.
- XE. XE: (USD/AUD) US dollar to australian dollar rate, May 2014. URL <http://www.xe.com/currencyconverter/convert/?Amount=129&From=USD&To=AUD#converter>.

- Xinhua. S korea opens 2nd antarctic research station - china daily asia, 2014. URL <http://www.chinadailyasia.com/news/>.
- P. J. Zarco-Tejada, M. L. Guillén-Climent, R. Hernández-Clemente, A. Catalina, M. R. González, and P. Martín. Estimating leaf carotenoid content in vineyards using high resolution hyperspectral imagery acquired from an unmanned aerial vehicle (UAV). *Agricultural and Forest Meteorology*, 171–172(0):281 – 294, 2013. ISSN 0168-1923. doi: <http://dx.doi.org/10.1016/j.agrformet.2012.12.013>. URL <http://www.sciencedirect.com/science/article/pii/S0168192313000026>.

Appendices

APPENDIX A. PROJECT SPECIFICATION

A.1. Introduction. The project specification is a framework for this research project and details the planned achievements.

A.2. Project Aim. The objective of this project is to determine the suitability of a low cost hyper-spectral imaging system to detect meteorites from aerial vehicles.

A.3. Programme.

- (1) Design and construct a low cost hyper-spectral line scan camera adapted from an existing machine vision video camera;
- (2) Design and construct a speed & frame logging system to enable the line scan images to be spatially reconstructed;
- (3) Collect hyper-spectral image data for analysis;
- (4) Research and develop a system to analyse the data to determine the presence and location of objects of interest;
- (5) Research and analyse the effectiveness of the system for meteorite detection.
As time permits:
- (6) Enable the meteorite detection system to perform the hyper-spectral analysis and meteorite detection on board an aerial vehicle.

APPENDIX B. HYPER-SPECTRAL CAMERA GEOMETRY CALCULATIONS -
PYTHON CODE

```

# -*- coding: utf-8 -*-
"""
Created on Thu Apr 10 20:18:51 2014

@author: _David_Moorhouse
"""
import numpy as np
import matplotlib.pyplot as plt

## Collimator Calculations
src_aperture_rad = 1
coll_focal_len = 40

beam_radius = src_aperture_rad*coll_focal_len;
coll_div_angle = src_aperture_rad/coll_focal_len;

## Prism Calculations
#Edmund Optics 25mm Equilateral Prism N-SF11 Uncoated Stock
#No.#47-277
#refractive index of N-SF11

wl_band_start = 0.400;      #start of band (um)
wl_band_stop = 1;
wl_step = 0.001;
wl = np.arange(wl_band_start:wl_band_stop:wl_step)

n_prism=np.sqrt(1+1.73759695/(1-0.013188707/(wl**2))
               +0.313747346/(1-0.0623068142/(wl**2))
               +1.89878101/(1-155.23629/(wl**2)));

n_air = 1.000277;  #refractive index of air
prism_angle = np.deg2rad(60);  # Equalateral prism angles
                               (deg)
prism_side_len = 25e-3;      #prism side length (m)

deviation = (2*np.arcsin((n_prism/n_air)*(np.sin(0.5*
prism_angle))))
-prism_angle;
deviation_max = max(deviation);
deviation_min = min(deviation);
spectral_beam_angle = deviation_max - deviation_min;
# angle from white beam to centre of image sensor

```

```

deviation_avg = (deviation_max+deviation_min)/2;

prism_swivel = deviation_avg/2;      #prism base to incident
    beam angle (deg)

prism_height = prism_side_len*np.sin(prism_angle);
#prism center to incident beam offset
b = (np.deg2rad(90)-(np.deg2rad(180)-(deviation_avg/2)-
    prism_angle));
Y = (np.sin(deviation_avg/2)*((prism_height/2)/(np.sin(
    prism_angle))))
prism_offset = Y * np.cos(b)

print ('Prism_swivel_=%f_deg') % (np.rad2deg(prism_swivel))
print ('Prism_offset_=%f_mm') % (prism_offset*1000)

f=1;

plt.figure(f);
plt.plot(wl,n_prism)
plt.grid()
plt.xlabel('Wavelength_(um)');
plt.ylabel('Refractive_index');
plt.title('Prism_Refractive_Index');
f=f+1;

plt.figure(f);
plt.plot(wl, deviation);
plt.grid()
plt.xlabel('Wavelength_(um)');
plt.ylabel('Deviation_Angle_(deg)');
plt.title('Prism_Deviation');
f=f+1;

## Camera Placement

optical_fmt = (1/1.8)*0.0254;
pixel_size = 5.3e-6;
horiz_res = 1280;
vert_res = 1024;
sensor_width = pixel_size*horiz_res;
sensor_height = pixel_size*vert_res;
sensor_angle = deviation_avg;

```

```
sensor_dist = (sensor_width/2)/np.tan(spectral_beam_angle/2)
;

print ('Sensor_distance = %f mm' % (sensor_dist*1000))

plt.show()
```

APPENDIX C. METEORITICAL BULLETIN ANALYSIS - PYTHON CODE

```

# -*- coding: utf-8 -*-
"""
Created on Sun May 11 20:51:06 2014

@author: David Moorhouse
"""

import numpy as np
#import xlwt
import matplotlib.pyplot as plt
from scipy import stats

def count_unique(data):
    data_unique=np.unique(data)
    count_unique=np.empty((len(data_unique)),dtype=float)

    i=0
    for name in data_unique:
        count_unique[i]=len((np.where(data==name))[0])
        i+=1
    return(data_unique, count_unique)

def bin_other(data, records_no, labels, frac_limit):
    data_frac = data/records_no
    other_count = 0
    i=0
    for frac in data_frac:
        if frac < frac_limit:      #set upper threshold for '
            other' category
            labels[i] = 'Other\n'      #replace original
            name with 'Other'
            other_count += data[i]      #sum number in 'other
            ' category
        i+=1

    labels_short = np.unique(labels)
    data_count_short = np.empty((len(labels_short)))
    i=0
    for label in labels_short:
        if label == 'Other\n':
            data_count_short[i] = other_count
        try:
            data_count_short[i]=data[(np.where(labels ==
            label))]

```

```

        except:
            pass
        i+=1
#     data_count_short[(np.where(labels_short == 'Other')
# )] = other_count
    return(data_count_short, labels_short)

#####
#Import Data
#####

data=np.empty((49000,12), dtype='a128')

row=0
col=0
#a=0
met_file = open('Meteoritical_Bulletin_Search_the_Database.
html', 'r')

for i,line in enumerate(met_file):
#     break
    if i > 475:
        if col < 12:
            line = line[19:]
            while line.find('<')!=-1 and line.find('>')!=-1
            and line != None:
                line_del_start = line.find('<')
#                 print line
#                 print line_del_start
                line_del_stop = line.find('>')
#                 print line_del_stop
                line = line[0:line_del_start]+line[
                    line_del_stop+1:len(line)]
                line = line.replace('&nbsp;','')
                data[row][col]=(line)

            col+=1
            if col > 13:
                row+=1
                col=0

            if row >48972:
                break

```

```

records = len(filter(None, data[:,0]))
data = data[:records,:10]
#h=np.histogram(data[:,4],bins=[1880, 1982, 2008, 2012])

#####
#Programs
#####

programs = np.empty((len(data[:,9])),dtype='a128')
#remove unwanted characters from notes
i=0
for note in data[:,9]:
    note=note.replace('_', '')
    note=note.replace('0', '')
    note=note.replace('1', '')
    note=note.replace('2', '')
    note=note.replace('3', '')
    note=note.replace('4', '')
    note=note.replace('5', '')
    note=note.replace('6', '')
    note=note.replace('7', '')
    note=note.replace('8', '')
    note=note.replace('9', '')
    note=note.replace('(', '')
    note=note.replace(')', '')
    note=note.replace('JNIPR', 'NIPR')
    programs[i]=note
    i+=1

(programs_unique_names, programs_unique_count) =
    count_unique(programs)
(program_other_count, program_other_labels) = bin_other(
    programs_unique_count,
    len(programs), programs_unique_names, 0.01)
programs_unique_names[0]='Not_Listed'
program_other_labels[0]='Not_Listed'

f=1
plt.figure(f)
plt.pie(program_other_count, labels=program_other_labels,
    autopct='%1.1f%%',
    shadow=True, startangle=210, pctdistance=0.85)

```

```
#plt.title('Program Responsible for Collection of Catalogued  
Meteorites')
```

```
#####  
#Meteorite Group  
#####
```

```
met_group=np.empty((len(data[:,6])),dtype='a128')
```

```
i=0
```

```
for met in data[:,6]:
```

```
    met=met[:-1]
```

```
    while not str.isalnum(met):
```

```
        try:
```

```
            met=met[:-1]
```

```
        except:
```

```
            print 'can_not_shrink_further'
```

```
            break
```

```
    met=met.replace('Howardite','HOW')
```

```
    met=met.replace('Eucrite','EUC')
```

```
    met=met.replace('Diogenite','DIO')
```

```
    met=met.replace('Mesosiderite','MES')
```

```
    met=met.replace('Brachinite','BRA')
```

```
    met=met.replace('Acapulcoite','ACA')
```

```
    met=met.replace('Aubrite','AUB')
```

```
    met=met.replace('Winonaite','WIN')
```

```
    met=met.replace('Ureilite','URE')
```

```
    met=met.replace('Angrite','ANG')
```

```
    met=met.replace('Enst','Ungrouped')
```

```
    met=met.replace('Lodranite','LOD')
```

```
    met=met.replace('Pallasites','PAL')
```

```
    met=met.replace('_Achondrites','Achondrites')
```

```
#    if len(met)>3:
```

```
#        print met
```

```
    met_group[i]=met
```

```
    i+=1
```

```
(met_group_unique_names, met_group_unique_count) =  
    count_unique(met_group)
```

```
(met_group_other_count, met_group_other_labels) = bin_other(  
    met_group_unique_count,  
    len(met_group), met_group_unique_names, 0.008)
```

```

f+=1
plt.figure(f)
plt.pie(met_group_other_count, labels=met_group_other_labels
        , autopct='%1.0f%%',
          shadow=True, startangle=180, pctdistance=1.1,
          labeldistance=1.25,
          colors=('c', 'w'))
#plt.title('Catalogued Meteorite Types')

```

```

#####
#Meteorite Class
#####

```

```

met_group_char=np.empty((len(met_group)), dtype='a128')
met_class=np.empty((len(met_group)), dtype='a128')
i=0
for met in met_group:
    met_group_char[i]=met.strip('0123456789')
    i+=1

```

```

#Ordinary Chondrites
met_class[met_group_char=='H']='Ordinary'
met_class[met_group_char=='L']='Ordinary'
met_class[met_group_char=='LL']='Ordinary'

```

```

#Carbonaceous Chondrites
met_class[met_group_char=='CI']='Carbonaceous'
met_class[met_group_char=='CM']='Carbonaceous'
met_class[met_group_char=='CO']='Carbonaceous'
met_class[met_group_char=='CV']='Carbonaceous'
met_class[met_group_char=='CK']='Carbonaceous'
met_class[met_group_char=='CR']='Carbonaceous'
met_class[met_group_char=='CH']='Carbonaceous'
met_class[met_group_char=='CB']='Carbonaceous'

```

```

#Enstatite Chondrites
met_class[met_group_char=='EH']='Enstatite'
met_class[met_group_char=='EL']='Enstatite'

```

```

#Other Chondrites

```

```

met_class[met_group_char=='R'] = 'Other'
met_class[met_group_char=='K'] = 'Other'

(met_class_unique_names, met_class_unique_count) =
    count_unique(met_class)

#####
#Meteorite Type
#####

met_type=np.empty((len(met_group)), dtype='a128')
met_ct=np.empty((len(met_group)), dtype='a128')

#Chondrites
met_type[met_group_char=='H'] = 'Chondrites'
met_type[met_group_char=='L'] = 'Chondrites'
met_type[met_group_char=='LL'] = 'Chondrites'
met_type[met_group_char=='CI'] = 'Chondrites'
met_type[met_group_char=='CM'] = 'Chondrites'
met_type[met_group_char=='CO'] = 'Chondrites'
met_type[met_group_char=='CV'] = 'Chondrites'
met_type[met_group_char=='CK'] = 'Chondrites'
met_type[met_group_char=='CR'] = 'Chondrites'
met_type[met_group_char=='CH'] = 'Chondrites'
met_type[met_group_char=='CB'] = 'Chondrites'
met_type[met_group_char=='EH'] = 'Chondrites'
met_type[met_group_char=='EL'] = 'Chondrites'
met_type[met_group_char=='R'] = 'Chondrites'
met_type[met_group_char=='K'] = 'Chondrites'
met_type[met_group_char=='Chondrite'] = 'Chondrites'

#Primitive Achondrites
met_type[met_group_char=='URE'] = 'Primitive_Achondrites'
met_type[met_group_char=='BRA'] = 'Primitive_Achondrites'
met_type[met_group_char=='ACA'] = 'Primitive_Achondrites'
met_type[met_group_char=='LOD'] = 'Primitive_Achondrites'
met_type[met_group_char=='WN'] = 'Primitive_Achondrites'
met_type[met_group_char=='IAB'] = 'Primitive_Achondrites'
met_type[met_group_char=='IICD'] = 'Primitive_Achondrites'

#Achondrites
met_type[met_group_char=='ANG'] = 'Achondrites'

```

```

met_type[met_group_char=='AUB']='Achondrites'
met_type[met_group_char=='EUC']='Achondrites'
met_type[met_group_char=='DIO']='Achondrites'
met_type[met_group_char=='HOW']='Achondrites'
met_type[met_group_char=='MES']='Achondrites'
met_type[met_group_char=='MG_PAL']='Achondrites'
met_type[met_group_char=='ES_PAL']='Achondrites'
met_type[met_group_char=='PP_PAL']='Achondrites'
met_type[met_group_char=='PP_PAL']='Achondrites'
met_type[met_group_char=='IC']='Achondrites'
met_type[met_group_char=='IIAB']='Achondrites'
met_type[met_group_char=='IIC']='Achondrites'
met_type[met_group_char=='IID']='Achondrites'
met_type[met_group_char=='IIE']='Achondrites'
met_type[met_group_char=='IIAB']='Achondrites'
met_type[met_group_char=='IIIE']='Achondrites'
met_type[met_group_char=='IIIF']='Achondrites'
met_type[met_group_char=='IVA']='Achondrites'
met_type[met_group_char=='IVB']='Achondrites'
met_type[met_group_char=='SHE']='Achondrites'
met_type[met_group_char=='NAK']='Achondrites'
met_type[met_group_char=='CHA']='Achondrites'
met_type[met_group_char=='OPX']='Achondrites'
met_type[met_group_char=='Iron']='Achondrites'
met_type[met_group_char=='Mesosiderite']='Achondrites'
met_type[met_group_char=='PAL']='Achondrites'
met_type[met_group_char=='Lunar']='Achondrites'
met_type[met_group_char=='Martian']='Achondrites'
met_type[met_group_char=='Achondrite']='Achondrites'
met_type[met_group_char=='_Achondrite']='Achondrites'

(met_type_unique_names, met_type_unique_count) =
    count_unique(met_type)
(met_type_other_count, met_type_other_labels) = bin_other(
    met_type_unique_count,
    len(met_type), met_type_unique_names, 0.008)

i=0
for met in met_class:
    met_ct[i] = met + '_' + met_type[i]
    i+=1

(met_ct_unique_names, met_ct_unique_count) = count_unique(
    met_ct)

```

```

f+=1
plt.figure(f)
plt.pie(met_ct_unique_count, labels=met_ct_unique_names,
        autopct='%1.1f%%',
        shadow=True, startangle=140, pctdistance=0.93,
        colors=('c', 'w'))
#plt.title('Catalogued Meteorite Types')

#####
#Spectral Characteristics Meteorite Type
#####

met_spec_type=np.empty((len(met_group)), dtype='a128')

#Chondrites
met_spec_type[met_group_char=='H']='Ordinary_Chondrites'
met_spec_type[met_group_char=='L']='Ordinary_Chondrites'
met_spec_type[met_group_char=='LL']='Ordinary_Chondrites'
met_spec_type[met_group_char=='CI']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='CM']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='CO']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='CV']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='CK']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='CR']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='CH']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='CB']='Carbonaceous_Chondrites',
,
met_spec_type[met_group_char=='EH']='Enstatite_Chondrites'
met_spec_type[met_group_char=='EL']='Enstatite_Chondrites'
met_spec_type[met_group_char=='R']='Ungrouped'
met_spec_type[met_group_char=='K']='Ungrouped'
met_spec_type[met_group_char=='Chondrite']='Ungrouped'

#Primitive Achondrites
met_spec_type[met_group_char=='URE']='Ureilites'
met_spec_type[met_group_char=='BRA']='Ungrouped'

```

```

met_spec_type[met_group_char=='ACA']='Ungrouped'
met_spec_type[met_group_char=='LOD']='Ungrouped'
met_spec_type[met_group_char=='WIN']='Ungrouped'
met_spec_type[met_group_char=='IAB']='Ungrouped'
met_spec_type[met_group_char=='IICD']='Ungrouped'

```

#Achondrites

```

met_spec_type[met_group_char=='ANG']='Angrites'
met_spec_type[met_group_char=='AUB']='Enstatite_Achondrites'
met_spec_type[met_group_char=='EUC']='Basaltic_Achondrites'
met_spec_type[met_group_char=='DIO']='Basaltic_Achondrites'
met_spec_type[met_group_char=='HOW']='Basaltic_Achondrites'
met_spec_type[met_group_char=='MES']='Stony_Irons'
met_spec_type[met_group_char=='MG_PAL']='Stony_Irons'
met_spec_type[met_group_char=='ES_PAL']='Stony_Irons'
met_spec_type[met_group_char=='PP_PAL']='Stony_Irons'
met_spec_type[met_group_char=='PP_PAL']='Stony_Irons'
met_spec_type[met_group_char=='IC']='Iron'
met_spec_type[met_group_char=='IIAB']='Iron'
met_spec_type[met_group_char=='IIC']='Iron'
met_spec_type[met_group_char=='IID']='Iron'
met_spec_type[met_group_char=='IIE']='Iron'
met_spec_type[met_group_char=='IIAB']='Iron'
met_spec_type[met_group_char=='IIIE']='Iron'
met_spec_type[met_group_char=='IIIF']='Iron'
met_spec_type[met_group_char=='IVA']='Iron'
met_spec_type[met_group_char=='IVB']='Iron'
met_spec_type[met_group_char=='SHE']='Ungrouped'
met_spec_type[met_group_char=='NAK']='Nakhlites'
met_spec_type[met_group_char=='CHA']='Chassignites'
met_spec_type[met_group_char=='OPX']='Ungrouped'
met_spec_type[met_group_char=='Iron']='Ungrouped'
met_spec_type[met_group_char=='Mesosiderite']='Ungrouped'
met_spec_type[met_group_char=='PAL']='Stony_Irons'
met_spec_type[met_group_char=='Lunar']='Ungrouped'
met_spec_type[met_group_char=='Martian']='Ungrouped'
met_spec_type[met_group_char=='Achondrite']='Ungrouped'
met_spec_type[met_group_char=='Achondrites']='Ungrouped'
met_spec_type[met_group_char=='_Achondrites']='Ungrouped'
met_spec_type[met_group_char=='Primitive_Achondrites']='
    Ungrouped'

met_spec_type[met_spec_type=='_Achondrites']='Ungrouped'
met_spec_type[met_spec_type=='Other_Chondrites']='Ungrouped'
met_spec_type[met_spec_type=='_Ordinary']='Ungrouped'

```

```

met_spec_type[met_spec_type=='_Primitive_Achondrites'] = '
    Ungrouped'
met_spec_type[met_group_char=='Iron'] = 'Iron'

(met_spec_type_unique_names, met_spec_type_unique_count) =
    count_unique(met_spec_type)

f+=1
plt.figure(f)
plt.pie(met_spec_type_unique_count, labels=
    met_spec_type_unique_names, autopct='%1.1f%%',
        shadow=True, startangle=170, pctdistance=1.1,
        labeldistance=1.25,
        colors=('r', 'c', 'g', 'b', 'm'))
#plt.title('Catalogued Meteorite Types by Spectral Group')

met_spec_char=np.empty((len(met_group)), dtype='a128')

#Chondrites
met_spec_char[met_spec_type=='Ordinary_Chondrites'] = 'Strong'
met_spec_char[met_spec_type=='Basaltic_Achondrites'] = 'Strong'
,

met_spec_char[met_spec_type=='Nakhlites'] = 'Strong'
met_spec_char[met_spec_type=='Angrites'] = 'Strong'
met_spec_char[met_spec_type=='Chassignites'] = 'Strong'
met_spec_char[met_spec_type=='Ureilites'] = 'Weak'
met_spec_char[met_spec_type=='Stony_Irons'] = 'Weak'
met_spec_char[met_spec_type=='Iron'] = 'Featureless'
met_spec_char[met_spec_type=='Enstatite_Chondrites'] = '
    Featureless'
met_spec_char[met_spec_type=='Enstatite_Achondrites'] = '
    Featureless'
met_spec_char[met_spec_type=='Carbonaceous_Chondrites'] = '
    Weak/Featureless'
met_spec_char[met_spec_type=='Ungrouped'] = 'Ungrouped'

(met_spec_char_unique_names, met_spec_char_unique_count) =
    count_unique(met_spec_char)

f+=1
plt.figure(f)
plt.pie(met_spec_char_unique_count, labels=
    met_spec_char_unique_names, autopct='%1.1f%%',

```

```

        shadow=True, startangle=175, pctdistance=1.1,
        labeldistance=1.2,
        colors=('r','c','g','b','m'))
#plt.title('Catalogued Meteorite Types by Spectral
          Characteristics ')

#####
#Locations
#####

location_names_all = np.unique(data[:,5])
location_count_all = np.empty((len(location_names_all),1),
                               dtype=float)
location_frac_all = np.empty((len(location_names_all),1),
                              dtype=float)
weight_multiplier = np.ones((len(data[:,7]),1),dtype=float)
data_loc_short = data[:,5]

i=0
for location in data_loc_short:
    loc_del = location.find(',')
    if loc_del!=-1:
        data_loc_short[i] = location[loc_del+1:]
        data[i,5] = data_loc_short[i].replace('?', '')
        data[i,5] = data_loc_short[i].replace('(', '')
        data[i,5] = data_loc_short[i].replace(')', '')
        #remove leading spaces from names
        if len(data[i,5])>1:
            while (data[i,5])[0]=='_':
                data[i,5]=(data[i,5])[1:]
        #remove
        if len(data[i,7])>1:
            while '_' in data[i,7] and len(data[i,7])>1:
                if 'kg' in data[i,7]:
                    weight_multiplier[i]=1000
                if 'mg' in data[i,7]:
                    weight_multiplier[i]=0.001

            data[i,7]=((data[i,7])[:(len(data[i,7])-1)])
            data[i,7]=float(data[i,7])
    i+=1

location_names_short= np.unique(data_loc_short)
location_count_short = np.empty((len(location_names_short)),
                                 dtype=float)

```

```

i=0
for location in location_names_all:
    location_count_all[i] = np.sum(data[:,5]==location)
    i+=1

#location_names_short = np.unique(location_names_short)

i=0
for location in location_names_short:
    location_count_short[i] = np.sum(data_loc_short==
        location)
    i+=1
location_count_short[0] = 0

location_frac_all = (location_count_all)/records
location_frac_short = (location_count_short)/records

#Bin countries into northwest africa
location_names_short[location_names_short=='Algeria\n']='
    Northwest_Africa\n'
location_names_short[location_names_short=='Algeria_or_
    Morocco\n']='Northwest_Africa\n'
location_names_short[location_names_short=='Morocco\n']='
    Northwest_Africa\n'
location_names_short[location_names_short=='Morrocco\n']='
    Northwest_Africa\n'
location_names_short[location_names_short=='Mauritania\n']='
    Northwest_Africa\n'
location_names_short[location_names_short=='Mali\n']='
    Northwest_Africa\n'
location_names_short[location_names_short=='Western_Sahara\n
    ']='Northwest_Africa\n'
location_names_short[location_names_short=='United_States\n'
    ']='USA\n'
location_names_short[location_names_short=='United_States_of
    _America\n']='USA\n'

location_names_short_unique = np.empty((len(np.unique(
    location_names_short))), dtype='a128')
location_count_short_unique = np.empty((len(
    location_names_short_unique)))
i=0

```

```

j=0
NWafrica_count = 0
USA_count = 0

for location in location_names_short:
    if location == 'Northwest_Africa\n':
        NWafrica_count += location_count_short[j]
    elif location == 'USA\n':
        USA_count += location_count_short[j]
    else:
        location_names_short_unique[i] =
            location_names_short[j]
        location_count_short_unique[i] =
            location_count_short[j]
        i+=1
    j+=1
location_names_short_unique[i] = 'Northwest_Africa\n'
location_count_short_unique[i] = NWafrica_count
i+=1
location_names_short_unique[i] = 'USA\n'
location_count_short_unique[i] = USA_count

#
#(location_names_short2,location_count_short2) =
    count_unique(location_names_short)
#print location_names_short2
(location_frac_v_short, location_names_v_short) = bin_other(
    location_count_short_unique, records,
    location_names_short_unique, 0.01)

f+=1
plt.figure(f)
plt.pie(location_frac_v_short, labels=location_names_v_short
,
    autopct='%1.1f%%',shadow=True, startangle=295,
    pctdistance=1.1,
    labeldistance=1.25, colors=('c', 'w', 'c', 'w', 'c',
    'w', 'c', 'w'))
#plt.title('Location of Catalogued Meteorites')

#####
#Weight
#####

```

```

i=0
no_weight = 0
weights=np.empty((len(data[:,7])),dtype=float)
for weight in data[:,7]:
    try:
        weights[i]=float(data[i+no_weight,7])*
            weight_multiplier[i,0]
        i+=1
    except:
        no_weight +=1
weights=weights[:len(weights)-no_weight]
weights_sorted=np.sort(weights,axis=None)

weight_bins = np.r_[0:10000:5]
hist, bins = np.histogram(weights, bins=weight_bins, density
    =True)
width = 0.7 * (weight_bins[1] - weight_bins[0])
center = (weight_bins[:-1] + weight_bins[1:]) / 2

f+=1
plt.figure(f)
#plt.bar(center, hist, align='center', width=width, log=
    False)
plt.semilogy((np.r_[0:(len(weights)):1]),(weights_sorted))
plt.plot([0,50000],[np.mean(weights)]*2)
plt.plot([0,50000],[np.median(weights)]*2)
plt.legend(('Sample_Weight', 'Mean_Weight', 'Median_Weight'),
    loc=2)
plt.grid()
#plt.title('Meteorites Catalogued By Weight')
plt.xlabel('Sample_Number_Ordered_by_Weight')
plt.ylabel('Weight_(g)')

#####
#Volume
#####

met_vol = np.zeros((len(weights)))
ord_chon_weight = np.zeros((len(weights)))
density_H = 3.42 #g/cm^3
density_L = 3.36 #g/cm^3
density_LL = 3.22 #g/cm^3

i=0

```

```

for group in met_group_char:
    if group=='H':
        met_vol[i]=(weights[i])/density_H
        ord_chon_weight[i]=weights[i]
    elif group=='L':
        met_vol[i]=(weights[i])/density_L
        ord_chon_weight[i]=weights[i]
    elif group=='LL':
        met_vol[i]=(weights[i])/density_LL
        ord_chon_weight[i]=weights[i]
    i+=1
    if i==len(weights):
        break

vol_sorted=np.sort(met_vol,axis=None)
vol_sorted = vol_sorted[(max((np.where(vol_sorted==0))[0])
+1):]
vol_mean = np.mean(vol_sorted)
vol_median = np.median(vol_sorted)

met_med_volume = np.median(vol_sorted)
met_dia = 2*((np.median(vol_sorted))/((4*np.pi)/3)**(1/3.0))
met_area=(np.pi)*(met_dia/2)**2
f+=1
plt.figure(f)
plt.semilogy((np.r_[0:(len(vol_sorted)):1]),vol_sorted)
plt.plot([0,50000],[np.mean(vol_sorted)]*2)
plt.plot([0,50000],[np.median(vol_sorted)]*2)
plt.xlabel('Sample_Number_Ordered_by_Volume')
plt.ylabel('Volume_(cm^3)')
#plt.title('Volume of Catalogued Ordinary Chondrites')
plt.legend(('Sample_Volume','Mean_Volume','Median_Volume'),
loc=2)
plt.grid()

#####
#Target Weight
#####

ord_chon_weight_sorted=np.sort(ord_chon_weight,axis=None)
ord_chon_weight_sorted = ord_chon_weight_sorted[(max((np.
where(ord_chon_weight_sorted==0))[0])+1):]
ord_chon_weight_mean = np.mean(ord_chon_weight)

```

```

ord_chon_weight_median = np.median(ord_chon_weight)

f+=1
plt.figure(f)
plt.semilogy((np.r_[0:(len(ord_chon_weight_sorted)):1]),
             ord_chon_weight_sorted)
plt.plot([0,50000],[np.mean(ord_chon_weight_sorted)]*2)
plt.plot([0,50000],[np.median(ord_chon_weight_sorted)]*2)
plt.xlabel('Sample_Number_Ordered_by_Weight')
plt.ylabel('Weight_(g)')
#plt.title('Volume of Catalogued Ordinary Chondrites')
plt.legend(('Sample_Weight', 'Mean_Weight', 'Median_Weight'),
          loc=2)
plt.grid()

print 'Ordinary_Chondrites'
print 'Mean_Volume_=%0.3f_m^3' % (vol_mean/10)
print 'Median_Volume_=%0.3f_cm^3' % (vol_median)
print 'Mean_Weight_=%0.3f_kg' % (ord_chon_weight_mean/1000)
print 'Median_Weight_=%0.3f_g' % (ord_chon_weight_median)
print

#####
#Years
#####

years = np.r_[1800:2030:10]
hist, bins = np.histogram(data[:,4], bins=years)
width = 0.7 * (bins[1] - bins[0])
center = (bins[:-1] + bins[1:]) / 2
f+=1
plt.figure(f)
plt.bar(center, hist, align='center', width=width)
plt.grid()
#plt.title('Meteorites Catalogued By Decade of Discovery')
plt.xlabel('Decade')
plt.ylabel('Meteorites_Catalogued')

#####
#Print Stats
#####

print 'All_Meteorites'

```

```

print 'Catalogued_Meteorites_=%i' % (records)
print 'Catalogued_Meteorite_Weights_=%i' % (records-
    no_weight)
print 'Mean_Weight_=%0.3f_kg' % (np.mean(weights)/1000)
print 'Max_Weight_=%0.3f_kg' % (np.max(weights)/1000)
print 'Min_Weight_=%0.6f_g' % (np.min(weights))
print 'Median_Weight_=%0.3f_g' % (np.median(weights))
print 'Mode_Weight_=%0.3f_g' % (stats.mode(weights))[0]
print 'Median_Volume_=%0.3f_g' % (np.median(vol_sorted))
print

plt.show()

#####
#Spreadsheet Export
#####

#book = xlwt.Workbook(encoding="utf-8")
#
#sheet1 = book.add_sheet("Sheet 1")
#
#sheet1.write(0,0, "Name")
#sheet1.write(0,1, "Abbreviation")
#sheet1.write(0,2, "Status")
#sheet1.write(0,3, "Fall")
#sheet1.write(0,4, "Year")
#sheet1.write(0,5, "Place")
#sheet1.write(0,6, "Type")
#sheet1.write(0,7, "Mass (g)")
#sheet1.write(0,8, "MetBul")
#sheet1.write(0,9, "Antarctic")
#sheet1.write(0,11, "Notes")
#sheet1.set_panes_frozen(True) # frozen headings instead of
    split panes
#for i, row in enumerate(data):
#    for j, col in enumerate(row):
#        sheet1.write(i+1, j, col)
#
#book.save("meteorite.xls")

```

APPENDIX D. GAFFEY SPECTRA ANALYSIS - PYTHON CODE

```
# -*- coding: utf-8 -*-
"""
Created on Tue May 27 23:37:52 2014

@author: _David_Moorhouse
"""

def ord_chondrite_reflectance():
#
    import numpy as np
    import matplotlib.pyplot as plt
    # from scipy import stats
    import glob
    from scipy import spatial

    start_wl = 400
    stop_wl = 1000
    step_wl = 1

    f=0
    # plt.close('all')

    data={}
    spectra = glob.glob('../data/Gaffey/data/
        Ordinary_Chondrites/*.tab')

    # plt.figure(f)

    for spectrum in spectra:
        data[spectrum] = np.empty((171,3))
        spec_file = open(spectrum, 'r')
        for i,line in enumerate(spec_file):
            (data[spectrum])[i,0]=line[:5]
            (data[spectrum])[i,1]=line[8:15]
            (data[spectrum])[i,2]=line[17:25]
            if i==170:
                break
    # plt.plot(((data[spectrum][:,0]),((data[
    spectrum][:,1]))
```

```

#         plt.title('Ordinary Chondrite Spectral Reflectance
#         ')
#         plt.xlim((350,1200))
#         plt.xlabel('Wavelength (nm)')
#         plt.ylabel('Reflectance (%)')
#         plt.grid('on')
#         f+=1

```

```

#

```

```

#####

```

```

#Normalisation

```

```

#

```

```

#####

```

```

#     plt.figure(f)

```

```

max_reflect = np.zeros((len(data)))
i=0
for spectrum in data:
    max_reflect[i]=max((data[spectrum])[:,1])
    i+=1
max_reflect_tot = max(max_reflect)
wl_at_max=0
i=0
for spectrum in data:
    for ref in (data[spectrum])[:,1]:
        if ref==max_reflect_tot:
            wl_at_max = (data[spectrum])[i,0]
            i+=1
    i=0
norm_data = data
reflect_at_max = np.zeros((len(data)))
i=0
for spectrum in data:
    reflect_at_max[i]=(data[spectrum])[(((norm_data[
        spectrum])[:,0])==wl_at_max),1]
    i+=1

```

```

scale_factor = max_reflect_tot/reflect_at_max
i=0
for spectrum in data:
    (norm_data[spectrum])[:,1]=(((data[spectrum])[:,1])*
        scale_factor[i])/max_reflect_tot
#     plt.plot(((norm_data[spectrum])[:,0]),((data[
spectrum])[:,1]))
    i+=1

#plt.title('Ordinary Chondrite Spectral Reflectance\n
Normalised at '+str(wl_at_max)+'nm')
#     plt.xlim((350,1200))
#     plt.xlabel('Wavelength (nm)')
#     plt.ylabel('Reflectance Normalised to '+str(
wl_at_max)+'nm')
#     plt.grid('on')
#     f+=1

#
#####

#Normalised Average
#
#####

reflectance=np.zeros(((len(norm_data[spectrum])),len(
norm_data)))
i=0
for spectrum in norm_data:
    reflectance[:,i]=(norm_data[spectrum])[:,1]
    i+=1

norm_avg=np.zeros((len(reflectance)))
i=0
for wl in reflectance:
    norm_avg[i]=np.mean(reflectance[i,:])
    i+=1
#     plt.figure(f)
#     plt.plot(((norm_data[spectrum])[:,0]),norm_avg)
#     #plt.title('Ordinary Chondrite Spectral Reflectance
\n Average Normalised at '+str(wl_at_max)+'nm')
#     plt.xlim((350,1200))
#     plt.xlabel('Wavelength (nm)')

```

```

#     plt.ylabel('Reflectance Normalised to '+str(
        wl_at_max)+'nm')
#     plt.grid('on')
#     f+=1

#
#####

#Match to norm avg
#
#####

#     match = np.zeros((len(norm_data)))
#     samples=np.r_[0:len(match)]
#     i=0
#     for wl in match:
##         match[i]=1-np.mean(abs(norm_avg-reflectance[:,i]))
##         i+=1
#         match[i]=spatial.distance.canberra(norm_avg,
reflectance[:,i])
#
#     best_match = (np.max(match)*100)
#     worst_match = (np.min(match)*100)
#     avg_match = (np.mean(match)*100)
#     plt.figure(f)
#     plt.plot(samples,100*match,'bo')
#     #plt.plot([0,len(samples)],[best_match]*2)
#     #plt.plot([0,len(samples)],[worst_match]*2)
#     plt.plot([0,len(samples)],[avg_match]*2, 'r-')
#     #plt.title('Ordinary Chondrite Sample Spectra Match to
Reference')
#     plt.legend(('Sample Match','Mean Match'),loc=4)
##     plt.xlim((0,len(data)))
##     plt.ylim(0,100)
#     plt.xlabel('Sample Number (nm)')
#     plt.ylabel('Match (%)')
#     plt.grid('on')
#     f+=1
#
#     print 'Best match = %.2f%%' % (np.max(match)*100)
#     print 'Worst match = %.2f%%' % (np.min(match)*100)
#     print 'Mean match = %.2f%%' % (np.mean(match)*100)

```

```

#     plt.show()

#Trim

wl_start_trim= (((np.where(((data[spectrum])[ :,0]) ==
    start_wl)) [0]) [0])
wl_stop_trim=  (((np.where(((data[spectrum])[ :,0]) ==
    stop_wl)) [0]) [0])
#     ((norm_avg)[wl_start_trim:wl_stop_trim])
#data[spectrum]=(data[spectrum])[wl_start_trim:
    wl_stop_trim, :]

#Resize
old_wl_start=((data[spectrum])[0,0])
old_wl_stop=((data[spectrum])[ (len((data[spectrum])
    [:,0])) -1,0])
old_wl_size=(old_wl_stop-old_wl_start+1)
norm_avg_full = np.zeros(old_wl_size, dtype=float)
norm_avg_full[:]=np.nan
i=0
for pos in range(0, int(old_wl_size), 5):
    norm_avg_full[pos]=norm_avg[i]
    i+=1

#Interpolate

lin_start_pos=np.nan
lin_stop_pos=np.nan
j=0
for ref in norm_avg_full:
#     try:
#         if j>0:
#             if (np.isnan(ref)) and (not np.isnan(
#                 norm_avg_full[j-1])):
#                 lin_start=norm_avg_full[j-1]
#                 lin_start_pos=j-1
#                 print
#                 print 'start_pos %i'%(lin_start_pos)
#                 print 'start_val %f'%(lin_start)
#             elif (not np.isnan(ref)) and (np.isnan(
#                 norm_avg_full[j-1])):

```

```

        lin_stop=ref
        lin_stop_pos=j
#         print 'stop poss %i'%(lin_stop_pos)
#         print 'stop val %f'%(lin_stop)
        if ((not np.isnan(lin_start_pos)) and (not
np.isnan(lin_stop_pos))):
#         print 'eval1'
        if lin_start_pos < lin_stop_pos:
            norm_avg_full[lin_start_pos:
                lin_stop_pos]=np.linspace(
                    lin_start, lin_stop, (lin_stop_pos
                    )-(lin_start_pos))
#         print 'eval2'
#         print data_interp[lin_start_pos:
lin_stop_pos, i]
            j+=1

norm_avg_trim =100*(norm_avg_full)[start_wl-old_wl_start
: len(norm_avg_full)-(old_wl_stop-stop_wl)]
return(norm_avg_trim, norm_avg, reflectance, old_wl_size,
        spectrum, data)

```

APPENDIX E. IMAGE SENSOR SPECTRAL RESPONSE - PYTHON CODE

```
# -*- coding: utf-8 -*-
"""
Created on Fri May 30 21:41:08 2014

@author: _david
"""

import numpy as np

def sensor_response():
    wl_min = 400
    wl_max = 1000
    wl_step = 1

    wl = np.r_[wl_min:wl_max+1:wl_step]
    response = np.zeros(len(wl))

    response[wl==400]=28
    response[wl==405]=29
    response[wl==410]=30
    response[wl==415]=31
    response[wl==420]=32
    response[wl==425]=32
    response[wl==430]=32
    response[wl==435]=32
    response[wl==440]=33
    response[wl==445]=33
    response[wl==450]=34
    response[wl==455]=35
    response[wl==460]=35
    response[wl==465]=36
    response[wl==470]=37
    response[wl==475]=38
    response[wl==480]=38
    response[wl==485]=38
    response[wl==490]=39
    response[wl==495]=39
    response[wl==500]=39
    response[wl==505]=40
    response[wl==510]=41
    response[wl==515]=42
    response[wl==520]=42
```

response [wl=525]=43
response [wl=530]=43
response [wl=535]=43
response [wl=540]=43
response [wl=545]=43
response [wl=550]=43
response [wl=555]=44
response [wl=560]=44
response [wl=565]=44
response [wl=570]=44
response [wl=575]=45
response [wl=580]=43
response [wl=585]=43
response [wl=590]=43
response [wl=595]=43
response [wl=600]=44
response [wl=605]=46
response [wl=610]=46
response [wl=615]=44
response [wl=620]=45
response [wl=625]=47
response [wl=630]=46
response [wl=635]=48
response [wl=640]=47
response [wl=645]=46
response [wl=650]=46
response [wl=655]=47
response [wl=660]=47
response [wl=665]=47
response [wl=670]=46
response [wl=675]=45
response [wl=680]=44
response [wl=685]=43
response [wl=690]=43
response [wl=695]=42
response [wl=700]=42
response [wl=705]=41
response [wl=710]=40
response [wl=715]=39
response [wl=720]=38
response [wl=725]=37
response [wl=730]=38
response [wl=735]=39
response [wl=740]=40
response [wl=745]=41

response [wl=750]=41
response [wl=755]=40
response [wl=760]=39
response [wl=765]=38
response [wl=770]=37
response [wl=775]=35
response [wl=780]=34
response [wl=785]=33
response [wl=790]=32
response [wl=795]=32
response [wl=800]=31
response [wl=805]=30
response [wl=810]=29
response [wl=815]=28
response [wl=820]=27
response [wl=825]=27
response [wl=830]=26
response [wl=835]=25
response [wl=840]=24
response [wl=845]=23
response [wl=850]=23
response [wl=855]=22
response [wl=860]=20
response [wl=865]=20
response [wl=870]=19
response [wl=875]=18
response [wl=880]=18
response [wl=885]=18
response [wl=890]=17
response [wl=895]=17
response [wl=900]=17
response [wl=905]=17
response [wl=910]=16
response [wl=915]=15
response [wl=920]=14
response [wl=925]=13
response [wl=930]=13
response [wl=935]=12
response [wl=940]=11
response [wl=945]=11
response [wl=950]=10
response [wl=955]=10
response [wl=960]=9
response [wl=965]=8
response [wl=970]=7

```

response[wl==975]=7
response[wl==980]=6
response[wl==985]=6
response[wl==990]=5
response[wl==995]=5
response[wl==1000]=4

i=0
for sensitivity in response:
    try:
        if (sensitivity == 0 and response[i-1]!=0):
            lin_start=response[i-1]
            lin_start_pos=i-1
            #         print lin_start_pos
            #         print lin_start
        elif sensitivity != 0 and response[i-1]==0 :
            lin_stop=sensitivity
            lin_stop_pos=i
            #         print lin_stop_pos
            #         print lin_stop
            #         print
        if wl[lin_start_pos] < wl[lin_stop_pos]:
            response[lin_start_pos:lin_stop_pos]=np.
                linspace(lin_start , lin_stop ,(
                    lin_stop_pos)-(lin_start_pos))
    except:
        pass
    i+=1
return wl, response

#     plt.plot(wl, response)
#     plt.xlabel('Wavelength (nm)')
#     plt.ylabel('LSB(10 bits)/(nJ/cm^2)')
#     #plt.ylabel('Spectral Response \nLSB(10 bits)/(nJ/cm^2)')
#     plt.grid()
#     plt.show()

```

APPENDIX F. FRONT-END OPTICS - PYTHON CODE

```
# -*- coding: utf-8 -*-
"""
Created on Sun Oct 26 12:35:57 2014

@author: David Moorhouse
"""
import numpy as np

A=np.pi*1.2**2
r=0.0119455 #m
d=r*2
spatial_res=680
fov_dia=float(spatial_res*d)
fov_angle=np.deg2rad(np.array([22.7,14.5,10.4,6.2]))
fl=np.array([16,25,35,50], dtype=float)

alt=(fov_dia/2)/np.tan(fov_angle/2)

v=1/((1/fl)-(1/(alt*1000)))
```

APPENDIX G. XIPYAPI - XIMEA CAMERA API PYTHON CONNECTOR -
PYTHON CODE

```

# -*- coding: utf-8 -*-
"""
Created on Mon Sep  8 19:52:48 2014

@author: _David_Moorhouse
"""

import ctypes
import os

if os.name == 'posix' or os.name == 'mac':
    try:
        xi=ctypes.cdll.LoadLibrary('libm3api.so')
    except:
        xi=ctypes.cdll.LoadLibrary('/usr/lib/libm3api.so')
elif os.name == 'nt':
    xi=ctypes.cdll.LoadLibrary('m3api.dll')

def NewDeviceHandle():
    xiH=ctypes.pointer(ctypes.c_void_p(0))
    return xiH

def NewImagePointer():
    image = ctypes.pointer(XI_IMG(ctypes.sizeof(XI_IMG),None
    ,512))
    return image

def OpenDevice(xiH):
    print
    print('Opening device ... ')
    stat=xi.xiOpenDevice(0, xiH)
    print('xiApi_error_code:_%i'%(stat))
    assert stat == 0
    return None

def StartAcquisition(xiH):
    print
    print('Starting aquisition ... ')
    stat = xi.xiStartAcquisition(xiH.contents)
    print('xiApi_error_code:_%i'%(stat))
    assert stat == 0
    return None

```

```

def GetImage(xiH, timeout, image):
    # print
    # print('Getting Image...')
    if isinstance(timeout, int):
        timeout=ctypes.c_uint32(timeout)
    assert isinstance(timeout, ctypes.c_uint)
    stat = xi.xiGetImage(xiH.contents, timeout, image)
    # print('xiApi error code: %i'%(stat))
    raw = (ctypes.c_uint8*image.contents.width*image.
           contents.height).from_address(image.contents.bp)

    assert stat == 0
    return raw

def CloseDevice(xiH):
    print
    print('Closing device ...')
    stat=xi.xiCloseDevice(xiH)
    print('xiApi_error_code: %i'%(stat))
    assert stat == 0
    return None

def SetParamInt(xiH, value, param):
    print
    print('Setting %s ...'%(param))
    if isinstance(value, float):
        print
        print('Warning: data_type_float, converting to int')
        print
        value=int(value)
    if isinstance(value, int):
        value=ctypes.c_uint(value)
    assert type(value) is ctypes.c_uint, 'Invalid data_type: %s, int required'%((str(type(value))[7:])[:-2])
    stat = xi.xiSetParamInt(xiH.contents, param, value)
    print('xiApi_error_code: %i'%(stat))
    assert stat == 0
    return None

def GetParamInt(xiH, param):
    print
    print('Getting %s ...'%(param))
    value=ctypes.c_uint()

```

```

assert type(value) is ctypes.c_uint, 'Invalid_data_type:_
    %s,_int_required'%((str(type(value))[7:])[:-2])
stat = xi.xiGetParamInt(xiH.contents, param, ctypes.
    byref(value))
print('xiApi_error_code:_%i'%(stat))
assert stat == 0
return int(value.value)

def SetParamFloat(xiH, value, param):
    print
    print('Setting_%s...'%(param))
    if isinstance(value, int):
        print
        print('Warning:_data_type_int,_converting_to_float')
        print
        value=float(value)
    if isinstance(value, float):
        value=ctypes.c_float(value)
    assert type(value) is ctypes.c_float, 'Invalid_data_type:_
        %s,_float_required'%((str(type(value))[7:])[:-2])
    stat = xi.xiSetParamFloat(xiH.contents, param, value)
    print('xiApi_error_code:_%i'%(stat))
    assert stat == 0
    return None

class XI_IMG(ctypes.Structure):
    _fields_ = [("size", ctypes.c_uint32),
                ("bp", ctypes.c_void_p),
                ("bp_size", ctypes.c_uint32),
                ("frm", ctypes.c_uint32),
                ("width", ctypes.c_uint32),
                ("height", ctypes.c_uint32),
                ("nframe", ctypes.c_uint32),
                ("tsSec", ctypes.c_uint32),
                ("tsUSec", ctypes.c_uint32),
                ("GPI_level", ctypes.c_uint32),
                ("black_level", ctypes.c_uint32),
                ("padding_x", ctypes.c_uint32)]

```

APPENDIX H. HYPER-SPECTRAL CAMERA CONTROLS - PYTHON CODE

```

# -*- coding: utf-8 -*-
"""
Created on Sat Sep 27 16:39:39 2014

@author: David Moorhouse
"""
import numpy as np
import matplotlib.pyplot as plt
import cv2

def exposure_ctrl(xiH, xp, image_processed, exposure, image_size):
    saturated_pixels=float(np.size(np.where(image_processed>
        =236)[0]))
    max_pix=float(np.max(image_processed))
    print('Exposure_=%i us'%(exposure))
    print('Saturated_Pixels_=%i'%(saturated_pixels))
    print('Max_Pixel_Value_=%i'%(max_pix))

    if max_pix<235 or saturated_pixels>10:
        print('1')
        target=False
        if max_pix == 0:
            exposure_val = exposure*10
        else:
            exposure_val=int(exposure*(235/max_pix))
            print(exposure_val)
        if exposure_val>10 and exposure_val<900000:
            exposure = exposure_val
            xp.SetParamInt(xiH, exposure, "exposure")
    else:
        target=True
    return exposure, target

def save_results(spectrum, now, sample):
    y=str(now.year)
    m=str(now.month)
    d=str(now.day)
    h=str(now.hour)
    mi=str(now.minute)
    sample=str(sample)
    date_name =y+'-'+m+'-'+d+'-'+h+mi

```

```

filename1 = date_name+'_Sample_'+sample
np.save(filename1 ,spectrum)
filename2=filename1+'.txt'
text_file = open(filename2 , "w")
text_file.write("Sample_#_%s\n" % sample)
text_file.write("Distance_Result:\n")
text_file.close()
return None

def dual_plot (figure , plot_title , spectrum , wl , distance , frame ,
target , thresh , frame_limit ) :
plt.figure(figure)
plt.subplot(2,1,1)
plt.cla()
plt.subplot(2,1,1,title=plot_title ,
            ylabel='Pixel_Intensity')
plt.grid('on')
plt.plot(wl,spectrum)
plt.plot(wl,target)
plt.legend(('Live','Target'),loc='upper_left')
plt.ylim((0,254))
plt.subplot(2,1,2,title='Target_Detection', ylabel='
Distance_Result', xlabel='Frame_Number')
plt.grid('on')
if distance <= thresh:
    plt.plot(frame,distance,'go')
elif distance < thresh*2:
    plt.plot(frame,distance,'bo')
else:
    plt.plot(frame,distance,'ro')
plt.xlim((0,frame_limit))
plt.ylim((0,0.05))
plt.pause(0.0001)
return None

def smooth(x,window_len=11,window='hanning'):
    if x.ndim != 1:
        raise ValueError, "smooth_only_accepts_1_
dimension_arrays."
    if x.size < window_len:
        raise ValueError, "Input_vector_needs_to_be_
bigger_than_window_size."
    if window_len<3:
        return x

```

```

    if not window in ['flat', 'hanning', 'hamming', 'bartlett', 'blackman']:
        raise ValueError, "Window is one of 'flat', 'hanning', 'hamming', 'bartlett', 'blackman'"
    s=np.r_[2*x[0]-x[window_len-1:-1],x,2*x[-1]-x[-1:-window_len:-1]]
    if window == 'flat': #moving average
        w=np.ones(window_len,'d')
    else:
        w=eval('np.'+window+'(window_len)')
    y=np.convolve(w/w.sum(),s,mode='same')
    return y[window_len:-window_len+1]

def pre_process_line(raw):
    npimage=np.fromstring((raw),dtype='uint8')
    image_processed = np.flipud(npimage[0:np.size(npimage):4])
    image_processed = smooth(image_processed)
    return image_processed

def setup_plot(figure):
    cv2.destroyAllWindows()
    plt.clf()
    plt.figure(figure)
    plt.ion()
    plt.isinteractive()
    plt.show()
    return None

def plot_spectrum(wl,image_processed,target,wl_start,wl_stop,fps,ax):
    plt.cla()
    plt.grid('on')
    plt.plot(wl,image_processed)
    plt.xlim((wl_start,wl_stop))
    plt.ylim((0,254))
    plt.title('Spectral Reflectance')
    plt.xlabel('Approximate Wavelength (nm)')
    plt.ylabel('Pixel Intensity')
    plt.pause(0.0001)
    return None

def update_stats(fps,exposure):
    plt.clf()

```

```
plt.figtext(0.75,0.92,('Frame_Rate: %.2f_fps\nExposure: %i_us'%(fps, exposure)))  
return None
```

APPENDIX I. HYPER-SPECTRAL CAMERA MAIN - PYTHON CODE

```
# -*- coding: utf-8 -*-
"""
Created on Sat Sep 27 14:47:02 2014

@author: David Moorhouse
"""

import xipyApi as xp
import numpy as np
import cam_control as cc
import matplotlib.pyplot as plt
import matplotlib
import datetime
from scipy import spatial
import gpsdData as gps
matplotlib.use('TkAgg')

timeout = 50000
exposure = 10000
ON = 1
OFF = 0
fps = 10
gain = 0
aeag_level = 50
ag_max_limit = 0
ae_max_limit = 50000
width = 4
height=1024
x_offset = 640
center_px_roi = 285
wl_start=400
wl_stop=900
fps_update=30
fps_target=12.0
figure=1
thresh=0.00246
freerun=0
fps=0
frame_limit=1

gps.gpsp = gps.GpsPoller() # create the thread
gps.gpsp.start() # start it up
```

```

xiH = xp.NewDeviceHandle()
image=xp.NewImagePointer()
wl=np.linspace(wl_start ,wl_stop ,num=height)
xp.OpenDevice(xiH)
xp.SetParamInt(xiH, exposure, "exposure")
xp.SetParamInt(xiH, gain, "gain")
xp.SetParamInt(xiH, ON, "bpc")
xp.SetParamInt(xiH, width, "width")
xp.SetParamInt(xiH, x_offset, "offsetX")
xp.SetParamInt(xiH, height, "height")
xp.StartAcquisition(xiH)
ax=plt.figure(1)
target = np.load('2014-10-5-166_Sample_10.npy')
cc.setup_plot(figure)

if __name__=='__main__':
    for i in range(0,30):
        frame=0
        target_ex=False
        if freerun==0:
            try:
                sample=int(raw_input('Enter sample number or
                _a_letter_for_free_run_mode:'))
            except ValueError:
                print "Not_a_number,_entering_free_running_
                mode..."
                freerun=1
                frame_limit=1e3
                target_ex=False

        while target_ex==False:
            image_raw = xp.GetImage(xiH,timeout,image)
            image_processed = cc.pre_process_line(image_raw)
            exposure,target_ex=cc.exposure_ctrl(xiH,xp,
            image_processed,
            exposure,
            height)

        start_time = datetime.datetime.now()
        time1=start_time

        while frame<frame_limit:

```

```

print frame
image_raw = xp.GetImage(xiH, timeout, image)
image_processed = cc.pre_process_line(image_raw)
exposure, target_ex = cc.exposure_ctrl(xiH, xp,
    image_processed,
    exposure,
    height)
distance = spatial.distance.correlation(
    image_processed, target)
cc.dual_plot(figure, 'Spectral_Reflectance',
    image_processed, wl,
    distance, frame, target, thresh,
    frame_limit)
newlog = np.array([gps.gpsd.utc, gps.gpsd.fix.
    latitude,
    gps.gpsd.fix.longitude, gps.gpsd.fix.altitude
    ,
    gps.gpsd.fix.speed, distance])
try:
    log = np.vstack((log, newlog))
    np.save('logs/result_log_' + log[0, 0], log)
except:
    log = newlog
if frame == 1: # np.mod(frame, fps_update) == 0 and
    frame > 0:
    time2 = datetime.datetime.now()
    time_delta = time2 - time1
    sec_delta = time_delta.microseconds
    if sec_delta > 0:
        fps = float(fps_update) / (float(sec_delta)
            / 1e6)
    time1 = datetime.datetime.now()
    if freerun == 1:
        exposure = xp.GetParamInt(xiH, "exposure")
    cc.update_stats(fps, exposure)
frame += 1
if freerun == 0:
    cc.save_results(image_processed, start_time,
        sample)

plt.cla()
xp.CloseDevice(xiH)

```

APPENDIX J. HYPER-SPECTRAL MATCHING - PYTHON CODE

```

# -*- coding: utf-8 -*-
"""
Created on Sat Aug 30 14:19:47 2014

@author: David Moorhouse
"""

import numpy as np
from scipy import spatial
import glob
import os
import pylab as plt

def chosen(thresh_chosen, distance_chosen, f, title, labels):
    plt.figure(f)
    margin = ((thresh_chosen - distance_chosen) / thresh_chosen)
             * 100
    width = 0.35
    plt.bar(np.array(terr)+1, margin[terr], width, color='r')
    plt.bar(np.array(exterr)+1, margin[exterr], width, color='b')
    plt.grid()
    plt.xlabel('Sample')
    plt.ylabel('Above_Acceptance_Threshold_(%)')
    # plt.xticks(np.array(range(samples))+1+width/2., np.
    array(range(samples))+1)

    plt.xticks(np.array(range(samples))+1+width/2., labels,
               rotation='vertical')
    plt.title(title)
    plt.legend(('Terrestrial', 'Extra-terrestrial'))
    # plt.ylim(-800,200)
    f+=1
    return f

def score(match_meth, samples, false_neg, high_pos):
    j=0
    false_pos_temp=np.zeros(np.size(exterr))
    false_neg_temp=np.zeros(np.size(exterr))
    thresh=np.zeros(np.size(exterr))
    # if high_pos==1:
    #     for sample1 in range(8,16):
    #         thresh[j] = (np.mean(match_meth[0:7, sample1]) +
    np.mean(match_meth[8:15, sample1]))/2

```

```

#         thresh[j]= np.max(match_meth[(terr),sample1])
#         false_pos_temp[j]=(sum(i > thresh[j] for i in (
match_meth[terr, sample1]))
#         false_neg_temp[j]=(sum(i <=thresh[j] for i in (
match_meth[exterr, sample1]))
#         j+=1
if high_pos==0:
    for sample1 in range(8,16):
#         thresh[j]= (np.mean(match_meth[0:7, sample1])+np
.mean(match_meth[8:15, sample1]))/2
#         print sample1
#         min_loc=np.where(np.min(match_meth[terr, sample1
])==match_meth[terr, sample1])[0][0]
#         try:
#             thresh1=np.min(match_meth[terr[: (min_loc)],
sample1])
#         except:
#             thresh1=10e6
#         try:
#             thresh2=np.min(match_meth[terr[(min_loc+1)
:], sample1])
#         except:
#             thresh2=10e6
#         print thresh1
#         print thresh2
#         thresh[j]=np.min(thresh1, thresh2)

#         thresh[j]= np.min(match_meth[terr, sample1])
thresh[j]= np.min(match_meth[(terr), sample1])
#         print thresh[j]
false_pos_temp[j]=(sum(i < thresh[j] for i in (
match_meth[terr, sample1]))
false_neg_temp[j]=(sum(i >= thresh[j] for i in (
match_meth[exterr, sample1]))

        j+=1
#     false_neg_stack=np.hstack((false_neg, false_neg_temp))
#     print false_neg_temp
#     return false_pos_temp, false_neg_temp, thresh,
false_neg_stack
return false_neg_temp

def plot_results(match_meth, f, labels, title, dist_max, method):
    for sample1 in range(8,16):
        plt.figure(f)

```

```

width = 0.35
plt.plot(np.array(terr)+1,match_meth[terr , sample1] , '
        ro')
plt.plot(np.array(exterr)+1,match_meth[exterr ,
        sample1] , 'bo')
plt.errorbar(np.array(terr)+1,match_meth[terr ,
        sample1] ,(dist_max[method])[terr] , ls='_' , color='
        red')
plt.errorbar(np.array(exterr)+1,match_meth[exterr ,
        sample1] ,(dist_max[method])[exterr] , ls='_' , color
        ='blue')
#     plt.grid()
plt.xlabel('Sample')
plt.ylabel('Distance_Result')
plt.xticks(np.array(range(samples))+1+width/2. ,
        labels , rotation='vertical')
plt.title(title+str(sample1+1))
plt.legend(('Terrestrial' , 'Extra-terrestrial'))
plt.ylim(ymin=0)
f+=1
return f

def get_data():
sample=np.zeros((1024 , samples))
for file in glob.glob("*.npy"):
    print file
    for sample_no in range(1 , samples+1):
        if file.find('Sample_'+str(sample_no)+' .npy') !=
            -1:
            sample[:, sample_no-1]=np.load(file)
os.chdir('..')
return sample

def get_distance(wl_crop_start , wl_crop_stop , samples , sample):
row=0
col=0
#     pearson=np.zeros((samples , samples))
braycurtis=np.zeros((samples , samples))
canberra=np.zeros((samples , samples))
chebyshev=np.zeros((samples , samples))
cityblock=np.zeros((samples , samples))
correlation=np.zeros((samples , samples))
cosine=np.zeros((samples , samples))
euclidean=np.zeros((samples , samples))
for sample_u in range(samples):

```

```

        for sample_v in range(samples):
#            pearson[sample_u, sample_v], p_value = stats.
pearsonr(sample[wl_crop_start:wl_crop_stop, sample_u],
sample[wl_crop_start:wl_crop_stop, sample_v])
            braycurtis[sample_u, sample_v]=spatial.distance.
                braycurtis(sample[wl_crop_start:wl_crop_stop
                    , sample_u], sample[wl_crop_start:wl_crop_stop
                    , sample_v])
            canberra[sample_u, sample_v]=spatial.distance.
                canberra(sample[wl_crop_start:wl_crop_stop,
                    sample_u], sample[wl_crop_start:wl_crop_stop,
                    sample_v])
            chebyshev[sample_u, sample_v]=spatial.distance.
                chebyshev(sample[wl_crop_start:wl_crop_stop,
                    sample_u], sample[wl_crop_start:wl_crop_stop,
                    sample_v])
            cityblock[sample_u, sample_v]=spatial.distance.
                cityblock(sample[wl_crop_start:wl_crop_stop,
                    sample_u], sample[wl_crop_start:wl_crop_stop,
                    sample_v])
            correlation[sample_u, sample_v]=spatial.distance.
                correlation(sample[wl_crop_start:
                    wl_crop_stop, sample_u], sample[wl_crop_start:
                    wl_crop_stop, sample_v])
            cosine[sample_u, sample_v]=spatial.distance.
                cosine(sample[wl_crop_start:wl_crop_stop,
                    sample_u], sample[wl_crop_start:wl_crop_stop,
                    sample_v])
            euclidean[sample_u, sample_v]=spatial.distance.
                euclidean(sample[wl_crop_start:wl_crop_stop,
                    sample_u], sample[wl_crop_start:wl_crop_stop,
                    sample_v])
            row+=1
        col+=1
#        row=0
sample_dist = {'braycurtis': braycurtis, 'canberra':
    canberra,
    'chebyshev': chebyshev, 'cityblock': cityblock,
    'correlation': correlation, 'cosine': cosine,
    'euclidean': euclidean}
return sample_dist

def get_distance_all(wl_crop_start, wl_crop_stop, samples,
    sample):
    row=0

```

```

col=0
# pearson=np.zeros((samples,samples))
braycurtis=np.zeros((samples,samples))
canberra=np.zeros((samples,samples))
chebyshev=np.zeros((samples,samples))
cityblock=np.zeros((samples,samples))
correlation=np.zeros((samples,samples))
cosine=np.zeros((samples,samples))
euclidian=np.zeros((samples,samples))

for sample_u in range(samples):
    for sample_v in range(samples):
#         pearson[sample_u,sample_v], p_value = stats.
pearsonr(sample[wl_crop_start:wl_crop_stop,sample_u],
sample[wl_crop_start:wl_crop_stop,sample_v])
        braycurtis[sample_u,sample_v]=spatial.distance.
            braycurtis(sample[wl_crop_start:wl_crop_stop,
                sample_u],sample[wl_crop_start:wl_crop_stop,
                sample_v])
        canberra[sample_u,sample_v]=spatial.distance.
            canberra(sample[wl_crop_start:wl_crop_stop,
                sample_u],sample[wl_crop_start:wl_crop_stop,
                sample_v])
        chebyshev[sample_u,sample_v]=spatial.distance.
            chebyshev(sample[wl_crop_start:wl_crop_stop,
                sample_u],sample[wl_crop_start:wl_crop_stop,
                sample_v])
        cityblock[sample_u,sample_v]=spatial.distance.
            cityblock(sample[wl_crop_start:wl_crop_stop,
                sample_u],sample[wl_crop_start:wl_crop_stop,
                sample_v])
        correlation[sample_u,sample_v]=spatial.distance.
            correlation(sample[wl_crop_start:
                wl_crop_stop,sample_u],sample[wl_crop_start:
                wl_crop_stop,sample_v])
        cosine[sample_u,sample_v]=spatial.distance.
            cosine(sample[wl_crop_start:wl_crop_stop,
                sample_u],sample[wl_crop_start:wl_crop_stop,
                sample_v])
        euclidian[sample_u,sample_v]=spatial.distance.
            euclidean(sample[wl_crop_start:wl_crop_stop,
                sample_u],sample[wl_crop_start:wl_crop_stop,
                sample_v])
    row+=1
col+=1

```

```

#         row=0
return braycurtis , canberra , chebyshev , cityblock ,
       correlation , cosine , euclidian

#class dist_methods:
#    def __init__(self):
#        cn=spatial.distance.canberra
#class samp:

f=1
dates=['2014-9-29', '2014-10-4', '2014-10-5']

labels=['Unknown_1', 'Unknown_2', 'Unknown_3', 'Unknown_4', '
Unknown_5', 'Unknown_6',
'Unknown_7', 'Unknown_8', 'NWA869\nL4-6_Crust', 'NWA869\nL4
-6_Cut',
'NWA774\nH4_Crust', 'NWA774\nH4_Cut', 'NWA775\nL6_Cut', '
Gao-Guenie\nH5_Crust',
'NWA791\nL6_Cut', 'Kharabali\nH5_Crust', 'Shale', 'Granite',
'Limestone',
'Sandstone', 'Basalt']

samples=21
terr=[0,1,2,3,4,5,6,7,16,17,18,19,20]
exterr=[8,9,10,11,12,13,14,15]

wl_crop_start=0
wl_crop_stop=1023-3
height=1024
wl_start=400
wl_stop=900
wl=np.linspace(wl_start, wl_stop, num=height)
dist_methods=['braycurtis', 'canberra', 'chebyshev', 'cityblock',
'correlation', 'cosine', 'euclidean']

os.chdir(dates[0])
sample0 = get_data()
sample0_dist = get_distance(wl_crop_start, wl_crop_stop,
samples, sample0)
#braycurtis0, canberra0, chebyshev0, cityblock0, correlation0,
cosine0, euclidian0 = get_distance(wl_crop_start,
wl_crop_stop, samples, sample0)

```

```

os.chdir(dates[1])
sample1 = get_data()
sample1_dist = get_distance(wl_crop_start, wl_crop_stop,
                             samples, sample1)
#braycurtis1, canberra1, chebyshev1, cityblock1, correlation1,
  cosine1, euclidian1 = get_distance(wl_crop_start,
                                     wl_crop_stop, samples, sample1)

os.chdir(dates[2])
sample2 = get_data()
sample2_dist = get_distance(wl_crop_start, wl_crop_stop,
                             samples, sample2)
#braycurtis2, canberra2, chebyshev2, cityblock2, correlation2,
  cosine2, euclidian2 = get_distance(wl_crop_start,
                                     wl_crop_stop, samples, sample2)

sample3=sample2

dist_samples={'2014-9-29': sample0, '2014-10-4': sample1, '
             2014-10-5': sample2}

dist_error={'braycurtis': np.zeros((6, samples)), 'canberra': np
           .zeros((6, samples)),
           'chebyshev': np.zeros((6, samples)), 'cityblock': np.zeros((6,
           samples)),
           'correlation': np.zeros((6, samples)), 'cosine': np.zeros((6,
           samples)),
           'euclidean': np.zeros((6, samples))}
dist_max={'braycurtis': np.zeros((samples)), 'canberra': np.
         zeros((samples)),
         'chebyshev': np.zeros((samples)), 'cityblock': np.zeros((
         samples)),
         'correlation': np.zeros((samples)), 'cosine': np.zeros((samples
         )),
         'euclidean': np.zeros((samples))}
dist_mean=dist_max
false_neg={'braycurtis': np.zeros((1)), 'canberra': np.zeros
         ((1)),
         'chebyshev': np.zeros((1)), 'cityblock': np.zeros((1)),
         'correlation': np.zeros((1)), 'cosine': np.zeros((1)),
         'euclidean': np.zeros((1))}

false_pos=false_neg

```

```

thresh=false_neg

sample_all=np.concatenate((sample0 ,sample1 ,sample2) ,axis=1)
#for sample_no in range(samples):
##    sample_min[sample_no,0]=
distuv=np.zeros((samples))
distuw=np.zeros((samples))
distvw=np.zeros((samples))

for sample_no in range(samples):
    u=sample0[wl_crop_start:wl_crop_stop ,sample_no]
    v=sample1[wl_crop_start:wl_crop_stop ,sample_no]
    w=sample2[wl_crop_start:wl_crop_stop ,sample_no]
    x=sample3[wl_crop_start:wl_crop_stop ,sample_no]
    pairs=[u,v,u,w,u,x,v,w,v,x,w,x]
    pair_no=0
    for pair in pairs:
        for method_name in dist_methods:
            method=getattr(spatial.distance , method_name)
            (dist_error[method_name])[pair_no,sample_no]=
                method(pair , pairs[pair_no+1])
            if pair_no==6-2:
                break
        pair_no+=1
    for method_name in dist_methods:
        (dist_max[method_name])[sample_no]=np.max((
            dist_error[method_name])[ : ,sample_no])
        (dist_mean[method_name])[sample_no]=np.mean((
            dist_error[method_name])[ : ,sample_no])

plt.figure(f)
plt.plot(wl[wl_crop_start:wl_crop_stop] ,u)
plt.plot(wl[wl_crop_start:wl_crop_stop] ,v)
plt.plot(wl[wl_crop_start:wl_crop_stop] ,w)
plt.grid()
title='Sample_'+str(sample_no+1)+'_'+labels[sample_no]
plt.title(title)
plt.xlabel('Wavelength_Approximate_(nm)')
plt.ylabel('Pixel_Intensity')
plt.legend(dates ,loc=2)

f+=1

```

```

#braycurtis , canberra , chebyshev , cityblock , correlation , cosine ,
  euclidian = get_distance_all(wl_crop_start , wl_crop_stop ,
    samples , dist_mean)

#braycurtis , canberra , chebyshev , cityblock , correlation , cosine ,
  euclidian=get_distance(wl_crop_start , wl_crop_stop ,
    samples , sample2)
false_neg2=[]
#pr_false_pos , pr_false_neg , pr_thresh , false_neg = score(
  pearson2 , samples , false_neg , 1)
#for method_name in dist_methods:
#   method=getattr(spatial.distance , method_name)
#   false_pos[method_name] , false_neg[method_name] , thresh[
method_name] , false_neg = score(braycurtis2 , samples ,
  false_neg , 0)

for method in dist_methods:
  false_neg[method] = score(sample2_dist[method] , samples ,
    false_neg2 , 0)
  print(method+' \tfalse negatives , \tmin: %i \tmean: %f'%(
    np.min(false_neg[method]) , np.mean(false_neg[method])
  )

date=dates[2]
for method in dist_methods[4:6]:
  title=( 'Meteorite_Detection_Test: %s \nDistance_Algorithm
    : ' + method+' \nReference: Sample_')%(date)
  f=plot_results(sample2_dist[method] , f , labels , title ,
    dist_max , method)

plt.figure(2)
plt.plot(np.r_[0: samples + 2] , [0.00246]*( samples+2) , 'g')
plt.show()

```

APPENDIX K. USGS & GAFFEY MATCHING - PYTHON CODE

```

# -*- coding: utf-8 -*-
"""
Created on Mon Jun  9 14:49:46 2014

@author: David Moorhouse
"""

from image_sensor import sensor_response
from solar import solar_irradiance
from usgs_spectra import usgs_speclib
from Ord_Chondrite_spectra import ord_chondrite_reflectance
import numpy as np
import matplotlib.pyplot as plt
from scipy import spatial
import glob

ordchon_avg, norm_avg, reflectance2, old_wl_size, a, data =
    ord_chondrite_reflectance()
np.save('ordchon_avg', ordchon_avg)
sensor_wl, sensor_sensitivity = sensor_response()

start_wl = 400
stop_wl = 1000
step_wl = 1

f=0

data={}
spectra = glob.glob('../data/Gaffey/data/Ordinary_Chondrites
/*.*tab')

for spectrum in spectra:
    data[spectrum] = np.empty((171,3))
    spec_file = open(spectrum, 'r')
    for i, line in enumerate(spec_file):
        (data[spectrum])[i,0]=line[:5]
        (data[spectrum])[i,1]=line[8:15]
        (data[spectrum])[i,2]=line[17:25]
        if i==170:
            break

try:
    solar_data=np.load('solar_data_array.npy')

```

```

irradiance_norm=np.load('irradiance_norm_array.npy')
solar_title=np.load('solar_title_array.npy')
except:
    solar_data,irradiance_norm,solar_title =
        solar_irradiance()
    np.save('solar_data_array',solar_data)
    np.save('irradiance_norm_array',irradiance_norm)
    np.save('solar_title_array',solar_title)
    pass

solar_wl = solar_data[:,0]

try:
    usgs_data = np.load('usgs_data_array.npy')
    usgs_material = np.load('usgs_material_array.npy')
except:
    usgs_data, usgs_material = usgs_speclib()
    np.save('usgs_data_array',usgs_data)
    np.save('usgs_material_array',usgs_material)
    pass

assert (len(sensor_wl) == len(solar_wl)), 'ERROR: _mismatched_
    wavelengths'

wl=solar_wl
reflectance=np.zeros((len(reflectance2[:,0]),len(data)))
col=0
for index in data:
    #    print col
    reflectance[:,col] = (data[index])[:,1]
    col+=1

f=100

match = np.zeros((len(ordchon_avg)))
samples=np.r_[0:len(match)]
i=0
for x in match:
    match[i]=100-np.mean(abs(ordchon_avg-usgs_data[:,i]))
    i+=1

```

```

match2 = np.zeros((len(reflectance[0,:])))
samples=np.r_[0:len(match2)]
i=0
for gaff_sample in range(len(reflectance[0,:])):
    match2[gaff_sample]=spatial.distance.correlation(
        norm_avg,reflectance[:,gaff_sample])

#Interpolate
gaff_wide=np.zeros((len(wl),len(reflectance[0,:])))
gaff_wide[:]=np.nan

i=0

for spectrum in reflectance.T:
    j=0
    k=0
    lin_start_pos=np.nan
    lin_stop_pos=np.nan
    for pos in range(0,len(wl),5):
        gaff_wide[pos,i]=spectrum[k]
        k+=1
    for ref in gaff_wide[:,i]:
        if j>0:

            if (np.isnan(ref)) and (not np.isnan(gaff_wide[j-1,i])):
                lin_start=gaff_wide[j-1,i]
                lin_start_pos=j-1
            elif (not np.isnan(ref)) and (np.isnan(gaff_wide[j-1,i])):
                lin_stop=ref
                lin_stop_pos=j
            if ((not np.isnan(lin_start_pos)) and (not np.isnan(lin_stop_pos))):

                if lin_start_pos < lin_stop_pos:
                    gaff_wide[lin_start_pos:lin_stop_pos,i]=
                        np.linspace(lin_start,lin_stop,(
                            lin_stop_pos)-(lin_start_pos))

                j+=1
        i+=1

total=np.hstack((usgs_data,gaff_wide))
i=0

```

```

cut=0
dist_full=np.zeros((len(total[0,:])))
ref_no=0
f=0
log=np.zeros((77))
ref_lib=np.vstack((gaff_wide[:,33],gaff_wide[:,34],gaff_wide
[:,69]))
for reference in ref_lib:
    for spectrum in total.T:
        cell_no=0
        trim_start=0
        trim_stop=len(spectrum)
        for cell in spectrum:
            if cell_no>0:
                if (not np.isnan(cell)) and (np.isnan(
                    spectrum[cell_no-1])):
                    trim_start=cell_no
                elif (np.isnan(cell)) and (not np.isnan(
                    spectrum[cell_no-1])):
                    trim_stop=cell_no-1
            cell_no+=1

        dist_full[i]=spatial.distance.correlation(spectrum[
            trim_start:trim_stop],reference[trim_start:
            trim_stop])
        i+=1
i=0
dist=np.delete(dist_full,np.where(np.isnan(dist_full))
[0])
plt.figure(f)
samples=range(1,len(dist)+1)
plt.plot(samples[:77],dist[:77], 'ro')
plt.plot(samples[77:],dist[77:], 'bo')
plt.grid()
plt.legend(('USGS', 'Gaffey'))
plt.xlabel('Sample_Number')
plt.ylabel('Distance_Result')
plt.ylim(ymin=0)
plt.xlim(xmin=1)
f+=1

thresh = np.max(dist[-77:])+0.0001
false_pos=len(np.where(dist[:77]<thresh)[0])
true_pos=len(np.where(dist[-77:]<thresh)[0])
false_neg=len(np.where(dist[-77:]>thresh)[0])

```

```

true_neg=len(np.where(dist[: -77]>thresh)[0])
print('Reference_Number:_%i'%(ref_no))
print('Min_USGS_=%f'%(np.min(dist[: -77])))
print('Min_Gaffey_=%f'%(np.min(dist[-77:])))
print('Max_Gaffey_=%f'%(np.max(dist[-77:])))
print('USGS_<_Thresh_=%i'%(false_pos))
print('Gaffey_<_Thresh_=%i'%(true_pos))
print('USGS_>_Thresh_=%i'%(true_neg))
print('Gaffey_>_Thresh_=%i'%(false_neg))
print(usgs_material[(np.where(dist_full[: -77]<thresh))
[0]])
print('Gaffey_Accepted_=%0.1f%%'%(100*float(true_pos)
/(true_pos+false_neg)))
print('USGS_Rejected_=%0.1f%%'%(100*float(true_neg)/(
true_neg+false_pos)))
print
log[ref_no]=100*true_pos/(true_pos+false_neg)
ref_no+=1

plt.show()

```