UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING AND THE ENVIRONMENT

Astronautics Research Group

Onboard computer technology for cubesats

by

Warinthorn Kiadtikornthaweeyot

Thesis for the degree of Doctor of Philosophy

June 2015

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND THE ENVIROMENT

Doctor of Philosophy

ONBOARD COMPUTER TECHNOLOGY FOR CUBESATS

Warinthorn Kiadtikornthaweeyot

This thesis addresses the problem of cubesat limitations on transmission power and onboard memory storage. The number of small satellites is continuing to increase. The reduced amount of time and budget required for the development of these satellites has considerable advantages. The short time leads to data becoming available faster than from a larger satellite. Consequently, the communication system is very important to ensure that all data from the cubesat can be transmitted to the receiving station. In this thesis the link budget of a cubesat has been studied to identify the constraints on power and data transmission.

As cubesat satellites become more complex, additional constraints and requirements are placed on system components. For more complex missions, greater flexibly of the onboard computer architecture is required to support the mission adaptation or changing specifications of onboard devices. Alternative onboard computer architecture for the next generation of cubesats is presented in this thesis and hybrid onboard computer architecture is proposed.

There are many cubesats which have provided remote sensing imagery. An issue is how to store the data onboard and how to transmit these data with limited power. A solution is to reduce the size of the original image by pre-image processing. The potential for using image compression and defining the region of interest to decrease the original satellite image size has been examined in this research. Three approaches are studied and described in the context of the region of interest technique. There is image segmentation based on edge, histogram and texture detection. The presented evaluation is focused on the detection of the land part of the image that contains dynamic information and rejecting the ocean where there is less interest. The technique, however, is equally applicable for any region of interest that can be characterised and this is illustrated by considering some examples. The proposed adaptive image compression system is made up of two parts. The first part consists of the identification of the region of interest and the second part the image compression of this region of interest.

The accuracy of the proposed system has been examined by comparing the number of different pixels between the proposed automatic region of interest system and the manual detection of the region of interest.

Morphological methods are the main technique that has been used in the system. The morphology structure element has different shapes and size and it is necessary to understand how the shape and size of the structure elements affects the proposed system. A study of structure element has been conducted.

In the real implementation of the proposed system on a cubesat, additional power would be required. To quantify this increase, a particular proposed system based on edge segmentation for region of interest automatic detection has been studied. The potential for using the proposed image compression to detect the region of interest and image compression was examined using a standard microcontroller. The result shows that the proposed system could be used on a cubesat satellite with reasonable additional power and mass.

Contents

List of tables

List of figures

Nomenclature

D	Antenna diameter	(m)
e	Elevation angle	(deg)
E_b/N_o	Bit energy per noise-density	(dB)
f	Frequency	(Hz)
G_{rx}	Peak receiver antenna gain	(dB)
G_{tx}	Peak transmitter antenna gain	(dB)
k	Boltzmann's constant = 1.38066×10^{-23}	(J/K)
K˚	Kelvin	(K)
L_a	Additional path loss	(dB)
L_s	Free space loss	(dB)
L_{tx}	Transmitter to antenna line loss	(dB)
ŋ	Antenna efficiency	
P_{tx}	Transmitter power	(dBW)
R	Data rate	(bps)
R_e	Earth radius 6,375 km	(m)
T_s	System noise temperature	(K)
λ	Wave length	(m)

Definitions and Abbreviations

AIS	Automatic identification system
AMSAT	Amateur radio satellite
AFSK	Audio frequency shift keying
AOCS	Attitude and orbit control systems
BER	Bit error rate
BPE	Bit-plan encoder
Bps	Bit per second
CanX	Canadian advanced nano-space experiment
CCSDS	Consultative committee for space data system
CDHS	Command & data handling subsystem
CMOS	Complementary metal oxide semiconductor
CNES	French space agency
COTS	Commercial of the shelf
DCT	Discrete cosine transform
DWT	Discrete wavelets transform
EO	Earth observation
ESA	European space agency
FSK	frequency shift keying
GMSK	Gaussian minimum shift keying
GPS	Global positioning system
GSD	Ground sampling distance
ISIS	Innovative solutions in space (Dutch company)
JAXA	Japan aerospace exploration agency
JPEG	Joint photographic experts group
KB	Kilo byte
MCU	Micro control unit
MEMS	Micro electro mechanical system
MHz	Megahertz
MSE	Mean square error
MSK	Minimum shift keying
NASA	National aeronautics and space administration
Non-ROI	Non region of interest
OBC	Onboard computer

P-POD	Poly-picosat orbital deplorer
Prom	Programmable read only memory
PSNR	Peak signal to noise ratio
RAM	Random access memory
RGB	Red, Green, Blue
RF	Radio frequency
ROI	Region of interest
RX	Receiver
SNR	Signal to noise ratio
SSTL	Surrey satellite technology limited
TC	Telecommand
TM	Telemetry
TNC	Terminal node control
TX	Transmitter
UHF	Ultra high frequency
VHF	Very high frequency

Declaration of Authorship

I, Warinthorn Kiadtikornthaweeyot declare that the thesis entitled

Onboard Computer technology for Cubesats

and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- none of this work has been published before submission, or parts of this work have been published as:

  1. W. Kiadtikornthaweeyot and A. Tatnall, (2010), "Cubesat Satellite Telemetry, Telecommand, Data Handling and Processing", *5th ESA International Workshop on Tracking, Telemetry and Command Systems for Space Applications*, 21-23 September 2010, ESTEC, Noordwijk, The Netherlands.

  2. W. Kiadtikornthaweeyot and A. Tatnall, (2011), "A highly efficient image compression system for a picosatellite", *28th ISTS (The 28th International Symposium on Space Technology and Science)*, 5-12 June 2011, Okinawa, Japan.

  3. W. Kiadtikornthaweeyot and A. Tatnall, (2012), "Adaptive Image Compression For Cubesats", *In 4s Symposium (Small Satellite Systems & Services Symposium),* 4-8 June 2012, Portoroz, Slovenia.

  4. W. Kiadtikornthaweeyot and A. Tatnall, (2013), "Alternative Image Compression for Cubesat", *5th European Cubesat Symposium*, 3-5 June 2013, Royal Military Academy, Brussels, Belgium.

Signed: ……..

Date: 18 June 2015

Acknowledgements

This thesis would not have been possible without the support of many people. First and foremost, I would like to express the deepest appreciation to my supervisor, Dr. Adrian Tatnall. He was very generous with his time and knowledge and assisted me in each step to complete the thesis. Without this dedicated support, advice and multitudes of patience, this work would have been impossible to finish.

I would like to express a deep gratitude to my family, father, mother, brother and my husband Martin Evans for their inspiring and untiring support and prayers.

Without the scholarship provided by Geo-Informatics and Space Technology Development Agency (GISTDA) this challenging learning path that I have been fortunate enough to journey would not have happened. Thank you for all the motivation and the budget provided to support me for the last four years.

I would like to thank every member of the Astronautics research group for helping, supporting and welcoming me here. It has been a great time working with you all.

To start the PhD programme was a big decision in my life, to be far away from home and going to a place where I knew no one. Along this journey, I have learnt a lot of new cultural experiences. I have made many new friends, which has made me realise that such nice people still exist. They have made me feel like I am not very far away from home. This journey has been very difficult and challenging but it has been such a great experience for my life and I know I made the right decision.

With sincere appreciation

Thank you all again from the bottom of my heart

Chapter 1

Introduction

The objective of this research is to analyse the onboard computer of miniaturised satellites, identify alternative architectures and examine the potential for onboard processing. Furthermore, this research focuses on the key technologies for spacecraft image compression. Particular attention will be paid to the cubesat; a type of miniaturised satellite (picosatellite) that usually has a volume of 10 x 10 x 10 cm and a mass of no more than one kilogram. Cubesats have successfully operated in orbit and have been used for a number of applications including Earth observation and the demonstration of space technologies.

Satellite project cost is related to satellite mass. In the past Earth observation satellites were expensive because their mass was high and they took a long time to design, build and launch (Sandau 2010). This has provided an opportunity for the small satellite sector to take advantage of Micro electro mechanical system (MEMS) technology, which has opened up the possibility of miniaturisation on cubesats (de Rooij, Gautsch et al. 2009; Sandau 2010). These MEMS devices enable reduction of the mass, power and also the cost of the mission (Bilhaut and Duraffourg 2009). The current development of very small satellites is focused on cubesats, which have a specific cubic shape and have been of interest to many universities and companies. The development of cubesat technology is focusing on two main points. Firstly miniaturisation, which enables the capability of cubesats to increase for the same mass and power. Secondly cost reduction, through using both "Commercial Off-The-Shelf (COTS)" technology (which enables innovative and cost cutting ways to develop the spacecraft) and also off-the-shelf components and instruments. However, various technical challenges exist, for example, providing high resolution imaging missions on small satellites (Kramer and Cracknell 2008) is currently impossible for cubesat designers although the cameras on the Planet Lab 3 unit satellites recently launched are said to have a resolution of between 3 and 5 meters.

The two main objectives of cubesat projects that have already flown are shown in Figure 1-1. Firstly education; the construction of a cubesat gives students the opportunity to be involved in design, to give them experience and technical skills in space technology and to work on a real space mission (Alminde and Laursen 2009). The second objective is the use of cubesats for the demonstration of new technologies in-orbit. For example Delfi-C3: a student

nanosatellite test-bed for the in-orbit demonstration of micro systems technology (Ubbels, Bonnema et al. 2005). Cubesat missions can be used to demonstrate a range of application areas. Clyde-Space defined 40 cubesat missions in to four categories (Greenland 2010; ClydeSpace 2012).

- Earth imaging; including COTS complementary metal-oxide-semiconductor (CMOS) cameras, dedicated processor and attitude determination algorithms.
- Novel communication including non-AX.25 protocol, antenna, redundancy link and new modulation techniques.
- Science; measuring airglow, earthquakes, atmospheric atomic oxygen and DNA (Deoxyribonucleic acid) denaturing, etc.
- Utility; ship (Automatic Identification System (AIS)) monitoring and data relay, technology demonstration testing, etc.

Currently, there is high interest in cubesat missions in many research and education areas (Bouwmeester and Guo 2010; Lucking, Colombo et al. 2012). There is also increased interest in high performance COTS computing for small spacecraft in many missions. This includes Earth observation missions that use high performance COTS technology to provide higher functioning with a lower mass and cost (Samson 2012). The transmission rate from a satellite is dependent on the amount of power available and the quantity of data to be downloaded to the receiving station. Pre-image processing and image compression to improve performance must be achieved with low power consumption, because of the limited power available on a cubesat. The research described in this thesis is to investigate and develop high performance onboard processing and pre-image processing in order to increase its operational capability and thus increase the applicability of cubesat Earth observation missions. Consequently, the research concentrates on technology demonstration and Earth imaging as shown in the red path in Figure 1-1.

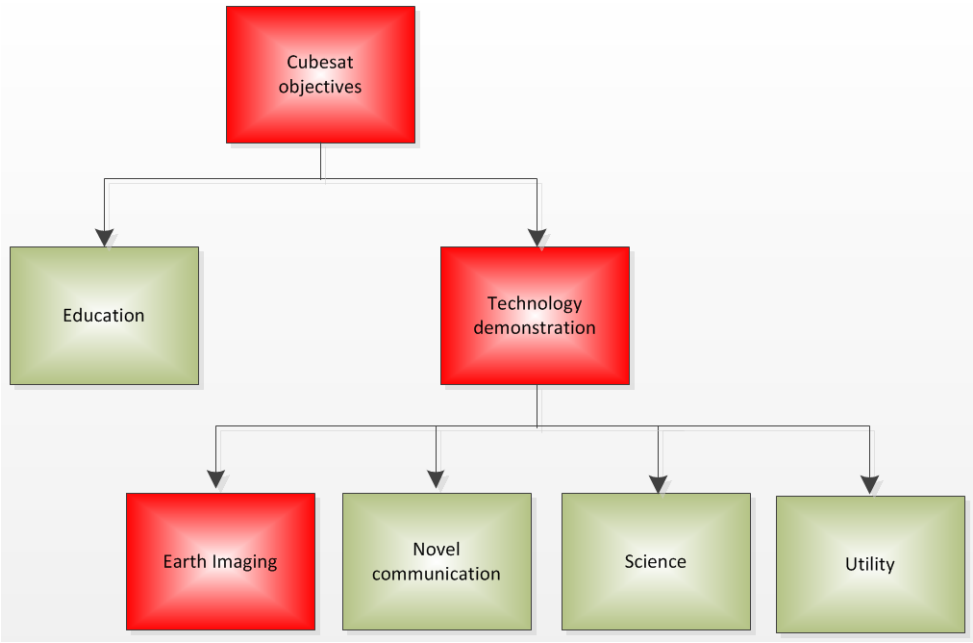


Figure 1-1 : Cubesat mission objectives and research position.

1.1 Motivation for research

The reduced amount of time and budget required for the development of small satellites has considerable advantages. The short time leads to data becoming available faster than from a larger satellite. The lower budget required gives the opportunity for countries or organisations with limited budget or little experience in space technology, to be able to have their own satellite to serve their countries and gain their experience in space industry. Due to the size of the cubesat, there are limitations on the power available and the data storage capacity. This places important constraints on the design of each subsystem. One of the key subsystems affected by these constraints is the communication subsystem. This impact is explored by considering the link budget in this research.

Design of the command and data handling subsystem for a cubesat involves many important constraints. These are low power usage, reliability, flexibility, simplicity to develop and automatic operation. It needs to be effectively managed to use the power available and to be compatible with the overall satellite power budget. The more flexible and reliable the architecture of the onboard computer, the higher the operational performance in space.

Many of the applications of cubesat missions involve imaging instruments. These instruments produce large amounts of data, and the increased resolution that is expected in the near future will further increase the data rate required. These will lead to the necessity to store

considerable amounts of data on-board. In order to modify the transmission time, to save power during transmission and to reduce the amount of onboard data, image processing such as image compression algorithms can be applied to the spacecraft onboard computer. This technique can reduce the image data size and transmission time (Yu 2009; Lara and Yunhong 2011; Faria, Fonseca et al. 2012).

The onboard computer technology used for a cubesat is a challenging area that must be developed to enable higher performance. Three related subsystems that interface with the onboard computer, i.e. the communication, the command & data handling and the image processing, will also be investigated in this research. Figure 1-2 shows the research diagram overview.

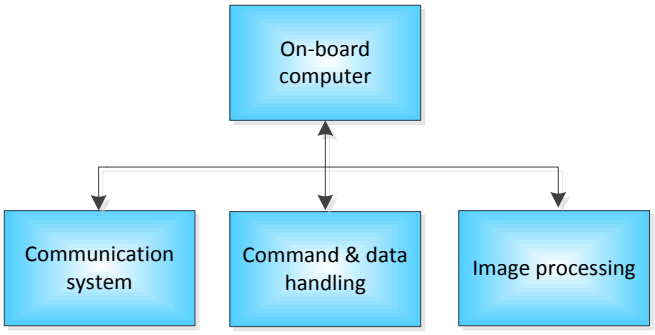


Figure 1-2 : Research diagram overview.

1.2 Research aims and objectives

The primary aim is to develop the onboard computer capability of a cubesat by investigating the onboard computer architecture and onboard processing. The research will focus on the design of a cubesat's onboard computer, which is capable of increasing both the system performance and improving flexibility for application in complex missions, whilst also reducing data transmission time. A proposed onboard computer architecture concept will be developed to meet this objective. In addition image processing onboard for cubesat is analysed and proposed.

1.3 Organisation of the thesis

This chapter outlines the motivation, objectives and scope of the research. The organisation of the thesis flowchart, shown in Figure 1-3, shows the relationship of the work

described in each chapter. There are eight chapters as follows: Chapter 1 introduces the work performed in this research. Chapter 2 presents the literature survey of the related research domains. The survey focuses on the limitations of the main subsystems of a cubesat and discusses methods to improve the operating performance. This chapter will therefore begin by introducing cubesat technology and then the communication system and onboard computer system. The techniques of region of interest identification and image compression will then be discussed. Chapter 3 presents the radio frequency (RF) communication and link budget calculation. Chapter 4 analyses the onboard computer for cubesats and discusses its architecture. This leads on to a proposal and a discussion of a new onboard computer architecture. Chapter 5 presents the region of interest automatic detection and considers several methods of image segmentation. Chapter 6 describes the wavelet image compression. Chapter 7 describes the proposed system, which is an adaptive image compression algorithm for cubesats. In addition the experiment and results are presented in this chapter. Finally, the future work and conclusions of the research are discussed in Chapter 8.

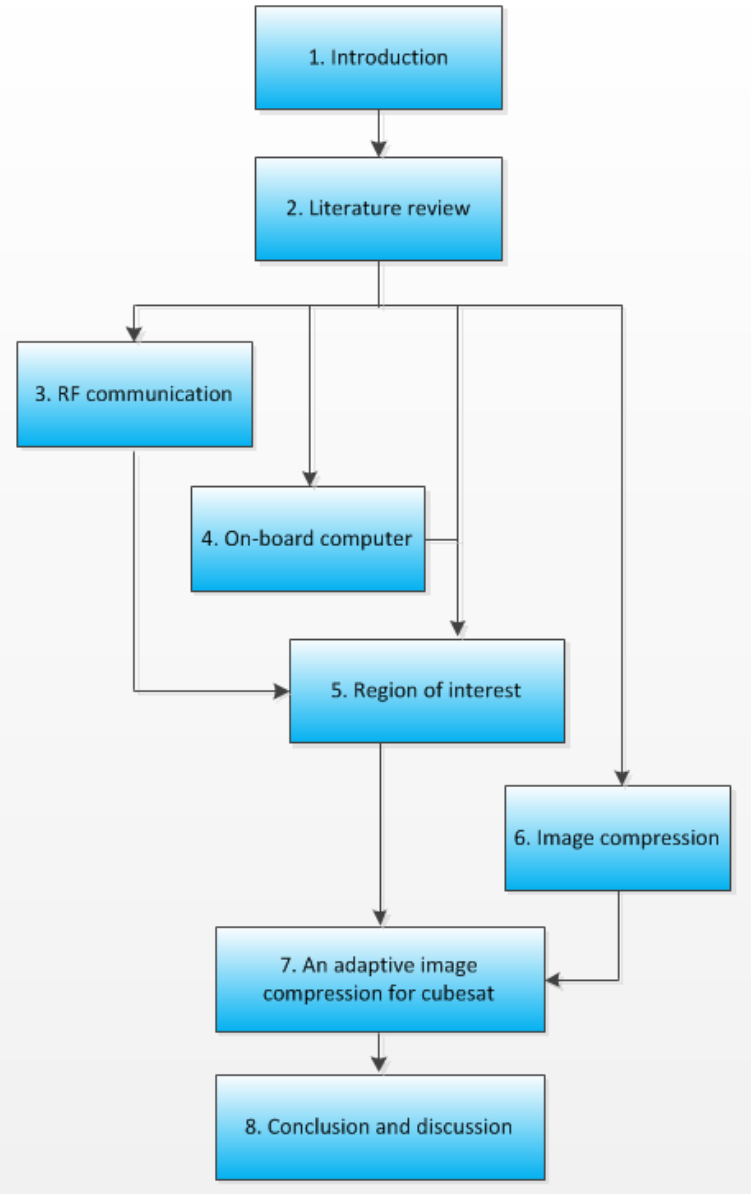


Figure 1-3 : Thesis organisation flow chart.

1.4 Summary

This chapter has provided the necessary perspective to the research. In particular the motivation for the research, research objectives and scope of work have been detailed. The following chapter presents a literature review of related research.

Chapter 2

Literature review

The research objective is to examine the design of the cubesat onboard computer, and to investigate ways to improve performance. This includes an examination of the onboard architecture and the image processing that could be implemented on a cubesat. This is related, in general, to the communication, onboard computer and image processing subsystems of Earth observation satellites.

2.1 The Earth observation satellite

The need for Earth observation (EO) information to support the understanding and planning of global changes to our planet is increasing rapidly. It is only satellites that provide the global coverage required to understand global issues. Small satellites are powerful devices to obtain Earth information, are cost effective and provide an opportunity to access information quicker than for conventional satellites (Briess 2010).

Electronic technologies have developed significantly and provide the opportunity for developing countries, universities and companies to have their own satellite. Satellite size can now be smaller for the same operational performance (Karatas and Ince 2009).

The small satellites for the EO missions have a variety of different instruments. Most of these are visible and infrared camera or spectrometers with different wavelengths. The different combinations of spectral bands used depends on the different objectives of the mission. For example;

- A “true colour” image would consist of red, green and blue channels. This is particularly suitable for analysing objects as it replicates human vision.
- The green, red and infrared channels are a particularly useful for applications involving vegetation, as this is particularly sensitive to infrared.
- Blue, Near-Infrared (NIR) and Mid-Infrared (MIR) channels would allow the determination of water depth, vegetation coverage, soil moisture content and the presence of fires.

An Earth observation image system can classify objects in an image using the information below (Mather 2004);

- A monospectral or panchromatic imaging system; the image is a single channel detector that can measure within a broad wavelength range. It is characterised by high spatial resolution. Satellite Pour l'Observation de la Terre (SPOT) carry a panchromatic instrument as one of several satellite sensors flown.
- A multispectral imaging system; this system consists of measurements at several discrete spectral bands e.g. SPOT, Landsat multispectral scanner sensor (MSS) and Landsat thematic mapper (TM) super spectral imaging systems; in general this imaging system is composed of many more spectral channels that multispectral systems. This system consists of tens of spectral bands. Examples of satellites that have this type of imaging system are Moderate resolution imaging spectrometer (MODIS) and Medium resolution imaging spectrometer (MERIS).
- Hyperspectral imaging systems; these systems combine hundreds of spectral bands to provide a continuous spectrum. An example of this type of instrument is the Hyperion instrument on the EO1 satellite. Potential applications of this instrument are monitoring agriculture and the health and maturity of crops.

A large variety of instruments and sensing principles are used in Earth observation. They include;

- Optical and infrared (e.g. SPOT, Landsat, MODIS and WorldView2 etc.)
  - Camera
  - Multiband scanner
  - Imaging spectrometer
  - Light detection and ranging (LIDAR)
- Microwave and radiowave (e.g ERS1/2, Alos Palsar and Radarsat etc.)
  - Radiometer
  - Radio detection and ranging (RADAR)

Earth applications can be classified into the groups shown below (Chuvieco 2008; Pearlman, Holt et al. 2008; Altan 2011);

- Agriculture
  - Forest Mapping and Change Monitoring
  - Training Modules for Agriculture
  - Improving Measurements of Biomass
  - Agricultural Risk Management

Operational Agricultural Monitoring System

- Climate — Sustained Reprocessing and Reanalysis Efforts
  Key Climate Data from Satellite Systems
  Key Terrestrial Observations for Climate
  Global Ocean Observation System
- Disasters — Tsunami Early Warning System at Global Level
  Fire Warning System at Global Level
  Risk Management for Floods
  Risk Management for Earthquakes
  Risk Management for Landslides
  Risk Management for Volcanoes
- Energy — Using New Observation Systems for Energy
  Management of Energy Sources
  Energy Environmental Impact Monitoring
- Health — Forecast Health Hazards
  Environment and Health Monitoring and Modelling
  Integrated Atmospheric Pollution Monitoring, Modelling and Forecasting
- Water — In-situ Water Cycle Monitoring
  Water Resource Management
  Global Water Quality Monitoring
- Weather — Surface-based Global Observing System for Weather
  Space-based Global Observing System for Weather
  Numerical Weather-Prediction Capacity Building
  Etc.

The satellite orbit selected depends on the objectives of the instrument of the satellite. The orbit parameters selected such as altitude, shape and inclination are related to the required performance of the instruments (Mather 2004). The main considerations are the radiometric sensitivity, revisit time and the resolution of the image. Most Earth observation satellite orbits are in near polar low Earth orbits with an inclination of about 98 degrees (Mather 2004) (Figure 2-1). Consequently the satellite is able to cover most of the Earth's surface in some period of time.

A sun-synchronous orbit is frequently selected for an Earth observation satellite as this leads to the satellite observing the ground at the same local solar time every day (Floyd F.

Sabins 1997; Mather 2004). The nearly consistent surface illumination angle is useful for a variety of instruments that rely on visible/IR sun reflected radiation.

Figure 2-1 : Sun-synchronous orbit

Small satellites are powerful devices to get Earth information, are cost effective and provide an opportunity to access information quicker than for conventional satellites (Briess 2010).

Electronic technologies have developed significantly and provide the opportunity for developing countries, universities and companies to have their own satellites. Satellite size can now be smaller for the same operational performance (Karatas and Ince 2009).

2.2 Cubesat technology

Small satellites have demonstrated that space missions can be conducted at a much lower cost than has been achieved in the past (Puig-suari, Turner et al. 2001). The most important feature of very small satellite projects, like cubesats, is mission development time (approximately 6 – 36 months) which is much lower than conventional missions. The low cost of development and launch has opened up the possibility of using cubesats as valuable education tools. The typical development time of cubesats makes them suitable for students to be involved in from the beginning of the funding until delivery to space. In 1999, California Polytechnic State University and Stanford University introduced the cubesat standard (Puig-suari, Turner et al. 2001). Since this introduction the number of pico- and nano- satellites has increased massively, particularly at universities (Barnhart, Vladimirova et al. 2007; Bouwmeester and Guo 2010). More than 300 cubesats have now been launched, with the launch rate exceeding 100 per year (Reinhard 2013). Due to the standard structure of cubesats, cubesat subsystems (which are space qualified) are available from specialised cubesat shops (Greenland 2010).

The limitations of power, volume, mass, data rate and attitude control precision on cubesats have restricted their use outside educational establishments and developing countries. Recent miniaturisation technologies have provided major advantages to cubesat developers, and enabled them to reduce the effects of these limitations. Examples of miniaturisation technologies include new propulsion, s-band, x-band telecommunications, solar panels, miniaturised star trackers, de-orbiting, light-weight materials etc. (Bouwmeester and Guo 2010; Selva and Krejci 2012; Klofas and Leveque 2013; Reinhard 2013).

Klofas (Klofas and Leveque 2013) summarises cubesat missions from June 2003 to February 2013. During these ten years, 79 cubesats have been launched from 18 different launchers. There have been some failures of launchers carrying a number of cubesats, e.g. Dnepr-1, Falcon-1 and ELaNa-1. The status of the cubesats as of April 2013 is shown in Figure 2-2. There are six different categories (Klofas 2013);

- DOA: Dead on arrival, the satellite was never heard from in space
- Deorbited: The spacecraft has deorbited
- Dead: Spacecraft is now no longer transmitting
- Alive: Satellite is beaconing data, but not achieving its mission, or the mission has ended
- Active: Cubesat is performing its intended mission
- Unknown

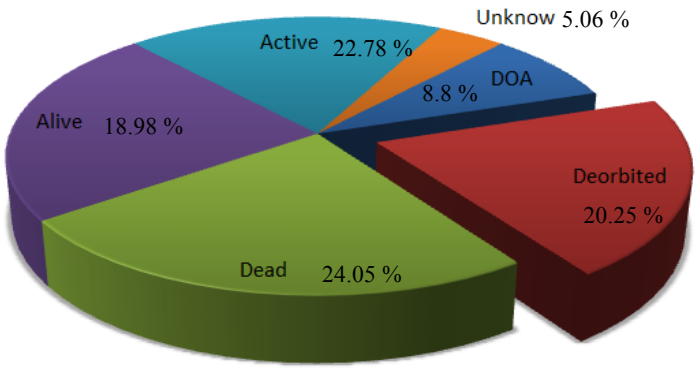


Figure 2-2 : Status of Cubesats at April 2013.

Their services are classified into amateur (71), experimental (17), government (10) and other (10), which includes, Earth exploration, meteorological, space research and unknown. Some of them provide more than one service. Figure 2-3 shows the number of cubesats, separated into four different standard size structures. There are 48, 6, 3 and 22 cubesats made up of 1, 1.5, 2 and 3 standard units respectively.

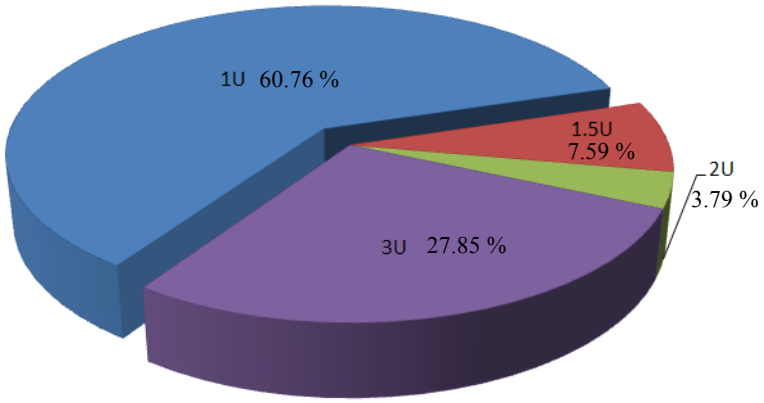


Figure 2-3 : Number of cubesat in standard sizes.

Figure 2-4 shows the geographical spread of pico- and nano- satellite developers. The graph noticeably shows that the USA is the greatest developer in this field followed by Japan (Bouwmeester and Guo 2010).

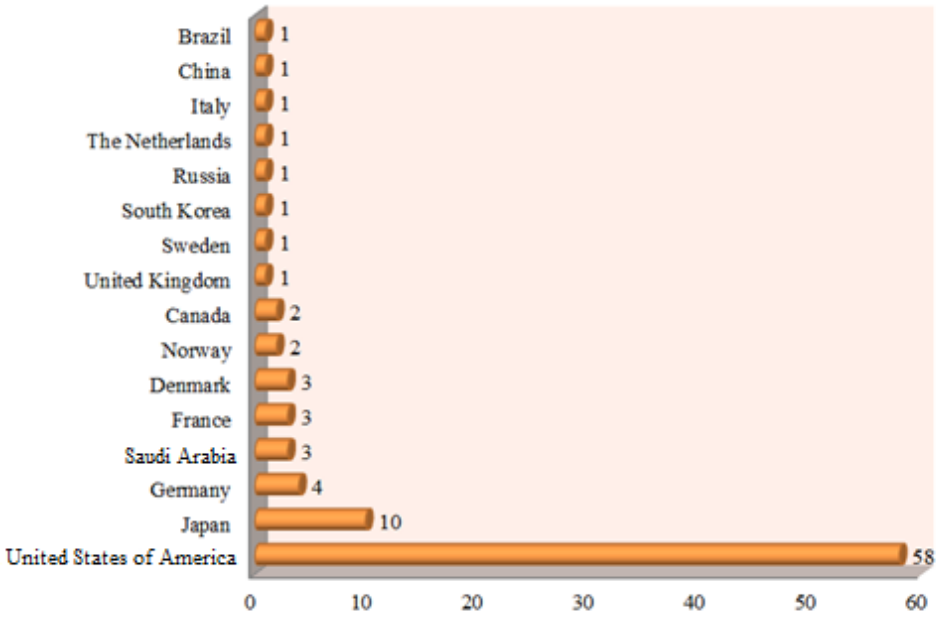


Figure 2-4 : Geographical distribution of pico- and nano- satellite developers (Bouwmeester et al. 2010).

Figure 2-5 shows the number of nano and micro- satellites (1 - 50 kg) launched and expected to be launched from 2000 to 2020 by Space Works Commercial company (Langton

2002). The information contains the currently known past and future nano- and micro- satellite launches into orbit. The blue bars show the past satellites that have been launched and the red bars show the known future launches. The bar graph shows that nano- and micro- satellite launch demand has grown since 2006 and is expected to rapidly increase from the year 2013. The red bar is the future projection based on the announced plans of nano- and micro- satellites. It is expected that annual launch demand will increase to over 188 globally in 2020 (Langton 2002).

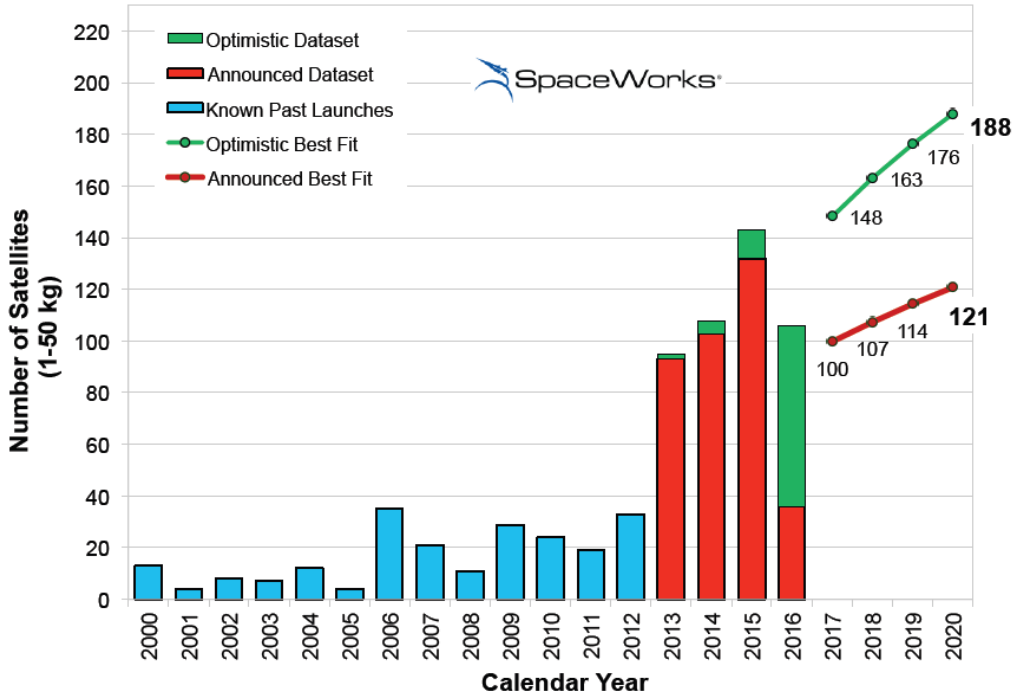


Figure 2-5 : Nano- & micro- satellite launch demand history and projections (SpaceWorks Enterprises, Inc., 2013).

A most interesting development is the QB50 project. It is a network of 50 cubesats, with the main mission to make in-situ, multi-point measurements of the lower thermosphere region for the 3 months period lifetime of the network. The 50 cubesats will be launched into orbit at 320 km altitude on a single rocket in 2015. There are about 40 atmospheric 2U cubesats and about 10 of 2U or 3U cubesats for science and technology demonstration. By measuring atmospheric parameters, information about re-entry will be obtained. The 50 cubesats will be provided by universities from all over the world. Atmospheric models and users will benefit from the measurement data obtained (Smith 2012; Asma 2013).

The cubesat definition is a picosatellite of 10 x 10 x 10 cm^3 with a maximum mass of 1 kg. Figure 2-6 illustrates the cubesat standard structure (Greenland 2010). Lightweight materials are used for the structure to enable the mass limitation to be achieved. The reason for developing the cubesat shape in the form of a cube is so that it can fit in standard launcher systems, P-POD (Poly-picosat orbital deployer).

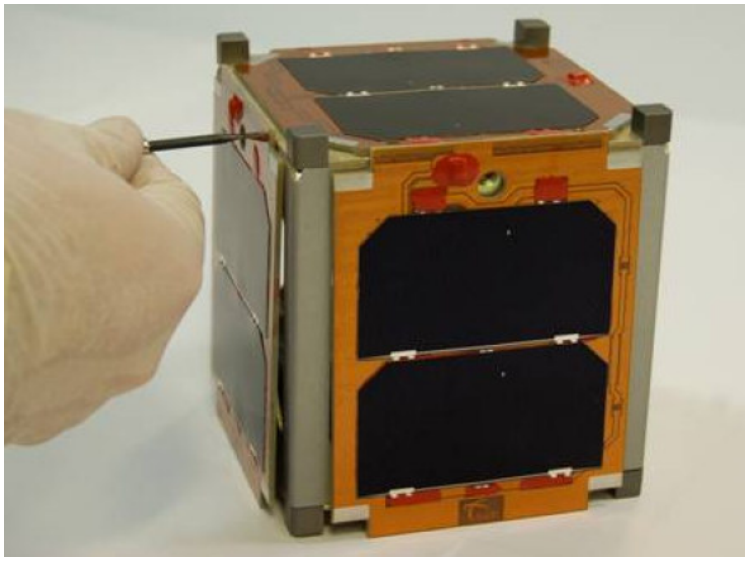


Figure 2-6 : Cubesat standard structure (Greenland 2010).

Data from a cubesat is usually transmitted using the amateur radio channel. The AX.25 is the standard protocol of communication by amateur radio. It is reliable and has a simple protocol (Huerta 2006).

Cubesat constraints include size limitations and actual physical space available for the payload and solar cells. Power is restricted as it is a function of the surface area of the solar panels. If solar cells are body mounted, the limited size of the cubesat structure will restrict the amount of power that can be generated. Deployed arrays increase the power available at the expense of increased complexity and reliability. The average power of a body mounted cubesat array ranges from 10 mW to 7 W (Bouwmeester and Guo 2010). Consequently there is very limited power to be distributed to many applications such as; onboard computer, communication and payload subsystems. These limitations can lead to a slow communication link and the absence of any redundancy within the system. In addition to the above limitations, cubesats may also have other limitations when built by universities. There are problems such as lack of time to build, lack of students with satellite experience and insufficient financial support. Nevertheless, there are many cubesat satellites which have considerable complexity. Delfi-C3

satellite is one such cubesat, the design of Delfi C3 has been an important discipline at Delft University of Technology. It was launched in April 2008 and still operates in orbit. It demonstrates three different experiments; novel thin film solar cells, an autonomous wireless sun sensor and an advanced transceiver (Delfi-C3-team 2005; Jansen, Reinders et al. 2010).

MEMS devices are highly suitable to fit inside cubesats. There have been many MEMS components that have been designed, developed and tested for satellite subsystems. For instance, radio frequency, switching MEMS technologies for communication systems, micro propulsion for attitude and orbit control systems and optical MEMS for small photonic systems. MEMS technology has been used in space for many applications (de Rooij, Gautsch et al. 2009; Shea 2009).

Recent research has shown that cubesats technologies have the potential to be developed for Earth observation missions that are currently being flown by much larger spacecraft (Selva and Krejci 2012). Currently, environmental monitoring is essential for determining global status and disaster monitoring. The Earth observation satellite helps to gather data on the Earth's resources and environment (Sandau 2010). There are various types of imaging devices covering the visible and the infrared spectrum (Simmons 2008). The spatial resolution in geographic information systems is often equated to the GSD, which is the pixel spacing on the Earth's surface. The image GSD can be in the range between a few centimetres to around one hundred meters.

Figure 2-7 demonstrates the different application areas using Earth observation images with a wide variability in spatial and spectral resolution. For instance, agriculture (light green) requires Earth observation images to have a resolution of between 2-100 m (Sandau 2010).

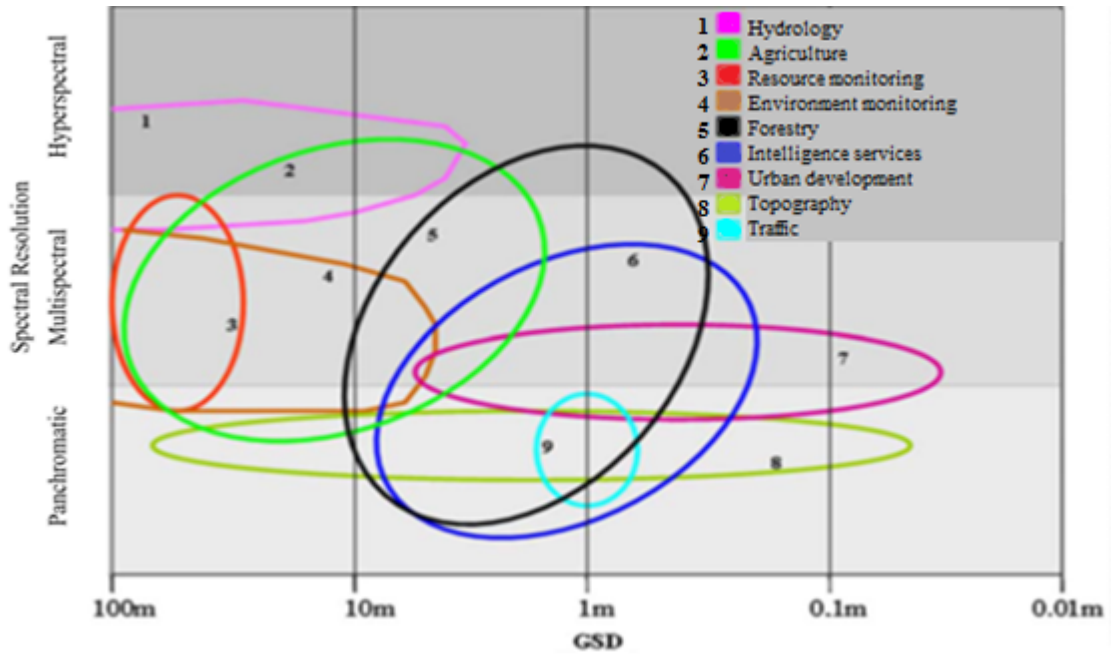


Figure 2-7 : Earth observation image - GSD vs. spectral resolution, and the requirement in different applications (Sandau 2010).

There are many previous cubesat projects that have been designed for Earth monitoring using cameras, for instance CanX-1, SwissCube, AAU-Cubesat, Compass-1, Cubesat XI-V and AeroCube-2 (Braegen, Hayward et al. 2007).

The cubesat community is increasing rapidly. Cubesats have allowed access to space for many universities and small countries. Since a cubesat was defined, the twin attractions of low cost and short development time have led to many satellites being developed and launched (Reinhard 2013). Figure 2-8 is a bar chart summarising the cubesat missions chronologically for each launch from May 2009 to September 2012 (Klofas 2013). This section describes some of the main cubesat missions, which have flown cameras.

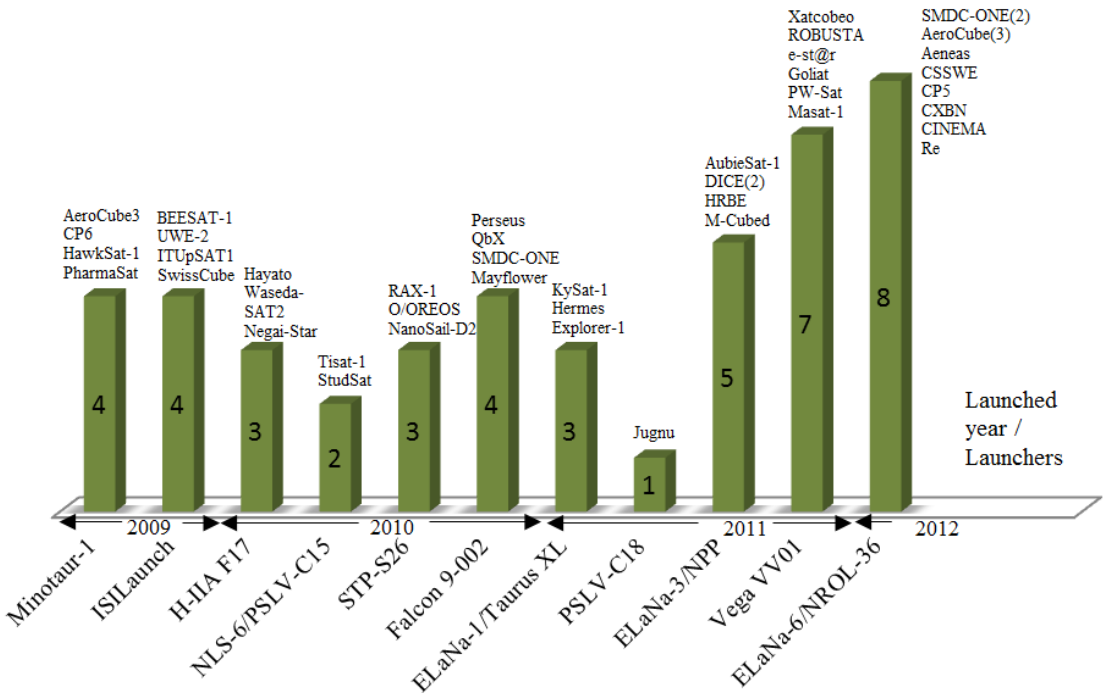


Figure 2-8 : Launcher and launch year of cubesats between May 2006 to September 2012.

2.2.1 Cubesat missions

2.2.1.1 BEESAT-1

BEESAT-1 was launched on 23 September 2009 from Sriharikota, India by PSLV-C14 (Polar satellite launch vehicle) rocket (Klofas 2013). The main objective of BEESAT-1 was to demonstrate a new micro reaction wheel in orbit. It was the first cubesat from the University of Berlin. BEESAT-1 included an onboard camera, which was used to take an Earth image and then downlink data to the ground station. The image is 640x480 pixels in size (Klofas 2013). The onboard processor uses an NXP LPC2292 at 60 MHz, RAM 2 MB and 20 MB flash memory. It is now in operation and in contact with the ground station at Berlin (Brieß 2011; Klofas and Leveque 2013; Rupprecht 2013). Figure 2-9 shows an artists impression of BEESAT-1 in orbit (Rupprecht 2013).

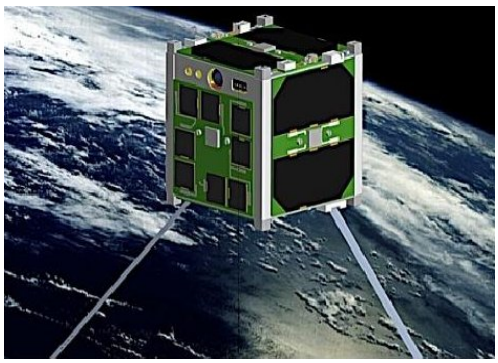


Figure 2-9 : BEESAT-1 (Rupprecht 2013).

In 2013 there were launches of the follow-on missions Beesat-2 and 3.

2.2.1.2 SwissCube

SwissCube was launched by PSLV-C14 with BEESAT-1. SwissCube was developed at the Ecole polytechnique fédérale de Lausanne (EPFL) in collaboration with engineering schools, universities, and private firms in Switzerland. The mission is focusing on the observation of the airglow phenomena, photoluminescence of the atmosphere and oxygen emission (Borgeaud, Scheidegger et al. 2010; Grosse 2011). The main payload of SwissCube is a telescope, which is integrated with a 188 x 120 pixel CMOS detector with a pixel size of 24 μm to capture images of the airglow emission (Klofas and Leveque 2013). There are two communication links. Firstly, a high power and high data rate RF (radio frequency) link using FSK (frequency shift keying) at a frequency at 437 MHz. The second communication is via a low power beacon. The uplink uses a frequency of 145 MHz and AFSK (audio frequency shift keying) modulation (Muriel, Fabien et al. 2009). Figure 2-10 shows an artist's impression of SwissCube observing airglow phenomena (image credit ; EPFL), (ESA 2013).

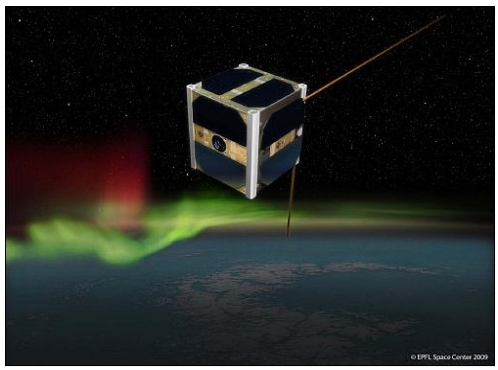


Figure 2-10 : Artist's view of SwissCube observing airglow phenomena (image credit; EPFL) (ESA 2013).

2.2.1.3 StudSat

StudSat was launched by PSLV-C15 on 12 July 2010. The project mission was the education of students and the promotion of space technology into the education system. StudSat was under the guidance of ISRO (Indian space research organisation) (Jakhar 2012). The payload was a CMOS imager with a 90 meters ground resolution. An Atmel 32-bit - UC3A0512 was the main onboard microcontroller. The transmitter was operated at 437.505 MHz under an amateur license for both uplink and downlink. The modulation FSK technique was used with 500 mW of power output. The beacons were received by amateur radio operators around the world. Unfortunately, the cubesat never received any uplink command due to a problem with an on orbit propagator and spacecraft receiver. Figure 2.11 shows the StudSat cubesat (ESA 2013).

Figure 2-11 : Photo of the StudSat-1 Cubesat (Image credit: Team StudSat) (ESA 2013).

2.2.1.4 Masat-1

Masat-1 was launched on 13 February 2012 from Kourou, French Guiana by Vega VV01 rocket and was the first Hungarian cubesat. The primary mission of the project is student education. Masat-1 was developed by students at the Budapest University of Technology and Economics in Hungary. The payload is a visible camera, 640x480 pixel image with a variable ground resolution between 2 to 10 km (Dudás 2009). Figure 2.12 shows Masat-1 (ESA 2013). Figure 2-13 shows the Earth surface taken by Masat-1 (ESA 2012).

Figure 2-12 : Masat-1 (ESA 2013).

Figure 2-13 : Western Australia photographed by Masat-1 (ESA 2012).

2.2.1.5 AeroCube-3

AeroCube-3 was built by the Aerospace Corporation. It consisted of several payloads, imagers and a deorbit balloon. AeroCube-3 contains a freewave technology frequency-hopping at 915 MHz radio and the modification of the link delay parameter as the communication subsystem. A surface-mounted patch antenna is integrated into this cubesat. The primary ground station is in the Aerospace Corporation in EL Segundo, California, which had received 52 MB of data from the AeroCube-3. The deorbit balloon did deploy but it did not inflate (Klofas and Leveque 2013). Figure 2-14 shows an image taken by AeroCube-3 (Hinkley 2010).

Figure 2-14: AeroCube-3's Earth photo (Hinkley 2010).

2.2.1.6 CanX-1

CanX-1 is the Canadian advanced nanospace eXperiment (CanX) and was developed by the space flight laboratory at the University of Toronto institute for aerospace studies. It was the first Canadian cubesat project (Polaschegg 2005). The main objectives of this project were research and education. CanX-1 was launched in 2003 with the mission of verifying novel electronic technologies in space (Stras, Kekez et al. 2003). CanX-1's mission was to test a number of technologies in space, this included a star tracking experiment using a CMOS imager, an active three axis magnetic stabilisation, a Global positioning system (GPS)-base determination and an ARM7 central processor. The onboard computer of CanX-1 was custom-

designed (Stras, Kekez et al. 2003). The cubesat was built to cubesat standard specifications and implemented with colour and monochrome CMOS imagers for imaging star fields, the moon and the Earth. In addition a global positioning system receiver and an active magnetic control system were included as an extra payload. It was deployed into space at an altitude of 650 km. Further satellites in the same nano-satellite program include CanX-2, CanX-3, CanX-4 and CanX-5.

2.2.2 Cubesat and Earth observation

The use of cubesats in space has become a popular topic. Several cubsats carry low resolution imaging instruments as the primary or secondary payloads such as KUTESat Pathfinder, ICECube1 and AAUSat-2 Explorer-1 Prime, Goliat, UniCubesat, etc.

A large number of cubesats have been developed to carry space weather sensors with modest resolution e.g. the optical camera on SwissCube-1. This passive optical instrument is to measure oxygen emission in the upper atmosphere.

A constellation of satellites provides more frequency access to regions of interest on the Earth surface and allows almost continuous monitoring (Karatas and Ince 2009). The very low cost of cubesats provides the opportunity to have satellite constellations (Marshall 2014). The world's largest constellation of Earth imaging satellites is FLOCK1. This mission consists of many cubesats and was developed and operated by Planet labs in San Francisco. Since the mission began, Planet labs has launched constellations of small satellites to the ISS orbit and sun-synchronous orbit. They have launched 78 satellites within 23 months, but only 20 remain in orbit due to the low ISS orbit used for most of them, which results in their decay within about a year. Eleven of the constellation were launched on Dnepr into Sun-synchronous orbit at 620 km, hence have longer lifetimes and polar coverage. They provide imagery with a resolution of 3-5 m and operate in a short lived orbit at 400 km and an inclination of 52°. Each cubesat mass is approximately 5 kg. The main applications of FLOCK1 are in the areas of disaster monitoring; improved agriculture and monitoring the change of polar ice caps. Images can also possible be used in oil monitoring and gas installation. Moreover the satellites downlink frequency is 8025-8400 MHz for remote sensing. Whereas to receive the command signal is in the 2025-2011 MHz. The mission provides unique capabilities particularly the revisit time.

Cubesats are excellent spacecraft for education and technology. Cubesat technologies will become increasingly significant in the next few years and provide inspiration for further applications.

2.2.3 Cubesat instruments

In this section instruments that have flown on a variety of earth observation cubesats are described.

- **Multispectral imaging**

Pumpkin MISC refractive

Pumpkin was one of the first developers of cubesats. The company created the CubeSat Kit ™ (CSK), colony spacecraft buses and spaceflight proven solutions for low-cost LEO mission. Pumpkin have provided their design and service to build cubesats for many governments, companies and universities (Pumpkin 2011). A deployable solar array has been designed to be able to be pluggable into the CSK bus and has been flown by universities and industry. Pumpkin have also built optics and imaging systems for cubesats (Kalman 2011).

Pumpkin‘s Miniature Imaging Spacecraft (MISC) is a 3U cubesat. MISC combines a COTS nanosatellite device with an imager payload. The MISC is designed to be able to operate for around 18 months. At a satellite altitude of 540 Km, the performance of the MISC imager is (Kalman and Reif 2007);

- GSD is approximately 7.5 m
- Ground area per image 702 km^2
- Image size 16 MP
- Focal Length 600 mm
- Pixel Size 7.4 mm x 7.4 mm
- FOV 4 degree

Currently the MISC design provides the capability to download the data at 38.4 kbps or 57.6 kbps (AstroDev Helium-10 UHF/VHF radio). With this downlink performance, the cubesat can download around 196 images per day (Kalman and Reif 2007).

Aalto-1 ; VTT multi-spectral

The Aalto-1 is a student built 3U cubesat built by Aalto University in Finland (A. Kestilä 2012). The main objective is to demonstrate the feasibility of an imaging spectrometer to observe the Earth environment (Hanninen 2012). The main payload of Aalto-1 is a very small spectrometer but it also includes a very small radiation detector and a deorbiting device. The payloads are demonstration technology.

The spectrometer has been developed at the VTT technical research centre of Finland. This payload is based on a tunable FPE (Fabry-Pérot Interferometer), and is capable of measuring 2D spatial images. The instrument specification parameters are shown below (Holmlund, Heikki Saari et al. 2012; K. Viherkanto 2013);

- The instrument size is 9.7 cm x 9.7 cm x 4.8 cm with a mass of 500 g.
- The camera comprises a CMOS sensor with 2048 x 2048 pixels.
- The reference altitude is 700 km, and the ground pixel size is approximately 240 m x 240 m.
- The instrument is multispectral and can select wavelength bands from 500-900 nm.
- The Field of View (FOV) is 10° x 10° degree.
- The SNR is less than 50

A VIS camera has been integrated with this instrument to verify spectral image targets (K. Viherkanto 2013).

The Aalto-1 transmitter operates at S-band frequency by using the Texas Instruments TI CC2500 transceiver. The transceiver provides a data rate of 1 Mbit/s. The maximum power consumption is 4.2 W. Approximately 41.6 Mb of data can be downloaded to the station per pass.

- **Hyperspectral Imaging**

Hyperspectral imaging is a powerful technique to observe the Earth surface in different ways (Robinson 2011). A visualisation of the technique is shown in Figure 2-15. There are many that can benefit from hyperspectral imaging including military, geology, forestry, agricultures, atmospheric, environmental, land cover and ocean applications (Robinson 2011). Hyperspectral cameras have been implemented on a small satellite, e.g. Tacsat-3 has a payload of approximately 170 kg (Robinson 2011). Technology is being developed rapidly to simplify the size and mass of the instrument and enable possible implementation on a cubesat.

Over the past few years hyperspectral data has become of increasing interest (Christophe et al, 2005). It is a challenge to achieve excellent optical performance in very small spacecraft such as cubesats. There are numerous design and development activities ongoing to implement hyperspectral imaging on cubesat (HyperCube) for remote sensing missions (Roger, 2015) of which there are numerous applications (Ian, 2001)(ESA, 2014). Implementation of hyperspectral data on cubesats has constraints on power consumption and storage due to the large data storage requirements (Daniel and David, 2012). Therefore the HyperCube is being developed by ESA (ESA, 2014).

Advances in the technology of electronics and materials also give increased chance for miniaturising payloads like HyperCube (ESA, 2014).

The AALTO-1 is an example of a nano satellite for hyperspectral remote sensing developed by students in Finland, which is 3 units in size (34 cm x 10 cm x 10 cm) with a mass less than 4 kg. The main payload of the satellite is a novel imaging spectrometer (Jann et al, 2013) and it is the world's smallest remote sensing imaging spectrometer for earth observation. The target to be achieved for signal-to-noise ratio (SNR) of 50 and a spatial resolution of 240 m. The spectral resolution will be in between 10 to 30 nm. Also the onboard computer has been developed in-house by using a Linux real-time operating system with a double-redundant design (Kestila et al, 2013).

Currently ESA, under the General Support Technology Programme funding, is developing a hyperspectral instrument that can be implemented on a cubesat (ESA 2014). The main developments required for an instrument of this type are low power focal plane assembly and electronics, a low power and mass cooling system and capability to provide more than 300 Mbit/sec (Robinson 2011).

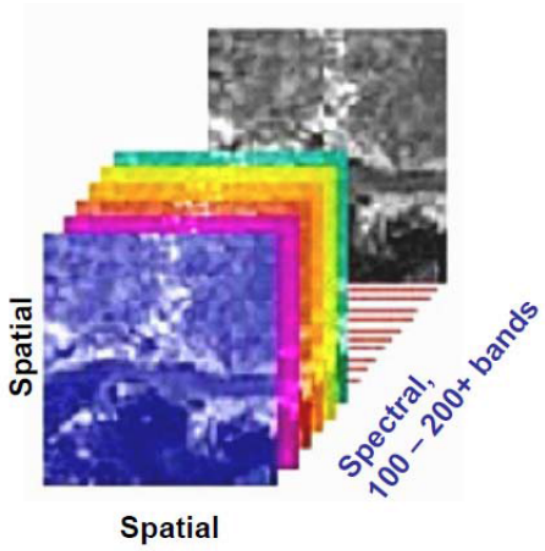


Figure 2-15 : Hyperspectral imaging overview (Robinson 2011).

- **Planet Labs Dove imager**

The instrumentation on the Planet Labs consists of a Maksutove-Cassegrain telescope with a 91 mm aperture. The relatively large focal length of 1.14 m requires the focal array to be placed 32 cm from the telescope aperture at the very rear of the

telescope. The design relies on MEMS and hyper-integration based on the use of common resources and an absence of the usual box structure that can usually be found on spacecraft. An example of an image from Planet labs is shown in Figure 2-16 (Cowing 2013).

Figure 2-16 : Image taken by Planet Labs dover 2.

2.2.4 State of the art and advances in cubesat technologies

Cubesat spacecraft can be flown as for example 1U, 1.5U, 2U, 3U or 6U systems. A strong feature of the cubesat concept is flexibility within the scope of the cubesat standard definition, which provides more opportunity for users to design and develop.

- **Attitude determination and control**

The purpose of this subsystem is to give the accurate orientation of the cubesat to enable pointing of the instrument and the solar array to generate the energy required and to operate the payload. The state of the art devices are passive with three axis stabilization by magnetic coils (Brieß 2009).

With current technology available, the miniaturized sun sensor gives the top attitude determination performance and normally it will operate with a magnetometer (Selva and Krejci 2012). For example CanX-2 has been shown to achieve accuracies better than 1.5° (Kahr, Montenbruck et al. 2011).

There are now star trackers being developed (e.g. ST-200 Pico-Tracker) and it should be possible to provide an accuracy of 0.01° for yaw and pitch and 0.05° for roll (Segert, Engelen et al. 2011). This improvement will enable an accuracy compatible with the geolocation requirements. Selva et al (Selva and Krejci 2012) state that "*future determination accuracy devices are possible that will make the system more accurate. Assuming a satellite altitude at 500 km (ex, Canx-2) and an attitude determination at 2°, this will lead to a spatial uncertainty of around 17.5 km. In the near future, the technology development will achieve the accuracy at 0.02°. This would give spatial on ground around 175 meters".*

The active control of a spacecraft is potentially complex and expensive. Passive magnetic control using a bar magnet is a very effective and simple way for attitude control of a cubesat which are mostly in LEO (Bouwmeester and Guo 2010). However, the pointing control accuracy and particularly the stability provided by passive magnets during the image exposure time are not sufficient for CubeSats with optical payloads, and this is the case also for magnetotorquers. In general, mini reaction wheels are required to meet the pointing stability needs of the instruments

Pointing control of cubesats has been challenging due mainly to volumetric constraints and lack of satisfactorily small attitude sensing and control components. Consequently most cubesat control designs implement an effective magnetic control, although there still remains large attitude errors (Daniel and David, 2012). As mission complexity and payload demands increase, highly accurate pointing control is increasingly required for cubesat (James et al, 2009) to enable better image quality. A COTs 3-axis mini reaction wheel has now been developed which is an improvement on existing technologies and this can be used in missions such as Aalt0-1 (Kestila et al, 2013) (Pumpkin, 2011).

- **Onboard processing**

Many small spacecraft platforms use COTS onboard components for quicker advances and shorter qualification timelines (NASA, 2014). Onboard computer technologies have benefited from advances in commercial industries and now onboard processing is included on cubesats. Nowadays the onboard computer systems have superior processing capability with lower mass, power and volume requirements. This gives the cubesat community the ability to develop cubesat technology quicker and apply this to a broader range of missions.

The current onboard hardware commercially available are PIC (Peripheral interface controller), MSP (Mixed signal processors) and ARM (Advanced RISC machines) series from

different suppliers. Bus and internal communication currently use standard bus systems such as CAN, I^2C.

The onboard memory for data storage is in the range of 32 kb to 8 MB. Some cubesats include flash memory to upgrade the storage up to 8 GB (Toorian, Diaz et al. 2008). Selva et al (Selva and Krejci 2012) give the example of the amount of memory that has been used to record a VGA camera image, which has 640 x 480 pixels for a single band. If there is no compression it is possible to store roughly 3,500 images on 8 GB. This is considerably reduced if a 32 band multispectral scanner is considered which can store approximately 32 multispectral cubes. In the future there is a need for the development of storage that can be used for the large amount of the data expected from hyperspectral images (Selva and Krejci 2012).

- **Communication system**

Most of the cubesat communication systems use the amateur radio band at UHF frequency. The downlink rate is typically 1,200 and 9600 bps. For the higher rates at VHF and S-band the data rates are 9,600 bps and 256 kbps respectively. Moreover, Syrlinks have demonstrated that using an x band transmitter it is possible to increase the downlink capability up to 50 Mbps (Syrlinks, 2015). The COT s-band transmitter from Clyde space is able to transmit data at rates of up to 2 Mbps (Clyde space, 2015). The uplink communication for commands data is usually communicated by s-band. The limitation of the communication system is in the link budget of the cubesat not in the technology of the radio frequency itself (Bouwmeester and Guo 2010).

2.2.5 Earth observation data processing

- **Image processing**

Remote sensing optical imagery is the most well known of the remote sensing techniques (Simmons 2008). The imager development is based on the requirements of the camera payload and the limitations of a cubesat mission. The typical image processing acquisition process is described in this section. Figure 2-17 shows a simple imaging system. The system starts with the light ‘a’ from the object passing through the aperture of the camera to the lens. The lens focuses the light onto the focal plane to form an image (Braegen, Hayward et al. 2007).

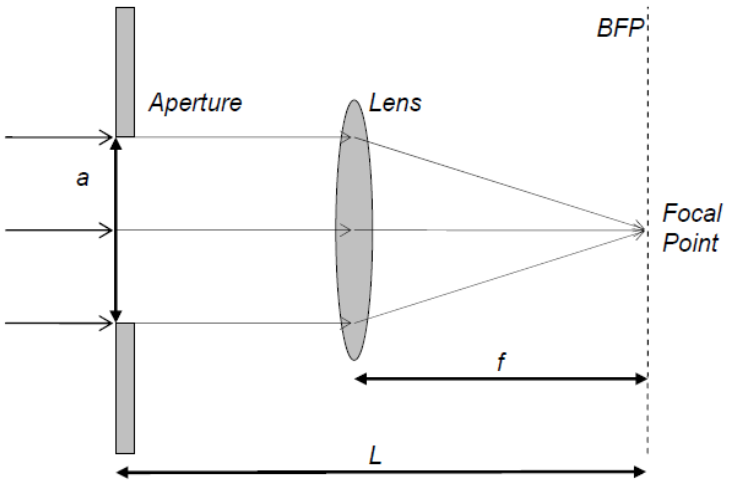


Figure 2-17 : Simple imaging system (Braegen, Hayward et al. 2007).

The camera is the passive optical sensor that observes the scene on the earth's surface and passes the energy through a lens (Floyd F. Sabins 1997). Figure 2-18 shows the framing system for acquiring remote sensing images. The object is the Earth's surface and the energy passes to the camera via the shutter, which is the device that allows the light to enter the camera. The light passes directly to the lens and forms the image at the focal plane on the satellite.

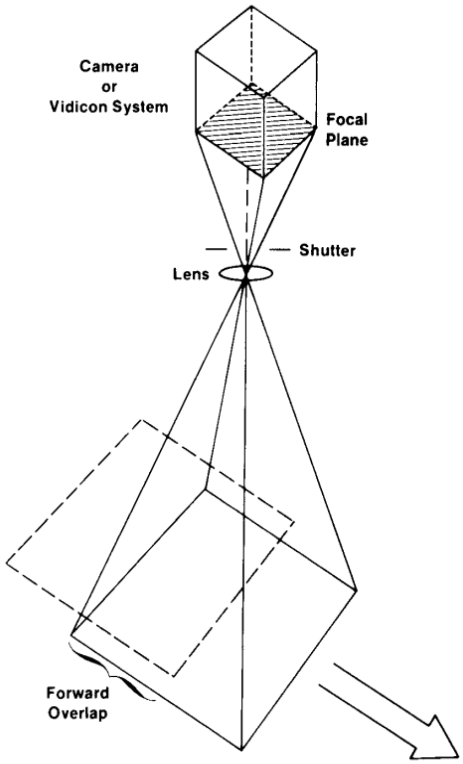


Figure 2-18 : Framing system for acquiring remote sensing images (Floyd F. Sabins 1997)

Due to the limitation of the cubesat specification, the simple way to fit the optical sensing into a cubesat is to use a flat area image chip. There are two types of sensors for visible spectral imaging, the CCD (Charged Couple Device) and the CMOS (Complementary Metal Oxide Semiconductor). Both are simple and have low power consumption operation. Typically they include an analogue to digital converter.

The satellite image processing on the ground starts when the satellite image is received by the ground station antenna. This image is processed before being stored in the data archiving system. Finally the image will be delivered to the users according to the customer requests. The overview of the image processing on ground is shown in Figure 2-19, which is the overview of the ground segment from JAXA (Masataka et al, 2013).

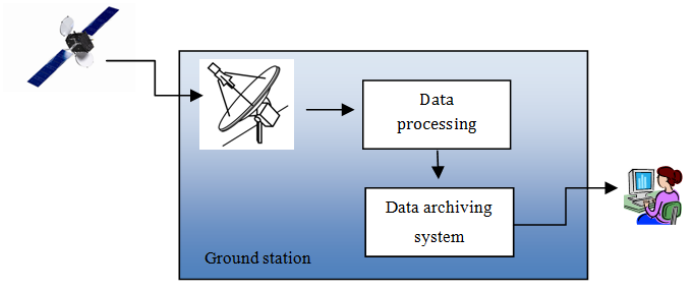


Figure 2-19 : Satellite image processing on ground.

Remote sensing imaging not only records the Earth's surface in different resolutions, but also extracts geo information for specific applications. If the user knows the level of the satellite image, it would be beneficial for the user in selecting the correct image behaviour for their application (Fancey et al, 1996). Data from the EO systems are generally referred to the term of 'product level'. This indicates the level of post processing that has been applied to the raw data by NASA standards for Earth remote (Di and Kobler, 2000).

The standard product levels are important because it gives basic information about selecting images that obtain the appropriate data as per application usage. The standard of production of each level is described by the characteristics below (Fancey et al, 1996) (ESA, 2015):

Level 0 The products are raw data at full instrument resolution. The image is as corrected at the sensor.

Level 1A This product level is the basic data set of raw image ingested, reconstructed, unprocessed instrument data at full resolution.

Level 1B The product from level 1A is processed to improve its geometric qualities by applying system corrections, which is predicted and measured by system distortions.

Level 2 The product is corrected based on satellite attitude data.

Level 3 The image data is rectified to given a cartographic projection using GCPs.

Level 4 The product is the mapping of level 2 images together to cover large areas of image and using a digital elevation model (DEM) or model output.

- **Image classification methods**

Classification can be considered as a form of pattern recognition that is achieved by identifying each pixel in the image in terms of the characteristics of the object (Mather 2004). Pattern recognition method is a well-known technique in many fields such as military applications. Texture is also widely applied in the interpretation of images. To classify the objects in a remote sensing image, there are two approaches to labelling the pixels, known as the supervised and unsupervised classification process.

The supervised process requires prior information from the user. It can be implemented using statistical or neural algorithms (Mather 2004). The input can be from the field collection data, image analysis or the report of the information etc. The statistical algorithm requests the input and parameters from the sample data that is available. On the other hand, in the neural process the algorithm is trained with the sample data directly.

The unsupervised process does not require any information from the user to process the algorithm. There are many unsupervised classification methods such Simple One-Pass Clustering, K Means, Fuzzy C Means, Minimum Distribution Angle, Isodata Classification, Self-Organization and Adaptive Resonance (Mather 2004).

- **Potential EO cubesat applications**

Many cubesats have been developed for the Earth observation missions. This section describes the cubesat technology that is currently used for Earth observation applications and identifies future applications that might be feasible to implement on cubesats (Selva and Krejci 2012).

° The measurement of the aerosols properties such as optical thickness, concentration, size and composition can be measured using a multispectral radiometer and

polarimeter (Richard J. Peralta, Carl Nardell et al. 2007). Aalto-1 and Cloud cubesat were cubesat missions intended to measure the aerosols properties. The Aalto-1 provides the optical thickness measurement of the aerosol and the Cloud cubesat mission was to measure the shape and size of aerosols by using polarimetric infrared passive measurements (Selva and Krejci 2012).

- Atmospheric temperature and humidity and pressure fields can be measured by using the multispectral or hyperspectral nadir infrared sounders, which can consist of hundreds of channels.
- A vegetation application measures status and biomass by measuring the Normalized differential vegetation index (NDVI), which uses the NIR and red channel. This can be measured using a cubesat. For example, CanX-2 carries the payload that has the characteristic similar to an instrument that could measure the vegetation index.
- Ocean colour instrument requires high spectral resolution in the UV, VIS and NIR channel. There has been no cubesat demonstration of ocean colour measurement yet but a miniaturized spectrometer designed for a cubesat would make this possible (Puschell and Stanton, 2012).
- The measurement of ocean surface temperature is the main information required to understand the ocean circulation. It is feasible to develop an instrument to measure the ocean surface temperature by using the passive measurement in the thermal infrared region.
- In the area of snow cover and sea ice cover it is possible to measure the information by using cubesat, which implements millimetre wave radiometers.
- Disaster monitoring is a large area of application. There are some cubesats that have been launched for this application including M-cubed, Ion, SwissCube and QuakeSat. These missions have demonstrated that cubesats are able to monitor disasters. For instance the QuakeSat monitored earthquakes using an ultra low frequency wave sensor.

2.3 RF communication subsystem

2.3.1 General overview

One of the most important parts of a cubesat is the communication subsystem. This subsystem presents a gateway for contact between the satellite and the ground station. Typically satellite communication systems consist of three main sections; these are the space, ground and

data transmission sections. The general requirements for the cubesat's communication subsystem consists of firstly, the uplink communication must to able to receive telecommand from the ground station and send it to the onboard computer for distribution of these commands to related subsystems. Secondly, the downlink communication must be able to receive all data and information from other subsystems, such as image data and housekeeping telemetry. Thirdly, the communication system has the responsibility for modulation and demodulation of the basic signal from the carrier frequency. Fourthly, the data packets must be in the protocol format for the RF link. Lastly, a beacon for emergency mode should also be included. As always, the main constraints for the design for the communication system on a cubesat are low power consumption and low mass.

The communication subsystem functions are illustrated in Figure 2-20 (Bruun, Hedegaard et al. 2001), the hardware configuration consists of the following components:

- Transceiver; it consists of both a transmitter (TX) and receiver (RX). The transmitter's function is to receive modulated data and amplify this data for transmission to the ground via an RF signal. The receiver's function is to receive the signal from the ground station before passing this data to the demodulation process.
- Modem; a modem is a short name for modulator and demodulator. It is defined as a device that modulates and demodulates signals. The modem functions are to translate the digital signal to an analogue signal and to change the analogue signal to a digital signal before sending the digital data to other subsystems.
- Antenna; an antenna is required to collect and radiate electromagnetic waves. The antennas function is to change the radio frequency electrical energy into radiated electromagnetic energy.
- Terminal Node Control (TNC); in the past this was known as micro control unit (MCU), which functions as a downlink and uplink; i.e. it reads the data to be transmitted, packs the data into protocol format and sends data to the modem or unpacks commands from the modem and sends this to the onboard computer, etc.

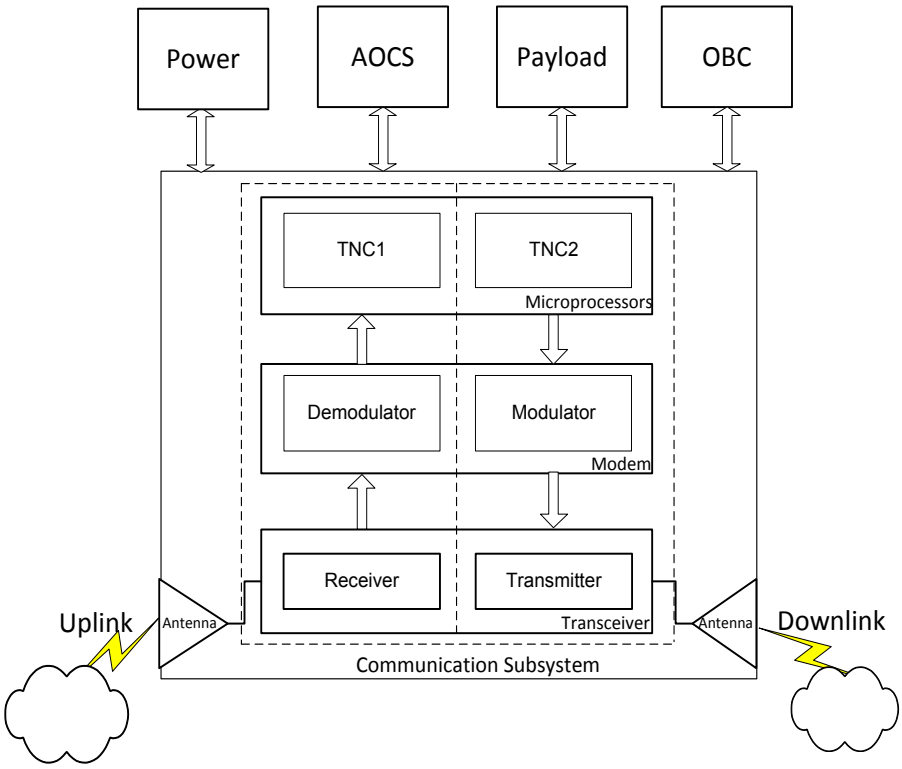


Figure 2-20: An overview of the cubesat communication components.

2.3.2 Related historical work

Information about communication subsystems for cubesats have been summarised in Table 2-1, which comes from "A Survey of Cubesat Communication Systems : 2009-2013" (Klofas and Leveque 2013). The table shows the communication system parameters from launched cubesat projects. This table also shows that most cubesats use the amateur radio frequency band for both transmission and reception systems. The amateur radio frequency organisation is non-profitable, and can provide the specific frequency for a cubesat faster than other organisations, such as the Federal communication commission (FCC), which has a time consuming licence requesting process (Huerta 2006). Moreover, using amateur radio frequencies increases the team's ability to track the satellite around the world because the amateur radio community has operators world-wide who can downlink data from the spacecraft (Pisane 2008; Muri and McNair 2012). The cubesat communication system uses standard protocols such as the AX.25 protocol, which is simple and reliable. The general band for uplink communication is VHF; where the frequency is between 145.8–146 MHz (2 m wavelength).

The UHF band is used for the downlink side, with a frequency between 435–438 MHz (0.7 m wavelength).

The approaches used for device selection in cubesat communication subsystems can be classified into three categories.

- Firstly, the use of COTS: many cubesats use a COTS approach to component selection. Table 2.2 shows COTS communication devices, which have been used in several projects. For instance, Cal-Poly CP2 uses the chip-con CC1000 transceiver, which is a COTS device developed by Texas instruments incorporated (Noe 2004). The Yaesu, which is a commercial dual-band transceiver, is used in the Montana Earth-Orbiting Pico-Explorer (MEROPE) cubesat (Hunyadi, Klumpar et al. 2002).
- Secondly, the use of modified COTS. These COTS components have required modification of the COTS component e.g. reduced mass, size, drilling mounting holes and increased transmission power, etc.
- Thirdly, the use of custom-built components. For instance, the RxTNC PIC16c716 receiver and TxTNC H8/3048 transmitter, which are custom made by Nishi RF laboratory. These custom-built devices have been used in the Prism project for downlink communication (Shimizu and Kusakawa 2010). There are, however, many custom-built transceivers that have not been successful due to difficulties in RF board design (Hunyadi, Klumpar et al. 2002)

Table 2-1 : Summary of cubesat communication subsystems (Credit for Bryan Klofas).

Satellite	Radio	Frequency	Satellite service	Power	TNC	Baud rate	Amount download	Antenna
AeroCube-3	Freewave FGRM	915 MHz	Experimental	2 W	Proprietary	77 kbaud GFSK	52 MB	patch
CP6	CC100/RF2117	437.365 MHz	Amateur	1 W	PIC18LF6720	1200 baud FSK	-	dipole
HawkSat-1	Microhard MHX-425	437.345 MHz	Amateur	1 W	Integrated	-	0 MB	monopole
PharrmaSat	Microhard MHX-2400 Stensat (beacon)	2.4 GHz 437.465 MHz	Experimental Amateur	1 W 500 mW	Integrated Integrated	10 kbps 1200 baud AFSK	650 MB N/A	patch monopole
BEESAT-1	-	436.000 MHz	Amateur	500 mW	CMX909B	4800/9600 baud GMSK	-	monopole
UWE-2	PR430	437.385 MHz	Amateur	1 W	Internal	1200 baud AFSK	-	dipole
ITUpSAT-1	Microhard MHX-425 BeeLine/CC1050	437.325 MHz 437.325 MHz	Amateur Amateur	1 W 350 mW	Integrated	19200 baud	0 MB N/A	dipole monopole
Swisscube	Butler oscillator /RF5110G RF2516	437.505 MHz 437.505 MHz	Amateur Amateur	1 W 100 mW	MSP430F1611 Integrated	1200 baud FSK 10 WPM	0 MB N/A	monopole monopole
Hayato	Custom	13.275 GHz	Earth Exploration	100 mW	Integrated	10 kbps/1 Mbps BPSK	0 MB	patch
Waseda-SAT2	TXE430-301A TXE430301A (beacon)	437.485 MHz 437.485 MHz	Amateur Amateur	150 mW 100 mW	H8/3052F H8/3052F	9600 baud FSK	0 MB N/A	monopole dipole

Satellite	Radio	Frequency	Satellite service	Power	TNC	Baud rate	Amount download	Antenna
Negai-Star	Data Beacon Radio	437.305 MHz 437.305 MHz	Amateur Amateur	150 mW 100 mw	-	1200 baud FSK 50 WPM	N/A	dipole dipole
TIsat-1	Alinco DJ-C6 CC1010 (beacon)	437.305 MHz 437.305 MHz	Amateur Amateur	500 mw 400 mw	MSP430F169 MSP430F169	1200 baud AFSK 15-110 WPM	N/A	monopole monopole
StudSat	CC1020 MAX1472 (beacon)	437.505 MHz 437.860 MHz	Amateur Amateur	500 mw 10 mw	UC3A0512 UC3A0512	4800 baud FSK 22 WPM	0 MB N/A	monopole monopole
RAX-1	Lithium-1	437.505 MHz	Amateur	750 mw	AX-25	9600 baud GMSK	4.8 MB	turnstile
O/OREOS	Microhard MHX-2400 Stensat (beacon)	2.4 GHz 437.305 MHz	Experimental Amateur	1w 500 mw	Proprietary AX.25	Variable 1200 baud AFSK	8 MB N/A	patch monopole
NanoSail-D2	Microhard MHX-2400 Stensat (beacon)	2.4 GHz 437.270 MHz	Experimental Amateur	1w 500 mw	Proprietary AX.25	Variable 1200 baud AFSK	N/A	patch monopole
Perseus(4)	-	450 MHz	Government	-	-	-	-	-
QbX(2)	TTC	UHF	Government	1w	-	9600 baud GMSK	-	quadrafila r helix
SMDC-ONE	Pericle	437.000 MHz	Government	-	-	-	-	Turnstile
Mayflower	Microhard MHX-425 Stensat (beacon)	437.600 MHz	Unlicensed Unlicensed	1w 1w	Proprietary AX.25	Variable 1200 baud AFSK	0 MB N/A	dipole
Jugnu	CC1070/RF5110 G MAX1472 (beacon)	437.505 MHz 437.505 MHz	Amateur Amateur	1w 10 mw	AX.25 CW	2400 baud FSK 20 WPM	N/A	monopole monopole
AubieSat-1	Melexis TH72011	437.475 MHz	Amateur	800 mw	CW	20 WPM	0 MB	dipole
DICE(2)	L3 Cadet	465 MHz	Meteorological	1w	Proprietary	2.6 Mbps BPSK	8.4 GB	dipole
HRBE	CC1000	437.505 MHz	Amateur	850 mw	AX.25	1200 baud FSK	7.6 MB	monopole
M-Cubed	Lithium-1	437.485 MHz	Amateur	1w	AX.25	1200 baud FSK	0 MB	monopole
RAX-2	Lithium-1	437.345 MHz	Amateur	1w	AX.25	9600 baud GMSK	242 MB	turnstile
Xatcobeo	GomSpace U482C	437.365 MHz	Amateur	500 mw	AX.25/CW	1200 band MSK/20 WPM	-	turnstile
ROBUSTA	MC12181 /MAX2608	437.325 MHz	Amateur	800 mw	AX.25	1200 baud AFSK	0 MB	dipole
e-st@r	BHX2-437-5	437.445 MHz	Amateur	500 mw	AX.25	1200 baud AFSK	0 MB	dipole
Goliat	Alinco DJ-C7 Microhard MHX-2420	437.485 MHz 2.4 GHz	Amateur	500 mw 1w	AX.25/CW Proprietary	1200 baud AFSK /20 WPM Variable	0 MB	monopole patch
PW-Sat	ISIS TRXUV	145.900 MHz	Amateur	200 mw	AX.25/CW	1200 baud BPSK /12 WPM	-	dipole
Masat-1	Si4432	437.345 MHz	Amateur	100/400 mw	Custom/CW	GFSK/120 CPM	305 MB	monopole
UniCubeSat-GG	AstroDev Custom	437.305 MHz	Amateur	500 mw	AX.25/CW	9600 baud GMSK	0 MB	dipole
SMDC-ONE(2)	Pericle	UHF	Government	-	-	-	-	turnstile
AeroCube-4(3)	FreeWave MM2 CC1101	915 MHz 915 MHz	Experimental Experimental	2w 1.3w	Proprietary Proprietary	38.4 kbaud 500 kbps FSK	-	Patch Patch
Aeneas	MHX-425 Stensat (beacon)	437.000 MHz 437.600 MHz	Experimental Amateur	1w 1w	Proprietary AX.25	Variable 1200 baud FSK	N/A	monopole monopole
CSSWE	Lithium-1	437.345 MHz	Experimental	1w	AX.25	9600 baud GFSK	60 MB	monopole
CP5	CC1000/RF21 17	437.405 MHz	Amateur	500 mw	AX.25	1200 baud FSK	500 MB	dipole
CXBN	Lithium-1	437.525 MHz	Amateur	1w	AX.25	9600 baud GFSK	-	turnstile
CINEMA	Emhiser	2200 MHz	Space research	1w	Proprietary	1 Mbps FSK	-	patch
RE	Helium-100	915 MHz	Government	1w	AX.25	57.6 kbps FSK	-	dipole

Table 2-2 : Communications Subsystem: TNC and Transceiver.

Project	FM band	TNC	Transceiver	Output power
CANX-1	900 MHz	-	CMX469	0.5 W

Cal-Poly, CP1	440 MHz	-	Alinco DJ-5C	0.3 W
Cal-Poly, CP2	440 MHz	PIC18	CC1000	1 W
DTUSat	440 MHz	-	CMX469	1 W
Hawaii	440 MHz	PIC16+MX614	VX-1R	0.5 W
Montana State	144/440 MHz	Pico-Packet	VX-1R	1 W
Stanford	2.4 GHz	MHX 2400	MHX 2400	1 W
Tokyo	144/440 MHz	PIC16	Nishi RF Lab	0.8 W

The antenna is an important device for cubesat communication. The monopole and dipole are often used in cubesat missions as they are very small and simple to assemble. Although these types of antennas provide low gain they are also low cost (Hunyadi, Klumpar et al. 2002; Braegen, Hayward et al. 2007).

2.3.3 Radio frequency allocation

Cubesats have been launched successfully into many different orbits. The typical cubesat over the past eight years has utilised transceivers and beacons to downlink in the UHF band and has utilised the AX.25 protocol (Singh 2009). Table 2-3 shows the details of uplink and downlink frequencies, which is the official frequency scheme by AMSAT (radio amateur satellite corporation) (Traussnig 2007). These bands are internationally allocated and are provided by the AMSAT. The uplink frequencies are generally applied for telecommand and handshaking and it is sufficient to use low frequencies. On the downlink side higher frequencies are used to transmit the data. The license free bands are the fastest way to get a frequency allocation for the cubesat because no time is spent on the formal requirement process. Nonetheless the disadvantage of these bands is that there is no protection and there is noise and interference, especially in urban areas. The S-band covers the range between 2 to 4 GHz and frequencies in this band have less noise and interference. The relatively short wavelength enables small antennas to be used for this band. Most of the S-band frequencies request an official license except for the amateur frequency band from 2.4 to 2.45 GHz.

There are some cubesats operating in S-band frequencies, for instance MAST (multi-application survivable tether) (2.4 GHz), CanX-2 (2.2 GHz) and KySat-1 (2.4 GHz) (Singh 2009). With the new generation of cubesat technology, higher frequencies will enable smaller components and increased antenna gain through these new technologies, an update to the AX.25 protocol is needed to further cubesat mission capability (Klofas, Anderson et al. 2009).

Table 2-3 : The alternative uplink-downlink frequencies by AMSAT.

Item	Alternative 1	Alternative 2	Alternative 3
Uplink frequency	144-146 MHz	435-438 MHz	435-438 MHz
Wavelength	2 m	70 cm	70 cm
Downlink frequency	435-438 MHz	1260-1270 MHz	2400-2450 MHz
Wavelength	70 cm	24 cm	12.5 cm

2.4 Onboard computer architecture

2.4.1 An overview of the onboard computer for cubesats

The main objective of the onboard computer is managing all functions, which are performed on the spacecraft including the following list (Wertz and Larson 1999; Scholz 2005; Rani, Santhosh et al. 2010):

- Telecommand receiving, execution and distribution.
- Parameters monitoring, satellite health monitoring and report.
- Memory read and writes.
- Telemetry storage and management.
- Data flow management.
- Function activation and deactivation.
- Power application management
- Time synchronisation, etc.

An overview of the cubesat onboard computer architecture is shown in Figure 2-21 (Clausen, Hedegaard et al. 2001). The main device in the onboard computer is the micro-controller unit (MCU). It operates continuously in orbit. Consequently selection of a low power MCU is required for optimisation of the power consumption on a cubesat. In addition the onboard computer contains the programmable read-only memory (RROM), which contains the basic program. In addition it has a flash memory, which holds the operating system and software updates. Finally, the random-access memory (RAM) is used as a temporary memory for data, such as measurement values or images. In order to transmit power, data or other information between each subsystem inside the cubesat spacecraft, there must be an internal interface to enable communication between each subsystem.

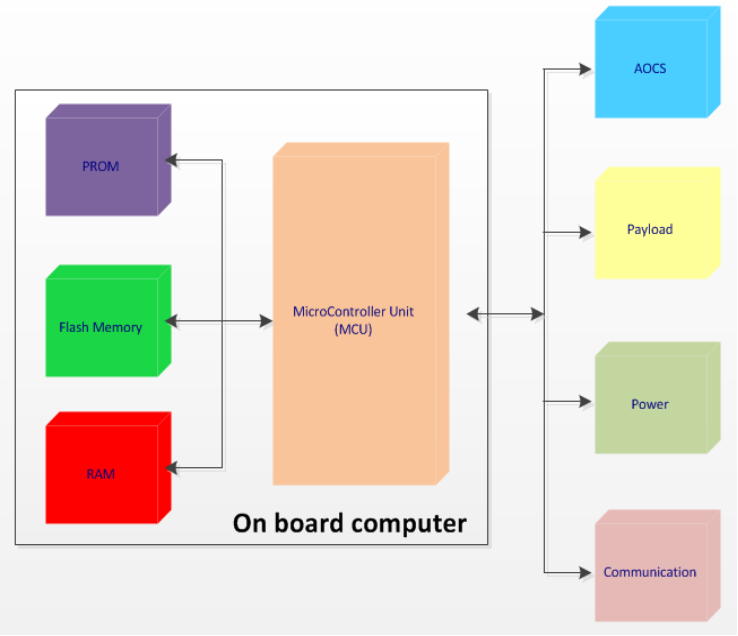


Figure 2-21 : Overview of the architecture of the onboard computer hardware in cubesat.

2.4.2 Processors on cubesats

There are many cubesats that have been launched with COTS processors. Figure 2-22 shows the flight module FM431, which has been used in several cubesat projects, such as the AIS Pathfinder 2, ITU-pSAT1, HawkSat-1, Delfi-C3, MAST and Libertad-1 (Teney 2009; Rani, Santhosh et al. 2010; Pumpkin 2011). There are many companies that have provided cubesat onboard computers, such as GomSpace, Pumpkin Inc & Clyde-space, Innovative solutions in space (ISIS) and space and systems development laboratory (SSDL).

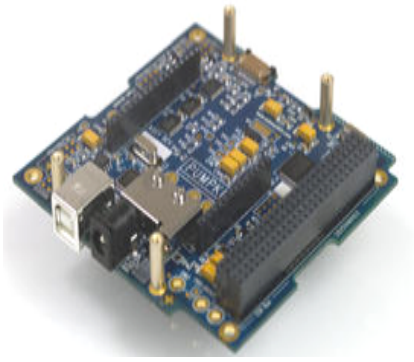


Figure 2-22 : The onboard computer: Pumpkin cubesat kit FM 4 (Pumpkin 2011).

Selecting the appropriate microprocessor depends on the mission requirements of each cubesat. The hardware used for an onboard computer cubesat missions can be in various configurations even if it has the same onboard computer specification. Table 2-4 presents the characteristics of the processor and microcontrollers which have been used on cubesat projects (Teney 2009).

Table 2-4 : The characteristics of microprocessors used on cubesats.

Cubesats	**Processor**	**Clock**	**Memory**			**Power supply**
AAUSat-I (Aalborg University CubeSat-1)	Siemens C161	10 MHz	RAM ROM Flash	512 512 256	kB kB kB	5 V 150 mW
AAUSat-II (Aalborg University CubeSat-2)	Atmel ARM7	8/40 MHz	RAM Flash	2 8	MB MB	3.3 V 80 mW at 8MHz 300 mW at 40MHz
CanX-1 (Canadian Advanced Nanosatellite eXperiment)	Atmel ARM7	40 MHz	RAM ROM Flash	512 128 32	kB kB MB	3.3 V 400 mW
CP1 and CP2 (CalPoly-1 and CalPoly-2)	PIC 18LF6720	4 MHz	RAM ROM Flash	4 1 128	kB kB kB	2V< 10 mW
CubeSatXI-IV	PIC 16F877	4 MHz	RAM ROM	368 32	Bytes kB	5 V <10 mW
CUTE-1 (Cubical Tokyo Tech Engineering Satellite-1)	Hitachi H8/300	Unknown	RAM ROM Flash	521 256 4	kB kB MB	Unknown
DTUSat-I (Technical University of Denmark, (DTU))	Atmel AT91M40800	16 MHz	RAM ROM Flash	1 16 2	MB kB MB	Unknown
KUTEsat (Kansas Universities' Technology Evaluation Satellite)	PIC 18F4220	8 MHz	RAM ROM EEP ROM	512 32 256	MB kB bytes	5V < 10 mW
MEROPE (Montana St. University)	Freescale MC68HC812A4	8 MHz	RAM EEPROM	1 4	kB kB	5V 150 mW
nCube (Norwegian CubeSat)	Atmel AVR ATmega32L	4 MHz	RAM ROMF lash	2 32 1024	kB kB bytes	5 V 12 mW
Delphi C-3, Libertad-1	Taxes Instruments MSP 430	8 MHz	RAMF lash	10 55	kB kB	3.3 V or 5 V < 10 mW
Sacred and Ricon	PIC 16C77	4 MHz	RAM	64	kB	5 V< 10 mW

2.5 Region of interest (ROI)

2.5.1 The Region of interest (ROI) automatic detection

To provide the capability to transmit many Earth observation images, but still operate within the limitation of resources of a cubesat, is a challenging issue. One of the possible approaches is to reduce the size of the original image by image segmentation techniques and image compression methods (Tahoces, Varela et al. 2008).

A related technique is the identification of the region of interest. In any image only part of the image is of interest and needs to be processed. The rest can be discarded. The region of interest technique has been applied in many areas such as; telemedicine, remote sensing imagery, web browsing, image database and video compression, etc. (Liu, Xia et al. 2004; Fei, Jishang et al. 2007; Rajkumar and Latte 2011; Sutha 2013). In any remote sensing image, there are particular areas which are of interest; e.g. forests, farm land, rivers, land marks, lakes, etc. Consequently, the ROI will depend on the properties of that satellite image and also the user area applications requested.

There have been many studies that have developed automatic ROI detection by using image segmentation techniques (Guarnieri and Vettore 2002; Fukuma, Tanaka et al. 2005; Chen and Chih-Chang 2007; Guang, Ming et al. 2009). Identification of the ROI of an image can be performed both manually and automatically. The image segmentation techniques identifies and separates the original image into a group of areas, each having their own distinct characteristics, such as grey level, texture or colour scale. It then uses the classification to identify the region of interest.

2.5.2 Image segmentation

Image segmentation is well-known in the field of image processing and has been developed for extracting different features or textures inside an image. This can be done a number of different ways using the properties of the image. In practice a real scene observed from a satellite will contain a variety of features, textures or shadows etc., and so the images can be very complex. Therefore, extraction of a feature of an image is very difficult using only one image segmentation technique. The solution is to find the appropriate image segmentation techniques and combine different methods to detect the region of interest more effectively.

There are many image segmentation methods, which are often based on the basic properties of intensity values, discontinuity and similarity (Gonzalez and Woods 2007; Al-amri, Kalyankar et al. 2010):

- Discontinuity: the methodology is to divide an image based on discontinuities in intensity. The characteristic objects to detect are corners, crossing points, T-junctions or objects represented as dots.
- Similarity: this approach partitions an image into groups of pixels, which have similar characteristics.

The main methods that result from these properties are the following:

- Edge segmentation based method: The image segmentation can be obtained by the detection of edges, defined by a change of the grey level intensity in a narrow area. Improvement of the image segmentation process performance can be achieved by a combination of techniques.
- Histogram segmentation based method: The image histogram shows the number of pixels of a particular value that are included in each group. The image can be classified using histogram thresholding. Normally this segmentation process is based on the image grey level histogram. The threshold for each group needs to be selected, such as the threshold for ocean, urban or forest regions, etc.
- Texture segmentation based method: The texture segmentation has been studied widely in combination with other features, for instant shape, spectral and models. The textural features are based on statistics, which summarise the relative frequency distribution of the pixel value in the image. This describes how often one grey tone will appear in a specified spatial relationship to another grey tone on the image (Shanmugan, Narayanan et al. 1981). The common statistics applied in the use of texture analysis are energy, entropy, contrast, variance, homogeneity and correlation, etc. (Shanmugan, Narayanan et al. 1981). The co-occurrence matrix has been used to determine texture similarity for grouping pixels in a region. Texture segmentation techniques generally consist of two stages. The first stage finds the local structure of an image and converts each pixel into a vector of these local properties. Secondly the clustering method groups pixels which have similar properties together (Ershad 2011).

2.6 Image compression

2.6.1 Overview of image compression

The limitations of transmission bandwidth, power and storage capacity on cubesats are the key drivers for the development of satellite onboard image compression technologies. Earth observation provides an enormous amount of data (Han and Fang 2008) and to avoid the requirement for continuous ground transmission the data has to be stored onboard the satellite. As more innovative payloads are used on cubesats, the data storage requirements onboard will increase (Yu 2009). The end result is that improved image compression on spacecraft is extremely necessary to fulfil future satellite missions. Image compression, in addition to reducing the volume of the original image, will also decrease the hardware requirements and costs of the communication system (Lambert-Nebout and Moury 1999; Han and Fang 2008).

A typical image contains both information and redundancy, where the information part is the part of interest. It is therefore essential to preserve this part in the original data. The redundant data are represented as irrelevant or repeated information. Data compression is a technique to decrease the redundancy in the image and so reduce the size but retain the information. By processing the image to a smaller size, the possibility of storage in a limited memory capacity or transmission in a limited band width transmission channel is increased.

There are two categories of image compression methods; lossy and lossless (Yu 2009; Kiadtikornthaweeyot 2011). The details of each method are as follows:

- Lossless compression is a data encoding technique, which in general is used to compress multimedia data such as audio, video and still images. The reconstructed image after compression is exactly the same as the original image. This type of image compression method is suitable for no-loss image applications.
- Lossy compression technique removes some of the data with the purpose of reducing the size of output. With the subsequent reconstruction of the compressed data, there is some degradation from the original image (Yu 2009). The reconstructed image would have some information loss with a varying degree.

The general data flow of an image compression system is illustrated in Figure 2-23 which consists of encoder and decoder modules (Al-amri, Kalyankar et al. 2010). The encoder executes compression and the decoder performs the complementary process of decompression. The original and decompressed image is represented by the parameter $I(x, y)$ and $I'(x, y)$ respectively. The (x, y) represents the pixel position in the image. The details of each step function are described as follows:

- The input image $I(x, y)$ is transformed into the first image, mapper. This process is designed to reduce spatial and temporal redundancy (Gonzalez and Woods 2007). The image is transformed to a set of less correlated transform coefficients. The next step is the symbol coding. There are different types of coding, such as Huffman coding, Run-length coding or Bit-plan encoding. The output from the encoder is compressed data, which is ready for storage or transmission.
- The decoder module performs decompression. This module consists of a symbol decoding step and an inverse mapper. After receiving the compressed data from a storage or transmission channel, the data is relayed to the symbol decoding process and inverse mapper to reconstruct the output image, $I'(x, y)$.

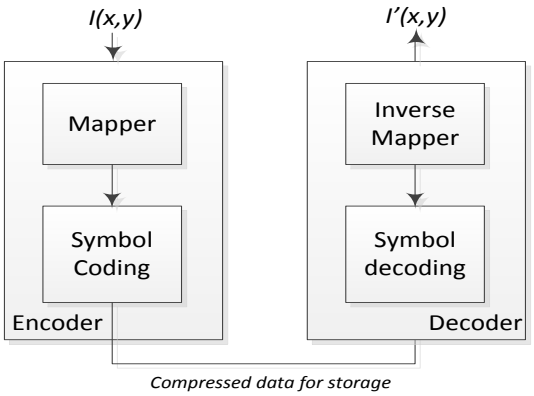


Figure 2-23 : The image compression system (Gonzalez and Woods 2007).

The performance comparison of each image compression method can be made using the mean square error (MSE) and the peak signal to noise ratio (PSNR) (Gonzalez and Woods 2007). The MSE is given as follows by equation 2.1 and the PSNR is given as follows by equation 2.2, where the M and N represent row and column of the image respectively. The PSNR is most ordinarily used as a measure of image compression reconstruction quality. Normally, a high PSNR and low MSE of the reconstruction image is required for high image exactness and low error respectively. In the equation M and N are the number of rows and columns in the image.

$$MSE = \frac{1}{MN} \sum_{y=0}^{M-1} . \sum_{x=0}^{N-1} [I(x, y) - I'(x, y)]^2 \quad (2.1)$$

$$PSNR = 20\, log_{10}\left(225/\sqrt{MSE}\right) \quad (2.2)$$

In addition, the measurements of the compression ratio and compression time are used to identify and compare the performance of the image compression methods. The compression ratio is the ratio between the original image size and compressed image size. The compression time is the period of time, which is required for both compression and decompression processing. It is dependent on various factors, for instance, the complexity of the process, the capability of the implementation and the performance of the processor.

2.6.2 Image compression in space

Image compression techniques are very important to reduce the size of the image before it is sent back to the ground station. There are many image compression schemes being used in space, such as Joint photographic experts group (Jpeg), Jpeg2000 and Lossless Jpeg (Jpeg-LS). These are designed for lossless or lossy high quality image compression (Chien-Wen, Tsung-Ching et al. 2009). Table 2-5 illustrates the image compression systems onboard space missions, where information is based on 40 satellites (Yu 2009). The table has been revised focusing on the algorithms used and the theoretical basis for the algorithms. A summary of the table is detailed as follows:

- More than half of the satellites reviewed are Earth observation satellites (23 out of 40).
- The compression systems in the table can be divided into four categories. These are the transform-based discrete cosine transform (DCT), transform-based discrete wavelet transform (DWT), prediction-based and block truncation coding (BTC)-based compression systems.
- The launch time slot of the satellites has been divided in five periods.
- The trend of using the DWT for image compression on satellites has increased in recent years.
- There are many different types of image compression which have been implemented on different satellites. This review introduces adaptive moment preserving BTC (AMPBTC), differential pulse code modulation (DPCM), adaptive DPCM (ADPCM), CCSDS-Lossless data compression (CCSDS : Consultative committee for space data system), (CCSDS-LDC), Lossless Jpeg (Jpeg-LS), low-complexity lossless compression (LOCO). There are more details in “Image compression systems on board satellites” (Yu 2009).

Table 2-5 : Onboard image compression system on satellites (Yu 2009).

Satellite name	Owner	Year	Algorithm	Main type of compression	Application
1. PoSAT-1	SSTL Portugal	1993	AMPBTC	BTC	EO
2. Tsinghua-1	SSTL TsingHua U.	2000	AMPBTC	BTC	EO
3. TiungSAT	SSTL Malaysia	2000	Improved AMPBTC	BTC T805	EO
4. SPOT-1	CNES	1986	Fixed-rate DPCM	DPCM	EO
5. SPOT-2	CNES	1990	Fixed-rate DPCM	DPCM	EO
6. SPOT-3	CNES	1993	Fixed-rate DPCM	DPCM	EO
7. SPOT-4	CNES	1998	Fixed-rate DPCM	DPCM	EO
8. IKONOS	USGeoEye	1999	ADPCM Kodak	DPCM	EO
9. QuickBird	US DigitalGlobe	2001	ADPCM Kodak	DPCM	EO
10. WorldView-1	US DigitalGlobe	2007	ADPCM Kodak	DPCM	EO
11. MTI	US DOE	2000	CCSDS-LDC	RICE	EO
12. EO-1	NASA	2000	CCSDS-LDC	RICE	EO
13. FedSAT	Australia	2002	Adaptive JPEG-LS	JPEG-LS	EO
14. SPOT-4	CNES	2002	DCT (with rate controlled)	DCT	EO
15. Proba-2	ESA	2009	JPEG-baseline and LZW	DCT	EO
16. SUNSAT	SouthAfrica	1999	JPEG-baseline	DCT	EO
17. PROBA-1	ESA	2001	JPEG-baseline	DCT	EO
18. Beijing-1	SSTL-China	2005	JPEG-baseline	DCT	EO
19. ALOS	JAXA	2006	JPEG-baseline and Lossless JPEG	DCT Lossless JPEG	EO
20. Cartosat-1	ISRO	2005	JPEG-baseline (CR = 3.2)	DCT	EO
21. Cartosat-2	ISRO	2007	JPEG-baseline (CR = 3.2)	DCT	EO
22. RapidEye	SSTL-Germany	2008	Lossless and Lossy JPEG	DCT	EO
23. THEOS	CNES-Thailand	2008	DCT(CR = 2.8 or 3.7)	DCT	EO
24. BilSAT-1	SSTL-Turkey	2003	JPEG2000	DWT	EO
25. RASAT	Turkey	2009	JPEG2000	DWT	EO
26. IMS-1	ISRO	2008	JPEG2000	DWT	EO
27. X-SAT	Singapore	2009	Content-driven version JPEG2000	DWT	EO

Satellite name	Owner	Year	Algorithm	Main type of compression	Application
28. PLEIADES-HR	CNES	2010	DWT+BitPlaneEncoder	DWT	EO
29. Meteisat-8	ESA	2002	JPEG-baseline and Lossless JPEG	DCT Lossless JPEG	Weather
30. STEREO	NASA	2006	RICE and a lossy wavelet (H-compress)	RICE and DWT	Sun Expl
31. PICARD	CNES	2009	CCSDS-LDC	RICE	Sun Expl
32. TRACE	NASA	1998	JPEG-baseline	DCT	Sun Expl
33. Solar-B	JAXA	2006	12bit JPEG-baseline and 12bit DPCM	DCT and DPCM	Sun Expl
34. Mars Odyssey	NASA	2001	CCSDS-LDC fast lossless predictive compressor or slower lossy DCT compressor	RICE	Mars Expl
35. Phobos	Soviet Union	1988	DCT+scalar quantizer+ fixed length coding	DCT	Mars Expl
36. Mars Exploration Rovers	NASA	2003	ICER and LOCO	DWT and JPEG-LS	Mars Expl
37. Chang'E-1	China	2007	Differential Predictive+ Bit Plane	DPCM	Moon Expl
38. Clementine	NASA	1994	DCT+Quantizer+Zig Zag+ RLE+Huffman(very close to JPEG)	DCT	MoonExpl
39. MicroLabSat	JAXA	2002	JPEG-baseline and Lossless JPEG	DCT Lossless JPEG	Science Demo
40. TEAMSAT	ESA	1997	JPEG-baseline	DCT	Science Demo

There is little information available about image compression on cubesats although there are many cubesats which have COTS digital cameras. Normally the COTS digital camera has the following characteristics:

- Image compression on-board, which has been integrated off-the-shelf with Jpeg format.
- Internal memory storage which is useable from the onset.
- Software implemented and ready to use.

Nevertheless, some COTS digital cameras provide the output image in a raw data format, e.g. the OV7678FB camera module, on the compass-1 cubesat (Gulzar 2009).

2.7 Summary

This review shows that the development of cubesats is aimed at improving their capability so they can be used in more complex missions. The limitations of power transmission and storage onboard are significant issues in the design of cubesats. The power transmission restricts the transmission of data to the ground station. The communication system between ground and cubesat is the important issue to identify the parameters for the system design. Consequently this research studies the cubesat communication system to analyse the limitations of power and the amount of the data that can be transmitted.

The literature review has identified the developments that have taken place in three key technological areas on cubesats; i.e. the RF communication system, the onboard computer and onboard image processing. There has been a proliferation of cubesats in recent years and a growing number are concerned with providing high resolution imagery. This has led to greater demands on communications and onboard processing, etc.

Chapter 3

RF Communication

One of the challenges in cubesat development is the design of high performance and reliable operation of the communication subsystem. The communication link is affected by numerous constraints from the satellite, ground station and environmental conditions. Moreover, a cubesat has limited power available to transmit the data. This can be understood by examining the link budget.

This chapter presents an analysis of the communication subsystem design. An example of cubesat link budget calculations are introduced at the end of this chapter.

3.1 Link budget calculation

The link budget analysis is a significant part of satellite communication design as it is used to determine the key parameters in the communication subsystem. It is very necessary that satellite comunication links are carefully designed within the available margin resources (Kiadtikornthaweeyot 2010).

Normally, the communication systems can be divided into three main parts; transmission, communication and reception. Each part consists of the common parameters such as antenna gain, transmitter power, bandwidth, uplink/downlink frequencies, etc. The various losses in the system contributes to the effectiveness of the system transmission (David Calcutt 1994). The relationship between each of the parameters is given in equation 3.1 (Polaschegg 2005; Addaim, Kherras et al. 2007; Stengel 2012).

$$\frac{\mathrm{E_b}}{\mathrm{N_o}} = \frac{\mathrm{P_{tx}L_{tx}G_{tx}L_sL_aG_{rx}}}{\mathrm{kT_sR}} \qquad 3.1$$

where;

$\frac{E_b}{N_o}$	:	Bit energy (E_b) to noise-density(N_o)	[W]
P_{tx}	:	Transmitter power	[W]
L_{tx}	:	Transmitter to antenna loss	

G_{tx}	:	Peak transmitter antenna gain	
L_s	:	Free space loss	
L_a	:	Additional path loss	
G_{rx}	:	Peak receiver antenna gain	
k	:	Boltzmann's constant = 1.38×10^{-23}	[J/K]
T_s	:	System noise temperature	[K]
R	:	Data rate of the system	[bps]

The bit energy per noise density is the most common parameter that is used to compare the communication systems even when they differ in bit rate modulation. The quantity E_b is the bit energy and N_o is the noise density. N_o is the total noise power in the frequency band of the signal divided by the bandwidth of the signal. In a digital communication system, the E_b/N_o is the measure of the signal to noise ratio. It relates to the bit error rate (BER) and depends on the modulation techniques (Zyren and Petrick 2000). The BER required is determined by the application. Most commonly a BER of around 10^{-5} is chosen for voice and around 10^{-9} for high speed data (David Calcutt 1994; Langton 2002). If for example the communication system requires a BER approximately 10^{-6}, an E_b/N_o of 10.3 dB is required if PSK modulation is used (Zyren and Petrick 2000). Therefore this analysis used E_b/N_o values of 10, 10.3 and 13 dB.

Antenna gain:

The antenna gain is defined as the ratio between the power propagated in a given direction relative to the power that would be produced by an isotropic antenna, a reference antenna which radiates equally in all directions. The receiving station antenna gain can be determined from equation 3.2 given below.

$$G = \eta\left(\frac{\pi D}{\lambda}\right)^2 \qquad 3.2$$

where;

G	:	Antenna gain	
η	:	Antenna efficiency	
D	:	Antenna diameter	[m]
λ	:	Wave length	[m]

Effective Isotropic Radiated Power (*EIRP*):

EIRP is the amount of power that would have to be emitted from a theoretically isotropic antenna, to produce the peak power density observed in the direction of maximum antenna gain. *EIRP* is dependent on the selected transmitter specification. The relationship of parameters to calculate EIRP is shown in equation 3.3 (Braegen, Hayward et al. 2007; Elbert 2008; Warren 2013):

$$\mathrm{EIRP} = \mathrm{P_t L_{tx} G_{tx}} \qquad 3.3$$

where;

P_t	:	Transmitter power	[w]
L_{tx}	:	Transmitter to antenna loss	
G_{tx}	:	Peak transmitter antenna gain	

Path losses:

A path loss is the summation of all the losses between the transmitter and the receiver. It consists of free space loss and an additional path loss.

Free space loss:

The free space loss can be calculated from equation 3.4 (Fortescue, Stark et al. 2003). As seen in the equation, the free space loss depends on the signal frequency and the distance between the spacecraft and the receiving station. The loss increases with an increase in distance and frequency.

$$L_s = \left(\frac{4\,\pi\,d}{\lambda}\right)^2 \qquad 3.4$$

Where;

L_s	:	Free space loss	
λ	:	Wavelength	[m]
d	:	Distance	[m]

Slant range:

The distance between the receiving station and a satellite is called the slant range distance and is shown in Figure 3-1 (Stengel 2012). If the satellite is located at an elevation of 90 degrees, which is vertical to the ground surface, this is the shortest distance between the ground

station and the satellite. It is defined by '*h*' or satellite altitude. The slant range distance is given by equation 3.5.

$$d = -R_e \sin\varepsilon + \sqrt{(R_e + h)^2 - R_e{}^2(\cos\varepsilon)^2} \qquad 3.5$$

where;

d	:	Slant range	[km]
R_e	:	Earth radius (6,375 km)	[km]
h	:	Satellite altitude	[km]
ε	:	Elevation angle	

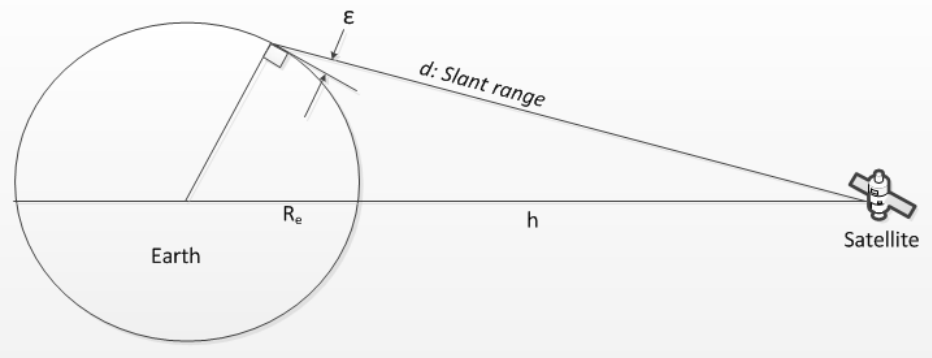


Figure 3-1 : Slant range calculation (Stengel 2012).

Table 3.1 shows the slant range and free space loss at various heights of the satellite ranging from 350 to 800 km, at an elevation angle of 10 degrees and a frequency of 437 MHz. As the satellite altitude increases the slant range distance becomes longer and there is higher free space loss.

Table 3-1 : The calculation of slant range and free space loss.

Satellite altitude : h (km)	Slant range : d (km)	Free space loss : L_s(dB)
350	1303.39	-147.55
400	1439.58	-148.42
450	1569.78	-149.17
500	1694.81	-149.83
550	1815.37	-150.43

600	1931.96	-150.97
650	2045.04	-151.47
700	2154.97	-151.92
750	2262.06	-152.34
800	2366.57	-152.73

System noise temperature:

In any communication system, there are many possible sources of noise, such as noise from the environment, noise generated within the receiver and noise from the antenna and the feed. The sum of the noise from the effect of the temperatures of the antenna and the receiver are the system noise temperature. The system noise temperature in a cubesat can be up to 5,000 K and the ground station system noise temperature can be up to 2,000 K for a turnstile antenna (Addaim, Kherras et al. 2007).

3.2 Cubesat data transmission in one orbit

In this section, the link budget of a cubesat is discussed. In a low Earth orbit, the maximum contact time depends on the latitude of the ground station and the satellite altitude but it is approximately 10 minutes per pass. On a cubesat, there is a limitation on the transmission power to a few watts. Assuming that a single side of a cubesat (0.01 m^2) is fully illuminated by the sun this would provide a DC power of 3-4 watts if it was assumed that efficient multi-junction solar arrays (30%) are used. In practice the attitude control of the cubesat is unlikely to maintain a single sun-pointing face and more than one face will have to be covered in solar arrays. These considerations typically limit the RF power available to 1-2 Watts so that in sunlight the battery is not discharged. Figure 3-2 shows typical link budget parameters in the communication system that includes satellite, link and ground station parts. The frequency is 437 MHz, elevation angle is 10˚ and the additional loss is 3.5 dB (Addaim, Kherras et al. 2007). The gain of the receiving antenna is 16 dBi, if a yagi antenna is assumed for use. The system noise temperature is assumed to be 2,000 K (Addaim, Kherras et al. 2007). The summary of the parameters is given in Table 3.2.

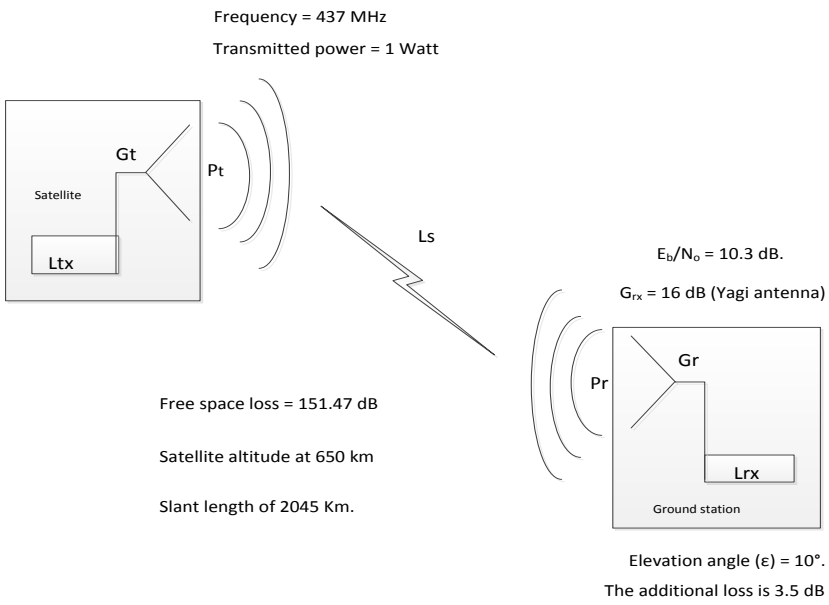


Figure 3-2 : Link budget parameters.

Table 3-2 : The summary of link budget parameters.

Satellite		
Antenna gain	0	dBi
Frequency	437	MHz
Transmitted power	1	W
Channel		
Free space loss[1]	151.47	dB
Additional loss	3.5	dB
Ground		
Antenna gain	16.00	dBi
T_{system}	2,000	K
E_b/N_o required	10.3	dB

The graph in Figure 3-3 shows the amount of the data that can be transmitted from a cubesat down to the ground station in 10 minutes for typically one visibility period. The results

[1] The free space loss parameter depends on the slant range of the satellite and ground station. This free space loss is calculated for 437 MHz frequency, a satellite altitude of 650 km and an elevation of 10 degrees.

of this graph is based on the link budget calculation from the parameters in Table 3-2 with different numbers of E_b/N_o (10, 10.3 and 13 dB). Different satellite altitudes from 350 to 1,250 km were used. For instance, at a satellite altitude of 650 km with a power transmission of 1 Watt and aE_b/N_o of 10.3 dB, the cubesat will then be able to transfer 545 Mbit of data to the receiving station. If the required E_b/N_o is 13 dB, around 293 Mbit of data can be transmitted to the ground. (Further details of the link budget calculation are given in Appendix A).

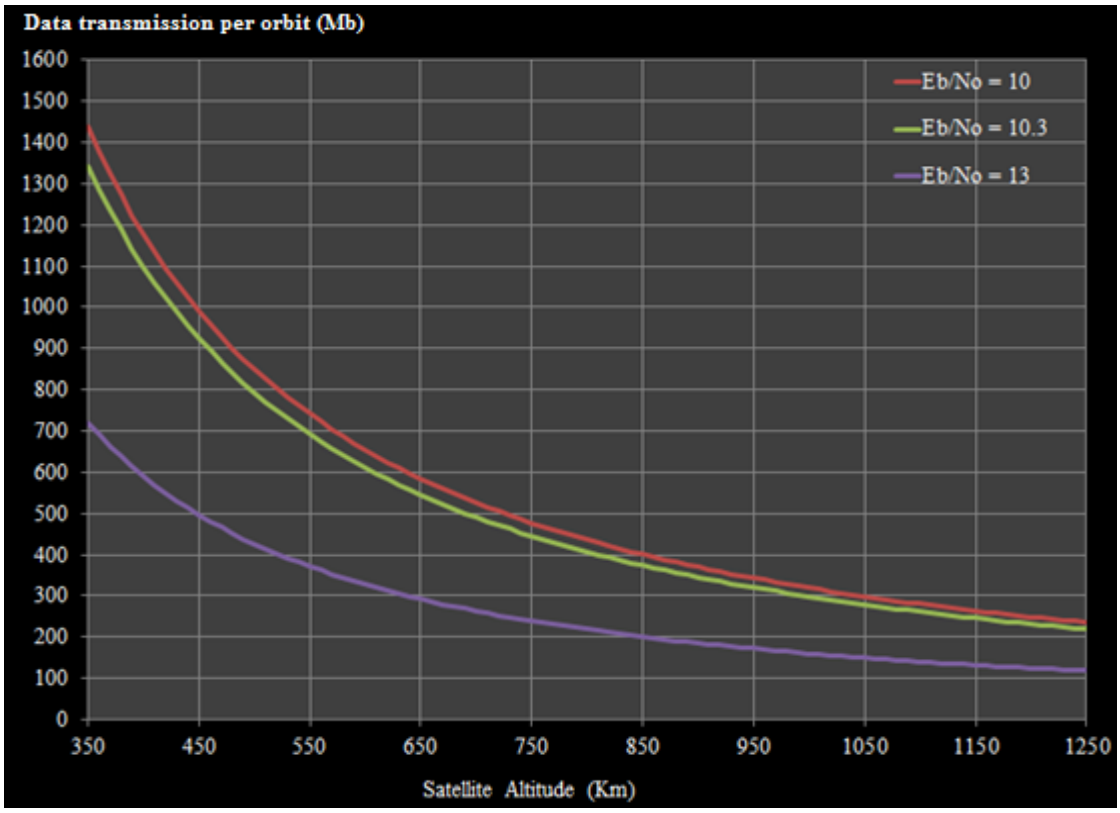


Figure 3-3 : The relationship between satellite altitude, E_b/N_o and data transmission rate.

The graph in Figure 3-4 illustrates the result of the link budget calculation for different satellite altitudes and various power transmissions of 0.5, 1 and 2 Watts and with a constant E_b/N_o of 10.3 dB. As expected, the results show that the system with the higher transmission power provides the higher data transmission rate. For example, at 650 km, the 2 Watt power transmission can download approximately 1,084 Mbits of data for a 10 minutes visibility window. Furthermore, the results show that less data can be transmitted to the ground station when the satellite is at higher altitude with fixed power transmission.

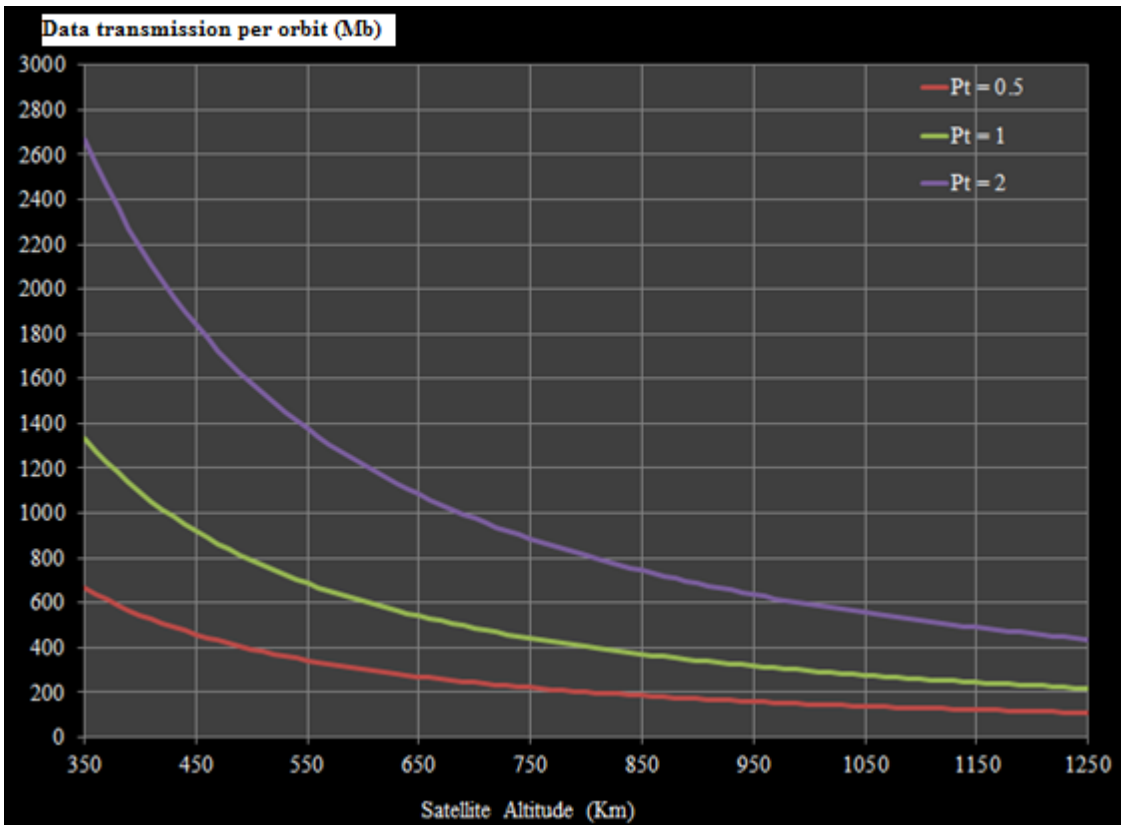


Figure 3-4 : The relationship between satellite altitudes, power transmission and data transmission rate.

3.3 Cubesat data transmission to the polar station

Most Earth observation satellites are sun-synchronous with an inclination around 98 degrees and a satellite attitude between 350 - 1300 km. In this section the orbit accessible to a ground station sited near the pole is shown. In the simulation, the assumption is that the ground station is at Kiruna (67.86° N and 20.97° E) and the satellite is in a 750 km sun-synchronous orbit with an inclination of 98.0° and the minimum elevation angle is 10°. Figure 3-5 shows the passes over the Kiruna station between UTC 11:17 02-May-2014 and 11:00 05-May-2014 (3 days).

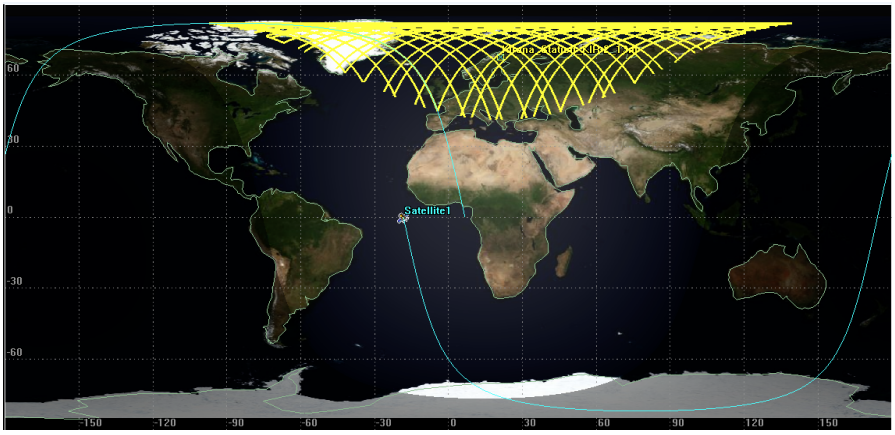


Figure 3-5 : The orbit of the satellite over Kiruna station (3 days).

Table 3-3 shows the summary report of the satellite (altitude at 750 km) access to the Kiruna station between 02-May-2014 until 05-May-2014 (3 days). There are 37 orbits visible to the ground station over these 3 days. The minimum, maximum and average visibility durations are 60, 855.76 and 725.25 seconds respectively in these 3 days. The total visibility duration is 27,301.29 seconds. The average visibility duration per day is around 9,100.43 seconds or 151.67 minutes.

Table 3-3 : The simulation access summary report of Satellite1-To-Facility-Kiruna_Station_KIR-2 between UTC 11:17 02-May-2014 to 11:00 05-May-2014.

Access	Start time (UTC)	Stop time (UTC)	Duration(Sec)
1	2 May 2014 11:17:45.034	2 May 2014 11:23:38.897	353.863
2	2 May 2014 16:26:07.933	2 May 2014 16:32:04.897	356.964
3	2 May 2014 18:04:42.154	2 May 2014 18:16:31.525	709.371
4	2 May 2014 19:43:47.528	2 May 2014 19:58:05.080	857.552
5	2 May 2014 21:22:51.074	2 May 2014 21:37:34.276	883.202
6	2 May 2014 23:01:38.727	2 May 2014 23:15:32.002	833.275
7	3 May 2014 00:39:54.170	3 May 2014 00:52:41.267	767.098
8	3 May 2014 02:17:25.109	3 May 2014 02:29:57.180	752.071
9	3 May 2014 03:54:30.530	3 May 2014 04:07:55.190	804.660
10	3 May 2014 05:32:02.671	3 May 2014 05:46:32.061	869.390
11	3 May 2014 07:10:50.989	3 May 2014 07:25:30.943	879.954
12	3 May 2014 08:51:30.661	3 May 2014 09:04:37.447	786.786
13	3 May 2014 10:34:37.747	3 May 2014 10:43:32.987	535.241
14	3 May 2014 17:24:17.954	3 May 2014 17:34:18.197	600.243
15	3 May 2014 19:03:18.249	3 May 2014 19:16:53.116	814.867
16	3 May 2014 20:42:24.339	3 May 2014 20:57:08.937	884.598

17	3 May 2014 22:21:20.210	3 May 2014 22:35:39.584	859.374
18	3 May 2014 23:59:50.895	4 May 2014 00:13:02.253	791.357
19	4 May 2014 01:37:40.096	4 May 2014 01:50:09.076	748.979
20	4 May 2014 03:14:50.777	4 May 2014 03:27:48.459	777.682
21	4 May 2014 04:52:05.207	4 May 2014 05:06:11.453	846.246
22	4 May 2014 06:30:17.674	4 May 2014 06:45:03.310	885.636
23	4 May 2014 08:10:08.250	4 May 2014 08:24:08.333	840.083
24	4 May 2014 09:52:09.823	4 May 2014 10:03:11.949	662.126
25	4 May 2014 11:37:30.423	4 May 2014 11:41:20.076	229.652
26	4 May 2014 16:44:02.700	4 May 2014 16:51:31.655	448.956
27	4 May 2014 18:22:49.663	4 May 2014 18:35:18.377	748.714
28	4 May 2014 20:01:55.933	4 May 2014 20:16:26.151	870.218
29	4 May 2014 21:40:57.571	4 May 2014 21:55:35.698	878.127
30	4 May 2014 23:19:40.567	4 May 2014 23:33:20.604	820.037
31	5 May 2014 00:57:48.134	5 May 2014 01:10:26.909	758.775
32	5 May 2014 02:35:11.526	5 May 2014 02:47:49.242	757.717
33	5 May 2014 04:12:17.565	5 May 2014 04:25:55.586	818.021
34	5 May 2014 05:50:00.678	5 May 2014 06:04:37.826	877.148
35	5 May 2014 07:29:07.356	5 May 2014 07:43:39.143	871.786
36	5 May 2014 09:10:11.369	5 May 2014 09:22:45.488	754.119
37	5 May 2014 10:53:52.601	5 May 2014 11:00:00.000	367.399
Min Duration orbit number 25			229.652
Max Duration orbit number 22			885.636
Mean Duration			737.873
Total Duration			27301.287

Table 3-4 shows the access duration of the satellite passes over the Kiruna station at different satellite altitudes from 350 to 1,250 km between midday 02-May-2014 to midday of 05-May-2014. It clearly shows that the average visibility duration of the satellite to the ground station is longer when the satellite altitude is higher.

Table 3-4 : The average visibility duration over Kiruna station for satellites at different altitudes

Satellite altitudes	The average duration of the visibility from the Kiruna ground station	
	Seconds	Minutes
350 km	5,069.80	84.50
450 km	6,134.05	102.23
550 km	7,073.67	117.89
650 km	8,032.68	133.88
750 km	9,100.43	151.67
850 km	10,054.21	167.57
950 km	11,070.45	184.51

1,050 km	12,135.09	202.25
1,150 km	12,935.98	215.60
1,250 km	13,613.97	226.90

This information will be used to calculate the data that can be transmitted to the ground station by the cubesat with different constraints, such as satellite altitude, power transmission and Eb/No.

Table 3-5 shows the amount of data that can be transmitted per day with the limitation of the E_b/N_o of 10.3 dB as a function of transmitted power. The higher the transmitted power the more data can be downloaded to the ground station.

The graph in Figure 3-6 illustrates the result of the link budget calculation for different satellite altitudes and various power transmissions of 0.5, 1 and 2 Watts with a constant E_b/N_o of 10.3 dB. As expected, the results show that the system with the higher transmission power provides the higher data transmission rate with more transmission data. For example, at 650 km, the 2 Watt power transmission can download approximately 14,515.4 Mbits (14.5 Gbits) of data per day to the polar station.

Table 3-5 : Total data download possible to the ground station at Kiruna per day for cubesat in the different altitudes and power transmission at E_b/N_o of 10.3 dB.

Satellite altitudes	Kiruna ground station			
	Average duration of the visibility	Data download per day (Mb)		
	Minutes	P = 0.5	P = 1.0	P = 2.0
350 km	84.50	5,638.48	11,276.95	22,553.90
450 km	102.23	4,702.89	9,405.79	18,811.58
550 km	117.89	4,055.16	8,110.32	16,220.63
650 km	133.88	3,628.86	7,257.71	14,515.43
750 km	151.67	3,360.09	6,720.19	13,440.37
850 km	167.57	3,116.83	6,233.66	12,467.31
950 km	184.51	2,941.20	5,882.39	11,764.79
1,050 km	202.25	2,808.12	5,616.24	11,232.49
1,150 km	215.60	2,641.71	5,283.41	10,566.83
1,250 km	226.90	2,480.07	4,960.15	9,920.30

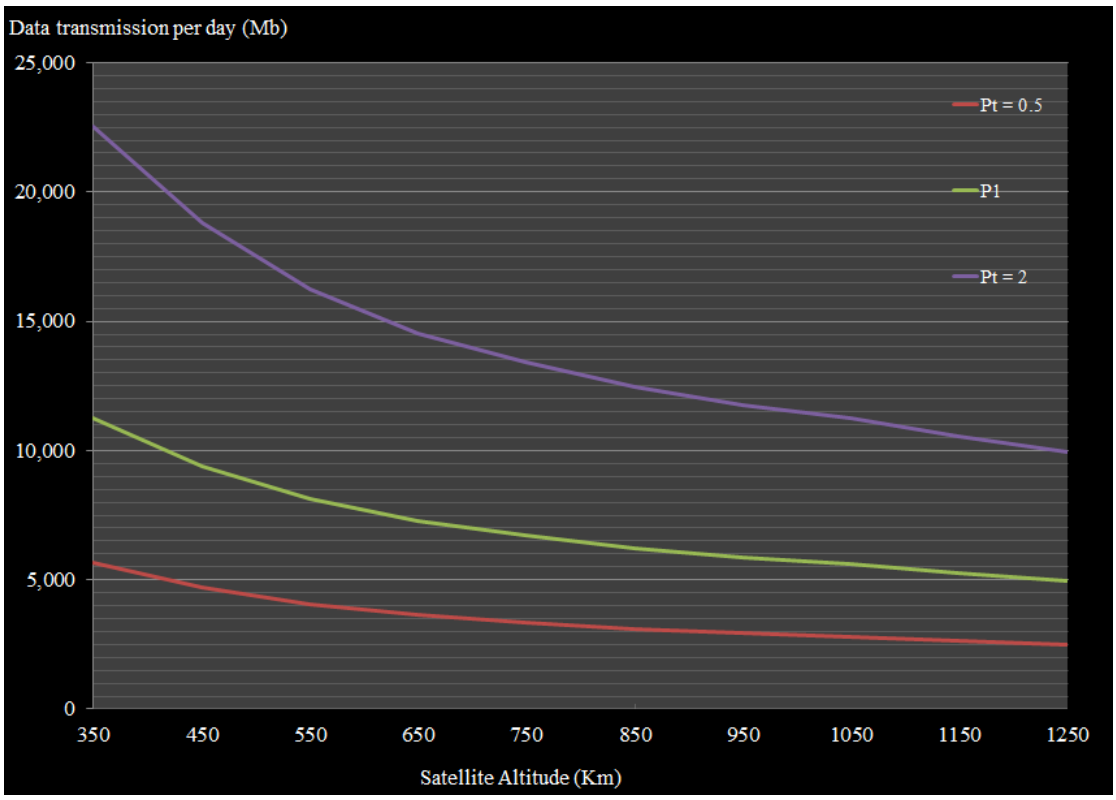


Figure 3-6 : The relationship between satellite altitudes, power transmission and data transmission per day at polar ground station.

Table 3-6 shows the relationship between the cubesat altitudes from 350 to 1,250 km and the amount of data that can be transmitted per day over Kiruna station using different E_b/N_o. The E_b/N_o used was 10, 10.3 and 13.0 dB. The less E_b/N_o required, the more data that can be downloaded to the ground station.

The graph in Figure 3-7 shows the amount of the data that can be transmitted from a cubesat down to the ground station per day. Different satellite altitudes from 350 to 1,250 km were used. For instance, at a satellite altitude of 650 km with a power transmission of 1 Watt and a E_b/N_o of 10.3 dB, the cubesat will then be able to transfer 7,293.43 Mbit (7.2 Gbit) of data to the Kiruna station. If the required E_b/N_o is 13 dB, around 3,916.81 Mbit (3.9 Gbit) of data can be transmitted to the ground. (Further details of the link budget calculation are given in Appendix A).

Table 3-6 : Total data download possible to the ground station at Kiruna per day for cubesat in the different altitude and E_b/N_o of power transmission at 1 Watts.

Satellite altitudes	Kiruna ground station			
	Average duration of the visibility	Data download per day (Mb)		
	Minutes	Eb/No = 10	Eb/No = 10.3	Eb/No = 13
350 km	84.50	12,142.94	11,332.46	6,085.89
450 km	102.23	10,128.09	9,452.08	5,076.07
550 km	117.89	8,733.13	8,150.23	4,376.93
650 km	133.88	7,815.06	7,293.43	3,916.81
750 km	151.67	7,236.25	6,753.26	3,626.72
850 km	167.57	6,712.36	6,264.33	3,364.15
950 km	184.51	6,334.12	5,911.34	3,174.58
1,050 km	202.25	6,047.53	5,643.89	3,030.95
1,150 km	215.60	5,689.14	5,309.42	2,851.326
1,250 km	226.90	5,341.05	4,984.56	2,676.87

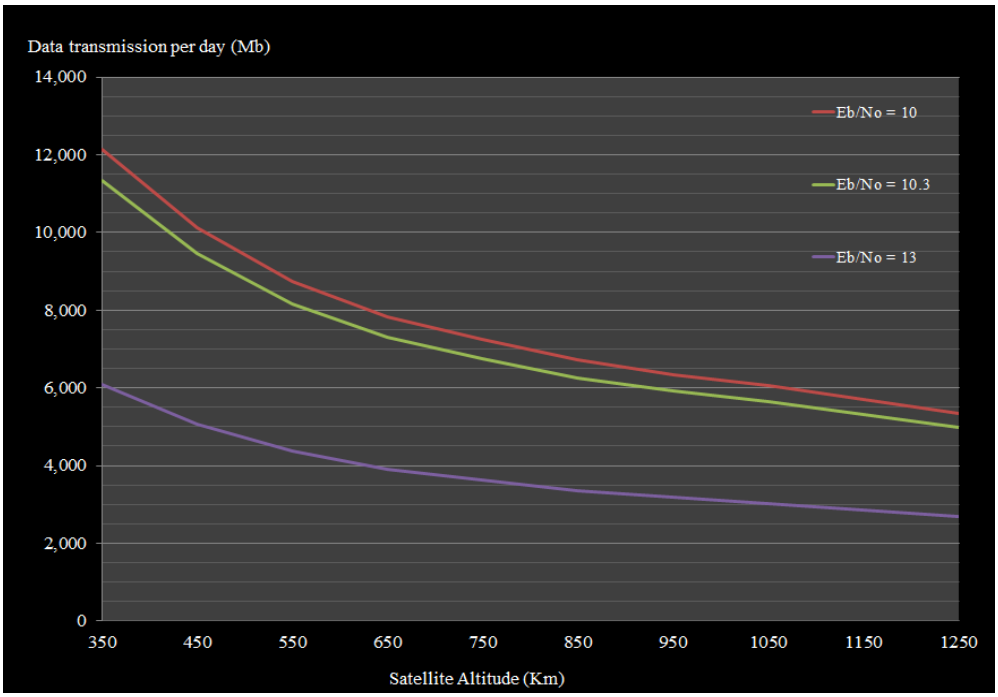


Figure 3-7 : The relationship between satellite altitude, E_bN_o and data transmission per day at polar ground station.

This can be illustrated by considering the NanoCam camera from Gomspace Company. One image consists of 2048x1536 pixels with 10-bit colour (Pedersen 2007-2012). This camera acquisition duration is 5 seconds for one image and it can process the image in approximately 90 seconds. One image can therefore be acquired and processed in 95 seconds.

One uncompressed image of this camera would occupy 94.37 Mbit (3.1M x 10 x 3 colour channels) for one image. Using the link budget calculation and assuming the satellite is able to download data whenever the ground station is visible with the limitation of one day Kiruna visibility at 650 km altitude and transmits 1 Watt, it is possible to download 7,293.43 Mbits (7.3 Gbits) of data. Consequently, a cubesat can transmit approximately 76.9 (7293 /94.37) images to the Kiruna receiving station without compression or header data in one day. This is a small fraction of the images that could be acquired

The following compares typical data generation volume with downlink volume to estimate the required data reduction factor required for compression algorithms. In the following example, the assumption is that cubesats have sufficient power and storage to capture and store images on-board continuously during the time window that the sun's solar power can be harnessed from highly efficient solar panels systems. The NanoCam camera needs 95 seconds to capture and process an image (Pedersen 2007-2012). In one day a satellite in orbit will maintain contact with the sun for approximately 720 minutes ((24 hours x 60 minutes)/2). Consequently, cubesats can capture images of the earth continuously and generate 454.7 images per day ((720 minutes x 60 seconds)/95 seconds to capture an image). In the case that a cubesat will download images to the Kiruna station for example, 76.9 images can be transferred per day (as simulation result above). Therefore to download all 454.7 images to ground station in the available time window, a compression factor for the onboard images required is 5.9 (454.7/76.9).

It can be seen that the data downlink rate imposes a considerable constraint on the imaging capability of a cubesat. It is therefore particularly important that only data that is required is transmitted and data compression is implemented where possible.

An alternative approach is to use a high speed S-band link. According to Equation 3.1, if the frequency is changed from 437 MHz to 2.4 GHz whilst keeping the other variables constant, the resultant data transmission rate will change by a ratio of $1/\lambda^2$ so the data rate will increase by a factor of (0.686/0.125)^2 i.e., 30.16. In practice this is not possible because it assumes that the satellite antenna gain increases such that it would have to track the ground station. A typical data rate that can be expected from a high speed communications link is 2Mb/s. This would

allow 16,066 Mbits to be downloaded or 170 uncompressed images. This is still a fraction of the images that could be available.

3.4 Summary

The link budget is useful for preliminary communication system design. The device parameters can be adjusted to find the suitable range of values that can be used. Moreover, the link budget preliminary study is a trade-off between internal communication system parameters and other satellite sub-systems, such as the command and data handling system design that restricts the data sent to a receiving station during visibility. Due to the limitation of power transmission, the data transmission rate is limited. Consequently, any significant reduction in data or image size can have an important effect on the design of the cubesat.

In this chapter it can be seen that the number of images that can be transmitted is significantly curtailed by the cubesat communication subsystem.

Chapter 4

Onboard computer

There are a considerable number of limitations on the available power and data storage capacity of a cubesat. These constitute important constraints on the cubesat spacecraft subsystems such as communication, command and data handling subsystem (CDHS) and payload subsystem. This research analyses the onboard computer of a small satellite such as a cubesat, identifies alternative architectures and examines the potential for onboard processing.

Design of the onboard computer has many important constraints and requirements. These include; low power usage, automatic operation, reliability and simplicity. On the other hand there is a significant potential for more flexibility to be incorporated into a cubesat onboard computer. The computer needs to be effectively managed to use the power available from the electrical power system and be compatible with the overall satellite power budget. This chapter presents the concept of a new hybrid onboard computer architecture, which is an alternative architecture applicable to the design of general onboard computer architectures of future cubesat projects.

4.1 Onboard computer architecture

There are two general architectures possible for an onboard computer; central and distributed architecture.

A central architecture performs the CDHS processing at the centre of the satellite. It is connected with all the other sub-systems as seen in Figure 4-1 (Davidsen 2001; Aalbers, Gaydadjiev et al. 2005). The advantage of this architecture is reduced complexity. As a consequence of this, the system can be designed with simpler hardware, which reduces the chance of communication problems between subsystems. On the other hand, the disadvantages are a large single point of failure, minimal flexibility, non-scalability and processing power sharing between each subsystem (Teney 2009).

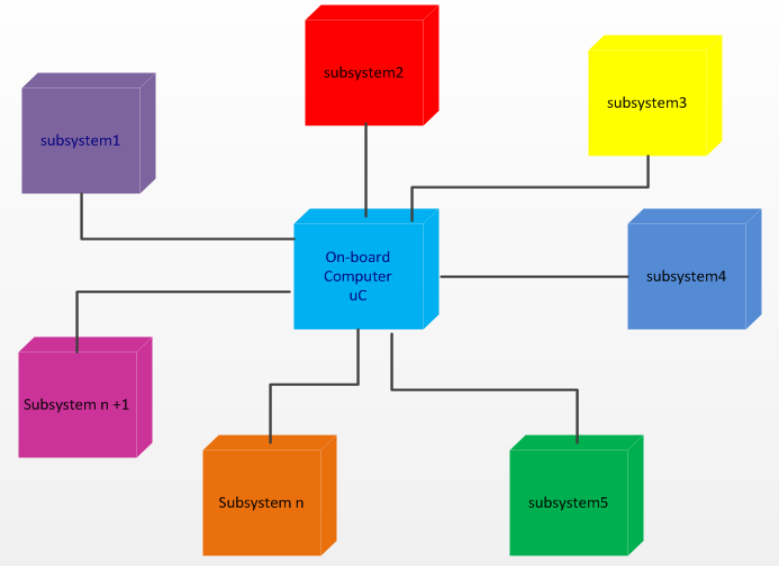


Figure 4-1 : The central architecture.

In the distributed architecture the onboard computer is still the main priority, as it's responsible for managing the functions of the satellite. However each subsystem is equipped with its own processor or microcontroller. This means the different subsystems can be independently developed. The onboard computer and the other subsystems are connected to each other via a data bus. Each subsystem is able to contact each other without going through the onboard computer as shown in the architecture diagram in Figure 4-2 (Davidsen 2001; Aalbers, Gaydadjiev et al. 2005). The advantages of the distributed architecture are that it is modular, scalable, flexible and that independent testing of each sub-system is more practical. There are some disadvantages, such as increased complexity of hardware and software; one faulty sub-system could interrupt the bus and thus all the other subsystems. The advantages and disadvantages of both architectures are summarised in Table 4-1 (Davidsen 2001; Aalbers, Gaydadjiev et al. 2005).

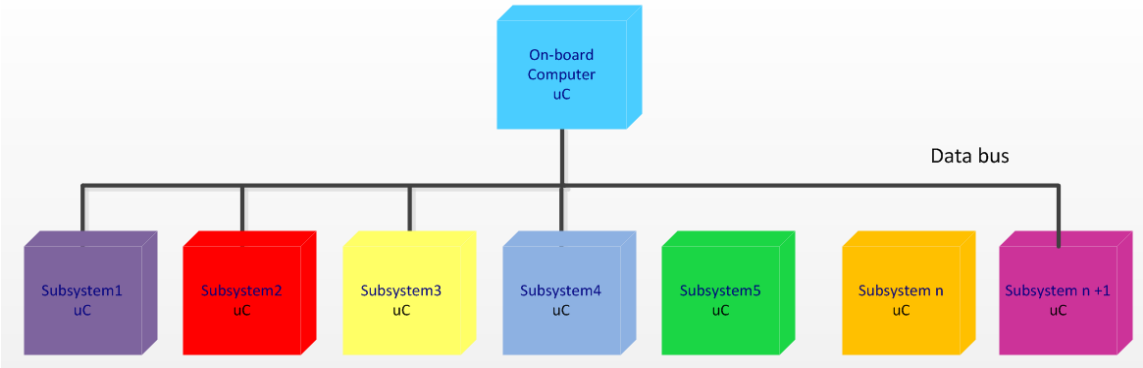


Figure 4-2 : The distributed architecture.

Table 4-1 : Central vs. Distributed onboard computer architecture.

Architecture	Advantages	Disadvantages
Central	• Simple to develop and maintain • Simple to debug • Fewer communication problems • Simple hardware	• Difficult to upgrade • Lack of flexibility and redundancy • Shared processing power between subsystems
Distributed	• Reliable and robust • Easy to upgrade • Excellent for complex systems • Many possible communications • Independent testing of subsystems is more practical	• Complex hardware/software • Long implementation schedule • Large size • Component number increased • Power consumption increased

Cubesat requirements and specifications focus on small size, low mass and low power consumption. The onboard computer operates continuously after launch has been completed. The onboard computer consumes various amounts of power depending on the operational mode. For example in the standby mode the onboard computer requires a very small amount of power. The power budget calculation report from the design class that developed a cubesat platform for science ionosphere modelling at university of Washington (Waydo, Henry et al. 2002) shows that the estimated standby power for the onboard computer is approximately 0.05 Watts. Whereas during operational mode the onboard computer will operate to full capacity with corresponding maximum power usage i.e., 0.3 W (Waydo, Henry et al. 2002). As cubesat satellites grow in complexity, this places additional constraints and requirements on system components. For more advanced missions, a high speed onboard computer for calculation, execution or heavy processing is necessary. Greater flexibly of the onboard computer

architecture is required to support the mission adaptation or changing specifications of onboard devices.

4.2 Hybrid onboard computer architecture

The onboard computer architecture needs to be very carefully designed as the volume and resources available on a cubesat are limited. The onboard computer is responsible for the control and management functions of the satellite. It is always operating continuously in orbit after launch. In this section, an alternative onboard computer architecture for the next generation of cubesats that can be used for more complex missions is presented. In addition, the architecture should be sufficiently flexible that it can be used in various missions (Kiadtikornthaweeyot 2010). There are advantages and disadvantages of both central and distributed architectures (Aalbers, Gaydadjiev et al. 2005; Scholz 2005; Teney 2009), and it seems logical to combine both systems if possible by using the advantages of one architecture in combination with the advantages of the other architecture. This chapter focuses on the hybrid onboard computer architecture that benefits from the advantages of both the central and distributed architecture (Teney 2009; Kiadtikornthaweeyot 2010). There are two key points for the new hybrid onboard computer architecture design:

- Firstly, the communication system is vital for a cubesat; without it the cubesat will not be able to communicate with the ground. Giving the communication subsystem redundancy gives the cubesat more reliability.
- Secondly, the payload subsystem is designed and implemented with its own individual microprocessor. This method gives an individual development opportunity for the payload engineer and makes design for more complex missions possible. Also this gives more flexibility to the system. Separating the payload subsystem from other subsystems enables the payload to be altered or updated, depending on the mission or available space technology.

Consequently, the new hybrid onboard computer architecture has been designed to provide benefits for both the central and distributed architectures. The alternative hybrid onboard architecture diagram is illustrated in Figure 4-3. The architecture consists of two main microcontrollers.

- The first microcontroller is for the supporting subsystems, which consist of onboard computer-1 (OBC-1), power, altitude and orbit control subsystem (AOCS), beacon and main communication subsystems (Com-1). The OBC-1 is

the main subsystem of this board.

- The second microcontroller is the payload for the satellite. The onboard computer-2 (OBC-2) is the main controller of the payload device. The OBC-2 type depends on the satellite mission. This board can be designed independently by a different team, which gives the system design more flexibility.

The hybrid onboard computer architecture has both a main and redundant communication subsystem and bus system to increase system reliability. Therefore, both microcontrollers are developed separately. However both boards are able to operate in a complementary way to recover the satellite in case of failures during operation. Table 4-2 shows the status of each subsystem during different operating scenarios including failure cases.

Firstly during normal operation, the satellite operates using OBC-1 as the main control system as shown as the 'green coloured line' in Figure 4-3. The OBC-2 operates as the payload subsystem.

The second case is when the OBC-1 is inactive. The system will switch to use OBC-2 to be the main control system. The OBC-2 can control all subsystems according to software implemented in the development phase for this failure recovery case.

The next failure case is if the communication-1 fails, which is controlled by OBC-1. The redundant communication system (communication-2) will be activated to operate under the control of OBC-1 as shown in the diagram (Figure 4-3).

Fourthly is the case where the OBC-2 fails. This board controls the payload of the satellite. The proposed architecture gives reliability to the satellite by adding back up operation capability between OBC-1 and OBC-2. Consequently both OBC's are a back-up system to each other. Hence OBC-1 is able to operate the payload subsystem if OBC-2 should become inactive.

The last case is if the main Bus-1 fails, which is shown as the 'green coloured line' in Figure 4-3. In this case the satellite is able to use the redundancy Bus-2 to continue operation.

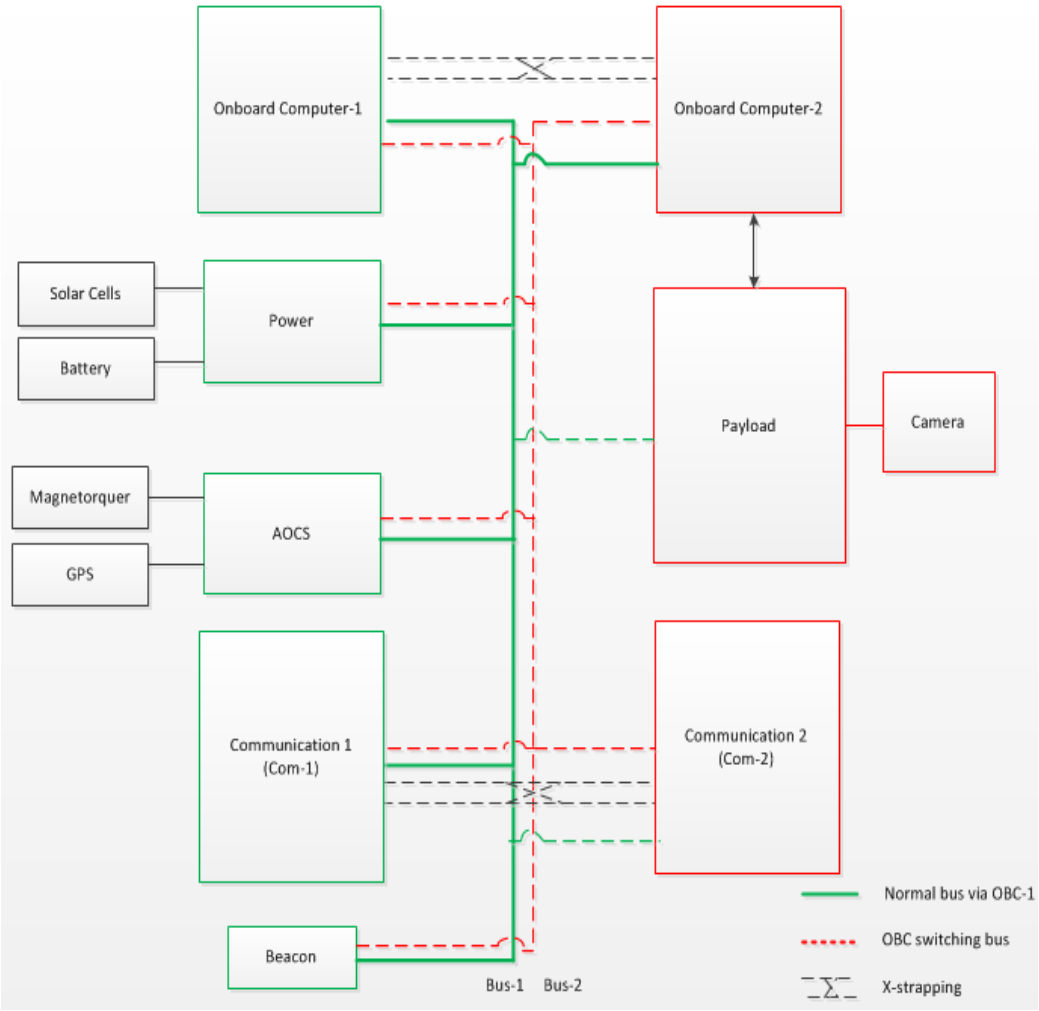


Figure 4-3 : The new hybrid onboard computer architecture.

Table 4-2 : Hybrid onboard computer operation scenarios for different failure cases.

Operation case scenarios	**Satellite subsystem and status**									
	OBC-1	Power	AOCS	Com-1	Beacon	Bus-1	OBC-2	Payload	Com-2	Bus-2
1. Normal operation	ON	ON	ON	ON	ON	ON	ON	ON	OFF	OFF
2. OBC-1 fail	OFF	ON	ON	ON	ON	ON	ON	ON	OFF	OFF
3. Com-1 fail	ON	ON	ON	OFF	ON	ON	ON	ON	ON	OFF
4. OBC-2 fail	ON	ON	ON	ON	ON	ON	OFF	ON	OFF	OFF
5. Bus-1 fail	ON	ON	ON	ON	ON	OFF	ON	ON	OFF	ON

The benefits of the new hybrid onboard computer architecture are summarised in Table 4-3. The hybrid architecture provides more flexibility to the system by dividing the subsystems into two different board controls. Both microcontrollers can be designed and developed independently. If the second board has to be upgraded for payload functions, it is possible and does not affect the whole of the first board. Moreover, the second microcontroller can be developed as the project requires within the limitations of the available resources. Additionally the architecture gives redundancy for the communication system and data bus system.

Table 4-3 : The hybrid architecture advantages.

Hybrid advantages	**Architecture**	**Determinant**
1. More flexibility than,	Central	Two different boards provide the development phase flexibility due to the development teams being able to work separately
2. More redundancy than,	Central	The redundancy communication subsystem increases the reliability for the cubesat communication system
3. Not as complex to develop as,	Distributed	There are only two boards that need to be integrated in the assembly process
4. Easier to upgrade than,	Central	As the payload and supporting subsystems are separated, each team can develop their own system regarding the final interface
5. Lower power consumption than,	Distributed	In the hybrid architecture, the power is required to operate for two boards, whereas in distributed architecture extra power is required to distribute the power for every subsystem

The hybrid architecture requires more power and mass than the central architecture. Therefore, the power and mass required are investigated in order to find the possibilities/limitations of implementing the new onboard computer in a cubesat satellite (Kiadtikornthaweeyot 2010).

4.3 Preliminary analysis

In the previous section, it can be seen that the implementation of the new hybrid onboard computer architecture consists of at least two microcontrollers. Consequently, the system requires more power to operate (Kiadtikornthaweeyot 2011). Moreover, the satellite mass will increase compared to the traditional onboard computer system. In this section a study is made of the increase in the power and mass of the cubesat that is required using the proposed hybrid onboard computer architecture.

4.3.1 Power study

To study the power consumption of a cubesat subsystem, an example cubesat mission has been considered. Table 4-4 shows the power consumption for each subsystem of the cubesat mission, for an ICUBE-1 cubesat, which is the student satellite program from Pakistan. This information's source is from "*Institute of Space Technology CubeSat : ICUBE-1 Subsystem analysis and design"* (Mahmood, Khurshid et al. 2011). The cubesat has a passive attitude control and a camera.

Table 4-4 : The power budget for each subsystem of the ICUBE-1 mission.

Subsystem	Power (mW)
Onboard computer	66
Electrical power supply	100
Payload	200
Transmitter	2,000
Receiver	300
Beacon	700
Total	3,366

The key points to note:

- The onboard computer requires 66 mW to operate. The new hybrid onboard computer architecture uses two microcontrollers and if both boards operate at the same time, the command and data handling system will consume 132 mW (66 mW for each board).

- The payload subsystem consumes 200 mW of power. The idea is to separate this subsystem and integrate it in the first board. The second board is for the supporting subsystem. If both boards are in the operational mode the following occurs: The first board for the payload subsystem (camera) consumes 200 mW and 66 mW (i.e. 266 mW). The second board, which consists of the rest of the subsystems (electrical power supply, transmitter, receiver and beacon), consumes 3,100 mW (100 mW + 2,000 mW + 300 mW + 700 mW) plus the power for the onboard computer of 66 mW. The overall power consumption for both boards will be 3,432 mW (266 mW + 3,166 mW).
- The communications subsystem consumes most of the power, around 3,000 mW. The design of a redundant communication subsystem involves adding an additional communication subsystem in cold redundancy. In case the main communication subsystem device fails, the system will switch to use the redundant one.

From this review, it can be seen that the new onboard computer hybrid architecture design can be implemented in the cubesat spacecraft, with a total power increase of no more than 66 mW, i.e. the power for the second microcontroller.

4.3.2 Mass study

Table 4-5 shows hardware devices specification, mass and power consumption for COTS onboard computer and communication systems (Kiadtikornthaweeyot 2010). The information is compiled from space industry companies, for example Pumpkin Inc and GOMspace. There is also information from compass 1 and ICUBE-1, which are university cubesat projects (Karatas and Ince 2009).

Assuming the cubesat project plans to use two onboard computers, such as the microcontroller FM430, this will require an extra mass of 74 g. The communication subsystem also requires redundancy, i.e. another 185 g. In total the project has an additional microcontroller and communication system and so has an extra mass of 259 g (74 g and 185 g). In addition a very small mass for the redundancy bus is required. It can be seen that the new onboard hybrid architecture design makes it possible to implement multi-microcontrollers, with limited additional mass required.

Table 4-5 : Power and mass of the COTS onboard computer and communication subsystem.

Subsystem	Company	Model	Power	Mass
Onboard computer	Pumpkin Inc.	FM430	66 mW	74 g
	GOM space	NanoMind A702	250 mW	73 g
	Silicon Laboratories	C8051F12X	56 mW	61.5 g
Communication	ICUBE-1 mission	Transceiver and antenna	-	185 g
	RF Data Tech	LRT470TR-5	0.05 – 0.75 W	115 g
	Compass1 cubesat	A custom built transceiver	1.02 W	130 g

4.4 Summary

The proposed alternative onboard computer has a hybrid architecture that has the advantages of both a central and a distributed architecture. The preliminary power and mass study show that the proposed architecture has potential and can be implemented on a cubesat project with COTS devices. As a result, extra power and mass are required to use the proposed architecture. Therefore, a trade-off study would have to be conducted to fully evaluate the feasibility of using a hybrid architecture in the real design for a particular mission.

Chapter 5

Region of interest (ROI)

There are many cubesats with imaging instruments that provide high resolution remote sensing images. These instruments are producing increasingly large amounts of data (Jian-wei and Jian-dong 2008; Yu 2009). This increase requires more data transmission time, power and onboard storage memory, so the provision of high performance cubesat image processing opens further possibilities for cubesats for many applications (Pen-Shu, Armbruster et al. 2005; Tahoces, Varela et al. 2008). The challenge is how to provide this capability within the limited resources available on a cubesat. As a consequence of all the above problems, one of the possible approaches is to reduce the size of the images produced by image segmentation techniques and image compression methods onboard (Ye, Gu et al. 2008). In this research an adaptive image compression algorithm to decrease the onboard satellite image size that is suitable for adoption on a cubesat is described. The proposed algorithm consists of two main systems and the first part, the ROI automatic detection, is presented in this chapter. The second module, the image compression module, is demonstrated in chapter 6. The experimental results of these image segmentations are presented and the results of varying the structure element size and shape for the morphological technique are described at the end of this chapter.

Identification of the ROI of an image can be performed using a manual or automatic process. A real satellite image will contain a variety of features, textures or shadows and can be very complex. Consequently, automatic extraction of the required ROI from within an image is very difficult using only one image segmentation technique. The solution proposed is to use a combination of different image segmentation techniques to detect the region of interest effectively. There are many image segmentation methods. This research identifies the suitable image segmentation techniques that can be used to detect the ROI of the satellite image automatically. There are three basic image segmentation techniques that have been studied and tested:

- Edge segmentation
- Histogram segmentation
- Texture segmentation

All of these proposed segmentation methods based on the edge, histogram and texture have some common steps. There are as follows:

- RGB conversion to grey scale method if three channels are available
- Black and white image conversion
- Morphological dilation
- Morphological fill image the regions and holes
- ROI code management

These five processes have been described in section 5.1 and 5.2.

5.1 Edge segmentation

The edge segmentation is based on finding the edges in an input image by finding the gradient magnitudes of the image. The proposed edge segmentation flow chart is summarised in Figure 5-1.

The process starts with converting an input satellite image into grey scale images (Gonzalez and Woods 2007; Saravanan 2010). This process converts the brightness of each pixel to a grey level. There are many methods to make this conversion. A suitable method for each image depends on the nature of the colour image and the application (Tanner 2012). The next step is edge detection; the edges are defined by a change of the grey level intensity in a narrow area. Then the edge segmented image is applied by using a process called morphological dilation, which groups nearby areas together by connecting edges that are near to each other. The dilation method operates by using morphology structure elements to process the input image. As a result, associated edges are linked to each other. However, if the edges have a distance between each other that is longer than the size of the structure element, this space will be left as a hole in the image.

The next process is the morphological filling of image regions and holes. This process fills the holes in an image by detecting when an area of dark pixels is surrounded by white pixels. The dark pixels in the middle of white pixels will be converted to white pixels. The technique is based on dilation, complementation and intersection for filling holes with a symmetric structuring element (Gonzalez and Woods 2007; Al-amri, Kalyankar et al. 2010). The final step is the application of the ROI mask to the original image. The ROI image is the result of this step. The process of the ROI automatic detection based on edge detection has been described for each sub-process as detailed below:

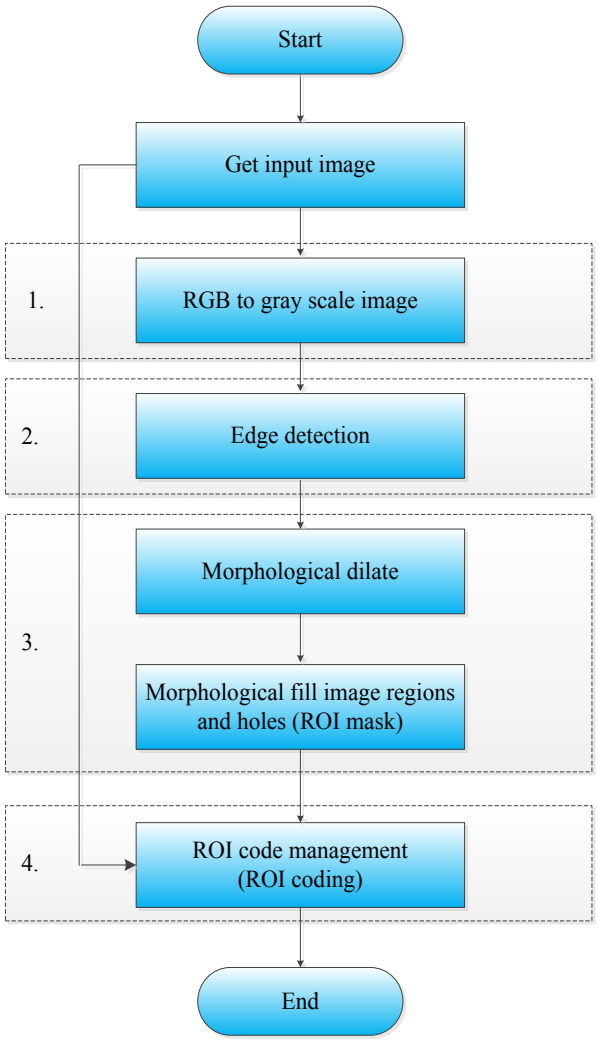


Figure 5-1 : The flow chart of ROI automatic detection based on edge segmentation.

The experiment has been performed by Matlab R2010a to demonstrate the behaviour of the proposed algorithm. The input images are from NASA's website (Przyborski 2010) and the details of each satellite image are presented in Appendix B. This section shows the experimental results for an example image and demonstrates the algorithm's behaviour for each sub-process, step by step. Further results on various satellite images can be found in Appendix C. Figure 5-2 shows the example input, which is a satellite image of the Sarychev volcano at Matua Islan with the clouds on the left hand side of the image shown as a white pattern. The image was acquired on June 2010 by NASA's Earth observing-1 (EO-1) satellite (Przyborski 2010).

Figure 5-2 : The input satellite image for the proposed ROI automatic detection algorithm based on edge detection.

5.1.1 Grey scale image

After retrieval of the input satellite image, the process starts with converting an RGB input image to a grey scale image. This process converts the brightness of each pixel to a grey level, which usually consists of 256 levels (Saravanan 2010; Gonzalez and Woods 2007; Koyanagi 2012). Equation 5.1 shows one way a colour image can be converted to a grey scale image (Grundland and Dodgson 2005; Pratt 2007; Kanan and Cottrell 2012; Tanner 2012). Each of the RGB channels could be used to generate a grey level image individually but using a combination of the channels can prove useful. As an example the constants used for each channel shown in Equation 5.1 results in a better approximation of the human visualizer's sensitivity to each of those colours (Kanan and Cottrell 2012). I_{gray} is the grey scale image, and I_r , I_g and I_b represent the image pixel value in red, green and blue channels respectively.

$$I_{grey} = (I_r \times 0.30) + (I_g \times 0.59) + (I_b \times 0.11) \qquad (5.1)$$

Figure 5-3 illustrates the grey scale image, which is converted from the RGB original image.

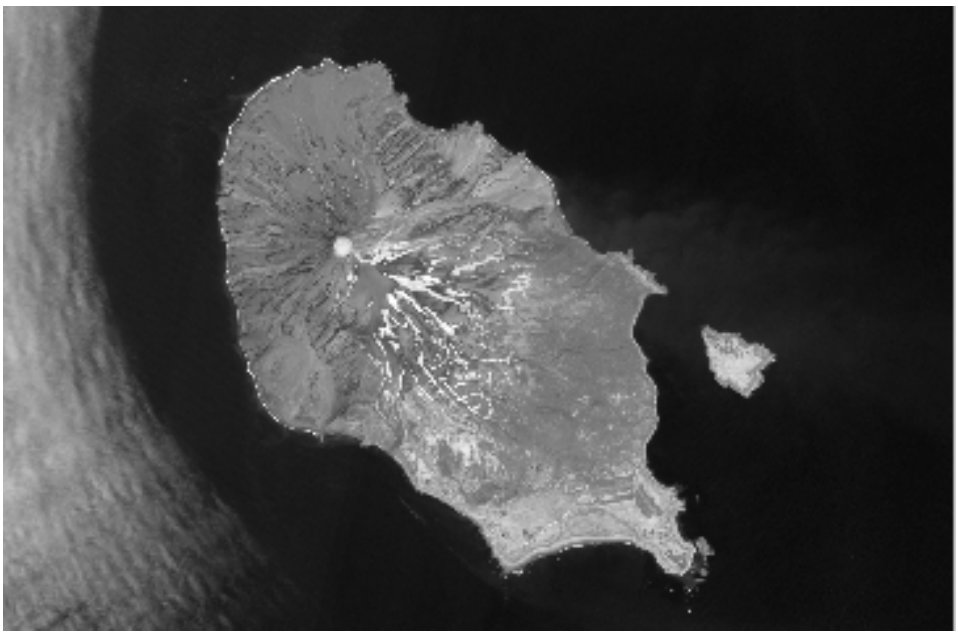


Figure 5-3 : The grey scale image.

5.1.2 Edge detection

The assumption made is that the ROI can be characterised by edges and structures, which can be identified using edge detection. The edge detection technique is based on finding the edges in an input image by approximating the gradient magnitudes of the image. There are many edge detection methods such as Canny, Laplacian, Roberts, Sobel and gradient (Ali and Clausi 2001). The Canny edge detection method is selected for use in this module due to its good performance because Canny edge detection supports filling of gaps at the detected edges (Muthukrishnan.R and Myilsamy 2011; MathWork 2012). The Canny algorithm processes are explained in the following (Gonzalez and Woods 2007; Al-amri, Kalyankar et al. 2010; Al-Kubati, Saif et al. 2012):

- ***Compute gradient*:** The first step is to filter out noise in the input image before defining any edges. A gaussian filter is used to smooth the image and can be computed using a mask with convolution methods (Gonzalez and Woods 2007). The gradient of an image *I(x,y)* is defined as the vector by the equation 5.2.

$$\nabla I = [G_x, G_y] = [\partial_I/\partial_x\,, \partial_I/\partial_y] \qquad (5.2)$$

Where G_x and G_y are the gradients in direction x and y of the image $I(x, y)$ respectively. Subsequently the edge gradient strength and direction is found for each pixel. The magnitude and direction of the gradient are given by equation 5.3 and 5.4 respectively.

$$|\nabla I| = \sqrt{{G_x}^2 + {G_y}^2} \tag{5.3}$$

$$\alpha(x, y) = \tan^{-1}(G_y / G_x) \tag{5.4}$$

- ***Non-maximum suppression and hysteresis*:** the first step is to find the edge by tracing along the image by using magnitude and direction. If the pixels are not a part of the local maxima they are suppressed. For the edges defined at this stage, some edges are likely true edges but some edges might be noise or colour variation.

  The next step is hysteresis processing to classify edges using double thresholding. The Canny edge detection method defines an edge by using upper and lower thresholds. There are many techniques to select these threshold values and a standard method does not exist to determine the most successful result for image processing (Fang, Yue et al. 2009). If a user specifies a value for the threshold, this number can be used for the high threshold, and 0.4 x threshold is used for the low threshold. In Matlab programming, if the user has not specified the threshold, then the threshold is chosen automatically by choosing a value for this threshold relative to the highest value of the gradient magnitude of the image (MathWork 2012). This was the method used in this research. If the current pixel has a gradient strength greater than the upper threshold, this pixel is defined as an edge, but if it is below the lower threshold, then it is not defined as an edge. If the gradient of the pixel is between the two thresholds, then it will be an edge only if it is linked to a pixel that is defined as an edge. The pixel defined as belonging to an edge will be set to a 'white' colour, while all other pixels are set to a 'black' colour. Figure 5-4 shows the image after edge detection.

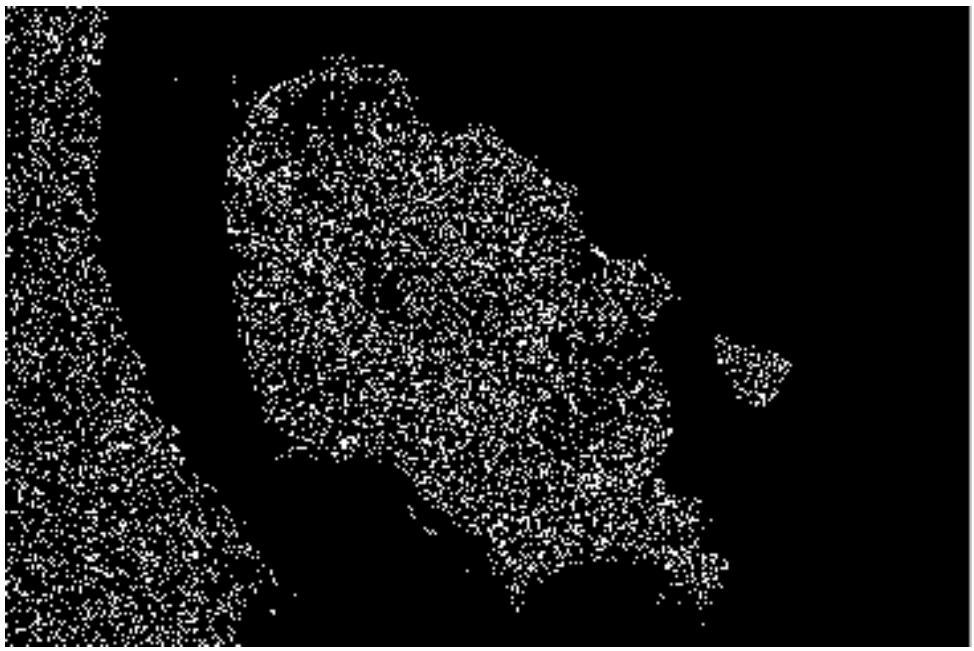


Figure 5-4 : The edge detected image.

5.1.3 Mathematical morphology

This process consists of two morphology techniques; 'dilation' and 'fill the regions and holes'.

- ***The dilation morphology technique***: After Canny edge detection has been performed the output image consists of many disconnected edges. A mathematical morphology technique is used to connect areas which are separated by smaller spaces than that of the structuring element used. The dilation which is implemented using this method gradually enlarges the boundaries of regions of objects. The dilation operator has two data inputs. The first one is the image and the second one is the structure element which is defined by size and shape (Ves, Benavent et al. 2006; Gonzalez and Woods 2007). A considerable amount of research has been performed on the morphology of the structuring elements (Hedberg, Dokladal et al. 2009; Wang 2009). The selection of a structuring element size and shape depends on the geometric shapes that need to be extracted from the input image. For instance, for extracting shapes from geographic aerial images of a city, a square or rectangular element will provide a good result. On the other hand, line structure elements are good at detecting roads in an image (IDL-online 2005). The selection of the structure element shape and size will be discussed

later in this chapter in section 5.5. Figure 5-5 demonstrates the result after image dilation has been performed.

Figure 5-5 : The image after dilation using a square structure element has been performed.

- ***Fill the holes technique:*** The image from the previous step is then processed by the region and hole filling technique. This technique fills image regions and holes and enables a boundary line to be identified (Xiao-hong, Zhu et al. 2009). The process works by detection of an area of dark pixels surrounded by white pixels. The ROI mask is generated in this step. The technique is based on dilation, complementation and intersection for filling holes (Gonzalez and Woods 2007; Al-amri, Kalyankar et al. 2010). More information about the technique used to fill holes is described in chapter 7, section 7.2. Figure 5-6 shows the regions and holes filled image and also the binary ROI mask image. The ROI location is defined by the value of a pixel; areas of interest have pixels with a value of '0' (black), whereas areas not of interest or the background have pixels with a value of '1' (white).

Figure 5-6 : The image is processed by filling of regions and holes.

5.1.4 ROI code management

This process applies the ROI mask to the original image by comparison of the two images. The parts of the original image, which are identified by the ROI mask, will be transferred to the final image. On the other hand, the part of original image which is marked by the non-ROI mask will be replaced by '1' in the final image (i.e. those areas not of interest). Figure 5-7 shows the output ROI automatic detection image; ocean areas are non-ROI and the island regions have been identified as ROI. Some clouds in the left-hand side of the image have also been included in the ROI. In this example the proposed ROI automatic detection based on edge segmentation has reduced the size to be 57.50 % of the original image.

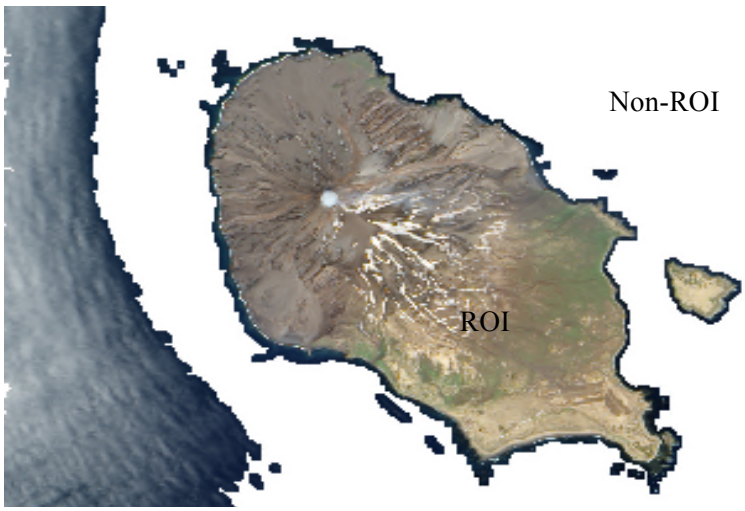
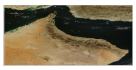


Figure 5-7 : The original image masked by ROI mask (ROI image).

Testing of the proposed ROI automatic detection based on edge segmentation has been conducted on another seven satellite images as shown in Table 5-1. The proposed ROI automatic detection based on edge segmentation has identified the ROI of these images by edge detection and morphology processes. The results show that the sizes of the tested images are reduced on average to 66.70 % of the original images. The various sizes and areas of the non-ROI leads to different image size reductions, for instance image-3 with a large amount of ocean in the image is reduced to 35.04 % of its original size.

Table 5-1 : The proposed edge segmentation ROI automatic detection experimental results.

Input image	Proposed edge segmentation		
	Input size	ROI image size	(ROI/Input) x100
1. 	3,207 KB	2,415 KB	75.30 %
2. 	3,988 KB	2,293 KB	57.50 %
3. 	10,184 KB	3,568 KB	35.04 %
4. 	3,381 KB	2,008 KB	59.40 %
5. 	13,426 KB	9,390 KB	69.94 %
6. 	12,685 KB	1,1847 KB	93.40 %
7. 	1,678 KB	1,281 KB	76.34 %

5.2 Histogram segmentation

The pixel values in an image can be represented by intensity histograms. The concept of histogram segmentation is classification of an image using intensity. Different regions are identified by assuming that different intensities can be matched to different regions (Kurugollu, Sankur et al. 2001; Agus and Asano 2004; Brancati, Frucci et al. 2008). The flow chart in Figure 5-8 illustrates the proposed ROI automatic detection based on histogram image segmentation algorithm. There are five sub-processes in the proposed system.

- Histogram classification
- Histogram segmentation
- Black and white image conversion
- Application of morphological techniques
- ROI code management

The process of histogram segmentation has been described for each sub-process as detailed below:

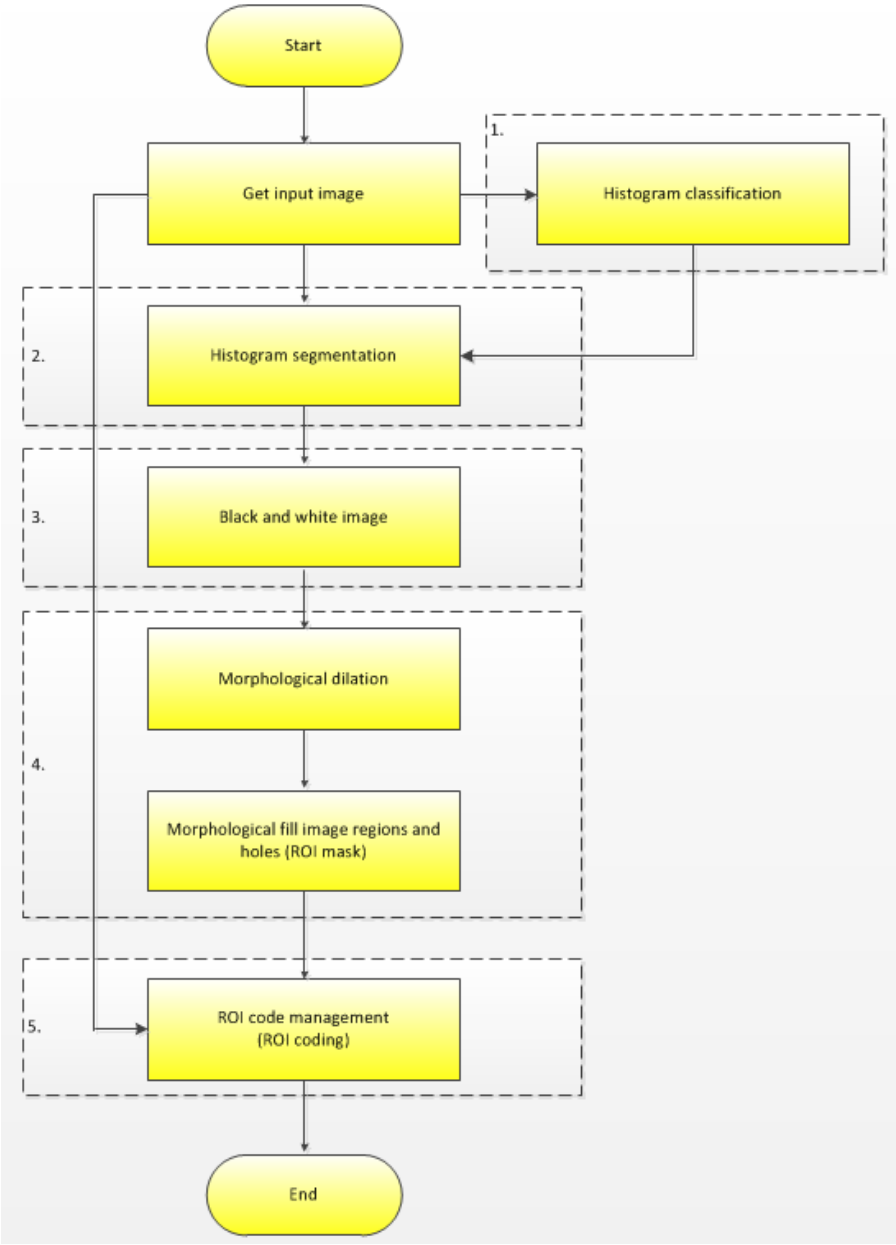


Figure 5-8 : The flow chart of ROI automatic detection based on histogram segmentation.

Figure 5-9 shows the example input which is a satellite image of the port city of Sevastopol in Ukraine. The image was acquired on August 2009 and taken by a Nikon D3 digital camera by the ISS Crew (Przyborski 2010).

Figure 5-9 : The input satellite image for ROI automatic detection based on histogram segmentation.

5.2.1 Histogram classification

The histogram segmentation method is based on threshold classification. The objects in the image are separated into the different groups according to the intensity of the pixel. There are a number of different techniques to identify the clusters corresponding to regions of interest. Supervised classification techniques which rely on labelled reference samples have been applied to classify the satellite image (Bruzzone and Persello 2009). In addition active learning, neural networks and support vector machines have all successfully been used to classify imagery (Tuia, Volpi et al. 2011; Xiuming and Jing 2011; Crawford, Tuia et al. 2013). This research classifies the object by labelling four groups; ocean, urban, forest and agriculture using referenced samples. Therefore, at the end of the algorithm, objects classified will be determined as belonging to the ROI or the non-ROI areas of the image depending which groups are considered of interest.

The process to classify the objects in the image to different groups has been developed as detailed in the following:

For the initial design the threshold is calculated by the random selection of pixels for each object and the identification of the appropriate group by human intervention. This sampling information is analysed for statistical features such as the maximum, minimum and average values. The threshold is set based on these statistical features of each group for the example pixels, and the test data recorded is shown in Table 5-2.

Table 5-2 : Sample test data.

Sample number	Pixel value			
	Ocean	Forest	Agriculture	Urban
1	52	76	120	213
2	52	74	132	183
3	44	79	133	196
4	48	75	150	190
5	58	76	149	179
6	48	101	158	187
7	54	94	133	166
8	43	70	139	182
9	44	89	149	180
10	40	76	162	189
11	45	80	160	234
12	41	78	167	195
13	39	76	158	173
14	37	75	145	182
15	31	74	151	180
16	41	77	150	190
17	48	73	123	187
18	50	75	146	191
19	49	75	141	163
20	53	78	157	196
Average	45.85	78.55	146.15	187.8
Max	58	101	167	234
Min	31	70	120	163

Secondly, this segmentation process is based on the image grey level histogram as shown in Figure 5-10. The statistical values from the first process are applied to the histogram image to compare the intensity range of each object. After obtaining the minimum and maximum intensity of each object, these thresholds (histogram thresholding set) are applied to the histogram segmentation (next process). For example the values of pixels of the ocean are between 31 to 58 and from the test the average pixel value of the ocean is 45.85. If pixels are classified as forest they have values between 70 to 101 and the average pixel value is 78.55. Therefore, some pixel values are not classified in to any group. Those pixels that are not classified as any group are identified by the white colour in the histogram segmented image. The assumption is that the objects are characterised uniquely by their brightness. In practice this will not be the case but it provides a starting point for the analysis.

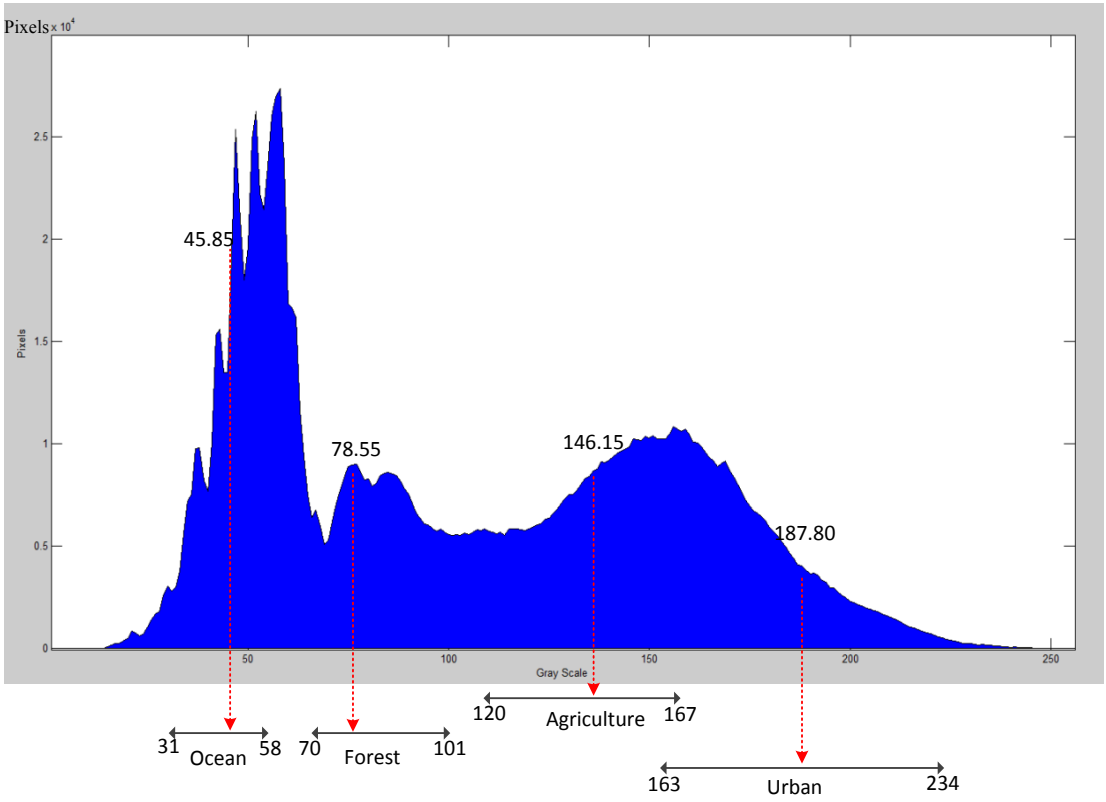


Figure 5-10 : Histogram of grey scale test image and object statistical numbers.

The process of thresholding classification classifies the object with manual thresholding, i.e. the thresholding is done by manual intervention. However for the real cubesat design, the payload designer has to find the thresholds for the camera and then install these on the module. After the cubesat has been launched, image calibration may be required to adjust the correct histogram thresholds.

5.2.2 Histogram segmentation

Every pixel of the original image is compared with the histogram thresholds set and allocated to a group if possible. The pixels belonging to each group are represented by different colours in Figure 5-11 showing the output image after processing by histogram segmentation. The pixels are classified into four categories; ocean, forest, agriculture, and urban, which are represented by blue, green, yellow and red respectively. The pixels not classified in any group will be represented by white colour in the histogram segmented image.

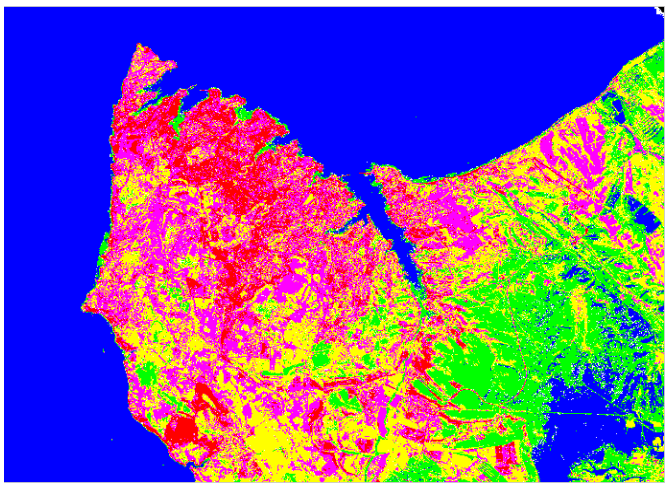


Figure 5-11 : Image processed by histogram segmentation.

5.2.3 Black and white image

This process converts the histogram segmented image to a binary image. The image has its pixels replaced by either a black or white colour depending on whether the pixel is in a classified category in a region of interest. The input image pixels are replaced with the value '1' (white) if the input pixels are in land areas, i.e. forest, urban and agriculture which are defined as the region of interest, whereas other pixels are replaced with the value '0' (black) if they are categorised as the sea. Figure 5-12 shows the conversion of the histogram segmented image, to a black and white image. In this example using the manual histogram classification, there are some errors. In particular the area in the bottom right that is classified as sea is forest to the East of Sevastopol. Both the sea and forest are categorized by low reflectance. The overlap between groups produces some error in the classification of the groups.

Figure 5-12 : Black and white image.

5.2.4 Mathematical morphology

This process applies dilation morphology and fills the holes of the image to get the ROI mask. The description of this process is the same as detailed in section 5.1.3. Figure 5-13 shows the ROI mask image for this test.

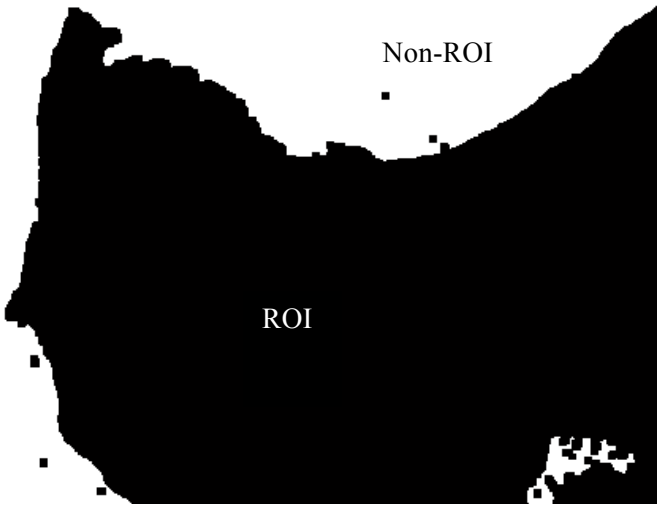


Figure 5-13 : ROI mask of the histogram segmentation test.

5.2.5 ROI code management

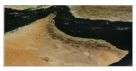
The final step applies the ROI mask to the original test image. This process is the same process as described in section 5.1.4. The ROI image detected, based on histogram segmentation, is illustrated in Figure 5-14.

Figure 5-14 : The ROI image of histogram segmentation test.

Table 5-3 shows the result of the ROI automatic detection based on histogram segmentation for a set of satellite images. The histogram segmentation has reduced the image size to an average of 73.39 % of the original image although it should be noted that this value is determined by the percentage of sea in the scene, i.e. the definition of the region of interest. The major drawback to the proposed system which is based on histogram thresholding approaches is that they often lack the sensitivity and specificity needed for accurate group thresholding (Dai and Zhang 2011). This image segmentation technique requires knowledge of the characteristics of the objects in the image so that the correct thresholds can be applied (Al-amri, Kalyankar et al. 2010). There are more test results for the ROI detection based on histogram segmentation in Appendix D.

Table 5-3 : The proposed histogram segmentation ROI detection experimental results.

Input Image	Proposed histogram segmentation		
	Input size	ROI image size	(ROI/Input) x100

1. 	3,207 KB	2,446 KB	76.27 %
2. 	3,988 KB	2,104 KB	52.76 %
3. 	10,184 KB	6,285 KB	61.71 %
4. 	3,381 KB	2,382 KB	70.45 %
5. 	13,426 KB	9,957 KB	74.16 %
6. 	12,685 KB	12,384 KB	97.63 %
7. 	1,678 KB	1,355 KB	80.78 %

5.3 Texture segmentation

Texture segmentation identifies groups of the object based on their texture features. The proposed ROI automatic detection based on texture segmentation consists of six sub-processes as shown in Figure 5-15. The texture segmentation technique has been implemented to segment the example satellite image.

The Grey Level Co-occurrence Matrix (GLCM) is widely used for texture measurement in image analysis (Haralick, Shanmugam et al. 1973; MathWork 2012). The textural features are based on statistics, which summarise the relative frequency distribution. GLCM describes how often one grey tone will appear in a specified spatial relationship to another grey tone in the image at a certain distance and direction (Shanmugan, Narayanan et al. 1981; Soh and Tsatsoulis 1999; Bartels, Hong et al. 2005). Haralick et al. (Haralick, Shanmugam et al. 1973) first introduced the use of GLCM for extracting various texture features. The common statistics applied in texture analysis are energy, entropy, contrast, variance, homogeneity and correlation etc., (Shanmugan, Narayanan et al. 1981).

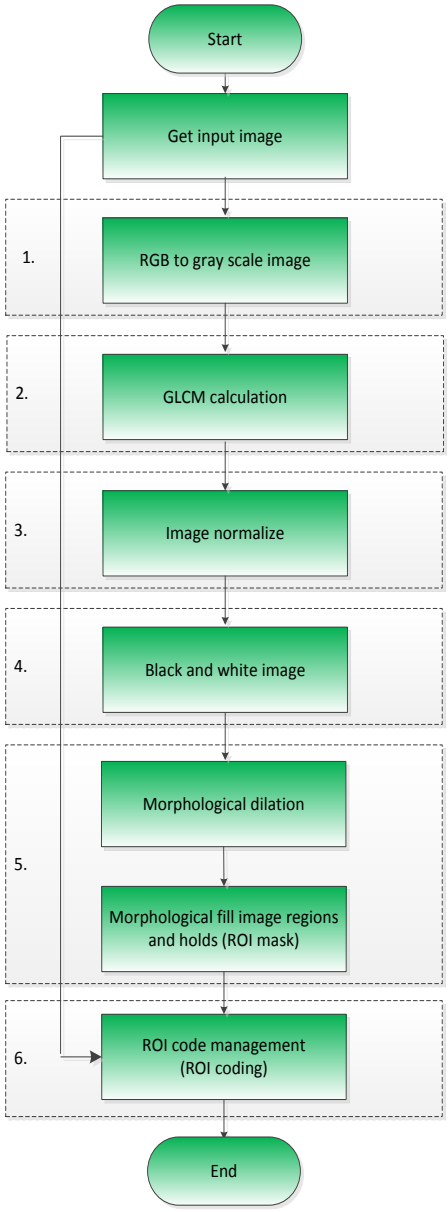


Figure 5-15 : The flow chart of ROI automatic detection based on texture segmentation.

The steps of this segmentation used to obtain the ROI are:

- Convert the RGB image into a grey scale image
- Calculate the GLCM matrix. The analysis used a window size of 9 pixels, orientation 0 degrees and distance 1 pixel, which is the typical parameter used for image processing (Gonzalez and Woods 2007).
- Calculate the statistics for each pixel, which consists of Energy, Contrast, Correlation, Homogeneity and Entropy. Normalise the matrix image with the statistical features
- Convert the image to a black and white image

- Obtain the ROI mask by using grey thresholding and morphological techniques
- Apply the ROI mask to the input image

Figure 5-16 shows the example input, which is a satellite image of Rogers City on lake Huron; one of the Great lakes of North America. It also shows the largest carbonate mine in the world, Calcite Quarry. The image was acquired on May 2006 by the ISS digital camera. The details of sub-processes are described in the following:

Figure 5-16 : Input image for the proposed ROI automatic detection based on the texture segmentation.

5.3.1 Grey scale image

This step converts the RGB input satellite image to a grey scale image. This process is the same process as described in section 5.1.1.

5.3.2 GLCM calculation

The GLCM has been used to determine texture similarity of groups of pixels in a region. Texture segmentation techniques generally consist of two stages. The first stage finds the local structure of an image and converts each pixel in the image into a vector of these local properties. Secondly the clustering method groups pixels which have similar properties together (Ershad 2011).

The GLCM can be defined by $P_{(i,j),}$ where *'i'* is a grey level and the other grey level is *'j'*. The GLCM is constructed by calculating how frequently the intensity level *'i'* occurs adjacent to a pixel with intensity level *'j'*. In other words $P_{(i,j)}$ is the probability of occurrence of the *(i,j)* relationship occurring in the image (Ershad 2011). There are many texture features that can be extracted from GLCM (Haifang, Lihuan et al. 2008). In this research the focus is on the five statistics defined below in equations (5.5 – 5.9);

$$Energy = \sum_i \sum_j P^2(i,j) \tag{5.5}$$

$$Contrast = \sum_i \sum_j (i-j)^2 P(i,j) \tag{5.6}$$

$$Correlation = \sum_i \sum_j \frac{(i,j)\ x\ P(i,j) - \left(u_x * u_y\right)}{\sigma_x * \sigma_y} \tag{5.7}$$

$$Homogeneity = \sum_i \sum_j \frac{P(i,j)}{1 + |i-j|} \tag{5.8}$$

$$Entropy = \sum_i \sum_j P(i,j) \log P(i,j) \tag{5.9}$$

Where σ_x and σ_y are the standard deviations of the rows and columns of $P_{(i,j)}$ respectively. Also μ_x ,μ_y are the means of the rows and columns of $P_{(i,j)}$ respectively. The features extracted from the GLCM are used to classify the contents of an image. Energy describes texture uniformity (Baraldi and Parmiggiani 1995). Contrast shows the change between the highest and the smallest values of the nearby pixels in the considered group. Correlation measures the uniformity of the distribution of the matrix, whereas homogeneity measures the similarity of grey scale levels across the image. Finally entropy gives a measure of disorganisation of the image (Dipti and Mridula 2011). In this step the GLCM is calculated for these five statistical features.

5.3.3 GLCM features

Figures 5-17 to 5-21 illustrate the result after the image is processed by the texture feature of energy, contrast, correlation, homogeneity and entropy respectively.

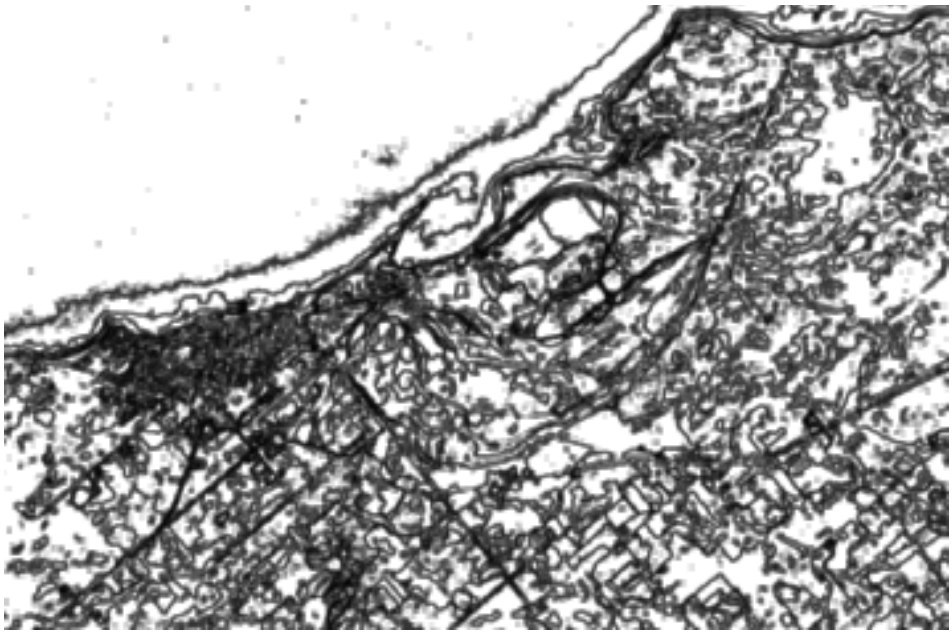


Figure 5-17 : GLCM energy.

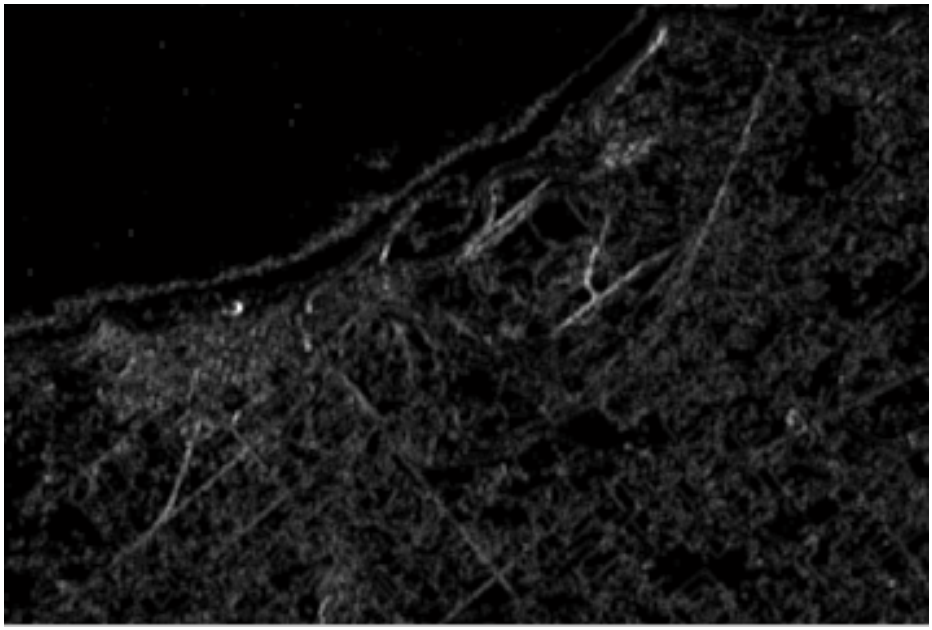


Figure 5-18 : GLCM contrast.

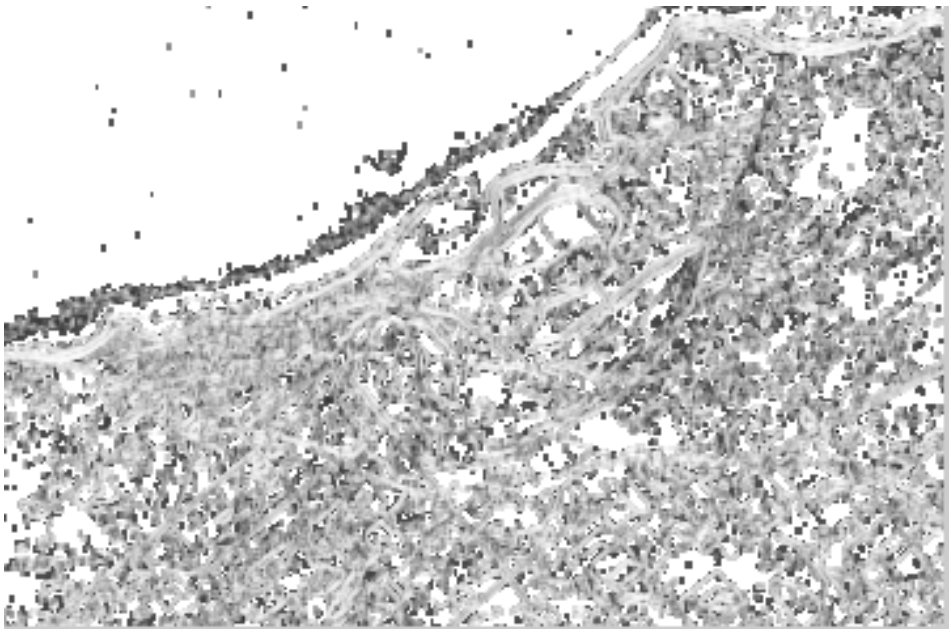


Figure 5-19 : GLCM correlation.

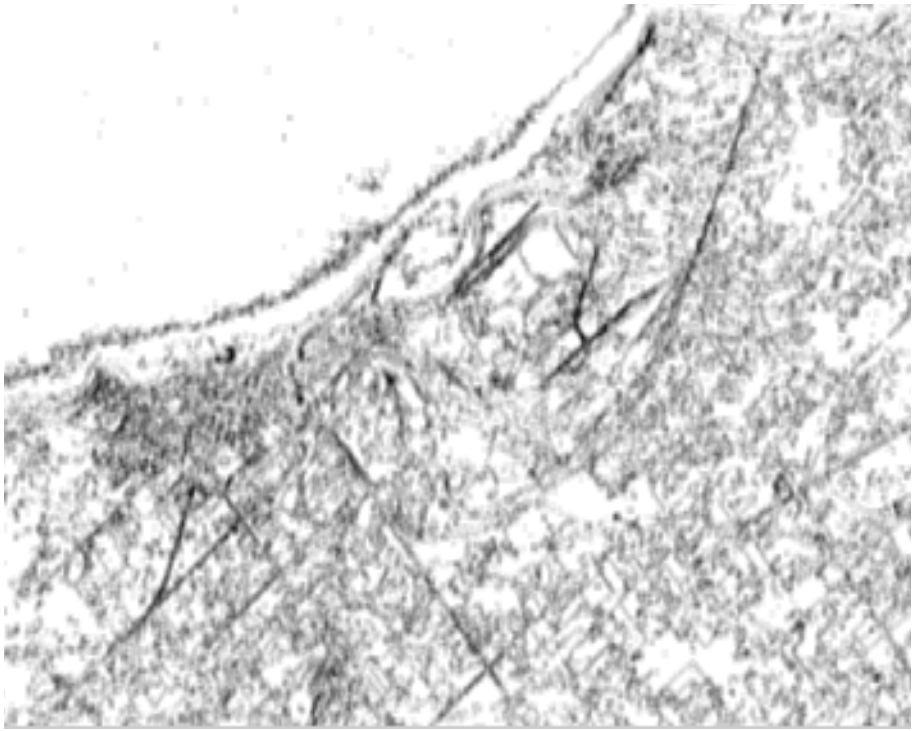


Figure 5-20: GLCM homogeneity.

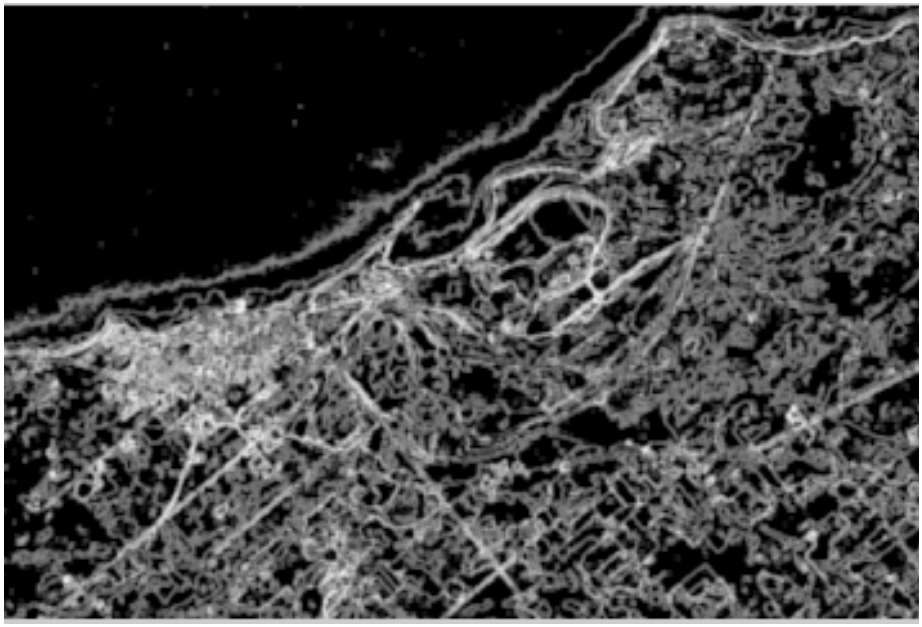


Figure 5-21 : GLCM entropy.

5.3.4 Black and white image

This process converts the output image from the previous step to a black and white image for the next process; the morphological technique. The grey scale image is replaced by black or white pixels depending on the pixel luminance level. This process used the level value of 0.5 to be the threshold Matlab programming default (MathWorks 2012). The normalised image pixels are replaced with the value '1' (white) if the input pixels are greater than this level. Pixels below this level is replaced the value '0' (black). The black and white image, which is converted from the energy image, due to the energy image shows areas that have uniformity and non-uniformity. In this example this non-uniform area is defined as the ROI area. Figure 5-22 shows the black and white image from this texture segmentation test.

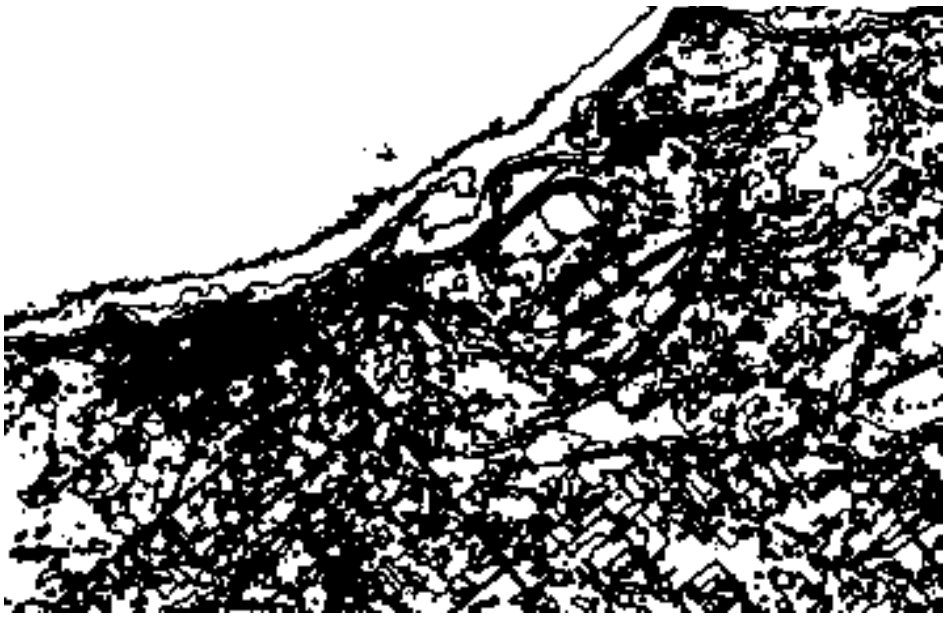


Figure 5-22 : Black and white image.

5.3.5 Mathematical morphology

This process consists of morphological dilation and filling of holes, which is the same process as described in section 5.1.3 Mathematical morphology. Figure 5-23 shows the image processed by morphology dilation. Baraldi and Parmiggiani (Baraldi and Parmiggiani 1995) found "*energy and contrast, were the most efficient parameters for the discrimination of different textural patterns*" in an image. The energy measures textural uniformity and is high when the grey level distribution is constant or has a periodic form. Energy detection in the image shows that it is possible to distinguish the ROI from the sea background. This research uses energy detection in the image to find the ROI. Figure 5-24 shows the ROI mask, which is the result of processing the image shown in Figure 5-23 by filling regions and holes.

Figure 5-23 : The result after morphological dilation.

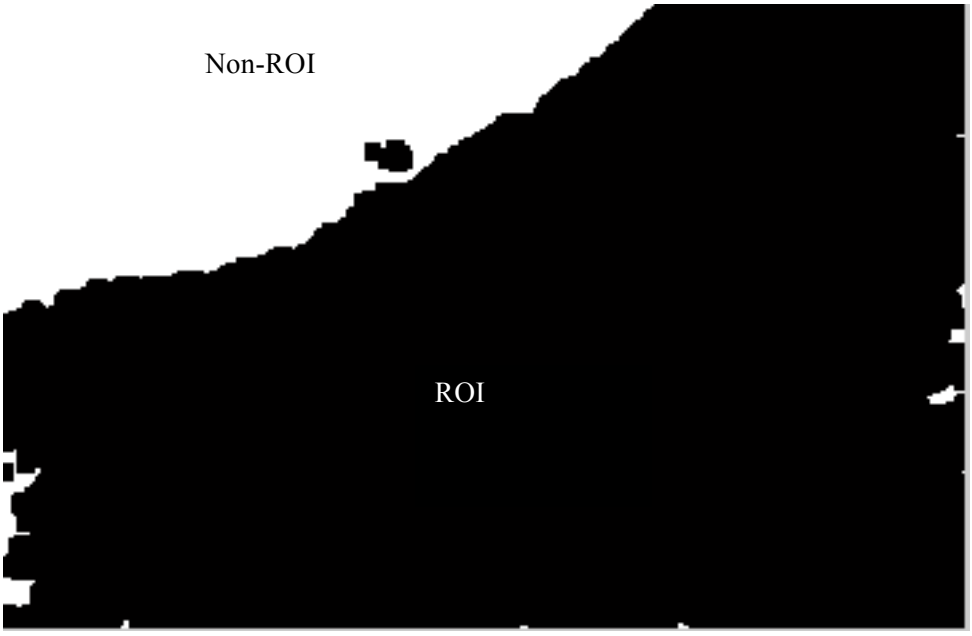


Figure 5-24 : The ROI mask after filling regions and hole processing.

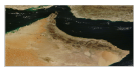
5.3.6 ROI code management

This process is the same as detailed in section 5.1.4 ROI code management. Figure 5-25 is the output image after application of the ROI mask from the original image. The texture segmentation technique has identified the land and the coastal region characterised by distinctive texture as being the ROI.

Figure 5-25 : The ROI image.

Table 5-4 demonstrates the results of the proposed ROI automatic detection based on texture segmentation applied to seven satellite images. The results show that the system has identified the ROI and consequently reduced the original image size. The histogram segmentation has reduced the image size to an average of 73.00 % of the original image. The average time taken for processing is comparatively high however (904.27 seconds, test performed on Intel core 2 Duo CPU 3.16 GHz and RAM 4.00 GB) due to the time spent on the GLCM calculation process. This method therefore is time consuming, and this could potentially affect the power consumption (Haralick, Shanmugam et al. 1973; Baraldi and Parmiggiani 1995; de O. Bastos, Liatsis et al. 2008). Power consumption and memory usage for the real hardware implementation of the proposed system will be discussed in chapter 7. There is more information on the ROI automatic detection based on texture segmentation test results in Appendix E.

Table 5-4 : The proposed texture segmentation ROI automatic detection experimental results.

Input image	Proposed texture segmentation		
	Input size	ROI image size	(ROI/Input) x100
1. 	3,207 KB	2,745 KB	85.59 %
2. 	3,988 KB	2,862 KB	71.77 %
3. 	10,184 KB	3,765 KB	36.97 %
4. 	3,381 KB	2,145 KB	63.44 %
5. 	13,426 KB	10,934 KB	81.44 %
6. 	12,685 KB	11,674 KB	92.03 %
7. 	1,678 KB	1,338 KB	79.74 %

5.4 Segmentation of land areas

ROI automatic detection based on Histogram segmentation for specific object

The region of interest of the satellite image depends on the Earth observation application fields. Some applications are concerned with identifying one object of interest. This section presents the process to detect the ROI of the image, which consists of one specific feature using the proposed histogram segmentation technique. There are three tests that have been performed, to detect forest, urban and agriculture.

The proposed system to detect the specific object consists of 5 steps as shown in Figure 5-26. These are RGB conversion, histogram segmentation, isolate small area, morphological dilation and fill region and hole and ROI management.

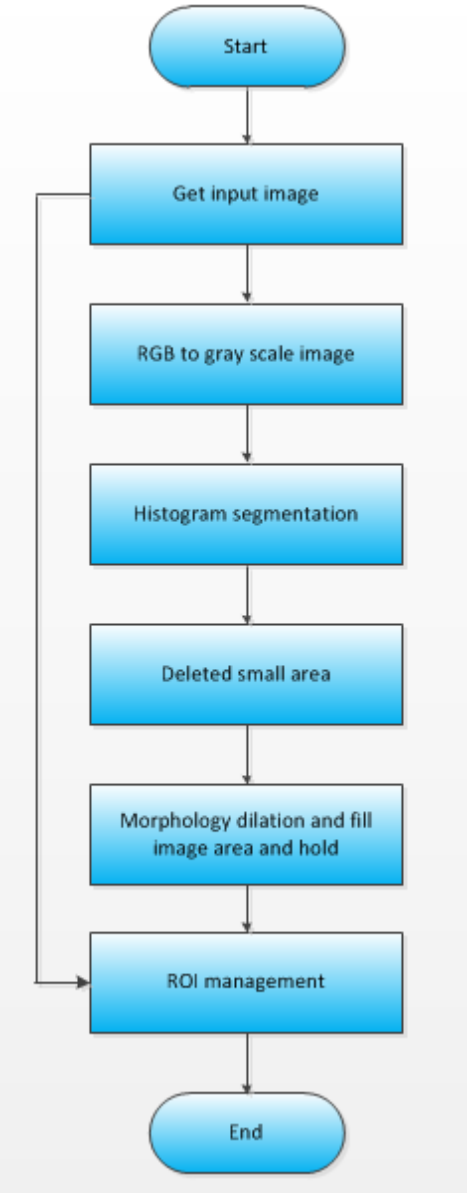


Figure 5-26 ROI automatic detection based on histogram segmentation.

Figure 5-27 shows the input image, which is located in Burghause, Germany. It was taken by RapidEye with 5 meter resolution (Romeijn, 2011). Figure 5-28 shows the histogram of the image grey scale with the sampling set of each object with the minimum, maximum and average value of the sampling set.

Figure 5-27 : Input image for the test

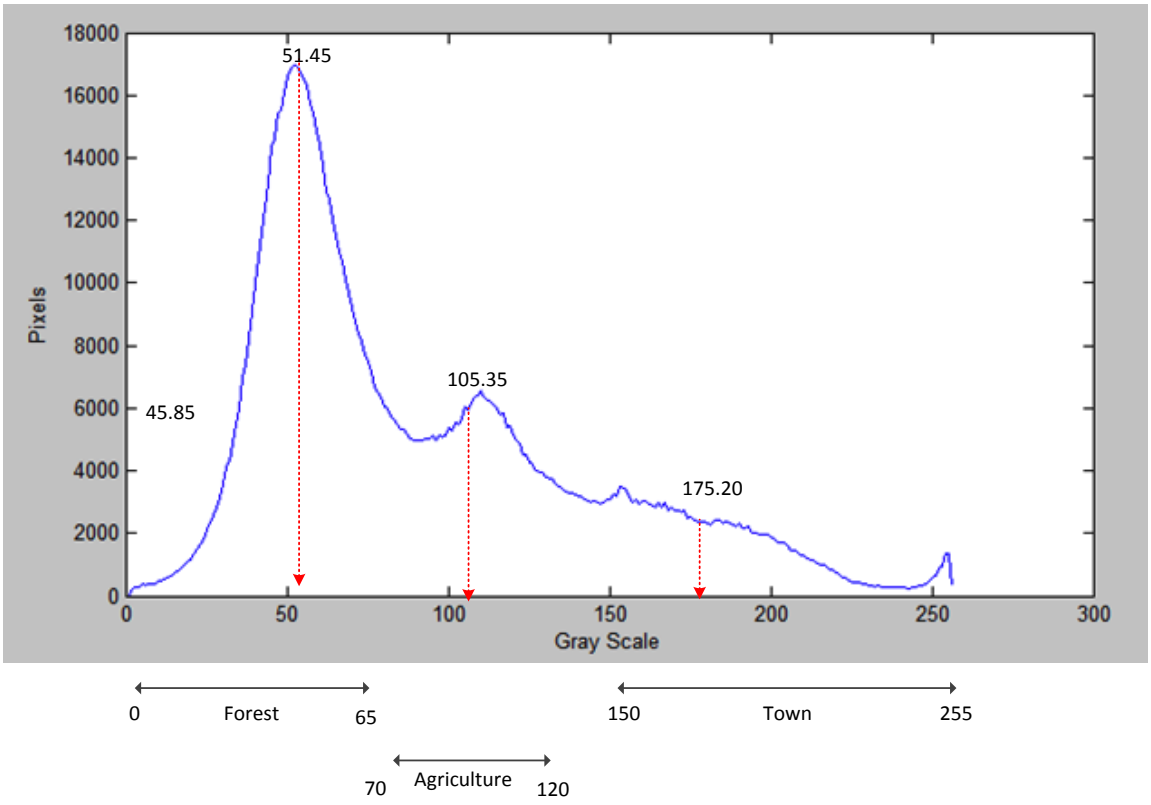


Figure 5-28 : Grey scale histogram image

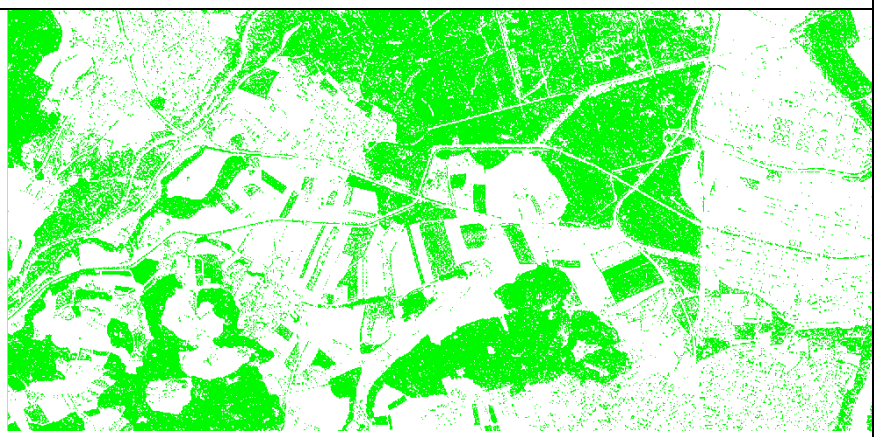
Table 5-5 shows the result of each process to detect the forest in the image. First the process starts with converting the image from the RGB image to a grey scale image. The second process applies the histogram threshold of the forest to the grey scale image and the pixels that are in the range of the forest are labelled green as shown in Table 5-5 'Histogram segmented image'. The third process is to delete small classified points to reduce noise and to identify the areas of interest. The next process is morphological dilation to connect close small areas in the same are of interest together. The application of the morphological fill of regions and holes, gives the ROI mask at the end of this process. The final step is application of the ROI mask to the original image to produce the ROI image.

Table 5-6 and 5-7 show the images that result from each step required to detect the town and agriculture in the image respectively.

Table 5-8 shows the size of the ROI images, which are detected using the proposed system on a specific object of interest. The ROI images for forest, town and agriculture have been found to be 48.47 %, 40.156 % and 26.89 % of the original image respectively.

The test result shows that the ROI automatic detection based on histogram segmentation is possible to detect the specific object in the image as the ROI object.

Table 5-5 : The result of the proposed system of each process to detect only forest in the sample image

Step	Result of each step
1. Histogram segmentation	

2. Delete small area	
3. Dilation	
4. ROI mask	

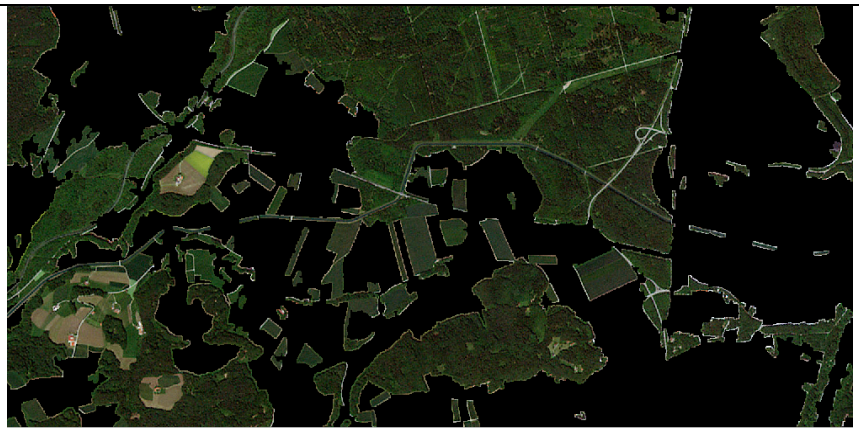
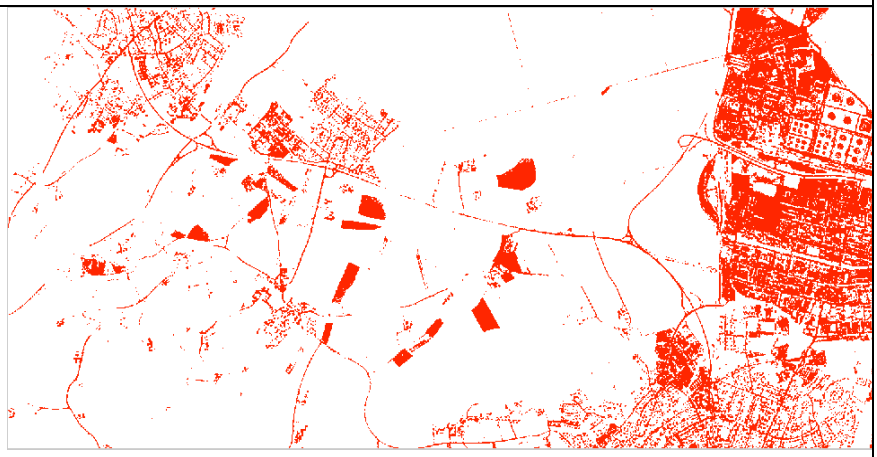
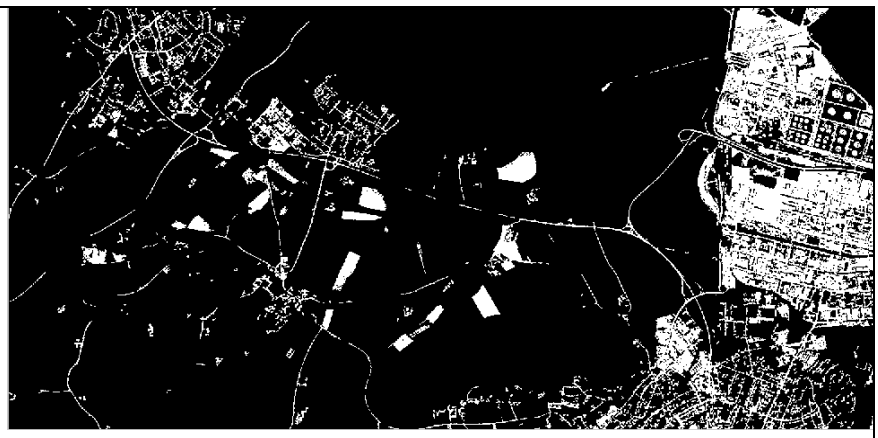
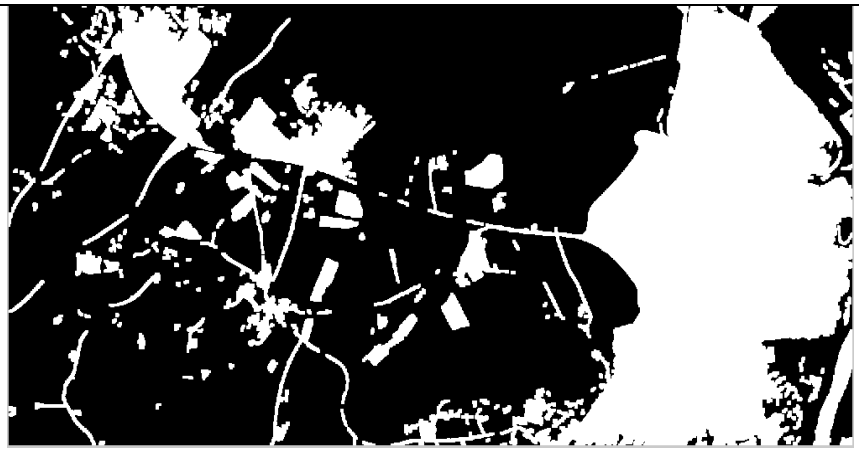
5. ROI image	

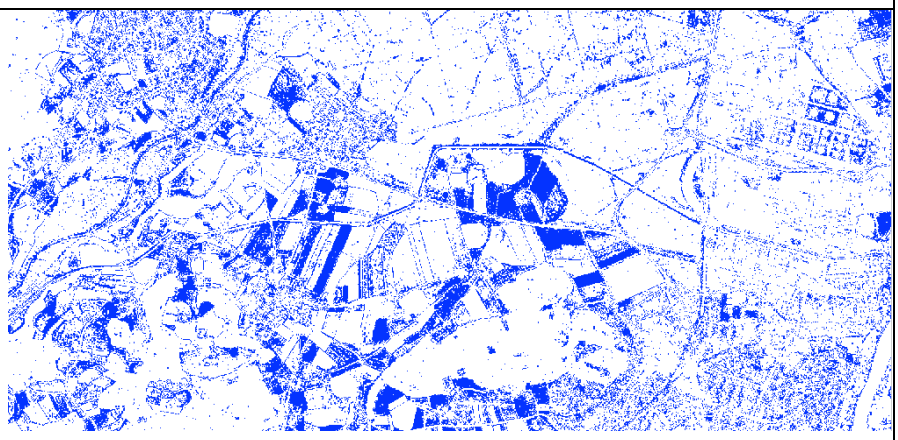
Table 5-6 : The result of the proposed system of each process to detect only towns in the sample image

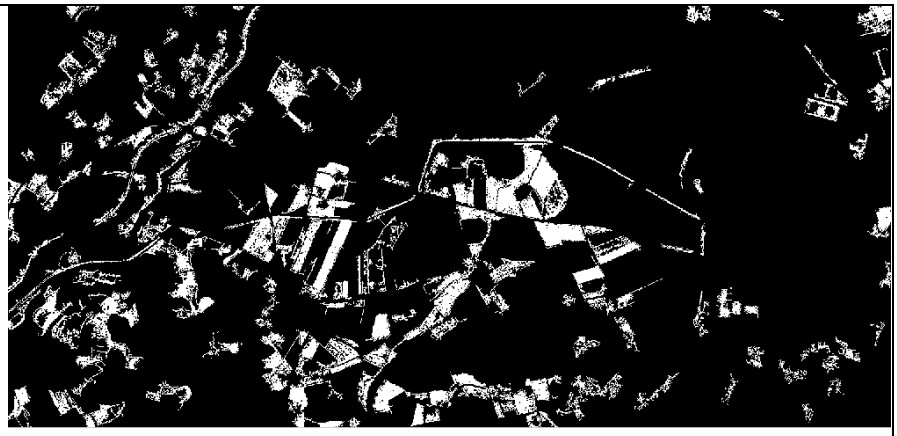
Step	Result of each step
1. Histogram segmentation	

2. Delete small area	
3. Dilation	
4. ROI mask	

5. ROI image	

Table 5-7 : The result of the proposed system of each process to detect only agriculture in the sample image

Step	Result of each step
1. Histogram segmentation	

2. Delete small area	
3. Dilation	
4. ROI mask	

5. ROI image	

Table 5-8 : The size of the ROI image by using the proposed histogram segmentation to detect ROI specific objects

ROI detection object	ROI automatic detection based on histogram segmentation for specific objects		
	.ROI process		
	Input size : A	ROI image size	Reduced size
1. Forest	2,886 KB	1,399 KB	48.47 %
2. Towns	2,886 KB	1,159 KB	40.16 %
3. Agriculture	2,886 KB	776 KB	26.89 %

Figure 5-29 shows the histogram segmented image. There are 3 classified areas; forest, towns and agriculture, represented by green, red and blue colours respectively. The pixels which are not classified as any object are shown in white.

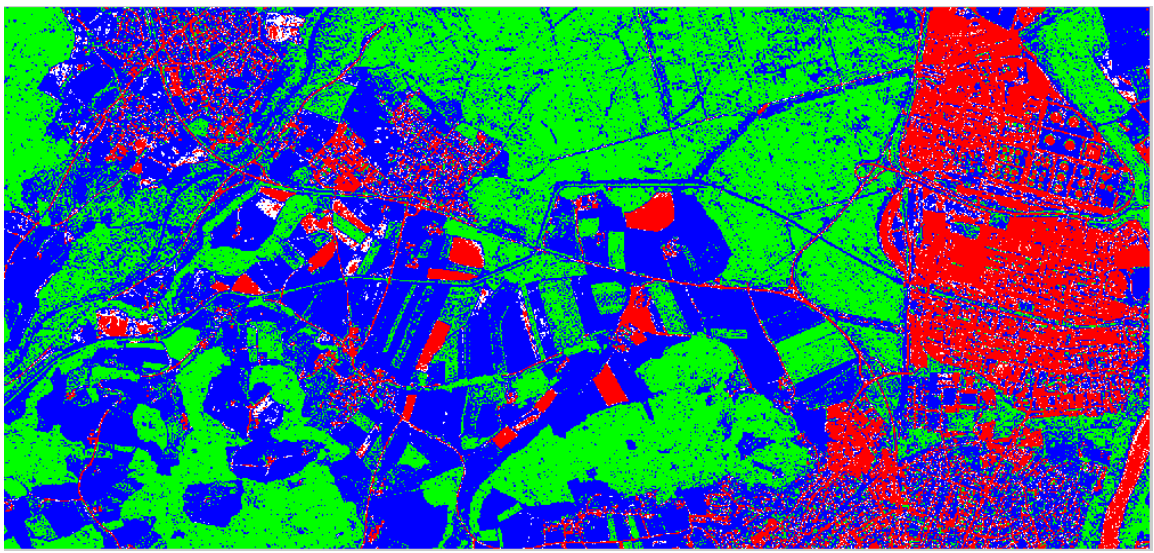


Figure 5-29 : Histogram segmented image

5.5 System accuracy

The accuracy of the proposed system has been examined by comparing the ROI image identified by the proposed system with that which would be obtained by manual detection.

To test the system accuracy the following steps have been conducted. An input image is selected, e.g. Figure 5-30 and then both automatic ROI detection using the edge segmentation based method and manual ROI detection have been performed. This test used the edge automatic detection method because it is fully automatic and requires less computational time than alternative techniques. The manual detection is performed by drawing the outline of the area of interest by eye, as shown in Figure 5-31. The ROI mask operation is then computed automatically and also manually. The resulting ROI masks are shown in Figure 5-32 and Figure 5-33 for automatic and manual detection respectively. The resulting ROI images are shown in Figure 5-34 and Figure 5-35 for automatic and manual detection respectively. Finally, the ROI automatic detection image and the ROI manual image are compared to find the number of different pixels. Figure 5-36 shows the different pixel values by brown colour. For this image test, there are 35,939 different pixels and the percentage difference between the automatic and manual ROI methods is 2.31 %.

Figure 5-30 : Input image for system accuracy test.

Figure 5-31 : ROI outline manual drawing.

Figure 5-32 : ROI mask from the proposed system.

Figure 5-33 : ROI mask from manual detection.

Figure 5-34 : ROI image by automatic detection.

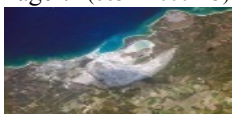
Figure 5-35 : ROI image by manual detection.

Figure 5-36 : The image showing the differences between the ROI automatic and ROI manual detected images.

More test results concerning the accuracy of ROI detection are shown in Table 5-9. Seven satellite images were used. For each image the outline of the region of interest was made three separate times giving a total of 21 tests for the seven images. The percentage difference between the proposed automatic ROI detection and manual ROI detection on average is 7.11 %. The higher values obtained for Images 2 and 6 are due to the greater uncertainty in defining the region of interest by eye at least partly due to the complex structure observed in the sea. If the region of interest had been different, e.g. regions close to the coast, the percentage difference would have been correspondingly higher and the accuracy lower.

Table 5-9 : The comparison of results between ROI automatic detection and ROI manual detection.

Image and test number		Number of different pixels	Percentage difference	Average percentage
Image 1 (1080 x 1440 x 3) 	Test 1	35,939	2.31 %	2.24 %
	Test 2	32,271	2.08 %	
	Test 3	36,375	2.34 %	
Image 2 (1189 x 1784 x 3) 	Test 1	376,935	17.77 %	17.76 %
	Test 2	377,396	17.79 %	
	Test 3	375,705	17.71 %	

Image and test number		Number of different pixels	Percentage difference	Average percentage
Image 3 (2200 x 3300 x 3) 	Test 1	490,470	6.76 %	6.72 %
	Test 2	485,433	6.69 %	
	Test 3	487,256	6.71 %	
Image 4 (977 x 1996 x 3) 	Test 1	87,155	4.47 %	4.15 %
	Test 2	78,589	4.03 %	
	Test 3	77,111	3.95 %	
Image 5 (3367 x 2245 x 3) 	Test 1	197,469	2.61 %	2.21 %
	Test 2	178,301	2.36 %	
	Test 3	126,230	1.67 %	
Image 6 (2732 x 2732 x 3) 	Test 1	1,062,532	14.24 %	13.82 %
	Test 2	1,028,147	13.78 %	
	Test 3	1,004,036	13.45 %	
Image 7 (663 x 1000 x 3) 	Test 1	21,591	3.26 %	2.86 %
	Test 2	18,727	2.82%	
	Test 3	16,522	2.49%	
Average				7.11 %

5.6 Morphology structure element study

The ROI image segmentation based on edge, histogram and texture segmentation have common techniques in their processes; the morphology dilation and the morphology fill regions and holes. The structure element (SE) is the main parameter used in the morphological technique for image processing. Hence it is necessary to understand how the shape and size of SE affects the output image of the proposed ROI automatic detection system.

This research has investigated the shape and size of SE selection to find an appropriate method for the proposed system. The morphology SE has different shapes shown as the grey shadow and defined by the pixel value of '1' in Figure 5-37 (a)-(d); square, diamond, disk and line respectively (MathWork 2012).

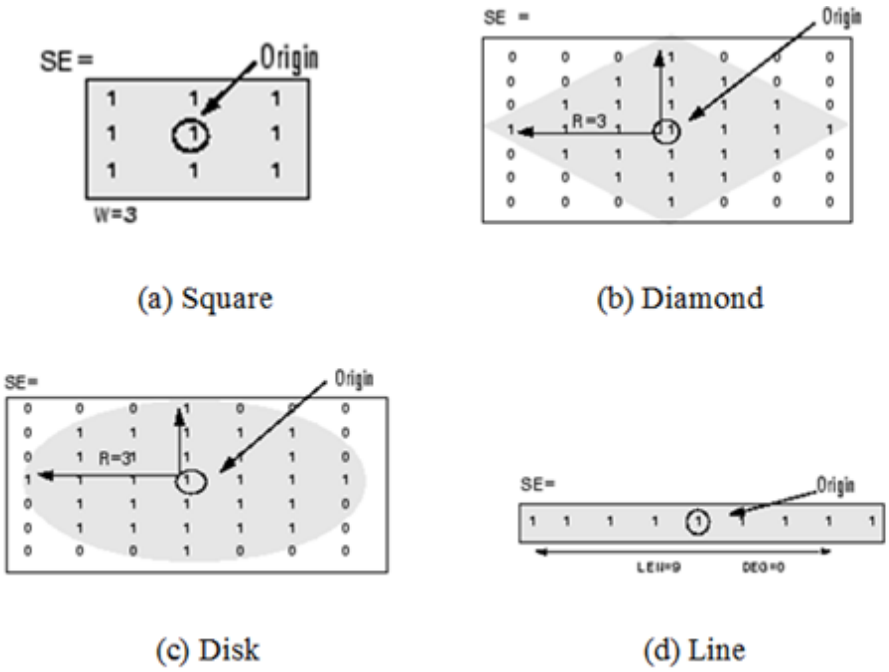


Figure 5-37 : Structure element shapes (MathWork 2012).

The different SE shapes have been tested to study the behaviour of the proposed ROI detection. Table 5-10 illustrates the test results of different shapes of SE changing the output images. The table shows the ROI mask, ROI image and the percentage of different. The results show the percentage of difference between the ROI automatic detection and ROI manual detection by using different SE. The ROI image from square, diamond and disk SE give similar results. However the line shape cannot detect some information in the ocean and the land that have large regions with similar pixel values. The results shows that the smallest percentage error between the ROI automatic detection and manual detection is from using square, diamond, disk and line.

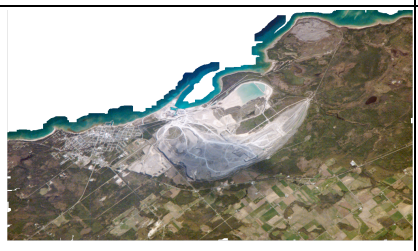
Table 5-10 : The percentage difference between the ROI automatic detection and ROI manual detection using different shapes of structure element.

SE shape	ROI mask	ROI image	Percentage difference
Square			10.27 %
Diamond			10.52 %
Disk			10.74 %
Line			11.48 %

In addition five different sizes of the SE have been applied using the proposed system to test the effect on the resultant image. There are five different sizes varying from size-1 up to size-5 (size-1 is the smallest size of each shape). The size of this test refers to the number of pixels in the structuring element. Size-1 is 5x5 pixels, whereas sizes-2 to -5 are 10x10, 15x15,

20x20 and 25x25 pixels respectively. Table 5-11 shows the ROI masks and ROI images which are produced by the proposed system based on edge segmentation using the square shape of five different sizes. The table also shows the difference between the ROI automatic detection and ROI manual method using SE of different sizes. The results show that the ROI mask using size 3 gives the smallest percentage of difference between ROI automatic detection and ROI manual.

Table 5-11 : ROI mask and ROI image by applying different sizes of square structure element.

SE size	ROI mask	ROI image	Percentage of difference
1 5x5			5.60 %
2 10x10			4.44 %
3 15x15			3.33 %
4 20x20			4.38 %

SE size	ROI mask	ROI image	Percentage of difference
5 25x25			4.54 %

The accuracy of the different types and sizes of SE has been quantified for all the structural elements considered. The test analyses the number of different pixels between the manual and the proposed automatic ROI detection. The assumption is that the lower the percentage of difference, the more accurate the ROI automatic detection (i.e. it is nearer to the manual ROI detection). This test has divided the SE into four different shapes with five different sizes of each, so in total there are 20 tests. Figure 5-38 shows the input image that was used in this test and Table 5-12 shows the number of different pixels between the ROI automatic detection and the ROI manual detection for different shapes and sizes of SE.

The results shows that the square structure element, at size-3, gives the lowest different number of pixels with the percentage of difference between the manual and automatic ROI detection at 1.37 %. This means for this experiment, the ROI automatic detection using the SE square shape with size-3 provides the result most similar to the ROI manual detection. It does not automatically follow that size-3 is the best method, just that it provides the closest fit to the manual interpretation. A larger structure element produces a greater percentage difference, especially with line shapes as shown in Figure 5-39. More results for this test are shown in Appendix F.

Figure 5-38 : Satellite image of the Balearic Islands for the system accuracy test.

Table 5-12 : Results for the comparison of ROI automatic detection and ROI manual detection with different shapes and sizes of SE.

Structure element shape	Number of different pixels				
	Structure element size-1	Structure element size-2	Structure element size-3	Structure element size-4	Structure element size-5
Square	39,770	26,310	21,297	27,169	40,598
Diamond	28,774	22,164	27,725	48,154	72,192
Disk	28,774	22,503	32,848	58,766	91,303
Line	42,779	25,042	87,961	206,501	359,097
Structure element shape	Percentage of difference (%)				
	Structure element size-1	Structure element size-2	Structure element size-3	Structure element size-4	Structure element size-5
Square	2.56 %	1.69 %	1.37 %	1.75 %	2.61 %
Diamond	1.85 %	1.43 %	1.78 %	3.10 %	4.64 %
Disk	1.85 %	1.45 %	2.11 %	3.78 %	5.87 %
Line	2.75 %	1.61 %	5.66 %	13.27 %	23.09 %

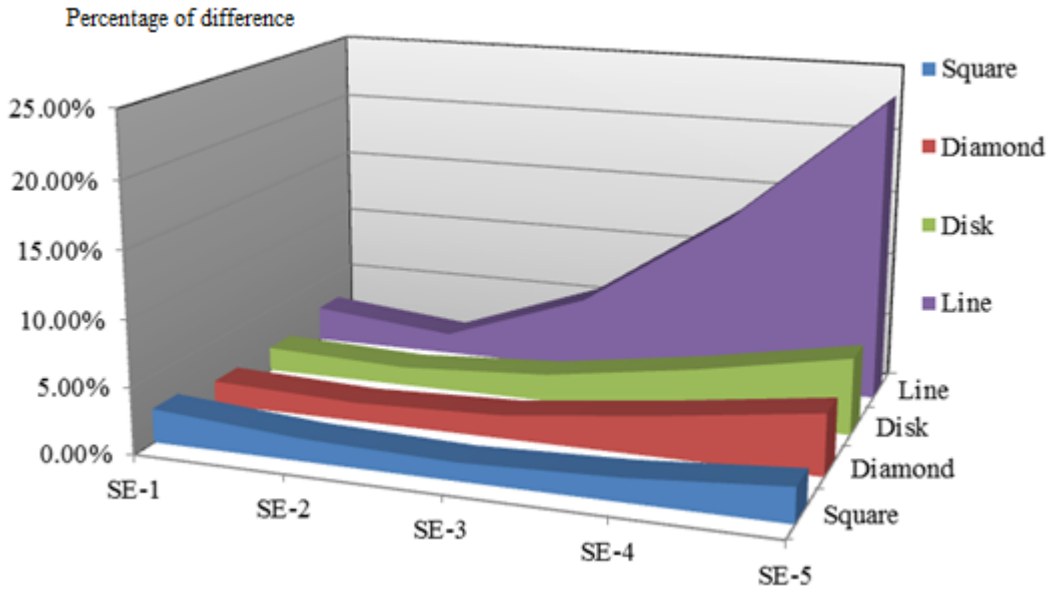


Figure 5-39 : The graph displaying percentage of difference between ROI automatic and ROI manual detection applying different SE shapes and sizes.

It can be seen that the size and shape of the SE impact on the size of the ROI image. Table 5-13 shows the test results of the image (Figure 5-38) size from the ROI automatic detection using different shape and sizes of SE to detect the ROI image. Figure 5-40 shows the result of the relationship between the ROI image size and SE shapes and sizes. In the graph, the SE line shape gives a larger size of ROI image compared to other SE shapes. The SE square shape gives the smallest size of the ROI image. This is because the square structure element follows the region of interest more closely that the other structure elements particularly if the structure element is large.

Table 5-13: The ROI automatic detection size for the different shapes and sizes of SE.

Structure element shape	ROI image size				
	Structure element size-1	Structure element size-2	Structure element size-3	Structure element size-4	Structure element size-5
Square	1,297 KB	1,343 KB	1,391 KB	1,436 KB	1,479 KB
Diamond	1,332 KB	1,376 KB	1,442 KB	1,502 KB	1,560 KB
Disk	1,332 KB	1,374 KB	1,457 KB	1,525 KB	1,600 KB
Line	1,394 KB	1,301 KB	1,607 KB	1,867 KB	2,178 KB

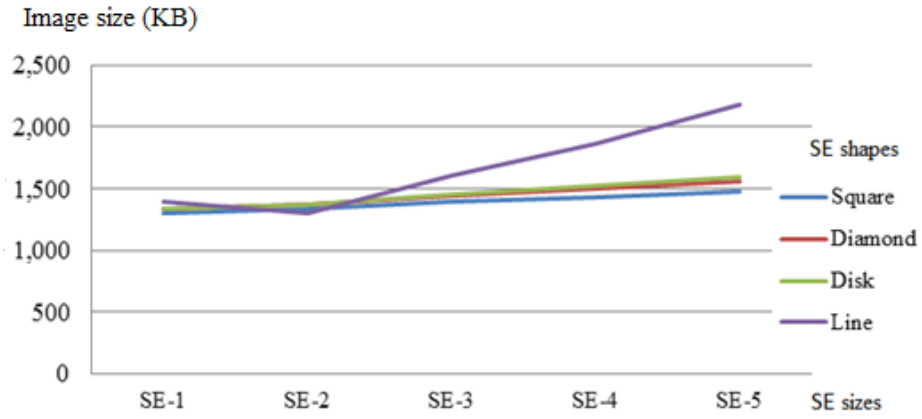
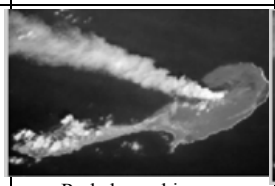
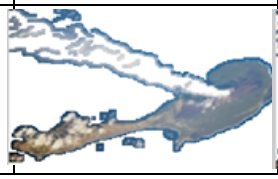
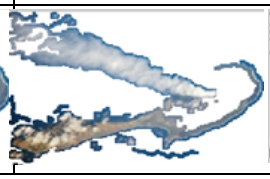
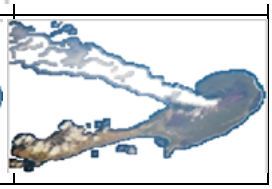


Figure 5-40 : The graph of ROI image size with different shapes and sizes of the structure element.

5.7 Colour and grey scale image study of image segmentation

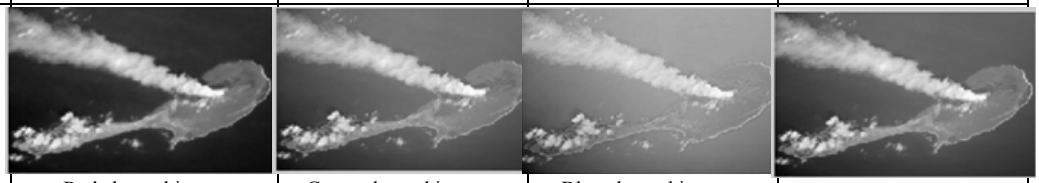
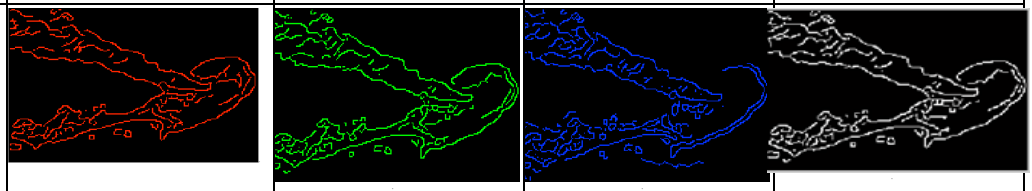
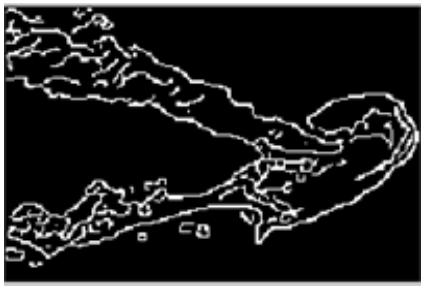
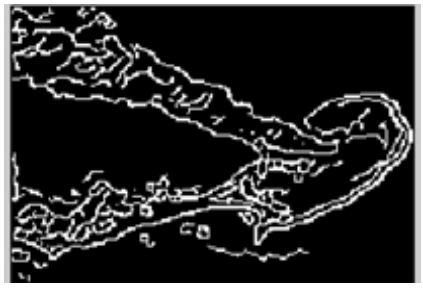
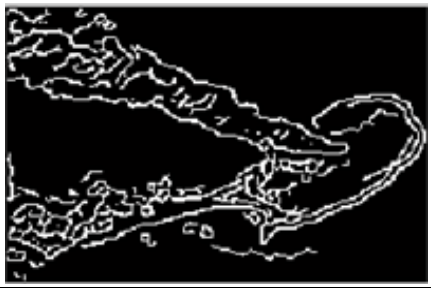
This section studies the results of edge image segmentation using four different inputs and comparing them; grey scale, red, green and blue channel images. The pixel colour in an image is a combination of the three colours red, green and blue (RGB). In the previous analysis the bands have been combined and the overall grey level determined. It is also possible to perform the analysis on each colour. The analysis starts with the separation of the image into three different sub bands and then each sub image is designated as the input for the proposed ROI automatic detection based on edge segmentation. The result of this test is shown in Table 5-14 and displays the image in red, green and blue channels and grey scale along with the output edge detection, ROI mask and ROI images. The results show that the single channel input image for edge detection produces a ROI image with some areas of interest missing (see the blue channel image test). The reason for this is that there is a very small difference in the blue channel pixel values between the missing areas on land and the background ocean nearby. Therefore the edge detection faces the difficulty of detecting the edge boundary between these two areas.

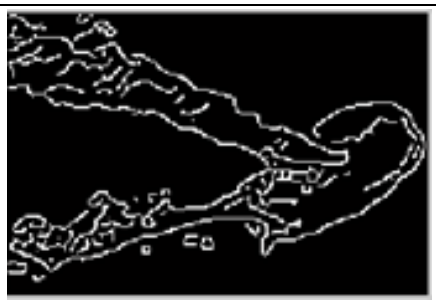
Table 5-14 : ROI automatic detection based on edge detection with four different inputs for edge detection (R, G, B channel image and grey scale image).

Test	Red channel test	Green channel test	Blue channel test	Grey scale test
Image	 Red channel image	 Green channel image	 Blue channel image	 Grey scale image
Edge Detected				
ROI mask				
ROI image				

Additionally, the combination of edge detected images from the three red, green and blue channels has been conducted. Table 5-15 shows the result of this experiment.

Table 5-15 : Edge detected combination from R, G and B channel images and greyscale image test result.

Test	Red channel test	Green channel test	Blue channel test	Grey scale test
Image	 Red channel image		Green channel image	Blue channel image	Grey scale image
Edge Detected				
Combination of edge detected images	1. Combination of edges from red and green channels			
	2. Combination of edges from red and blue channels			
	3. Combination of edges from green and blue channels			

	4. Grey scale edge detected	

For example the combination of edge detected image from red and green channels is shown in Figure 5-41. The coloured regions identified are those in which the edges can only be observed in either the red or the green image.

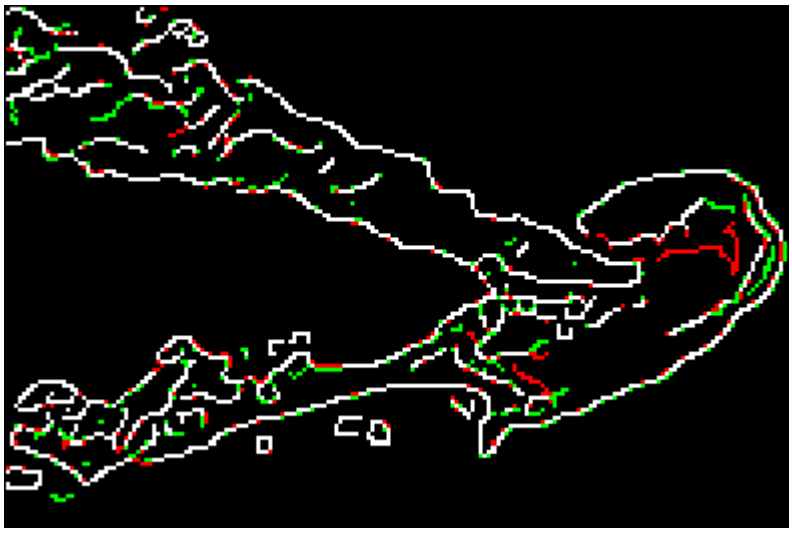


Figure 5-41 : Edge combination from red and green channel edge detected image.

The combined edge detected image results shows that more edges have been detected, which reduces the possibility of missing areas resulting from the ROI detection. However using a combination of edge detection is a more complicated process for the system due to the double edge detection for both channel images and also the combination process itself. From experiments it can be seen that edge detection from the grey scale image, in general, gives the best result. Whilst the single channel image processing could be used for some specific applications the combination method is the one pursued in this thesis because it provides a greater generic capability.

5.8 Summary

This chapter presents ROI automatic detection consisting of edge, histogram and texture segmentation. Each detection process producing an ROI image has been described. The ROI automatic detection methods have been implemented and tested with satellite images. The results illustrate that the ROI techniques can reduce the size of the satellite images. To verify the accuracy of the proposed system a number of tests have been presented comparing the number of different pixels between ROI automatic and manual detection methods. It is found that there is a very small difference between both systems. The structure element that is used in morphology mathematics has been studied to find the appropriate size and shape for application in the proposed system. Both the structure element size and shape affect the size of the output ROI image but it has been found that the square element with size 15 x 15 pixels represents the best approximation to the manual identification of the region of interest.

Chapter 6

Image compression

The main purpose of image compression is to decrease the redundancy of the image and reduce the data size, but retain all information. Image size reduction is one of the solutions to achieve a reduction in size of the onboard satellite image data before transmission to the ground station. This technique also decreases the satellite communications bandwidth required. Over the past few years, the development and implementation of a variety of powerful wavelet based image compression algorithms have become popular because of many advantages such as; the very high compression ratio achievable, the high efficiency of wavelet compression at low bit rates and a better correspondence between the wavelet based image compression and the Human Visual System (Talukder and HaradaII 2007).

There are many other techniques for image compression. Some methods are designed for specific image applications, consequently when applied for other applications they may perform well. However some compression techniques are sufficiently flexible that the compression parameters can be adjusted to achieve better results.

This chapter considers three different methods for image compression. Firstly there is the technique recommended for use in spacecraft, which is the data compression algorithm from the CCSDS based on wavelet transform. This research also investigates other wavelet image compression methods based on wavelet transform, namely the 'Jpeg2000' and 'Haar' image compressions. The results of implementing these techniques are detailed and discussed. The standard deviation of an image, which relates to the image compression ability, has also been investigated and is presented in this chapter.

6.1 CCSDS image compression recommendation

The limitations of transmission bandwidth, downlink data rate and storage capacity are the key drivers to develop the image compression technology (Kiadtikornthaweeyot 2012). As more innovative payloads are used on cubesats, the data storage requirements onboard will increase (Yu 2009). The end result is that improved image compression on spacecraft is

extremely necessary to fulfil future satellite missions. Image compression, in addition to reducing the size of the original image, will also decrease the hardware requirements and costs of the communication system (Jian-wei and Jian-dong 2008).

The CCSDS data compression working group started to establish an appropriate image compression for satellites in 1998. The CCSDS has recommended the specific standard image compression, which is based on discrete wavelet transform (DWT), which has attracted much attention because of its outstanding decorrelation properties. The DWT is very efficient at high compression ratios and has become a productive transform for image compression in spacecraft (Fukuma, Tanaka et al. 2005; Pen-Shu, Armbruster et al. 2005; Guang, Ming et al. 2009).

The standard can be used to produce both lossy and lossless compression. The process consists of two functional parts; the first part is the discrete wavelets transform and the second part is the bit-plan encoder (BPE). The general operation of the algorithm is illustrated in Figure 6-1 (Pen-Shu, Armbruster et al. 2005; CCSDS 2007). The DWT processing performs decomposition of the image and the BPE encodes the decomposition data coefficients from the first module. The details of the image compression recommendation will be described in the next section.

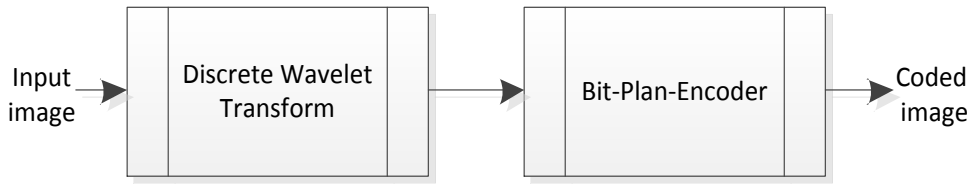


Figure 6-1 : Discrete Wavelet Compression system.

6.1.1 Discrete wavelets transform (DWT)

The CCSDS recommendation standard uses a wavelet image compression method, the 9/7 biorthogonal filter. The nine filter taps compute the low-pass filter output, whereas the high-pass output uses seven filters for calculation (Pen-Shu, Armbruster et al. 2005; Xiaodong, Huaichao et al. 2009). The process performs three levels of two-dimensional wavelet decompositions. The filter coefficients specified in the recommendation are shown in Table 6-1 (Pen-Shu, Armbruster et al. 2005; CCSDS 2007; Garcia-Vilchez and Serra-Sagrista 2009).

Table 6-1 : The recommended analysis filter coefficients.

Analysis filter coefficients		
i	Lowpass filter	Highpass filter
0	0.852698679009	-0.788485616406
±1	0.377402855613	0.418092273222
±2	-0.110624404418	0.040689417609
±3	-0.023849465020	-0.064538882629
±4	0.037828455507	

The DWT can be described in terms of a filter bank as shown in Figure 6-2. The analysis filter bank consists of a low-pass filter *(L)* and high-pass filter (*H*) at each decomposition stage. The original image is passed to low and high pass filters, so the image is separated into two bands. The low pass filter relates to an averaging operation, which extracts the detailed information of the image.

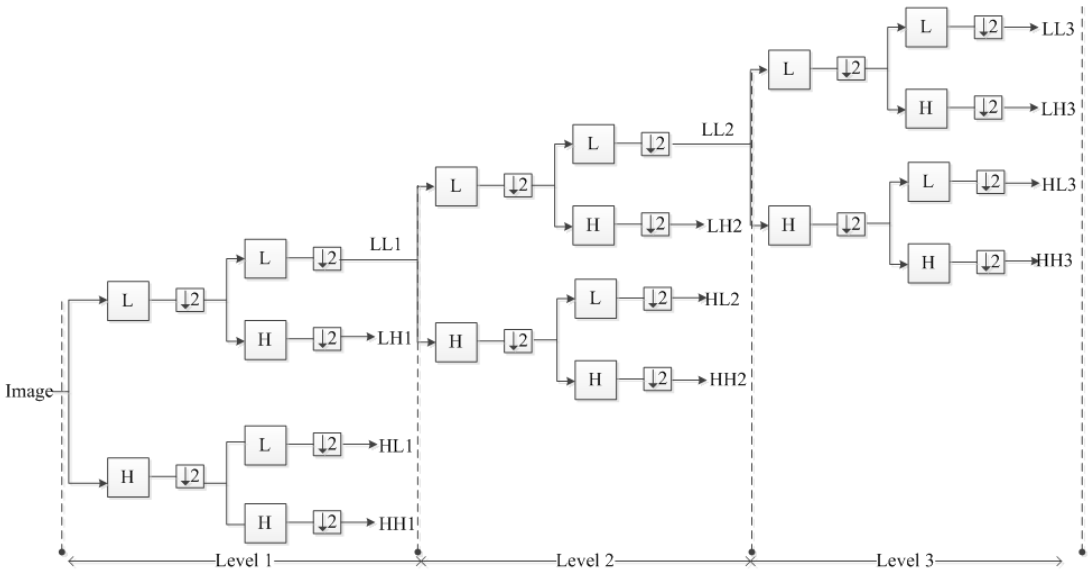


Figure 6-2 : The three level wavelet decomposition filter bank

The first level decomposition separates the image into four bands which are represented by LL_1, LH_1, HL_1 and HH_1. This has the following format:

LL1 low_low shows the source signal with a coarse resolution

LH1 low_high shows the details of the discontinuities along the column direction
HL1 high_low shows the details of the discontinuities along the row direction
HH1 high_high shows the details of the discontinuities along the diagonal direction

For second level decomposition, the LL_1 sub-band is subject to the same 2-dimension DWT and this process is repeated until the image is fully deconstructed. In the three levels of decomposition, the LL_3 is the coarsest sub-band at the top of the structure and the finest sub-band is at the bottom (Pen-Shu, Armbruster et al. 2005; CCSDS 2007; Garcia-Vilchez and Serra-Sagrista 2009). The visualisation of the wavelet decomposition producing 10 sub-bands shown in

Figure 6-3 is a 3-level two dimensional decomposition structure (Usevitch 2001). Figure 6-4 shows the example results of the compressed image in each level by separating the output into each sub-band.

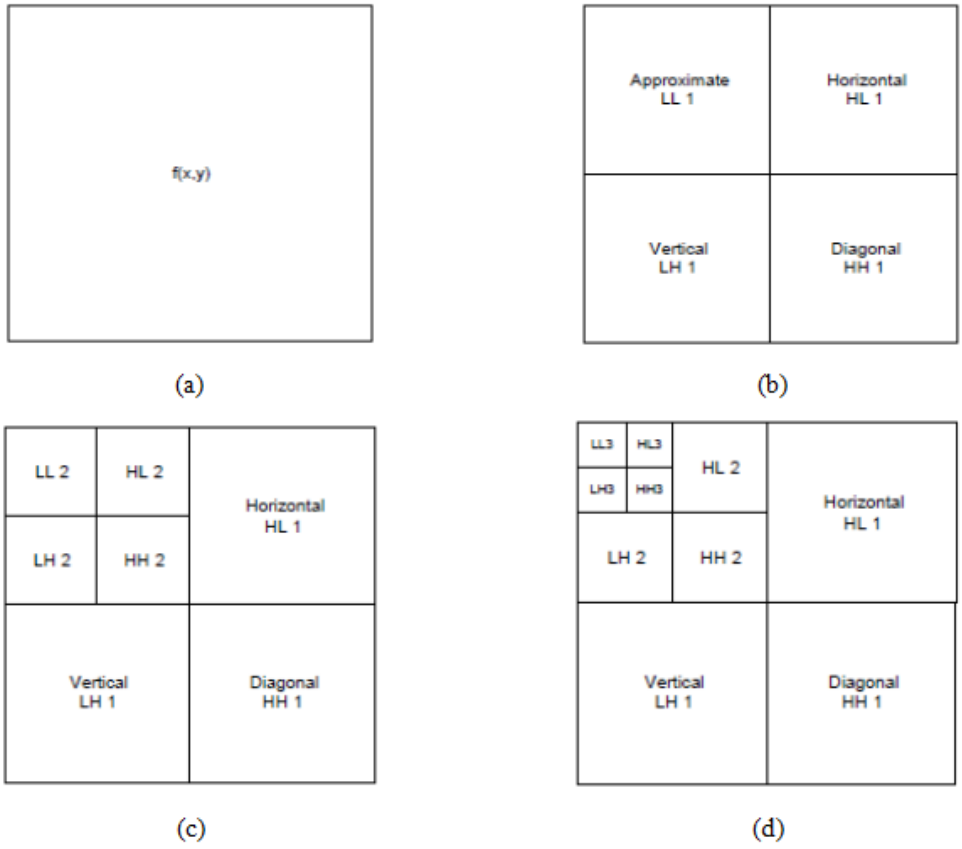


Figure 6-3 : Two-dimensional DWT (a) The original image. (b) First level decomposition. (c) Second level decomposition. (d) Third level decomposition.

Figure 6-4 : Three level decomposition of an example satellite image. Left hand side is the original image, right hand side demonstrates the three level decomposition.

6.1.2 Bit-Plane Encoding (BPE)

The BPE processes the wavelet coefficients into groups of 64 coefficients. These coefficients are a group of the number from the image pixel values, which have been processed using the wavelet decomposition method. These groups of coefficients are called blocks. According to the previous section, the DWT process generates a single DC coefficient, which is taken from the lowest spatial frequency sub-band (LL_3). The remaining coefficients are AC coefficients. These are taken from the other sub-bands using the hierarchical structure given by the DWT step. The BPE codes the DWT coefficients block by block and each block is coded independently. The final result is a compressed code bit stream, which is composed of an embedded data format. This data can be stored or transmitted (Pen-Shu, Armbruster et al. 2005).

Reconstruction of the image can be performed by using IDWT (inverse discrete wavelet transforms). The IDWT is the inverse process of wavelet decomposition, which consists of up-sampling and inverse filtering. The reconstructed image is expected to be exactly the same as the original image, as the image compression method is lossless (Antonini, Barlaud et al. 1992).

6.2 Jpeg2000

The Joint Photographic Experts Group (Jpeg) committee was begun in 1986 with the first published Jpeg image compression standard and then Jpeg2000. The image compression standard Jpeg 2000 is based on the discrete wavelet transform (DWT). The performance of Jpeg2000 has been significantly improved over the discrete cosine transform (DCT). Jpeg2000

can compress many types of still images such as bi-level, grey-level and color with different image characteristics such as medical, natural and rendered graphics images. Jpeg2000 has the following improvements over standard jpeg compression; higher compression ratio, higher efficiency, lower complexity and more memory efficiency (Langton 2002). Jpeg2000 image compression starts with the decomposition of the image frequencies into sub bands then each sub band is separated into different blocks, which are coded by embedded block coding with the optimized truncation (EBOT) algorithm. The next process is quantization (Singh 2009).

6.3 Haar

The Haar wavelet image compression is the simplest wavelets algorithm, involving calculating averages and differences of adjacent pixels, and has been used in many methods of discrete image transformation and processing (Karatas and Ince 2009). The Haar wavelet compression can be performed by both lossless and lossy methods.

The Haar transform has been used for image processing and pattern recognition for the primary reason that it requires low computation. Unfortunately the Haar transform has low quality and energy compaction for images (Kamrul and Koichi 2007). The energy compaction is a measure of the efficiency of the transform to compress the image and is the ability of the compression scheme to compact the signal energy into as few coefficients as possible (Andrew Yeung and Chen 1998). Therefore research is on-going to develop improvements (Waydo, Henry et al. 2002).

6.4 Image compression analysis

These three image compression methods have the same image decomposition based technique; wavelet image transform. This section presents the main process of image decomposition using wavelet image transforms.

The analysis was performed using Matlab R2010a programming to illustrate the behaviour of the image compression process for each image compression method. The results below are from one sample of biorthogonal 9/7 image compression. The outputs of each level decomposition were reconstructed to be visualised as an image. Figure 6-5 illustrates the decomposition of data, which is performed by the two-dimension DWT at level 1. There are four output sub-bands, LL_1, LH_1, HL_1 and HH_1.

Figure 6-6 shows the decomposition of data, which is performed by the two-dimension DWT at level 2. There are a further four more output sub-bands, LL_2, LH_2, HL_2 and HH_2.

Figure 6-7 illustrates decomposition of data, which is performed by the two-dimension DWT at level 3. There are another further four more output sub-bands, LL_3, LH_3, HL_3 and HH_3.

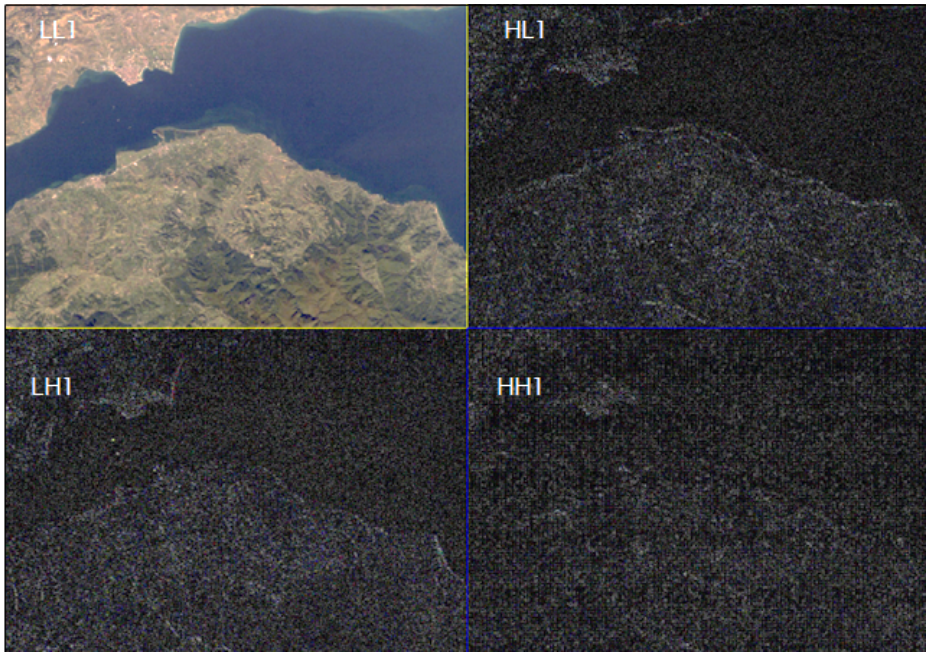


Figure 6-5 : Decomposition at level 1.

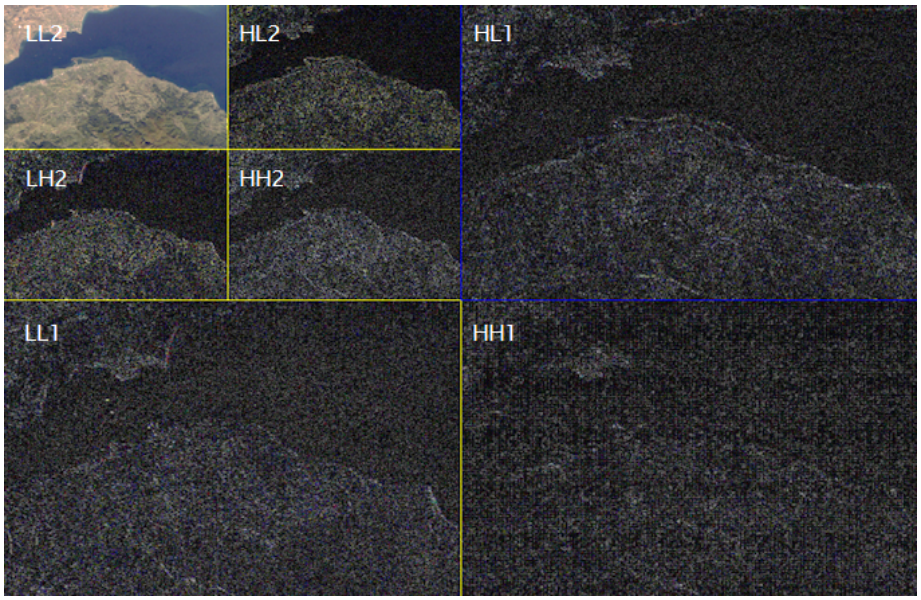


Figure 6-6 : Decomposition at level 2.

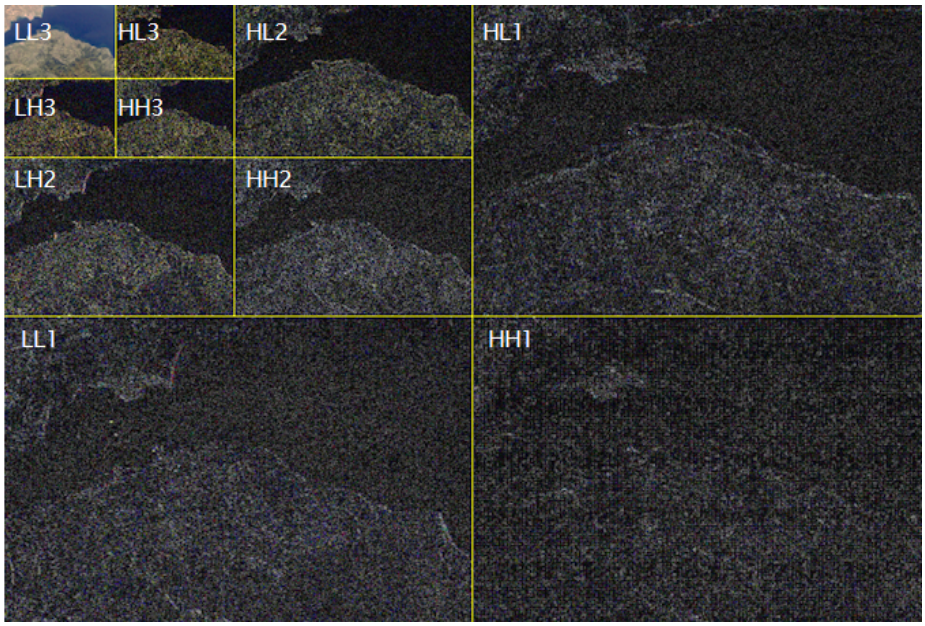
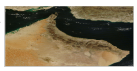


Figure 6-7 : Decomposition at level 3.

The objective of these image compression tests is to understand the capability of these three different image compression techniques. Table 6-2 illustrates the application of three different wavelet image compressions to a set of satellite images. The table shows the compressed image size and the compression ratios. The average compression ratio is 69.38 % for the image compression set based on biorthogonal 9/7 compression. The image reconstructions are exactly the same as the original images. If the compressed image can be recovered exactly as the original, the compression is said to be lossless. Therefore the wavelet image compression based on biorthogonal 9/7 is a lossless image compression method. Lossless compression is only possible using low compression ratios, which allow one to trade off compression ratio with image quality. Therefore whether the application can accept a lossy compression method is an important consideration in determining the possible compression (Pinho and Neves 2006; Manjinder Kaur 2013).

The results from the Haar image compression show that the average compression can reduce the size of original imges to 26.44 % of original image, whilst for the Jpeg2000 the average is 30.39 %.

Table 6-2 : The results of image compression in three different methods.

Input image	Input size (A) KB	9/7 biorthogonal			Haar			Jpeg2000		
		Compressed size (B) KB	(B/A)x100	CR1	Compressed size (C) KB	(C/A)x100	CR2	Compressed size (D) KB	(D/A)x100	CR3
1. 	3,207	2,523	78.67%	1.27: 1	1,176	36.67%	2.73 :1	1,040	32.43%	3.08 :1
2. 	3,988	2,390	59.93%	1.67: 1	810	20.31%	4.92 :1	1,051	26.35%	3.79 :1
3. 	10,184	4,254	41.77%	2.39: 1	1,474	14.47%	6.91 :1	1,981	19.45%	5.14 :1
4. 	3,381	2,123	62.79%	1.59: 1	834	24.66%	4.05 :1	1,069	31.62%	3.16 :1
5. 	13,426	8,908	66.35%	1.51: 1	3,706	27.60%	3.62 :1	4,212	31.37%	3.19 :1
6. 	12,685	11,497	90.63%	1.10: 1	3,945	31.10%	3.22 :1	3,307	26.07%	3.84 :1
7. 	1,678	1,435	85.52%	1.17: 1	508	30.27%	3.30 :1	763	45.47%	2.20 :1

The image compression based on the biorthogonal 9/7 is the CCSDS recommended compression technique for spacecraft. It is different from the Jpeg2000 standard in the following ways (Pen-Shu, Armbruster et al. 2005; Chen, Lin et al. 2009);

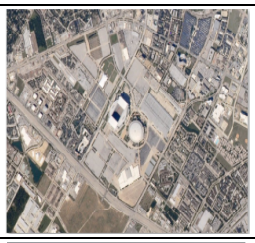
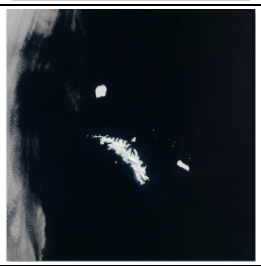
- This recommendation is for particular use with high data rate instruments onboard satellites.
- The recommendation from CCSDS is easy to implement with less complex hardware and software and can be implemented quickly using low power hardware.
- It can be implemented without in-depth algorithm knowledge.

Therefore the different image compression technique selection for the system/projects is dependent on the project requirements. The trade-off between the quality of the image compression and the compression ratio are very important considerations.

The relationship between the image characteristics and the compression capability has also been studied. Brien (O'Brien 2013) stated that "*The 'business' of an image is possible to describe by an image entropy quantity"*. If any image has very little contrast or there are many pixels that have a similar value, then the image has low entropy. For example this is the case if an image contains large areas of an ocean surface, black sky or desert areas. In addition the entropy is nearly zero in the case that an image is perfectly flat. In these cases the image can be

compressed to a very small size. On the other hand, higher entropy images cannot be compressed as much as low entropy images, because higher entropy images contain a high density of information (O'Brien 2013). Table 6-3 shows the entropy value of four different satellite images. There are two high resolution satellite images of land areas and two satellite pictures of islands in the ocean. An image will have a high entropy value when there are many details or a lot of change, such as image land-1 and land-2 that have entropy at 7.3566 and 7.4489 respectively. Entropy values are lower in image clear ocean-1 and ocean-2 with entropy at 5.8813 and 5.4750 respectively. The reason for this is that both pictures are very smooth and do not have much information variation if compared to the first two land images.

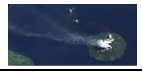
Table 6-3 : The entropy of four different satellite images.

Image		Entropy	Image size
1. Land-1		7.3566	22,629 KB
2. Land-2		7.4489	3,650 KB
3. Clear ocean-1		5.8813	2,464 KB
4. Clear ocean-2		5.4750	6,361 KB

The relationship between entropy of the image and compression capability has been investigated. The computation of the ratio between the size of the original image and compressed image is presented by the compression ratio (Alaitz, Raffaele et al. 2012). For instance a compression ratio of 1.5 : 1 for an image means that it is compressed by one and a half times compared to the original image. Table 6-4 shows the test results of the relationship between entropy and compression ratio for these seven satellite images.

Table 6-4 shows the relationship in a line graph. For instance, input image number-3 has an entropy value of 6.1526 and can be compressed to 41.77 % of the original image (compression ratio 2.4: 1). This should be compared to the test result of image number-6, which has a higher entropy value of 7.2343, this can be compressed to 90.63 % of the original image (compression ratio 1.1: 1). The results show that images with lower entropy have the possibility to achieve a higher compression ratio and conversely images with higher entropy might only be compressed with a lower compression ratio. Therefore the image compression ratio is dependent on the characteristics of an image.

Table 6-4 : Test results of entropy and the compression ratio of satellite images.

Input image	Compression percentage	Compression ratio	Entropy
1. 	78.67 %	1.27 : 1	7.5486
2. 	59.93 %	1.67 : 1	7.2863
3. 	41.77 %	2.40 : 1	6.1526
4. 	62.79 %	1.59 : 1	6.5383
5. 	66.35 %	1.51 : 1	6.4090
6. 	90.63 %	1.10 : 1	7.2343
7. 	85.52 %	1.17 : 1	7.5151

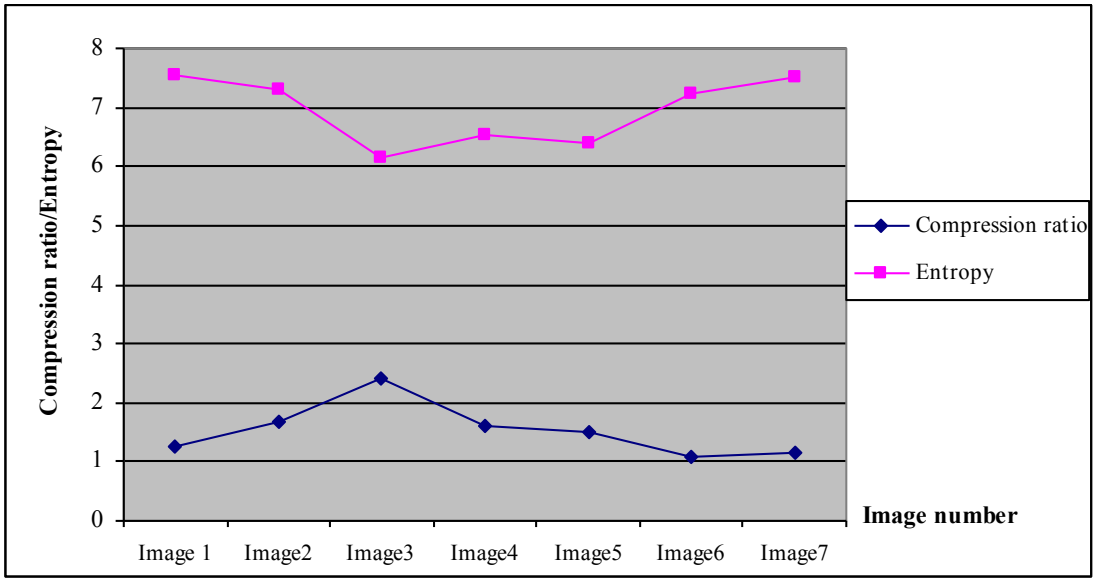


Figure 6-8 : The relationship between the entropy number and compression ratio of example images.

6.5 Summary

The image compression techniques presented in this chapter, consists of the 9/7 biorthogonal, the Haar and Jpeg2000 wavelet based image compression. The image wavelet transform using a 9/7 biorthogonal filter, is the CCSDS image compression recommendation for satellites and has therefore been investigated in this thesis. Although the 9/7 biorthogonal filter gave the lowest compression ratio compared to the Haar and Jpeg2000 it has been studied and tested for implementation in hardware by the CCSDS working group. Their report shows that, compared with alternative techniques, the 9/7 biorthogonal image compression is easiest to implement with less complex hardware and software and can be implemented quickly using low power hardware. This thesis will therefore consider the 9/7 biorthogonal image compression method in more detail.

The results of the analysis described in this chapter show that the image size can be reduced by a compression algorithm based on wavelet transforms. In the following chapter the proposed adaptive image compression system for detecting the ROI automatically for satellite images is described.

Chapter 7

An image compression method for a cubesat

This research proposes a new image compression algorithm to decrease the onboard satellite image size that can be implemented on a cubesat. The proposed algorithm consists of two main systems, firstly the region of interest automatic detection and secondly the image compression module. Three different techniques have been used to detect the region of interest automatically; edge, histogram and texture segmentations. In addition, three different image compression methods have been investigated; 9/7 biorthogonal, Haar and Jpeg2000 wavelet image compression.

This chapter describes the proposed alternative adaptive image compression system for a cubesat and presents the simulation results. The final section gives the results of an investigation into the implementation of this system on the hardware used in a cubesat.

7.1 The proposed system description and experiment

The proposed algorithm combines the ROI automatic detection technique and the image compression system together (Kiadtikornthaweeyot 2012) as illustrated in Figure 7-1.

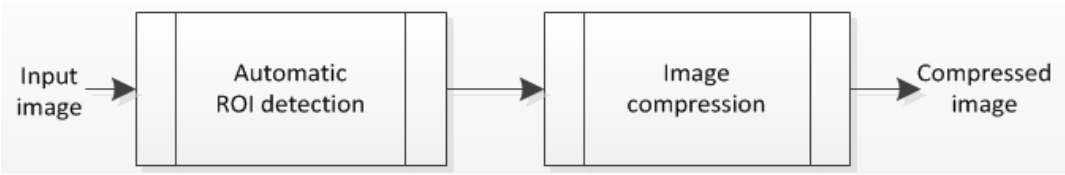


Figure 7-1 : The flow chart of the proposed alternative adaptive image compression based on ROI automatic detection for cubesat.

The first module is the automatic ROI detection, which reduces the size of the original image by removal of non-interesting areas in the image. The ROI is an area of an image that contains useful information, whereas non-ROI is the background in the image. Typical non-ROI areas might be oceans or clouds, which consist of groups of similar valued pixels and therefore have low entropy.

In order to detect and analyse a ROI of the image, a combination of several image segmentation techniques have been studied. The techniques can be classified into three groups; edge, histogram and texture segmentation techniques (Kiadtikornthaweeyot 2012). Experiments for these have been conducted and implemented to study the suitable ROI module for a cubesat.

The second part of the proposed system is the image compression module. There are many types of image compression, however this research implements three techniques based on wavelet image compression. There are wavelet image compression based on 9/7 biorthogonal, Haar and Jpeg2000. The proposed system has been implemented and tested for four different applications:

- Test 1: The ROI automatic detection based on the *edge* image segmentation method and 9/7 biorthogonal wavelet image compression.
- Test 2: The ROI automatic detection based on the *histogram* image segmentation method and 9/7 biorthogonal wavelet image compression.
- Test 3: The ROI automatic detection based on the *texture* image segmentation method and 9/7 biorthogonal wavelet image compression.
- Test 4: The ROI automatic detection based on the *edge* image segmentation method and three different types of image compression techniques.

These first three tests use the same wavelet image compression method. The results from test 1, 2 and 3 are shown in Table 7-1, Table 7-2 and Table 7-3 respectively. For example, input image-1 in Table 7-1, the proposed system can reduce the size of the original image from 3.21 MB to 2.42 MB after the image passes through the ROI automatic detection module based on edge segmentation. After processing by the image compression module, the output image size is 74.06 % of the original.

Table 7-1：The proposed image compression experiment results based on edge segmentation.

Input image	ROI automatic detection based on edge segmentation and 9/7 biorthogonal wavelet image compression.				
	Input size : A	ROI image size	Wavelet compressed : B	(B/A)x100	CR1

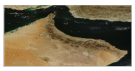
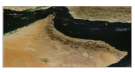
1. 	3,207 KB	2,415 KB	2,375 KB	74.06 %	1.35 : 1
2. 	3,988 KB	2,293 KB	2,149 KB	53.89 %	1.86 : 1
3. 	10,184 KB	3,568 KB	3,330 KB	32.70 %	3.06 : 1
4. 	3,381 KB	2,008 KB	1,934 KB	57.20 %	1.75 : 1
5. 	13,426 KB	9,390 KB	8,079 KB	60.17 %	1.66 : 1
6. 	12,685 KB	11,847 KB	11,343 KB	89.42 %	1.12 : 1
7. 	1,678 KB	1,281 KB	1,247 KB	74.31 %	1.35 : 1
Summary	Average reduction to 63.11%				

Table 7-2 : The proposed image compression experiment results based on histogram segmentation.

Input image	ROI automatic detection based on histogram segmentation and 9/7 biorthogonal wavelet image compression.				
	Input size : A	ROI image size	Wavelet compressed : C	(C/A)x100	CR
1. 	3,207 KB	2,446 KB	2,394 KB	74.65 %	1.34 : 1
2. 	3,988 KB	2,104 KB	1,978 KB	49.60 %	2.02 : 1
3. 	10,184 KB	6,285 KB	4,230 KB	41.54 %	2.41 : 1
4. 	3,381 KB	2,382 KB	2,066 KB	61.11 %	1.64 : 1
5. 	13,426 KB	9,957 KB	8,321 KB	61.98 %	1.61 : 1
6. 	12,685 KB	12,384 KB	11,186 KB	88.18 %	1.13 : 1
7. 	1,678 KB	1,355 KB	1,309 KB	78.01 %	1.28 : 1
Summary	Average reduction to 65.01%				

Table 7-3: The proposed image compression experiment results based on texture segmentation.

Input image	ROI automatic detection based on texture segmentation and 9/7 biorthogonal wavelet image compression.				
	Input size : A	ROI image size	Wavelet compressed : D	(D/A)x100	CR
1. 	3,207 KB	2,745 KB	2,520 KB	78.58 %	1.27 : 1
2. 	3,988 KB	2,862 KB	2,367 KB	59.36 %	1.68 : 1
3. 	10,184 KB	3,765 KB	3,676 KB	36.10 %	2.77 : 1
4. 	3,381 KB	2,145 KB	2,045 KB	60.48 %	1.65 : 1
5. 	13,426 KB	10,934 KB	8,439 KB	62.86 %	1.59 : 1
6. 	12,685 KB	11,674 KB	11,322 KB	89.26 %	1.12 : 1
7. 	1,678 KB	1,338 KB	1,272 KB	75.80 %	1.32 : 1
Summary	Average reduction to 66.06%				

On average over the seven images the proposed ROI detection based on the edge detection and image compression reduces the image size to 63.11 % of the original. The ROI detection based on histogram and texture segmentation decreases the size of satellite images to an average of 65.01 % and 66.06 % of the original respectively. Whilst the value of size reduction depends on the type of image the small differences indicate that all methods are equally good at identifying the ROI. In addition the experiment shows that the proposed ROI detection based on texture segmentation and subsequent image compression takes a considerably longer time (950.16 seconds) to compute compared to other segmentations that are 7.12 and 14.36 seconds for edge and histogram segmentation respectively (test performed on Intel core 2 Duo CPU 3.16 GHz and RAM 4.00 GB).

The objective of test 4 (the ROI automatic detection based on edge segmentation and three types of image compression) is to evaluate the three different image compression techniques. There are three different subsystem tests:

- The ROI detection and the 9/7 biorthogonal image compression (same as Test 1)
- The ROI detection and the Haar image compression
- The ROI detection and the Jpeg2000 image compression

Table 7-4 shows the result of the image compressions from these three subsystem tests. For example image number 3, using 9/7 biorthogonal, the proposed system can reduce the image size at a compression ratio of 3.06 : 1. The Haar and Jpeg2000 techniques with the proposed ROI automatic detection system can reduce the size of original image at compression ratios of 10.94 : 1 and 8.65 : 1 respectively.

Table 7-5 shows the image compression of images that have been compressed by using only standard image compressions. There is also information in the table which shows the image compression of images that have been compressed using the proposed system based on ROI automatic detection and standard image compression methods combined. The comparison shows that the proposed system can reduce the size of the original image to be smaller than the size of image compressed by only standard image compression methods. For example image number 7, the compressed image by using 9/7 biorthogonal image compression can reduce the size to be 85.52 % of the original image. Whereas, using the proposed system (ROI and 9/7 biorthogonal) the system can reduce the image to be 74.31 % of the original image size.

Table 7-4 : Comparison of the ROI automatic detection based on edge segmentation and three types of wavelet image compressions.

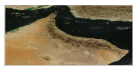
Input image	Input size	ROI size	ROI based on edge segmentation and 9/7 biorthogonal		ROI based on edge segmentation and Haar compression		ROI based on edge segmentation and Jpeg2000	
			Output1 Size	CR1	Output2 Size	CR2	Output3 Size	CR3
1. 	3,207 KB	2,415 KB	2,375 KB	1.35 : 1	1,144 KB	2.80 : 1	986 KB	3.25 : 1
2. 	3,988 KB	2,293 KB	2,149 KB	1.86 : 1	743 KB	5.37 : 1	812 KB	4.91 : 1
3. 	10,184 KB	3,568 KB	3,330 KB	3.06 : 1	931 KB	10.94:1	1178 KB	8.65 : 1
4. 	3,381 KB	2,008 KB	1,934 KB	1.75 : 1	779 KB	4.34 : 1	940 KB	3.6 : 1
5. 	13,426 KB	9,390 KB	8,079 KB	1.66 : 1	3,262 KB	4.12 : 1	3489 KB	3.85 : 1
6. 	12,685 KB	1,1847 KB	11,343 KB	1.12 : 1	3,827 KB	3.31 : 1	3204 KB	3.96 : 1
7. 	1,678 KB	1,281 KB	1,247 KB	1.35 : 1	470 KB	3.57 : 1	600 KB	2.81 : 1

Table 7-5 : Comparison of the standard image compressions only and the proposed system based on ROI automatic detection and standard image compression

Input image	Image compression rate by using					
	1. Standard image compression only			2. The proposed ROI detection and standard image compression		
	9/7 biorthogonal	Haar	Jpeg2000	ROI + 9/7 biorthogonal	ROI + Haar	ROI + Jpeg2000
1. 	78.67 %	36.67 %	32.43 %	74.01 %	35.67 %	30.75 %
2. 	59.93 %	20.31 %	26.35 %	53.88 %	18.63 %	20.36 %
3. 	41.77 %	14.47 %	19.45 %	32.70 %	9.14 %	11.57 %
4. 	62.79 %	24.66 %	31.62 %	57.20 %	23.04 %	27.80 %
5. 	66.35 %	27.60 %	31.37 %	60.14 %	24.29 %	25.98 %
6. 	90.63 %	31.10 %	26.07 %	89.42 %	30.17 %	25.25 %
7. 	85.52 %	30.27 %	45.47 %	74.31 %	28.01 %	35.76 %
Average	69.38 %	26.44 %	30.39 %	63.09 %	24.14 %	25.35 %

7.2 Cloud identification and the proposed system

Some satellite images contain cloud. In this section the ROI automatic detection based on histogram segmentation is used to identify clouds in the input image. The objective is to detect the ROI image of this sample image and discard the clouds from the ROI image.

The process consists of two main processes; the first one is the ROI automatic detection of the image described previously. The second process is detection of cloud percentage in the ROI image. The proposed system detecting the ROI and discarding of clouds from the sample satellite image is shown in Figure 7-2, and Figure 7-3 shows the sample satellite image used for this test. Figure 7-4 is the histogram of the input grey scale image.

From the test result we can see that it is possible to detect the ROI by using histogram segmentation. Clouds are detected by using histogram segmentation separately. At the end of the process the cloud detected will be used as a mask to delete from the ROI of the sample image (ROI from process 1). Table 7-6 shows the result of the first process to define the ROI of

the sample image. Table 7-7 shows the result of the second process to identify and discard clouds from the ROI of the sample image. To fully explore the robustness of the method a more definitive test on a range of different cloud types should be done as the cloud in Figure 7-3 is particularly distinctive.

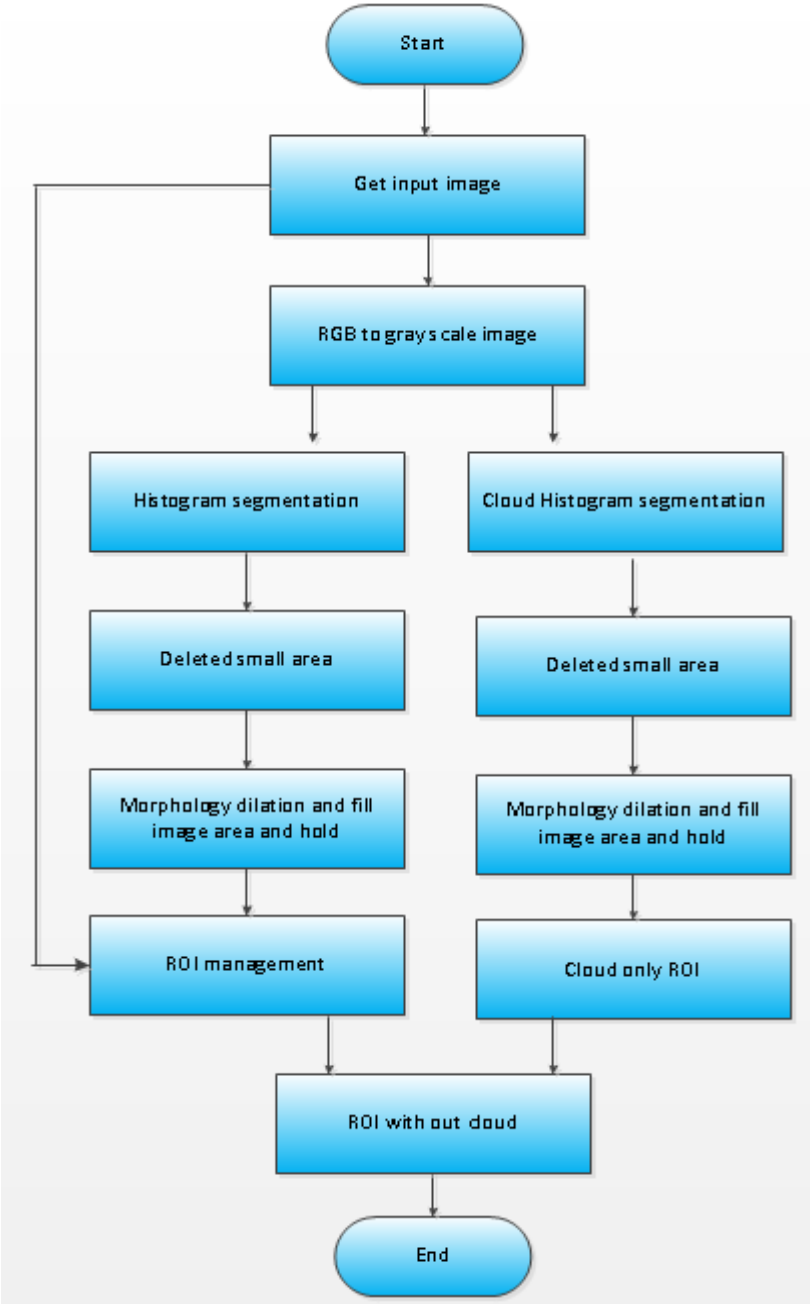


Figure 7-2 : ROI automatic detection based on histogram segmentation and cloud isolation process.

Figure 7-3 : Input image for the test.

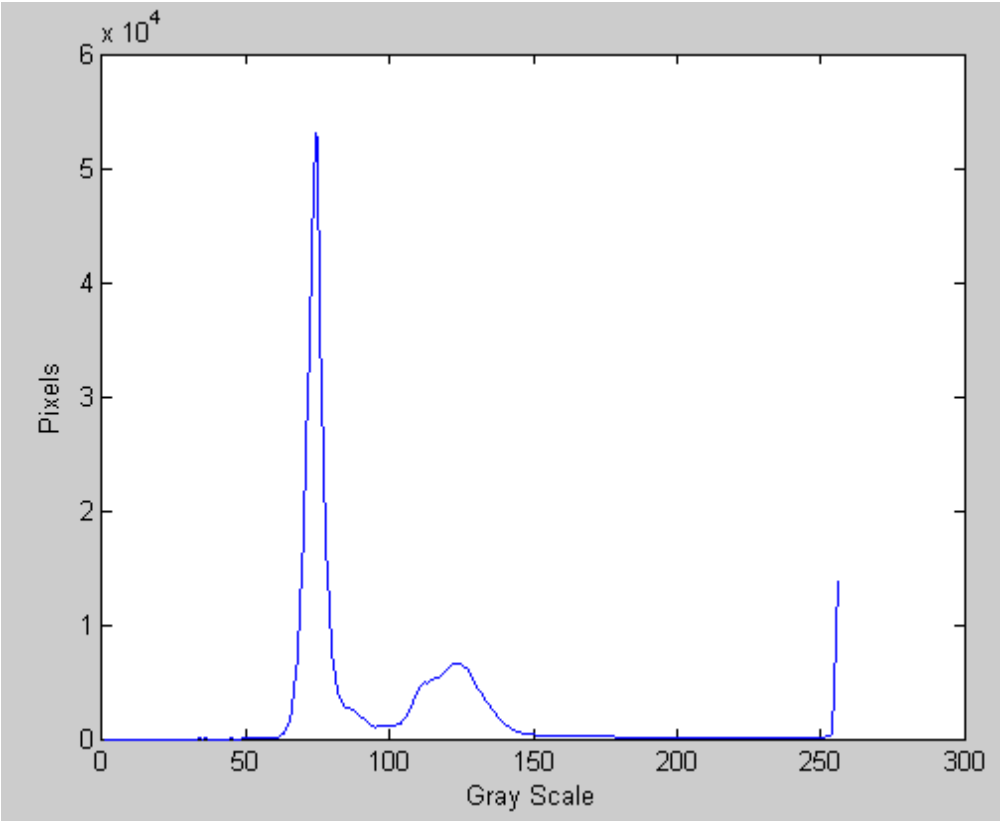
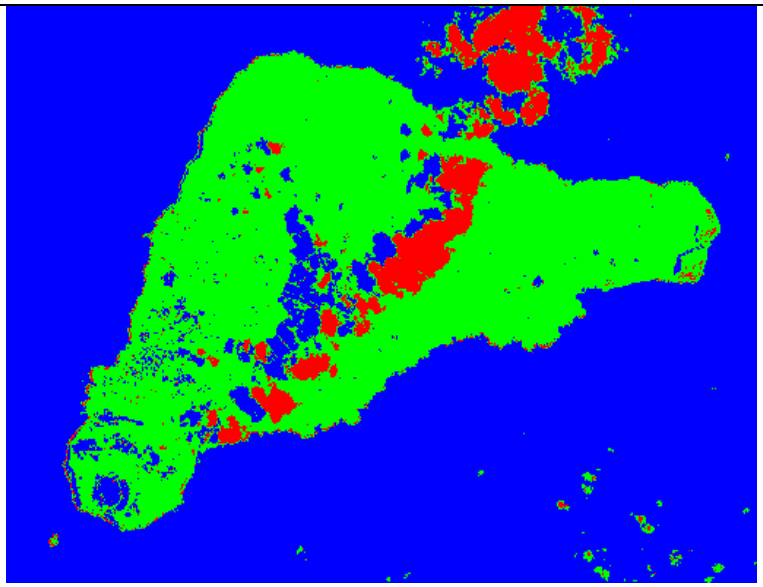


Figure 7-4 : Grey scale histogram image.

Table 7-6 : The result of the first process to detect ROI (including cloud).

Step	Result of each step
1. Histogram segmentation	
2. Delete small area	

3. Dilation	
4. ROI mask	

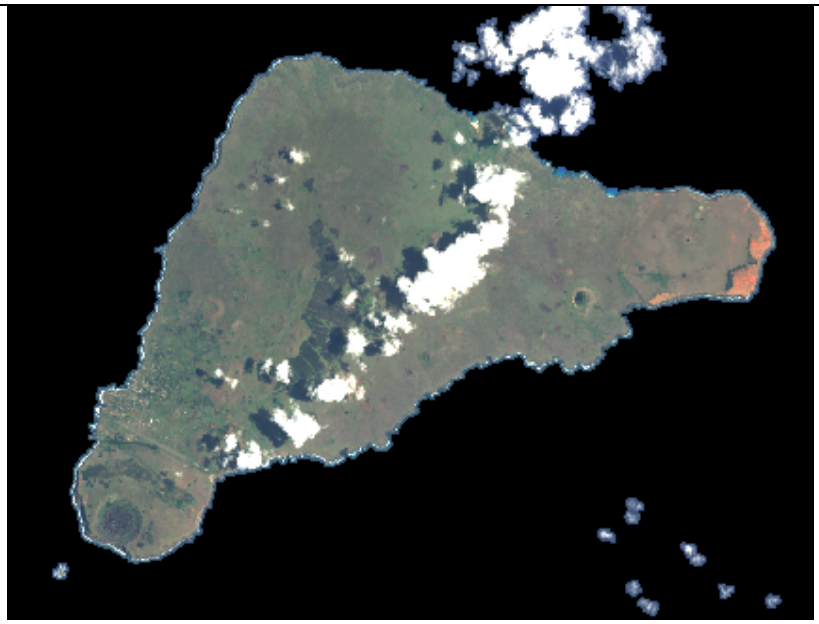
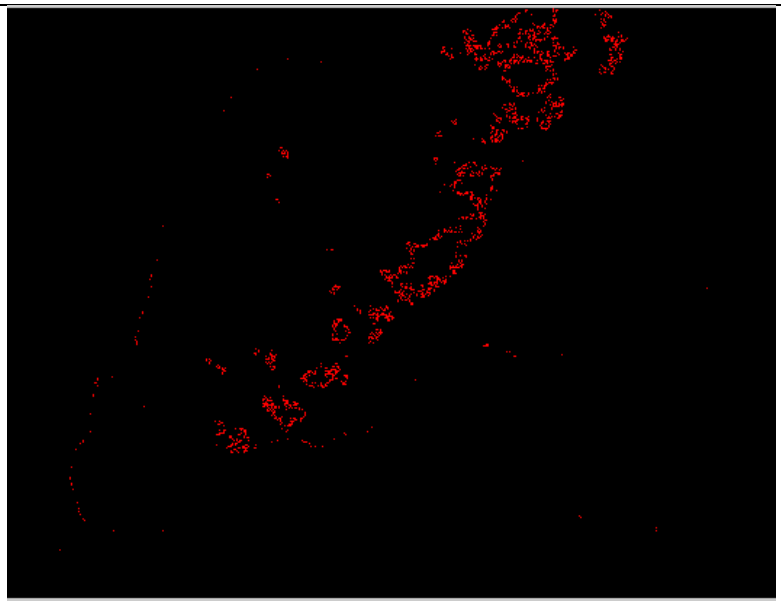
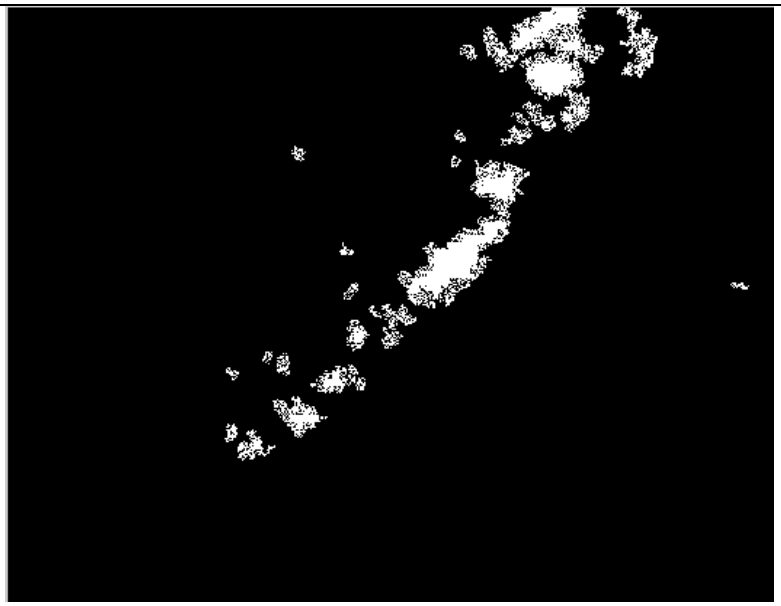
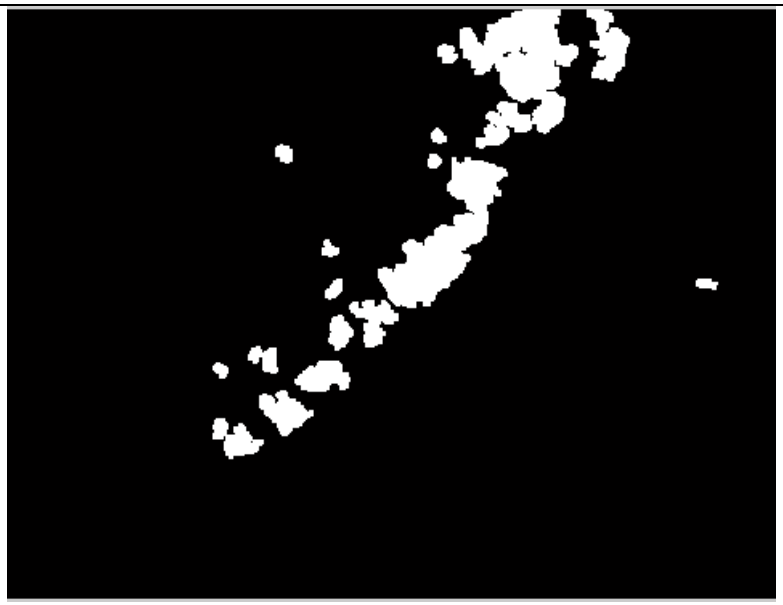
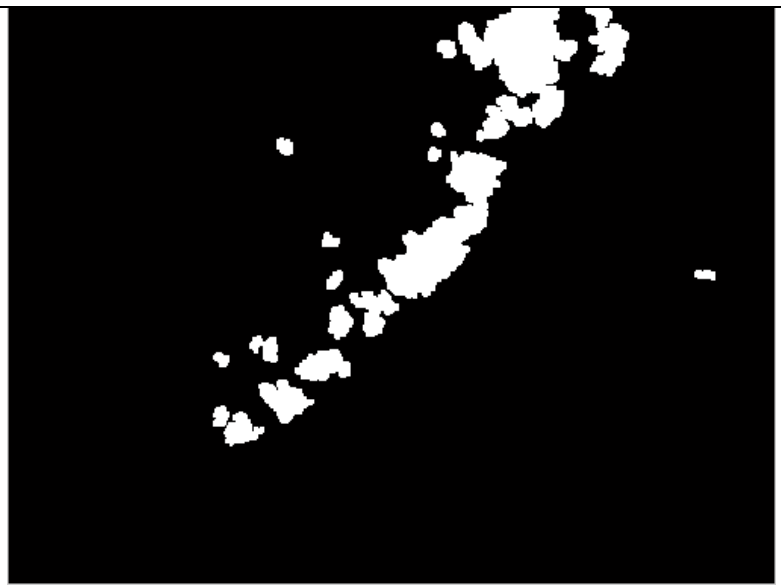
5. ROI image	

Table 7-7 : The result of the second process to detect cloud and then discard cloud for the ROI.

Step	Result of each step
1. Cloud histogram segmentation	

2. Delete small area	
3. Dilation	

4. Cloud mask	
5. Cloud ROI Only image	

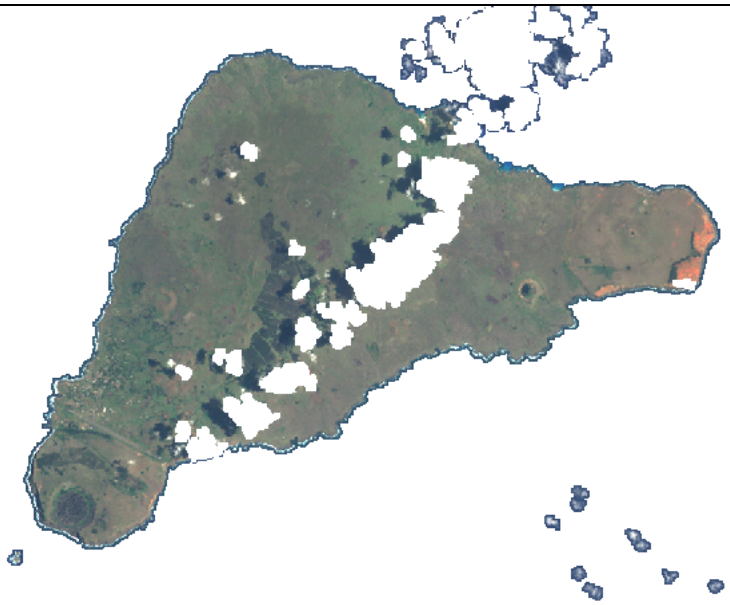
6. ROI image without cloud	

Table 7-8 shows the size of the ROI images, which are detected by using the proposed system based on histogram segmentation. There are three test results shown in this table.

- The ROI image size using the histogram segmentation to detect the ROI of the image. The ROI processing has reduced the size to be 46.45 % of the original image. If the ROI image is passed through the compression the size can be reduced to 44.62 % of the original image.
- The second stage is the detection of the clouds in the image. The output image size of the ROI image is 41.51 % of original image. When this output is applied for image compression the compressed image is 40.81 % of the original size.
- The last test is the image compression without ROI identification. The compressed image size is 56.50 % of the original image.

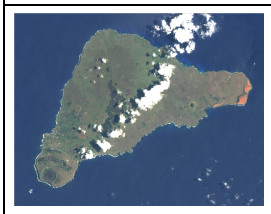
The result of the test of using the proposed system based on histogram segmentation to detect the ROI automatically and isolate the cloud automatically, shows that it is possible to reduce the size of the original image more than using image compression alone.

Table 7-8 : The size of the ROI image using the proposed histogram segmentation to detect ROI specific objects.

ROI detection	ROI automatic detection based on histogram segmentation for specific objects				
	.ROI process			Image compression	
	Input size : A	ROI image size	Reduction size	Compressed image	Compression ratio
1. ROI image	1,154 KB	536 KB	46.45 %	515 KB	44.62 %
2. ROI isolate cloud	1,154 KB	479 KB	41.51 %	470 KB	40.81 %
No ROI detection	1,154 KB	-		652 KB	56.50 %

Clouds in the image are random in shape and can be located anywhere in the image. To perform the manual detection it is difficult to find the ROI area. Therefore it is possible to outline the ROI area and quantify the difference between the proposed system and manual detection. The accuracy of the proposed system based on histogram ROI detection without cloud has been examined by comparing the ROI image identified by the proposed system with that which would be obtained by manual detection. The test procedure is the same as in the earlier section 5.5 system accuracy. Table 7-9 shows the comparison of the automatic ROI detection and the manually detected ROI. The comparison between automatic and manual detection was made three separate times. The percentage difference between the ROI detection methods is on average 3.69 %.

Table 7-9 : The comparison of results between cloud ROI automatic detection and cloud ROI manual detection.

Image and test number		Number of different pixels	Percentage difference	Average percentage
 Image 1 (670 x 893 x 3)	Test 1	21,607	3.61%	3.69%
	Test 2	22,362	3.74%	
	Test 3	22,326	3.73%	

7.3 Comparison of the different ROI segmentation schemes performance for the land, coastal/island, and partially cloudy images

A system based on edge detection to detect the ROI in the satellite images is capable of identifying ocean, coastal and islands easily since there are significant changes of the pixel values between these areas.

An alternative system is based on histogram segmentation. With this method it is possible to identify a specific region of interest in the image. In the example discussed in this research histogram segmentation shows how the forest can be extracted from the input image. The proposed system based on histogram segmentation can be applied to a more complex image and the image can be classified into several groups. Histogram segmentation can also be used to identify and exclude clouds from a ROI image. An examination of this method has demonstrated that high accuracy can be achieved, although the robustness of the method for a range of different types of cloud has not been performed. It is expected that thin, semi-transparent cloud will present a more demanding classification problem. If multiple wavebands are available, parallelepiped classification involving segmentation can be used. In this case the boundaries of each class are represented by pixel values for each waveband. The extra information available provides a more robust method than that based solely on the grey level.

7.4 A hardware design synthesis

A key reason for the method proposed is its potential application on a cubesat. Many cubesat projects have used COTS processors, for example the Atmel ARM7 and PIC16F877 flight module board which were used on the AAUsat-II and CubeSatXI-IV respectively (Teney 2009).

This section presents the proposed hardware design synthesis on one of the proposed systems only, the system based on edge segmentation. There have been numerous implementations of edge detection in real hardware. These have been implemented using C-language or Verilog simulations (Andrej and Baldomir 1999; Vega, Sanchez et al. 2004; Daggu Venkateshwar Rao 2006). A number of studies have aimed to reduce the time and resource consumption associated with the more demanding software processes, e.g. convolution (Chao and Parhi 2004; Vega, Sanchez et al. 2004).

The implementation of a microprocessor based system involves the selection of appropriate hardware followed by the suitable application program software development to

enable the system to work properly on the selected microprocessor. The cost of the hardware for the microcontroller itself is a very small proportion of the overall system development cost. The main portion of the project budget is related to software development. When the software has been developed, simulator programs are used to test the process and a combination of tests are conducted between the software and the hardware to ensure the overall system is compatible (Kochhar A.K. 1983; Harrington 2012).

In the overall system the microcontroller requires interfacing to other microcontrollers or devices. This interfacing needs design control; for example the converter between outputs of a microcontroller to external devices or the transfer data process between the microcomputer and external devices. Consequently the development process requires knowledge of the design circuit, and memory storage (Kochhar A.K. 1983). Accordingly, the system implementation on a multi microcontroller or connection to external devices adds additional time to the total system operation (Kochhar A.K. 1983; Harrington 2012).

In this section, the proposed system hardware design will consider the execution time to operate the whole process. This time to execute is not the full time taken as there is additional time taken for hardware interfacing operations. Nevertheless it will enable a comparison to be made of the techniques employed.

Microprocessors are capable of performing basic arithmetic operations and moving data (Kochhar A.K. 1983). The 'instruction count' in a microprocessor is the measurement of the number of instructions executed whilst the system is processing. The basic operations consist of addition, subtraction, comparison, bit operations and shift operation. Therefore the number of cycles for each operation of multiplication, addition, division and comparison can be different due to the number of clock cycles requested by each instruction being dependent on the hardware device (Bakos 2001; Akers 2007; Altera 2013). Hence various types of instructions take different amounts of time on different hardware devices (Akers 2007).

For example the number of clock cycles for multiplication, addition, division and comparison are 4 clock cycles for each process in a PIC16F877 microcontroller and 1 clock cycle for an AVR Atmega128 (Loukas 2004; Patterson and Hennessy 2009; Atmel 2013; MikroElektronika 2013; Wikimedia 2013). The main points to consider for hardware implementation are energy and time consumption. Time consumption is a measure of a computer's performance and different applications show different performance aspects of a computer system. The execution time must depend on the number of instructions in a program.

In the implementation of the proposed system consisting of ROI automatic detection and wavelet image compression on a cubesat, additional power and processing time will be required. To quantify this increase, a particular proposed system based on edge segmentation for ROI

automatic detection has been studied. This information will support the next step of hardware design and implementation. Based on this information the hardware that is suitable is selected. This research estimates the proposed system energy usage and time consumption by estimating the number of instructions required. The execution time of the proposed system can be calculated by using information about the speed of the processor and the total number of instructions.

7.4.1 The hardware design for the proposed system based on edge segmentation

The proposed system consists of many evaluation processes as illustrated in Figure 7-5. To identify the number of calculation operations in the processes, the data flow of each process requires clarification and this is presented in the form of data flow diagrams. The number of calculation units in each process has been recorded and analysed as well as the approximate total calculation units in the whole system.

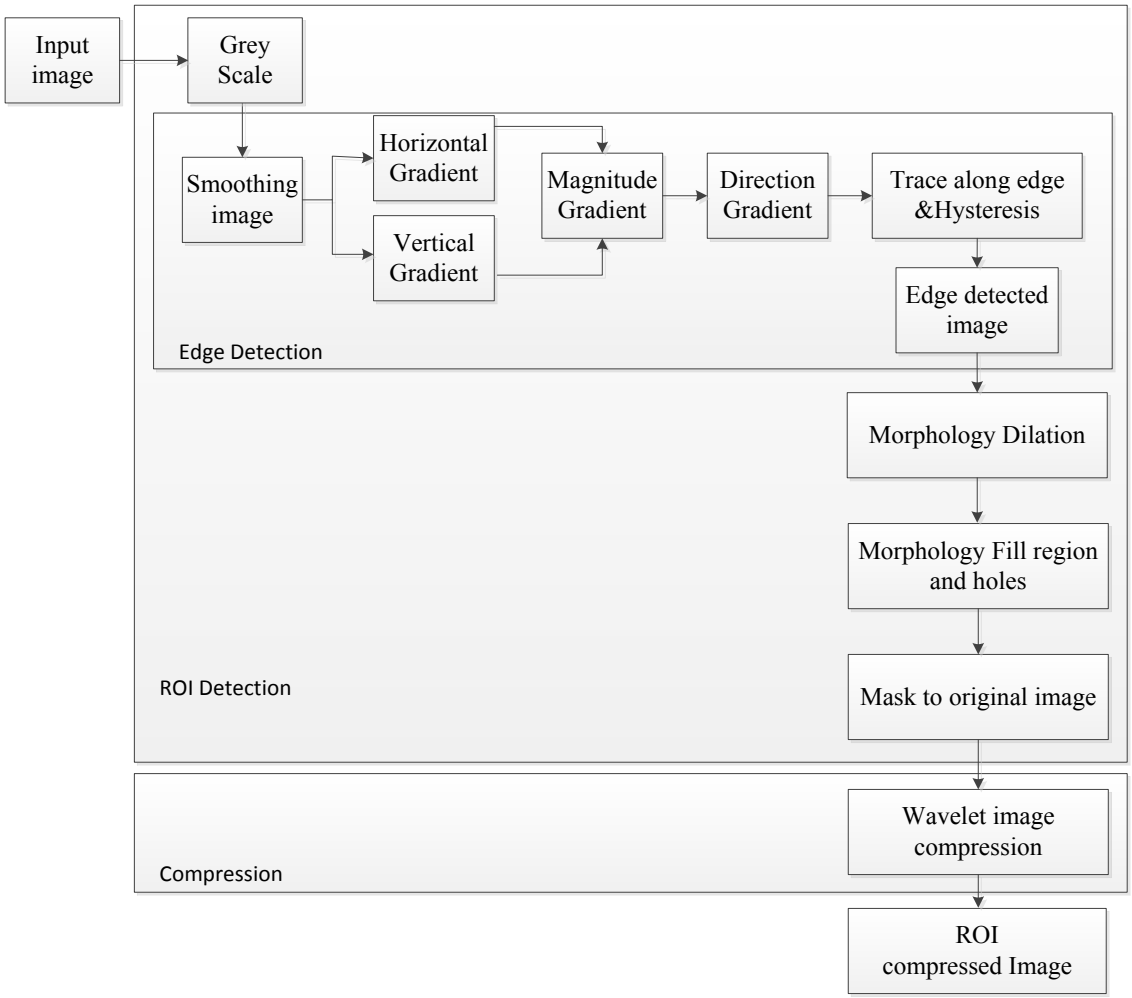


Figure 7-5 : Block diagram of the proposed system based on edge detection segmentation.

The analysis has been focused on edge segmentation but an equivalent analysis can be performed for other techniques. The hardware data flows are demonstrated and are given in more detail in sub topics in the following.

7.4.1.1 Hardware design for ROI detection processing based on edge detection

7.4.1.1.1 Grey scale image

To convert the RGB to grey scale image, equation 5.1 is applied. The process consists of multiplication and addition operations to perform this conversion and the data flow is shown in Figure 7-6. From this data flow, the number of operations can be counted to estimate the total amount of operations step by step.

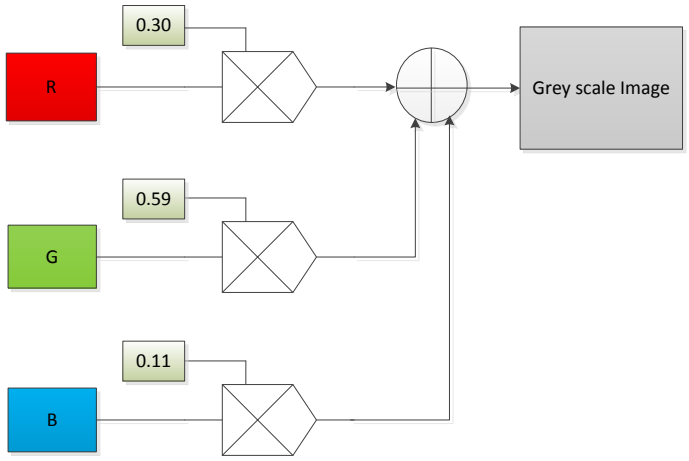


Figure 7-6 : The conversion of RGB to grey scale image.

7.4.1.1.2 Gaussian smoothing

In general an image will contain noise which has to be reduced to minimise the possibility of noise being mistaken for edges. The image is often smoothed by applying a Gaussian filter. Figure 7-7 demonstrates the mask used for the Gaussian filter. It is a size of 5x5 pixels, which is an appropriate integer valued convolution mask that approximates a Gaussian function with a standard deviation of 1.4 (Singh 2009). P_{ij} represents the position of the pixel in the image P. Smoothing of the image is performed by a convolution process between the input image (P_{ij})

and the Gaussian mask (G). The main operation is convolution which consists of multiplication and addition operation units. The convolution process can defined by symbol '∗'. For example the convolution process by G mask on P_{ij}, this can be described by equation 7-1.

$$(P_{ij} * G) = \sum_{j=0}^{N-1} P_{ij} G \quad , \qquad k = 0, 1, 2, \ldots\ldots\ldots \qquad (7\text{-}1)$$

For the equation the computation process consists of the multiplication between P_{ij} and G. Moreover there is the summation of each multiplication result. Consequently, number of multiplications and summations can be counted.

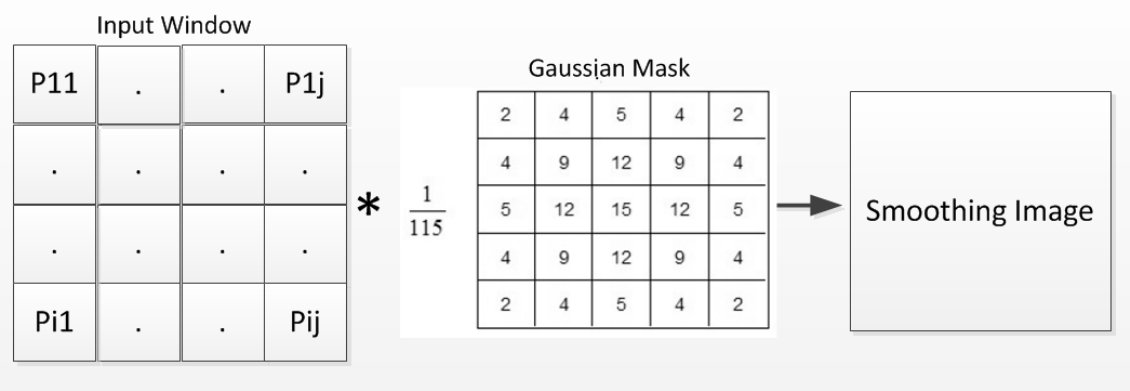


Figure 7-7 : The data flow of image smoothing by Gaussian mask.

7.4.1.1.3 Gradient magnitude

Generally, an edge is defined as the boundary between two regions where the grey scale intensity has a large change. These edges are found by determining gradients of the image. A horizontal and vertical convolution mask is applied to the smoothed image to define the gradient in the x-direction and y-direction respectively. The gradient masks in the x- and y- direction; d_x and d_y respectively are shown in Figure 7-8. This provides the approximate gradient in x- and y-direction and the gradient magnitude can then be determined from equation 5.3. To reduce the processing time, the gradient is approximated with the absolute sum of the horizontal and vertical gradient ($|d| = |d_x| + |d_y|$) (Daggu Venkateshwar Rao 2006; Ponneela and Rajendran 2012).

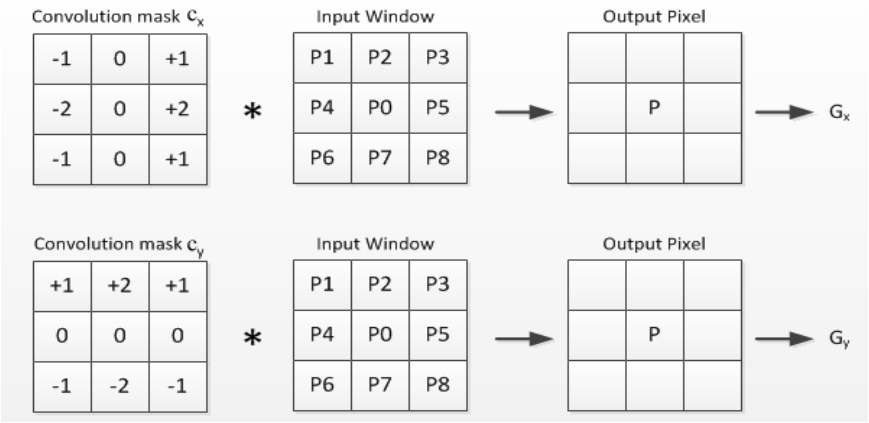


Figure 7-8 : The data flow of the gradient image in x-direction (a) and y- direction (b).

7.4.1.1.4 Gradient orientation

The direction of the edges is given by equation 5.4. Implementing this function in hardware is difficult because it is a very complex function. Instead, to calculate the arctangent, the gradient direction is analysed by approximation to one of the gradient orientation sectors (Daggu Venkateshwar Rao 2006). For example in the input window in Figure 7-8 at the position of pixel *'P0'*, there are eight directions surrounding the pixel (0, 45, 90, 135, 180, 225, 270 and 315 degrees). The pixel is defined in to these directions depending on which classification the pixel belongs to. For example if the size of d_x is higher than the size of d_y and both have positive values, then the pixel falls in an arctangent region between 0 - 45 degrees as illustrated in Figure 7-9. Consequently, the data flow for hardware design consists of comparison and selection as shown in Figure 7-10. The selection of the direction are defined by for example a, b, c and d. The process will select the direction from a, b, c and d according to the section of the gradient information.

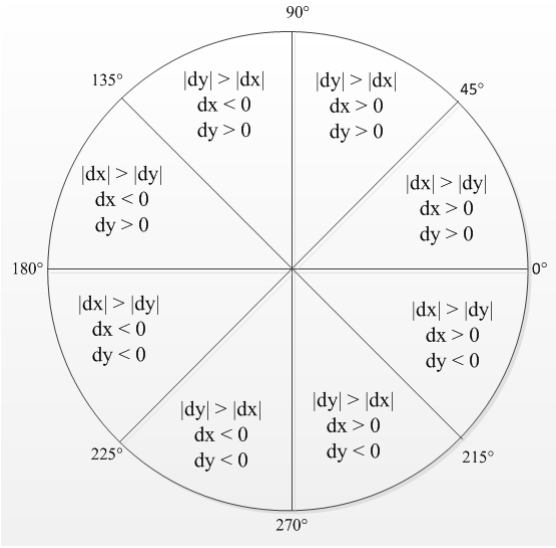


Figure 7-9 : The gradient orientation sectors.

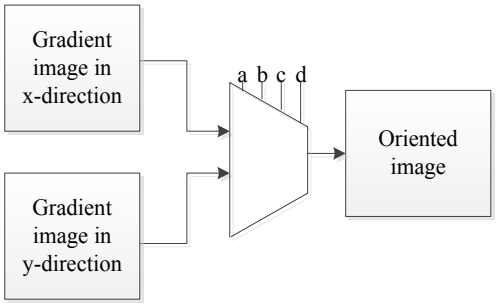


Figure 7-10 : The data flow of hardware gradient orientation.

7.4.1.1.5 Trace along the edge and hysteresis

The next step is finding the edge by non-maximum suppression which is used to suppress any pixel value that is not considered to be an edge by tracing along the edge direction. There are likely to be numerous false edges (noise, colour variations) that are not true edges, so the thresholding technique is applied to identify those that are strong edges. Canny edge detection algorithm uses double thresholding, which consists of high threshold and low thresholding. Edge pixels are defined as 'strong edges' or 'weak edges' depending on the high and low thresholding. Edge pixels which lie above the high threshold are designated as 'strong edges', those between the high and low thresholds as 'weak edges' and those that line below the low

threshold as ‘suppressed’. The next step is hysteresis. This process classifies the strong and weak edges. The strong edges definitely transfer to be the final edge. Nonetheless, some weak edges are included with the final edges if they connect to the strong edges. The process showing the data flow is presented in Figure 7-11.The selection of the final edge defined by for example e, f, g and h. The process will select the final edges from e, f, g and h according to the comparison information.

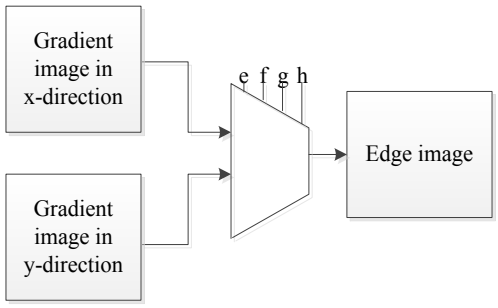


Figure 7-11 : Trace along the edge and hysteresis.

7.4.1.1.6 Morphology dilation

The dilation is performed by the maximum filter operation which is implemented by sliding a window of a structure element over an image. The centre pixel is replaced by the maximum value of the pixel in the window. The structure element is the main parameter used in the morphological techniques for image processing and affects the resultant final image. The morphology structure element has different shapes such as square, diamond, disk and line. Figure 7-12 shows the morphology dilation data flow, performed by convolution operation using a structure element square shape of size 3x3 pixels.

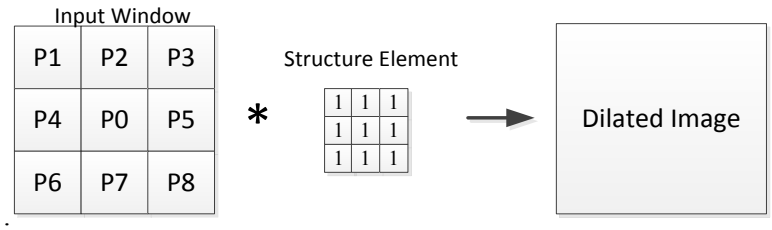


Figure 7-12 : The morphology dilation data flow.

7.4.1.1.7 Morphology fill the regions and holes

This process consists of convolution operations and another three basic operations; dilation, complementation and intersection for filling holes in an image. The process can be described by equation 7.1 which fills all the holes with 1's. Figure 7-13 illustrates the process of filling the regions and holes method (Gonzalez and Woods 2007). Figure 7-13(a) is '*A'*, which represents an image that has holes that need filling. Figure 7-13 (b) shows (A^c), which is the complement of '*A*' and *'B'* is the symmetric structure element (Figure 7-13 (c)). The process starts with a point inside the boundary, setting this point to be '1' as shown in Figure 7-13 (d) beginning with the point X_0. The next step follows the procedure according to equation 7.1 to calculate X_1(Gonzalez and Woods 2007). Figure 7-13 (e-h) are various steps of equation 7.1.

$$X_k = \left(X_{k-1} \oplus B\right) \cap A^c \qquad k = 1, 2, 3\ldots \qquad (7.1)$$

Subsequently the output X_k intersects with '*A'* to cover all boundaries and fill all holes as shown in Figure 7-13 (i) (X_7 is intersected with 'A'). The result of this is the ROI mask image.

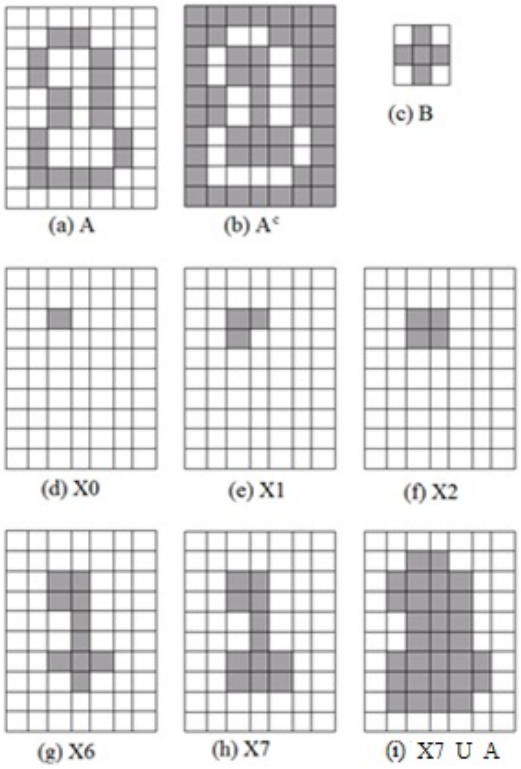


Figure 7-13 : Morphology image region and hole filling process (Gonzalez and Woods 2007).

7.4.1.1.8 Mask to the original image

This process compares the mask image from the previous step to the original image. If any pixel of the original image is in the same position as the ROI mask's value of '1', this final image position is assigned with the value of the original image. Otherwise, the pixel will be assigned as '0'. The data flow of this process is shown in Figure 7-14.

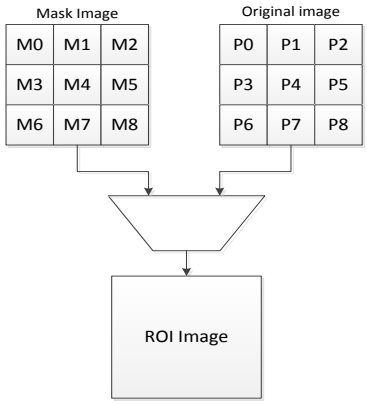


Figure 7-14 : Comparison of mask image to original image data flow.

7.4.1.2 Hardware design for image wavelet compression

The discrete wavelet transform is performed by convolution based implementation using the Finite Impulse Response (FIR) filters. The input image is filtered by a low-pass filter (*L*) and a high-pass filter (*H*) at each transform level. Figure 7-15 shows the three level wavelet decomposition of a sample image of size 16x16. The process performs the convolution between input image and the wavelet filters.

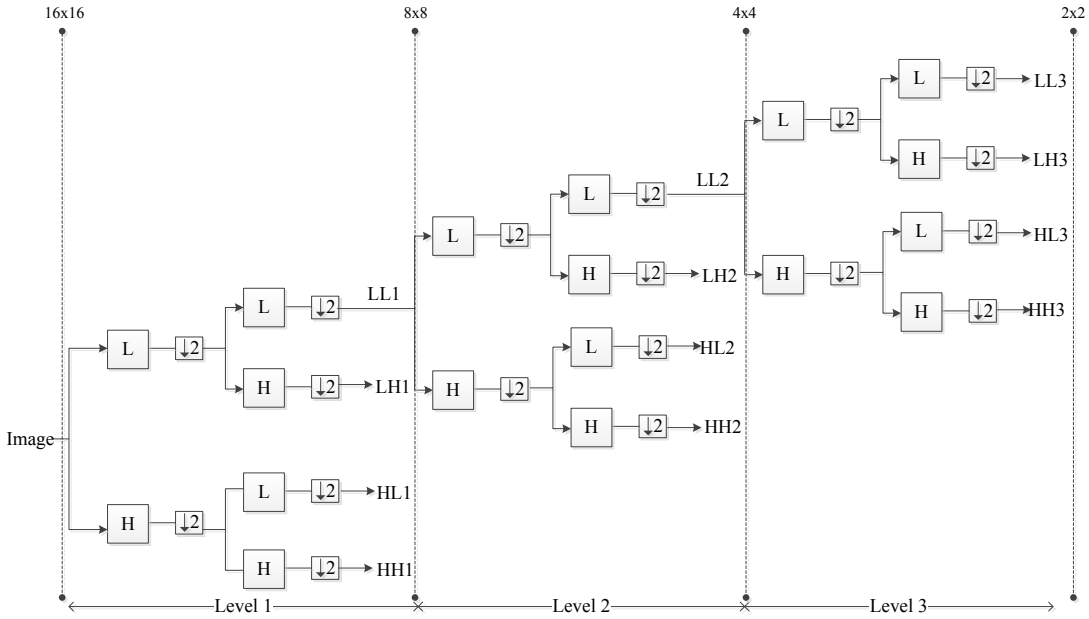


Figure 7-15 : The three level wavelet decomposition of a sample image of size 16x16.

To perform image wavelet decomposition on a 16x16 pixel image by using 9/7 orthogonal, a number of operation clock cycles are required during the convolution calculation. The convolution is performed by sliding the kernel or mask over the image. Figure 7-18 shows a calculation of the convolution process of image row-1 (16 pixels) with the 9 highpass filters. The convolution mask (filters) slides from the first pixel (X0) until the last pixel (X15) is convolved with the last filter. For example, the kernel slides through image pixel X3, at this stage there are four multiplications from X3*h0, X2*h1, X1*h2 to X0*h3. Also there are three additional sets of those multiplication results. Therefore there are 144 multiplication and 120 addition operations to perform for this convolution with the 9 highpass filter as shown in Figure 7-16. The next step is the convolution between the image and other filters according to the three level wavelet decomposition. The process will carry on until the final sub-band (LL) is performed.

X0	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	Multiplication	Addition
h0																1	0
h1	h0															2	1
h2	h1	h0														3	2
h3	h2	h1	h0													4	3
h4	h3	h2	h1	h0												5	4
h5	h4	h3	h2	h1	h0											6	5
h6	h5	h4	h3	h2	h1	h0										7	6
h7	h6	h5	h4	h3	h2	h1	h0									8	7
h8	h7	h6	h5	h4	h3	h2	h1	h0								9	8
	h8	h7	h6	h5	h4	h3	h2	h1	h0							9	8
		h8	h7	h6	h5	h4	h3	h2	h1	h0						9	8
			h8	h7	h6	h5	h4	h3	h2	h1	h0					9	8
				h8	h7	h6	h5	h4	h3	h2	h1	h0				9	8
					h8	h7	h6	h5	h4	h3	h2	h1	h0			9	8
						h8	h7	h6	h5	h4	h3	h2	h1	h0		9	8
							h8	h7	h6	h5	h4	h3	h2	h1	h0	9	8
								h8	h7	h6	h5	h4	h3	h2	h1	8	7
									h8	h7	h6	h5	h4	h3	h2	7	6
										h8	h7	h6	h5	h4	h3	6	5
											h8	h7	h6	h5	h4	5	4
												h8	h7	h6	h5	4	3
													h8	h7	h6	3	2
														h8	h7	2	1
															h8	1	0
																144	120

Figure 7-16 : First row image convolution with high pass filter process and summary of multiplication and addition operation requests.

To generate HH sub-band at the first level decomposition, the convolution process requires 144 x 2 multiplications and 120 x 2 additions to operate. Table 7-10 shows the number of instruction operations to generate sub-bands in all three levels of wavelet decomposition. To finish the wavelet decomposition in three levels, 1,792 multiplication and 1,400 addition operations are required for an image size of 16x16. The number of operations is dependent on the image size so larger image sizes require more operations to perform the process.

Table 7-10 : Number of instruction operations for three levels of wavelet decomposition of image size 16x16.

Decomposition level	Instruction operation						
	Sub-band	Image size	Multiplication	Addition	Image size	Multiplication	Addition
Level-1	HH	16	144	120	8	144	120
	HL	16	144	120	8	112	90
	LH	16	112	90	8	144	120
	LL	16	112	90	8	112	90
Level-2	HH2	8	72	56	4	72	56
	HL2	8	72	56	4	56	42
	LH2	8	56	42	4	72	56
	LL2	8	56	42	4	56	42
Level-3	HH3	4	36	24	2	36	24
	HL3	4	36	24	2	28	18
	LH3	4	28	18	2	36	24
	LL3	4	28	18	2	28	18
Total operations			896	700	-	896	700

Using these data flows for each sub-process, consisting of eight sub-processes in image edge detection technique and the wavelet image decomposition, the total number of operations has been calculated. The test has been applied to nine different image sizes in order to show the relationship between the size of the image and the number of instruction counts.

Table 7-11 shows the calculation of the total number of instruction counts for an image size of 16x16. Multiplication, addition, division and comparison instruction counts are 12,546, 5,738, 27 and 1,280 respectively. This data will be used to identify the approximate power usage and time consumption of microcontrollers in the next step.

The *CPI* is the 'cycles per instruction', which is defined by the following equation 7-2.

$$CPI = {\Sigma(X)(Y)}/{Total\ instruction\ count} \qquad (7\text{-}2)$$

Where X is the number of instructions and Y is the clock cycles for a given instruction type.

Table 7-11 : The number of instruction counts for the proposed image edge detection algorithm.

Process	Multiplication	Addition	Division	Comparison
1.RGB Pixels 16 x 16 = 256	768	512		
2.Edge 1. Smoothing				
mask size = 25				
Smoothing	1280		25	
2. Gradient				
mask size X,Y= 5				
Convolution A	1280	256		
Convolution B	1280	256		
|A|	1	256	1	
|B|	1	256	1	
|G|		2		
3. Edge direction				
Comparison				256
4. Non maxima and				
Comparison				256
3.Dilation SE size = 15				
Convolution	1280			
4.Fill regions and holes				
SE size = 15				
Convolution	1280			
Complement A				256
Union				256
5.Apply on original image				
Comparison				256
Total for edge segmentation	7170	1538	27	1280
6.Image wavelet compression	5376	4200		
Overall total	12546	5738	27	1280

Table 7-12 shows the number of clock cycles to operate the proposed system in different image sizes. The energy usage and time to execute are also shown. Each instruction requires 4 clock cycles to operate as the specification of microcontroller PIC16F877 (Loukas 2004; Atmel 2006; MikroElektronika 2013), which operates at 20 MHz (Patterson and Hennessy 2009). David and Patterson (Patterson and Hennessy 2009) present the calculation of the execution time usage which is shown in equation 7-3.

$$\text{Executiontime} = \text{TotalinstructionxCPI}/\text{Microcontrollerclock} \quad (7\text{-}3)$$

The execution time of the proposed system to process the different sizes of input images are shown in Table 7-12. Consequently, an estimation of the energy consumption can be

calculated. The energy is calculated from the specification of microcontroller power usage at the operation frequency multiplied by the total execution time. For example for an image size of 2048x1536, the total number of clock cycles to operate the proposed system is around 444.27 million instructions. By using PIC 16F877, the system uses an energy of 1,110.68 mJ over 22.21 seconds.

If the proposed system is implemented on a Atmega128 microcontroller, which requests one clock cycle to operate each instruction at 1 MHz (Loukas 2004; Atmel 2006; Atmel 2013; Wikimedia 2013), the total number of clock cycles to operate the proposed system is around 111.07 million instructions and the system uses energy at 366.52 mJ over 11.11 seconds as the result of the calculation is in Table 7-13.

Table 7-12 : The total number of clock cycles, energy usage and time consumption to operate the proposed system in different image sizes on a PIC16F877 microcontroller.

Image size	Number of clock cycles to operate the proposed system				Total number of clock cycles	Microcontrollers PIC16F877 50 mW @ 20 MHz	
	Multiplication (4 clock cycle)	Addition (4 clock cycle)	Division (4 clock cycle)	Comparison (4 clock cycle)		Time (ms)	Energy (mJ)
1. 16x16	50,184	22,952	108	5,120	78,364	3.92	0.20
2. 32x32	141,320	60,200	108	20,480	222,108	11.11	0.56
3. 64x64	479,240	171,560	108	20,480	671,388	36.64	1.83
4. 128x128	1,744,904	541,736	108	81,920	2,368,668	130.72	6.54
5. 256x256	6,586,376	1,828,904	108	1,310,720	9,726,108	486.31	24.36
6. 512x512	25,706,504	6,762,536	108	5,242,880	37,712,028	1,885.60	94.28
7. 1024x1024	101,695,496	26,066,984	108	20,971,520	148,734,108	7,436.71	371.84
8. 2048x1536	304,054,280	77,301,800	108	62,914,560	444,270,748	22,213.54	1,110.68
9. 2048x2048	404,717,576	102,467,624	108	83,886,080	591,071,388	29,553.57	1,477.68

Table 7-13 : The total number of clock cycles, energy usage and time consumption to operate the proposed system in different image sizes on an AVR Atmega128 microcontroller

Image size	Number of clock cycles to operate the proposed system				Total number of clock cycles	Microcontrollers AVR Atmega128 9.9mW @1 MHz	
	Multiplication (1 clock cycle)	Addition (1 clock cycle)	Division (1 clock cycle)	Comparison (1 clock cycle)		Time (ms)	Energy (mJ)
1. 16x16	12,546	5,738	27	1,280	19,591	19.59	0.07
2. 32x32	35,330	15,050	27	5,120	55,527	55.53	0.18
3. 64x64	119,810	42,890	27	20,480	183,207	183.21	0.61
4. 128x128	436,226	135,434	27	81,920	653,607	653.61	2.16
5. 256x256	1,646,594	457,226	27	327,680	2,431,527	2,431.53	8.02
6. 512x512	6,426,626	1,690,634	27	1,310,720	9,428,007	9,428.01	31.11
7. 1024x1024	25,423,874	6,516,746	27	5,242,880	37,183,527	37,183.52	122.71
8. 2048x1536	76,013,570	19,325,450	27	15,728,640	111,067,687	11,1067.69	366.52
9. 2048x2048	101,179,394	25,616,906	27	20,971,520	147,767,847	147,767.85	487.63

Figure 7-17 show the relationship between the images in different sizes with time consumption to operate the proposed system by using PIC16F877 and AVR Atmega128 microcontroller. Moreover Figure 7-18 shows the relationship between the images in different sizes with the energy that required processing the proposed system on PIC16F877 and AVR Atmega128 microcontroller. The result shows that the time and energy consumption increased according to a larger image size.

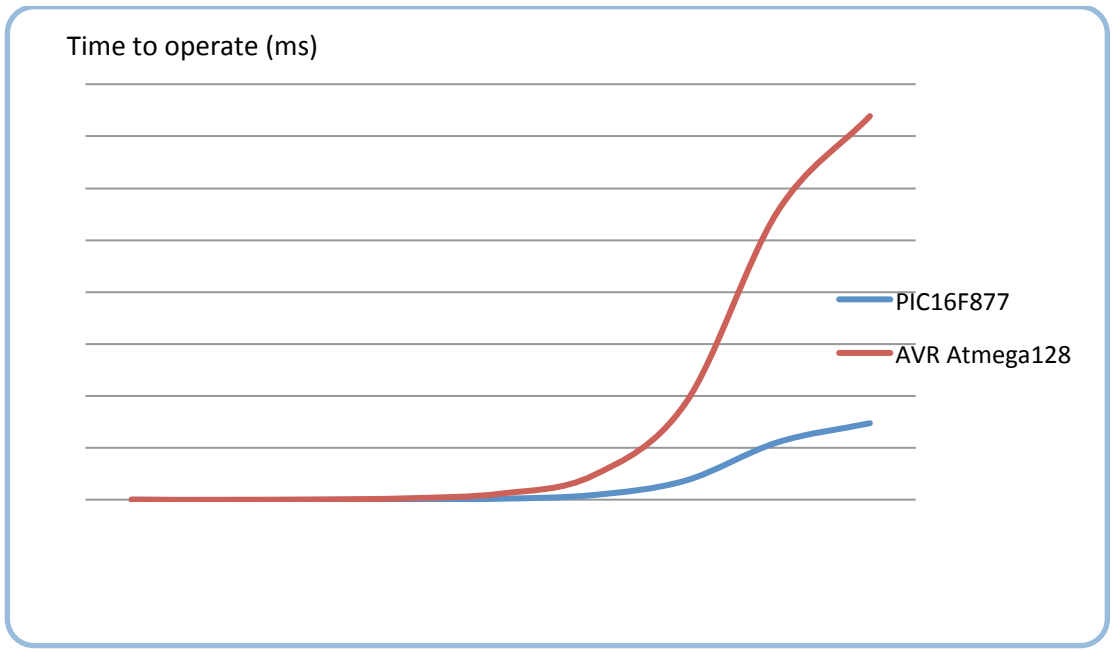


Figure 7-17 : The relationship between input image size and time consumption.

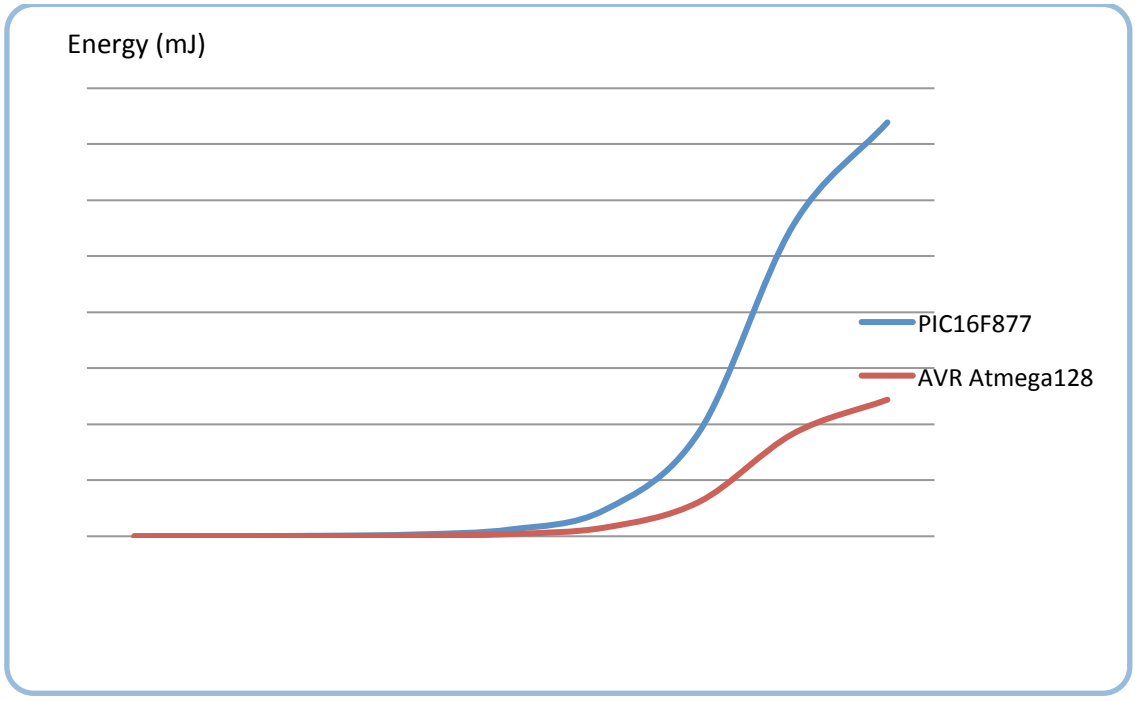


Figure 7-18 : The relationship between input image size and energy required.

The convolution operation has been used in many sub-processes in the proposed system. However it requires a considerable number of computation processes. It can also be seen that the number of operations is dependent on the size of the input image. A single cycle

multiplication requires a large amount of hardware and long delays through complicated logic. A design is necessary that offers reduction in the hardware required and the processing time to enable reduced power consumption but maintain performance (Daggu Venkateshwar Rao 2006).

7.4.2 The hardware design for the propose system based on histogram segmentation

The proposed automatic detection systems have been based on edge, histogram and texture segmentation. There are some processes which are common in all three systems, e.g. the morphological dilation, the morphological fill region and hole image, black and white image conversion and ROI management. In the following sections the computational load of the proposed histogram and texture segmentation systems have been evaluated.

The proposed system based on histogram segmentation consists of many processes as illustrated in Figure 7-19. To identify the number of calculations required in each process, the data flow in each process has been examined. The number of calculations required in each process has been recorded and analysed as well as the approximate total number of calculations for the whole system based on histogram segmentation.

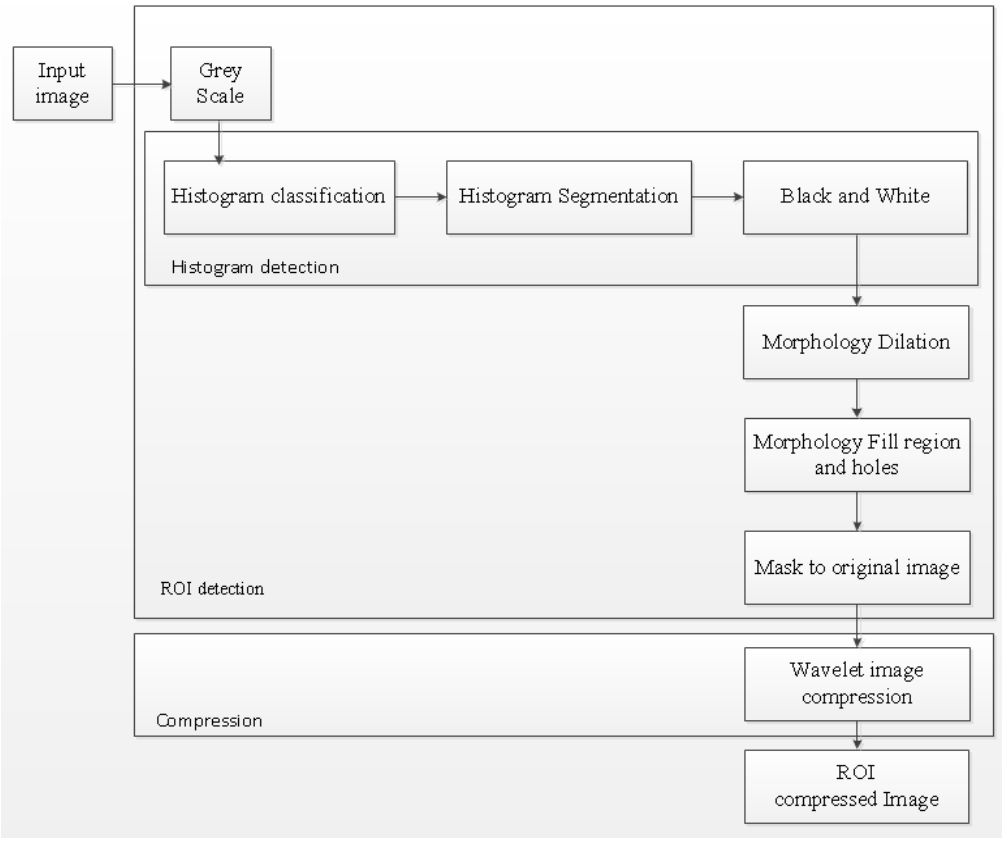


Figure 7-19 : Block diagram of the proposed system based on histogram detection segmentation.

7.4.2.1 Hardware design for ROI detection processing based on histogram processing

7.4.2.1.1 Grey scale image

This process consists of multiplication and addition operations to perform this conversion. The hardware data flows are the same as described in section 7.4.1.1.1.

7.4.2.1.2 Histogram classification and segmentation

This process separates the objects in the image into the different groups according to the intensity of the pixel. There are two parts to this section, histogram classification and histogram segmentation.

Histogram classification identifies the threshold values for each group. This threshold is used to classify each pixel into the appropriate group. In this research the threshold classification is defined manually.

The second part is histogram segmentation. This research classifies the objects by labelling into four groups; ocean, urban, forest and agriculture by using thresholds set from the first process. The process of classification consists of the comparison of the value of each pixel and the value set of the thresholds. Figure 7-20 shows this comparison with w, x, y and z, the value of the set threshold of each group of object. Each pixel is labelled by the colour designated to a particular class.

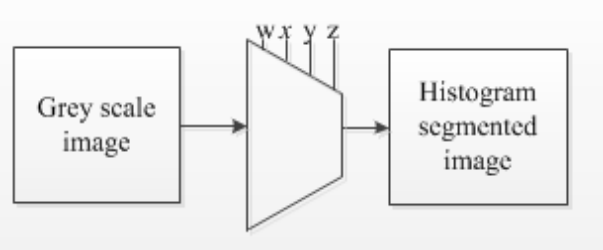


Figure 7-20 : The data flow of the histogram segmentation.

7.4.2.1.3 Black and white image

This process converts the histogram segmented image into a binary image. The image has its pixels replaced by either a black or white colour depending on whether the pixel is in a classified category in a region of interest. Figure 7-21 shows the conversion of the histogram segmented image to a black and white image. This process consists of the comparison of the selected pixel value with the black and white threshold.

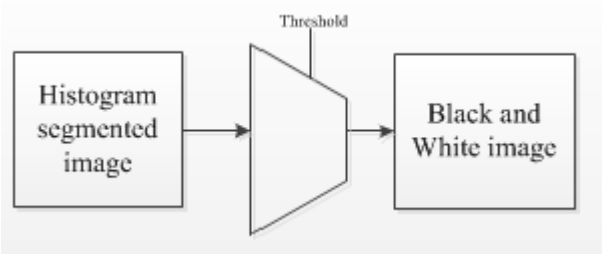


Figure 7-21 : The data flow of the black and white image processing.

7.4.2.1.4 Morphology dilation, filling and masking

The morphology dilation, morphology filling and masking are the same as that described in section 7.4.1.1.6, 7.4.1.1.7 and 7.4.1.1.8 respectively in the thesis.

7.4.2.2 Hardware design for image wavelet compression

The hardware data flow for this process is the same as that described in section 7.4.1.2. Table 7-14 shows the calculation of the total number of instruction counts for an image size of 16x16 pixels. Multiplication, addition, and comparison instruction counts are 8704, 4712 and 1,280 respectively. This data will be used to identify the approximate power usage and time consumption of microcontrollers in the next step.

Table 7-14 : The number of instruction counts for the proposed image histogram detection algorithm (an image size of 16x16 pixels).

Process					Multiplication	Addition	Division	Comparison
1. Grey scale image	Pixels	16	16	256	768	512		
2. Histogram classification and Histogram segmentation								
Comparison		256						256
3. Black and White image								
Comparison		256						256
4. Dilation								
SE size = 5		5						
Multiplication		1280			1280			
5. Fill regions and holes								
SE size = 5		5						
Convolution		1280			1280			
Complement A		256						256
U		256						256
6. Apply on original image								
Comparison		256						256
Total for edge segmentation					3328	512	0	1280
7. Image wavelet compression					5376	4200	0	0
Overall total					8704	4712	0	1280

Table 7-15 shows the number of clock cycles to derive the histogram segmentation system for different image sizes. The table shows the execution time of the proposed system to process the different sizes of input images and provides an estimate of the energy consumption. For example, for an image size of 2048x1536, the total number of clock cycles to operate the proposed system is around 255 million instructions. By using PIC 16F877, the system requires an energy of 638.82 mJ over 12.78 seconds.

If the proposed system is implemented on an Atmega128 microcontroller, which requests one clock cycle to operate each instruction at 1 MHz, the total number of clock cycles to operate the proposed system is around 15.72 million instructions and the system requires energy of 210.81 mJ over 63.88 ms. This is shown in Table 7-16.

Table 7-15 : The total number of clock cycles, energy usage and time consumption to operate the proposed system in different image sizes on a PIC16F877 microcontroller.

Image size	Number of clock cycles to operate the proposed system using 4 clock cycle for an instruction				Total number of clock cycles	Microcontroller	
	Multiply (4)	Plus (4)	Divide (4)	Compare (4)		PIC 16F877 50 mW @ 20 MHz	
						Time (ms)	Energy (mJ)
1. 16x16	34,816	18,848	0	5,120	58,784	2.94	0.15
2. 32x32	96,256	43,808	0	20,480	160,544	8.03	0.41
3. 64x64	299,008	106,016	0	20,480	425,504	24.35	1.22
4. 128x128	1,024,000	279,584	0	81920	1,385,504	81.56	4.08
5. 256x256	3,702,784	780,320	0	1,310,720	5,793,824	289.69	14.48
6. 512x512	14,172,160	2,568,224	0	5,242,880	21,983,264	1,099.16	54.96
7. 1024x1024	55,558,144	9,289,760	0	20,971,520	85,819,424	4,290.97	214.55
8. 2048x1536	165,642,240	26,970,144	0	62,914,560	255,526,944	12,776.35	638.82
9. 2048x2048	220,168,192	35,358,752	0	83,886,080	339,413,024	16,970.65	848.53

Table 7-16 : The total number of clock cycles, energy usage and time consumption to operate the proposed system in different image sizes on an AVR Atmega128 microcontroller

Image size	Number of clock cycles to operate the proposed system by using 1 clock cycle for an instruction				Total number of clock cycles	Microcontroller	
						AVR Atmega128 9.9 mW@ 1 MHz	
	Multiply (1)	Plus (1)	Divide (1)	Compare (1)		Time (ms)	Energy(mJ)
1. 16x16	8,704	4,712	0	1,280	14,696	14.7	0.05
2. 32x32	24,064	10,952	0	5,120	40,136	40.14	0.13
3. 64x64	74,752	26,504	0	20,480	121,736	121.74	0.41
4. 128x128	256,000	69,896	0	81,920	407,816	407.82	1.35
5. 256x256	925,696	195,080	0	327,680	1,448,456	1,448.46	4.78
6. 512x512	3,543,040	642,056	0	1,310,720	5,495,816	5,495.82	18.14
7. 1024x1024	13,889,536	2,322,440	0	5,242,880	21,454,856	21,454.86	70.8
8. 2048x1536	41,410,560	6,742,536	0	15,728,640	63,881,736	63,881.74	210.81
9. 2048x2048	55,042,048	8,839,688	0	20,971,520	84,853,256	84,853.26	280.02

Figure 7-22 shows the relationship between the images of different sizes and the time consumption using the PIC16F877 and AVR Atmega128 microcontroller.

Figure 7-23 shows the relationship between images of different sizes and the energy required for the proposed system on the PIC16F877 and AVR Atmega128 microcontroller. As expected the time and energy consumption increase as the size of the image increases. In this calculation, the size of images are not varying in linearly and the processing time and energy are a function of the number of pixels.

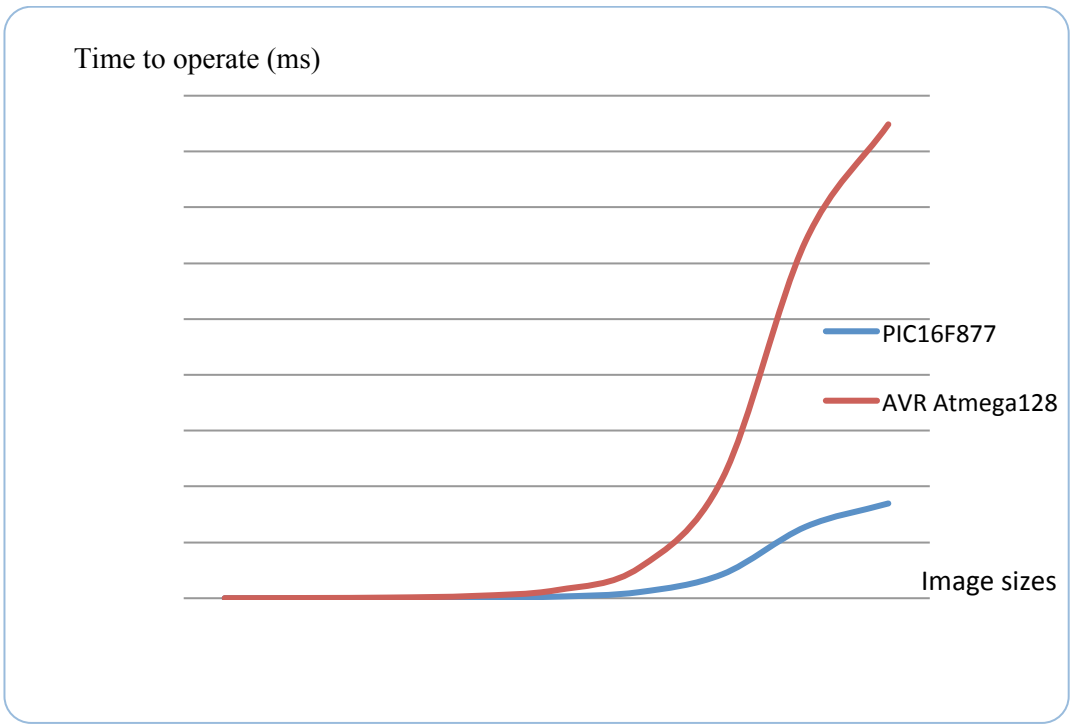


Figure 7-22 : The relationship between input image size and time consumption.

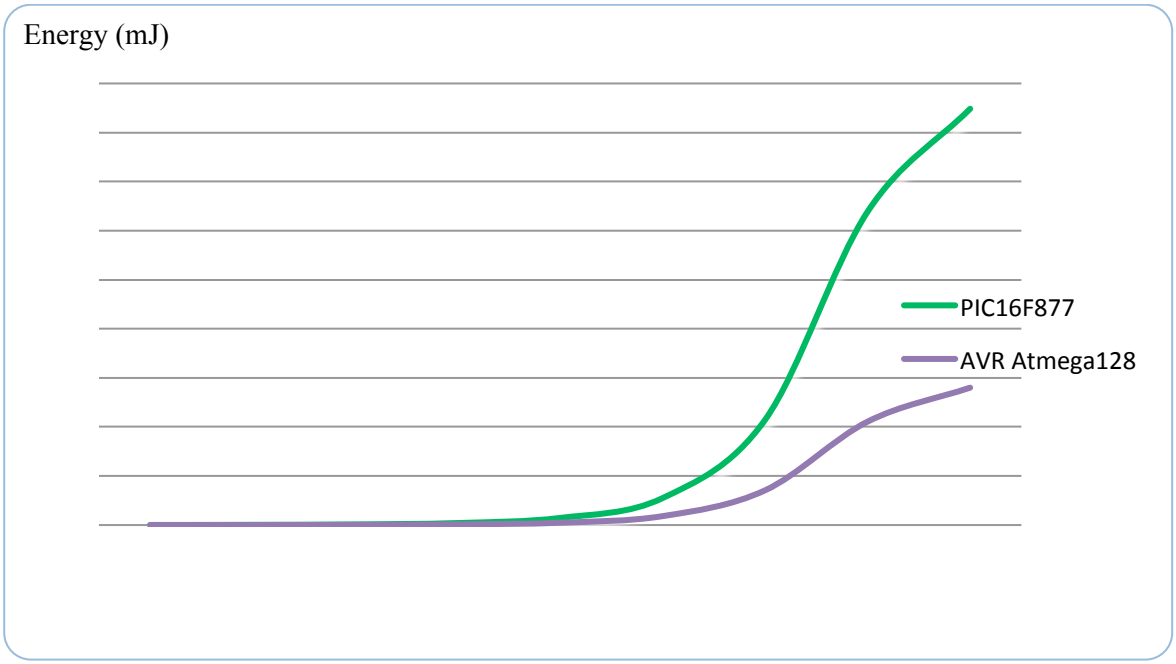


Figure 7-23 : The relationship between input image size and energy required.

7.4.3 The hardware design for the proposed system based on Texture segmentation

Figure 7-24 shows the block diagram for the ROI identification based on texture segmentation

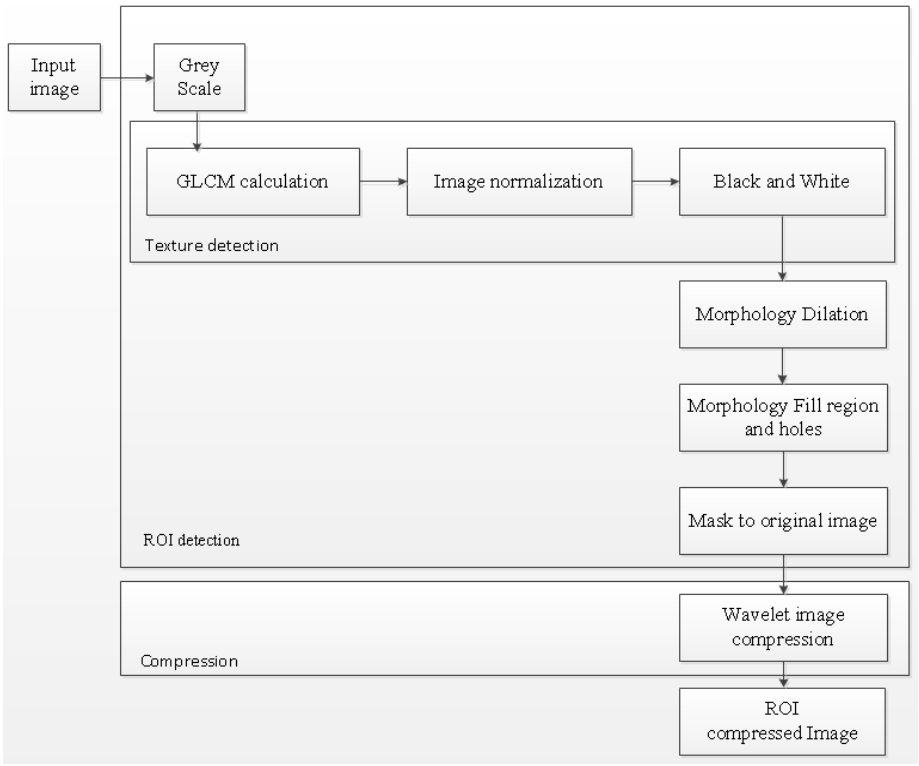


Figure 7-24 : Block diagram of the proposed system based on texture detection segmentation.

7.4.3.1 The hardware design for ROI process based on texture segmentation

7.4.3.1.1 Grey scale image

The hardware data flows are the same process as mentioned in section grey scale image above.

7.4.3.1.2 GLCM calculation

This process calculates the GLCM matrix. The GLCM is calculated by how often the pairs of pixel with specific values and in a specified spatial relationship occur in an image. This matrix will later extract the statistical measures. The process consists of the comparison between the intensity value of each pixel by pixel and the sum of the number of times that the pixel with value *"i"* occurred in the spatial relationship to a pixel with value "*j*" in the input image. Figure 7-25 shows the data flow of this process to calculate the GLCM matrix.

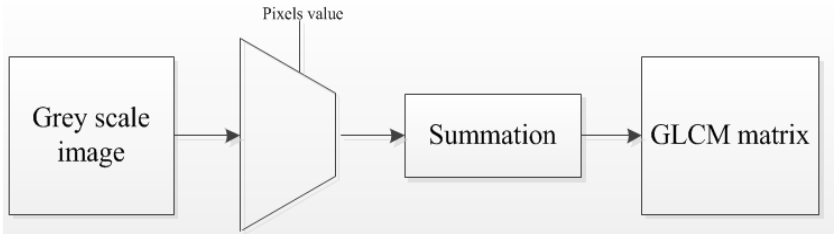


Figure 7-25 : The data flow of the GLCM calculation.

7.4.3.1.3 Image normalisation

This process calculates the statistics specified in feature from the GLCM matrix. This process normalizes the GLCM, hence the sum of its element is equal to 1. Each element in the normalised GLCM matrix is joint probability occurrence of pixel pairs with a defined spatial relationship having grey level values in the image. The result from chapter 5 section 5.3 shows that the energy feature possible to distinguish the ROI from the input image. Figure 7-26 shows the data flow of the energy calculation.

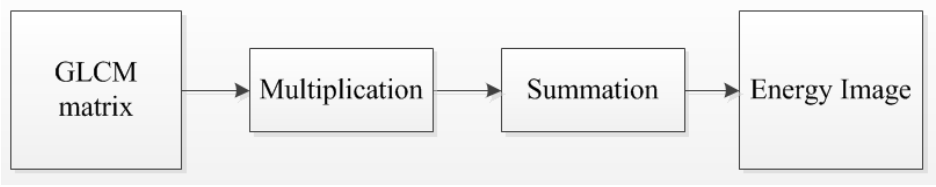


Figure 7-26 : The data flow of the energy image calculation.

7.4.3.1.4 Black and white image

The black and white image conversion hardware data flow had been described in section black and white image above.

7.4.3.1.5 Morphology dilation, filling and masking

The morphology dilation, morphology filling and masking are the same as that described in section 7.4.1.1.6, 7.4.1.1.7 and 7.4.1.1.8 respectively.

7.4.3.2 Hardware design for image wavelet compression

The hardware data flow for wavelet compression had been described in section 7.4.1.2. Table 7-17 shows the calculation of the total number of instruction counts for an image size of 16x16. Multiplication, addition, and comparison instruction counts are 134,400, 66,048 and 1,024 respectively. This data will be used to identify the approximate power usage and time consumption of microcontrollers in the next step.

Table 7-17 : The number of instruction counts for the proposed image based on texture detection algorithm (an image size of 16x16 pixels).

Process					Multiplication	Addition	Division	Comparision
1. Grey Scale	Pixels	16	16	256	768	512		
2. GLCM calculation								
	Comparision	65536			65536			
	Summation	32768				32768		
3. Normalisation								
	Multiplication	65536			65536			
	Summation	32768				32768		
4. Black and white conversion								
	Comparision	256						256
5. Dilation								
	SE size = 5	5						
	Multiplication	1280			1280			
6.Fill regions and holes								
	SE size = 5	5						
	Convolution	1280			1280			
	Complement A	256						256
	U	256						256
7. Apply on original image								
	Comparision	256						256
Total for texture segmentation					134400	66048	0	1024
8. Image wavelet compression					5376	4200		
Overall total					134400	66048	0	1024

Table 7-18 shows the number of clock cycles to operate the proposed system based on texture segmentation in different image sizes. The table shows the execution time of the proposed system to process different sizes of input images. Moreover, an estimation of the energy consumption is presented. For example for an image size of 2048x1536, the total number of clock cycles to operate the proposed system is around 1.19×10^{14} instructions. By using PIC 16F877, the system uses 29,7642.34 Joules over 5.9 million seconds.

If the proposed system is implemented on an Atmega128 microcontroller, the total number of clock cycles to operate the proposed system is around 2.98×10^{13} instructions and the

system uses an energy of 98,221.97 Joules over 29.76 million seconds (the result of the calculation in Table 7-19).

Table 7-18 : The total number of clock cycles, energy usage and time consumption to operate the proposed system based on texture segmentation on different image sizes on a PIC16F877 microcontroller (4 clock cycles per instruction).

Image size	Number of clock cycles to operate the proposed system by using 4 clock cycles for an instruction				Total number of clock cycles	Microcontroller	
	Multiply (4)	Plus (4)	Divide (4)	Compare (4)		PIC 16F877	
						50 mW @ 20 MHz	
						Time (s)	Energy (J)
1. 16x16	537600	264192	0	4096	805888	0.0402944	0.002
2. 32x32	8441856	4202496	0	16384	12660736	0.6330368	0.032
3. 64x64	134430720	67141632	0	20480	201592832	10.0818944	0.504
4. 128x128	2148335616	1073872896	0	81920	3222290432	161.1235328	8.05617664
5. 256x256	34363146240	17180393472	0	1048576	51544588288	2577.229414	128.8614707
6. 512x512	5.49769E+11	2.7488E+11	0	4194304	8.24654E+11	41,232.68	2061.634109
7. 1024x1024	8.79615E+12	4.39805E+12	0	16777216	1.31942E+13	659,710.96	32985.54806
8. 2048x1536	7.93713E+13	3.96856E+13	0	50397184	1.19057E+14	5,952,846.71	297642.3354
9. 2048x2048	1.40738E+14	7.03688E+13	0	67108864	2.11107E+14	10,555,327.5	527766.3783

Table 7-19 : The total number of clock cycles, energy usage and time consumption to operate the proposed system on different image sizes on an AVR Atmega128 microcontroller.

Image size	Number of clock cycles to operate the proposed system by using 1 clock cycles for an instruction				Total number of clock cycles	Microcontroller	
	Multiply (1)	Plus (1)	Divide (1)	Compare (1)		AVR Atmega128	
						9.9 mW@ 1 MHz	
						Time (s)	Energy(J)
1. 16x16	134,400	66,048	0	1,024	201,472	0.20	0.00066
2. 32x32	2,110,464	1,050,624	0	4,096	3,165,184	3.16	0.0104
3. 64x64	33,607,680	16,785,408	0	20,480	50,413,568	50.41	0.166
4. 128x128	537,083,904	268,468,224	0	81,920	80,5634,048	805.62	2.658
5. 256x256	8,590,786,560	4,295,098,368	0	262,144	1,288,6147,072	12,886.15	42.524
6. 512x512	1.37442E+11	68,720,001,024	0	1,048,576	2.06163E+11	206,163.41	680.339
7. 1024x1024	2.19904E+12	1.09951E+12	0	4,194,304	3.29855E+12	3,298,554.81	10,885.231
8.2048x1536	1.98428E+13	9.9214E+12	0	12,599,296	2.97642E+13	29,764,233.54	98,221.971
9. 2048x2048	3.51844E+13	1.75922E+13	0	16,777,216	5.27766E+13	52,776,637.83	174,162.905

Figure 7-27 shows the relationship between the images of different sizes with time consumption to operate the proposed system by using PIC16F877 and AVR Atmega128 microcontroller. Besides Figure 7-28 shows the relationship between the images in different sizes with the energy that required processing the proposed system on PIC16F877 and AVR Atmega128 microcontroller.

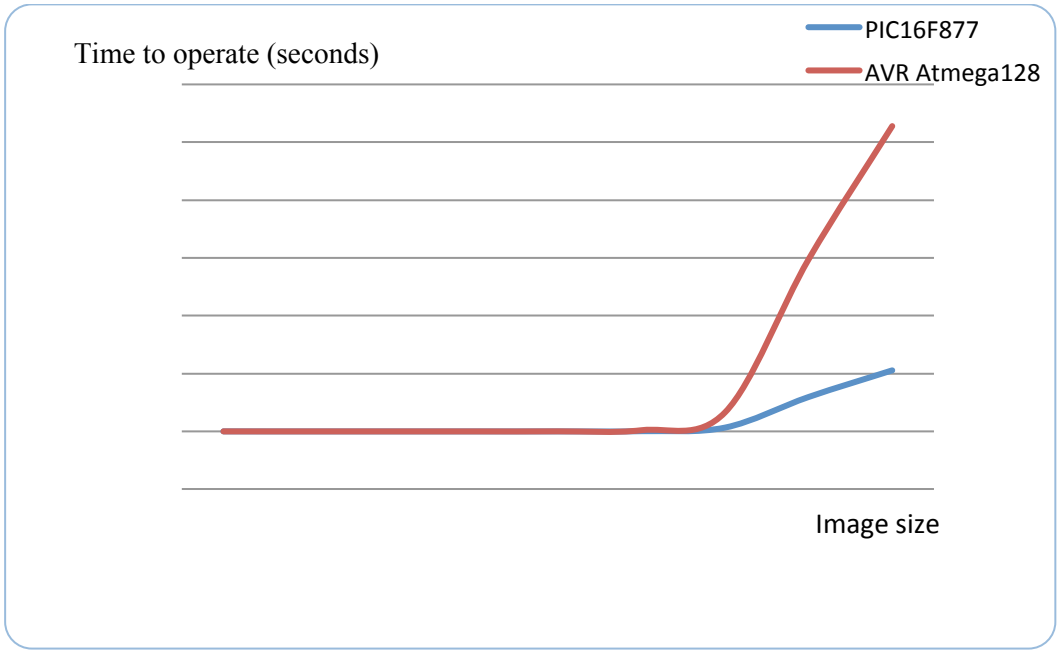


Figure 7-27 : The relationship between input image size and the time consumption.

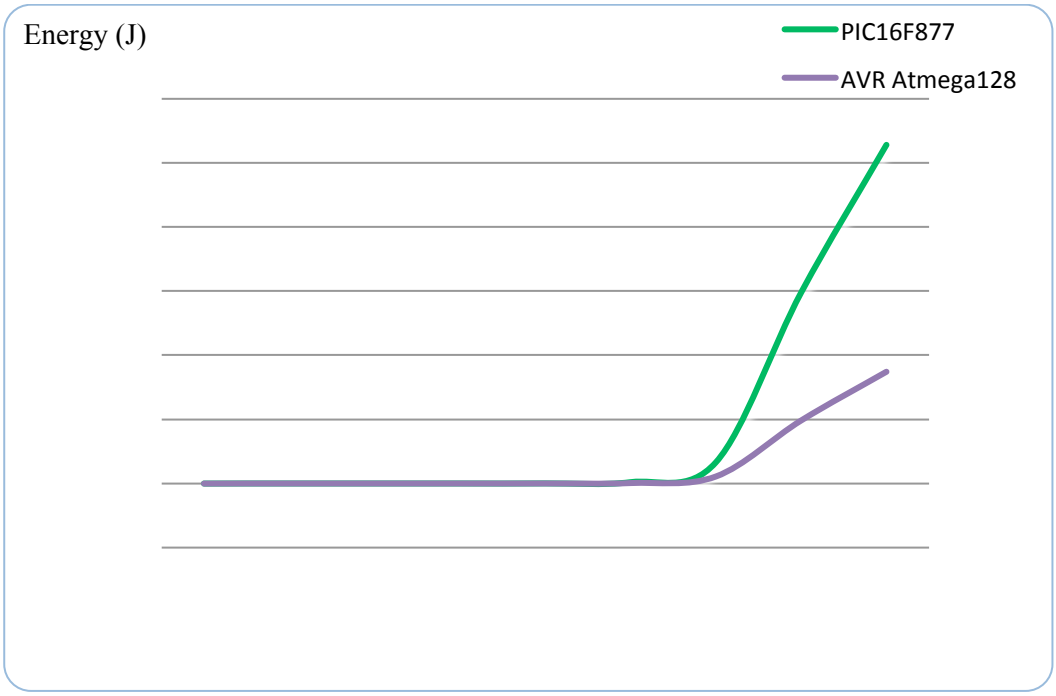


Figure 7-28 : The relationship between input image size and the energy required.

7.5 The hardware comparison

Table 7-20 demonstrates a quantitative comparison of the ratio of computational cost by data reduction factor for the three image compression schemes based on edge, histogram and texture segmentation. The table shows the computational cost (energy used) to compute an image size of 2048 x 1536 on a PIC16F877 microcontroller, which is 1,100.68, 638.82 and 297,642.33 mJ respectively. The average reduction size of each scheme is 66.70 %, 73.39 % and 73.00 % of the original image respectively. Dividing computational energy by data reduction factor gives a figure of merit for each scheme, this being 16.7, 8.7 and 4077.3 respectively. Therefore using this figure of merit the proposed scheme based on histogram segmentation is best.

Table 7-20 : Quantitative comparison of the ratio of computational cost by data reduction factor for the three image compression schemes based on edge, histogram and texture segmentation.

The proposed system based on	1. Edge	2. Histogram	3. Texture
Computation energy usage (mJ)	1,100.68	638.82	297,642.33
Average data reduction of original image	66.70 %	73.39 %	73.00 %
Figure of merit comparison (Computation energy / Data reduction)	16.65	8.7	4,077.29

In comparison the hardware synthesis for the system based on histogram segmentation give the shortest computation time and therefore the lowest energy to process the image.

The proposed system is therefore based on histogram segmentation as this gives the best result from the analysis above due to efficient processing and it can be used in many applications.

The hardware synthesis for the proposed systems shows that the system based on texture segmentation consumes a large amount of energy. There are methods proposed to reduce the processing time required to perform the GLCM (Mirzapour et al, 2013). However these necessitate some loss of accuracy as the number of pixels considered is reduced.

7.6 Summary

In this chapter a new image compression method has been described. The Region of Interest (ROI) of an image was detected by several image segmentation techniques; edge, histogram and texture. In addition three different techniques of image compressions have been implemented in the proposed system. The experimental results show that the proposed system can identify the ROI part of the satellite images and compress this part of the image without losing any information.

In addition, this chapter has presented the hardware synthesis results. The analysis was performed to investigate the possibility of implementing the proposed system on a cubesat and real hardware devices. This analysis was focused on the amount of energy that is required to process the imagery. The results show that it is possible to implement the proposed system on a hardware microcontroller on a real cubesat with the limitation of the amount of power available.

Chapter 8

Future work and Conclusions

The objective of this research was to develop the onboard computer for a cubesat to increase its capability. Cubesats that have been developed to take images of the Earth for remote sensing applications require considerable power and memory storage onboard the satellite. The work described in this thesis addresses the potential to reduce the size of images onboard these satellites automatically. The research focused on a hybrid onboard computer architecture and adaptive image compression based on ROI automatic detection to address the limitations on power transmission and memory storage of the spacecraft. This chapter presents the future work and conclusions of the research.

8.1 Future work

The study has shown how the use of ROI image detection and image compression technique can reduce the size of the satellite image before transmission to the receiving station. However there are improvements that could be developed to improve the ROI automatic detection and image compression:

- There are many parameters that are used in the proposed system. Some parameters used are recommended default values in the Matlab programming, such as grey scale thresholding level, black and white thresholding level and edge detection thresholds for low and high levels. These parameters need to be studied to find suitable values for particular images. Also the morphology structure element sizes and shapes, should be reconsidered for different types of images such as radar or different bands of a satellite image. Optimisation of these parameter values for particular spacecraft mission objectives and constraints should be performed.
- The ROI automatic detection based on histogram segmentation classifies using thresholds that are selected manually. It is possible that these thresholds could be identified automatically (Guarnieri and Vettore 2002).
- To reduce the size of an image by ROI automatic detection, some part of the original image is lost. The non-ROI part is discarded before compression of the ROI part. This

non-ROI area cannot be recovered on the ground. An alternative solution would be to preserve the non-ROI areas by compressing the ROI and non-ROI parts with two different compression levels. For instance, the ROI part can be compressed to a low compression ratio and the non-ROI part can be compressed to a high compression ratio. With this solution, the ground station can recover the whole image if required.

- To implement the proposed system on a real cubesat, real or simulated satellite imagery from cubesats should be used to determine the exact threshold levels or parameters that are appropriate for the mission. The reason for this is that the various sizes and accuracy of the ROI image is dependent on the characteristics of the original image.
- The automatic selection of the most appropriate adaptive compression scheme would be a valuable addition to the research undertaken. The appropriate proposed system for each image can be selected according to the image characteristics. Many of the processes such as grey scale image conversion, morphology dilation and morphology fill region and holes are common to all the systems investigated. Therefore a new automatic decision making system using the optimum system or a combination of two or three proposed systems should be investigated further.
- The fundamental limit of size restricts the spatial resolution that can be achieved on a cubesat but it is possible that multispectral information can be obtained. Different Earth surface types such as water, forest, sand and vegetation reflect and absorb radiation differently in various channels enabling each surface type to be distinguished from each other. It would therefore be possible to identify the object by using the spectral signature of each object and apply a system based on histogram segmentation to further reduce the image size using the image compression technique.

8.2 Conclusions

The current development of cubesats is aimed towards improving their capability so they can be used in more complex missions.

The more complex a cubesat mission is, the more constraints and requirements are placed on system components. Mission adjustments or device specification updates require flexibility of the cubesat onboard computer architecture. This research proposes an alternative onboard computer architecture using a hybrid architecture. The proposed architecture includes the advantages of both central and distributed onboard computer architectures. It is possible to

use this new architecture in future cubesat missions and the benefits are increased flexibility and reliability. The disadvantages of extra power and mass have been quantified in this thesis.

The transmission power limitations of a cubesat restrict the speed of transmission of data to the ground station. To address these constraints, an image compression algorithm has been developed in this research to reduce the size of an image that needs to be transmitted to ground. This has been done by examining ROI automatic detection and image compression.

ROI automatic detection

The ROI automatic detection is a technique that defines the area of an image that is useful to the user. Image segmentation is utilised to define the ROI area. There are three different techniques that have been used to identify the ROI of satellite images: edge, histogram and texture image segmentation.

The edge detection and dilation morphological technique have been used for this ROI automatic detection by detecting the area of the image where there are sharp changes or discontinuities in the grey level intensity. Canny edge detection has been applied in this method and the dilation technique has been used to group areas of similarity together. This combination of techniques is of benefit for ROI identification.

Histogram segmentation is based on histogram classification of an image. The pixel values in an image can be represented by histograms of the intensity observed. The concept of histogram segmentation uses classification of an image using thresholds. The thresholding is performed by considering each peak of the histogram and mapping different intensities to different objects. After the thresholds were determined each pixel value in the group was replaced by the group pixel value. In addition morphological dilation and “fill the holes” processes are implemented to find the ROI of an image.

The texture segmentation method classifies the image using the statistical features of the image using the Grey-Level Co-occurrence Matrix (GLCM). There are five statistical features studied in this research; energy, entropy, contrast, homogeneity and correlation. The texture segmentation technique has been implemented to segment the example satellite images. The next step applies the morphological techniques used in the other techniques to define the ROI of the satellite image.

There is little difference in the reduction in size achievable with the three different image segmentation methods. The ROI automatic detection based on edge detection gives the most satisfactory implementation for a cubesat. Firstly, it automatically detects the ROI whereas the histogram segmentation method has some part that is currently based on human intervention. Secondly, the ROI detection based on edge segmentation requires a lot less time to compute

compared to the time required by the ROI detection based on texture segmentation.

Image compression module

The image compression module is implemented using image compression based on the wavelet transform. There are three different types that have been studied; 9/7 biorthogonal, Haar and Jpeg2000. The recommended image compression method based on 9/7 biorthogonal gave a lower compression ratio than Haar and Jpeg2000 image compression, but it is simpler to implement and has less complex hardware and software configurations.

The accuracy of the overall proposed system was examined by comparing the number of different pixels between the proposed ROI automatic and manual detection techniques. The average percentage difference between the proposed automatic ROI detection and manual ROI detection is approximately 7%. The morphology structure element has been studied to find the appropriate shape and size that can be implemented on the proposed system. The morphology technique is the common process used in the proposed ROI automatic detections and so the shape and size of structure element are important to study due to its effects on the accuracy and the size of the final ROI image. The results show that the square shape structure element at size-3 (15x15 pixels) gives the smallest percentage of difference between the manual and automatic ROI detection, i.e. 1.37%, whilst a larger structure element size provides the larger output ROI image size, a smaller size gives significant areas missing in the resulting ROI image. The analysis procedure involving morphology structure element operation can be applied in many pre- and post- image processing applications, for instance document analysis, enhancing object structure, convex hull or object marking, etc.

To identify and compare the satellite power required for the image compression the number of instruction counts required to execute the proposed system has been assessed. For instance the proposed system implemented on a PIC16F877 microcontroller operating on an image size of 2048x1536 pixels required 50mW for 22 seconds to reduce the size of the image.

The proposed systems presented are based on edge, histogram and texture segmentation. An edge detection system is suitable to detect a region of interest identified as the coast, an island or ocean. Specific uses of land (e.g. forest/urban/agriculture) are better identified using histogram segmentation. Cloud detection and rejection can be applied using histogram segmentation. A system based on texture segmentation has been found to be very time and energy consuming and is not suitable for use on a satellite that has low power availability like a cubesat.

The proposed image compression process consists of a ROI automatic detection algorithm using segmentation and morphological techniques and subsequently an image

compression technique. The potential advantage in using the proposed image compression process on cubesat missions is to reduce the size of captured images compared to standard image compression techniques. The additional power required by the proposed system has been estimated and found to be very small and therefore has the potential to be easily accommodated by the cubesat's power system.

Appendices

Appendix A : Link budget calculation

Table A-1: The link budget result with power transmission 1 watt at E_b/N_o 10, 10.3 and 13 dB.

Altitude (km)	Slant length (km)	Data rate (Mbps), Pt = 1 Watt					
		E_b/N_o = 10		E_b/N_o = 10.3		E_b/N_o = 13	
		Value	dB	Value	dB	Value	dB
350	1303.399	2.077240	63.174867	1.938593	62.87	1.041086	60.174867
360	1331.161	1.991499	62.991802	1.858575	62.69	0.998114	59.991802
370	1358.651	1.911725	62.814254	1.784126	62.51	0.958132	59.814254
380	1385.878	1.837347	62.641911	1.714712	62.34	0.920855	59.641911
390	1412.851	1.767862	62.474484	1.649865	62.17	0.886030	59.474484
400	1439.578	1.702829	62.311710	1.589172	62.01	0.853436	59.311710
410	1466.066	1.641854	62.153344	1.532267	61.85	0.822876	59.153344
420	1492.322	1.584587	61.999161	1.478823	61.70	0.794175	58.999161
430	1518.354	1.530717	61.848950	1.428549	61.55	0.767176	58.848950
440	1544.169	1.479966	61.702518	1.381185	61.40	0.741740	58.702518
450	1569.772	1.432084	61.559684	1.336498	61.26	0.717742	58.559684
460	1595.169	1.386845	61.420279	1.294279	61.12	0.695069	58.420279
470	1620.366	1.344048	61.284148	1.254339	60.98	0.673620	58.284148
480	1645.37	1.303510	61.151143	1.216506	60.85	0.653302	58.151143
490	1670.184	1.265064	61.021126	1.180627	60.72	0.634034	58.021126
500	1694.814	1.228562	60.893970	1.146561	60.59	0.615740	57.893970
510	1719.265	1.193866	60.769554	1.114180	60.47	0.598350	57.769554
520	1743.542	1.160851	60.647765	1.083369	60.35	0.581804	57.647765
530	1767.648	1.129405	60.528496	1.054022	60.23	0.566043	57.528496
540	1791.589	1.099423	60.411647	1.026041	60.11	0.551017	57.411647
550	1815.367	1.070810	60.297123	0.999338	60.00	0.536676	57.297123
560	1838.988	1.043479	60.184836	0.973831	59.88	0.522978	57.184836
570	1862.454	1.017349	60.074701	0.949446	59.77	0.509882	57.074701
580	1885.77	0.992347	59.966638	0.926113	59.67	0.497352	56.966638
590	1908.939	0.968405	59.860572	0.903768	59.56	0.485352	56.860572
600	1931.964	0.945460	59.756431	0.882355	59.46	0.473852	56.756431
610	1954.849	0.923453	59.654148	0.861817	59.35	0.462823	56.654148
620	1977.596	0.902331	59.553659	0.842104	59.25	0.452237	56.553659
630	2000.21	0.882044	59.454903	0.823171	59.15	0.442069	56.454903
640	2022.691	0.862546	59.357821	0.804974	59.06	0.432297	56.357821
650	2045.044	0.843793	59.262358	0.787473	58.96	0.422898	56.262358
660	2067.272	0.825745	59.168461	0.770630	58.87	0.413853	56.168461
670	2089.376	0.808366	59.076081	0.754411	58.78	0.405143	56.076081
680	2111.359	0.791620	58.985170	0.738783	58.69	0.396750	55.985170
690	2133.224	0.775476	58.895682	0.723716	58.60	0.388659	55.895682
700	2154.974	0.759902	58.807574	0.709181	58.51	0.380853	55.807574

Altitude (km)	Slant length (km)	Data rate (Mbps), Pt = 1 Watt					
		E_b/N_o = 10		E_b/N_o = 10.3		E_b/N_o = 13	
		Value	dB	Value	dB	Value	dB
710	2176.609	0.744870	58.720804	0.695153	58.42	0.373319	55.720804
720	2198.134	0.730354	58.635331	0.681606	58.34	0.366044	55.635331
730	2219.549	0.716328	58.551119	0.668516	58.25	0.359014	55.551119
740	2240.857	0.702770	58.468131	0.655863	58.17	0.352219	55.468131
750	2262.06	0.689657	58.386331	0.643625	58.09	0.345647	55.386331
760	2283.16	0.676969	58.305687	0.631784	58.01	0.339288	55.305687
770	2304.158	0.664686	58.226166	0.620321	57.93	0.333132	55.226166
780	2325.058	0.652790	58.147737	0.609219	57.85	0.327170	55.147737
790	2345.86	0.641264	58.070371	0.598463	57.77	0.321394	55.070371
800	2366.566	0.630092	57.994040	0.588036	57.69	0.315794	54.994040
810	2387.178	0.619258	57.918716	0.577925	57.62	0.310364	54.918716
820	2407.698	0.608748	57.844373	0.568116	57.54	0.305097	54.844373
830	2428.127	0.598547	57.770985	0.558597	57.47	0.299984	54.770985
840	2448.467	0.588644	57.698529	0.549355	57.40	0.295021	54.698529
850	2468.719	0.579026	57.626981	0.540379	57.33	0.290200	54.626981
860	2488.884	0.569681	57.556319	0.531657	57.26	0.285517	54.556319
870	2508.965	0.560599	57.486520	0.523181	57.19	0.280965	54.486520
880	2528.963	0.551768	57.417564	0.514940	57.12	0.276539	54.417564
890	2548.879	0.543179	57.349430	0.506924	57.05	0.272234	54.349430
900	2568.713	0.534823	57.282100	0.499126	56.98	0.268046	54.282100
910	2588.469	0.526690	57.215553	0.491536	56.92	0.263970	54.215553
920	2608.146	0.518773	57.149773	0.484147	56.85	0.260002	54.149773
930	2627.747	0.511063	57.084742	0.476951	56.78	0.256138	54.084742
940	2647.272	0.503552	57.020442	0.469942	56.72	0.252374	54.020442
950	2666.722	0.496233	56.956857	0.463112	56.66	0.248706	53.956857
960	2686.099	0.489099	56.893972	0.456454	56.59	0.245130	53.893972
970	2705.404	0.482144	56.831770	0.449963	56.53	0.241645	53.831770
980	2724.638	0.475361	56.770238	0.443633	56.47	0.238245	53.770238
990	2743.801	0.468744	56.709360	0.437458	56.41	0.234929	53.709360
1000	2762.896	0.462288	56.649123	0.431432	56.35	0.231693	53.649123
1010	2781.922	0.455986	56.589513	0.425551	56.29	0.228534	53.589513
1020	2800.882	0.449833	56.530518	0.419809	56.23	0.225451	53.530518
1030	2819.775	0.443826	56.472123	0.414202	56.17	0.222440	53.472123
1040	2838.603	0.437957	56.414318	0.408726	56.11	0.219499	53.414318
1050	2857.367	0.432224	56.357091	0.403375	56.06	0.216625	53.357091
1060	2876.068	0.426622	56.300429	0.398146	56.00	0.213817	53.300429
1070	2894.707	0.421145	56.244321	0.393036	55.94	0.211073	53.244321
1080	2913.284	0.415792	56.188757	0.388039	55.89	0.208389	53.188757
1090	2931.8	0.410556	56.133726	0.383153	55.83	0.205766	53.133726
1100	2950.256	0.405435	56.079217	0.378374	55.78	0.203199	53.079217
1110	2968.654	0.400426	56.025222	0.373699	55.73	0.200688	53.025222
1120	2986.993	0.395524	55.971729	0.369125	55.67	0.198232	52.971729
1130	3005.275	0.390727	55.918729	0.364647	55.62	0.195827	52.918729
1140	3023.5	0.386030	55.866214	0.360264	55.57	0.193473	52.866214
1150	3041.669	0.381432	55.814174	0.355973	55.51	0.191169	52.814174
1160	3059.783	0.376930	55.762601	0.351771	55.46	0.188912	52.762601
1170	3077.842	0.372519	55.711486	0.347655	55.41	0.186702	52.711486

Altitude (km)	Slant length (km)	Data rate (Mbps), Pt = 1 Watt					
		E_b/N_o = 10		E_b/N_o = 10.3		E_b/N_o = 13	
		Value	dB	Value	dB	Value	dB
1180	3095.848	0.368199	55.660821	0.343623	55.36	0.184536	52.660821
1190	3113.8	0.363965	55.610598	0.339672	55.31	0.182415	52.610598
1200	3131.7	0.359816	55.560810	0.335800	55.26	0.180335	52.560810
1210	3149.548	0.355750	55.511447	0.332005	55.21	0.178297	52.511447
1220	3167.345	0.351763	55.462504	0.328285	55.16	0.176299	52.462504
1230	3185.092	0.347854	55.413973	0.324636	55.11	0.174340	52.413973
1240	3202.789	0.344021	55.365847	0.321059	55.07	0.172419	52.365847
1250	3220.436	0.340261	55.318118	0.317550	55.02	0.170534	52.318118
1260	3238.035	0.336572	55.270781	0.314107	54.97	0.168686	52.270781

Table A-2: The link budget result with E_b/N_o 10.3 and power transmission at 0.5, 1 and 2 Watts.

Altitude (km)	Slant length (km)	Data rate (Mbps), E_b/N_o = 10.3 dB					
		Pt = 0.5 Watts		Pt = 1 Watts		Pt = 2 Watts	
		Value	dB	Value	dB	Value	dB
350	1303.399	0.961277	59.828485	1.922554	62.84	3.845108	65.849085
360	1331.161	0.921599	59.645420	1.843198	62.66	3.686396	65.666020
370	1358.651	0.884682	59.467873	1.769364	62.48	3.538729	65.488473
380	1385.878	0.850262	59.295529	1.700525	62.31	3.401049	65.316129
390	1412.851	0.818107	59.128103	1.636215	62.14	3.272429	65.148703
400	1439.578	0.788012	58.965329	1.576024	61.98	3.152048	64.985929
410	1466.066	0.759795	58.806963	1.519589	61.82	3.039179	64.827563
420	1492.322	0.733294	58.652779	1.466587	61.66	2.933175	64.673379
430	1518.354	0.708365	58.502568	1.416729	61.51	2.833458	64.523168
440	1544.169	0.684879	58.356136	1.369757	61.37	2.739515	64.376736
450	1569.772	0.662720	58.213302	1.325440	61.22	2.650881	64.233902
460	1595.169	0.641785	58.073898	1.283571	61.08	2.567141	64.094498
470	1620.366	0.621980	57.937766	1.243961	60.95	2.487921	63.958366
480	1645.37	0.603221	57.804761	1.206441	60.82	2.412882	63.825361
490	1670.184	0.585429	57.674745	1.170859	60.69	2.341717	63.695345
500	1694.814	0.568537	57.547589	1.137074	60.56	2.274149	63.568189
510	1719.265	0.552481	57.423173	1.104962	60.43	2.209924	63.443773
520	1743.542	0.537203	57.301384	1.074406	60.31	2.148812	63.321983
530	1767.648	0.522651	57.182115	1.045301	60.19	2.090602	63.202714
540	1791.589	0.508776	57.065265	1.017552	60.08	2.035104	63.085865
550	1815.367	0.495535	56.950742	0.991070	59.96	1.982139	62.971342
560	1838.988	0.482887	56.838454	0.965774	59.85	1.931548	62.859054
570	1862.454	0.470795	56.728319	0.941590	59.74	1.883180	62.748919
580	1885.77	0.459225	56.620256	0.918450	59.63	1.836900	62.640856
590	1908.939	0.448145	56.514190	0.896291	59.52	1.792582	62.534790
600	1931.964	0.437527	56.410049	0.875054	59.42	1.750108	62.430649
610	1954.849	0.427343	56.307767	0.854686	59.32	1.709372	62.328367
620	1977.596	0.417569	56.207278	0.835137	59.22	1.670274	62.227878

Altitude (km)	Slant length (km)	Data rate (Mbps), E_b/N_o = 10.3 dB					
		Pt = 0.5 Watts		Pt = 1 Watts		Pt = 2 Watts	
		Value	dB	Value	dB	Value	dB
630	2000.21	0.408180	56.108521	0.816361	59.12	1.632721	62.129121
640	2022.691	0.399157	56.011439	0.798314	59.02	1.596629	62.032039
650	2045.044	0.390479	55.915976	0.780958	58.93	1.561916	61.936576
660	2067.272	0.382127	55.822080	0.764254	58.83	1.528509	61.842680
670	2089.376	0.374085	55.729700	0.748169	58.74	1.496339	61.750300
680	2111.359	0.366335	55.638789	0.732671	58.65	1.465342	61.659389
690	2133.224	0.358864	55.549301	0.717728	58.56	1.435457	61.569901
700	2154.974	0.351657	55.461192	0.703314	58.47	1.406628	61.481792
710	2176.609	0.344701	55.374422	0.689401	58.38	1.378803	61.395022
720	2198.134	0.337983	55.288950	0.675966	58.30	1.351932	61.309550
730	2219.549	0.331493	55.204738	0.662985	58.22	1.325970	61.225338
740	2240.857	0.325218	55.121749	0.650437	58.13	1.300873	61.142349
750	2262.06	0.319150	55.039950	0.638300	58.05	1.276600	61.060550
760	2283.16	0.313278	54.959305	0.626557	57.97	1.253114	60.979905
770	2304.158	0.307594	54.879784	0.615189	57.89	1.230378	60.900384
780	2325.058	0.302089	54.801356	0.604179	57.81	1.208358	60.821956
790	2345.86	0.296756	54.723990	0.593511	57.73	1.187023	60.744590
800	2366.566	0.291585	54.647659	0.583171	57.66	1.166342	60.668258
810	2387.178	0.286572	54.572334	0.573144	57.58	1.146287	60.592934
820	2407.698	0.281708	54.497991	0.563416	57.51	1.126832	60.518591
830	2428.127	0.276988	54.424604	0.553975	57.43	1.107951	60.445204
840	2448.467	0.272405	54.352148	0.544810	57.36	1.089619	60.372748
850	2468.719	0.267954	54.280600	0.535908	57.29	1.071815	60.301200
860	2488.884	0.263629	54.209937	0.527259	57.22	1.054517	60.230537
870	2508.965	0.259426	54.140138	0.518852	57.15	1.037705	60.160738
880	2528.963	0.255340	54.071182	0.510679	57.08	1.021358	60.091782
890	2548.879	0.251365	54.003048	0.502730	57.01	1.005460	60.023648
900	2568.713	0.247498	53.935718	0.494996	56.95	0.989992	59.956318
910	2588.469	0.243735	53.869172	0.487469	56.88	0.974938	59.889772
920	2608.146	0.240071	53.803392	0.480141	56.81	0.960283	59.823992
930	2627.747	0.236503	53.738360	0.473005	56.75	0.946011	59.758960
940	2647.272	0.233027	53.674060	0.466054	56.68	0.932108	59.694660
950	2666.722	0.229640	53.610476	0.459280	56.62	0.918560	59.631076
960	2686.099	0.226339	53.547590	0.452678	56.56	0.905355	59.568190
970	2705.404	0.223120	53.485389	0.446240	56.50	0.892481	59.505989
980	2724.638	0.219981	53.423856	0.439962	56.43	0.879925	59.444456
990	2743.801	0.216919	53.362978	0.433838	56.37	0.867676	59.383578
1000	2762.896	0.213931	53.302741	0.427862	56.31	0.855725	59.323341
1010	2781.922	0.211015	53.243132	0.422030	56.25	0.844060	59.263731
1020	2800.882	0.208168	53.184136	0.416336	56.19	0.832671	59.204736
1030	2819.775	0.205388	53.125742	0.410775	56.14	0.821550	59.146342
1040	2838.603	0.202672	53.067937	0.405344	56.08	0.810688	59.088537
1050	2857.367	0.200019	53.010709	0.400038	56.02	0.800075	59.031309
1060	2876.068	0.197426	52.954047	0.394852	55.96	0.789705	58.974647
1070	2894.707	0.194892	52.897940	0.389784	55.91	0.779568	58.918539

Altitude (km)	Slant length (km)	Data rate (Mbps), E_b/N_o = 10.3 dB					
		Pt = 0.5 Watts		Pt = 1 Watts		Pt = 2 Watts	
		Value	dB	Value	dB	Value	dB
1080	2913.284	0.192414	52.842376	0.384829	55.85	0.769658	58.862975
1090	2931.8	0.189992	52.787344	0.379983	55.80	0.759966	58.807944
1100	2950.256	0.187622	52.732836	0.375244	55.74	0.750488	58.753436
1110	2968.654	0.185304	52.678840	0.370607	55.69	0.741215	58.699440
1120	2986.993	0.183035	52.625347	0.366070	55.64	0.732141	58.645947
1130	3005.275	0.180815	52.572348	0.361630	55.58	0.723261	58.592948
1140	3023.5	0.178642	52.519833	0.357284	55.53	0.714567	58.540433
1150	3041.669	0.176514	52.467793	0.353028	55.48	0.706056	58.488393
1160	3059.783	0.174430	52.416220	0.348861	55.43	0.697721	58.436820
1170	3077.842	0.172389	52.365105	0.344779	55.38	0.689557	58.385705
1180	3095.848	0.170390	52.314440	0.340780	55.32	0.681560	58.335040
1190	3113.8	0.168431	52.264217	0.336862	55.27	0.673723	58.284817
1200	3131.7	0.166511	52.214428	0.333022	55.22	0.666044	58.235028
1210	3149.548	0.164629	52.165066	0.329258	55.18	0.658516	58.185666
1220	3167.345	0.162784	52.116123	0.325568	55.13	0.651137	58.136723
1230	3185.092	0.160975	52.067591	0.321951	55.08	0.643901	58.088191
1240	3202.789	0.159201	52.019465	0.318403	55.03	0.636805	58.040065
1250	3220.436	0.157461	51.971737	0.314922	54.98	0.629845	57.992337
1260	3238.035	0.155754	51.924400	0.311509	54.93	0.623017	57.945000

Table A-3: The link budget result for data rate with power transmission at 1 Watt at E_b/N_o 10, 10.3 and 13 dB at the polar ground station and different satellite altitudes.

Altitude (km)	Slant length (km)	Data transmission rate Pt =1 watt		
		E_b/N_o = 10	E_b/N_o = 10.3	E_b/N_o = 13
350	1303.399	2.395058	2.235198	1.200373
360	1331.161	2.296199	2.142938	1.150826
370	1358.651	2.204219	2.057097	1.104727
380	1385.878	2.118461	1.977063	1.061746
390	1412.851	2.038346	1.902295	1.021593
400	1439.578	1.963362	1.832316	0.984012
410	1466.066	1.893058	1.766704	0.948776
420	1492.322	1.827029	1.705083	0.915684
430	1518.354	1.764917	1.647117	0.884554
440	1544.169	1.706401	1.592506	0.855227

Altitude (km)	Slant length (km)	Data transmission rate Pt =1 watt		
		E_b/N_o = 10	E_b/N_o = 10.3	E_b/N_o = 13
450	1569.772	1.651193	1.540983	0.827557
460	1595.169	1.599033	1.492304	0.801415
470	1620.366	1.549688	1.446253	0.776684
480	1645.37	1.502947	1.402632	0.753258
490	1670.184	1.458620	1.361263	0.731041
500	1694.814	1.416532	1.321985	0.709948
510	1719.265	1.376527	1.284650	0.689898
520	1743.542	1.338462	1.249125	0.670820
530	1767.648	1.302204	1.215287	0.652648
540	1791.589	1.267635	1.183025	0.635322
550	1815.367	1.234644	1.152237	0.618788
560	1838.988	1.203131	1.122827	0.602994
570	1862.454	1.173004	1.094711	0.587895
580	1885.77	1.144177	1.067808	0.573447
590	1908.939	1.116572	1.042045	0.559611
600	1931.964	1.090115	1.017355	0.546352
610	1954.849	1.064742	0.993675	0.533635
620	1977.596	1.040388	0.970947	0.521429
630	2000.21	1.016997	0.949117	0.509706
640	2022.691	0.994515	0.928136	0.498438
650	2045.044	0.972893	0.907957	0.487602
660	2067.272	0.952085	0.888537	0.477173
670	2089.376	0.932046	0.869836	0.467130
680	2111.359	0.912739	0.851817	0.457453
690	2133.224	0.894124	0.834445	0.448123

Altitude (km)	Slant length (km)	Data transmission rate Pt =1 watt		
		$E_b/N_o = 10$	$E_b/N_o = 10.3$	$E_b/N_o = 13$
700	2154.974	0.876167	0.817686	0.439124
710	2176.609	0.858835	0.801511	0.430437
720	2198.134	0.842098	0.785891	0.422049
730	2219.549	0.825926	0.770799	0.413944
740	2240.857	0.810294	0.756210	0.406109
750	2262.06	0.795175	0.742100	0.398531
760	2283.16	0.780545	0.728447	0.391199
770	2304.158	0.766383	0.715230	0.384101
780	2325.058	0.752667	0.702430	0.377227
790	2345.86	0.739378	0.690028	0.370567
800	2366.566	0.726496	0.678006	0.364111
810	2387.178	0.714005	0.666348	0.357850
820	2407.698	0.701886	0.655038	0.351776
830	2428.127	0.690125	0.644062	0.345882
840	2448.467	0.678707	0.633406	0.340159
850	2468.719	0.667617	0.623057	0.334601
860	2488.884	0.656843	0.613001	0.329201
870	2508.965	0.646370	0.603228	0.323953
880	2528.963	0.636188	0.593726	0.318850
890	2548.879	0.626286	0.584484	0.313886
900	2568.713	0.616651	0.575492	0.309058
910	2588.469	0.607274	0.566741	0.304358
920	2608.146	0.598145	0.558222	0.299783
930	2627.747	0.589255	0.549925	0.295327
940	2647.272	0.580595	0.541843	0.290987

Altitude (km)	Slant length (km)	Data transmission rate Pt =1 watt		
		$E_b/N_o = 10$	$E_b/N_o = 10.3$	$E_b/N_o = 13$
950	2666.722	0.572157	0.533968	0.286758
960	2686.099	0.563932	0.526292	0.282635
970	2705.404	0.555912	0.518808	0.278616
980	2724.638	0.548092	0.511509	0.274697
990	2743.801	0.540462	0.504389	0.270873
1000	2762.896	0.533018	0.497441	0.267142
1010	2781.922	0.525752	0.490660	0.263500
1020	2800.882	0.518658	0.484040	0.259945
1030	2819.775	0.511731	0.477575	0.256473
1040	2838.603	0.504965	0.471261	0.253082
1050	2857.367	0.498355	0.465092	0.249769
1060	2876.068	0.491895	0.459063	0.246531
1070	2894.707	0.485581	0.453170	0.243367
1080	2913.284	0.479408	0.447409	0.240273
1090	2931.8	0.473371	0.441776	0.237248
1100	2950.256	0.467467	0.436266	0.234289
1110	2968.654	0.461691	0.430875	0.231394
1120	2986.993	0.456039	0.425601	0.228561
1130	3005.275	0.450508	0.420438	0.225789
1140	3023.5	0.445093	0.415385	0.223075
1150	3041.669	0.439792	0.410437	0.220418
1160	3059.783	0.434600	0.405592	0.217816
1170	3077.842	0.429515	0.400846	0.215267
1180	3095.848	0.424533	0.396197	0.212771
1190	3113.8	0.419652	0.391642	0.210324

Altitude (km)	Slant length (km)	Data transmission rate Pt =1 watt		
		E_b/N_o = 10	E_b/N_o = 10.3	E_b/N_o = 13
1200	3131.7	0.414868	0.387178	0.207927
1210	3149.548	0.410180	0.382802	0.205577
1220	3167.345	0.405583	0.378512	0.203273
1230	3185.092	0.401076	0.374306	0.201014
1240	3202.789	0.396656	0.370181	0.198799
1250	3220.436	0.392321	0.366135	0.196626
1260	3238.035	0.388068	0.362166	0.194495

Table A-4: The link budget result for data rate with E_b/N_o 10.3 and power transmission at 0.5, 1 and 2 Watts at polar station and satellite in different altitudes.

Altitude (km)	Slant length (km)	Data rate (Mb/s) E_b/N_o = 10.3 dB		
		Pt = 0.5 Watts	Pt = 1 Watts	Pt = 2 Watts
350	1303.399	1.112126	2.224251	4.448502
360	1331.161	1.066221	2.132443	4.264885
370	1358.651	1.023511	2.047022	4.094045
380	1385.878	0.983690	1.967380	3.934760
390	1412.851	0.946489	1.892978	3.785956
400	1439.578	0.911671	1.823342	3.646685
410	1466.066	0.879026	1.758052	3.516103
420	1492.322	0.848366	1.696732	3.393464
430	1518.354	0.819525	1.639050	3.278100
440	1544.169	0.792353	1.584707	3.169414
450	1569.772	0.766718	1.533436	3.066871
460	1595.169	0.742498	1.484995	2.969991
470	1620.366	0.719585	1.439170	2.878339

Altitude (km)	Slant length (km)	Data rate (Mb/s) E_b/N_o = 10.3 dB		
		Pt = 0.5 Watts	Pt = 1 Watts	Pt = 2 Watts
480	1645.37	0.697881	1.395762	2.791524
490	1670.184	0.677298	1.354596	2.709192
500	1694.814	0.657755	1.315510	2.631020
510	1719.265	0.639179	1.278358	2.556717
520	1743.542	0.621504	1.243007	2.486015
530	1767.648	0.604668	1.209335	2.418671
540	1791.589	0.588616	1.177232	2.354463
550	1815.367	0.573297	1.146594	2.293187
560	1838.988	0.558664	1.117328	2.234656
570	1862.454	0.544675	1.089349	2.178699
580	1885.77	0.531289	1.062578	2.125157
590	1908.939	0.518471	1.036942	2.073883
600	1931.964	0.506186	1.012372	2.024745
610	1954.849	0.494404	0.988808	1.977616
620	1977.596	0.483096	0.966191	1.932383
630	2000.21	0.472234	0.944468	1.888937
640	2022.691	0.461795	0.923590	1.847180
650	2045.044	0.451755	0.903510	1.807020
660	2067.272	0.442093	0.884185	1.768371
670	2089.376	0.432788	0.865576	1.731152
680	2111.359	0.423823	0.847645	1.695291
690	2133.224	0.415179	0.830358	1.660716
700	2154.974	0.406841	0.813682	1.627363
710	2176.609	0.398793	0.797586	1.595172
720	2198.134	0.391021	0.782042	1.564085

Altitude (km)	Slant length (km)	Data rate (Mb/s) E_b/N_o = 10.3 dB		
		Pt = 0.5 Watts	Pt = 1 Watts	Pt = 2 Watts
730	2219.549	0.383512	0.767024	1.534049
740	2240.857	0.376253	0.752507	1.505013
750	2262.06	0.369233	0.738466	1.476931
760	2283.16	0.362440	0.724880	1.449759
770	2304.158	0.355864	0.711728	1.423455
780	2325.058	0.349495	0.698990	1.397980
790	2345.86	0.343324	0.686648	1.373297
800	2366.566	0.337343	0.674685	1.349370
810	2387.178	0.331542	0.663084	1.326169
820	2407.698	0.325915	0.651830	1.303660
830	2428.127	0.320454	0.640908	1.281816
840	2448.467	0.315152	0.630304	1.260608
850	2468.719	0.310003	0.620005	1.240010
860	2488.884	0.304999	0.609999	1.219998
870	2508.965	0.300137	0.600273	1.200547
880	2528.963	0.295409	0.590818	1.181635
890	2548.879	0.290811	0.581621	1.163242
900	2568.713	0.286337	0.572674	1.145347
910	2588.469	0.281983	0.563965	1.127931
920	2608.146	0.277744	0.555488	1.110976
930	2627.747	0.273616	0.547232	1.094464
940	2647.272	0.269595	0.539189	1.078379
950	2666.722	0.265676	0.531353	1.062705
960	2686.099	0.261857	0.523714	1.047428
970	2705.404	0.258133	0.516267	1.032534

Altitude (km)	Slant length (km)	Data rate (Mb/s) E_b/N_o = 10.3 dB		
		Pt = 0.5 Watts	Pt = 1 Watts	Pt = 2 Watts
980	2724.638	0.254502	0.509004	1.018007
990	2743.801	0.250959	0.501918	1.003837
1000	2762.896	0.247502	0.495005	0.990010
1010	2781.922	0.244129	0.488257	0.976514
1020	2800.882	0.240835	0.481669	0.963339
1030	2819.775	0.237618	0.475236	0.950472
1040	2838.603	0.234476	0.468953	0.937905
1050	2857.367	0.231407	0.462814	0.925627
1060	2876.068	0.228407	0.456815	0.913629
1070	2894.707	0.225475	0.450951	0.901902
1080	2913.284	0.222609	0.445218	0.890436
1090	2931.8	0.219806	0.439612	0.879224
1100	2950.256	0.217065	0.434129	0.868258
1110	2968.654	0.214383	0.428765	0.857530
1120	2986.993	0.211758	0.423516	0.847032
1130	3005.275	0.209190	0.418379	0.836758
1140	3023.5	0.206675	0.413351	0.826701
1150	3041.669	0.204214	0.408427	0.816854
1160	3059.783	0.201803	0.403606	0.807211
1170	3077.842	0.199442	0.398883	0.797767
1180	3095.848	0.197128	0.394257	0.788514
1190	3113.8	0.194862	0.389724	0.779448
1200	3131.7	0.192641	0.385281	0.770563
1210	3149.548	0.190464	0.380927	0.761854
1220	3167.345	0.188329	0.376658	0.753317

Altitude (km)	Slant length (km)	Data rate (Mb/s) E_b/N_o = 10.3 dB		
		Pt = 0.5 Watts	Pt = 1 Watts	Pt = 2 Watts
1230	3185.092	0.186236	0.372473	0.744945
1240	3202.789	0.184184	0.368368	0.736736
1250	3220.436	0.182171	0.364342	0.728684
1260	3238.035	0.180196	0.360392	0.720784

Appendix B : Input satellite images

The input images are from NASA's Earth observatory website http://earthobservatory.nasa.gov/ (Przyborski 2010)

1. Input image-1

Figure B-1 Input image-1.

Name	:	ISS020-E-28072_lrg.jpg
Acquired	:	5th August 2009
Instrument	:	Nikon D3 digital camera fitted with a 400 mm lens by ISS Crew.

The information of this image on NASA, earth observation website informs that *"The port city of Sevastopol is located in southernmost Ukraine on the Crimean Peninsula. The city is an important naval base due to the numerous inlets and bays along the coastline. The main economy of the city is based on trade and shipbuilding, but Sevastopol is also a popular tourist and resort destination for visitors from the Commonwealth of Independent Countries (formed from former Soviet Republics).*

This astronaut photograph highlights the jagged coastline of the southern Crimean Peninsula and the various docking areas of Sevastopol. The image was taken by the Expedition 20 crew. The image in this article has been cropped and enhanced to improve contrast. Lens artifacts have been removed".

2. Input image-2

Figure B-2 Input image-2.

Name : sarychev_ali_2010162_lrg.jpg
Acquired : 11th June 2010

The information of this image on NASA, earth observation website informs that "*Almost a year after it erupted spectacularly, Sarychev Volcano showed more modest signs of activity. The Advanced Land Imager (ALI) on NASA's Earth Observing-1 (EO-1) satellite captured this natural-color image on June 11, 2010. Apparent steam fills Sarychev's summit crater, forming an almost perfect circle of white. A much fainter, barely discernible plume blows away from the summit toward the east. Sarychev Peak is a stratovolcano. Reachng a height of 1,496 meters (4,908 feet), it forms the highest point on Matua Island. This volcano ranks among the most active in the Kuril Islands, which extend southwestward from the Kamchatka Peninsula.*
NASA Earth Observatory image created by Jesse Allen, using EO-1 ALI data provided courtesy of the NASA EO-1 team. Caption by Michon Scott."

3. Input image-3

Figure B-3 Input image-3.

Name : manam_ali_2009179_lrg.jpg.jpg
Acquired : 28th June 2009

The information of this image on NASA, earth observation website informs that
"Manam Volcano, just off the coast of mainland Papua New Guinea, released a faint plume on June 28, 2009. The Advanced Land Imager (ALI) onboard NASA's Earth Observing-1 (EO-1) satellite captured this photo-like image of the volcano the same day. Bright white clouds hover over the volcano's summit. Clouds often collect over peaks, but these clouds could result from water vapor released by the volcano. Slightly darker in color, a pale blue-gray plume blows west-northwest from the summit and over the Bismarck Sea.
Manam is one of Papua New Guinea's most active volcanoes, and it has occasionally caused casualties, including 13 deaths from a pyroclastic flow in December 1996, and four deaths from a mudflow in March 2007. Large eruptions in late 2004 forced the evacuation of the entire island.NASA image created by Jesse Allen, using EO-1 ALI data provided courtesy of the NASA EO-1 Team. Caption by Michon Scott."

4. Input image-4

Figure B-4 Input image-4.

Name : Soufriere Hills Volcano Resumes Activity.jpg

Acquired : 29 December 2009

The information of this image on NASA, earth observation website informs that

" *In the waning days of 2009 and the first days of 2010, the lava dome on the summit of Soufrière Hills Volcano continued to grow rapidly. As the dome rises, rocks and debris can break off, cascading down the river valleys and gullies that radiate from the summit. These pyroclastic flows are among the major hazards created by Soufrière Hills. This natural-color satellite image shows the major drainages on the southern and eastern sides of Soufrière Hills. Tan deposits from volcanic flows fill the valleys, the product of almost 15 years of intermittent activity at the volcano. Green vegetation survives on ridges between valleys. The Advanced Land Imager (ALI) aboard NASA'sEarth Observing-1 (EO-1) satellite acquired the image on December 29, 2009. Montserrat is one of the Lesser Antilles Islands, an archipelago in the Caribbean Sea, north of South America. Many of the islands are volcanic, and their location roughly traces the edge of the Caribbean Plate along its boundary with the tectonic plates (North and South American Plates) beneath the Atlantic Ocean. The Caribbean Plate is overriding the North American Plate. As the plates collide, the mantle of the overriding Caribbean Plate melts, generating magma that rises to the surface and feeds Soufrière Hills and other volcanoes in the Lesser Antilles.NASA Earth Observatory image by Robert Simmon, using EO-1 ALI data provided by the NASA EO-1 team.Caption by Robert Simmon"*

5. Input image-5

Figure B-5 Input image-5.

Name : madagascar_ali_2012050_lrg..jpg

Acquired : 19 February 2012

The information of this image on NASA, earth observation website informs that

"In February 2012, Tropical Cyclone Giovanna made landfall in eastern Madagascar, packing winds of 125 knots (230 kilometers per hour) and heavy rains in excess of 250 millimeters (10 inches) along the coast. On February 20, the Integrated Regional Information Networks reported that Giovanna had damaged or destroyed thousands of homes in Madagascar and killed at least 23 people.The Advanced Land Imager (ALI) on NASA's Earth Observing-1 (EO-1) satellite captured this natural-color image of the sediment-choked Onibe River on February 19, 2012. The river appears muddy brown throughout this landscape, and delivers a thick plume of sediment to the Indian Ocean. The Onibe River arises in the highlands of Madagascar's interior and empties into the ocean just north of the coastal town of Mahavelona (also known as Foulpointe). The river lies along the track Giovanna followed when it came ashore. Giovanna's heavy rains spurred equally heavy runoff into the Onibe River. Cyclones rank among the most frequent natural hazards for Madagascar. After coming ashore in mid-February, Cyclone Giovanna blew westward over the island, traveled southward through the Mozambique Channel, then curved back to the east, skirting Madagascar's southern shore. NASA Earth Observatory image created by Jesse Allen and Robert Simmon, using EO-1 ALI data provided courtesy of the NASA EO-1 team. Caption by Michon Scott"

6. Input image-6

Figure B-6 Input image-6.

Name : oman_tmo_2008329_lrg.jpg
Acquired : 24th November 2008

The information of this image on NASA, earth observation website informs that "*Much of Oman's landscape is low-lying, hot, and dry. Its extensive sand seas stretch westward into the neighboring countries of Saudi Arabia and United Arab Emirates. Northern Oman, however, contains rugged mountains and some of the highest elevations in eastern Arabia. In this region, relatively cool, moist weather has given rise to a richly vegetated landscape. In northernmost Oman, north of the United Arab Emirates, steep-sloped mountains rise from ocean waters. The Moderate Resolution Imaging Spectroradiometer (MODIS) on NASA's Terra satellite captured this image of northern Oman. In contrast to the orange-tinted sands in the west, the mountains of northern Oman range in color from pale beige to deep green-brown.*

In contrast to the desert environment in the south, the mountains of northeastern Oman regularly receive precipitation, including fierce electrical storms and snow, from December to March. Temperatures rise between April and September, but Indian Monsoon winds can also bring the occasional thunderstorm. The limestone cliffs of this region support rich soils that give rise to varied vegetation. NASA image courtesy MODIS Rapid Response, NASA Goddard Space Flight Center. Caption by Michon Scott. Instrument:Terra - MODIS"

7. Input image-7

Figure B-7 Input image-7.

Name	:	SS013-E-14843_lrg.jpg
Acquired	:	May 6, 2006

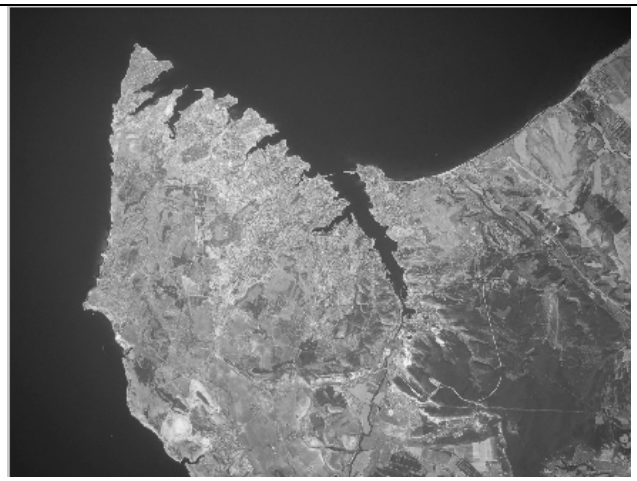
The information of this image on NASA, earth observation website informs that *"While the Great Lakes region of North America is well known for its importance to shipping between the United States, Canada, and the Atlantic Ocean, it is also the location of an impressive structure in the continent's bedrock: the Michigan Basin. Formed during the Paleozoic Era (approximately 540-250 million years ago) the Basin looks much like a large bullseye defined by the arrangement of exposed rock layers that all tilt inwards, forming a huge bowl-shaped structure. While this "bowl" is not readily apparent from the ground, detailed mapping of the rock units on a regional scale revealed the structure to geologists. The outer layers of the Basin include thick deposits of carbonates—rocks containing carbon and oxygen, such as limestone—deposited over millions of years when a shallow sea covered the region. Astronaut photograph ISS013-E-14843 was acquired May 6, 2006, with a Kodak 760C digital camera using an 800 mm lens, and is provided by the ISS Crew Earth Observations experiment and the Image Science & Analysis Group, Johnson Space Center".*

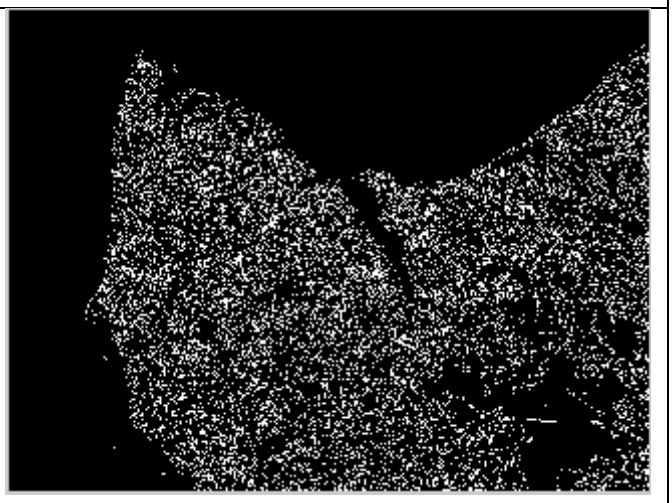
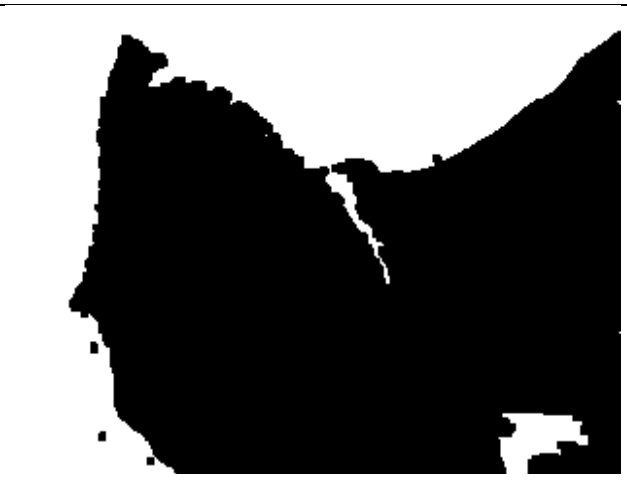
Appendix C : ROI results based on image edge segmentation

The experimental results of the ROI automatic detection based on edge segmentation module. There are 7 experiments.

Table C-1 The results of ROI automatic detection based on edge image compression of image-1.

Input image-1; Name : ISS020-E-28072_lrg.jpg

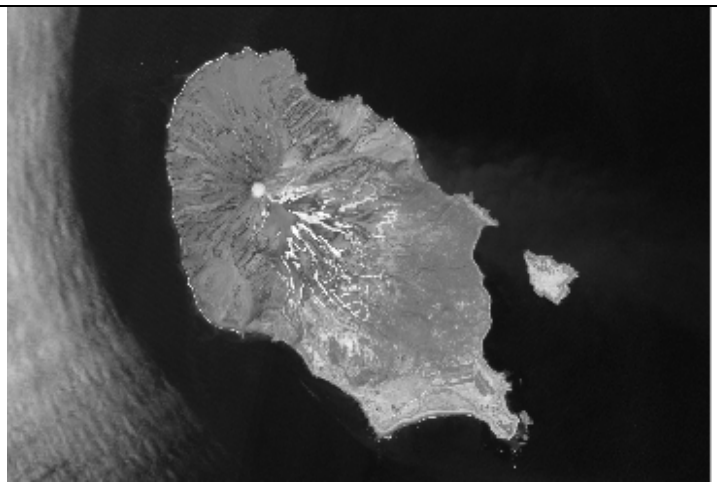
Step testing	Results
1. Original image	
2. Grey scale	

Step testing	Results
3. Edge detection	
4. Dilation	
5. ROI mask	

Step testing	Results
6. ROI Image	

Table C-2 The result of ROI automatic detection based on edge image compression of image-2.

Input image-2; Name : sarychev_ali_2010162_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Edge detection	

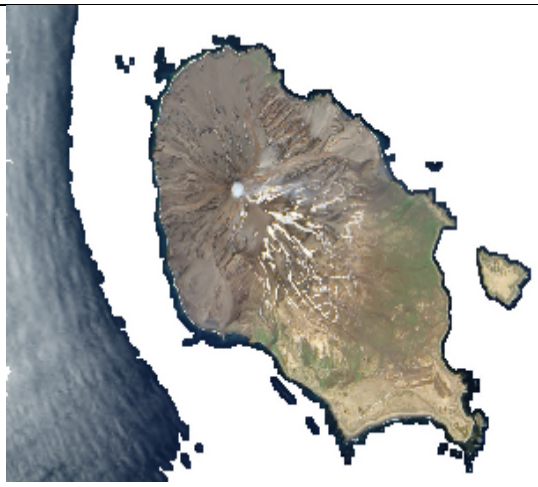
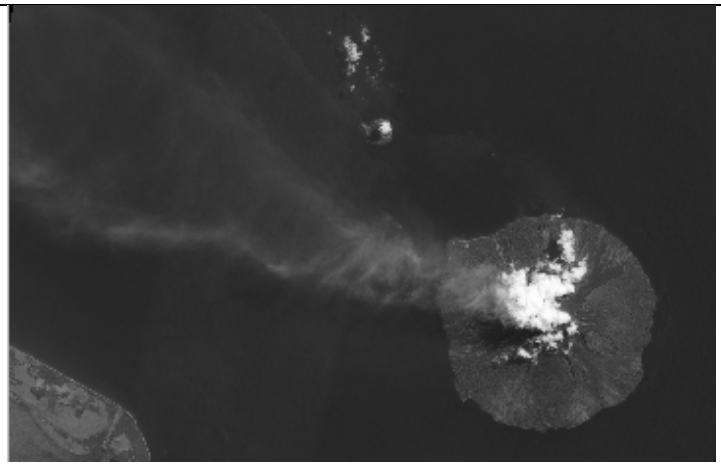
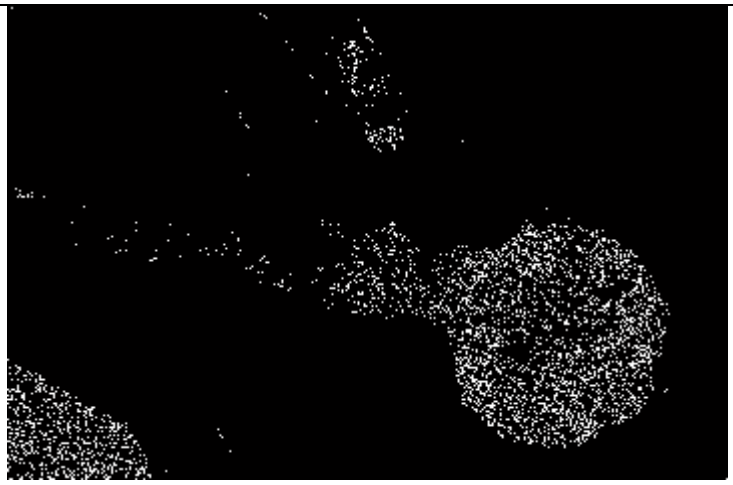
Step testing	Results
4. Dilation	
5. ROI mask	
6. ROI Image	

Table C-3 The result of ROI automatic detection based on edge image compression of image-3.

Input image-3; Name : manam_ali_2009179_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Edge detection	

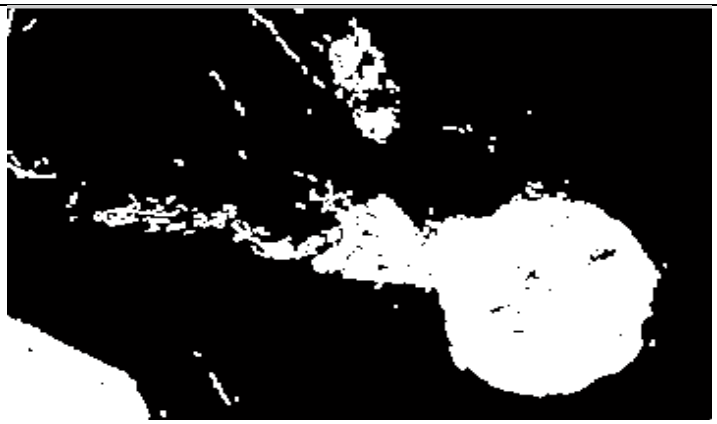
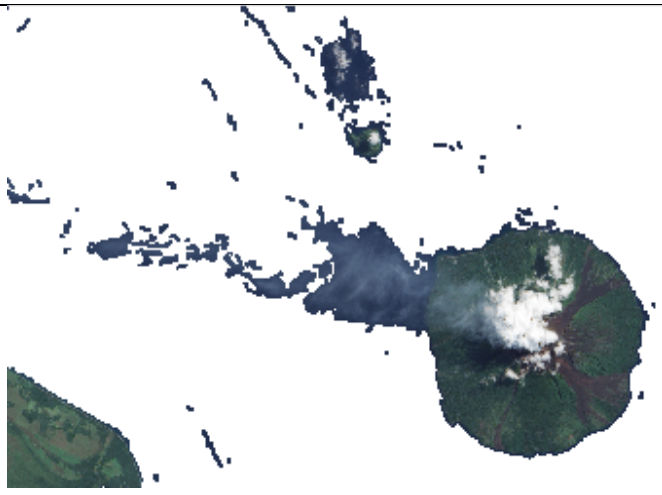
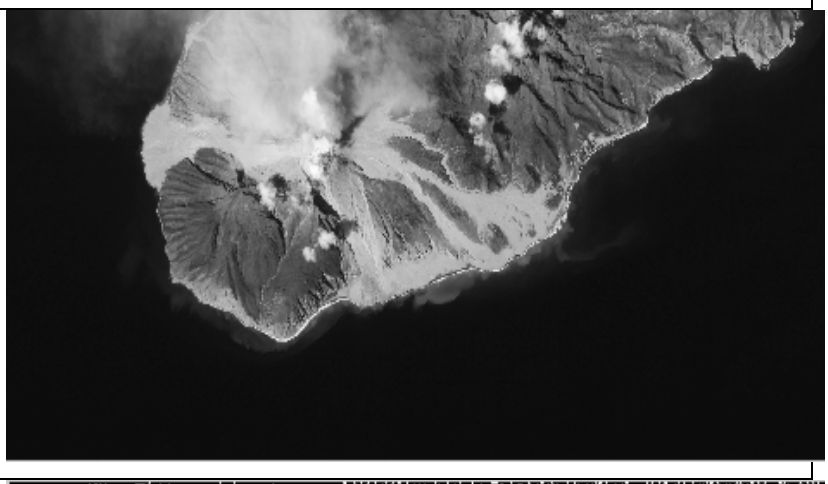
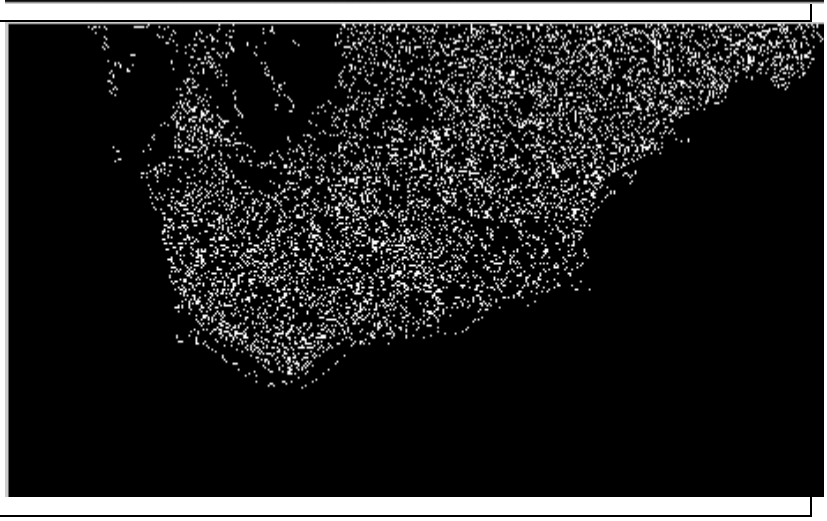
Step testing	Results
4. Dilation	
5. ROI mask	
6. ROI Image	

Table C-4 The result of ROI automatic detection based on edge image compression of image-4.

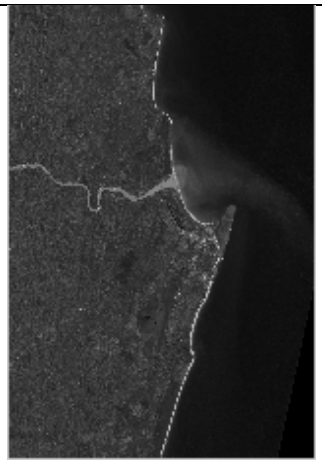
Input image-4; Name : Soufriere Hills Volcano Resumes Activity.jpg

Step testing	Results
1.Original image	
2. Grey scale	
3. Edge detection	

Step testing	Results
4. Dilation	
5. ROI mask	
6. ROI Image	

Table C-5 The result of ROI automatic detection based on edge image compression of image-5.

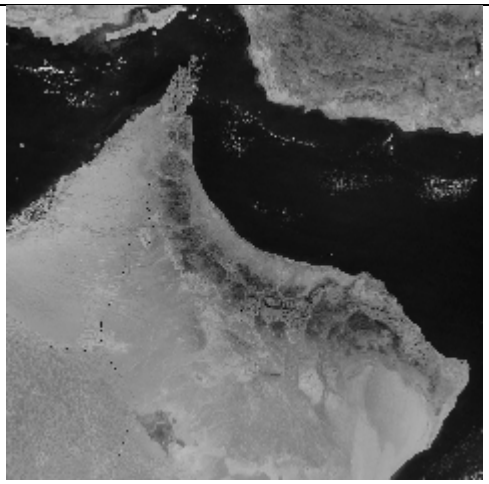
Input image-5; Name : madagascar_ali_2012050_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Edge detection	

Step testing	Results
4. Dilation	
5. ROI mask	
6. ROI Image	

Table C-6 The result of ROI automatic detection based on edge image compression of image-6.

Input image-6; Name : oman_tmo_2008329_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Edge detection	

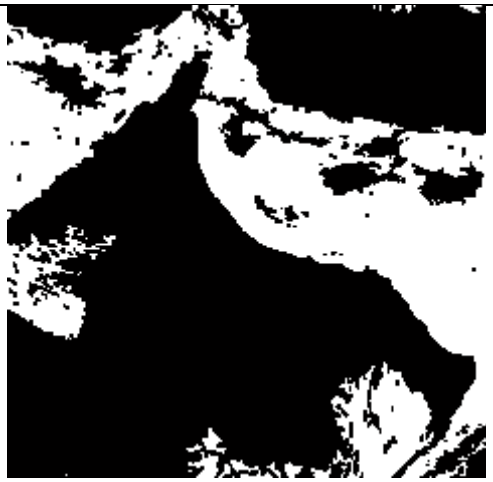
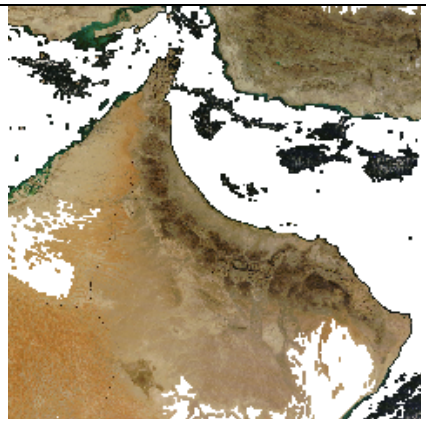
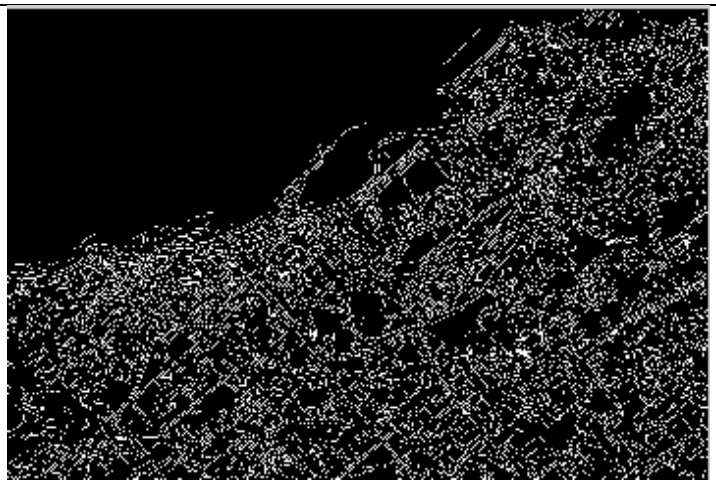
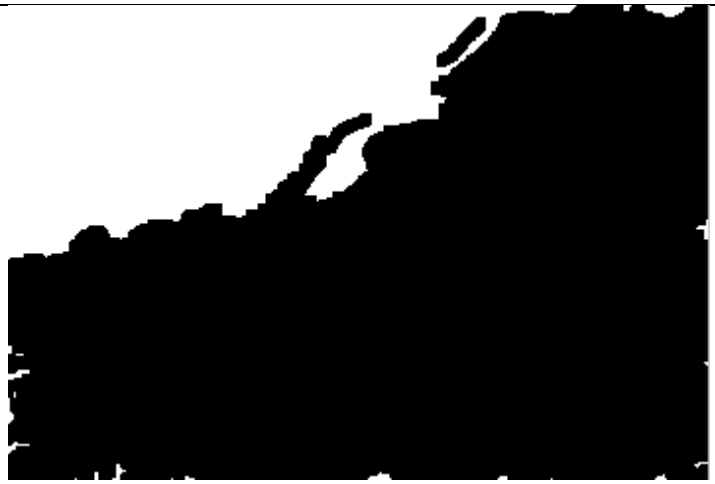
Step testing	Results
4. Dilation	
5. ROI mask	
6. ROI Image	

Table C-7 The result of ROI automatic detection based on edge image compression of image-7.

Input image-7; Name : ISS013-E-14843_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Edge detection	

Step testing	Results
4. Dilation	
5. ROI mask	
6. ROI Image	

Appendix D : ROI results based on image histogram segmentation

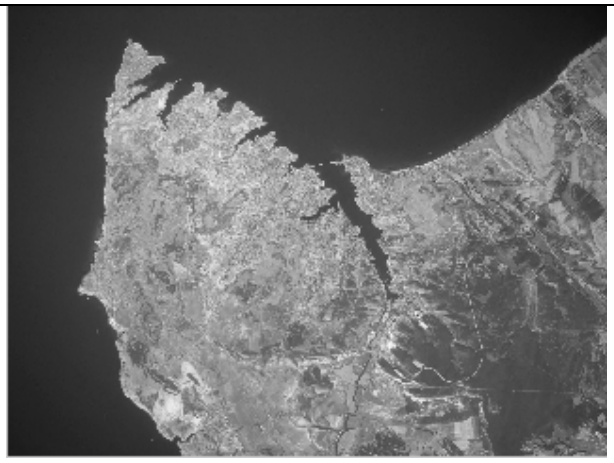
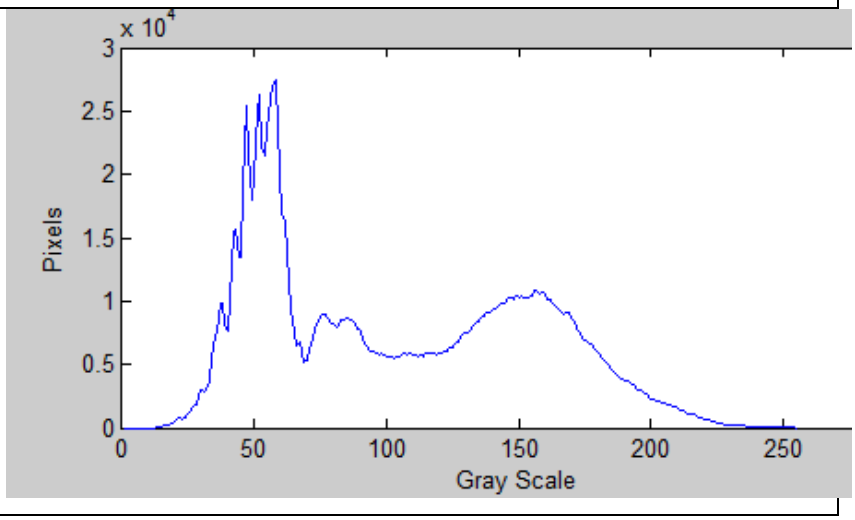
The experimental results of the ROI automatic detection based on histogram segmentation module. There are 7 experiments.

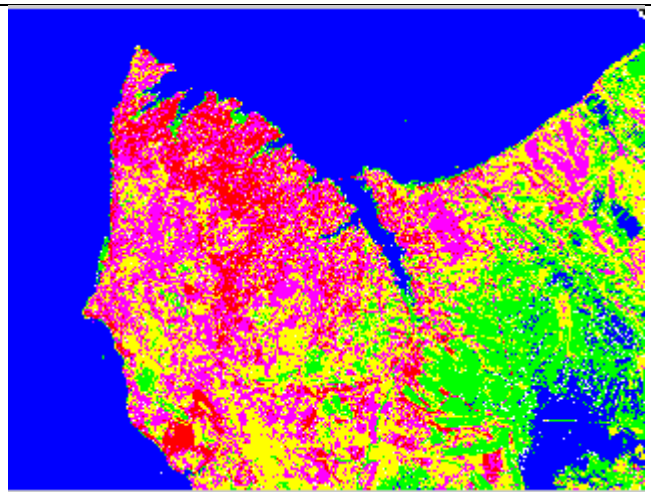
Table D-1.1 The objects pixel value recoding list.

Image 1/ Time test	Ocean	Forest	Agriculture	Urban
1	52	76	120	213
2	52	74	132	183
3	44	79	133	196
4	48	75	150	190
5	58	76	149	179
6	48	101	158	187
7	54	94	133	166
8	43	70	139	182
9	44	89	149	180
10	40	76	162	189
11	45	80	160	234
12	41	78	167	195
13	39	76	158	173
14	37	75	145	182
15	31	74	151	180
16	41	77	150	190
17	48	73	123	187
18	50	75	146	191
19	49	75	141	163
20	53	78	157	196
Average	45.85	78.55	146.15	187.8
Max	58	101	167	234
Min	31	70	120	163

Table D-1.2 The result of ROI automatic detection based on histogram image compression of image-1.

Input image-1; Name : ISS020-E-28072_lrg.jpg

Step testing	Results
1.Original image	
2. Grey scale	
3. Histogram	

Step testing	Results
4. Histogram segmentation	
5. BW	
6. Dilation	

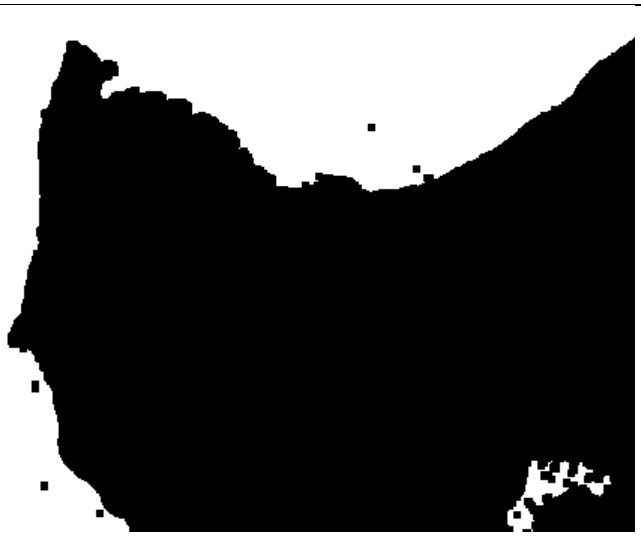
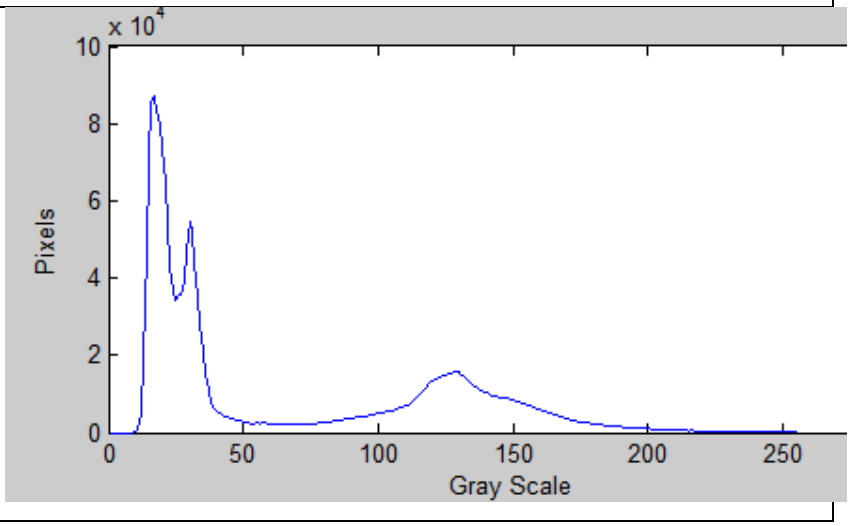
Step testing	Results
7. ROI mask	
8. ROI image	

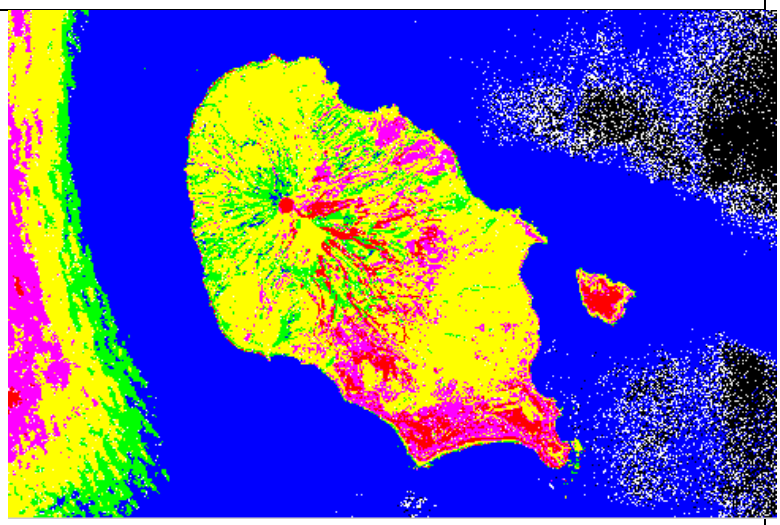
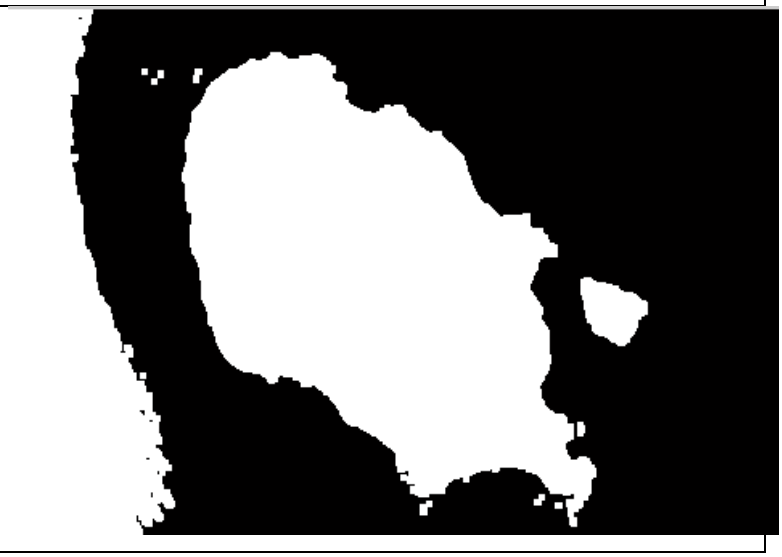
Table D-2.1 The objects pixel value recoding list.

Image2 / Time test	Ocean	Forest	Agriculture	Urban
1	18	125	142	126
2	16	130	125	129
3	12	121	127	156
4	22	118	126	151
5	16	151	131	130
6	15	175	132	127
7	14	151	138	127
8	15	148	183	179
9	16	176	116	129
10	15	141	135	137
11	20	139	122	126
12	13	142	114	135
13	14	118	126	137
14	13	123	182	146
15	17	123	154	137
16	15	123	141	121
17	12	123	136	131
18	15	118	132	137
19	16	118	135	137
20	15	123	127	143
Average	15.45	134.3	136.2	137.05
Max	22	176	183	179
Min	12	118	114	121

Table D-2.2 The result of ROI automatic detection based on histogram image compression of image-2.

Input image-2; Name : sarychev_ali_2010162_lrg.jpg

Step testing	Results
1.Original image	
2. Grey scale	
3. Histogram	

Step testing	Results
4. Histogram segmentation	
5. BW	
6. Dilation	

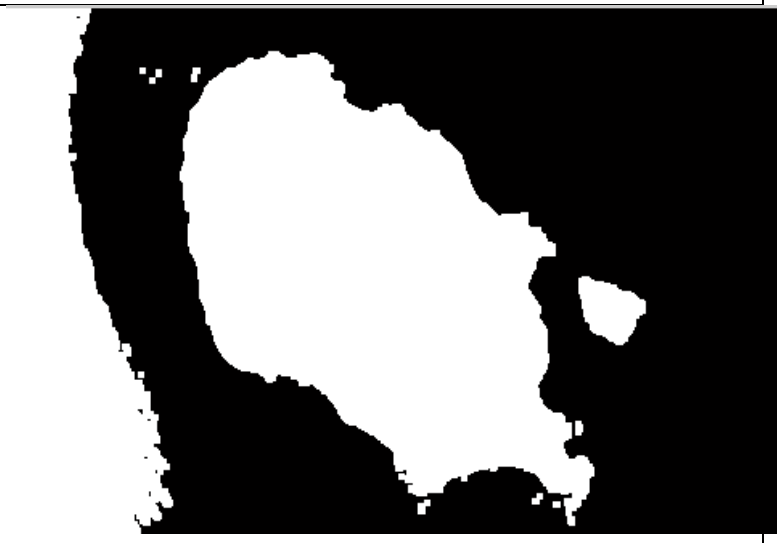
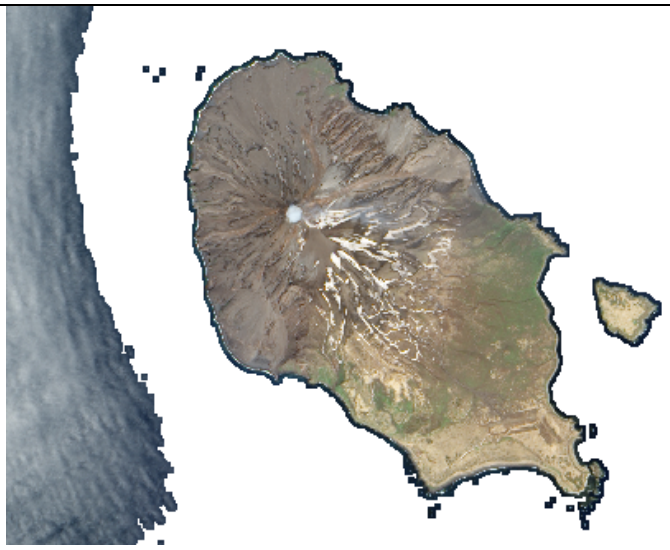
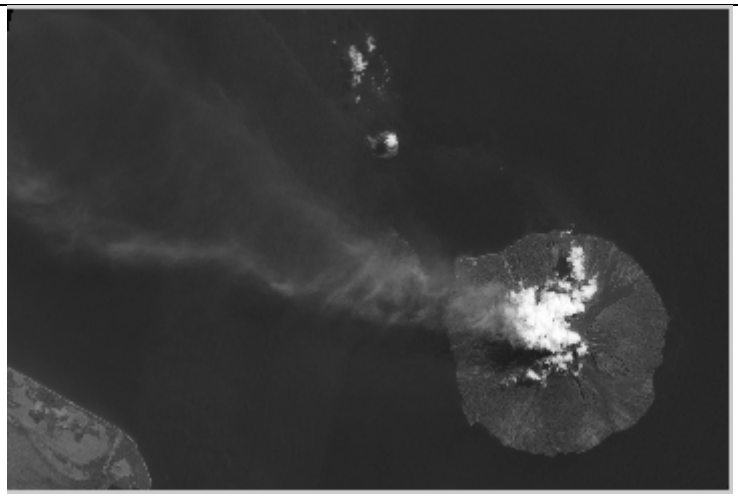
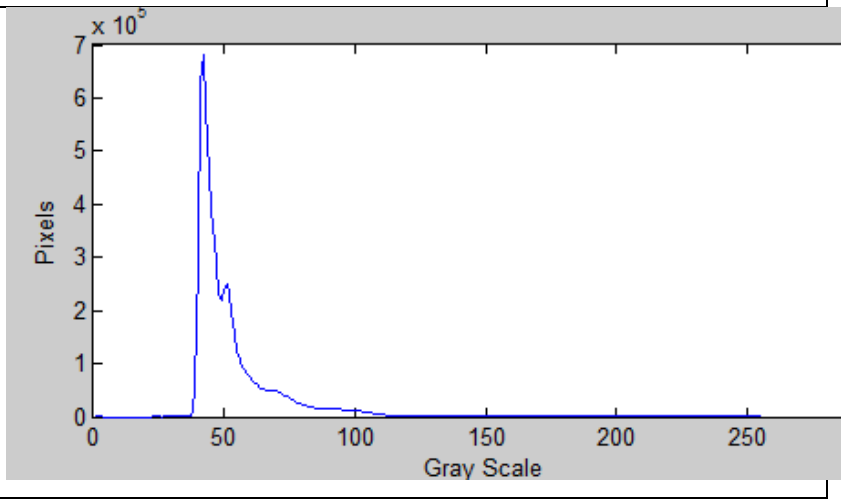
Step testing	Results
7. ROI mask	
8.ROI image	

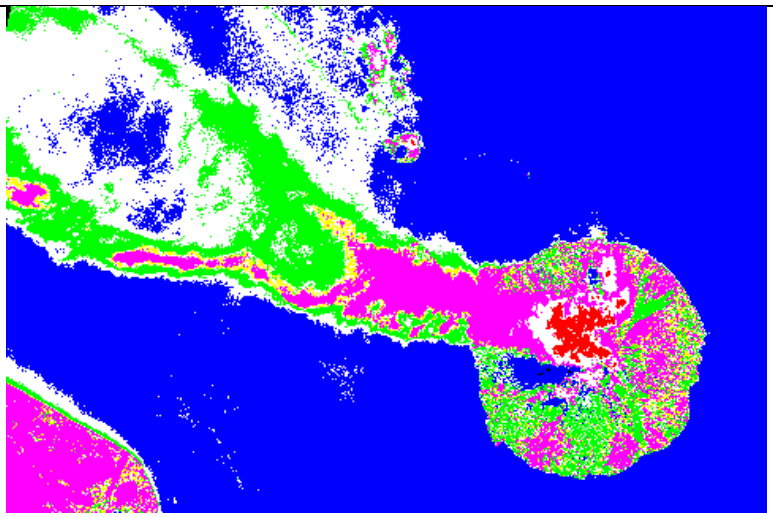
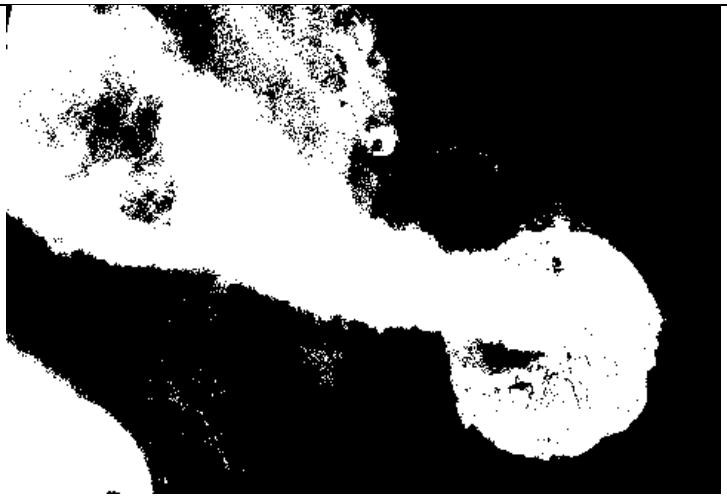
Table D-3.1 The objects pixel value recoding list.

Image3 / Time test	Ocean	Forest	Agriculture	Urban
1	46	63	92	106
2	44	60	99	102
3	43	63	85	102
4	41	68	94	104
5	40	70	99	105
6	43	67	102	103
7	44	64	101	98
8	45	53	95	102
9	45	52	100	100
10	40	67	99	104
11	41	62	97	102
12	44	60	109	103
13	44	60	101	104
14	43	50	104	100
15	45	57	103	101
16	43	58	103	102
17	43	58	90	104
18	47	53	95	100
19	48	57	103	104
20	48	61	102	102
Average	43.85	60.15	98.65	102.4
Max	48	70	109	106
Min	40	50	85	98

Table D-3.2 The result of ROI automatic detection based on histogram image compression of image-3.

Input image-3; Name : manam_ali_2009179_lrg.jpg

Step testing	Results
1.Original image	
2. Grey scale	
3. Histogram	

Step testing	Results
4. Histogram segmentation	
5. BW	
6. Dilation	

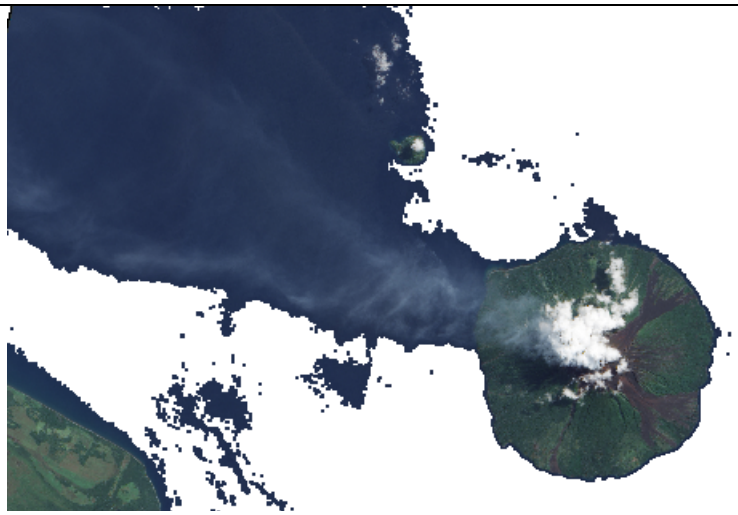
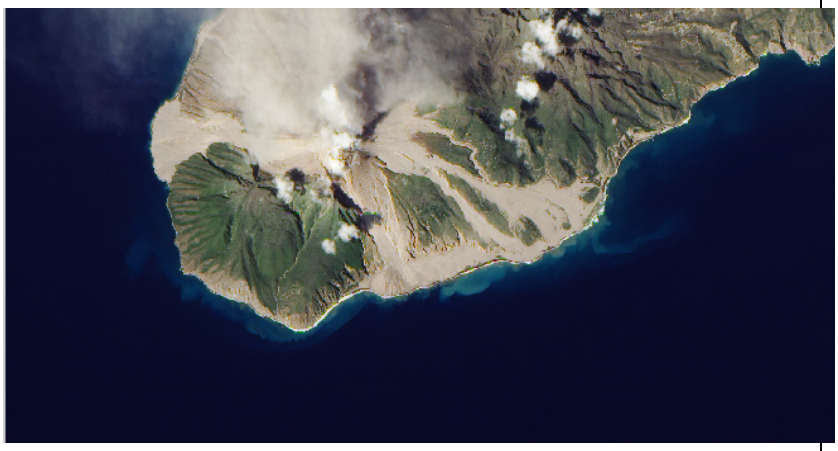
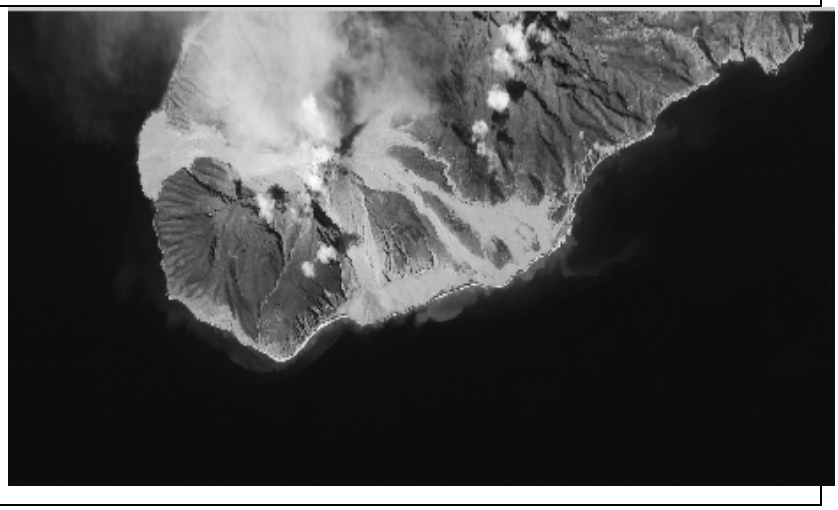
Step testing	Results
7. ROI mask	
8. ROI image	

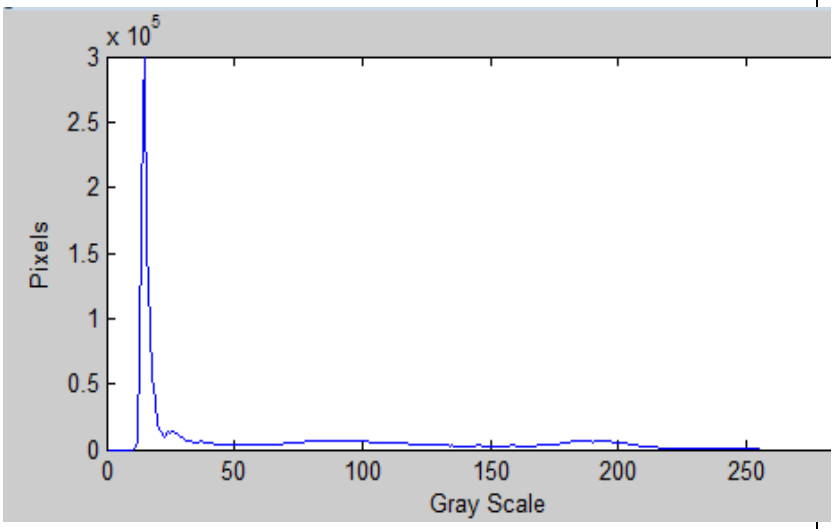
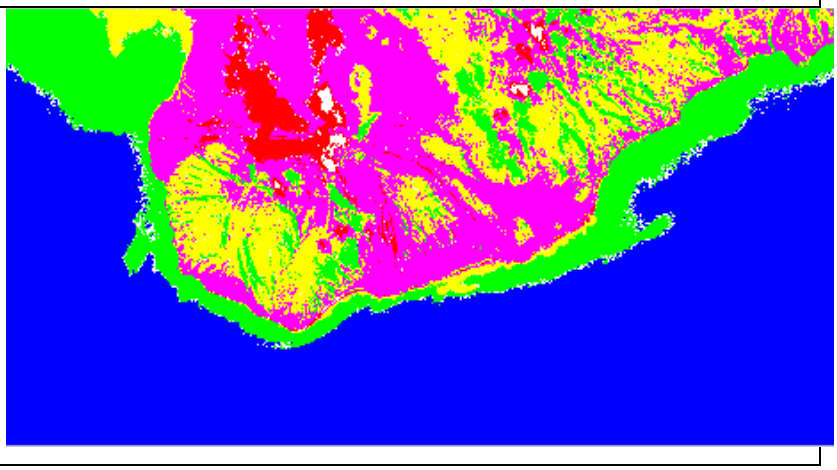
Table D-4.1 The objects pixel value recoding list.

Image4 / Time test	Ocean	Forest	Agriculture	Urban
1	13	71	88	169
2	14	74	101	184
3	13	74	86	184
4	13	83	27	169
5	13	83	84	191
6	12	78	82	143
7	12	76	95	172
8	13	81	68	159
9	14	81	77	180
10	13	92	61	180
11	14	75	101	167
12	15	82	46	185
13	12	69	112	177
14	14	79	112	151
15	12	78	88	175
16	14	77	101	130
17	13	77	86	154
18	14	69	77	177
19	13	84	61	187
20	13	76	85	155
Average	13.2	77.95	81.9	169.45
Max	15	92	112	191
Min	12	69	27	130

Table D-4.2 The result of ROI automatic detection based on histogram image compression of image-4.

Input image-4; Name : Soufriere Hills Volcano Resumes Activity.jpg

Step testing	Results
1. Original image	
2. Grey scale	

Step testing	Results
3. Histogram	
4.Histogram segmentation	
5. BW	

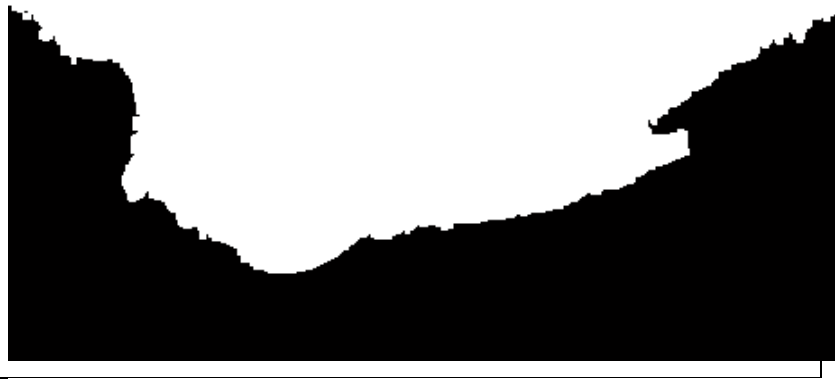
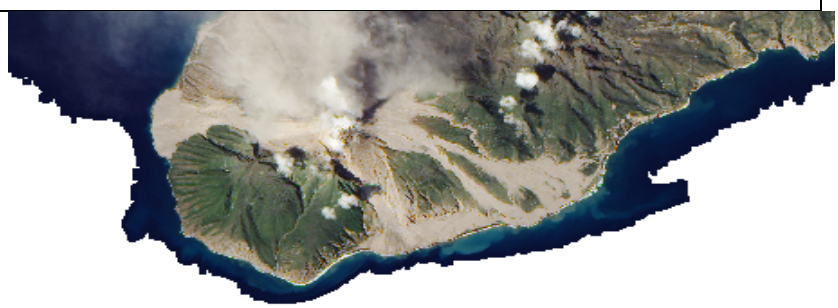
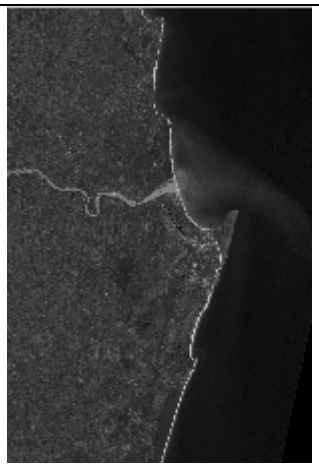
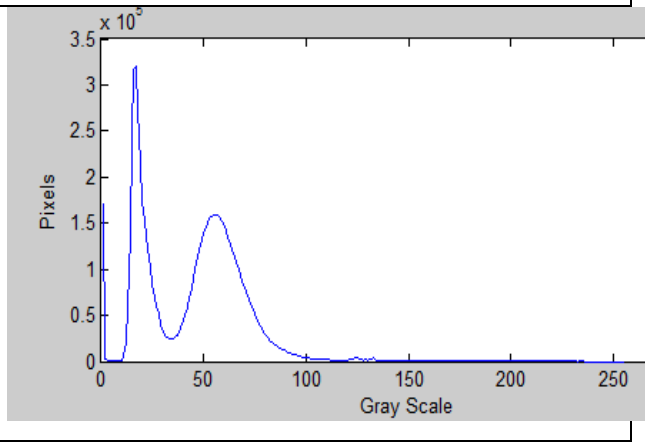
Step testing	Results
6. Dilation	
7. ROI mask	
8. ROI image	

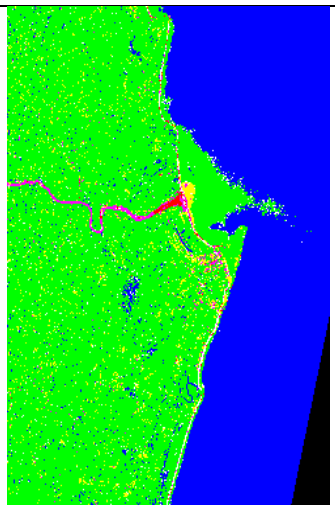
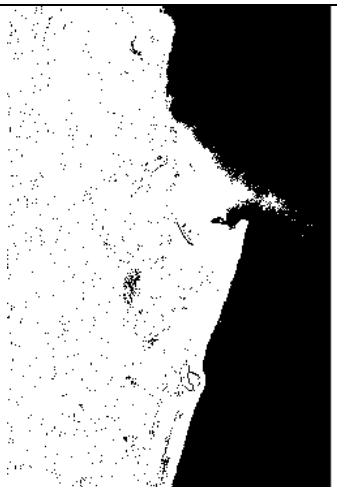
Table D-5.1 The objects pixel value recoding list.

Image5 / Time test	Ocean	Forest	Agriculture	Urban
1	22	60	68	140
2	16	59	68	140
3	19	67	68	81
4	14	50	40	103
5	17	61	40	97
6	19	84	75	60
7	17	62	44	205
8	15	73	68	75
9	16	58	80	109
10	15	72	97	109
11	22	90	56	169
12	15	65	49	98
13	18	68	59	110
14	19	69	68	140
15	17	76	68	204
16	17	57	59	109
17	17	51	68	102
18	18	49	95	110
19	20	74	58	132
20	19	38	69	132
Average	17.6	64.15	64.85	121.25
Max	22	90	97	205
Min	14	38	40	60

Table D-5.2 The result of ROI automatic detection based on histogram image compression of image-5.

Input image-5; Name : madagascar_ali_2012050_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Histogram	

Step testing	Results
4.Histogram segmentation	
5. BW	
6. Dilation	

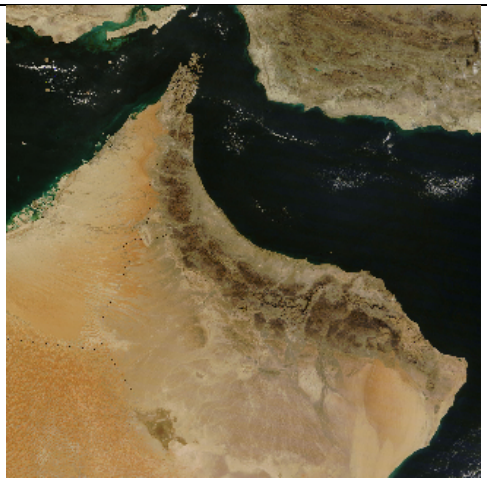
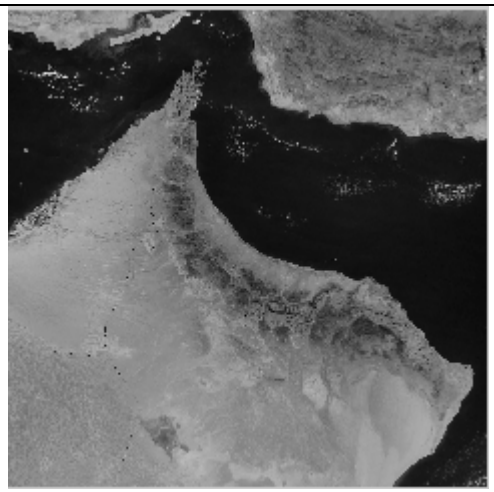
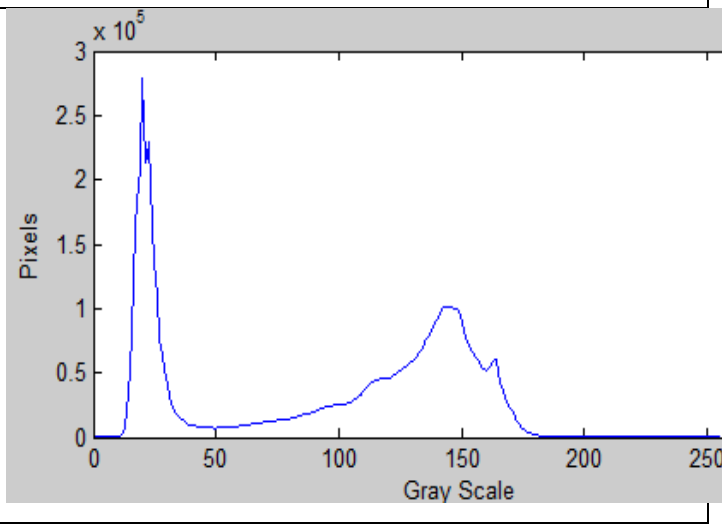
Step testing	Results
7. ROI mask	
8. ROI image	

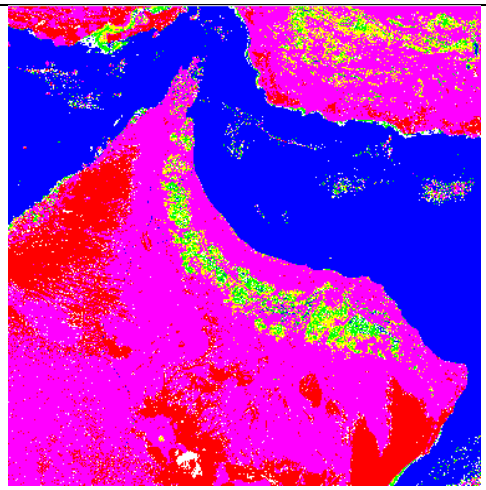
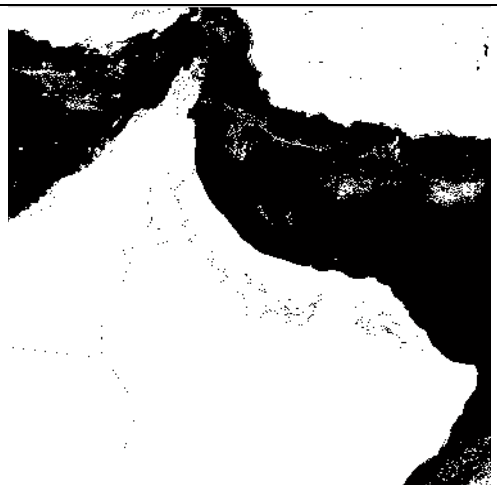
Table D-6.1 The objects pixel value recoding list.

Image6 / Time test	Ocean	Forest	Agriculture	Urban
1	16	37	84	160
2	20	37	105	160
3	19	38	73	169
4	18	38	99	164
5	19	38	107	146
6	18	20	100	166
7	18	28	102	159
8	17	45	85	166
9	17	38	104	164
10	16	20	96	165
11	18	27	121	135
12	16	27	105	161
13	17	20	104	164
14	18	21	90	161
15	17	38	73	165
16	17	38	92	167
17	21	65	90	165
18	19	23	97	162
19	19	25	94	169
20	19	35	103	162
Average	17.95	32.9	96.2	161.5
Max	21	65	121	169
Min	16	20	73	135

Table D-6.2 The result of ROI automatic detection based on histogram image compression of image-6.

Input image-6; Name : oman_tmo_2008329_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Histogram	

Step testing	Results
4.Histogram segmentation	
5. BW	
6. Dilation	

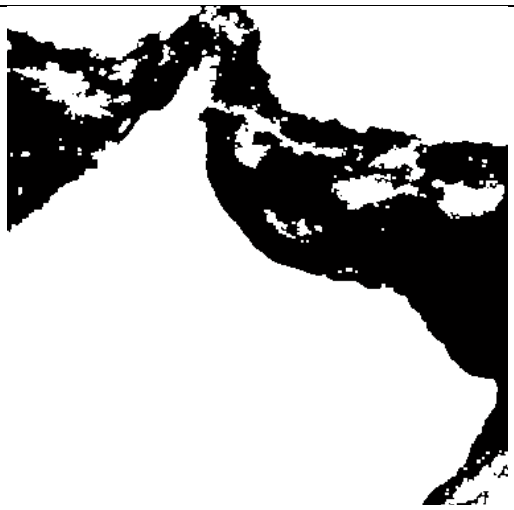
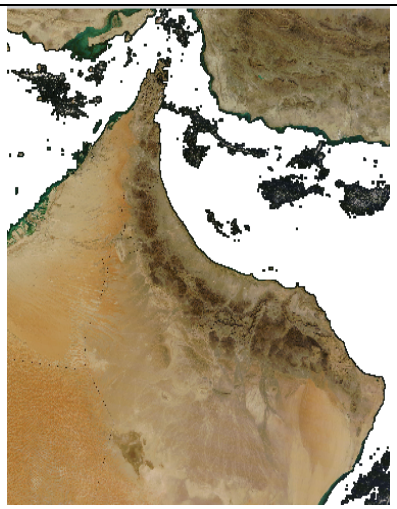
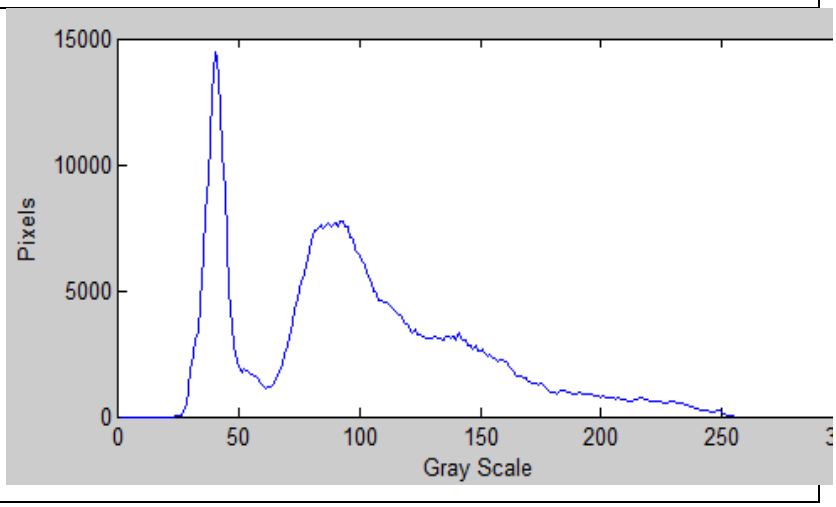
Step testing	Results
7. ROI mask	
8. ROI image	

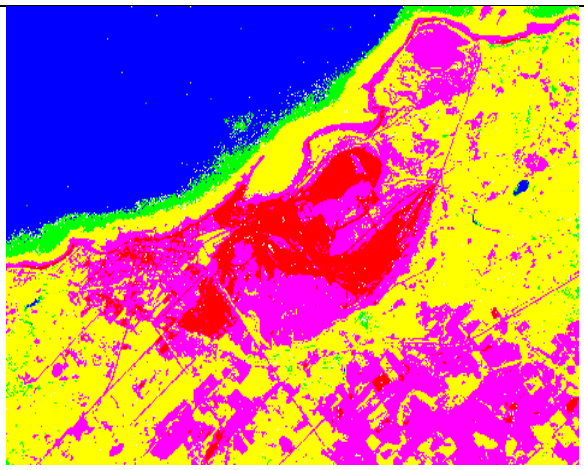
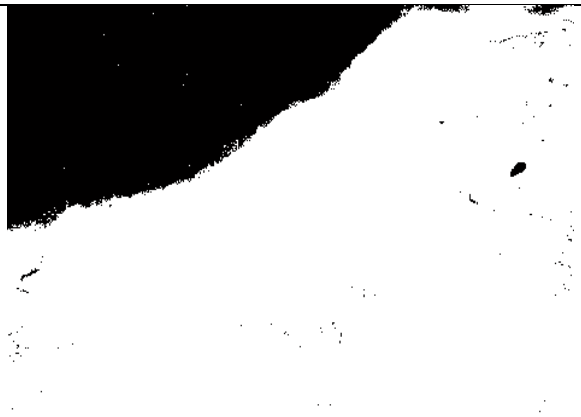
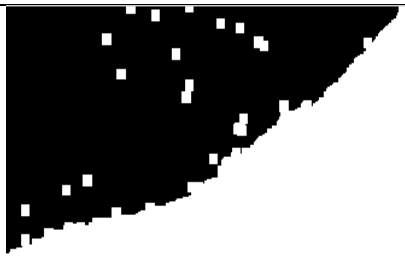
Table D-7.1 The objects pixel value recoding list.

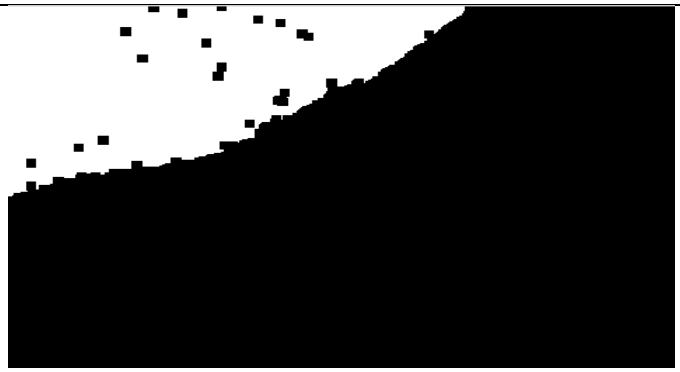
Image7 / Time test	Ocean	Forest	Agriculture	Urban
1	38	71	98	193
2	38	75	104	197
3	37	70	91	199
4	33	69	103	155
5	39	72	99	177
6	37	74	91	149
7	36	73	98	157
8	37	78	99	219
9	40	71	88	211
10	35	64	85	195
11	39	68	90	189
12	35	73	91	129
13	38	69	94	180
14	37	75	90	163
15	39	80	95	160
16	37	74	94	208
17	37	75	91	199
18	39	80	94	202
19	40	82	91	183
20	38	76	95	196
Average	37.45	73.45	94.05	183.05
Max	40	82	104	219
Min	33	64	85	129

Table D-7.2 The result of ROI automatic detection based on histogram image compression of image-7.

Input image-7; Name : ISS013-E-14843_lrg.jpg

Step testing	Results
1. Original image	
2. Grey scale	
3. Histogram	

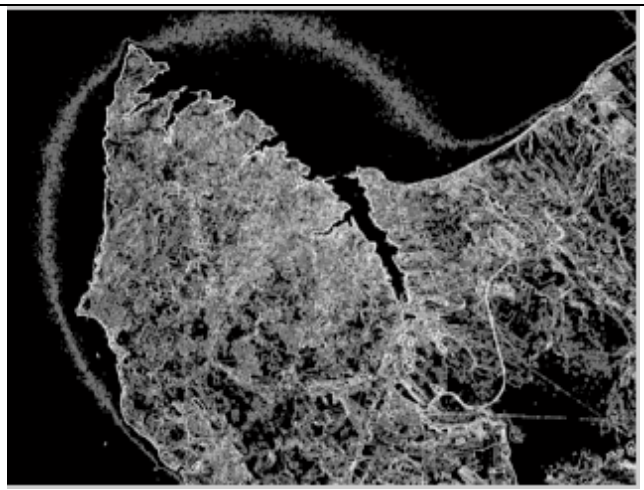
Step testing	Results
4.Histogram segmentation	
5. BW	
6. Dilation	

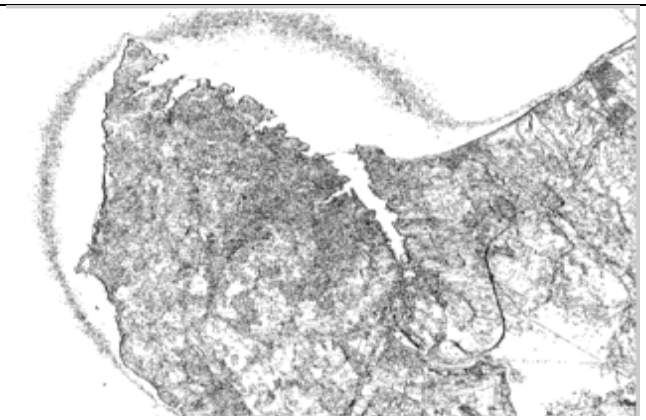
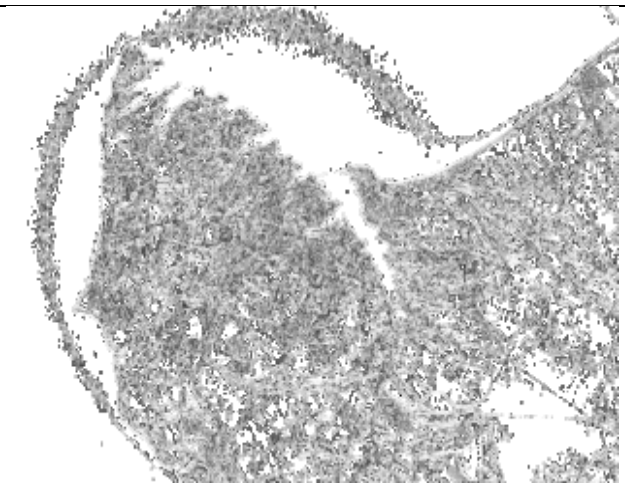
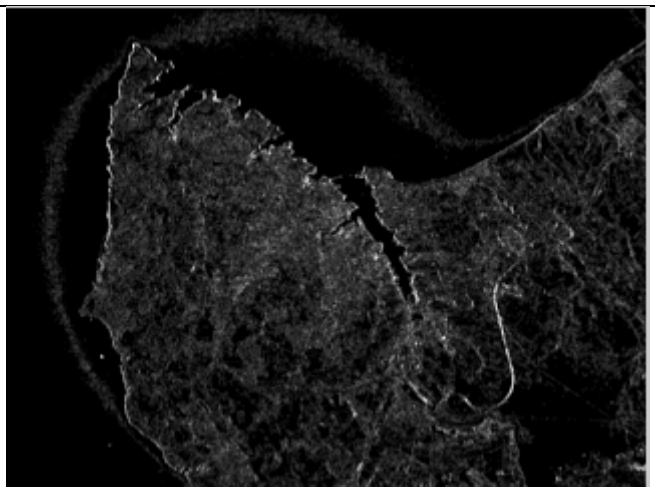
Step testing	Results
7. ROI mask	
8. ROI image	

Appendix E : ROI results based on image texture segmentation

Table E-1 The result of ROI automatic detection based on texture image compression of image-1.

Input image-1; Name : ISS020-E-28072_lrg.jpg

Step testing	Results
1.Original image	
2. Entropy	

Step testing	Results
3. Homogeneity	
4. Correlation	
5. Contrast	

Step testing	Results
6. Energy	
7. BW	
8. Dilation	

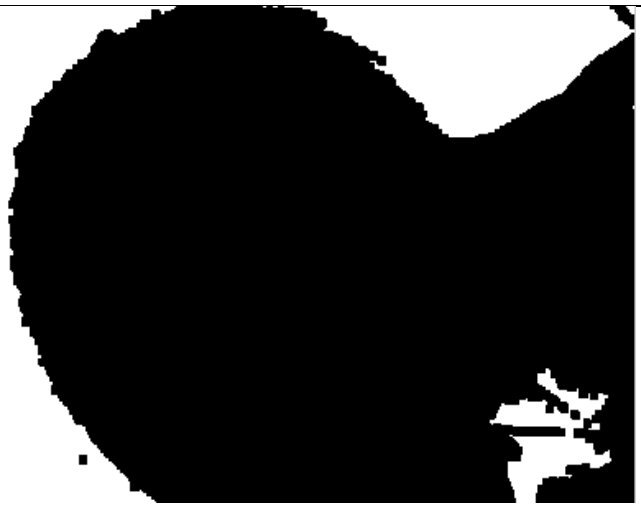
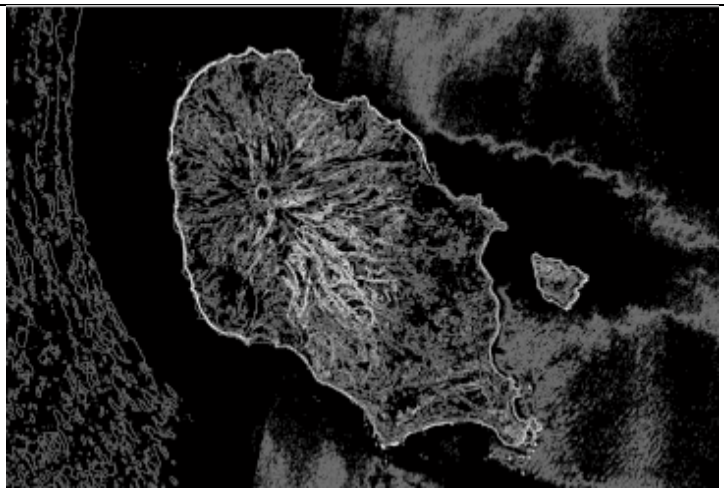
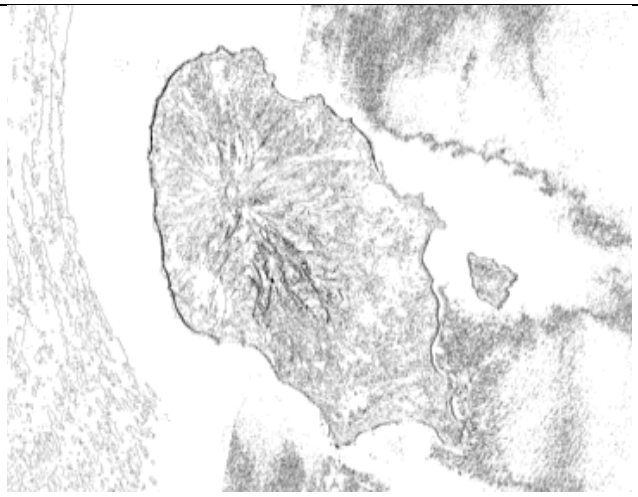
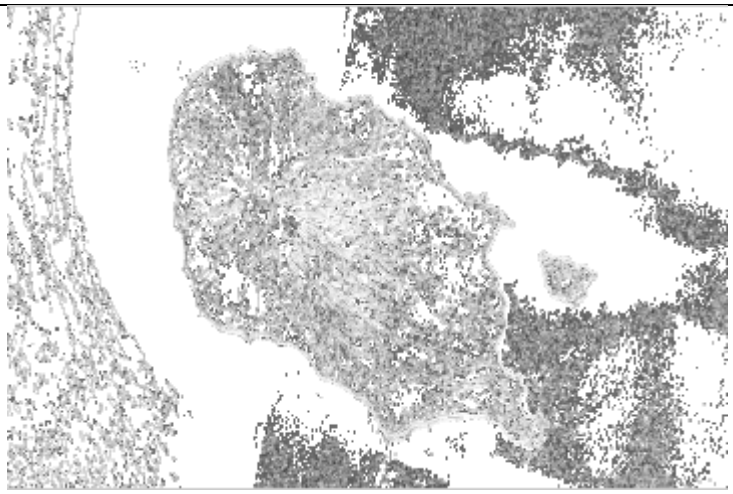
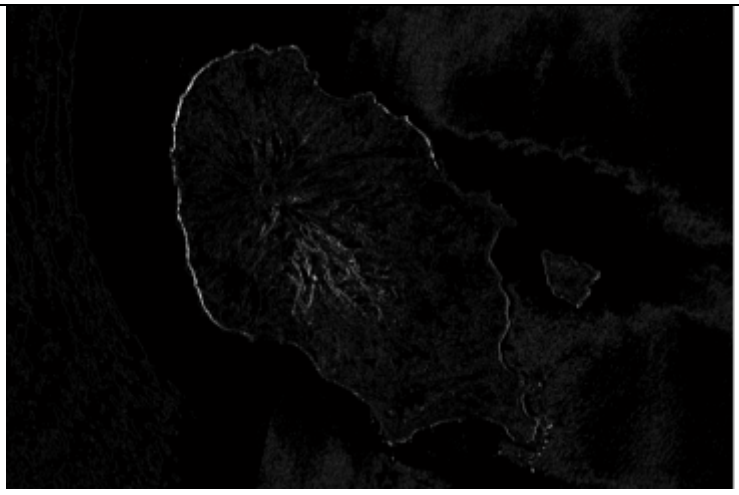
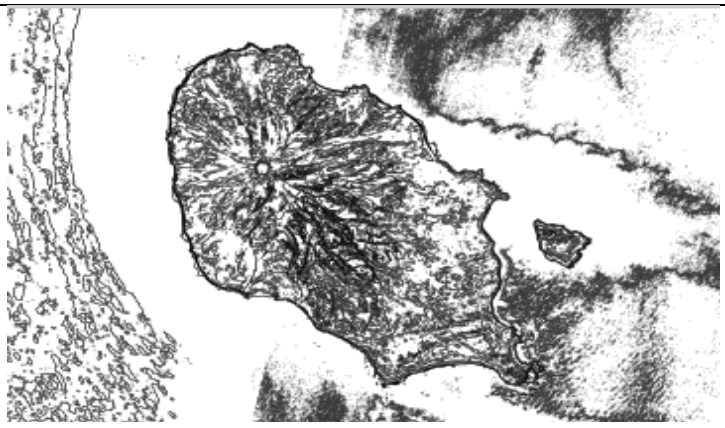
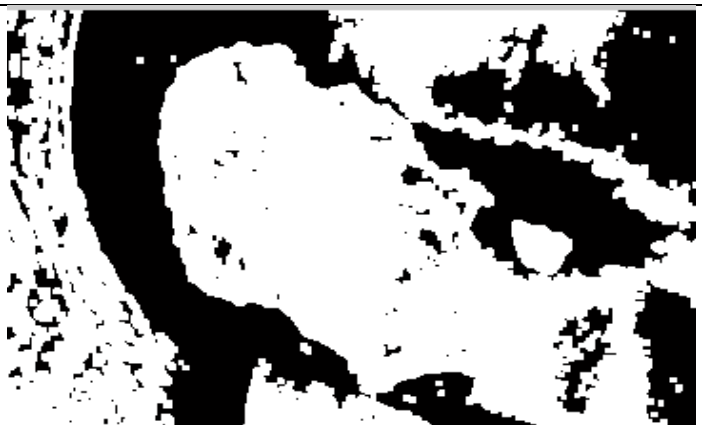
Step testing	Results
9. ROI mask	
10. ROI image	

Table E-2 The result of ROI automatic detection based on texture image compression of image-2.

Input image-2; Name : sarychev_ali_2010162_lrg.jpg

Step testing	Results
1. Original image	
2. Entropy	
3. Homogeneity	

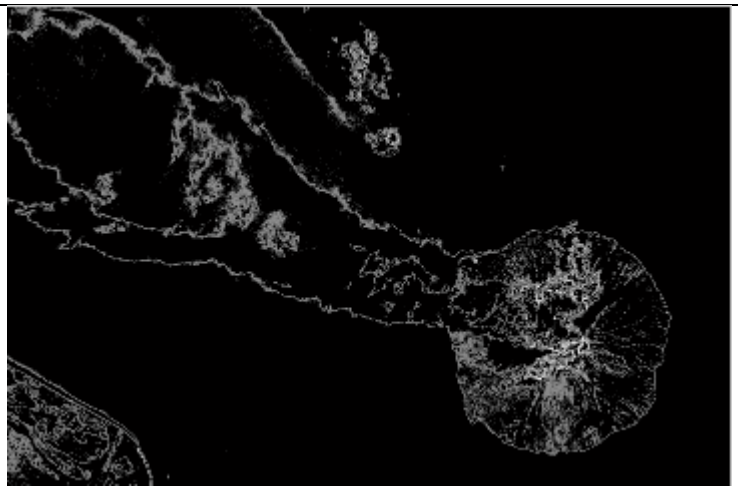
Step testing	Results
4. Correlation	
5. Contrast	
6. Energy	

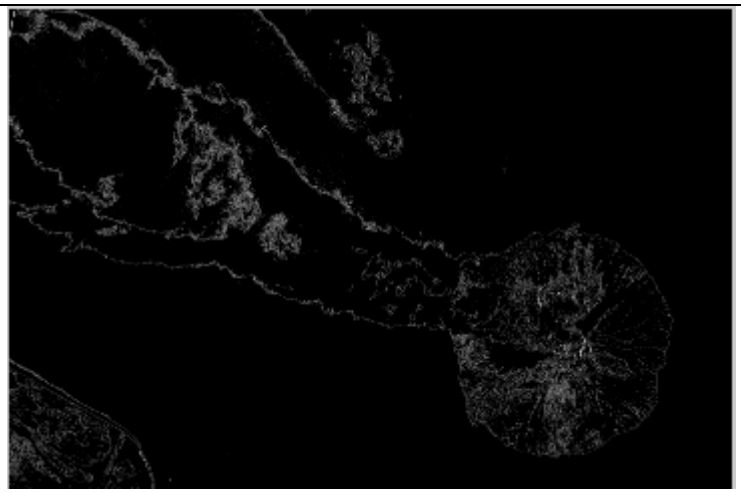
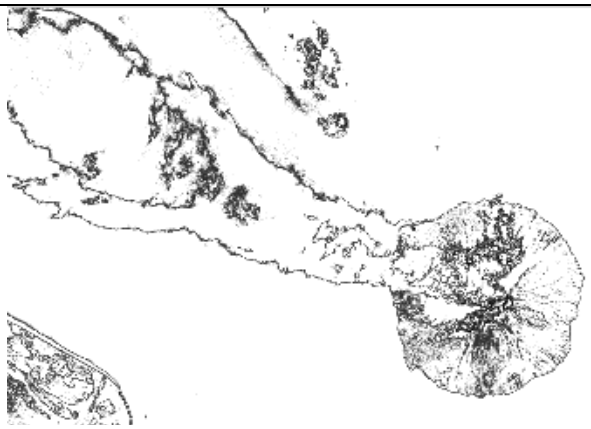
Step testing	Results
7. BW	
8. Dilation	
9. ROI mask	

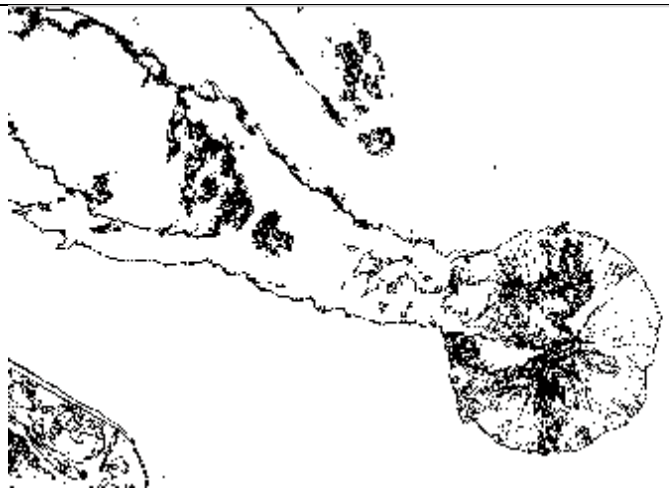
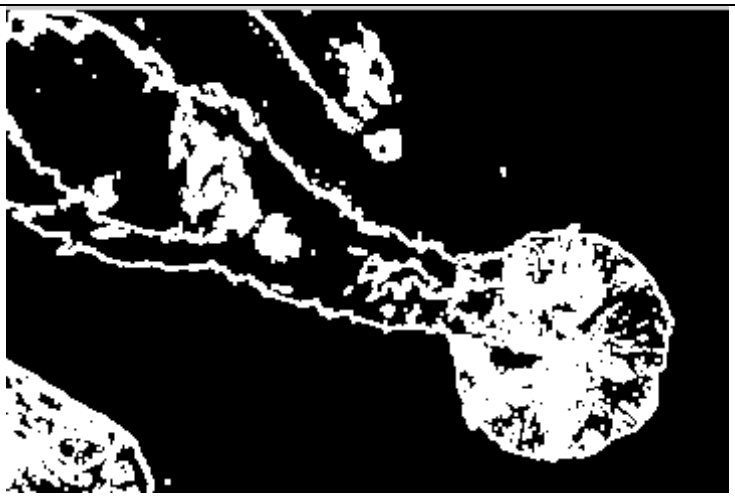
Step testing	Results
10. ROI image	

Table E-3 The result of ROI automatic detection based on texture image compression of image-3.

Input image-3; Name : manam_ali_2009179_lrg.jpg

Step testing	Results
1. Original image	
2. Entropy	
3. Homogeneity	

Step testing	Results
4. Correlation	
5. Contrast	
6. Energy	

Step testing	Results
7. BW	
8. Dilation	
9. ROI mask	

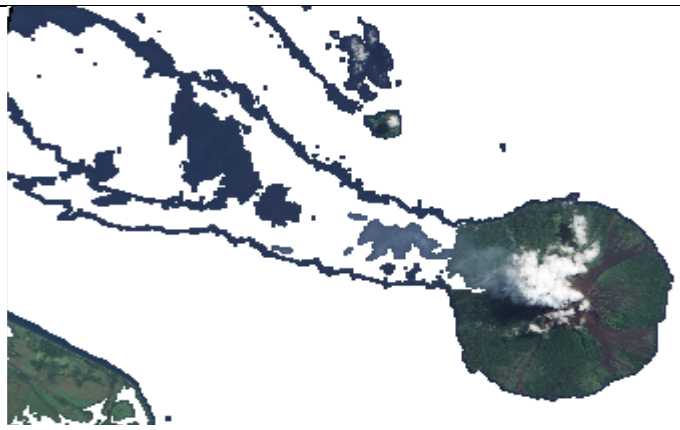
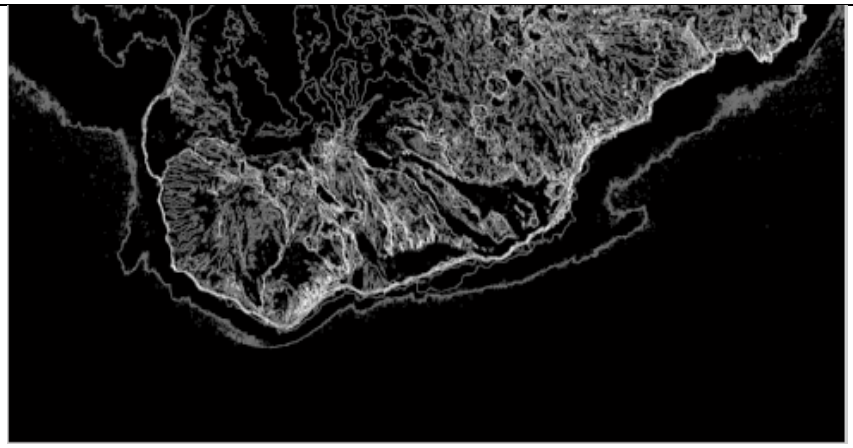
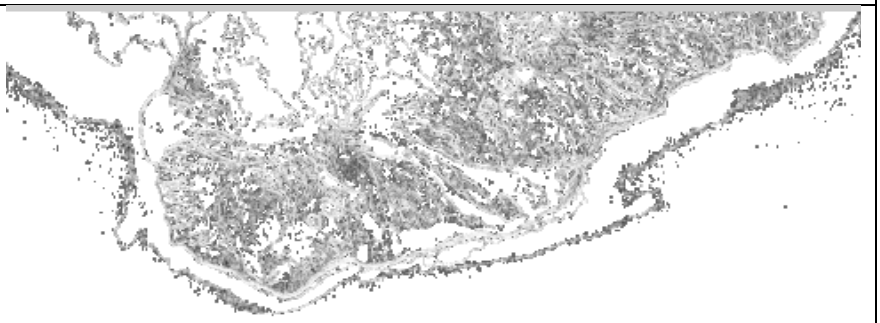
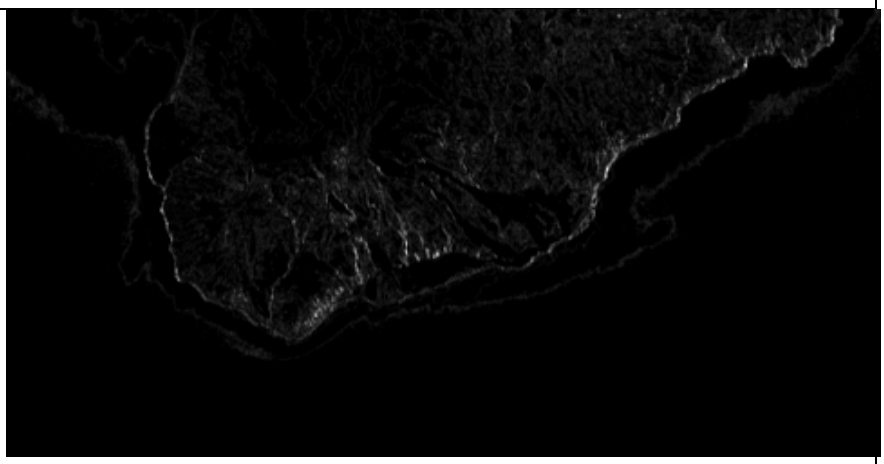
Step testing	Results
10. ROI image	

Table E-4 The result of ROI automatic detection based on texture image compression of image-4.

Input image-4; Name : Soufriere Hills Volcano Resumes Activity.jpg

Step testing	Results
1. Original image	
2. Entropy	
3. Homogeneity	

Step testing	Results
4. Correlation	
5. Contrast	
6. Energy	

Step testing	Results
7. BW	
8. Dilation	
9. ROI mask	

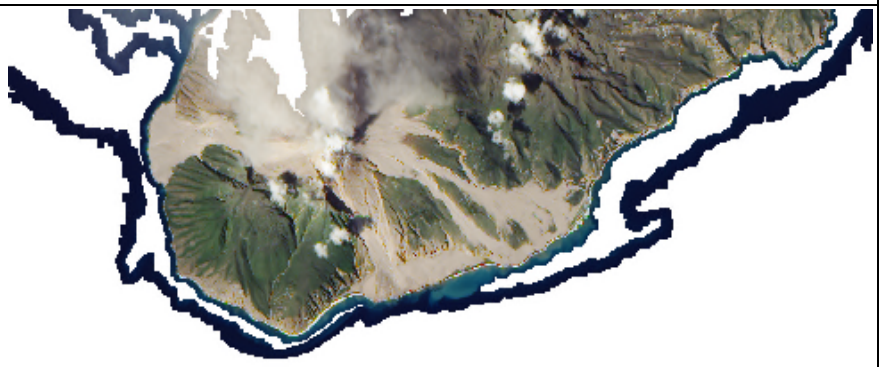
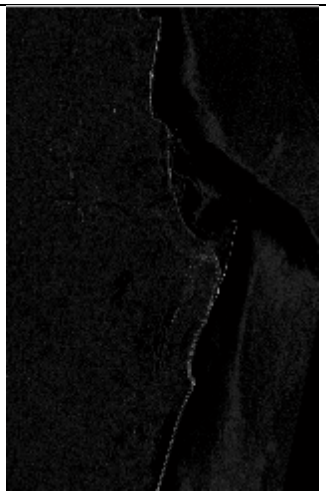
Step testing	Results
10. ROI image	

Table E-5 The result of ROI automatic detection based on texture image compression of image-5.

Input image-5; Name : madagascar_ali_2012050_lrg.jpg

Step testing	Results
1. Original image	
2. Entropy	
3. Homogeneity	

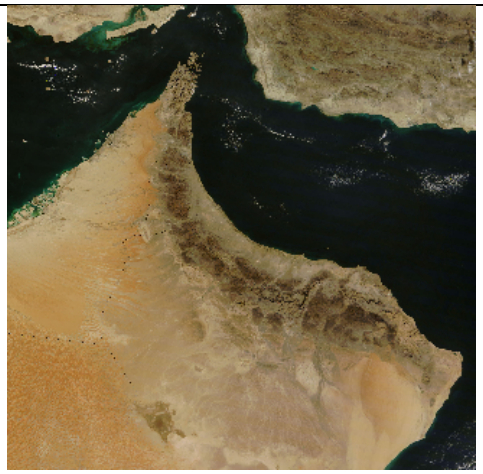
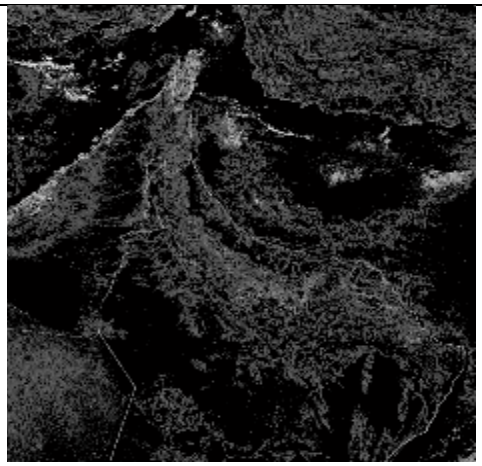
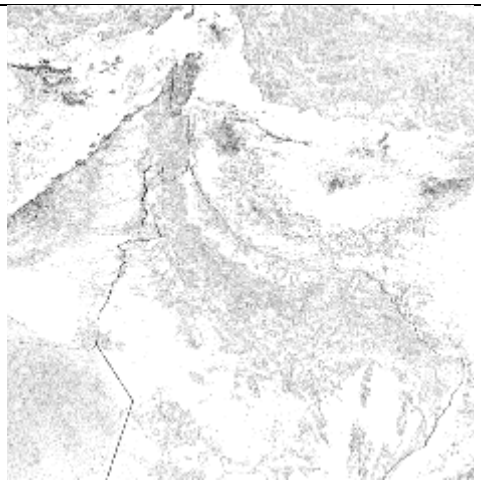
Step testing	Results
4. Correlation	
5. Contrast	
6. Energy	

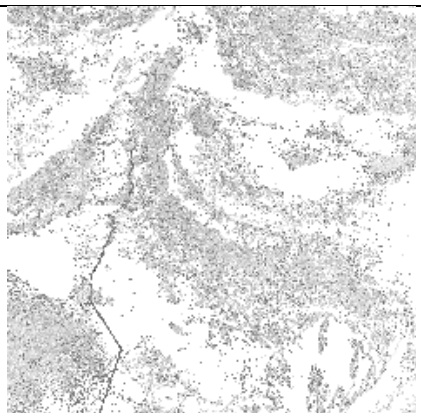
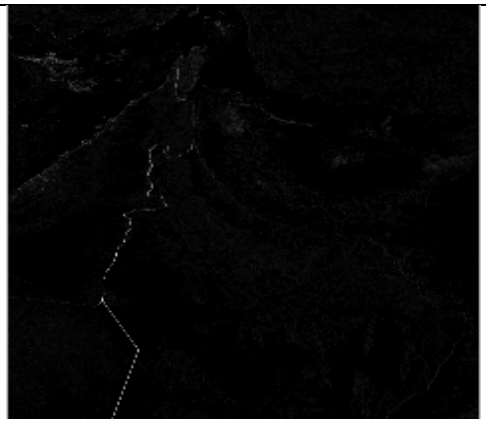
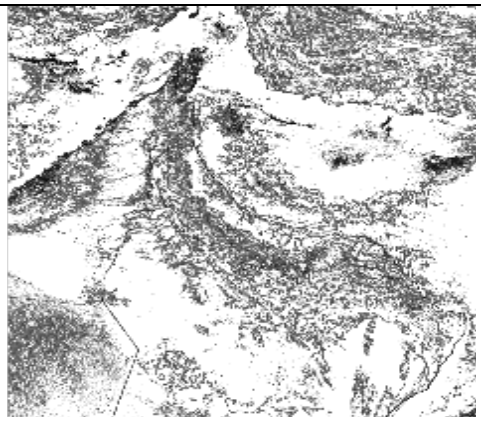
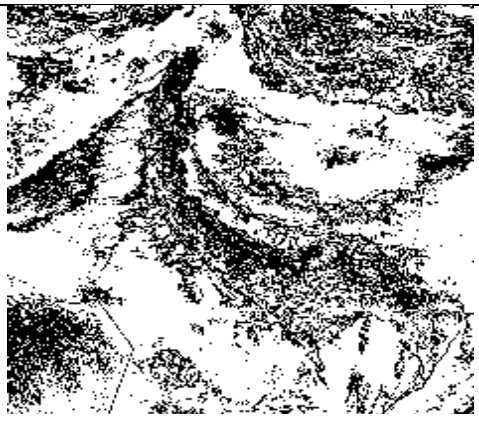
Step testing	Results
7. BW	
8. Dilation	
9. ROI mask	

Step testing	Results
10. ROI image	

Table E-6 The result of ROI automatic detection based on texture image compression of image-6.

Input image-6; Name : oman_tmo_2008329_lrg.jpg

Step testing	Results
1. Original image	
2. Entropy	
3. Homogeneity	

Step testing	Results
4. Correlation	
5. Contrast	
6. Energy	
7. BW	

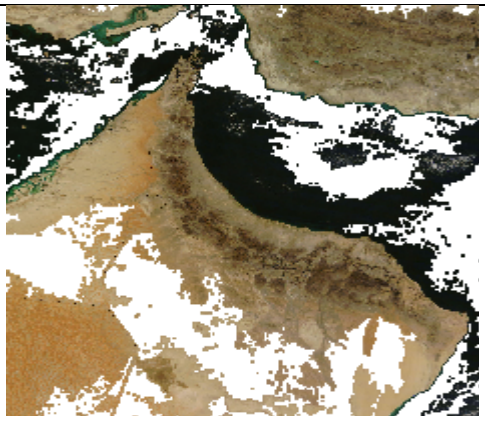
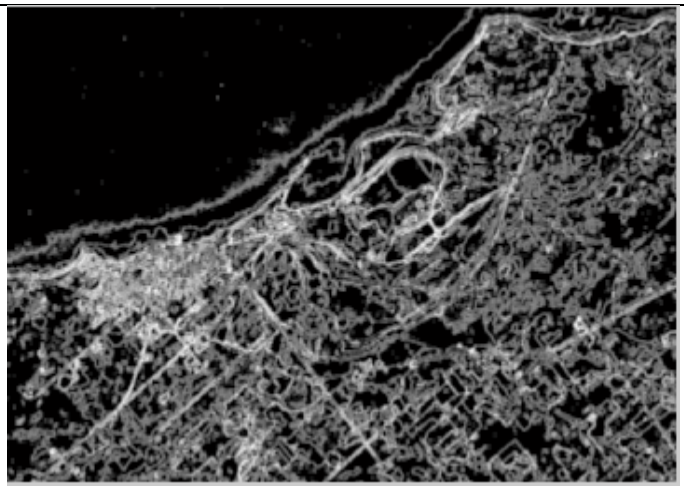
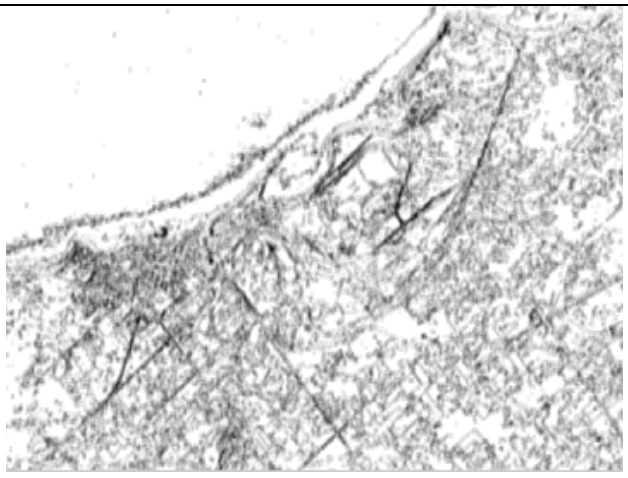
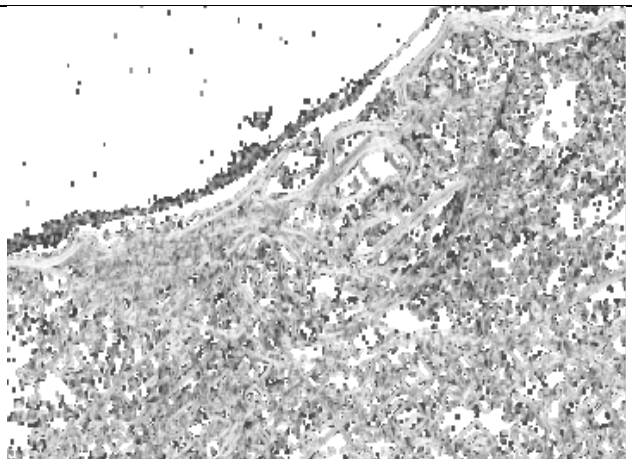
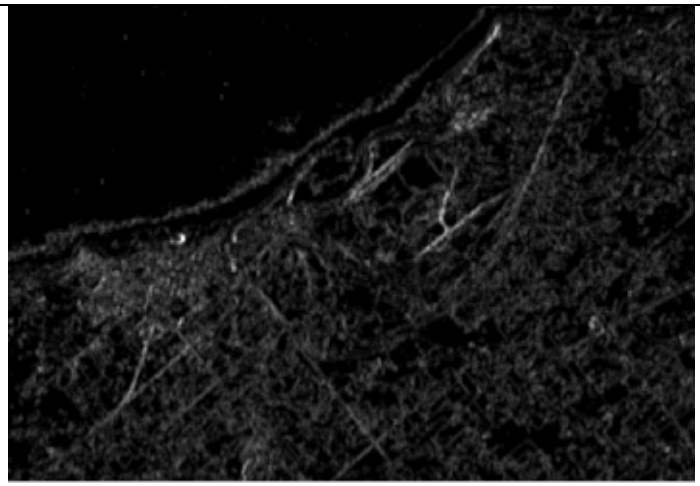
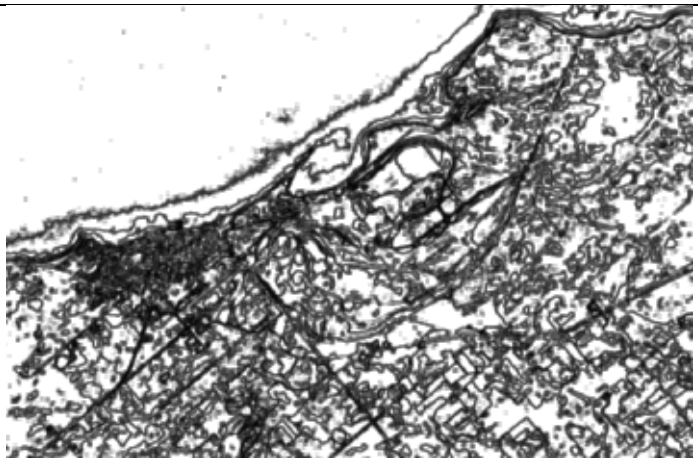
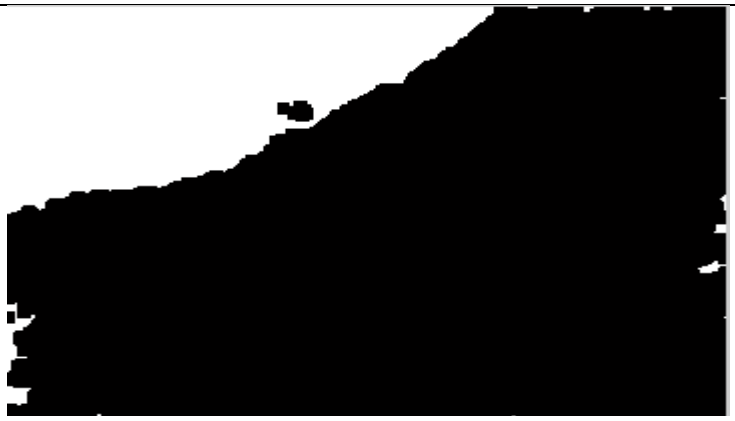
Step testing	Results
8. Dilation	
9. ROI mask	
10. ROI image	

Table E-7 The result of ROI automatic detection based on texture image compression of image-7.

Input image-7; Name : ISS013-E-14843_lrg.jpg

Step testing	Results
1. Original image	
2. Entropy	
3. Homogeneity	

Step testing	Results
4. Correlation	
5. Contrast	
6. Energy	

Step testing	Results
7. BW	
8. Dilation	
9. ROI mask	
10. ROI image	

Appendix F : Comparison of the results of ROI manual and automatic detection

This appendix presents the test results of the comparison of ROI automatic detection and ROI manual detection with different sizes and shapes of structure element. The test image is shown in Figure F-1 and the image has been outlined by the ROI manual detection method. Figure F-2 shows the ROI mask from ROI manual detection. Figure F-3 shows the ROI image from the manual detection.

Figure F-1 : Test image with the outline of ROI manual detection.

Figure F-2 : ROI mask from manual detection.

Figure F-3 : ROI image from manual detection.

Table F-1 : Test results of the comparison between ROI manual and ROI automatic detection using five different sizes of square structure element.

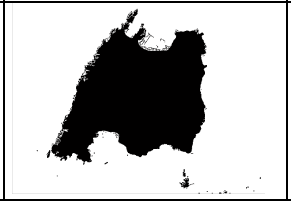
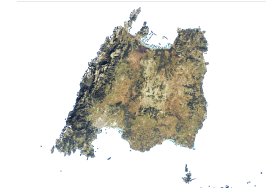
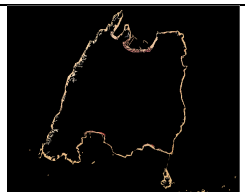
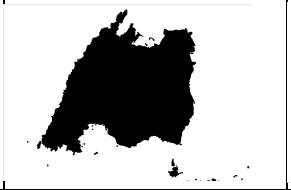
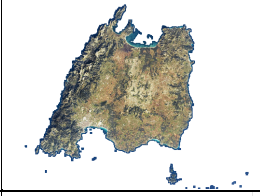
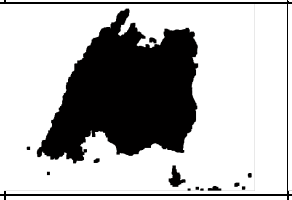
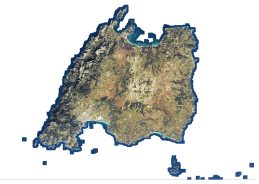
Test image-1 square shape of different size structure element.					
SE Size	ROI mask automatic	ROI image	Image of different pixels	Number of different pixels	Percentage of different
2				39,770	2.56%
5				26,310	1.69%
10				21,297	1.37%
15				27,169	1.75%
20				40,598	2.61%

Table F-2 : Test results of the comparison between ROI manual and ROI automatic detection using five different sizes of diamond structure element.

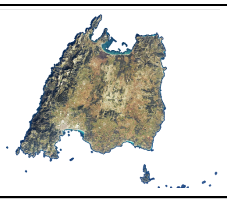
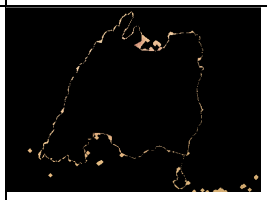
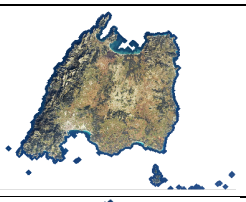
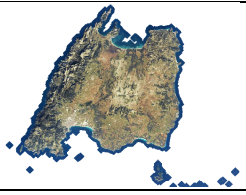
Test image-1 diamond shape of different size structure element.					
SE Size	ROI mask automatic	ROI image	Image of different pixels	Number of different pixels	Percentage of different
2				28,774	1.85%
5				22,164	1.43%
10				27,725	1.78%
15				48,154	3.10%
20				72,192	4.64%

Table F-3 : Test results of the comparison between ROI manual and ROI automatic detection using five different sizes of disk structure element.

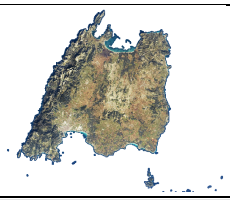
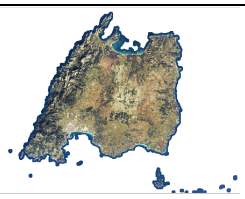
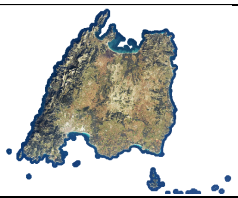
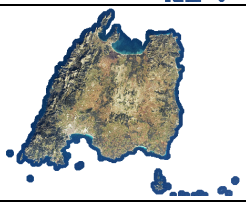
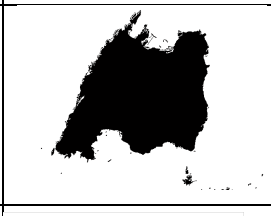
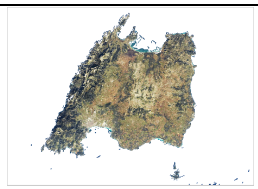
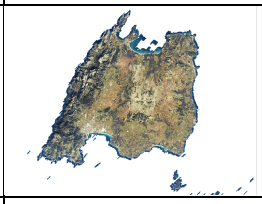
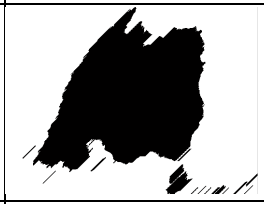
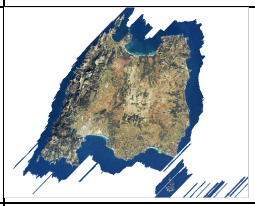
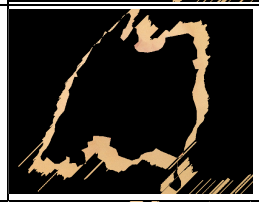
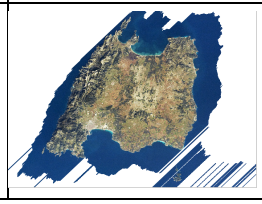
Test image-1 disk shape of different size structure element.					
SE Size	ROI mask automatic	ROI image	Image of different pixels	Number of different pixels	Percentage of different
2				28,774	1.85%
5				22,503	1.45%
10				32,848	2.11%
15				58,766	3.78%
20				91,303	5.87%

Table F-4 : Test results of the comparison between ROI manual and ROI automatic detection using five different sizes of line structure element.

Test image-1 line shape of different size structure element.					
SE Size	ROI mask automatic	ROI image	Image of different pixels	Number of different pixels	Percentage of different
2				42,779	2.75%
5				25,042	1.61%
10				87,961	5.66%
15				206,501	13.27%
20				359,097	23.09%

Bibliography

Aalbers, G., G. N. Gaydadjiev, et al. (2005). CDHS Design for a University Nanosatellite. Proceedings of the 56th International Astronautical Congress, Fukuoka, Japan.

Addaim, A., A. Kherras, et al. (2007). Design of Store and Forward Data Collection Low-cost Nanosatellite. Aerospace Conference, 2007 IEEE.

Addaim, A., A. Kherras, et al. (2007). Design of a Telecommand and Telemetry System for use on Board a Nanosatellite. Electronics, Circuits and Systems, 2007. ICECS 2007. 14th IEEE International Conference on.

Agus, Z. A. and A. Asano (2004). Image Thresholding by Histogram Segmentation Using Discriminant Analysis. In: Proceedings of Indonesia–Japan Joint Scientific Symposium 2004: IJJSS'04: 169-174.

Akers, E. L. (2007). Computer Architecture : Chapter 2, Center of Excellence in Remote Sensing Education and Research, North Carolina.

Al-amri, S. S., N. V. Kalyankar, and Khamitkar S.D. . (2010). "Image Segmentation by Using Threshold Techniques." Journal of Computing. vol2 Issue 5

Al-Kubati, A. A. M., J. A. M. Saif, et al. (2012). Evaluation of Canny and Otsu Image Segmentation. International Conference on Emerging Trends in Computer and Electronics Engineering (ICETCEE'2012), Dubai, India.

Alaitz, Z., V. Raffaele, et al. (2012). "Impact of CCSDS-IDC and JPEG 2000 Compression on Image Quality and Classification." Journal of Electrical and Computer Engineering2012, Article ID 761067.

Alex (2013). "SATELLITE IMAGES OF TYPHOON HAIYAN PATH OF DESTRUCTION." from http://news.satimagingcorp.com/.

Ali, M. and D. Clausi (2001). Using the Canny edge detector for feature extraction and enhancement of remote sensing images. Geoscience and Remote Sensing Symposium, 2001. IGARSS '01. IEEE 2001 International.

Alminde, L. and K. K. Laursen (2009). A strategic approach to developing space capabilities using Cubesat technology. Recent Advances in Space Technologies, 2009. RAST '09. 4th International Conference on.

Altan, O. (2011). Key scientific issues of present day disaster management scenario. General Assembly and Scientific Symposium, 2011 XXXth URSI.

Altera, C. (2013). "Altera measurable advantage : Support Solution." Retrieved 24 July 2013, from http://altera.us/support/kdb/solutions/rd03102006_962.html.

Andrej, T. and Z. Baldomir (1999). Design of Real-Time Edge Detection Circuits on Multi-FPGA Prototyping System. ELECO'99INTERNATIONAL CONFERENCE ON ELECTRICAL AND ELECTRONICS ENGINEERING.

Andrew Yeung and S. Chen (1998). "Energy Compaction by Orthonormal Transforms." Retrieved 10 October 2013, 2013, from http://scien.stanford.edu/pages/labsite/1997/ee392c/demos/yeung_chen/.

Antonini, M., M. Barlaud, et al. (1992). "Image coding using wavelet transform." Image Processing, IEEE Transactions on1(2): 205-220.

Asma, C. (2013). "QB50." Retrieved 4 July 2013, 2013, from https://www.qb50.eu/.

Atmel (2006). "Datasheet reference." Retrieved 24 July 2013, from http://datasheetreference.com/avr-atmega128-datasheet.html.

Atmel (2013). "DataSheet ATmega128, ATmega128L." Retrieved 25 July 2013, from www.atmel.com/Images/doc2467.pdf \.

Bakos, J. (2001). Chapter2 Performance: MIPS Instruction Set Architecrure, College of Engineering and Computing, University of South carolina.

Baraldi, A. and F. Parmiggiani (1995). "An investigation of the textural characteristics associated with gray level cooccurrence matrix statistical parameters." Geoscience and Remote Sensing, IEEE Transactions on33(2): 293-304.

Barnhart, D. J., T. Vladimirova, et al. (2007). "Very-Small-Satellite Design for Distributed Space Missions." Journal of Spacecraft and Rockets - J SPACECRAFT ROCKET44 (6): 1294-1306.

Bartels, M., W. Hong, et al. (2005). Wavelet packets and co-occurrence matrices for texture-based image segmentation. Advanced Video and Signal Based Surveillance, 2005. AVSS 2005. IEEE Conference on.

Bilhaut, L. and L. Duraffourg (2009). "Assessment of nanosystems for space applications." Acta Astronautica65(9–10): 1272-1283.

Blau, P. (2014). "Flock-1 CubeSat Constellation begins Deployment from Space Station." Retrieved 23 April, 2014, from http://www.spaceflight101.com/flock-1-cubesat-constellation-begins-deployment-from-iss.html.

Borgeaud, M., N. Scheidegger, et al. (2010). SwissCube: The First Entirely-Built Swiss Student Satellite with an Earth Observation Payload Small Satellite Missions for Earth Observation. R. Sandau, H.-P. Roeser and A. Valenzuela, Springer Berlin Heidelberg: 207-213.

Bouwmeester, J. and J. Guo (2010). "Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology." Acta Astronautica67(7–8): 854-862.

Braegen, E., D. Hayward, et al. (2007). AdeSat Final Report Level IV Honours., University of Adelaide School of Mechanical Engineering. Master.

Brancati, N., M. Frucci, et al. (2008). Image Segmentation Via Iterative Histogram Thresholding and Morphological Features Analysis. Image Analysis and Recognition. A. Campilho and M. Kamel, Springer Berlin Heidelberg. 5112: 132-141.

Brieß, K. (2009). Prospects for future CubeSat technology developments. The QB50 Workshop. Brussels, ESA.

Brieß, K., Baumann, F., Herfort, M.,Trowitzsch, S (2011). Present and Future Picosatellite Missions at TU Berlin. 8th IAA Symposium on Small Satellites for Earth Observation,. IAA. Berlin, IAA.

Briess, R. S. a. K. (2010). "The Role of Small Satellite Missions in Global Change Studies." Advances in Earth Observation of Global Change.

Bruzzone, L. and C. Persello (2009). Active learning for classification of remote sensing images. Geoscience and Remote Sensing Symposium,2009 IEEE International,IGARSS 2009.

CCSDS (2007) Report Concerning Space Data System Standards : Image data compression.

Chao, C. and K. K. Parhi (2004). "Hardware efficient fast parallel FIR filter structures based on iterated short convolution." Circuits and Systems I: Regular Papers, IEEE Transactions on51(8): 1492-1500.

Chen, C.-W., T.-C. Lin, et al. (2009). A Near Lossless Wavelet-Based Compression Scheme for Satellite Images. Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering - Volume 06, IEEE Computer Society: 528-532.

Chen, O. T. C. and C. Chih-Chang (2007). "Automatically-Determined Region of Interest in JPEG 2000." Multimedia, IEEE Transactions on9(7): 1333-1345.

Chien-Wen, C., L. Tsung-Ching, et al. (2009). A Near Lossless Wavelet-Based Compression Scheme for Satellite Images. Computer Science and Information Engineering, 2009 WRI World Congress on.

Christophe, E., Leger, D., Mailhes, C., "Quality Criteria Benchmark for Hyperspectral Imagery," Geoscience and Remote Sensing, IEEE Transactions on , vol.43, no.9, pp.2103,2114, Sept. 2005, doi: 10.1109/TGRS.2005.853931

Chuvieco, E. (2008). Earth Observation of Global Change: The Role of Satellite Remote Sensing in Monitoring the Global Environment, Springer.

Clausen, T.B,; Hedegaard A.;Rausmussen, K. B.; Olsen, R. L.;.Lundkvist, J. and P. E. Nielsen. (2001) Designing On Board Computer and Payload for the AAU Cubesat
. Aalborg University, Fredrik Bajers,Denmark.
ClydeSpace (2012). "Cubesat System : Some useful information about Cubesat." from http://www.clyde-space.com/cubesat/som_useful_info_about_cubesats

Cowing, K. (2013). "Planet Labs Reveals First Images from Space." Retrieved 23 April 2014, from http://spaceref.biz/2013/06/planet-labs-reveals-first-images-from-space.html.

Crawford, M. M., D. Tuia, et al. (2013). "Active Learning: Any Value for Classification of Remotely Sensed Data?" Proceedings of the IEEE101(3): 593-608.

Daggu Venkateshwar Rao, S. P., Naveen Anne Babu,V Muthukumar (2006). "Implementation and Evaluation of Image Processing Algorithms on Reconfigurable Architecture using C-based Hardware Descriptive Languages " International Journal of Theoretical and Applied Computer SciencesVolume 1 Number 1 (2006): 9–34.

Dai, F. and M. S. B. Zhang (2011). A Survey of Image Segmentation by the Classical Method and Resonance Algorithm. P.-G. Ho(Ed.).

Daniel Selva a,n , David Krejci b, 2012, A survey and assessment of the capabilities of Cubesats for Earth observation, Acta Astronautica, Volume 74, May–June 2012, Pages 50–68, doi:10.1016/j.actaastro.2011.12.014

David Calcutt, L. T. (1994). Satellite Communications: Principles and Applications, Butterworth-Heinemann.

Davidsen, P. (2001) Satellite Systems and Design "Architecture of On-Board System".

de O. Bastos, L., P. Liatsis, et al. (2008). Automatic texture segmentation based on k-means clustering and efficient calculation of co-occurrence features. Systems, Signals and Image Processing, 2008. IWSSIP 2008. 15th International Conference on.

de Rooij, N.F, Gautsch, S., Briand, D, Marxer, C & Staufer, U (2009). MEMS for space. In s.n. (Ed.), *Proc.of Solid-State Sensors, Actuators and Microsystems Conference, 2009. TRANSDUCERS 2009* (pp. 17-24). Denver, US: IEEE.

Delfi-C3-team (2005). The Delfi-C3 Student Nanosatellite. Proceedings of the AMSAT-UK Colloquium, Surrey, United Kingdom.

Di Liping , Kobler Ben , 2000, " International Archives of Photogrammetry and Remote Sensing". Vol. XXXIII, Part B2. Amsterdam

Dipti, P. and J. Mridula (2011). Featured based Segmentation of Color Textured

Images using GLCM and Markov Random Field

Model. World Academy of Science, Engineering and Technology, World Academy of Science.

Dudás, L. V., Lajos; Seller, Rudolf (2009). The communication subsystem of Masat-1, the first Hungarian satellite. Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2009, roceedings of the SPIE. 7502 (2009).

Elbert, B. R. (2008). Introduction to Satellite Communication, Artech House.

Ershad, S. F. (2011). "Texture Classification Approach Based on Combination of Edge & Co-occurrence and Local Binary Pattern." Int'l Conf. IP, Computer Vision and Pattern Recognition (cs.CV); Artificial Intelligence (cs.AI): 626-629.

ESA (2012). "Space in image." Retrieved 09 August, 2013, from http://spaceinimages.esa.int/Images/2012/03/Western_Australia_photographed_by_Masat-1.

ESA (2013). "Minigallery : Cubesats on the VEGA maiden flight." Retrieved 9 August 2013, 2013, from http://www.esa.int/Education/CubeSats_on_the_Vega_maiden_flight/(ipp)/9/(start)/9.

ESA (2013). "Satellite Mission Database : Directory Home satellite mission : https://directory.eoportal.org/web/eoportal/satellite-missions." Retrieved 10 August 2013.

ESA (2014). "Technology : Hyperspectral imaging by cubesat on the way." Retrieved 23 April, 2014, from http://www.esa.int/Our_Activities/Technology/Hyperspectral_imaging_by_CubeSat_on_the_way.

ESA, (2014), Hyperspectral imaging by cubesat on the way, Space engineering and technology, Retrieved 22 May 2015 from http://www.esa.int/Our_Activities/Space_Engineering_Technology/Hyperspectral_imaging_by_CubeSat_on_the_way.

ESA, (2015), 'Data types, level, formats' Retrieved 22 May 2015 from https://earth.esa.int/web/guest/-/data-types-levels-formats-7631,

Fancey, N.E, . I.D. Gardiner, R.A. Vaughan, (1996), "The determination of Geophysical parameters from Space" , Institute of Physics, London, Printed in Great Britain by J W Arrowsmith Ltd. Bristol

Fang, M., G. Yue, et al. (2009). The study on an application of Otsu method in Canny operation. Proceedings of the 2009 International Symposium on Information Processing (ISIP'09), Huangshan, P. R. China, ACADEMY PUBLISHER.

Faria, L. N., L. M. G. Fonseca, et al. (2012). "Performance Evaluation of Data Compression Systems Applied to Satellite Imagery." Journal of Electrical and Computer Engineering2012 (2012), Article ID 471857.

Fei, W., W. Jishang, et al. (2007). Automatic ROI Selection for JPEG2000 Compression of Remote Sensing Images. Semantic Computing, 2007. ICSC 2007. International Conference on.

Floyd F. Sabins, J. (1997). Remote sensing principles and interpretion : Chapter1

W.H.Freeman and company New York.

Floyd F. Sabins, J. (1997). Remote sensing principles and interpretion : Chapter3, W.H.Freeman and company New York.

Fortescue, P., J. Stark, and Swinerd. (2003). Spacecraft Systems Engineering : Chapter 12 Tele-communications. West Sussex: England, John Wiley and Sons Ltd.

Fukuma, S., Tanaka, T. and M. Nawate. (2005). "Switching Wavelet Transform for ROI Image Coding." IEICE Trans. Fundam. Electron. Commun. Comput. Sci.E88-A(7): 1995-2006.

Garcia-Vilchez, F. and J. Serra-Sagrista (2009). "Extending the CCSDS Recommendation for Image Data Compression for Remote Sensing Scenarios." Geoscience and Remote Sensing, IEEE Transactions on47(10): 3431-3445.

Gonzalez, R. and R. Woods (2007). Digital Image Processing Prentice Hall.

Greenland, S., Clark C. (2010). Cubesat platforms as an on-orbit technology validation and verification vehicle. European Small Satellite Service Symposium, Pestana conference center, Funchal, Madeira, Portugal.

Grosse, J. (2011). "Swisscube." from http://swisscube.epfl.ch/.

Grundland, M. and N. A. Dodgson (2005). The decolorize algorithm for contrast enhancing,color to grayscale conversion. Cambridge,UK, Computer Laboratory, University of Cambridge.

Guang, H., X. Ming, et al. (2009). Detecting Automatically and Compression Algorithm for Infrared Image Based on Region of Interest. Computer Science-Technology and Applications, 2009. IFCSTA '09. International Forum on.

Guarnieri, A. and A. Vettore (2002). Automated Techniques for satellite image segmentation. ISPRS Commission IV, Symposium 2002 on Geospatial Theory, Processing and Applications Ottawa, Canada, ISPRS Archives.

Gulzar, K. (2009). Camera design for Pico and Nano Satellite Application. Continuation courses Space Science and Technology,Department of Space Science. Kiruna, Sweden, Lulea University of Technology. Master in Space Technology.

Haifang, L., M. Lihuan, et al. (2008). A Method of the Extraction of Texture Feature. Proceedings of the 3rd International Symposium on Advances in Computation and Intelligence. Wuhan, China, Springer-Verlag: 368-377.

Han, J.-w. and J.-d. Fang (2008). Research on Remote Sensing Image Compression Based on Lifting Wavelet Transform. MultiMedia and Information Technology, 2008. MMIT '08. International Conference

Hanninen, J. (2012). "alto-1 – Student Satellite." from http://radio.aalto.fi/en/research/space_technology/aalto-1-student_satellite/.

Haralick, R. M., K. Shanmugam, et al. (1973). "Textural Features for Image Classification." Systems, Man and Cybernetics, IEEE Transactions onSMC-3(6): 610-621.

Harrington, P., Wai N. (2012). Investigation of the speed-up of a dual microcontroller parallel processing system in the execution of a mathematical operation, Northumbria Research Link.

Hedberg, H., P. Dokladal, et al. (2009). "Binary Morphology With Spatially Variant Structuring Elements: Algorithm and Architecture." Image Processing, IEEE Transactions on18(3): 562-572.

Hinkley, D. A. (2010) AeroCube 3 and AeroCube 4.

Holmlund, C., Heikki Saari, et al. (2012) The Aalto-1 Spectral Imager (AaSI) mission.

Huerta, D. C. (2006). Development of a Highly Integrated Communication System for use in Low Power Space Applications. The Faculty of California Polytechnic State University. San Luis Obispo, California Polytechnic State University. Master of Science in Aerospace Engineering.

Hunyadi, G., D. M. Klumpar, et al. (2002). A commercial off the shelf (COTS) packet communications subsystem for the Montana EaRth-Orbiting Pico-Explorer (MEROPE) CubeSat. Aerospace Conference Proceedings, 2002. IEEE, IEEE.

Ian S. Robinson, 2011 Raytheon Company , HYPERCUBE: Hyperspectral Imaging Using a CUBESAT, presentation, 8th Annual CubeSat Developers' Workshop, April 20 - 22, 2011, Cal Poly Campus - San Luis Obispo, CA, USA

IDL-online (2005). "Determining Structuring Element Shapes and Sizes." from http://star.pst.qub.ac.uk/idl/Determining_Structuring_Element_Shapes_and_Sizes.html.

Jaan Praks, Antti Kestila¨ , Martti Hallikainen, Heikki Saari, Jarkko Antila, Pekka Janhunen, Rami Vainio, AALTO-1 – AN EXPERIMENTAL NANOSATELLITE FOR HYPERSPECTRAL REMOTE SENSING, Geoscientific Instrumentation, Methods and Data Systems 02/2013; 2:121-130. DOI: 10.5194/gi-2-121-2013

Jakhar, D. K. (2012). "Team STUDSAT." from http://www.teamstudsat.com/about1.html.

James Armstrong, Craig Casey, Glenn Creamer and Gilbert Dutchover , 2009,"Pointing Control for Low Altitude Triple Cubesat Space Darts" , Small satellite conference Author(s)." *Proceedings of the AIAA/USU Conference on Small Satellites*

Jansen, T., A. Reinders, et al. (2010). Performance of the first flight experiment with dedicated space CIGS cells onboard the Delfi-C3 nanosatellite. Photovoltaic Specialists Conference (PVSC), 2010 35th IEEE.

Jian-wei, H. and F. Jian-dong (2008). Research on Remote Sensing Image Compression Based on Lifting Wavelet Transform. MultiMedia and Information Technology, 2008. MMIT '08. International Conference on.

K. Viherkanto, H. S., A. Näsilä (2013). AaSI – Aalto-1 Spectral Imager development status. 5th European CubeSat Symposium, Royal Military Academy, VKI (Von Karman Institute), Brussels, Belgium, ESA.

Kahr, Montenbruck, et al. (2011). GPS tracking on a nanosatellte - The Canx-2 flight experience. 8th International ESA conference on Guidance, Navigation and Control system. Czech Republic, EESA: European Space Agency.

Kalman, A. and A. Reif (2007). MISC A Novel Approach to Low-Cost Imaging Satellite. 22nd Annual AIAA/USU Conference on small satellite, Logan, Utah, USA, Kalman.

Kalman, A. E. (2011) Recent developments in Pumpkin's Cubesat Kit architecture.

Kanan, C. and G. W. Cottrell (2012). "Color-to-Grayscale: Does the Method Matter in Image Recognition?" PLoS ONE7(1): e29740.

Karatas, Y. and F. Ince (2009). The place of small satellites in fulfilling the Earth observation requirements of a developing country. Recent Advances in Space Technologies, 2009. RAST '09. 4th International Conference on.

Kestila A, T. Tikka , P. Peitso1 , J. Rantanen1 , A. Nasil a , K. Nordling , H. Saari , R. Vainio , P. Janhunen , J. Praks , and M. Hallikainen, 2013, "Aalto-1 nanosatellite – technical description and mission objectives", Geoscientific Instrumention methods and data systems, 2, 121–130, 2013.

Kiadtikornthaweeyot, W., Tatnall, A.R.L. (2010). Cubesat Satellite Telemetry, Telecommand, Data Handling and Processing. 5th ESA International Workshop on Tracking, Telemetry and Command Systems for Space Applications. ESTEC, Noordwijk, The Netherlands.

Kiadtikornthaweeyot, W., Tatnall, A.R.L. (2011). A highly efficient image compression system for a picosatellite. 28th ISTS (The 28th International Symposium on Space Technology and Science). Okinawa, Japan.

Kiadtikornthaweeyot, W., Tatnall, A.R.L. (2012). Adaptive Image Compression For Cubesats. In 4s Symposium (Small Satellite Systems & Services Symposium). Portoroz, Slovenia, ESA.

Klofas, B. (2013) CubeSat Communication Systems 2003-2013: A Historical Look : The Nanosatellite Ground Station Workshop.

Klofas, B., J. Anderson, et al. (2009). "A Survey of CubeSat Communication Systems." The AMSAT JournalNovember/December 2009.

Klofas, B. and K. Leveque (2013) A Survey of CubeSat Communication Systems: 2009-2012. 10th Annual Cubesat Developers Workshop. April 24-26,2013 Cal Poly.

Kochhar A.K. , B. N. D. (1983). Microprocessors and their Manufacturing Application. Bath, Great Britain, Edward Arnold

Kramer, H. J. and A. P. Cracknell (2008). "An overview of small satellites in remote sensing*." International Journal of Remote Sensing29(15): 4285-4337.

Kurugollu, F., B. Sankur, et al. (2001). "Color image segmentation using histogram multithresholding and fusion." Image and Vision Computing19(13): 915-928.

Lambert-Nebout, C. and G. Moury (1999). A survey of on-board image compression for CNES space missions. Geoscience and Remote Sensing Symposium, 1999. IGARSS '99 Proceedings. IEEE 1999 International.

Langton, C. (2002) Intuitive Guide to Principles of Communication; Link Budgets.

Lara, R. A. and W. Yunhong (2011). Lossless Compression On-Board Remote Sensing Satellites. Future Computer Science and Education (ICFCSE), 2011 International Conference on.

Liu, C., T. Xia, et al. (2004). ROI and FOI algorithms for wavelet-based video compression. Proceedings of the 5th Pacific Rim conference on Advances in Multimedia Information Processing - Volume Part III. Tokyo, Japan, Springer-Verlag: 241-248.

Loukas (2004) Beginning microcontrollers programming with PIC.

Lucking, C., C. Colombo, et al. (2012). Mission and system design of a 3U CubeSat for passive GTO to LEO transfer. 63rd International Astronautical Congress, Naples.

Masataka Naitoh, Haruyoshi Katayama, Masatomo Harada, Ryoko Nakamura, Eri Kato, Yoshio Tange, Ryota Sato, Koji Nakau, "Compact Infrared Camera (CIRC) for Earth Observation," Proceedings of the 29thISTS (International Symposium on Space Technology and Science), Nagoya-Aichi, Japan, June 2-8, 2013, paper: 2013-n-29

Mahmood, R., K. Khurshid, et al. (2011). Institute of Space Technology CubeSat: ICUBE-1 subsystem analysis and design. Aerospace Conference, 2011 IEEE.

Marshall, W. (2014). "Planet labs : Remote sensing satellite system : Flock-1." Retrieved 23 April, 2014, from http://www.planet.com/flock1/.

Mather, P. M. (2004). Computer processing of Remotely-Sensed Images, Wiley.

MathWork, I. (2012). Image Processing Toolbox™ User's Guide R2012b Matlab.

MikroElektronika (2013). "PIC Microcontrollers." from http://www.mikroe.com/chapters/view/2/.

Mirzapour, F.; Ghassemian, H., "Using GLCM and Gabor filters for classification of PAN images," *Electrical Engineering (ICEE), 2013 21st Iranian Conference on* , vol., no., pp.1,6, 14-16 May 2013doi: 10.1109/IranianCEE.2013.6599565

Muri, P. and J. McNair (2012). "A Survey of Communication Sub-systems for Intersatellite Linked Systems and CubeSat." JOURNAL OF COMMUNICATIONS7(4).

Muriel, N., J. Fabien, et al. (2009). Lessons Learned from the First Swiss Pico-Satellite: SwissCube. 23rd Annual AIAA/USU Conference on Small Satellites. Logan, USA.

Muthukrishnan.R and R. Myilsamy (2011). "Edge detection techniques for image segmentation." International Journal of Computer Science & Information Technology (IJCSIT)3.

NASA, (2014), "Small Spacecraft Technology state of the Art", Mission Design Division Staff Ames Research Center, Moffett Field, California, Retrieved 21 May 2015 from http://www.nasa.gov/sites/default/files/files/Small_Spacecraft_Technology_State_of_the_Art_2014.pdf

Noe, C. (2004). Design and Implementation of the Communications Subsystem for the Cal Poly CP2 Cubesat Project. Computer Engineering Department. San Luis Obispo, California Polytechnic State University. Master.

O'Brien, D. (2013). "Image Entropy." Retrieved 05 April 2013, 2013, from http://www.astro.cornell.edu/research/projects/compression/compression.html.

Patterson, D. and J. L. Hennessy (2009). Computer Organization and Design, Fourth Edition: The Hardware/Software Interface, Morgan Kaufmann.

Pearlman, J., S. Holt, et al. (2008). Creation and operation of a global earth observation system of systems. OCEANS 2008.

Pedersen, S. (2007-2012). NanoCam Payloads. Innovation Center Denmark.

Pen-Shu, Y., P. Armbruster, et al. (2005). The new CCSDS image compression recommendation. Aerospace Conference, 2005 IEEE.

Pisane, J. (2008). Design and implementation of the terrestrial and space telecommunication elements of the student Nanosatellite of the University of Liege. the Applied Science Faculty of the University of Liège. 2008, University of Liège. Master in Electrical Engineering.

Polaschegg, M. (2005). Study of a Cube-Sat Mission. Master of Sciences of the Post-graduate University Course Space Sciences. Graz, Austria, Karl Franzens University of Graz. Master: 111.

Ponneela, V. and R. Rajendran (2012). "Performance and Analysis of Edge detection using FPGA Implementation." International Journal of Modern Engineering Research (IJMER)2(2): 552-554.

Pratt, W. K. (2007). Digital image processing, New York, NY : Wiley-Interscience.

Przyborski, P. (2010). "The earth observatory." Retrieved 05 May http://earthobservatory.nasa.gov/Images/, 2012.

Puig-suari, J., Turner C. and R.J. Twiggs, (2001). "Cubesat: The Development and Launch Support Infrastructure for Eighteen Different Satellite Customers on One Launch." 15TH Annual/USU Conference 1 on Small Satellites1SSC01-VIIIb-5.

Pumpkin, Inc. (2011). "History & Performance of Pumpkin's Products in Space." Retrieved 22 October 2012, from http://www.cubesatkit.com/content/space.html.

Pumpkin Inc., (2011) "Miniature 3-Axis Reaction Wheel & Attitude Determination and Control System for CubeSat Kit". [Online]. Available: /http://www.cubesatkit.com/docs/datasheet/DS_CSK_A DACS_634-00412-A.pdfS. [Accessed: 25-Aug-2011].

Puscell J.J. and E. Stanton. Remote Sensing System Engineering IV, edited by Philip E. Ardanuy, Jeffery J. Puschell, Hal J. Bloom, Proc. of SPIE Vol. 8516, 85160B · © 2012 SPIE · CCC code: 0277-786X/12/$18 · doi: 10.1117/12.942304

Rajkumar, T. M. P. and M. V. Latte (2011). "ROI Based Encoding of Medical Images: An Effective Scheme Using Lifting Wavelets and SPIHT for Telemedicine." International Journal of Computer Theory and Engineering3(3): 9.

Rani, B. S., R. R. Santhosh, et al. (2010). A survey to select microcontroller for Sathyabama satellite's On Board Computer subsystem. Recent Advances in Space Technology Services and Climate Change (RSTSCC), 2010.

Reinhard, R. (2013). 5th European Cubesat Symposium Book of Abstracts. Brussels, Belgium.

Richard J. Peralta, Carl Nardell, et al. (2007). Aerosol Polarimetry Sensor for the Glory Mission. SPIE.

Robinson, I. S. (2011) Hypercube : Hyperspectral Imaging Using a Cubesat.

Roger Walker, 28 January 2015, space engineering and technology, cubesat http://www.esa.int/Our_Activities/Space_Engineering_Technology/CubeSats. Access 22 May 2015

Romeijn Leopold (2011) satellite image coporation Retrieved 9 July 2013, from http://news.J.satimagingcorp.com/2011/06/satellite_imaging_corporation_sic_signs_agreement_with_rapideye_ag_for_distribution_of_rapideye_satellite_image_data.html

Rupprecht, M. (2013). "BeeSat-1." Retrieved 8 August 2013, from http://www.dk3wn.info/sat/afu/sat_beesat.shtml.

Samson, J. R., Jr. (2012). Update on Dependable Multiprocessor CubeSat technology development. Aerospace Conference, 2012 IEEE.

Sandau, R. (2010). "Status and trends of small satellite missions for Earth observation." Acta Astronautica66(1–2): 1-12.

Saravanan, C. (2010). Color Image to Grayscale Image Conversion. Computer Engineering and Applications (ICCEA), 2010 Second International Conference on.

Scholz, A. (2005). Command and Data Handling System for the COMPASS-1 Picosatellite, 5th Symposium on Small Satellite for Earth Observation. 5th IAA Symposium on Small Satellites for Earth Observation, Berlin, Germany.

Segert, T., S. Engelen, et al. (2011). ThePicoStarTracker ST-200: a new approach in miniaturized star Sensor systems. The1st IAA Conference on University Satellite Missions.

Selva, D. and D. Krejci (2012). "A survey and assessment of the capabilities of Cubesats for Earth observation." Acta Astronautica74(0): 50-68.

Shanmugan, K. S., V. Narayanan, et al. (1981). "Textural Features for Radar Image Analysis." Geoscience and Remote Sensing, IEEE Transactions onGE-19(3): 153-156.

Shea, H. R. (2009). MEMS for pico- to micro-satellites. Moems and Miniaturized Systems Viii. D. L. Dickensheets, H. Schenk and W. Piyawattanametha. Bellingham, Spie-Int Soc Optical Engineering. 7208.

Shimizu, K. and Y. Kusakawa (2010). "Prism project." Retrieved 22 October 2012, 2012, from http://www.space.t.u-tokyo.ac.jp/prism_old/about-e.html.

Simmons, M. (2008) Remote Sensing with CubeSats.

Singh, J. (2009) Canny algorithm for edge detection in digital images.

Smith, A. (2012). QB50 Sensor Selection Working Group (SSWG) Final report. M. S. S. Laboratory(MSSL). Surrey, United Kingdom.

Soh, L. K. and C. Tsatsoulis (1999). "Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices." Geoscience and Remote Sensing, IEEE Transactions on37(2): 780-795.

Stengel, R. F. (2012) Lecture Slides for Space System design, Spacecraft Communications and Telemetry

Stras, L., D. D. Kekez, et al. (2003). "The Design and Operation of The Canadian Advanced Nanospace eXperiment (CanX-1)." Proc. AMSAT-NA 21st Space Symposium: 150-160.

Sutha, P. (2013). "Image compression with scalable ROI using adaptive huffman coding." International Journal of Computer Science and Mobile Computing2(4): 478-484.

Syrlinks, (2015) "X-band transmitter can trasmit up to 13.3 GB per pass with a 5 m station, designed fo r cubesat and nanosatellites (LEO)" ", Syrlinks space defense and security, access 25 May 2015,

Tahoces, P. G., J. R. Varela, et al. (2008). "Image compression: Maxshift ROI encoding options in JPEG2000." Computer Vision and Image Understanding109(2): 139-145.

Talukder, K. H. and K. HaradaII (2007). "Haar Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image.": IJAMIAENG International Journal of Applied Mathematics.

Tanner (2012). "Seven grayscale conversion algorithms (with pseudocode and VB6 source code)." from http://www.tannerhelland.com/3643/grayscale-image-algorithm-vb6/.

Teney, D. (2009). Design and implementation of on-board processor and software of student nanosatellite OUFTI-1. Master of Computer Science. Belgium, University of Liege.

Toorian, A., K. Diaz, et al. (2008). The CubeSat Approach to Space Access. Aerospace Conference, 2008 IEEE.

Traussnig, W. (2007). Design of a Communication and Navigation Subsystem for a CubeSat Mission. Post-graduate University Course Space Science. Graz, Austria, Karl Franzens University of Graz. Master of Sciences.

Tuia, D., M. Volpi, et al. (2011). "A Survey of Active Learning Algorithms for Supervised Remote Sensing Image Classification." Selected Topics in Signal Processing, IEEE Journal of5(3): 606-617.

Ubbels, W. J., A. R. Bonnema, et al. (2005). Delfi-C3: a student nanosatellite as a test-bed for thin film solar cells and wireless onboard communication. Recent Advances in Space Technologies, 2005. RAST 2005. Proceedings of 2nd International Conference on.

Usevitch, B. E. (2001). "A tutorial on modern lossy wavelet image compression: foundations of JPEG 2000." Signal Processing Magazine, IEEE18(5): 22-35.

Vega, R., M. A., P. Sanchez, J. M., et al. (2004). An optimized architecture for implementing image convolution with reconfigurable hardware. Automation Congress, 2004. Proceedings. World.

Ves, E. d., X. Benavent, et al. (2006). "Selecting the structuring element for morphological texture classification." Pattern Anal. Appl.9(1): 48-57.

Wang, X. (2009). Image Edge Detection Based on Adaptive Selection the Value of Structure Element of Mathematical Morphology. Electronic Computer Technology, 2009 International Conference on.

Warren, L., Stutzman Gary A. Thiele (2013). Antenna Theory and Design ; third edition. J. W. Sons.

Waydo, S., D. Henry, et al. (2002). CubeSat design for LEO-based Earth science missions. Aerospace Conference Proceedings, 2002. IEEE.

Wertz, J. R. and W. J. Larson (1999). Space Mission Analysis and Design, Microcosm.

Wikimedia (2013, 4 June 2013). "Cycles per instruction." Retrieved 25 July, 2013, from http://en.wikipedia.org/wiki/Cycles_per_instruction.

Xiao-hong, Z.-y. Zhu, et al. (2009). Region of Interest Automatic Extraction for Color Image Based on Mathematical Morphology.

Xiaodong, G., W. Huaichao, et al. (2009). Design and implementation of image compression core based on CCSDS algorithm. Computer Science & Education, 2009. ICCSE '09. 4th International Conference on.

Xiuming, J. and W. Jing (2011). Research of remote sensing classification about land survey based on SVM. Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on.

Ye, S., Y. Gu, et al. (2008). An automatic ROI coding algorithm based on multi-resolution target detection. Systems and Control in Aerospace and Astronautics, 2008. ISSCAA 2008. 2nd International Symposium on.

Yu, G., Vladimirova, T., Sweeting, M. (2009). "Image compression systems on board satellites." Acta Astronautica64(9–10): 988-1005.

Zyren, J. and A. Petrick (2000) Tutorial on Basic Link Budget Analysis.