

RELIABLE GROUND CONTROL
POINTS FOR REGISTRATION OF
HIGH RESOLUTION SATELLITE
IMAGES

Kyaw Sann Oo

A dissertation submitted to
Kochi University of Technology
in partial fulfillment of the requirements
for the degree of

Doctor of Engineering

Graduate School of Engineering
Kochi University of Technology
Kochi, Japan

September 2010

Reliable Ground Control Points for registration of high resolution satellite images (English)

Kyaw Sann Oo

RELIABLE GROUND CONTROL
POINTS FOR REGISTRATION OF
HIGH RESOLUTION SATELLITE
IMAGES

Kyaw Sann Oo

A dissertation submitted to
Kochi University of Technology
in partial fulfillment of the requirements
for the degree of

Doctor of Engineering

(Engineering Course)
Graduate School of Engineering
Kochi University of Technology
Kochi, Japan

September 2010

ABSTRACT

The satellite remote sensing and photogrammetry communities are finding a relevant ways to use a ground control points in many occasions of image registration. As long as the research achieves a new technique in collection of GCP to replace conventional GCP collection methods, the work would be a big contribution to the dedicated communities. This research work presents an approach to collect ground control points using simulated image control points (ICP). The collected GCP can be used to register satellite images.

In the proposed method, a reference image is simulated from digital surface model and an already geocoded high resolution multispectral satellite image. However, the orthorectified aerial photograph is required in generation of high resolution ICP simulation; the research applied a very high resolution multispectral satellite image as reference image. Collected ICPs are used to match with raw target image to select conjugated GCPs using least squares matching (LSM). A LSM program is developed with six decimal place accuracy.

For precise image registration, accurate GCPs are required. Thus, a GCP database also developed with 2 cm accuracy of planetary measurement. Those GCPs are collected by VRS-GPS from accessible places in the Shokoku island of Japan.

GCPs selected by simulated ICP and GCPs collected by VRS-GPS are compared to present potential of new method by drawing the error vector of selected control points. The error vectors show the error trend and the trend presented which are come from the elevated hilly area.

Moreover, three common geometric transformation models are observed using accurate GCPs and generated GCPs. In the step, important factors of GCP (accuracy of GCP, accuracy of ICP, number of GCP and distribution pattern of GCP) are observed using these geometric models. The observation found that 3D affine transformation model is an adequate transformation model for registration of ALOS-PRISM; even though, the image covered wide area field view.

SUMMARY

The thesis is composed with seven chapters. Introduction is placed at first then follow by methodology. Transformation models are discussed later on as demonstration on three common geometric models: 3D affine, 3D projective and separated 3D projective transformation models. Then, important factors in image registration are presented in chapter 4 which are spatial distribution of control points and numbers of GCP. Image control points generations is the main part of the study and it is presented in chapter 5 including explanation on simulation of image control points. Results are presented in each chapter. The chapter 6 is discussion and all results are concluded. Finally, extendable future works are attached in chapter 7. Summary of each chapter is presented below.

Chapter 1 Introduction

The chapter 1 introduces the background of the study at first place. Continuously, the state of the art of problem is uncovered. The way of new thought to solve the problems are systematically presented in objectives section. The role of the research work is follow by the objective section. Finally, selected study area and the information about the data are explained.

Chapter 2 Methodology

Overall methodology of the study is presented in the chapter 2. The main research methodology is presented by showing flow diagrams.

Chapter 3 Transformation Models

In chapter 3, mathematics transformation models are presented. Selected models are 3D affine, 3D projective and separated 3D projective. Initial result of investigation on geometric transformation models is presented in this chapter. Visual interpreted GCP are used as the initial condition.

Chapter 4 Spatial Distribution of Control Points and Numbers of GCP

The chapter explained importance of ground control points in satellite image registration. VRS-GPS (Virtual Reference System - Global Positioning System) device is used to collect ground control points for validation of the result. At the same time, a very precise GCP database with useful GCP information is developed for the whole Shikoku island of Japan in parallel with the research. These accurate GCPs are collected with VRS-GPS and already public the GCP database thought internet.

Two major factors of image registration are presented in chapter 4. Spatial distribution of control points and number of ground control points are considered. Error propagation on the satellite image by geometric models is demonstrated using three selected transformation models. By adding random error to control points, error trend and error vector are explored the error geometry. Moreover, a section included demonstration of how many numbers of GCP is a better choice. The evaluation of transformation models is followed by observing the empirical simulation with sample data.

On the other hand, the image geometry of nadir look satellite image and oblique (backward look) satellite image are observed using accurate GCPs.

Chapter 5 Image Control Points

Chapter 5 is the main part of the study and it is composed of a new thought of GCP selection using image control points as of point/vector GCP. At first, a reference image is simulated from existing high resolution multispectral satellite image and digital surface model. In simulation, a DSM is used to generate shade/shadow for both of the target and the reference image. Reference template patches (image control points - ICP) are simulated. Consequently, result of LSM is presented. Finally, error vectors are drawn based on the GCP and CP which are selected using LSM. All simulated ICP are verified with VRS-GPS GCP.

Chapter 6 Discussion

Discussions on all results are presented in this chapter. Discussion is done from both of positive or negative views. Whilst some points give advantage of using of the new method, some limitations are appeared and those are presented as controversy issues.

Chapter 7 Future works

Future studies, which can be continued in future, are dedicated in this chapter. Most of future works presented in this chapter are relating to main methodology of the study and which can appear as main branches of the future research. Suggestions, which should continue to make more effective research with this works, are presented in this chapter.

ACKNOWLEDGMENTS

I would like to express my sincerely thanks to advisor professor Masataka TAKAGI for his overall supports to form this research fulfillment. With his strong supports, advising and guidance; this research is successfully done. Continuously, I would like to add my gratitude to supervisory committee members: Seigo NASU, Kazuo OUCHI, Nobumitsu FUJISAWA, and Masayuki MATSUOKA for their valuable suggestions and comments to improve my research works.

It is impossible to skip Dr. Lal Samarakoon, Director of geoinformatics center of Asian Institute of Technology, allowed me to accept this research opportunity. A great supports for this study is the doctoral scholarship program of Kochi University of Technology, called Special Scholarship Program (SSP), without this SSP scholarship it is impossible to come and to do research in Japan. I would like to add my gratitude to all staffs of International Relationship Center of KUT for their valuable guidance on living, institutional documentation processes and many other helps. Moreover, all of Infrastructure Systems Engineering Department and KUT staffs for their direct/indirect supports to fulfill my research works.

I would like to mention all coworkers, who are collaborating in this research, from the Takagi laboratory

for their helps and sharing their times and knowledge in my endurance on the research.

I would like to mention to the great five important things in Buddhism (Buddha, Dhamma, Sangha, Parents and Teachers) which I obeys in my life. Finally, my work is dedicated to my long suffering and patient companion of years, my wife and my two sons, with love, affection and giving beautiful family life.

TABLE OF CONTENTS

Abstract.....	ii
Summary.....	iii
Acknowledgments.....	v
Table of Contents.....	vii
List of Figures.....	ix
List of Tables.....	xiii
List of Appendices.....	xv
Glossary /Abbreviation.....	xvi

CHAPTER 1 INTRODUCTION

1.1 Background.....	1
1.2 Problems.....	5
1.3 Objectives.....	7
1.4 Role of the research.....	8
1.5 Study area.....	8
1.6 Data.....	9
1.6.1 Digital Surface Model.....	10
1.6.2 Satellite images.....	10

CHAPTER 2 METHODOLOGY

2.1 Main methodology.....	14
---------------------------	----

CHAPTER 3 TRANSFORMATION MODELS

3.1 Parallel transformation model.....	18
3.1.1 3D affine transformation model.....	20
3.2 Perspective transformation model.....	21
3.2.1 3D projective transformation model.....	22
3.2.2 Separated 3D projective transformation model.....	23
3.3 Investigation of geometric error propagated by transformation models.....	24
3.4 Result of transformation models.....	31

CHAPTER 4 SPATIAL DISTRIBUTION OF CONTROL POINTS AND NUMBERS OF GCP

4.1 Ground Control Point.....	32
4.2 Development of GCP database.....	33
4.3 Background of the study.....	35
4.4 Overview of error propagation study.....	37

4.5	GCP and check points	38
4.6	Method of random error generation	39
4.7	Transformation methods.....	40
4.8	Addition of errors	42
4.9	Error vectors.....	42
4.10	Discussion on error trends.....	46

CHAPTER 5 IMAGE CONTROL POINTS

5.1	Background	47
5.2	Main thought of ICP.....	52
5.3	Data	54
5.4	Reference image generation	54
5.4.1	Removal of shade and shadow affects.....	55
5.4.2	Extraction of reflectance value & reflectance image simulation	58
5.4.3	Addition of shade and shadow for target image	59
5.5	Image matching	59
5.5.1	Selection of search image window	62
5.5.2	Result of image matching	62
5.6	Registration result using LSM generated ICP.....	65
5.7	Image registration	69
5.7.1	Result of image registration	69
5.8	Error propagation for geometric models observation	71
5.9	Error vectors for geometric transformation models	74

CHAPTER 6 DISCUSSION

6.1	Result of Transformation Models.....	75
6.2	Error Trends Based on the Distribution of GCP.....	75
6.3	Result of image matching	76
6.4	Result of image registration.....	77
6.5	Role of Research	78

CHAPTER 7 FUTURE WORKS

7	Future works	93
---	--------------------	----

LIST OF FIGURES

<i>Number</i>	<i>Page</i>
<i>Figure 1.1: Overview of image registration</i>	<i>5</i>
<i>Figure 1.2: Problematic issues of satellite images</i>	<i>6</i>
<i>Figure 1.3: GCP distribution and image geometry</i>	<i>7</i>
<i>Figure 1.4: Model of image simulation</i>	<i>8</i>
<i>Figure 1.5: Study area</i>	<i>9</i>
<i>Figure 1.6: Digital Surface Model</i>	<i>10</i>
<i>Figure 1.7: High resolution reference satellite image (IKONOS)</i>	<i>11</i>
<i>Figure 1.8: High resolution target satellite image (ALOS-PRISM)</i>	<i>13</i>
<i>Figure 2.1: Conventional Registration method</i>	<i>14</i>
<i>Figure 2.2: Image registration with RPC</i>	<i>15</i>
<i>Figure 2.3: Main methodology</i>	<i>16</i>
<i>Figure 2.4: Detail methodology</i>	<i>16</i>
<i>Figure 2.5: Breakdown methodology</i>	<i>17</i>
<i>Figure 3.1: 3D affine transformation</i>	<i>21</i>
<i>Figure 3.2: 3D projective transformation</i>	<i>23</i>

LIST OF FIGURES (CONTINUE)

<i>Number</i>	<i>Page</i>
<i>Figure 3.3: Separated 3D projective transformation</i>	<i>23</i>
<i>Figure 3.4: Distribution of GCP and CP on PRISM image</i>	<i>24</i>
<i>Figure 3.5 (a): Accuracy of distribution of GCP and number of GCP on PRISM images</i>	<i>28</i>
<i>Figure 3.5 (b): Accuracy of distribution of GCP and number of GCP on PRISM images</i>	<i>28</i>
<i>Figure 3.5 (c): Accuracy of distribution of GCP and number of GCP on PRISM images</i>	<i>29</i>
<i>Figure 3.5 (d): Accuracy of distribution of GCP and number of GCP on PRISM images</i>	<i>29</i>
<i>Figure 3.5 (e): Accuracy of distribution of GCP and number of GCP on PRISM images</i>	<i>30</i>
<i>Figure 3.5 (f): Accuracy of distribution of GCP and number of GCP on PRISM images</i>	<i>30</i>
<i>Figure 3.6: The result of visual interpreted GCP</i>	<i>31</i>
<i>Figure 4.1: GCP and CP</i>	<i>32</i>
<i>Figure 4.2: GCP database of Takagi Laboratory</i>	<i>33</i>
<i>Figure 4.3: VRS-GPS</i>	<i>34</i>
<i>Figure 4.4: GCPs distribution in remote area</i>	<i>34</i>
<i>Figure 4.5: GCP and its information</i>	<i>35</i>

LIST OF FIGURES (CONTINUE)

<i>Number</i>	<i>Page</i>
<i>Figure 4.6: Error depends on number of GCP and distribution (credit: Guoqing Zhou and Ron Li 2000)</i>	<i>36</i>
<i>Figure 4.7: The general flow of error geometry generation</i>	<i>37</i>
<i>Figure 4.8: Location of distribution of control points</i>	<i>38</i>
<i>Figure 4.9: Distributions of GCP and CP</i>	<i>41</i>
<i>Figure 4.10(a): Error vectors of each distribution (3D affine)</i>	<i>43</i>
<i>Figure 4.10(b): Error vectors of each distribution (3D projective)</i>	<i>44</i>
<i>Figure 4.10(c): Error vectors of each distribution (Separated 3D projective)</i>	<i>45</i>
<i>Figure 4.11: Selected distribution patterns</i>	<i>46</i>
<i>Figure 5.1: Factors causing exposure falloff</i>	<i>50</i>
<i>Figure 5.2: Sun-object-image angular relationship</i>	<i>51</i>
<i>Figure 5.3: Geometric effects that cause variations in focal plane irradiance</i>	<i>52</i>
<i>Figure 5.4: ICP as of GCP</i>	<i>53</i>
<i>Figure 5.5: Methodology of image control points generation</i>	<i>54</i>
<i>Figure 5.6: Reference image simulation</i>	<i>55</i>

LIST OF FIGURES (CONTINUE)

<i>Number</i>	<i>Page</i>
<i>Figure 5.7: Hill shading parameter by the sun illumination</i>	<i>56</i>
<i>Figure 5.8: Algorithm for shade/shadow removal</i>	<i>57</i>
<i>Figure 5.9: Algorithm for reference image generation</i>	<i>58</i>
<i>Figure 5.10: Algorithm for PRISM image simulation</i>	<i>59</i>
<i>Figure 5.11: Relation of search window size and residual</i>	<i>62</i>
<i>Figure 5.12: Simulated image and GCP/CP locations</i>	<i>63</i>
<i>Figure 5.13: Comparison of Selected GCP patches</i>	<i>64</i>
<i>Figure 5.14: Distribution of selected GCP and CP on reference IKONOS image</i>	<i>66</i>
<i>Figure 5.15: Error vector of GCP and CP</i>	<i>74</i>

LIST OF TABLES

<i>Number</i>	<i>Page</i>
<i>Table 1.1: Specifications of reference satellite images</i>	<i>11</i>
<i>Table 1.2: PRISM characteristics and observation modes</i>	<i>12</i>
<i>Table 1.3: Specifications of used target images</i>	<i>13</i>
<i>Table 3.1: Used GCP and CP values</i>	<i>25</i>
<i>Table 3.2: Results of distribution patterns and number of GCP (Unit: Pixel)</i>	<i>26</i>
<i>Table 3.3: Result of transformation models using visual interpreted GCP (Unit: Pixel)</i>	<i>31</i>
<i>Table 4.1: Geometric transformation result for each pattern around GCP</i>	<i>46</i>
<i>Table 4.2: Geometric transformation result for each pattern around CP</i>	<i>46</i>
<i>Table 5.1: Least Squares Image Matching Results</i>	<i>65</i>
<i>Table 5.2: Selected GCP and CP</i>	<i>67</i>
<i>Table 5.3: Result of transformation models using LSM generated GCP</i>	<i>70</i>
<i>Table 5.4: Registration results for each GCP and CP (Unit: Pixel)</i>	<i>70</i>
<i>Table 5.5: Error propagation to geometric transformation results (Unit: Pixel)</i>	<i>72</i>
<i>Table 6.1: Selection of Window size</i>	<i>76</i>
<i>Table 6.2: GCP/CP RMSE comparison between visual collection, simulation and VRS</i>	<i>77</i>

<i>Table 6.3: Observed ICPs, LSM results and deviation of LSM results from initial values</i>	<i>80</i>
<i>Table 6.4: Observed ICPs with histograms, horizontal profiles and vertical profiles</i>	<i>81</i>
<i>Table 6.5: Relationship of residual and window sizes and selected ICP 101×101 windows</i>	<i>88</i>

LIST OF APPENDICES

<i>Number</i>	<i>Page</i>
<i>Appendix A: Geometric transformation model code (example: 3D affine)</i>	<i>94</i>
<i>Appendix B: Error vector generation (ArcView GIS avenue script)</i>	<i>100</i>
<i>Appendix C: Least Squares Matching C code</i>	<i>110</i>
<i>Appendix D: Definition of remote sensing</i>	<i>130</i>
<i>Appendix E: Milestone in the history of remote sensing</i>	<i>132</i>
<i>Appendix F: Key factors of remote sensing</i>	<i>133</i>
<i>Appendix G: Application usages</i>	<i>134</i>
<i>Appendix H: How to access GCP database through Internet?</i>	<i>136</i>
<i>Bibliography</i>	<i>141</i>

GLOSSARY/ABBREVIATION

3D. Three Dimension space (x, y, z)

ALOS. Advanced Land Observing Satellite

CP. Check Point

DGPS. Differential Global Positioning System

DN. Digital Number

DSM. Digital Surface Model

GCP. Ground Control Point

GIS. Geographic Information Systems

GPS. Global Positioning System

ICP. Image Control Point

IKONOS. A commercial earth observation satellite with the availability of high resolution imagery

JAXA. Japan Aerospace Exploration Agency

LSM. Least Squares Image Matching Method

MOMS-2P. Modular Optoelectronic Multispectral Stereo Scanner

nm. Nanometer

PRISM. Panchromatic Remote-sensing Instrument for Stereo Mapping

RPC. Rational Polynomial Coefficients

RS. Remote Sensing (Satellite RS in this study)

VRS-GPS. Global Positioning System with Virtual Reference System

INTRODUCTION

1.1 Background

With the advancement of space technology, many earth observation satellites are acquiring atmosphere, land and ocean data of the earth surface in all the time (*MSS 1972, TM 1982, JERS1-OPS 1992, MOM2P-1996, HRVIR 1998, ETM+ 1999, IKONOS 1999, QuickBird 2001, WorldView-1 2007, GeoEye-1 2008*). Even for the same place; the images are differently acquired in spatial or temporal or and radiometric resolution based on their physical sensors. When the images are registered to specific location on the ground with related control points, the image becomes usable spatial information for many applications such as changes detection, natural disaster mapping, environmental management and monitoring. Thus, image registration is one of primitive tasks in satellite remote sensing.

There are two main branches in the satellite remote sensing; optical remote sensing and microwave remote sensing based on onboard sensors of satellite. In optical remote sensing, reflection of the solar energy or emission spectral of the earth surface energy is sense by mechanical or digital camera. Acquired spectral reflectance values of the earth surface are sampling to digital numbers as image pixels in digital satellite image. The spectral region used by the optical remote sensing is ranging from visible to near infrared and far infrared in some cases. Most of optical sensors are built to acquire spectral regions as digital bands: panchromatic, blue, green, red and near infrared. All these bands are appeared in multi-spectral images. However, image bands are collectedly define by the reflectance spectral of electromagnetic region from ground surface features; the defined values range are differ to each sensor (e.g. blue band: 420~500nm, green band: 520~600nm, red band: 610~690nm, near infrared band: 760~890nm for ALOS AVNIR-2 image; panchromatic band: 520~770nm for ALOS PRISM image; and panchromatic band: 526~929nm, blue band: 445~516nm, green band: 505~595nm, red band: 632~698nm, and near infrared band: 757~853nm for IKONOS image). Based on the differences of the spectral range, each specified digital numbers for ground features become difference. Moreover, the sampling spacing of the ground distance to the image grid (image spatial resolution) is also differing for each sensor. There are varieties of spatial resolution (ranging from sub-meter to a kilometer in pixel) are available nowadays. The reflectance of the ground features also depend on the spatial resolution. When spatial resolution becomes higher the visibility to the ground

features coming finer. Many optical sensors are schedule to acquire the ground surface features in day time especially around 10:00~11:00 AM on local time of the ground position. The rising sun gives enough electromagnetic energy for reflection of the ground surface to satellite sensor to form image's grid pixels in the image. Pixels are two dimensional arrays of reflectance values which are appear as Digital Number in the image. The DNs are influence by spatial resolution, spectral resolution and temporal resolution of satellite sensor. On the other hand, the ground surface features and topography are radiometrically/geometrically influence to DN values.

When images are integrated and continuously analyzed, the image data become useful information for users such as land planning engineers, foresters, agricultures, environmentalists, cartographers, GIS users, and many others. Moreover, images are needed to register each other before analysis processes start. There are many registration methods in remote sensing such as map-to-image registration, image-to-image registration as of conventional techniques and advance technique using Camera Replacement Mathematical Model with Rational Polynomial Coefficients. Conventional methods need many Ground Control Points to overlay one image onto another image when the advance methods are automatically geometrically correcting with a few GCP. GCPs are collect from the topographic map or collect by positioning devices (e.g. GPS, DGPS, VRS-GPS) in map-to-image registration techniques. On the other hand, the GCPs are collect from already geocoded satellite image in image-to-image registration. In this case, image-to-image registration is easier and faster than map-to-image registration; however, accurate GCPs are needed to improve registration accuracy of output images. Nowadays, map users are collecting GCP from the geocoded image to register new images; however, errors can be contaminated in the image registration processing with less accuracy in information. The errors can be come from poor visibility or features identification difficulty on image; the act of delaying of acquisition time; geometric orientation of ground topography contribution to the sun illumination; difficulties to identify the ground features in the image; or the changes of reflectance by surface objects.

Many researches are done with excellent accuracy in geometric correction when they use Rational Polynomial Coefficients (RPC) rivaling for the highly accumulated mathematical calculation works. On the other hand, some satellite images are still unable to provide RPC for geometric correction. In the contest, the research is able to present usefulness of simple mathematical models using GCP for accurate image registration. The achievement could be done after studying on the main influencing factors of image registration: accuracy of control points, distribution of control points, number of control points and transformation models.

Ebner and Strunz, (1988); Ebner et al., (1991, 1992); Fraser and Shao, (1996) are investigated the accuracy of Ground Control Point using MOMS-02 image. *Guoqing Zhou and Ron Li (2000); Li Z. (2000)* evaluated number of GCP and distribution patterns of GCP for IKONOS data. As the use of datasets, *Grodecki, 2002; Jacobsen, 2003; Gene Dial and Jacek Grodecki 2005* demonstrated that Very High Resolution satellite imagery has possibility to replace aerial photo in many photogrammetric applications. *Kersten et al., 2000* demonstrated accuracy of Image Control Points using Least Squares Image Matching. *Jacobsen, 2006* announced that 3D projective and 3D affine transformation required 15 GCP for IKONOS to reach the level of one pixel accuracy.

There are many factors which are mainly influences on geometric accuracy of image registration as presented above: mainly; the accuracy of GCP, accuracy of ICP, transformation models, numbers of GCP and distribution of GCP. However, the existing researches are covered all main factors and proof that the accurate GCP plays key role in image registration; a wide range of image extents are available as many choices. The transformation models could be depended on the spatial extent of images in registration process. The study investigated on the extents of ALOS-PRISM images; the first one is full extent of forward look and the nadir look and the second is subset of the image. Parallel and perspective transformations could be influence on a wide coverage of PRISM image, whilst the ground features are observed by the subset image. However, many researches observed on a narrow coverage (e.g., IKONOS); this research work evaluated on both of wide full coverage of PRISM image and the subset of the image to present images' consistency.

Guoqing Zhou and Ron Li (2000) evaluated simulated IKONOS image using bundle adjustment photogrammetric model with factors of number of GCP and their distribution patterns. The result showed that the average (x, y, z) error can reach 12 m without GCP. 4 GCP improved accuracy largely. They suggested that geometric accuracy cannot improve by increasing the number of GCP. Their results are matches with results from MOMS-02 (*Fraser and Shao, 1996; Ebner et al., 1991*). On the other hand, an affine model could precisely represent a satellite image, if the view angle were perfectly constant throughout the image. The affine model breaks down to the extent view angle is not constant throughout the image (*Gene Dial and Jacek Grodecki 2005*). For getting high accuracy, control points are required. 3D affine transformation to the control points should be used (*Jacobsen, 2002b*). In addition, control points have to use for exact location of the sense and RPC can be computed with many control points but this requires well distributed control points (*Jacobsen, 2003*). Semi-Automated Least Squares Image Matching (*Gruen 1984; Baltsavias and Stallmann, 1993; Kersten et al., 2000*) resulted 7.2 meter XY RMS when IKONOS image is matched with aerial orthophoto using affine transformation. In *Toutin, Th. (1995, 2003)* observation, the residuals of 3D

parametric model don't reflect the modeling accuracy but rather the errors in the input data when there are more GCP than the minimum required. Moreover, he found that the accuracy of Bundle Adjustment model is 1/3 pixel for medium resolution images and 1-2 pixel for high resolution images based on accurate GCP. In *Takeo Tadono et al. (2006)* observation, the geometric error exceeds 300 meters in the line direction (Y) for PRISM's nadir image. Error vectors were calculated using GCP and ancillary information of the image in his research. *Toshiaki Hashimoto (2006)* suggested that GCP are necessary in precise geometric correction. He evaluated the necessary number of GCP using error pattern.

Moreover, the difficulty of GCP collection in remote area and difficulty of identification of ICP on the image demanded a new method of GCP selection. *Yamakawa, 2002* used the centroid of rotaries on the road as control points. *Miyata, 2005* demonstrated the accurate geometric correction using centroid as control point in his master thesis. *Kunitomo, 2009* presented the used of road cross section and road curve as image control point.

The primitive task of this study is how to acquire GCP automatically. The former research on image matching shown that matching between reference image and target image is the powerful way of image registration (*Ackermann 1984, Gruen 1985, Förstner 1986*).

It is possible to reduce the errors using reference image simulated from very high resolution satellite image or air-photographs. The main thought of this study is simulation of reference image from a set of very high resolution satellite image and digital surface model (*RAMS-e 2000*). GCP image patches are collected from the simulated reference image. The GCP are image patches in this case when GCP are point in general. Then, conjugated GCP are selected by Least Squares Image Matching Method (*Gruen 1985*) between simulated GCP image patches and the target image. The selected GCP pairs are used to register the target image in registration process.

The study conducted investigation into a selection of ground control points for image registration using simulated image. Three data sets are used in this study; one meter spatial spacing digital surface model (DSM) data as topographic high source (*Proy et al 1989, Meyer et al 1993, Tzelepis and Nakos 2003*); one meter spatial resolution IKONOS multi-spectral image as reference reflectance value of earth surface; and ALOS-PRISM image as a target image which is needed to register. Data acquisition dates are different from each other. First, two sets of shade relief data are generated from DSM: one for reference reflectance value extraction and the other for target image simulation. In reference reflectance values generation, shade and shadow are replaced from the multi-spectral bands by one of generated shade relief (*Murai 1998, Murai et al 1998, 濱田哲伸, 2004*). Then, reference reflectance

value is generated from the average of the green ratio and red ratio (Holben and Justice 1980, Lillesand and Kiefer 1994, Laben et al 2000). Second, a simulated image is generated by integration of reference reflectance values and another shade relief data, which is generated with PRISM image acquisition parameters. Then, GCP are collected by LSM method from the simulated image. Finally, a transformation models are applied to investigate the errors image registration result with selected GCPs.

1.2 Problems

With the advancement of space technology, many earth observation satellites are acquiring atmosphere, land and ocean data of the earth surface in all the time. The data are downlinks to the operating ground stations as satellite imagery. However the sensors are observing the same points or places on the ground; the images are difference in spatial, temporal or radiometric resolution based on their onboard physical sensors. When the images are registered to a location on the ground with conjugated control points, the registered images become usable as useful spatial information for many applications such as changes detection, natural disaster mapping, and environment management and monitoring. Thus, image registration is one of primitive tasks in satellite remote sensing.

There are many registration methods in satellite remote sensing such as map-to-image registration, image-to-image registration (figure 1.1). They needed ground control points to overlay one data onto another data with very high precision. There are four important factors in satellite image registration. They are:

- Ground Control Point coordinates
- Image Control Point coordinates
- Geometric Model
- Spatial Distribution of Control Points

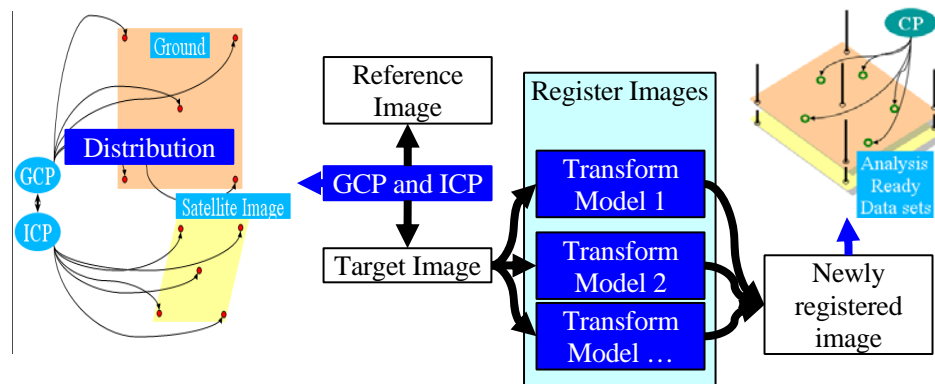


Figure 1.1: Overview of image registration

GCPs are collected from the topographic map or collected by positioning devices (e.g. GPS, DGPS, VRS-GPS) in map-to-image technique. On the other hand, GCPs are collected from the geocoded satellite image in image-to-image registration. In this case, image-to-image registration is easier and faster than map-to-image registration. Nowadays, map users are collecting GCP from the geocoded image to register new raw images; however, errors can be contaminated in the image integration processing with less accuracy in information (*Flowerden and Openshaw 1987, Maling 1989, Veregin 1989, Chrisman 1989, Maffini et al. 1989, and Lodwick 1989 among others*). The errors can be come from the act of delaying of acquisition time; geometric orientation of ground topography contribution to the sun illumination; difficulties to identify the ground features in the image; or the changes of reflectance by surface objects. The difficulty of GCP collection in remote area and difficulty of identification of point GCP on image demanded a new method of GCP selection from digital photogrammetry and satellite remote sensing communities.

The major problem, which consider in this research, is use of simulated image control points to replace conventional GCP collection. The problem is based on four problematic issues (*figure 1.2*): (1) high resolution satellite images are affecting by shade and shadow in fully interpretation and analysis of those images; (2) reflectance of ground surface features are influence upon satellite image; (3) GCP collection difficulties in the remote area; and (4) identification difficulty of GCP points in satellite images.

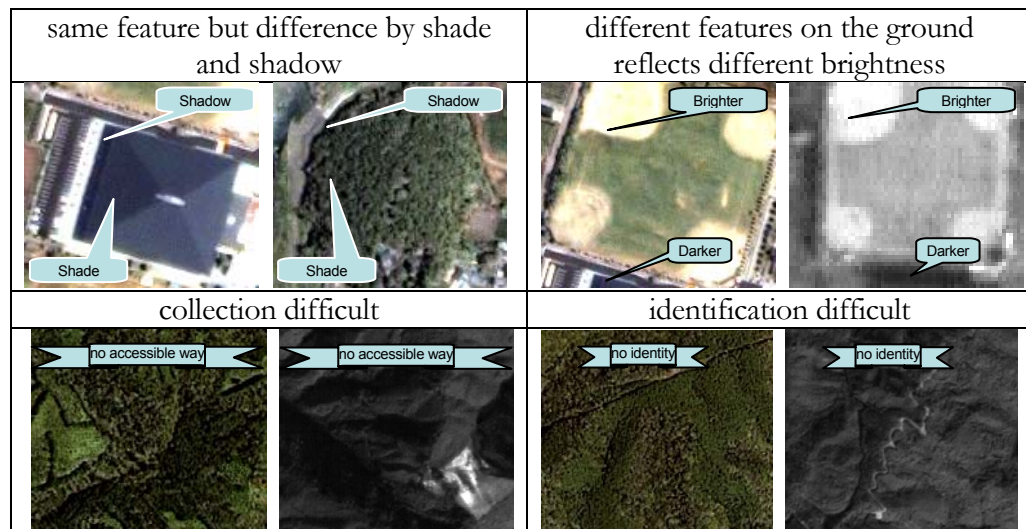


Figure 1.2: Problematic issues of satellite images

The new approach objectively aiming to explore reliable GCP using image patches (ICP) to replace conventional GCPs. ICP image patches are collect from

the simulated reference image. The reference image is simulate from existing registered digital surface model and registered aerial photograph or very high resolution satellite image. Shade and shadow affects are removed by the processes of deep shaded masking, fellow by bands ratios calculation and averaging the ratios for spectral adjustment between reference and target satellite images. Finally, conjugate GCP pairs are collect by least squares image matching.

1.3 Objectives

The primitive task of this research is how to acquire GCP automatically from simulated satellite image. The former research shown that least squares matching between reference image and target image is the powerful way of conjugated point collection (*Ackermann 1984, Gruen 1985, and Förstner, 1986 amount others*), if initial value provided as seed to calculation the iteration.

It is possible to calculate conjugated points using simulated satellite image from very high resolution digital surface model and reference authorectified aerial photo. When DSM dataset is available, shade and shadow can be simulated using satellite altitude parameters and the solar illumination parameters. Though, an image can be simulated from DSM, the simulated image cannot be totally similar to real satellite image. Moreover, spectral reflectance of the earth surface objects might be influences on the image’s brightness values. Thus, an authorectified aerial photographs or a very high resolution satellite image is require to generate spectral reflectance data while the seasonal spectral database is not available. The main thought of this study is simulation of reference image from a set of very high resolution satellite image and digital surface model. ICP image patches are collected from the simulated reference image. Then, conjugated GCP are select by least squares matching between simulated ICP image patches and the target image. The selected GCP values are used to register the target image. Finally, the error vectors are draw for each geometric transformation model to conclude error trends (*figure 1.3*).

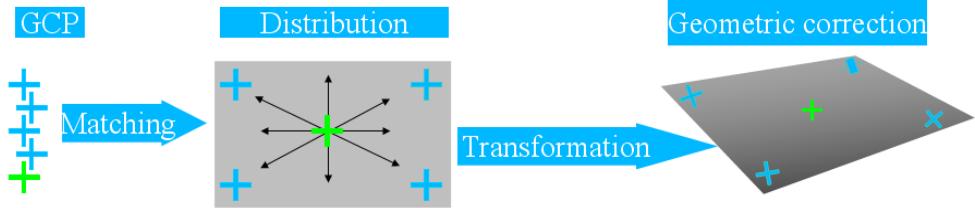


Figure 1.3: GCP distribution and image geometry

Simulation of ICPs has high possibility to use as GCP adding the ground status such as spectral reflectance of each object and the sun elevation and azimuth for shade and shadow (*figure 1.4*).

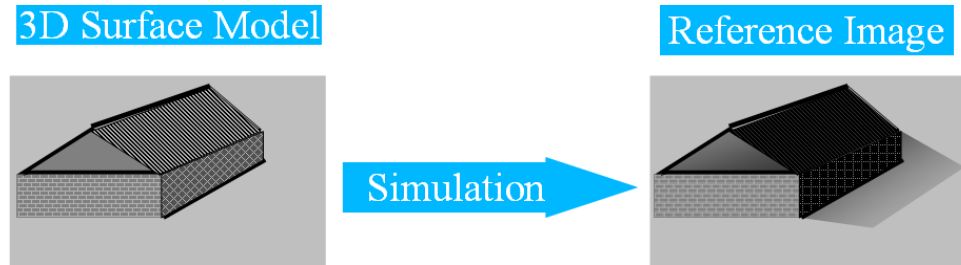


Figure 1.4: Model of image simulation

The research confirmed that the use of ICP as GCP by demonstrating transformation results comparison between generated GCP and measured GCP. The GCP are measured using VRS-GPS in the research for verification. Moreover, a GCP database is developed for the whole Shikoku island of Japan as part of the research. The required seasonal spectral database is developing as future research work.

1.4 Role of the research

The roles of the research which predicted as important contribution to the satellite remote sensing communities are listed below:

- possibility of automatic ground control point collection
- concrete the satellite image registration with high accuracy
- pattern recognition and automatic target recognition in robotic industry and
- development of related database which helping geoinformatics communities

The first two roles could be provided after the problems are uncovered by the research. The automatically selected ICP image patches are usable for pattern recognition and automatic target recognition in machine vision (*Davies 1997, Leondes 1998, You et al. 1999 and Snyder et al. 2004*). Thus, the research can be continues to future pattern and target recognition of robotic and automation industry by well understanding upon simulated 3D image control point.

1.5 Study area

A place, which involved various types of ground features, is selected as study area. It is situated on southern part of Nankoku city and on western part of Noichi city in Kochi prefecture, Japan. The study area is covered with artistic structures such as orchards, green houses, Kochi Ryoma Airport, sea dikes, bridges, roads and buildings; natural features such as mountains with forest cover; and Monobe River with sandbars and gravels. The red rectangular box is present to locate the study area in Japan in the figure (*figure 1.5*).

The area is also considered in the distribution of 3-axes (x, y, z) such as elevation, even/uneven land and waterbody. This selection also considers on the availability of required datasets (DSM and high resolution satellite images).

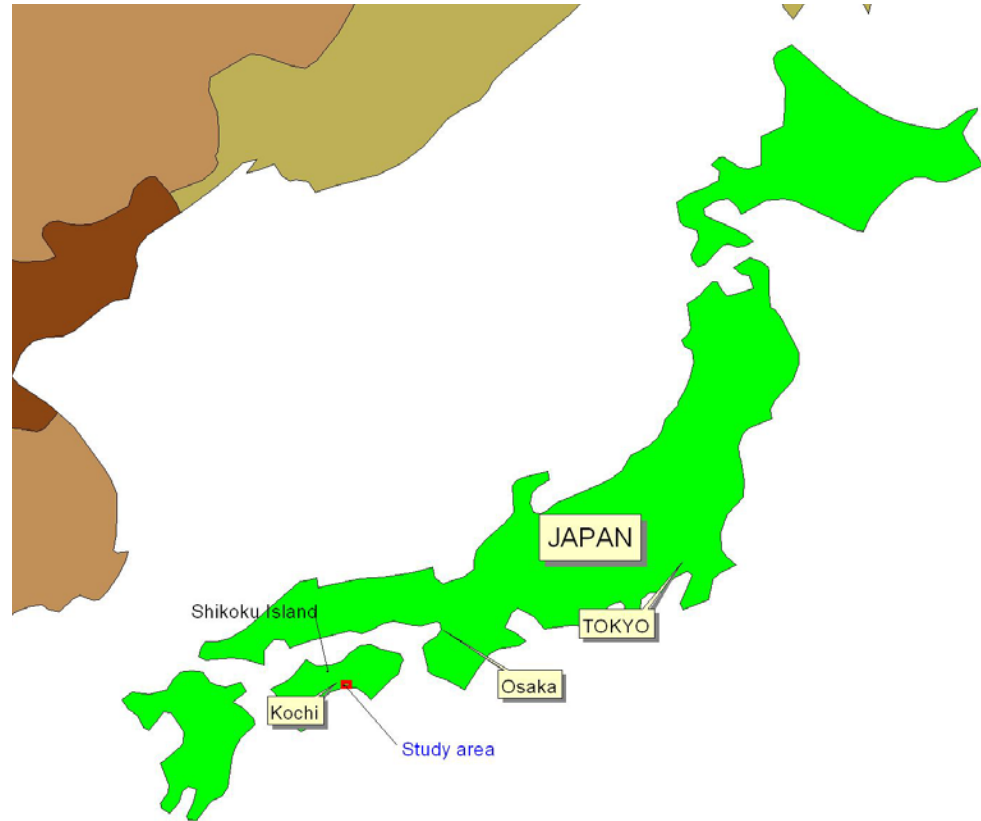


Figure 1.5: Study area

Moreover, the selected study area is closed to the research laboratory and very convenient to do field measurement for verification the research works.

1.6 Data

As the primitive thought of this study is to generate image control points (ICP) which can be used to select GCP automatically using LSM image matching technique. Ortho aerial photos are required to generate very high resolution ICP. Based on unavailability of ortho aerial photo for selected study area; the research used a high resolution DSM (Airborne Laser Scanner) and a high resolution multispectral satellite image (a pan-sharpened IKONOS image) to generate ICP. On the other hand, a raw image (ALOS-PRISM nadir image) is used as target image.

Thus, three data sets are used in the study, two of which are high resolution satellite images and the other of which is one meter ground spacing DSM.

1.6.1 Digital Surface Model

Airborne laser scanner DSM data (*figure 1.6*) plays a key role in the study. Shade and shadow data are generated from the same DSM with different sun illumination parameters (sun elevation angle and sun azimuth angle) for the reference and the target images. The DSM data is provided from RAMS-e: Kokusai Kogyo Co., Ltd., to Takagi laboratory for research purposes. The ground sampling resolution of DSM data is 1 meter.



Figure 1.6: Digital Surface Model

The bright features represent the elevated area and the dark represented the lowland of the study area. The airport runway presented as a smooth tone in the lower part of the selected area. The sea waterbody is presented the lowest part of the selected study area. Artistic structures are appeared distinctly on the image.

1.6.2 Satellite images

Principally an aerial photo is often used as reference data; however, IKONOS image (IKONOS © 2000) (*figure 1.7*) is applied as a source of reference reflectance data in the study. ALOS-PRISM's Level 1B nadir image (JAXA © 2006) is used as the raw target image (*figure 1.7*). Important image specifications of reference image are listed in the table (*table 1.1*). On the other hand, spatial resolutions of both images are different. When the reference data is one meter

grid spacing, the target image is 2.5 meters. The sun illumination parameters of both images also differ.

Table 1.1 Specification of reference satellite image

	IKONOS
Acquired date:	2000-Jun-15
Sun elevation angle	69.25115°
Sun azimuth angle	114.7576°
Spatial resolution:	1 meter
Spectral range:	445 516 nm (blue) 506 595 nm (green) 632 698 nm (red) 757 853 nm (near infra-red)

Based on the acquisition spectral ranges of both sensors data, PRISM's panchromatic band can be generated by the average of IKONOS's red ratio and green ratio.

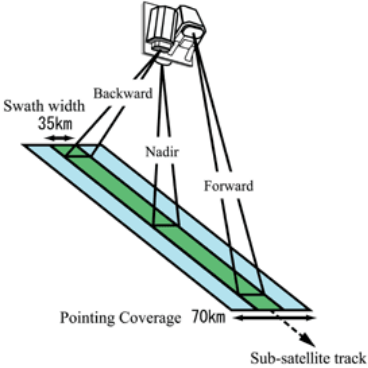
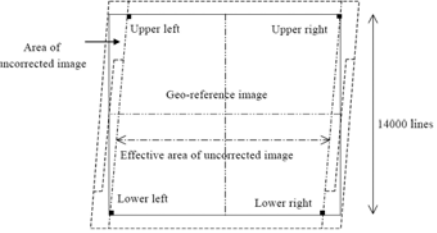


Figure 1.7: High resolution reference satellite image (IKONOS)

Target image can be all type of unregistered raw images which needed to register; though, a subset (4000 x 3000 pixels) of ALOS-PRISM nadir look level 1B2 image is assumed as target image in this study. Advanced Land Observing Satellite (ALOS) had three onboard sensors and Panchromatic Remote-sensing Instrument for Stereo Mapping (PRISM) sensor is one of the three onboard sensors of the ALOS. ALOS is observing the earth surface at a mean altitude of 691.65 km above the equator from the orbits of near-polar, near-circular and Sun-

synchronous. The ALOS satellite rotation as a mean revolution period about 17 hours with an inclination of 98.16°. PRISM is panchromatic band with 2.5 meter spatial resolution at nadir and PRISM can imagine the same place of earth surface with three look: forward, nadir and backward. PRISM is push broom sensor type and linear CCD arrays are apparatus to collect earth surface reflectance energy. Each CCD unit contains about 5000 detectors. PRISM images are produced in level 1B1 (radiometrically calibrated), 1B2R (systematically geo-referenced), 1B2D (corrected with coarse DEM) and 1B2G (systematically geo-coded) in CEOS format. The time gap between two looks observation of a same point of the earth surface is about 46 seconds (*Tadono 2006*). This short delay guarantees short time temporal change (*ALOS CAL/VAL 2006*). Characteristics and observation modes of PRISM are presented in the table (*table 1.2*).

Table 1.2: PRISM characteristics and observation modes

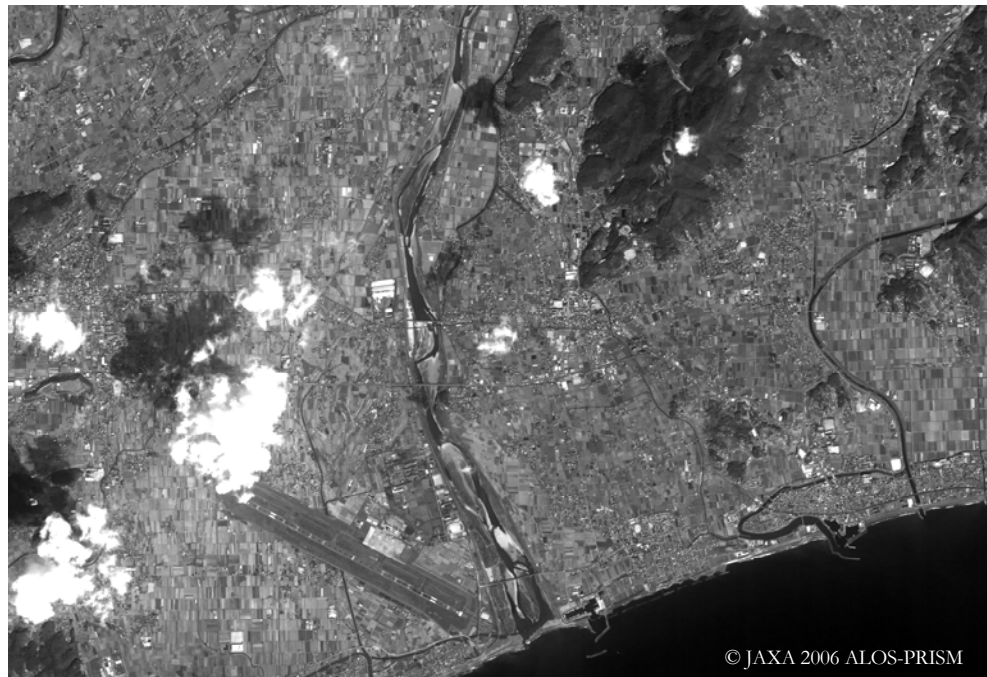
PRISM (Panchromatic Remote-sensing Instrument for Stereo Mapping sensor)	Level 1B2 Geo-reference image
	 <p>Concept of scene related information for PRISM 1B2 Geo-reference image</p>
PRISM characteristics	
Number of bands Wavelength Number of optics Base-to-Height ratio Spatial resolution Swath width S/N MTF Number of detectors Pointing angle Bit length	1 (panchromatic) 0.52 to 0.77 micrometers 3 (nadir; forward; backward) 1.0 (between forward and backward view) 2.5m (at nadir) 70km(nadir only)/35km(triplet mode) >70 >0.2 28000 / band (swath width 70km) 14000 / band (swath width 35km) -1.5 to +1.5 degrees (triplet mode, cross-track direction) 8 bits

Observation modes	
Mode 1	Triplet observation mode using forward, nadir, and backward views (swath width is 35 km)
Mode 2	nadir (70 km) + backward (35 km)
Mode 3	nadir (70 km)
Mode 4	nadir (35 km) + forward (35 km)
Mode 5	nadir (35 km) + backward (35 km)
Mode 6	forward (35 km) + backward (35 km)
Mode 7	nadir (35 km)
Mode 8	forward (35 km)
Mode 9	backward (35 km)

The selected target ALOS-PRISM (*figure 1.8*) is acquired in 2006 Sept 30. JAXA provided the dataset as primary investigator of research to Takagi laboratory of Kochi University of Technology, Japan. The important specifications of selected target image are presented in the table (*table 1.3*). Level 1B2R product is used.

Table 1.3: Specifications of used target image

	ALOS-PRISM
Acquired date:	2006-Sep-30
Sun elevation angle	51.05°
Sun azimuth angle	155.60°
Spatial resolution:	2.5 meters
Spectral range:	520–770 nm (panchromatic)



© JAXA 2006 ALOS-PRISM

Figure 1.8: High resolution target satellite image (ALOS-PRISM)

METHODOLOGY

2.1 Main Methodology

Image registration is a process of coordinating maps and images to the same location on the ground. When some remote sensing communities are use conventional registration methods (*figure 2.1*) using GCP, some are use Rational Polynomial Coefficients (RPC), which is simulated by satellite orbits and altitude parameters, for very precise image registration (*figure 2.2*). However both of methods are using widely, precise GCP is needed for accurate registration. Thus, GCP are collected by virtual interpretation from topographic maps or registered images for low resolution satellite image registration. In the case of medium resolution satellite image, GCP are collected by geodetic surveying device measurement. While high resolution satellite images are available, the requirements of accuracy for image analysis become increase in precision.

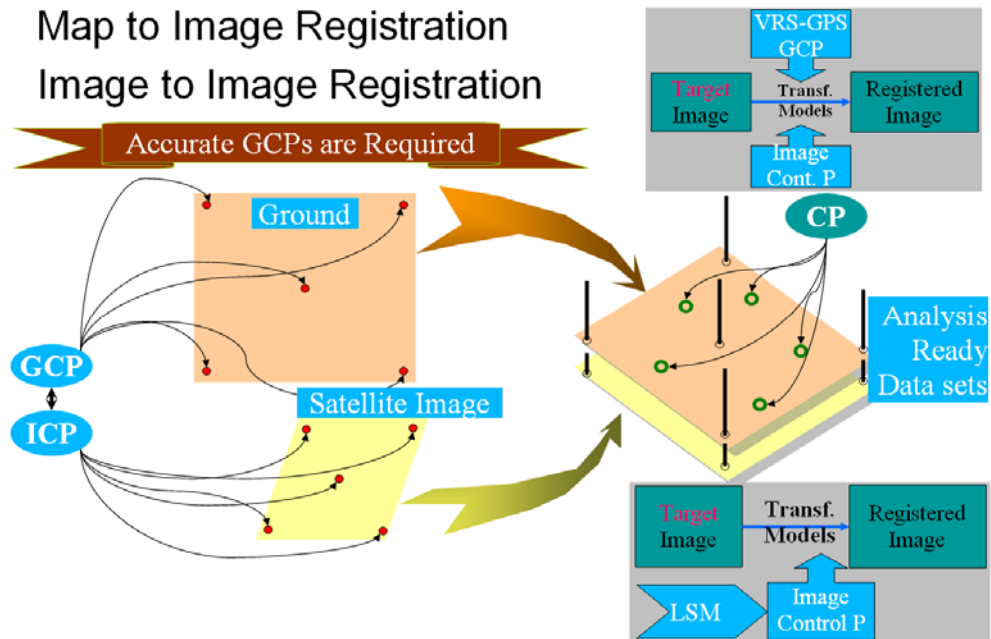


Figure 2.1: Conventional registration method

The common conventional registration methods (parallel transformation model and perspective transformation) in 3 dimensional are observed to suggest the potential of transformation model. Specifically, the study observed 3D affine as

parallel transformation model whilst 3D projective transformation model and separated 3D transformation model are observed as perspective transformation models.

However, some researchers are using complicated replacement camera model (figure 2.2); the model is out of the study purposes.

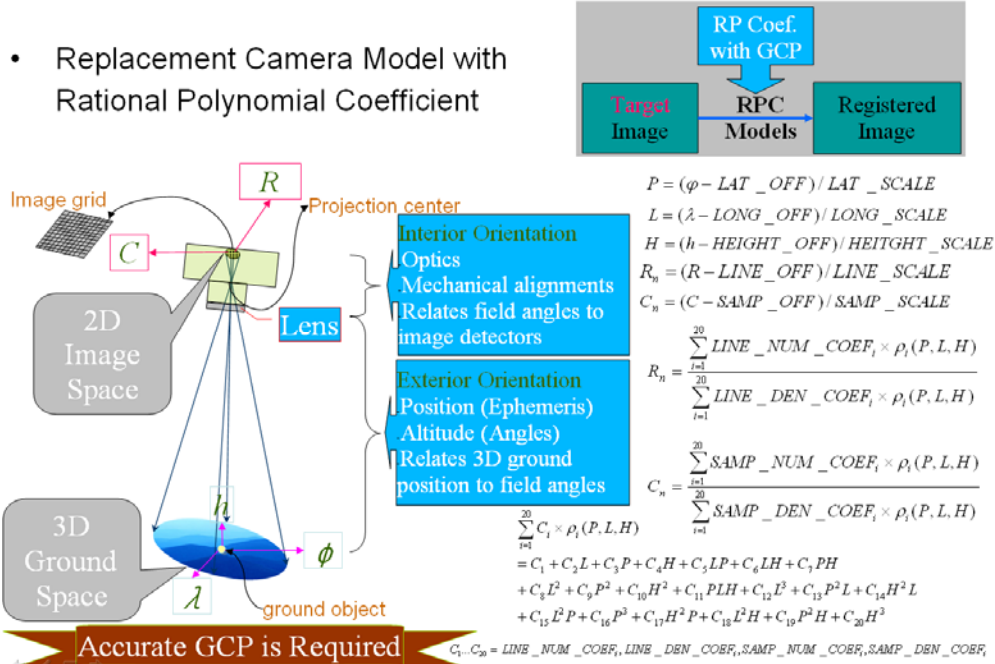


Figure 2.2: Image registration with RPC

The main methodology of the research includes two parts: reference data simulation for the target image and GCP selection by Least Squares Matching method (figure 2.3). DSM data is used as main input to the system to generate shade and shadow for both of the satellite images. Firstly, shade and shadow for the reference image are simulated from DSM by giving the sun illumination parameters of its sensor. The simulated data is used to eliminate shade and shadow from the reference image. Then, reflectance value is generated from the average of red ratio and green ratio of the reference multi-spectral image after shade and shadow effects are removed from the image ratios. Again, shade and shadow data are simulated with the sun illumination parameters of the targeted image and it is integrated with generated reflectance value as seasonal vegetation pattern contribution. Finally, simulated template image is generated for image matching algorithm.

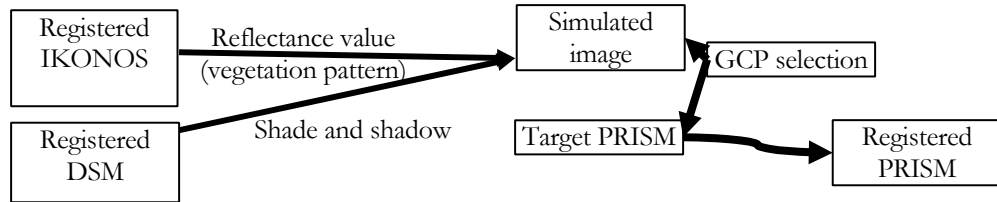


Figure 2.3: Main methodology

Above figure showing image-to-image registration using two high resolution satellite images and one DSM dataset. As a default requirement of the three-dimensional transformation model, we assumed high value (z) from DSM data and (x, y) values from the reference image in image registration. The target image is presented as (u, v) . Then, image registration can present with mathematical form as in equations 2.1.

$$\begin{aligned}
 u &= f(x, y, z) \\
 v &= g(x, y, z)
 \end{aligned}
 \tag{2.1}$$

This new approach for image registration is presented in more detail (figure 2.4). The core part of the research is ICP generation from DSM and reference satellite image. GCPs are collected by LSM image matching between ICP and target image. Then, image registration is done using selected conjugated points which are selected from results of image matching.

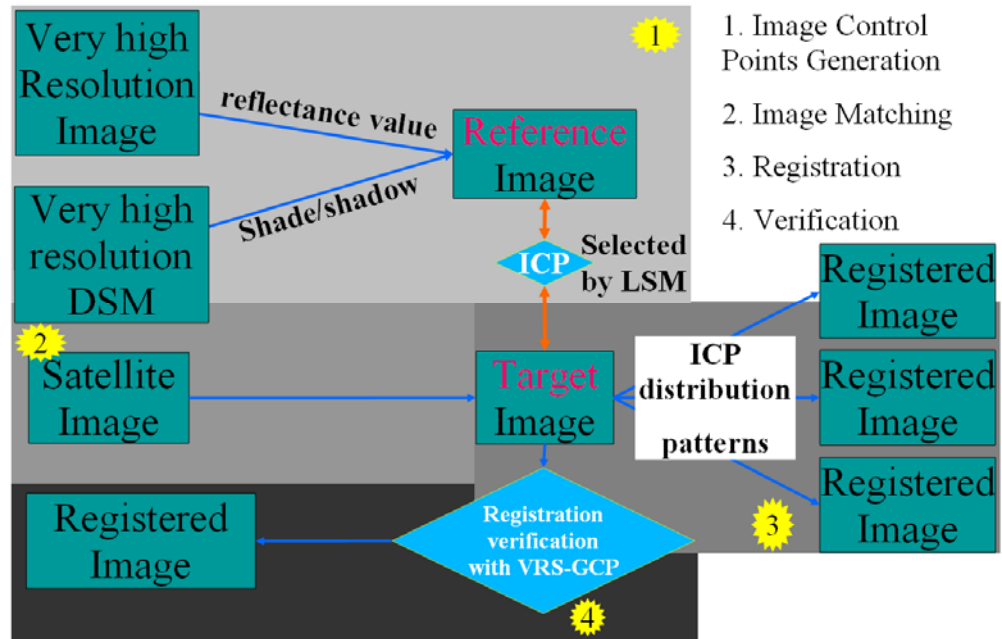


Figure 2.4: Detail methodology

The above methodology (figure 2.3 and 2.4) of the study could be break down to two main parts. They are reference data generation and data simulation for target image (figure 2.5).

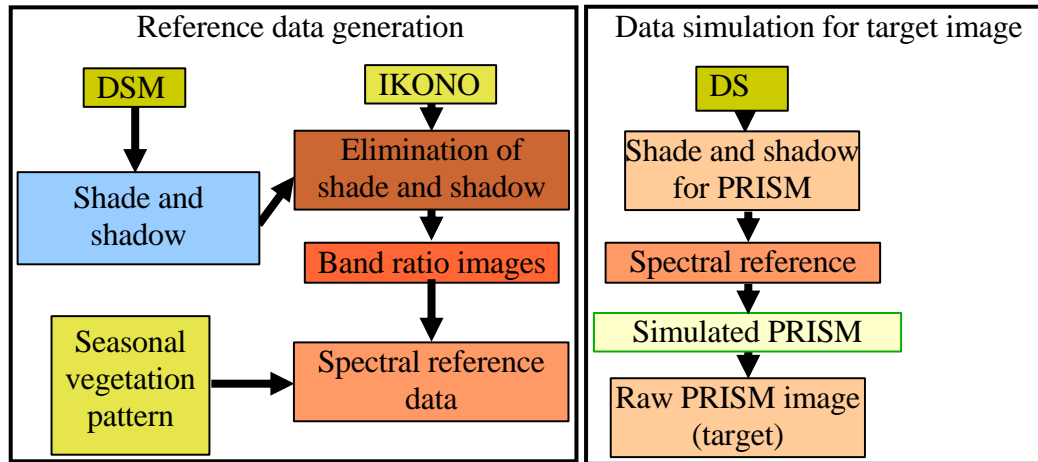


Figure 2.5: Breakdown methodology

A DSM dataset was used as main input to the algorithm to generate shade and shadow for both satellite images. In reference data generation, shade and shadow of the reference IKONOS image was calculated from DSM by inputting sun illumination parameters. The calculated data was used to eliminate shade and shadow affects from the reference image. Then, band ratio images were calculated from the reference multispectral image. In simulation of the target image, another generated shade and shadow dataset was added; this was calculated from the sun illumination parameters of target image. Finally, the artificial target image was simulated.

TRANSFORMATION MODELS

Transformation is typically used in satellite Remote Sensing, Geographic Information System, survey and mapping when converting separate datasets to a common coordinate system. When we want to use satellite images as geographic information source, the images are needed to transform prior before using it. Transformation is a process to change image's two dimension space to the ground coordinates by translating the position, scaling the size, rotation the orientation, shearing the shape. The satellite images cannot overlay each other without transformed it. Many generalized mathematical transformation models are applying in image space u, v to ground coordinate x, y, z . When we recalled equation 2.1 from chapter 2

$$\begin{aligned} u &= f(x, y, z) \\ v &= g(x, y, z) \end{aligned} \tag{3.1}$$

There are many transformation models exist in survey and mapping such as polynomial, conformal, affine, 3D affine, projective, 3D projective, separated 3D projective, Rational Polynomial Coefficients and so on. The three popular models are investigated in this study.

3.1 Parallel transformation Model

Affine transformation is one of parallel transformation models. In definition, an affine space is either the empty set, or a triple $(E, \vec{E}, +)$ consisting of a nonempty set E (of points), a vector space \vec{E} (of translations, or free vectors), and an action $+: E \times \vec{E} \rightarrow E$, satisfying with the conditions of: $a + \vec{0} = a$, for every $a \in E$; $(a + \vec{u}) + \vec{v} = a + (\vec{u} + \vec{v})$, for every $a \in E$, and every $\vec{u}, \vec{v} \in \vec{E}$; for any two points $a, b \in E$, there is a unique $\vec{u} \in \vec{E}$ such that $a + \vec{u} = b$. The unique vector

$\vec{u} \in \vec{E}$ such that $a + \vec{u} = b$ is denoted as \vec{ab} , or sometimes as $b-a$. Thus $b = a + \vec{ab}$. The dimension of the affine space is the dimension $\dim(\vec{E})$ of the vector space \vec{E} . For simplicity, it is denoted as $\dim(E)$.

Considering two dimension space, when the image $f(u, v)$ coordinates are needed to project on to the ground $g(x, y, z)$ coordinates we can use two dimensional affine transformation. The transformation involves two different scale factors, one in x direction and the other in the y direction; two translations in the origin; and a rotation about the origin plus a small non-orthogonality correction between the x and y axes.

As scaling, it is necessary to multiply the origin image coordinates x, y with a scale factor S to maintain line lengths as defined by the x, y coordinate system equal to their lengths in the X, Y system. Thus the scaled coordinates x' and y' are $x' = Sx$ and $y' = Sy$.

In Rotation, the XY coordinate system has been superimposed on the scaled x', y' system. The rotation angle, θ , is shown the rotation between the axes. To analyze the effects of this rotation, an X', Y' system was constructed parallel to the X, Y system such that its origin is common with that of the x', y' system. Expressions that give the (X', Y') rotated coordinates for any point in terms of its x', y' coordinates are $X' = x' \cos\theta - y' \sin\theta$ and $Y' = x' \sin\theta + y' \cos\theta$.

For translation, to finally arrive at X, Y coordinates for a point, it is necessary to translate the origin of the X', Y' system to the origin of the X, Y system. The translation factors can be added as $X = X' + T_x$ and $Y = Y' + T_y$. Combining these parameters, two dimensional affine mathematical model can be:

$$\begin{aligned} a_1x + a_2y + a_3 &= X + V_x \\ a_4x + a_5y + a_6 &= Y + V_y \end{aligned} \tag{3.2}$$

where:

X, Y are image coordinates

V_x, V_y are small non-orthogonally correction

x, y are two dimensional ground coordinates

a_1, a_2, a_3, a_4, a_5 and a_6 are rotation, scaling, translation parameters

The equation (equation 3.2) is linear and it can be solved uniquely when three control points exist. The three control points yield six equations involving six unknowns. Least squares solution can be obtained when control points are

available more than three. For example, consider four common points, the equation system will be

$$\begin{aligned}
a_1x_1 + a_2y_1 + a_3 &= X_1 + V_{x1} \\
a_4x_1 + a_5y_1 + a_6 &= Y_1 + V_{y1} \\
a_1x_2 + a_2y_2 + a_3 &= X_2 + V_{x2} \\
a_4x_2 + a_5y_2 + a_6 &= Y_2 + V_{y2} \\
a_1x_3 + a_2y_3 + a_3 &= X_3 + V_{x3} \\
a_4x_3 + a_5y_3 + a_6 &= Y_3 + V_{y3} \\
a_1x_4 + a_2y_4 + a_3 &= X_4 + V_{x4} \\
a_4x_4 + a_5y_4 + a_6 &= Y_4 + V_{y4}
\end{aligned} \tag{3.3}$$

In matrix notation, the *equation 3.3* are expressed as $AX=L+V$, where

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ x_3 & y_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_3 & y_3 & 1 \\ x_4 & y_4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_4 & y_4 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ X_2 \\ Y_2 \\ X_3 \\ Y_3 \\ X_4 \\ Y_4 \end{bmatrix} + \begin{bmatrix} V_{x1} \\ V_{y1} \\ V_{x2} \\ V_{y2} \\ V_{x3} \\ V_{y3} \\ V_{x4} \\ V_{y4} \end{bmatrix} \tag{3.4}$$

3.1.1 3D Affine Transformation Model

When 3D space is considered 3D affine transformation involved 8 parameters, three rotations, three translations, and two different scale factors for x, y directions.

$$\begin{aligned}
a_1x + a_2y + a_3z + a_4 &= X + V_x \\
a_5x + a_6y + a_7z + a_8 &= Y + V_y
\end{aligned} \tag{3.5}$$

where:

X, Y are image coordinates

V_x, V_y are small non-orthogonally correction

x, y, z are three dimensional ground coordinates

$a_1, a_2, a_3, a_4, a_5, a_6, a_7$ and a_8 are rotation, scaling, translation parameters

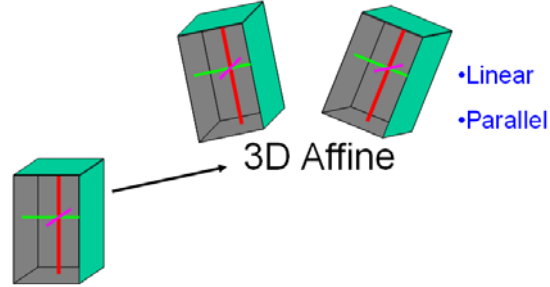


Figure 3.1: 3D affine transformation

3.2 Perspective transformation Model

In projective transformation, two dimension space of the image u, v coordinate are projected on to the ground x, y, z coordinate using the equation (3.6).

$$\frac{a_1x + a_2y + a_3}{a_7x + a_8y + 1} = X + V_x$$

$$\frac{a_4x + a_5y + a_6}{a_7x + a_8y + 1} = Y + V_y$$
(3.6)

where:

X, Y are image coordinates

V_x, V_y are small non-orthogonally correction

x, y are two dimensional ground coordinates

$a_1, a_2, a_3, a_4, a_5, a_6, a_7$ and a_8 are 8 parameters of rotation, scaling, translation

The equation (equation 3.6) is similar with affine transformation when two parameters a_7 and a_8 are equal to zero. Based on eight parameters, the 2 dimensional projective transformation required a minimum of four control points. The linearized form of this equation is

$$\begin{aligned}
& \begin{bmatrix} \left(\frac{\partial X}{\partial a_1}\right)_0 & \left(\frac{\partial X}{\partial a_2}\right)_0 & \left(\frac{\partial X}{\partial a_3}\right)_0 & 0 & 0 & 0 & \left(\frac{\partial X}{\partial a_7}\right)_0 & \left(\frac{\partial X}{\partial a_8}\right)_0 \\ 0 & 0 & 0 & \left(\frac{\partial X}{\partial a_4}\right)_0 & \left(\frac{\partial X}{\partial a_5}\right)_0 & \left(\frac{\partial X}{\partial a_6}\right)_0 & \left(\frac{\partial X}{\partial a_7}\right)_0 & \left(\frac{\partial X}{\partial a_8}\right)_0 \end{bmatrix} \begin{bmatrix} da_1 \\ da_2 \\ da_3 \\ da_4 \\ da_5 \\ da_6 \\ da_7 \\ da_8 \end{bmatrix} \\
& = \begin{bmatrix} X - X_0 \\ Y - Y_0 \end{bmatrix} \tag{3.7}
\end{aligned}$$

where

$$\begin{aligned}
\frac{\partial X}{\partial a_1} &= \frac{x}{a_7x + a_8 + 1} \quad \frac{\partial X}{\partial a_2} = \frac{y}{a_7x + a_8 + 1} \quad \frac{\partial X}{\partial a_3} = \frac{1}{a_7x + a_8 + 1} \\
\frac{\partial Y}{\partial a_4} &= \frac{x}{a_7x + a_8 + 1} \quad \frac{\partial Y}{\partial a_5} = \frac{y}{a_7x + a_8 + 1} \quad \frac{\partial Y}{\partial a_6} = \frac{1}{a_7x + a_8 + 1} \\
\frac{\partial X}{\partial a_7} &= \frac{a_1x + a_2y + a_3}{(a_7x + a_8 + 1)^2} x \quad \frac{\partial X}{\partial a_8} = \frac{a_1x + a_2y + a_3}{(a_7x + a_8 + 1)^2} y \\
\frac{\partial Y}{\partial a_7} &= \frac{a_4x + a_5y + a_8}{(a_7x + a_8 + 1)^2} x \quad \frac{\partial Y}{\partial a_8} = \frac{a_4x + a_5y + a_6}{(a_7x + a_8 + 1)^2} y
\end{aligned}$$

3.2.1 3D Projective Transformation Model

3D projective transformation involved 11 parameters of rotation, translation, and scaling factors. The model can be expressed as following mathematic form

$$\begin{aligned}
\frac{a_1x + a_2y + a_3z + a_4}{a_9x + a_{10}y + a_{11}z + 1} &= X + V_x \\
\frac{a_5x + a_6y + a_7z + a_8}{a_9x + a_{10}y + a_{11}z + 1} &= Y + V_y
\end{aligned} \tag{3.8}$$

where:

X, Y are image coordinates

V_x, V_y are small non-orthogonally correction

x, y, z are three dimensional ground coordinates

$a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}$ and a_{11} are 11 parameters of rotation, scaling, translation

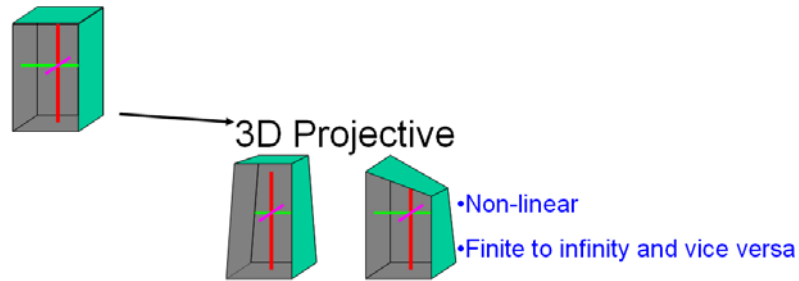


Figure 3.2: 3D projective transformation

3.2.2 Separated 3D Projective Transformation Model

When we considered about separation of denominator of 3D projective transformation, it can be called separated 3D transformation (Kojima and Takagi, 2008), involved 14 parameters of rotation, translation, and scaling factors. The model can be expressed as following mathematic form

$$\frac{a_1x + a_2y + a_3z + a_4}{a_9x + a_{10}y + a_{11}z + 1} = X + V_x$$

$$\frac{a_5x + a_6y + a_7z + a_8}{a_{12}x + a_{13}y + a_{14}z + 1} = Y + V_y$$
(3.9)

where:

X, Y are image coordinates

V_x, V_y are small non-orthogonally correction

x, y, z are three dimensional ground coordinates

$a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}, a_{12}, a_{13}$ and a_{14} are 14 parameters of rotation, scaling, translation

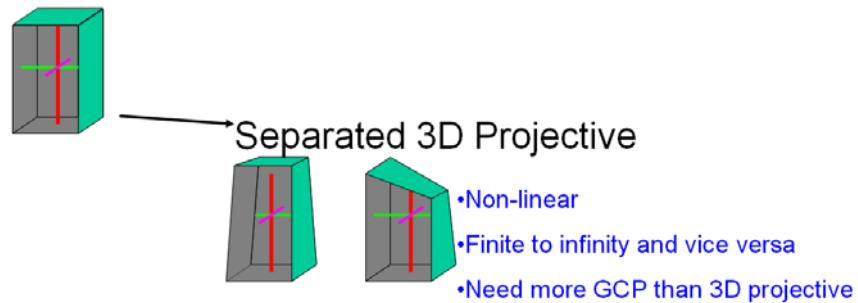


Figure 3.3: Separated 3D projective transformation

3.3 Investigation of geometric error propagated by transformation models

The study investigated the geometric errors on optical satellite images using transformation models: 3D projective model, separated 3D projective model and 3D affine model with Ground Control Points from inhouse GCP database. The mathematic models of transformation models are presented in chapter two of the thesis. The information about in-house GCP database can be found in chapter four. The GCP are in centimeter level accuracy and which are collected using VRS-GPS.

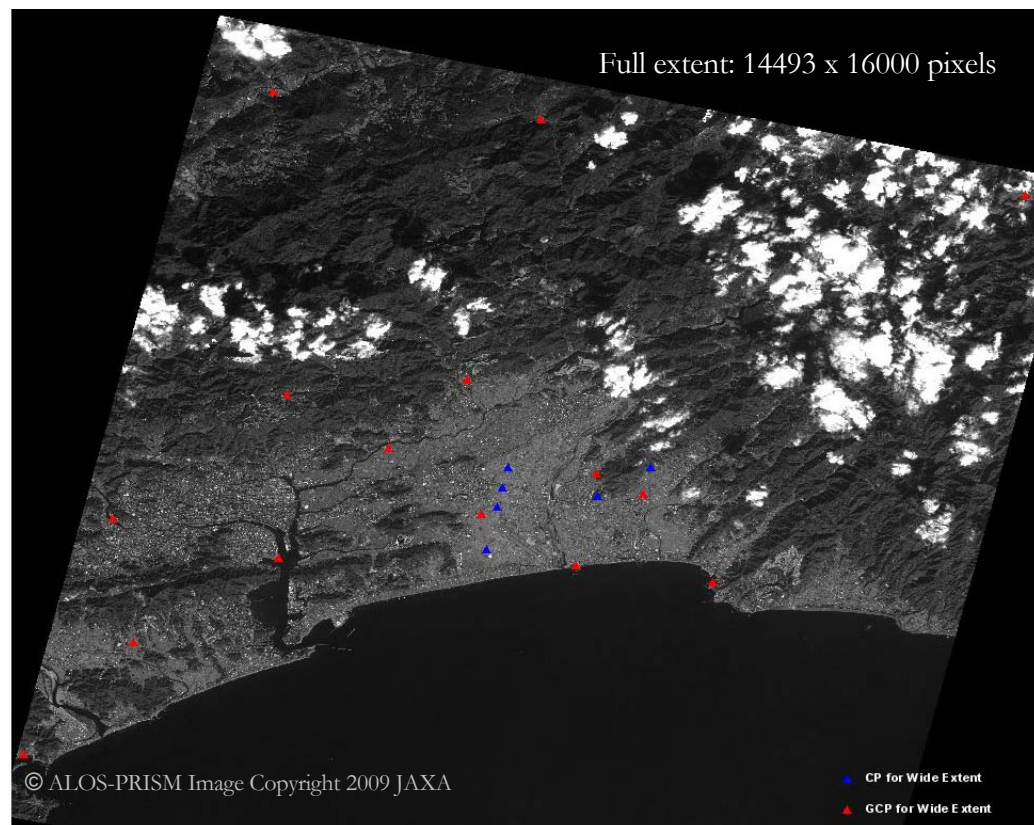


Figure 3.4: Distribution of GCP and CP on PRISM image

Two full scenes (approximately 35 x 35 km) of nadir and backward views are observed using 15 VRS-GPS's GCP and 5 Check Points (table 3.1). A transformed image with GCP and CP distribution is presented (figure 3.4).

Table 3.1 Used GCP and CP values

G C P /CP	Japanese Datum 2000 region 4			PRISM N a d i r		PRISM Backward	
	X (meter)	Y (meter)	Z (meter)	image u (pixel)	image v (pixel)	image u (pixel)	image v (pixel)
GCP 1	23621.023	58632.557	2.018	10377.0	8983.0	9955.0	9737.0
GCP 2	17965.617	59460.534	1.882	8118.0	9165.0	7911.0	9915.0
GCP 3	14048.085	62019.519	9.132	6379.5	8523.5	6335.5	9271.5
GCP 4	5644.525	59813.746	2.615	3331.0	10116.0	3578.0	10862.0
GCP 5	-383.864	55641.602	4.294	1373.5	12251.5	1804.0	12995.0
GCP 6	-4907.733	50097.369	2.691	119.0	14785.0	668.0	15524.0
GCP 7	-1216.034	61777.169	14.051	513.0	9965.0	1025.0	10708.0
GCP 8	5993.109	67888.338	293.154	2752.5	6977.5	3052.0	7667.0
GCP 9	5384.470	82867.230	250.500	1199.5	1271.5	1646.5	1972.5
GCP 10	13453.964	68622.731	30.223	5568.0	6035.0	5600.0	6779.0
GCP 11	10190.658	65276.659	4.727	4605.5	7612.5	4728.0	8360.0
GCP 12	20720.561	63015.812	8.912	8871.5	7554.5	8590.0	8305.0
GCP 13	36471.362	77845.163	252.523	13651.5	463.5	12917.0	1191.0
GCP 14	16469.955	81512.922	298.162	5597.0	810.0	5627.0	1507.0
GCP 15	18802.87	64035.018	68.269	8040.0	7329.0	7840.0	8075.0
CP 1	18847.463	62913.757	113.851	8152.0	7757.0	7941.0	8494.0
CP 2	14255.082	60255.588	41.416	6618.0	9185.5	6553.0	9932.0
CP 3	14699.241	62341.791	48.070	6602.5	8344.0	6539.0	9092.0
CP 4	14919.021	63330.773	51.330	6600.0	7945.0	6536.0	8693.0
CP 5	15125.409	64325.318	52.988	6594.0	7546.0	6530.0	8293.0

Five patterns of distribution and their numbers of GCP (7 GCP, 9 GCP, 11 GCP, 13 GCP, 15 GCP) are considered for two types of viewing images (nadir and backward image of ALOS-PRISM) in this study. Takagi laboratory's separated 3D projective model (Kojima et al., 2008) is used as primitive program for accuracy evaluation. Two new programs (3D affine and 3D projective) are regenerated based on the separated model program. The results of control points are presented (table 3.2 and figure 3.5). The result showed that viewing directions are not influence upon accuracy of image. Both of views give similar result on all models, distributions and numbers of GCP.

Table 3.2: Results of distribution patterns and number of GCP (Unit: Pixel)

ALOS PRISM Nadir full image				
Models	RMS in GCP		RMS in CP	
15 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.61	1.70	3.17	1.44
3D projective	1.80	2.00	3.46	1.53
3D separated	0.79	1.46	1.62	1.60

Models	RMS in GCP		RMS in CP	
13 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.63	1.33	3.32	1.27
3D projective	1.92	1.57	3.82	1.30
3D separated	0.61	0.80	1.60	1.60

Models	RMS in GCP		RMS in CP	
11 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.82	1.28	3.37	1.27
3D projective	2.02	1.47	3.97	1.44
3D separated	0.42	0.74	1.84	1.43

Models	RMS in GCP		RMS in CP	
9 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.72	1.17	3.53	1.27
3D projective	2.04	1.33	4.19	1.50
3D separated	0.27	0.48	1.72	1.24

Models	RMS in GCP		RMS in CP	
7 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	3.02	0.97	3.84	1.50
3D projective	2.18	1.27	4.08	2.00
3D separated	1.00E-09	1.40E-08	1.91	1.13

ALOS PRISM Backward full image				
Models	RMS in GCP		RMS in CP	
15 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.17	4.16	2.22	5.15
3D projective	2.44	3.41	3.83	5.55
3D separated	0.87	2.77	2.05	6.93

Models	RMS in GCP		RMS in CP	
13 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.11	3.80	2.33	4.63
3D projective	2.52	2.46	4.45	5.04
3D separated	0.82	1.56	1.84	6.92

Models	RMS in GCP		RMS in CP	
11 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.29	3.97	2.28	5.02
3D projective	2.70	2.47	4.63	5.43
3D separated	0.58	0.67	1.88	8.45

Models	RMS in GCP		RMS in CP	
9 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.14	3.90	2.42	5.04
3D projective	2.51	2.18	4.94	5.57
3D separated	0.44	0.12	1.82	8.08

Models	RMS in GCP		RMS in CP	
7 GCP & 5 CP	ΔU	ΔV	ΔU	ΔV
3D affine	2.26	4.05	2.68	4.10
3D projective	2.41	2.06	5.43	4.72
3D separated	0.00E+00	1.50E-08	1.68	7.95

Parallel transformation (3D affine model) produced good accuracy in overall region of image when perspective transformations (3D projective and separated 3D projective) gave very good accuracy around GCP.

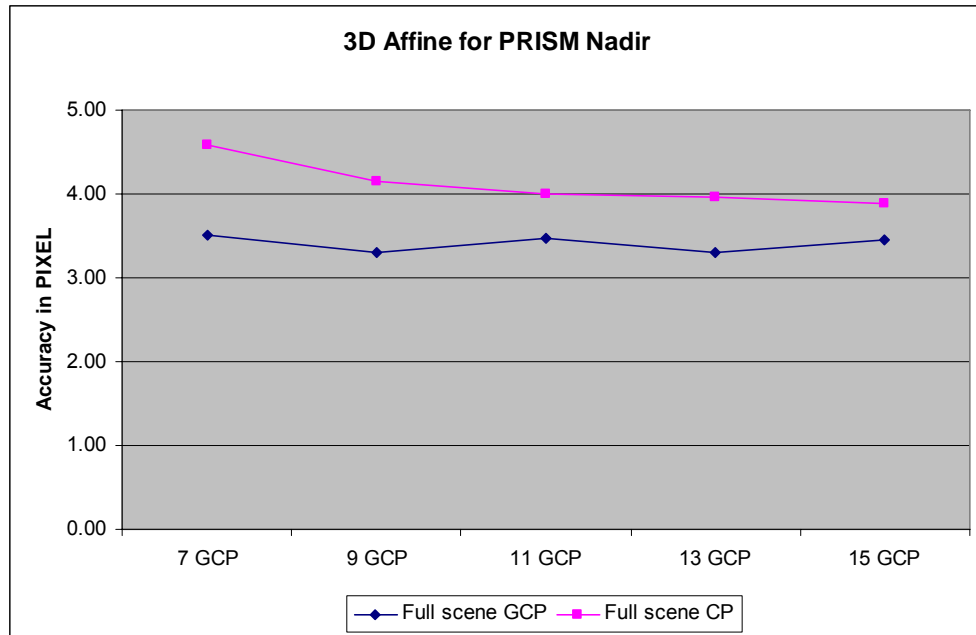


Figure 3.5(a): Accuracy of distribution of GCP and number of GCP on PRISM images

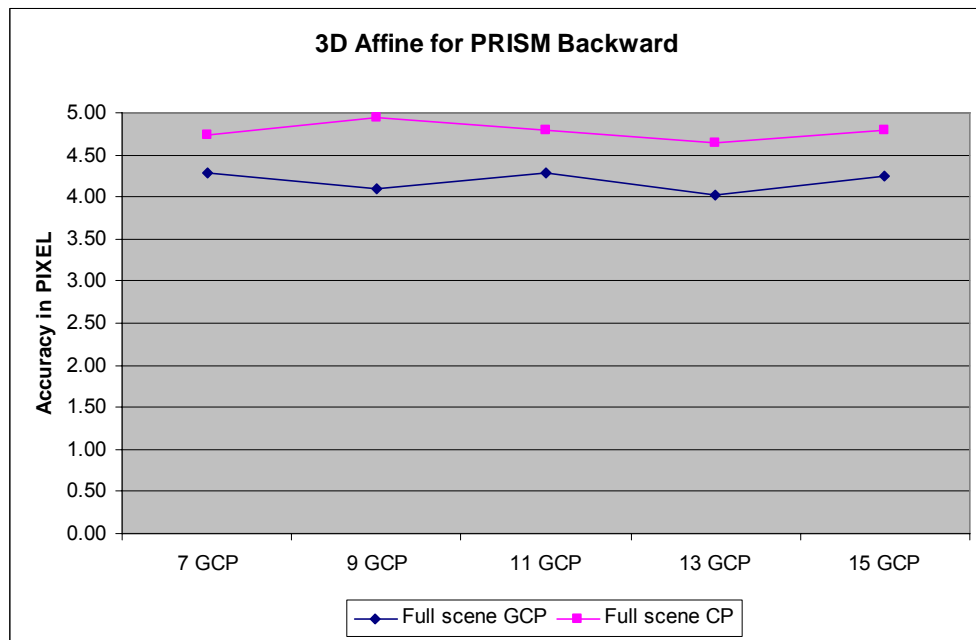


Figure 3.5 (b): Accuracy of distribution of GCP and number of GCP on PRISM images

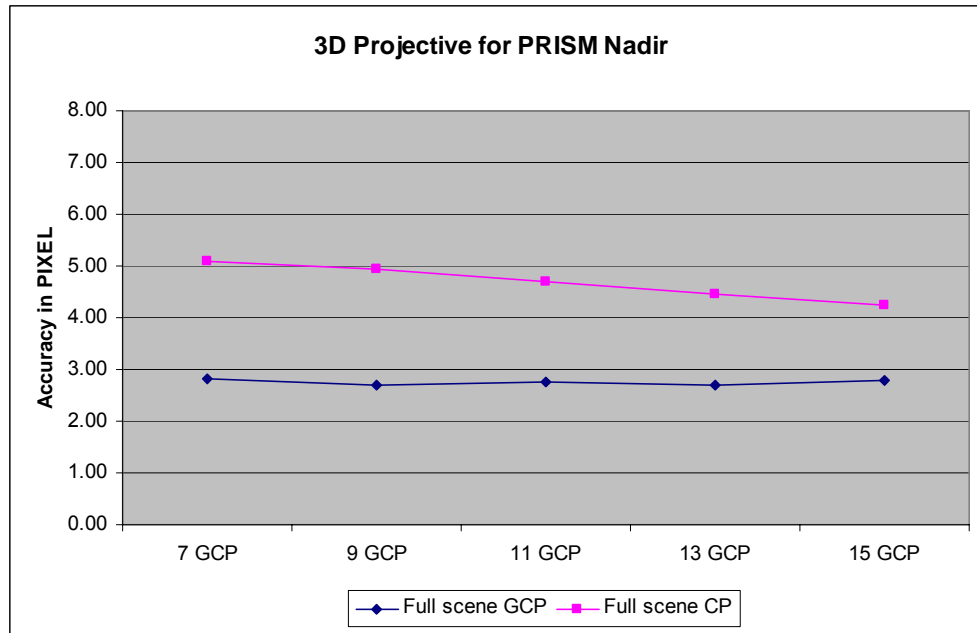


Figure 3.5 (c): Accuracy of distribution of GCP and number of GCP on PRISM images

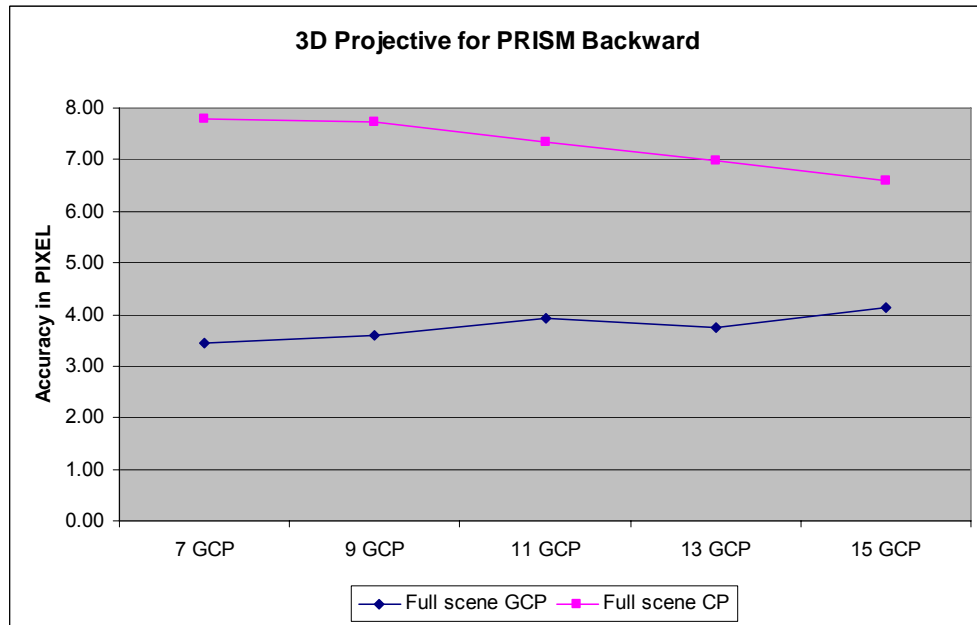


Figure 3.5 (d): Accuracy of distribution of GCP and number of GCP on PRISM images

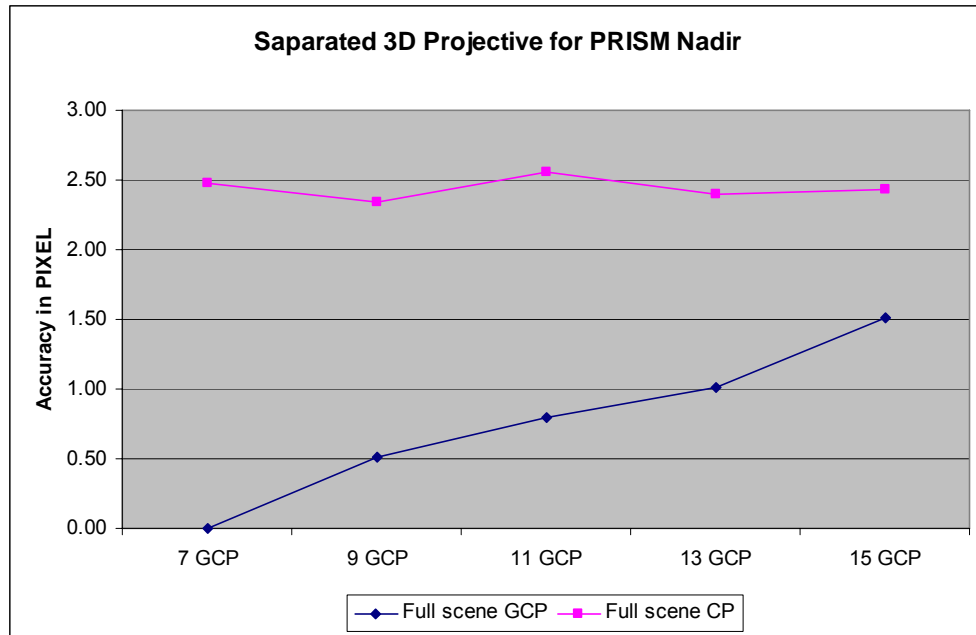


Figure 3.5 (e): Accuracy of distribution of GCP and number of GCP on PRISM images

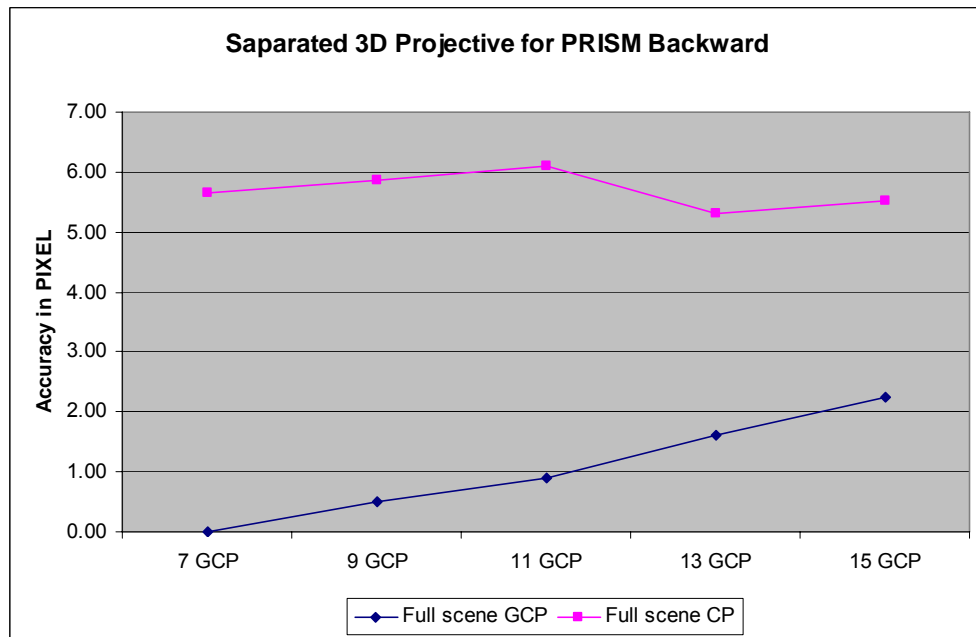


Figure 3.5 (f): Accuracy of distribution of GCP and number of GCP on PRISM images

3.4 Result of Transformation Models

Three transformation models are observed in the study. At the same time that 3D affine is observed as parallel model, 3D projective model and separated 3D projective model are observed as perspective transformation models. In this section, conjugated GCP are interpreted visual from subset (4000 x 3000 pixels) PRISM nadir image and IKONOS image. The transformation results are presented in the table (table 3.3).

Table 3.3: Result of transformation models using visual interpreted GCP (Unit in pixel)

	RMSE (GCP)		RMSE (CP)	
	u	v	u	v
10 GCP & 3 CP				
3D Affine	1.63	1.88	1.20	1.05
3D Projective	1.47	1.25	1.80	1.04
Separated 3D	1.34	1.25	1.44	1.19

In this transformation models, the study used 10 well distributed GCP which are collected visually from conjugated images. On the other hand, the check points are used very precise VRS-GPS control points.

The result shows less than 5 meters in all models (figure 3.6). Moreover, the parallel transformation model produced global accuracy improvement while the perspective models are giving more accurate response around control points.

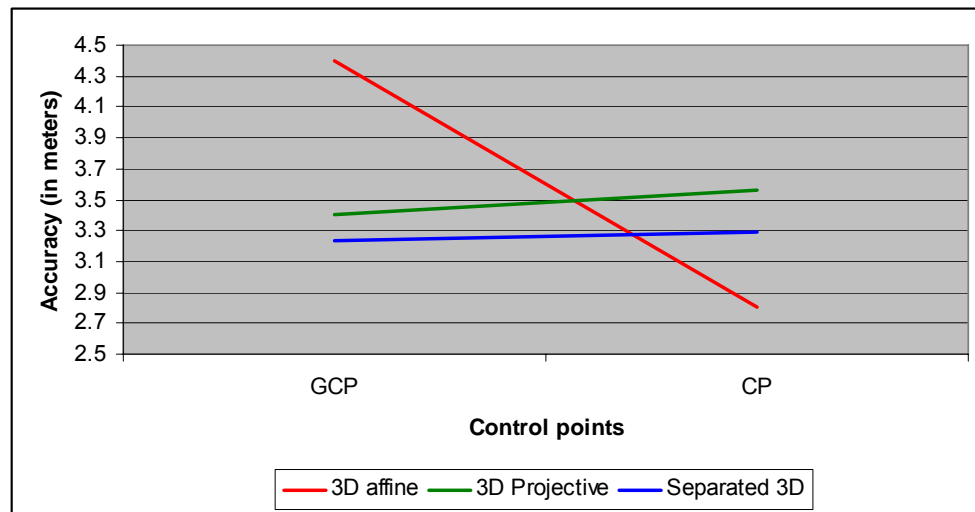


Figure 3.6: The result of visual interpreted GCP

SPATIAL DISTRIBUTION OF CONTROL POINTS AND NUMBERS OF GCP

4.1 Ground Control Point

GCP is a point with latitude, longitude, height (X, Y, Z) values of the respected ground location and which are use in satellite image registration (*figure 4.1*). Two type of control points are use in image registration: GCP and Check Points.

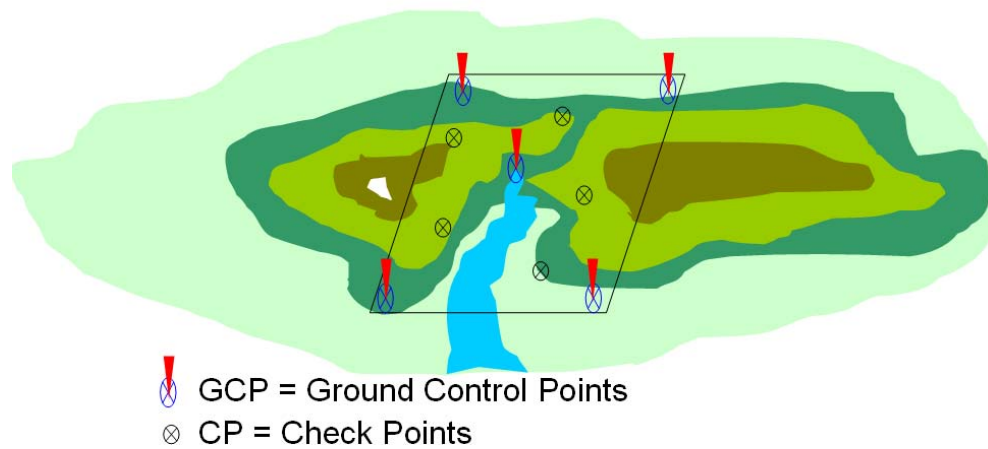


Figure 4.1: GCP and CP

GCP are required by satellite image registration. The 3D projective and 3D affine transformation required 15 GCP for IKONOS to reach the level of 1 pixel accuracy (*Jacobsen, 2006*). *Guoqing Zhou and Ron Li (2000)* showed that the average (x, y, z) error can reach 11 to 12 m without GCP but accuracy largely improved with GCP. Their result matches with results from MOMS-02 (*Fraser and Shao, 1996; Ebner et al., 1991*). *Toutin, Th. (1995, 2003)* presented in his 3D parametric model, the residuals don't reflect the modeling accuracy but rather the errors in the input data when there are more GCP than the minimum required; in his bundle Adjustment model: based on accurate GCP, the accuracy is 1/3 pixel for medium resolution images and 1-2 pixel for high resolution images. *Toshiaki Hashimoto (2006)* recommended GCP are necessary for precise geometric correction.

4.2 Development of GCP database

Precise Ground Control Points are an important requirement in satellite image registration. However, there are many precise geodetic survey points in Japan; the points are difficult to identify in satellite images. Therefore, we developed a web-base GCP database (*figure 4.2*) to provide users with free of charges. Nowadays, there are many advantages by advance survey technology; one of advancements in geoinformatics is development of VRS-GPS (*figure 4.3*) technology. This kinematic technology uses telecommunication network as a Virtual Reference Station for the Global Positioning System. VRS-GPS can use in collection of GCP in short time (about 2 minutes for one GCP) comparing with Static Differential GPS (about 60 minutes). We used a VRS-GPS system to collected GCP in this works.

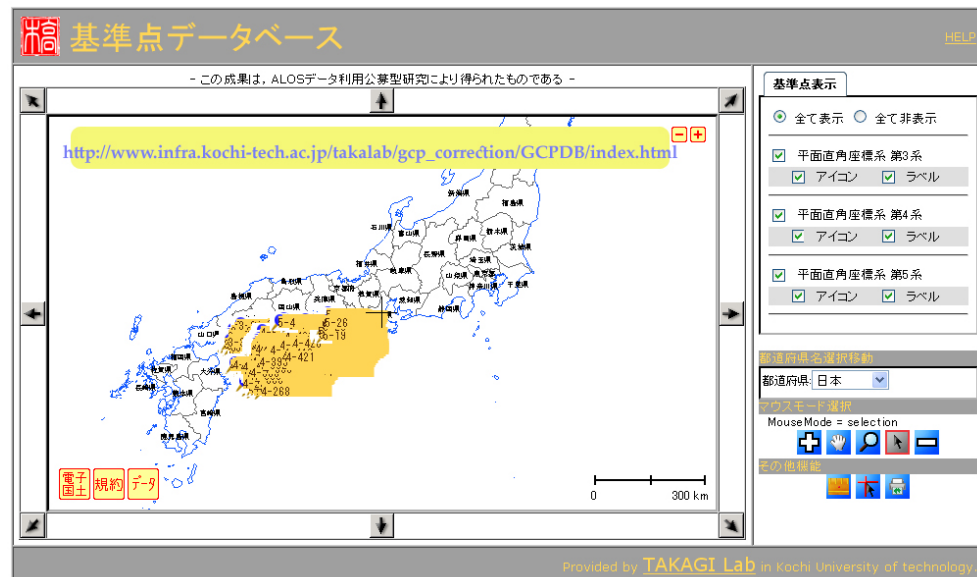


Figure 4.2: GCP database of Takagi Laboratory

We chose points which are visible in PRISM and AVNIR-2 images before we survey on the ground location. Thus, the GCP database is standardized with ALOS Satellite optical imageries. Whatever it is ALOS compliance; these GCPs can be used for many other satellite images. Recently, over 500 GCPs are observed for the forth island (SHIKOKU) of JAPAN. However we collected GCP mainly with VRS-GPS, sometime we had used static GPS survey when VRS network fails to receive telecommunication signals. Our GCP database covered the whole area of Shikoku (18,803 km²) island and the island is a hilly remote area of Japan.

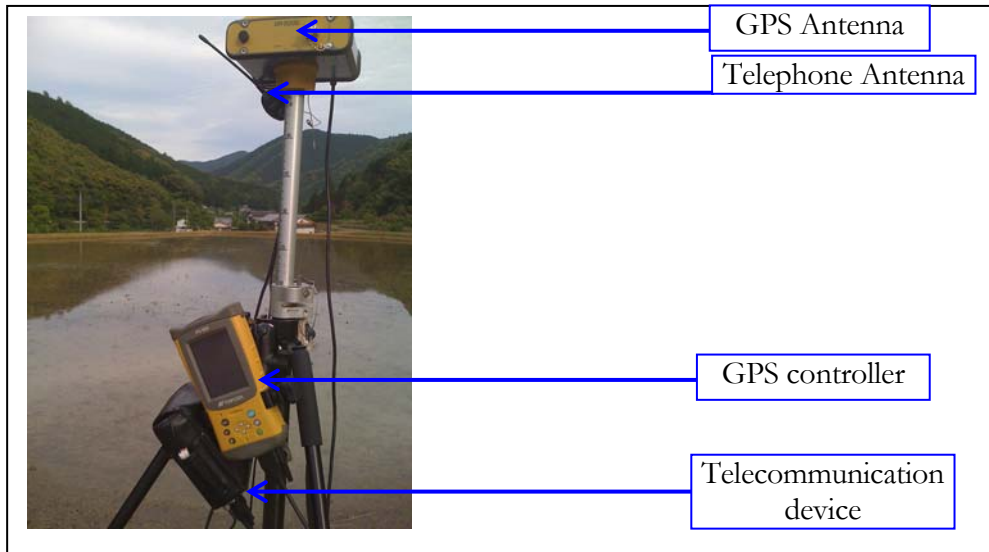


Figure 4.3: VRS-GPS

Collected GCPs are distributed widely (figure 4.4). A GCP page (figure 4.5) is composed of with three main parts: GCP metadata (X, Y, Z, Lat, Long, observation date, feature name, etc...), photographs (taken from difference directions) and location on satellite images (PRISM or/and AVNIR-2).

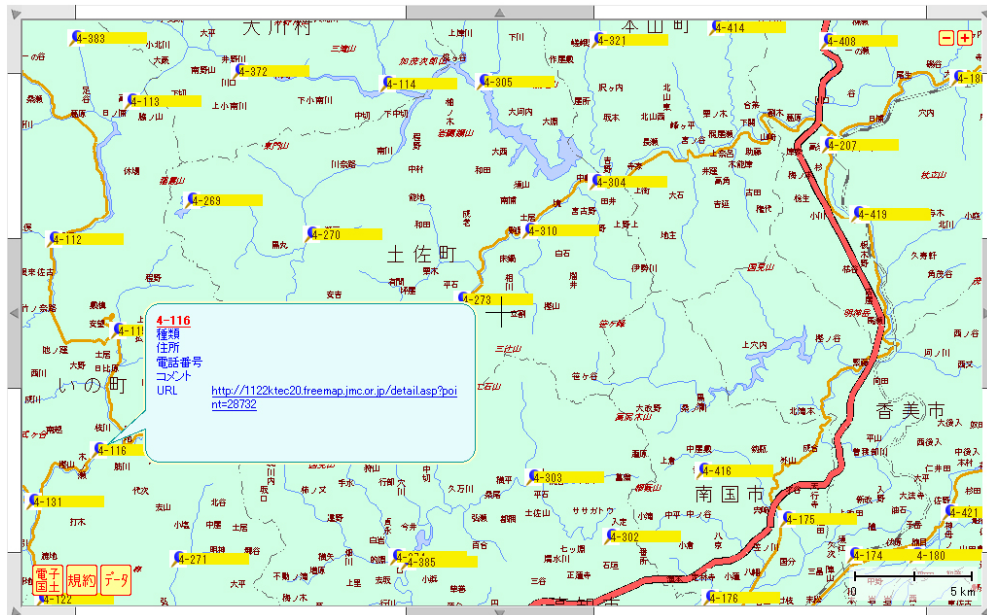


Figure 4.4: GCPs distribution in remote area

Using these three pieces of information, users can understand the location and can easily identify the location on satellite images.

基準点座標詳細 - この成果は、ALOSデータ利用公募型研究により得られたものである -

名称	4-269	画像をクリックすると別ウィンドウで拡大表示されます。
種類		
住所		
電話番号		
URL	http://www.infra.kochi-tech.ac.jp/takalab/gcp_correction/sikoku/cc/33.html	
ひと言		
測地系	世界測地 第4系	
X座標	X -11740.538	
Y座標	Y 82078.393	
標高	Z 1128.019	
緯度	Lat 33.740047	
経度	Long 133.373289	
測量日	Obs. date 2009/5/2	
観測箇所	Antenna 谷の角	

Location on AVNIR-II

VRS-GPS Antenna

Done

Figure 4.5: GCP and its information

4.3 Background of Study

Many previous researches say image registration accuracies are depend on number of GCP (Ebner and Strunz, 1988; Ebner et al., 1991, 1992). Fraser and Shao, 1997 concentrated on evaluating the effects of different numbers of GCP and different combinations of images. Guoqing Zhou and Ron Li (2000) found that the strength of GCP distribution affects the accuracy of planimetric and vertical (x, y, z) and demonstrated that GCP distribution on a straight line constitute a weak geometric configuration (figure 4.6). They recommended GCP distribution is a

critical key to achieving high accuracy. *Tao, C.V., (2001)* demonstrated Rational Function Model (RFM) solution is sensitive to the distribution of GCP and to the number of GCP. The accuracy of the RFM solution is dependent on the availability and the usage of GCP (*Grodecki, 2001*). If accurate RFM solutions are required, GCP are needed.

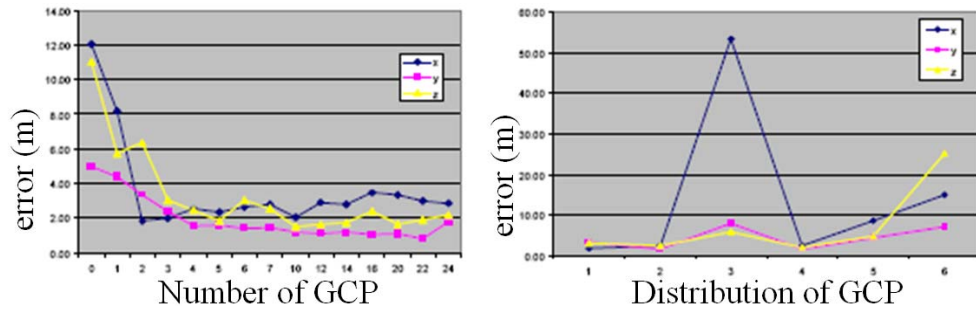


Figure 4.6: Error depends on number of GCP and distribution (credit: Guoqing Zhou and Ron Li 2000)

Error propagation can be tracked through the geometric processing steps as of number of GCP, distribution of GCP, accuracy of GCP (*Toutin, Th. 2003*).

Satellite observation to the earth surface becoming finer to finer; however error propagations in the imagery are still contaminating by the sensor-target's displacements. Even systematic errors could be removed by trajectory model and rigorous sensor model; random errors are remaining and removal of which are arduous because of they are decline from the physical law. An understanding on error distribution could be the way to select transformation model to register satellite imagery. The study was conducted by selecting two types of transformation models. The transformation methods are 3D affine and 3D projective; in other hand, ALOS's PRISM data was used in the process of errors investigation. A very high resolution laser scanner DSM data (*the dataset is provided by RaMSE: Kokusai Kogyo Co., Ltd.*) is used by assuming to be true height values. Two types of control points are selected carefully; ground control points (GCP) for accurate registration and check point (CP) to investigate error distributions. Using the visually selected control points, random errors are generated with Box-Muller random error generation method and added it to selected GCP. Error vectors are simulated from distribution of random errors and the vectors shown that the directional distribution of error geometry. Finally, the comparison will be listed to select suitable transformation model in satellite image registration.

Surveyors and engineers are believes that errors are propagated in all kind of measurements (*Charles D. Ghilani, 2006*). The errors could be systematic errors or random errors. Systematic errors could be removed by applying related physical conditions and correct measurement procedures; however, random errors are

complex and it is the arduous work to remove the random errors. Despite the fact the random errors could be the least by understanding the geometry of errors and iterating the observations (Charles D. Ghilani, 2006). The geometry of random errors was generated from the simulated ground control points (GCP) and check points (CP). Iterated error generation processes were applied when simulating GCP and CP. Random values are generated by Box-Muller method and it provides the random values from the standardized Gaussian distribution (Cox et al., 2001). The generated random values are added to the true value of control points, which formed error contaminated new control points. By iterating the transformation process with generated new control points, the geometric errors were calculated. The study compares the error geometry between 3D affine transformation and 3D projective transformation. The different distributions of GCP and CP are also demonstrated. The approach of this study investigated that the geometry and distribution of errors in ALOS PRISM image. The technique could be used in the processes of precise 3D measurement. The 3D measurement could be investigated the land displacement based on times or land form changes for location based monitoring and analysis such as land-use/land-cover mapping and management, disaster risk mapping and management, urbanization prediction and management, water resource management.

4.4 Overview of error propagation study

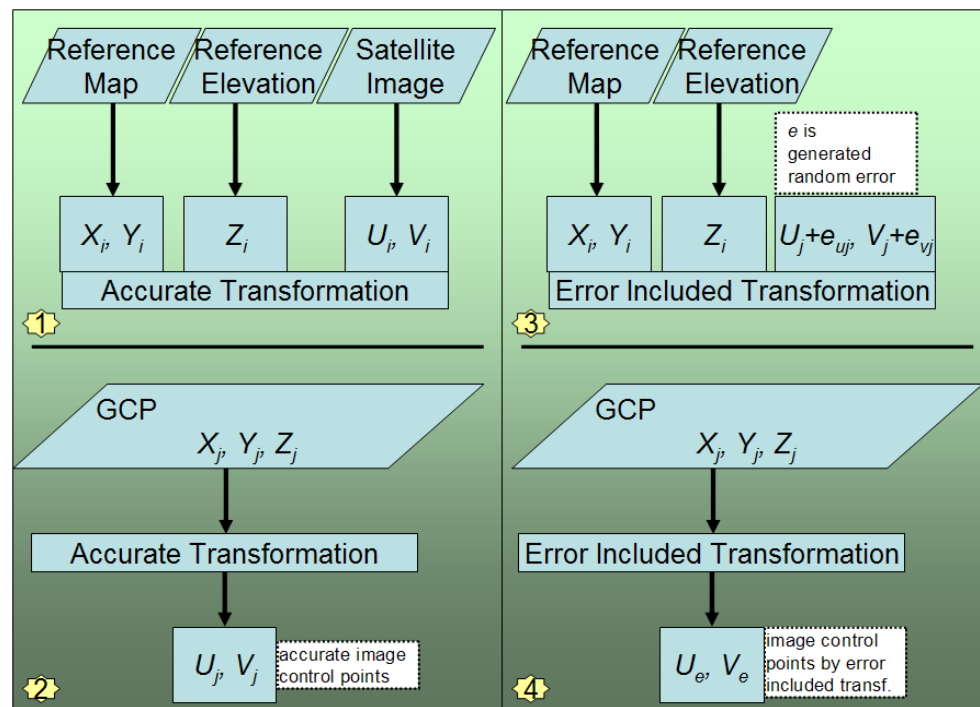


Figure 4.7: The general flow of error geometry generation

There are four main steps in the procedure (figure 4.7). X and Y values are collected from reference IKONOS image and Z values are from the DSM data. Assume U_i and V_i data are collected from target ALOS's PRISM data visually. True values U_j and V_j are produced by transform function (equation 4.1). In other side, generated errors (e_{up}, e_{vj}) are added to the assumed image control points. The process of errors contamination was iterated to 20 times.

$$\begin{aligned} U_i &= f_u(x, y, z) \\ V_i &= f_v(x, y, z) \end{aligned} \quad (4.1)$$

where U_i, V_i are true values; $f_u(x, y, z)$ and $f_v(x, y, z)$ are transform function for assume u and v .

4.5 GCP and check points

In this case of sample datasets, ground control points (GCP) and check points (CP) are visually interpreted as assumed control points. They are well distributed (figure 4.8) in the selected area. Each control points are recognized in both of target (ALOS-PRISM nadir image) and reference image (IKONOS image).

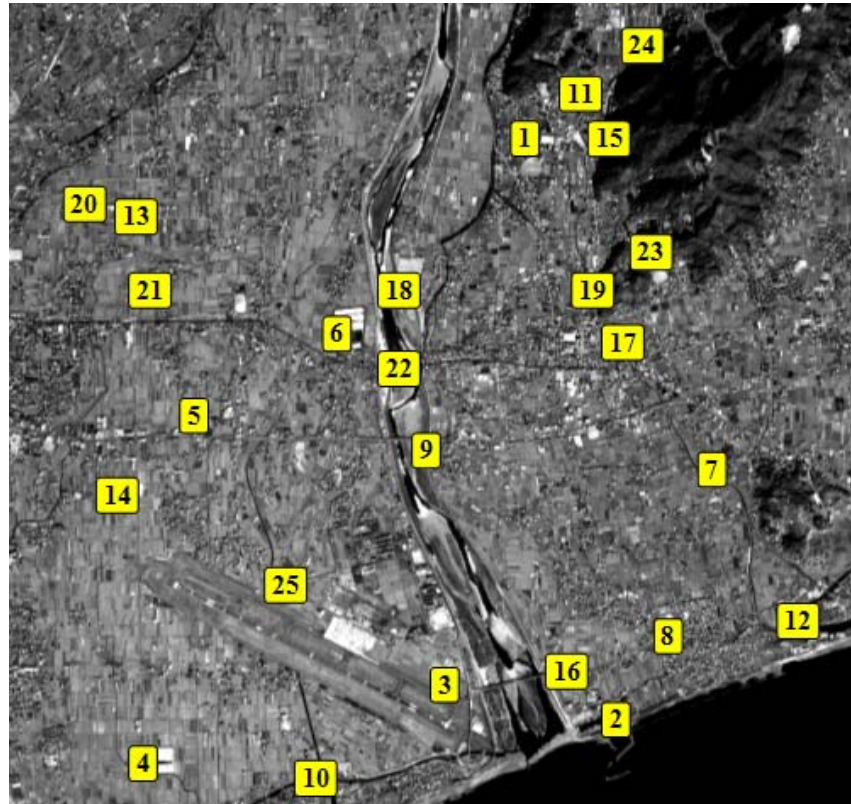


Figure 4.8: Location of distribution of control points

By inserting coordinates of GCP and CP to the (equation 4.1), a pair new transformation function (equation 4.2 & 4.3) is produced.

For GCP, transform function look like (equation 4.2).

$$\begin{aligned} U_{gcp} &= f_u(x, y, z) \\ V_{gcp} &= f_v(x, y, z) \end{aligned} \quad (4.2)$$

For CP, the transform function becomes equation (4.3).

$$\begin{aligned} U_{cp} &= f_u(x, y, z) \\ V_{cp} &= f_v(x, y, z) \end{aligned} \quad (4.3)$$

The transform function will be changed base on the method of transformation. Two transformation methods are applied in this study. Prior to the transformation, the generated errors are intruded the to true control points (GCP and CP).

4.6 Method of random error generation

Box-Muller transformation (equation 4.4) generated a pair of independent standard uniformly distributed random numbers from the standardized Gaussian distribution $N(0, 1)$. Suppose U_1 and U_2 are independent random variables that are uniformly distributed in the interval $(0, 1)$; then, X_0 and X_1 are independent random variables with a normal distribution of standard deviation 1.

$$\begin{aligned} X_0 &= \sqrt{-2 \ln U_1} \cos(2\pi U_2) \\ X_1 &= \sqrt{-2 \ln U_1} \sin(2\pi U_2) \end{aligned} \quad (4.4)$$

The errors were generated using this algorithm. Let u_g and v_g are independent errors with mean of true GCPs of standard deviation 1. The Box-Muller random variable generator is generated the error with specific mean u and v .

$$\begin{aligned} u_g &= \sqrt{-2 \ln U_1} \cos(2\pi U_2) + u \\ v_g &= \sqrt{-2 \ln U_1} \sin(2\pi U_2) + v \end{aligned} \quad (4.5)$$

where u_g and v_g are x, y coordinate of simulated error, which are intruded into the control points; and u and v are true location of the coordinate of the control points. This generation process was iterated for 20 times, producing 20 pairs of simulated control points.

If we substitute coordinate of control points to the equation 5, the error contaminated control points are calculated (equation 4.6 & 4.7).

For GCP,

$$\begin{aligned} u'_{gcp} &= \sqrt{-2 \ln U_1} \cos(2\pi U_2 + U_{GCP}) \\ v'_{gcp} &= \sqrt{-2 \ln U_1} \sin(2\pi U_2 + V_{GCP}) \end{aligned} \quad (4.6)$$

and for CP,

$$\begin{aligned} u'_{cp} &= \sqrt{-2 \ln U_1} \cos(2\pi U_2 + U_{CP}) \\ v'_{cp} &= \sqrt{-2 \ln U_1} \sin(2\pi U_2 + V_{CP}) \end{aligned} \quad (4.7)$$

The simulated control points are generated using the (equations 4.7 & 4.8). 20 pairs of control points are simulated by iterating 20 times. The iteration data will be the basic requirement for the error geometry investigation and errors vector drawing.

4.7 Transformation methods

Amount several types of transformation methods, we selected 3D affine and 3D projective transformation methods based on effectiveness of them for error propagation study.

In general, 3D affine transformation model can be represented mathematically by the (equation 4.8).

$$\begin{aligned} u &= ax + by + cz + d \\ v &= ex + fy + gz + h \end{aligned} \quad (4.8)$$

In other hand, 3D Projective transformation model can be represented by (equation 4.9).

$$\begin{aligned} \mathbf{u} &= \frac{a_1x + a_2y + a_3z + a_4}{b_1x + b_2y + b_3z + 1} \\ \mathbf{v} &= \frac{a_5x + a_6y + a_7z + a_8}{b_1x + b_2y + b_3z + 1} \end{aligned} \quad (4.9)$$

Based on parameter requirements of these functions, we selected 25 control points including 11 GCP and 14 CP scattering in the image. There four types of pattern distribution (figure 4.9) are using in transformation processes to investigate the influent of ground condition to error geometry.

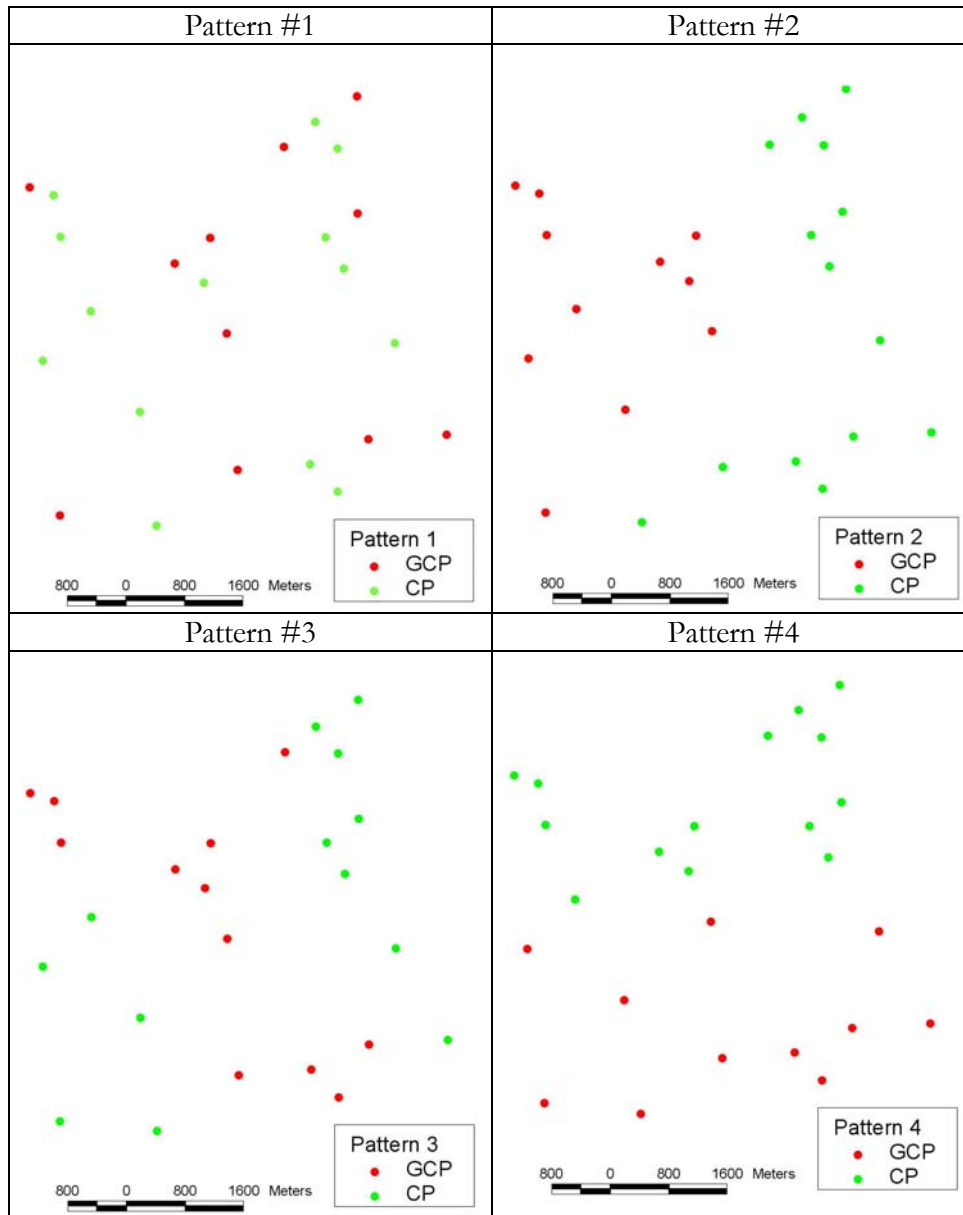


Figure 4.9: Distributions of GCP and CP

In distributions, pattern 1 is well distributed over the image; while others are less scattering. Pattern 2 image control points are selected in the non-elevated plain and it is located in the right side of image. When pattern 4 is distributed in the lower part of study area, pattern 3 is distributed diagonally ranging from elevated area through plain with narrow distribution. Based on these patterns, the propagated error geometry is calculated.

4.8 Addition of errors

The new values' computational process can be propagated with the errors, which are presented originally in the original direct observations. Therefore, the functions of the original errors could be containing as indirectly in next measurements (*Charles D. Ghilani, 2006*). By definition, the difference of a true value and its measured value is an error. Suppose, ε is the error in an observation, y is the measured value, and μ its true value, then

$$\varepsilon = y - \mu \quad (4.10)$$

The equation 10 will yield a new equation 11, when we substitute the control points.

$$\begin{aligned} \varepsilon_u &= U_t \pm U' \\ \varepsilon_v &= U_t \pm V' \end{aligned} \quad (4.11)$$

where U_p, V_t is true value and U', V' is simulated control points.

4.9 Error vectors

There are 20 times of 25 control points (11 GCP and 14 check points) were generated with different distribution patterns. By locating all those error coordinates, the vectors of error geometry are represented in (*figure 4.10*). The error vectors are originated on true coordinate; then which are headed to the trend of errors. In the figure 5 the errors vectors are magnified to 100 times to understand clearly. In patterns, red dots represent GCP; blue dots represent control points; green arrows represent 3D affine error vectors; and golden errors represent 3D projective error vectors.

In the results of root mean squares error (RMSE), the well distributed ground control points (GCP) produced the least RMSE comparing with others distribution patterns. The orientation of the height changed the vector of errors. The result shows that the aggregated ground control points (GCP) contributed low potential in distance check points. Uneven ground condition reflected more complex error geometry.

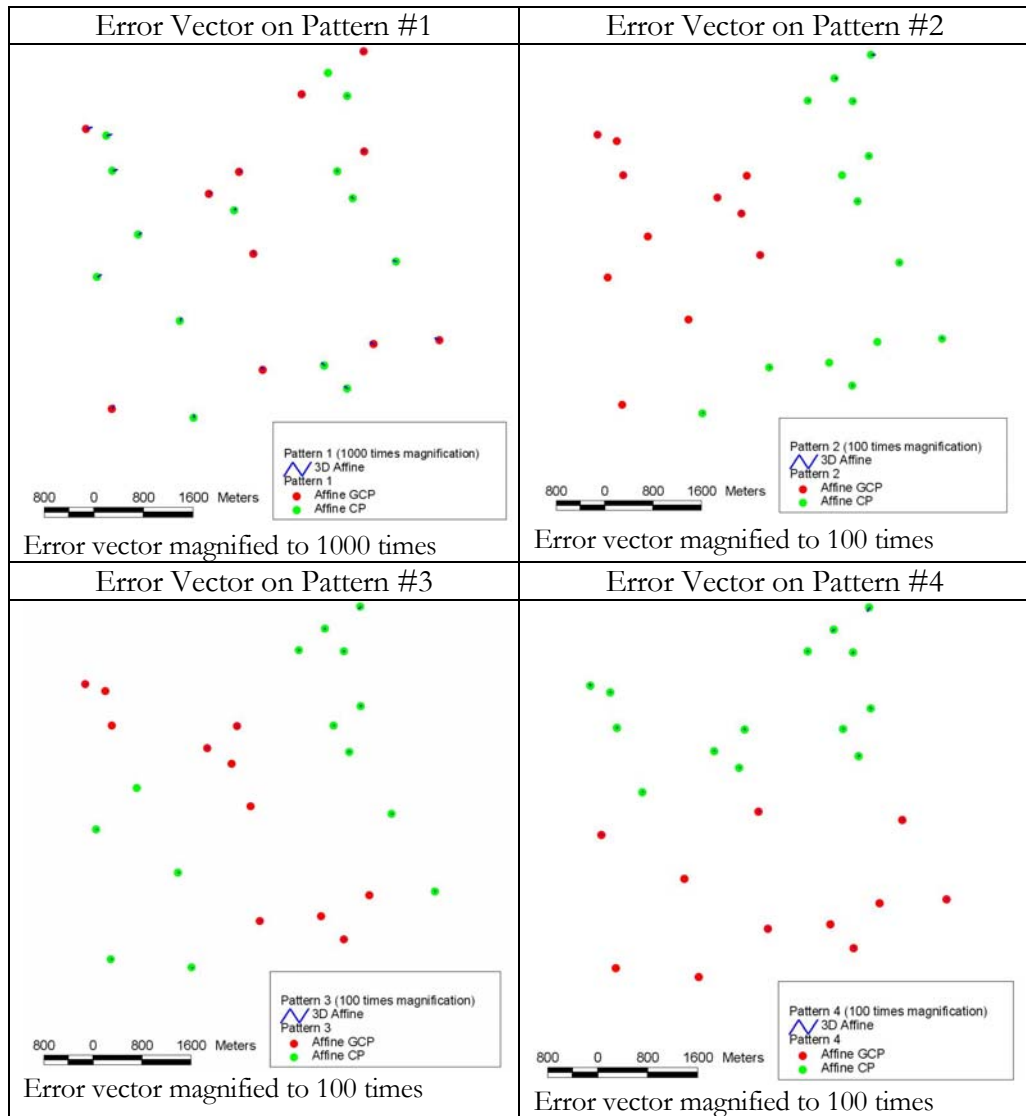


Figure 4.10(a): Error vectors of each distribution (3D affine)

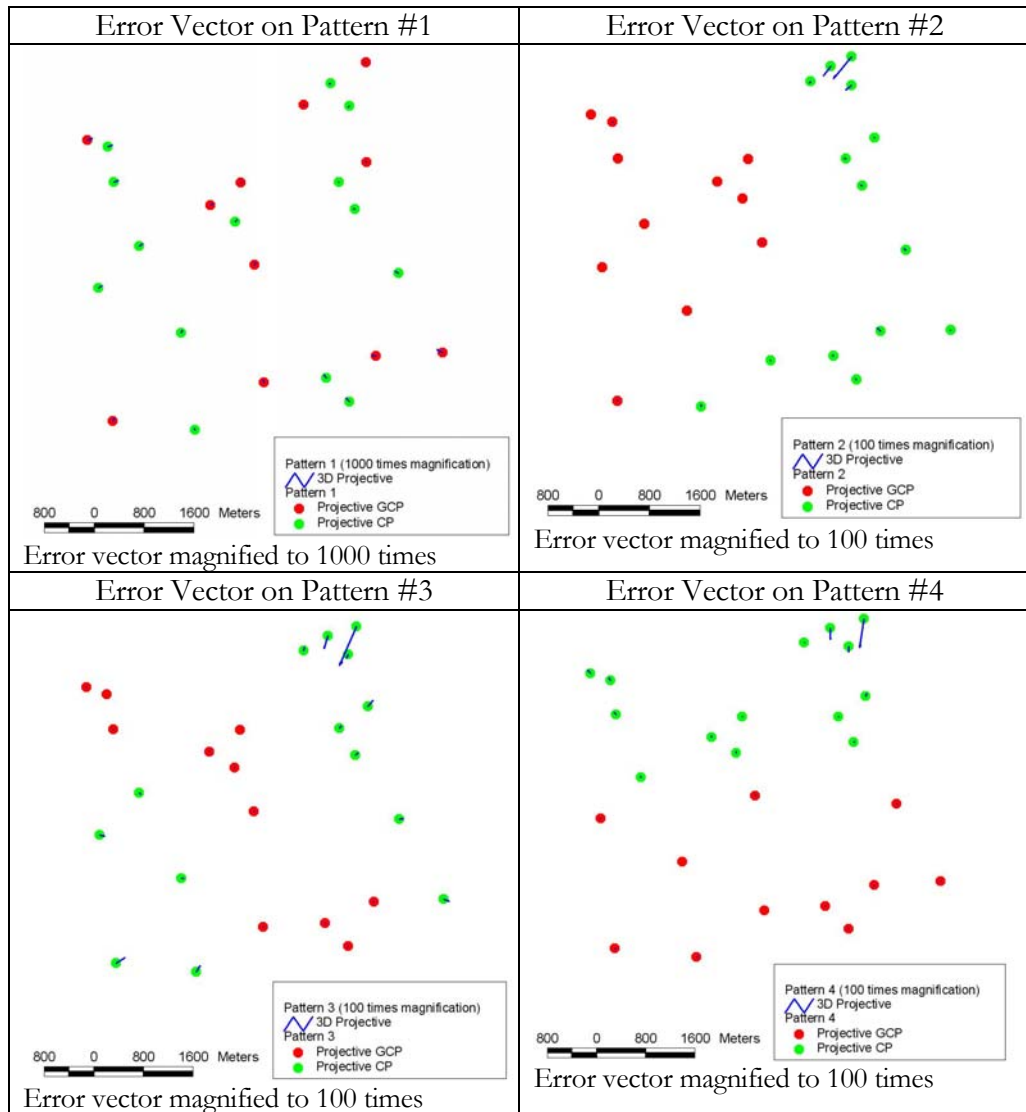


Figure 4.10(b): Error vectors of each distribution (3D projective)

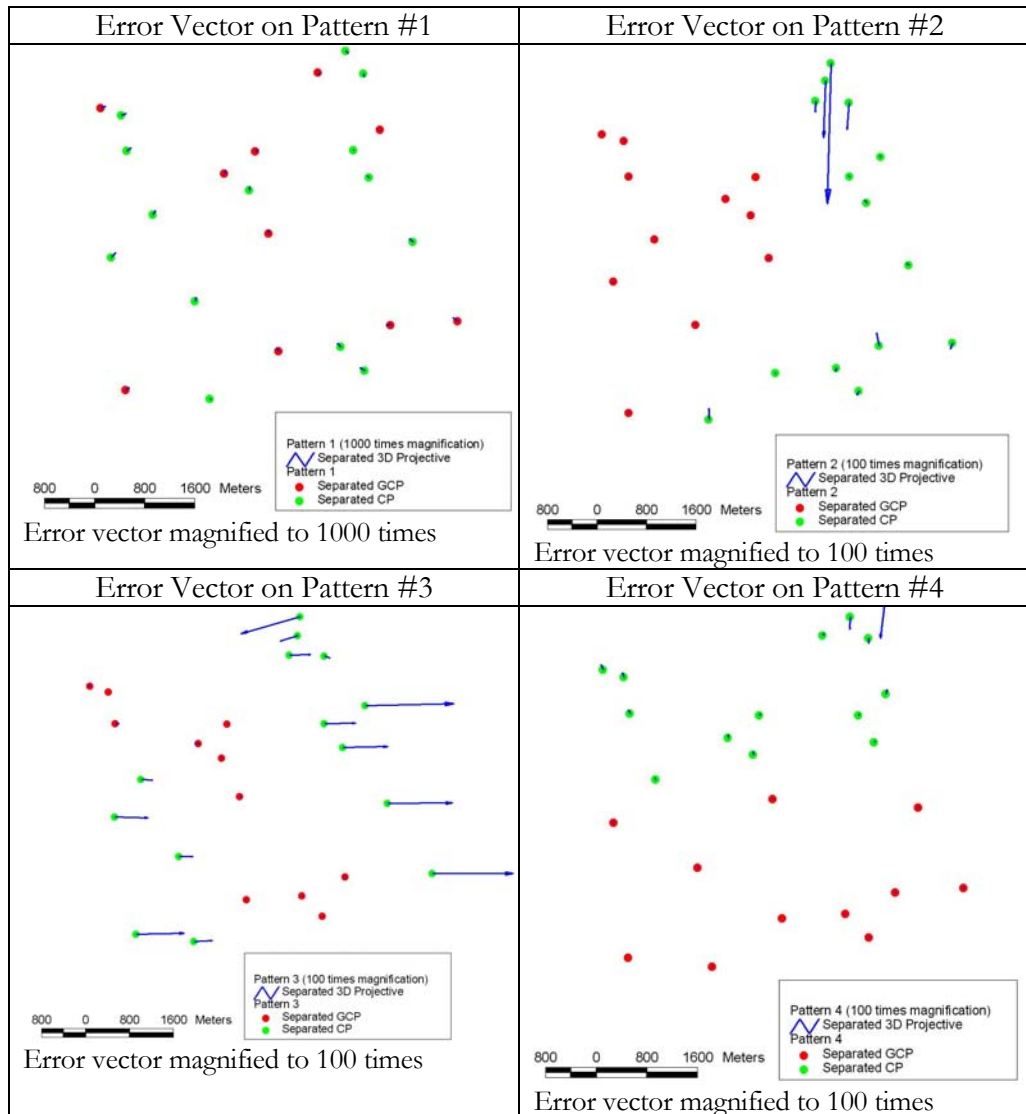


Figure 4.10(c): Error vectors of each distribution (Separated 3D projective)

As of transformations processes, 3D projective transformation gives more precision than 3D affine when the location is closed to the GCP. Apart from control points and transformation methods, the study proved that the selection and distribution pattern have positively related with condition for ground elevation.

4.10 Discussion on error trends

A mount four selected patterns (*figure 4.11*) Pattern 1, well distributed control points, had more precision consistency than others for both of parallel and perspective transformations. Pattern 2, GCPs from non-elevated area, precision consistency is high around GCPs when CPs has low precision. Pattern 3, narrow distribution GCPs, produced high accuracy around GCPs. Unexpectedly; 3D affine transformation had more potential around CPs. Pattern 4, GCPs from lower part of selected area, issued low precision for some GCP located in other part with elevated area.

Apart from this, the reference IKONOS image may be intruded by original errors. Accuracy may be increased whilst the accurate original reference image is considered.

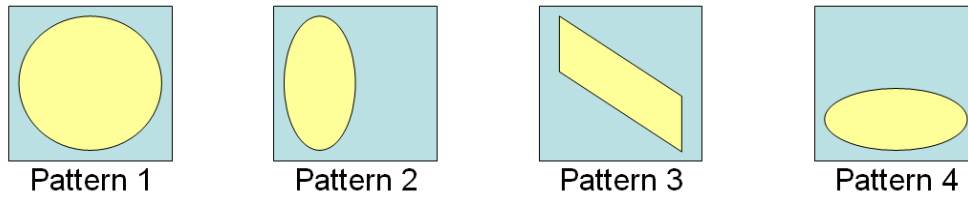


Figure 4.11: Selected distribution patterns

Geometric transformation results for three models are listed in the tables (*table 4.1 and table 4.2*) for four selected pattern. RMS errors around the GCP show perspective models with high accuracy.

Table 4.1: Geometric transformation results for each pattern around GCP

unit in pixel Distribution	3D Affine GCP		3D Projective GCP		Separated 3D Projective GCP	
	u	v	u	v	u	v
Pattern 1	3.45	3.5	2.282	2.976	0.706	2.602
Pattern 2	2.3	1.3	2.853	1.116	4.367	0.738
Pattern 3	1.9	1.2	2.157	2.412	5.587	1.259
Pattern 4	2.1	1.6	2.09	1.685	1.969	1.752

Parallel geometric model, 3D affine, produced less RMS errors around the CP. The errors show consistency of image global accuracy.

Table 4.2: Geometric transformation results for each pattern around CP

unit in pixel Distribution	3D Affine CP		3D Projective CP		Separated 3D Projective CP	
	u	v	u	v	U	V
Pattern 1	2.39	2.64	3.101	3.463	4.101	3.479
Pattern 2	7.38	9.85	44.57	41.38	96.09	15.77
Pattern 3	3.16	4.71	19.61	31.8	134.7	4.274
Pattern 4	2.71	3.72	4.353	43.63	4.308	16.55

IMAGE CONTROL POINTS

5.1 Background

An electromagnetic wave includes a *sinusoidal electric wave* (E) and a similar *magnetic wave* (M) at right angles, both being perpendicular to the direction of propagation. The wave theory describes electromagnetic energy as traveling in a harmonic, sinusoidal fashion at the “velocity of light,” c . The distance from one wave peak to the next is the *wavelength* λ , and the number of peaks passing a fixed point in space per unit time is the wave *frequency* ν . From the basic physics, wave obey the general equation

$$c = \nu\lambda \quad (5.1)$$

Since c is essentially a constant (3×10^8 m/sec), frequency ν and wavelength λ for any given wave are related inversely, and either term can be used to characterize a wave into a particular form.

In remote sensing, it is most common to categorize electromagnetic waves by their wavelength location within the *electromagnetic spectrum*. The most prevalent unit used to measure wavelength along the spectrum is the micrometer (μm). A micrometer equals 1×10^{-6} m.

The particle theory suggests that electromagnetic radiation is composed of many discrete units called *photons* or *quanta*. The energy of a quantum is given as

$$Q = h\nu \quad (5.2)$$

where

Q = energy of a quantum, Joules (J)

h = Planck’s constant, 6.626×10^{-34} J sec

ν = frequency

We can relate the wave and quantum models of electromagnetic radiation behavior by solving Eq. 5.1 for ν and substituting into Eq. 5.2 to obtain

$$Q = \frac{hc}{\lambda} \quad (5.3)$$

Thus, we see that the energy of a quantum is inversely proportional to its wavelength. *The longer the wavelength involved, the lower its energy content.* Systems operating at long wavelengths must “view” large areas of the earth at any given time in order to obtain a detectable energy signal.

All matter at temperatures above absolute zero (0°K, or -273°C) continuously emits electromagnetic radiation. Thus, terrestrial objects are also sources of radiation, though it is of considerably different magnitude and spectral composition than that of the sun.

How much energy any object radiates is, among other things, a function of the surface temperature of the object? This property is expressed by the *Stefan-Boltzmann law*, which states that

$$M = \sigma T^4 \quad (5.4)$$

where

M = total radiant existence from the surface of a material, watts (W) m⁻²

σ = *Stefan-Boltzmann constant*, $5.6697 \times 10^{-8} \text{ Wm}^{-2}\text{K}^{-4}$

T = absolute temperature (K) of the emitting material

It is important to note that the total energy emitted from an object varies as T^4 and therefore increases very rapidly with increases in temperature. Also, it should be noted that this law is expressed for an energy source that behaves as a *blackbody*. A blackbody is a hypothetical, ideal radiator that totally absorbs and reemits all energy incidents upon it.

The *dominant wavelength*, or wavelength at which a blackbody radiation curve reaches a maximum, is related to its temperature by *Wien's displacement law*,

$$\lambda_m = \frac{A}{T} \quad (5.5)$$

where

λ_m = wavelength of maximum spectral radiant existence, μm

$A = 2898 \mu\text{m K}$

T = temperature, K

Thus, for a blackbody, the wavelength at which the maximum spectral radiant existence occurs varies inversely with the blackbody's absolute temperature.

When electromagnetic energy is incident on any given earth surface feature, three fundamental energy interactions with the feature are possible. Various fractions of the energy incident on the element are *reflected*, *absorbed*, and/or *transmitted*.

Applying the principle of conservation of energy, we can state the interrelationship between these three energy interactions as

$$E_I(\lambda) = E_R(\lambda) + E_A(\lambda) + E_T(\lambda) \quad (5.6)$$

where E_I denotes the incident energy, E_R denotes the reflected energy, E_A denotes the absorbed energy, and E_T denotes the transmitted energy, with all energy components being a function of wavelength λ .

Because many remote sensing systems operate in the wavelength regions in which reflected energy predominates, the reflectance properties of earth features are very important.

$$E_R(\lambda) = E_I(\lambda) - [E_A(\lambda) + E_T(\lambda)] \quad (5.7)$$

That is, the reflected energy is equal to the energy incident on a given feature reduced by the energy that is either absorbed or transmitted by that feature.

The geometric manner in which an object reflects energy is also an important consideration. This factor is primarily a function of the surface roughness of the object. *Specular* reflectors are flat surfaces that manifest mirror-like reflectors, where the angle of reflection equals the angle of incidence. *Diffuse* (or *Lambertian*) reflectors are rough surfaces that reflect uniformly in all directions. Most earth surfaces are neither perfectly extremes. When the wavelength of incident energy is much smaller than the surface height variations or the particle sizes that make up a surface, the surface is diffuse. In remote sensing, we are most often interested in measuring the diffuse reflectance properties of terrain features.

The reflectance characteristics of earth surface features may be quantified by measuring the portion of incident energy that is reflected. This is measured as a function of wavelength and is called *spectral reflectance*, ρ_λ . It is mathematically defined as

$$\rho_\lambda = \frac{E_R(\lambda)}{E_I(\lambda)}$$

$$\rho_\lambda = \frac{\text{energy of wavelength } \lambda \text{ reflected from the object}}{\text{energy of wavelength } \lambda \text{ incident upon the object}} \times 100 \quad (5.8)$$

where ρ_λ is expressed as a percentage.

The atmosphere affects the “brightness,” or *radiance*, recorded over any given point on the ground in two almost contradictory ways. First, it attenuates

(reduces) the energy illuminating a ground object (and being reflected from the object). Second, the atmosphere acts as a reflector itself, adding scattered, extraneous *path radiance* to the signal detected by the sensor. By expressing these two atmospheric effects mathematically, the total radiance recorded by the sensor may be related to the reflectance of the ground object and the incoming radiation or irradiance using the equation

$$L_{tot} = \frac{\rho ET}{\pi} + L_p \quad (5.9)$$

where

L_{tot} = total spectral radiance measured by sensor

ρ = reflectance of object

T = transmission of atmosphere

L_p = path radiance, from the atmosphere and not from the object

It should be noted that all of the above factors depend on wavelength. Also the radiance (E) stems from two sources: (1) directly reflected “sunlight” and (2) diffuse “skylight,” which is sunlight that has been previously scattered by the atmosphere. The relative dominance of sunlight versus skylight in any given image is strongly dependent on weather conditions (e.g., sunny vs. hazy vs. cloudy). Likewise, irradiance varies with the seasonal changes in solar elevation angle and the changing distance between the earth and sun.

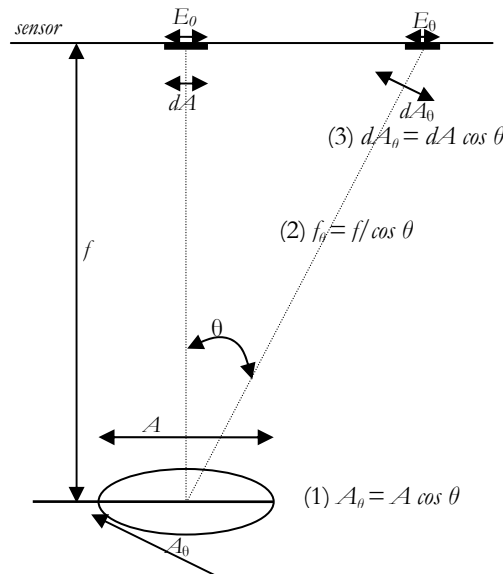


Figure 5.1: Factors causing exposure falloff

Probably the most important geometric effect influencing sensor exposure is *exposure falloff*. This extraneous effect is a variation in focal plane exposure purely associated with the distance an image point is from the image center. Because of falloff, a ground scene of spatially uniform reflectance does not produce spatially uniform exposure in the focal plane. Instead, for a uniform ground scene, exposure in the focal plane is at a maximum at the center of the sensor format and decreases with radial distance from the center.

The factors causing falloff are depicted in figure 5.1, which shows a sensor being exposed to a ground area assumed to be of uniform brightness. For a beam of

light coming from a point directly on the optical axis, exposure E_0 is directly proportional to the area, A , of the lens aperture and inversely proportional to the square of the focal length of the lens, f^2 . However, for a beam exposing a point at an angle θ off the optical axis, exposure E_θ is reduced from the E_0 for three reasons:

1. The effective light-collecting area of the lens aperture, A , decreases in proportion to \cos when imaging off-axis areas ($A_\theta = A \cos \theta$).
2. The distance from the camera lens to the focal plane, f_θ , increases as $1/\cos \theta$ for off-axis points, $f_\theta = f/\cos \theta$. Since exposure varies inversely as the square of this distance, there is an exposure reduction of $\cos^2 \theta$.
3. The effective size of a sensor area element, dA , projected perpendicular to the beam decreases in proportion to $\cos \theta$ when the element is located off-axis, $dA_\theta = dA \cos \theta$.

Combining the above effects, the overall theoretical reduction in sensor exposure for an off-axis point is

$$E_\theta = E_0 \cos^4 \theta \quad (5.10)$$

where

θ = the angle between the optical axis and the ray to the off-axis point

E_θ = the sensor exposure at the off-axis point

E_0 = the exposure that would have resulted if the point had been located at the optical axis

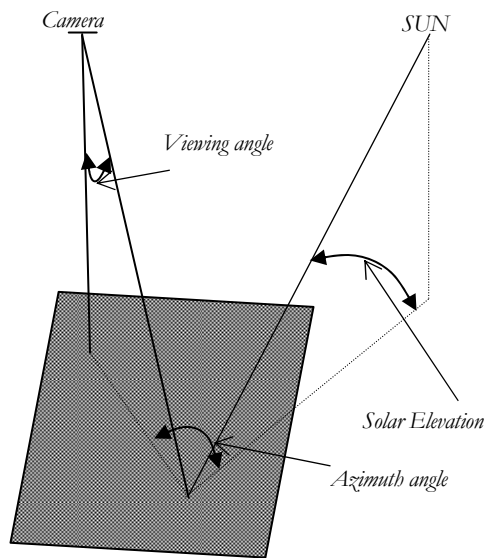


Figure 5.2: Sun-object-image angular relationship

For most cameras this relationship takes on the form

$$E_\theta = E_0 \cos^n \theta \quad (5.11)$$

Because modern cameras are normally constructed in such a way that their actual falloff characteristics are much less severe than the theoretical \cos^4 falloff, n in the above equation is normally in the range 1.5 to 4.

The location of an object within a scene can also affect the resulting sensor exposure as illustrated in Figure 5.2 and

5.3. Figure 5.2 illustrates the relationships that exist among *solar elevation*, *azimuth angle* and *camera viewing angle*.

Figure 5.3 illustrates geometric effects that can influence the apparent reflectance and exposure. In (a), the effect of differential shading is illustrated in profile view. Because of relief displacement, vertical features are imaged in an aerial photograph slightly in side view as well as in top view.

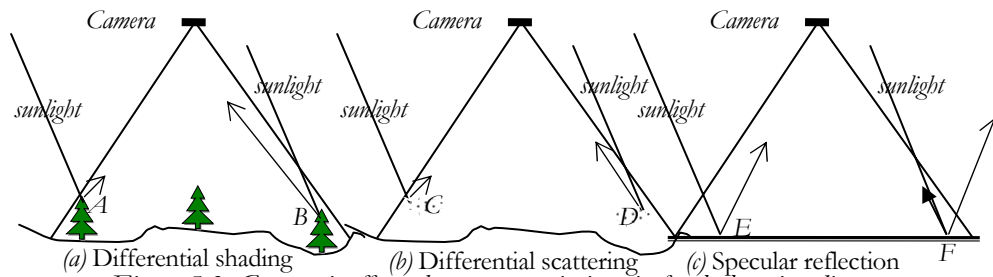


Figure 5.3: Geometric effects that cause variations in focal plane irradiance

Because the sides of features may be either sunlit or shaded, varied exposures can result from identical ground objects at different locations in the image. The sensor receives more energy from the sunlit side of the tree at B than from the shaded side of the tree at A. Differential shading is clearly a function of solar elevation and object height, with a stronger effect at low solar angles. The effect is also compounded by differences in slope and aspect (slope orientation) over terrain of varied relief.

Figure 5.3b illustrates the effect of *differential atmospheric scattering*. Back-scatter from atmospheric molecules and particles adds light to that reflected from ground features. The sensor receives more atmospheric backscatter from area D than from area C due to geometric relationships. In some analyses, the variation in this “airlight,” or path radiance, component is small and can be ignored. However, under hazy conditions, differential quantities of airlight often result in varied exposure across a photograph.

Yet another problem in many analyses is the presence of *specular* reflections in a scene. Photographs which are taken over water bodies often show areas of specular reflection. They represent the extreme in directional reflectance. Figure 2.9c illustrates the geometric nature of this problem. Immediately surrounding point E on the image, a considerable increase in exposure would result from specular reflection. These mirror-like reflections normally contribute little information about the true character of the objects involved. Because of the low information content of specular reflections, they are avoided in most analyses.

5.2 Main thought of ICP

Orthorectified aerial photo is required to generate image control points to register high resolution satellite imageries; however, very high resolution satellite images have possibility. Previous researches demonstrated that VHR satellite imagery have possibility to replace aerial photo in many photogrammetric applications (Grodecki, 2001, 2002; Jacobsen, 2003; Gene Dial and Jacek Grodecki 2005). An observation confirmed the potential of ICP in image matching between image to image (Kersten et al., 2000; Gruen, A., 2000).

Difficulty of identification of GCP on maps or on very high resolution satellite images promoted a consideration of using ICP (figure 5.4). Once we are generated ICP it can be used for many other satellite images. What we need is only we need to change satellite acquisition parameters.

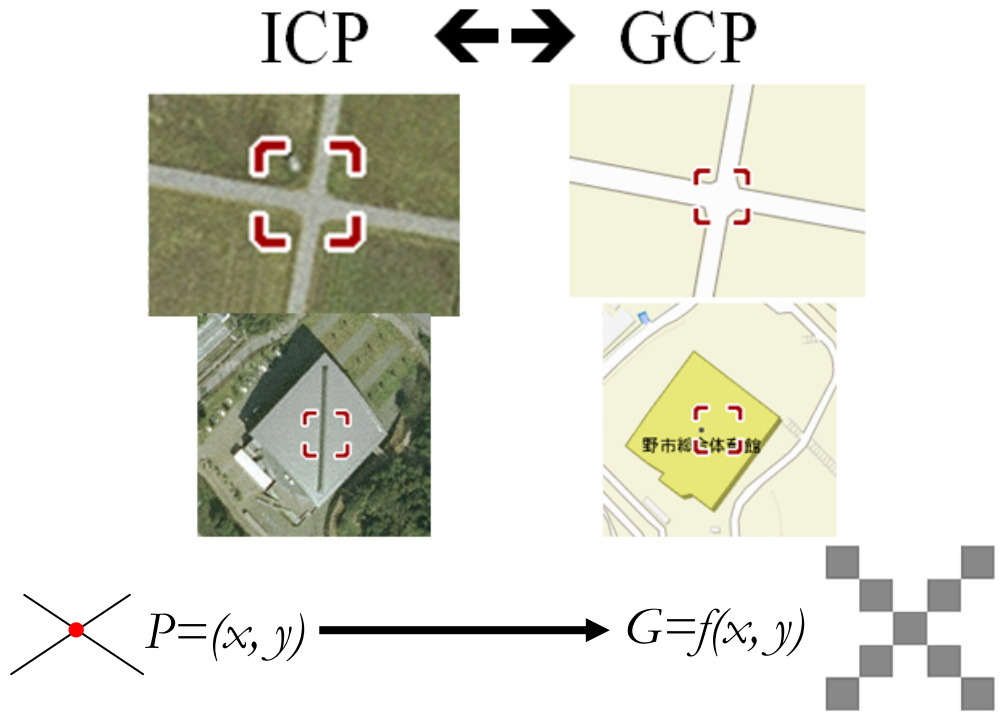


Figure 5.4: ICP as of GCP

The main methodology of image control points generation includes two parts: reference data simulation for the target image and GCP selection by Least Squares Image Matching method as shaded part in (figure 5.5). DSM data is used as main input to the system to generate shade and shadow for both of the satellite images. Firstly, shade and shadow for the reference image are simulated from DSM by giving the sun illumination parameters of its sensor. The simulated data is used to eliminate shade and shadow from the reference image. Then, reflectance value is generated from the average of red ratio and green ratio of the

reference multi-spectral image after shade and shadow effects are removed from the image ratios. Again, shade and shadow data are simulated with the sun illumination parameters of the targeted image and it is integrated with generated reflectance value as seasonal vegetation pattern contribution. Finally, simulated template image is generated for image matching algorithm.

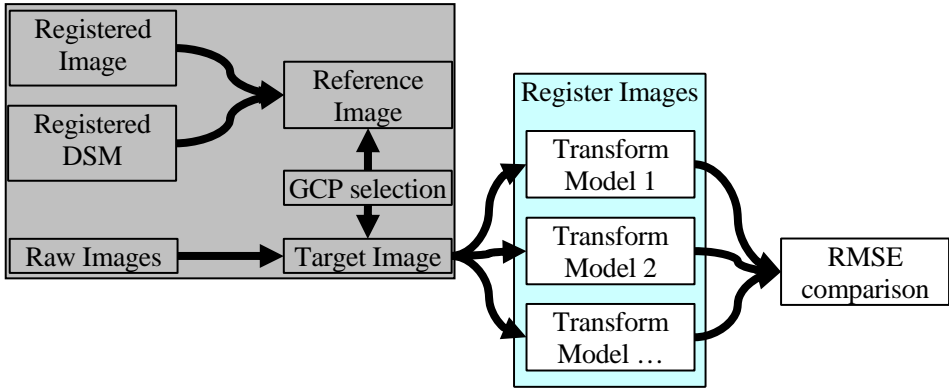


Figure 5.5: Methodology of image control points generation

The study presented simulated image-to-image registration using two high resolution satellite images and one DSM dataset. As a default requirement of the three-dimensional transformation model, we assumed high value (z) from DSM data and (x, y) values from the reference image. The target image is presented as (u, v) in the model.

5.3 Data

Three data sets are used for the study, two of which are high resolution satellite images and the other of which is a DSM. A study area is selected based on the availability of required DSM data and satellite images. The area is also considered in the distribution of 3-axes (x, y, z) such as elevation, even/uneven land and water body. The selected area covers Kochi Ryoma Airport and downstream of the Monobe river in Kochi prefecture of Japan.

5.4 Reference image generation

Shade and shadow are important factors for topographical contribution to satellite images (Proy et al. 1989, Meyer et al. 1993, Tzēlepis and Nakos 2003). The shade and shadow in satellite images are generated from DSM with cosine ratio method (Tanaka 1932, Tanaka 1950, Yoeli 1976, Horn 1982, Imbof 1982, Yoeli 1983, Murai 1998).

When similar shade and shadow are generated for reference DSM, they can be used to remove shade and shadow effects from the multi-spectral image. Without shade and shadow effects in the image, the multi-spectral images can be used as reflectance value for seasonal vegetation pattern contribution to reference image generation. Alternatively, another set of shade and shadow data are generated with the target image acquisition parameters. A simulated image is calculated by integrating reference reflectance values to shade and shadow data. Reference image patches are selected from the simulated image and matched with the target image. The matching process produces accurate GCPs. By the algorithm, GCP can be generated automatically from the image without field visit. Finally, selected GCP are use to register the target image. The methodology flow of reference image simulation is presented in (figure 5.6).

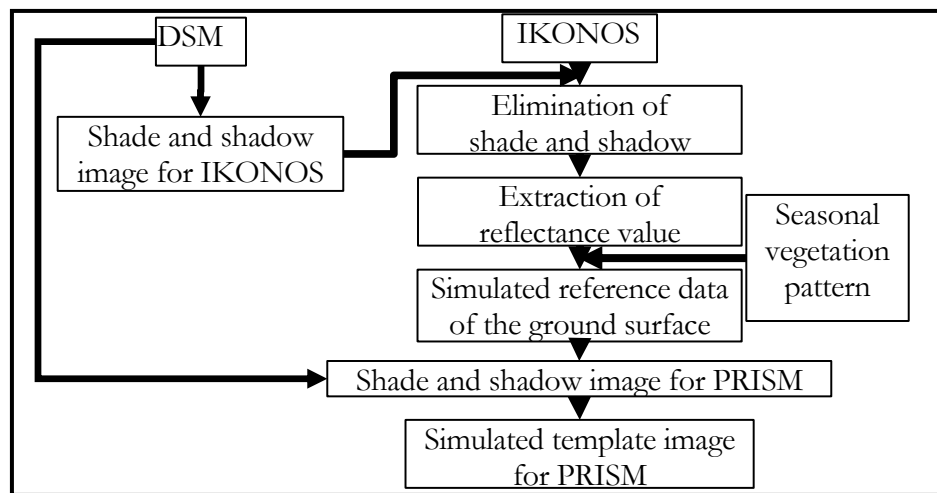


Figure 5.6: Reference image simulation

5.4.1 Removal of shade and shadow affects

In satellite images, shade and shadow effects are caused by sun illumination on topographic elevation of the earth's surface. When image acquisition times are different, the hill shading is different (figure 5.7). The effect of hill shading is based on Lambertian surface terrain (Moellering and Kimerling, 1990, Murai 1998).

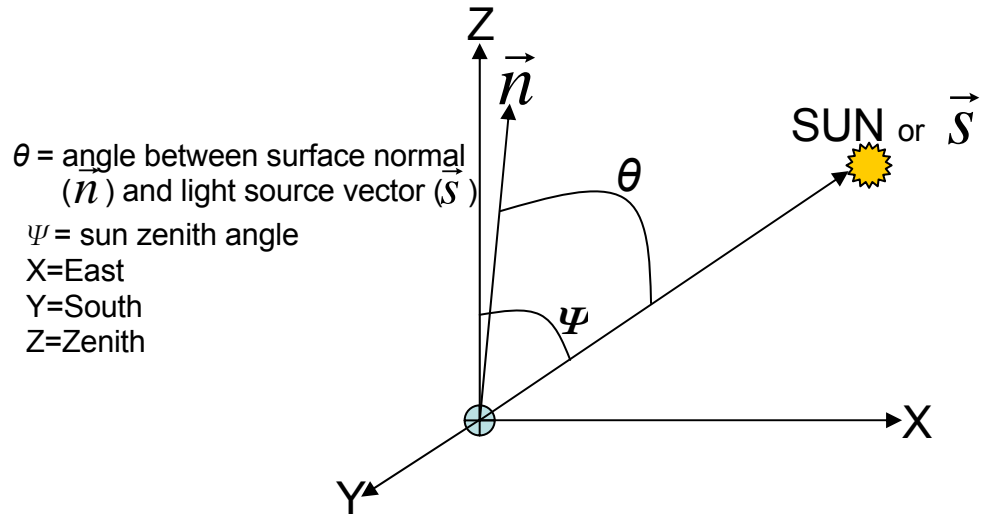


Figure 5.7: Hill shading parameter by the sun illumination

The mathematic formula for lambertian surface can express in equation 5.12, equation 5.13 and equation 5.14.

$$\text{shade's gray value} = \text{cosine ratio} (\cos \theta) = |n_x s_x + n_y s_y + n_z s_z| \leq 1.0. \quad (5.12)$$

where θ is angle between incident light vector \vec{s} and surface normal \vec{n} .

$$\vec{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \frac{1}{\sqrt{a^2 + b^2 + 1}} \begin{bmatrix} a \\ b \\ -1 \end{bmatrix}; a, b \text{ are Laplacian.} \quad (5.13)$$

$$\vec{s} = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} = \begin{bmatrix} \cos \beta \sin \alpha \\ \cos \beta \cos \alpha \\ \sin \alpha \end{bmatrix} \alpha = \text{azimuth from north, } \beta = \text{sun elevation angle} \quad (5.14)$$

In the generated cosine ratio, the values range from 0 to 1. It is used to describe the shading gradients of the topographic elevation in the image. When it is equal to 0, it is assumed that the surface under is shadow. The shaded effect becomes less when it is close to 1. Thus, using the cosine ratio values, the correction can recover the shade and shadow effects of the topographic contribution to images.

First, the shade relief data is generated from the DSM using satellite image's acquisition sun illumination parameters. To remove the shade and shadow effects in the reference image, each band of IKONOS multi-spectral image is multiplied

by threshold values of the cosine ratio (equation 5.15 and equation 5.16). The threshold is assumed by visual interpretation of the shade relief image.

$$ss_{(iko)} = \cos_{(iko)} \theta \begin{cases} \text{if } \cos_{(iko)} \theta = 0.0 \sim 0.6 \rightarrow \text{then } 0.0 \text{ for shadow and deep shade} \\ \text{if } \cos_{(iko)} \theta = 0.6 \sim 1.0 \rightarrow \text{then } 1.0 \end{cases} \quad (5.15)$$

$$f_{remove}(u, v) = f_{DN}(x, y) \times ss_{(iko)} \quad (5.16)$$

where,

$ss_{(iko)}$: defined shade/shadow layer for IKONOS (on the threshold of cosine ratio)
 $\cos_{(iko)} \theta$: cosine of the angle between incident light vector \vec{s} and surface normal \vec{n} of IKONOS image

$f_{remove}(u, v)$: grey level function to represent shade/shadow removed data (as reflectance)

$f_{DN}(x, y)$: grey level function to represent each band of multi-spectral image (IKONOS)

The generated cosine values range from 0 to 1. Based on the reference image's acquisition sun elevation angle on steeply sloping of land surface elements, the shadow and deep shade is threshold as *while* cosine values are 0 to 0.6; *otherwise* free from shade and shadow (Meyer et al. 1993). This threshold value is assumed as a weight in generation of reference reflectance values, when it is difficult to remove by band ratio. Other noise in the reference multi-spectral image will be removed by averaging the respected band ratios (Tzelepis et al. 2003), presented in following step (equations 5.17 & 5.18). Using the above algorithm some blank spaces appears as *zero* in value to represent defined shade and shadow. Those blank values are filled by Delaunay triangulation calculation from the surrounding values (figure 5.8).

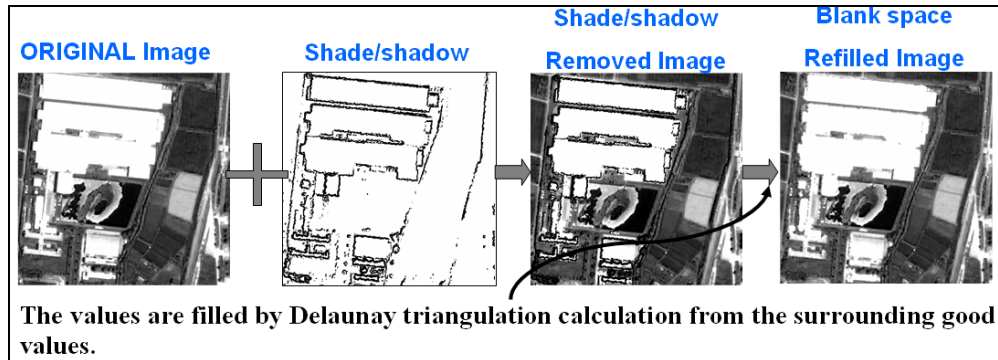


Figure 5.8: Algorithm for shade/shadow removal

5.4.2 Extraction of reflectance value and reference reflectance image simulation

The spectral range of PRISM panchromatic band is similar to the range of the red band and green band of IKONOS image (*table 1.1*). Based on this similarity, the PRISM's panchromatic band can be generated from the red and green bands of IKONOS image.

The reference spectral reflectance data is generated by the band ratio (*Holben and Justice 1980, Lillesand and Kiefer 1994*) of two bands - the red and the green (*equation 5.18*). Selection of band ratio in this study is based on the similarity of spectral bandwidth between PRISM panchromatic and the red to green band of IKONOS image (*table 1.1*).

$$\text{bandratio}_{red} = \frac{f_{red}(u, v)}{(f_{blue}(u, v) + f_{green}(u, v) + f_{red}(u, v) + f_{ir}(u, v))} \quad (5.17)$$

$$\text{bandratio}_{green} = \frac{f_{green}(u, v)}{(f_{blue}(u, v) + f_{green}(u, v) + f_{red}(u, v) + f_{ir}(u, v))}$$

$$f_{ave}(u, v) = \frac{\text{bandratio}_{red} + \text{bandratio}_{green}}{2.0} \quad (5.18)$$

where:

$f_{ave}(u, v)$: grey level function of generated reflectance value

$f_{blue}(u, v), f_{green}(u, v), f_{red}(u, v), f_{ir}(u, v)$: are multi-spectral bands after removing shadow and deep shade

Panchromatic band can simulate from the combination of appropriate multi-spectral bands (*Laben et al. 2000*). We used multi-spectral bands from IKONOS image in this study.

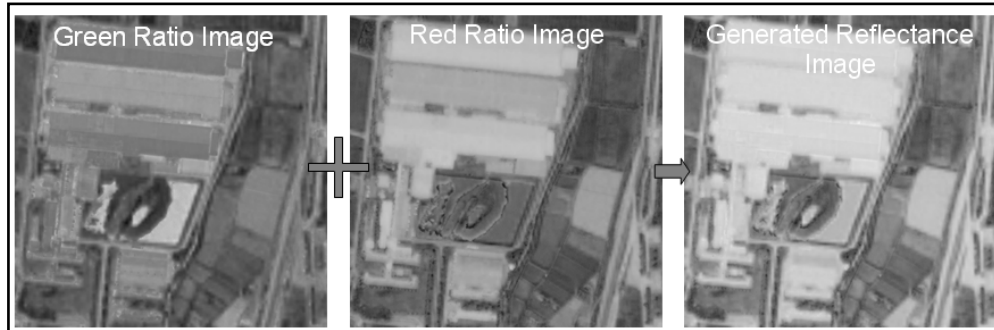


Figure 5.9: Algorithm for reflectance image generation

5.4.3 Addition of shade and shadow for target image

The simulated template image (*figure 5.10*) is generated by multiplying the generated PRISM's shade and shadow with generated reference reflectance value (*equation 5.20*).

$$SS_{(PRISM)} = \cos_{(PRISM)}\theta \quad (5.19)$$

$$f_{simulated}(u, v) = f_{ave}(u, v) \times SS_{(PRISM)} \quad (5.20)$$

where:

$SS_{(PRISM)}$: the shade relief data for PRISM

$\cos_{(PRISM)}\theta$: cosine of the angle between incident light vector \vec{s} and surface normal \vec{n} of PRISM image

$f_{simulated}(u, v)$: grey level function of simulated PRISM's reference image

$f_{ave}(u, v)$: grey level function of the generated reflectance value

Similarly, shade relief data is generated from the DSM for PRISM with its acquisition parameters. Then, the generated shade relief data is applied to the generated reference reflectance value. The reference reflectance value is assumed as reference seasonal vegetation in this study because both image acquisition times are in the same season. However, it should be separated for images acquired in different seasons.



Figure 5.10: Algorithm for PRISM image simulation

5.5 Image matching

The automatic determination of image coordinates of conjugated points for image registration is one of the main difficulties in satellite remote sensing. This topic can be defined as image matching, and image matching is the location of corresponding phenomena in different images. There are number of image matching techniques, two common branches are area based matching and feature based matching (*Usery 1988, Argialas 1990*). Area based matching is performed point by point, and small rectangular pixel arrays with the interest point at the

center are used as the basic unit. They are given in the form of a matrix array $g(x,y)$ of observed grey level values. The computational task is then to obtain an optimal matching by transforming one array of grey level values into the other (Ackermann, 1984). In this study we discuss single point matching, which is a semi-automatic matching method because the initial point needs to be selected virtually. In the branch of area based matching, the matcher rectangular matrix array $g_i(x,y)$ with interest point in center is matching to a pixel through the target image $g_s(x,y)$. However, cross correlation technique for area based single point matching is common; brightness values of image pixel promoting main disturbance to matching accuracy. There are four main factors responsible for the values in an image: illumination, reflectivity, geometry, and viewpoint (Marr, 1980). To achieve unambiguous matches, all the four factors should be considered, but in fact, only the positions and the orientation of the exposure are known. On the other hand, the geometric and radiometric distortions and improving matching accuracy, the least squares matching method was recommended by Gruen (1985) and Rosenholm (1987). LSM is the approach in which the optimum match is defined by a transformation, while the radiometric differences are modeled by an additive and a multiplicative parameter. The main purpose for the single point LSM in this study is to refine the approximate matching positions and to improve matching accuracy. The basic algorithm is represented with following generalized equations:

$$g_t(x_t, y_t) + n(x_t, y_t) = g_s(x_s^0 + dx, y_s^0 + dy)$$

$$\Delta g + n = \frac{dg_s}{dx} \partial x + \frac{dg_s}{dy} \partial y \quad (5.21)$$

where:

$g_t(x,y), g_s(x,y)$: gray level function of the target and the template patch

$n(x,y)$: error vector

$\frac{dg_s}{dx}, \frac{dg_s}{dy}$: gradients in x and y directions, which can be declared as g_{x_s}, g_{y_s} .

In the implementation, both the radiometric quality and the geometrical differences are considered. Equation (5.21) can be rewritten as (5.22). Ground control points are selected by Least Squares Image Matching method. The basic algorithm for matching between the target image and the reference patch can be rewrite in the following (equation 5.23).

$$g_t(x, y) = h_0 + h_1 * g_s[(a_0 + a_1x + a_2y), (b_0 + b_1x + b_2y)] \quad (5.22)$$

The observation equation can be obtained by linearizing the above (equation 5.22).

$$\begin{aligned} \Delta g(x_i, y_j) + v(x_i, y_j) = & g_x(x_i, y_j)da_0 + x_i g_x(x_i, y_j)da_1 + y_j g_x(x_i, y_j)da_2 \\ & + g_y(x_i, y_j)db_0 + x_i g_y(x_i, y_j)db_1 + y_j g_y(x_i, y_j)db_2 + dh_0 + g_1(x_i, y_j)dh_1 \end{aligned} \quad (5.23)$$

where:

$a_0, a_1, a_2, b_0, b_1, b_2$: unknown parameters for affine transformation

b_0, b_j : unknown coefficients for radiometric correction

$$\Delta g(x, y) = g_t(x, y) - g_s(x, y)$$

$g_x(x, y) = [g_t(x+1, y) - g_t(x-1, y)] / 2$: grey level function of the target image

$g_y(x, y) = [g_t(x, y+1) - g_t(x, y-1)] / 2$: grey level function of the reference patch

Though the least squares processing, the weighted mean of the right hand array (with the squares of the gradients as weights) can be chosen to represent the right image area:

$$x_s = \frac{\sum_{i,j} x_{i,j} [g_x(x_i, y_j)]^2}{\sum_{i,j} [g_x(x_i, y_j)]^2}, y_s = \frac{\sum_{i,j} y_{i,j} [g_y(x_i, y_j)]^2}{\sum_{i,j} [g_y(x_i, y_j)]^2} \quad (5.24)$$

This procedure is iterative until the magnitude of the computed translation is smaller than a threshold. After iteration, the search window is resampled from the original image. The bilinear interpolation function is applied to calculate the new grey level values for each grid in the search window. The mathematical formula is:

$$\begin{aligned} g_{x,y} = & g_{i,j}(x_{i+1} - x)(y_{j+1} - y) + g_{i+1,j}(x - x_i)(y_{j+1} - y) \\ & + g_{i,j+1}(x_{i+1} - x)(y - y_j) + g_{i+1,j+1}(x - x_i)(y - y_j) \end{aligned} \quad (5.25)$$

Where,

i, j are the number of rows and columns on the image.

$g_{i,j}$ is the corresponding grey level value, and

$g_{x,y}$ is the interpolated grey level value at the position (x, y) .

The initial approximation is very important because both the matching reliability and convergence of least squares algorithm depend on it. If the approximated position is close to the ideal position, the iterative procedure will gradually converge and the resulting coordinates are improved. Here, we use the results of correlation matching as the initial approximation. It is sufficient that the accuracy of these approximations is within two pixels. Hence, the convergence of computation is quicker and easier and final resulting coordinates are refined.

The linearized equation is implemented for corresponding pixels in target and reference image (equation 5.26). For $(n \times n)$ windows in both the target and reference image, we should have n^2 equation in 8 unknowns.

$$\begin{bmatrix} \Delta g^1 \\ \Delta g^2 \\ \Delta g^3 \\ \cdot \\ \cdot \\ \cdot \\ \Delta g^n \end{bmatrix} = \begin{bmatrix} g_x^1 & xg_x^1 & yg_x^1 & g_y^1 & xg_y^1 & yg_y^1 & 1 & g^1 \\ g_x^2 & xg_x^2 & yg_x^2 & g_y^2 & xg_y^2 & yg_y^2 & 1 & g^2 \\ g_x^3 & xg_x^3 & yg_x^3 & g_y^3 & xg_y^3 & yg_y^3 & 1 & g^3 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ g_x^n & xg_x^n & yg_x^n & g_y^n & xg_y^n & yg_y^n & 1 & g^n \end{bmatrix} \bullet \begin{bmatrix} da_0 \\ da_1 \\ db_0 \\ db_1 \\ db_2 \\ dh_0 \\ dh_1 \end{bmatrix} \quad (5.26)$$

The LSM improves both the matching reliability and the accuracy of parallax measurement. It also compensates for the radiometric and geometric distortions.

5.5.1 Selection of search image window

A study is conducted on selection of search image window. In the study, several image windows (51X51, 55X55, 61X61, 65X65, ..., 131X131) are investigated. The study demonstrated that 101X101 pixels produced good result (*figure 5.11*).

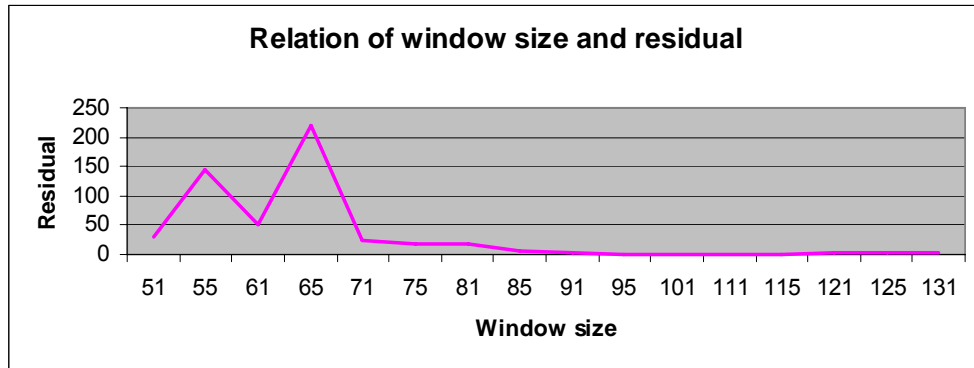


Figure 5.11: Relation of search window size and residual

5.5.2 Result of image matching

The initial values are visually interpreted from conjugated points of the target image and the simulated reference image. In the study, template patches are selected as the initial values from various land features such as bridge, building, roof corner and road crossing.

The patch size is 101X101 pixels in dimension. Although there are many control points are selected from generated reference image; only 10 GCP and 3 CP are selected to register the targeted image (*figure 5.12 & 5.13*) based on the minimum deviation values between LSM results and initial values.

The selection for GCP is done based on previous suggestion on distribution of GCP and number of GCP. Thus, number of GCP is minimized to the 10. Moreover, the GCP are spatially well distributed on the study area.

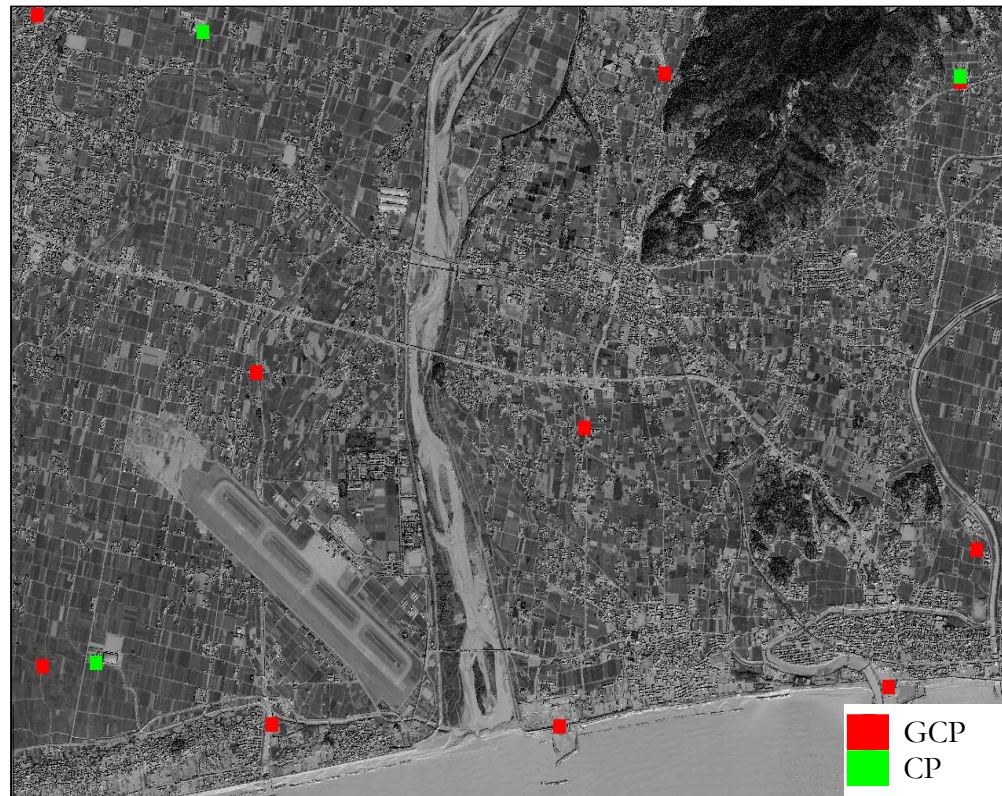


Figure 5.12: Simulated image and GCP/CP locations

The spatial resolution of simulated reference image is finer than the target image. This is because the original resolution of DSM and IKONOS images has higher than PRISM image.

The selected image control points are compared in spectral bands (blue, green, red and infrared), simulated reference image and subset of original PRISM image (figure 5.13). The patches size is 101 x 101 pixels. The image extent of original target PRISM image and reference simulated image are differed by the spatial resolution of original datasets. It means the reference datasets are 1 meters spatial resolution whilst the target image resolution is 2.5 meters.

GCP	Blue	Green	Red	Infrared	ICP	Original
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

Figure 5.13: Comparison of Selected GCP patches (bands, simulated image and origin PRISM image)

Moreover, the Least Squares Image Matching results for each selected GCP&CP template patches are listed in the table (table 5.1).

Table 5.1: Least Squares Image Matching Results

No	X	Y	Z	LSM Results	
				U	V
GCP (unit in meter) / LSM result (unit in pixel)					
1	18808.5	64034.5	31.181	2294.218082	876.447231
2	18169.5	61540.5	12.877	2267.229992	1899.832932
3	17966.5	59460.5	1.421	2370.935694	2732.387915
4	20586.5	59736.5	1.091	3370.599162	2389.322119
5	15676.5	59471.5	3.677	1475.983667	2929.657740
6	21157.5	63976.5	15.027	3211.878837	684.335257
7	21289.5	60691.5	2.842	3558.467598	1952.631823
8	15550.5	61937.5	8.648	1210.492061	1984.075516
9	13848.5	59876.5	3.245	731.922626	2934.947463
10	13807.5	64449.5	16.094	307.307190	1159.360909
Check Points (unit in meter) / LSM result (unit in pixel)					
1	21158.3	64012.9	17.28	3211	671
2	14274.7	59902.5	4.101	895	2888
3	15125.4	64325.3	15.99	830	1091
		Ground Control Points			
		Check Points			

5.6 Registration result using LSM generated ICP

Image registration is done using automatic selected GCP. GCPs are collected using generated ICP using LSM technique. 10 GCP and 3 check points are selected to register a subset ALOS PRISM image (figure 5.14 and table 5.2).



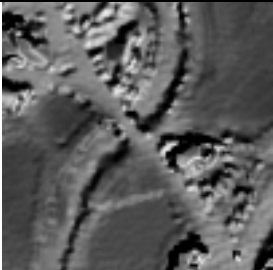
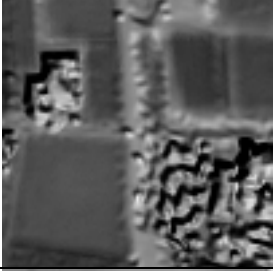

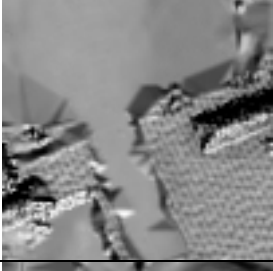
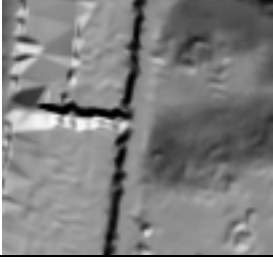
Figure 5.14: Distribution of selected GCP and CP on reference IKONOS image

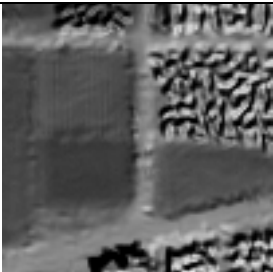
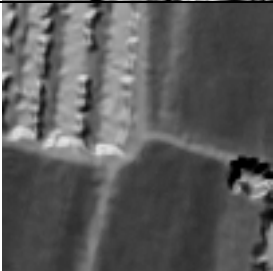
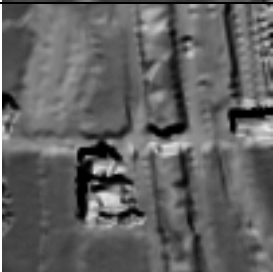
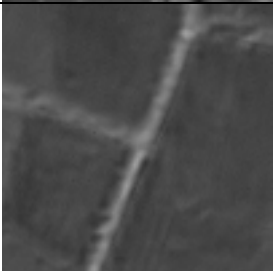

ICP accuracy is improved when artistic structures are selected; though, several ICP are collected from simulated reference image. Selected ICP are well distributed in the study area and the number ICP is minimized to 10.

The least squares matching result values are presented in the table and the selected 10 image control points are listed with LSM values (table 5.2).

Each ICP result RMS errors for each transformation model are presented in the table (table 5.3).

Table 5.2: Selected GCP and CP

VRS-GPS GCPs (unit in meter)				ICP
No	X	Y	Z	
1	18802.870	64035.018	68.269	
2	18167.078	61539.723	50.197	
3	17965.617	59460.534	1.882	
4	20586.083	59736.805	2.152	
5	15676.354	59471.203	4.303	

No	X	Y	Z	ICP
6	21157.932	63976.774	16.278	
7	21289.686	60691.948	3.159	
8	15550.528	61937.336	8.822	
9	13848.156	59876.205	3.586	
10	13807.500	64446.466	16.391	
No	X	Y	Z	
1	21158.3	64012.9	17.28	
2	14274.7	59902.5	4.101	
3	15125.4	64325.3	15.99	

5.7 Image registration

Image registration is demonstrated with three different transformation models. The models are considered 3 dimensionally. The new perspective model, the separated three-dimension transformation model (Kojima and Takagi 2008), produced the best result amount the three (*table 5.3*). In the new model, the denominators of u and v are the same in the three-dimension projective model; conversely, those are separated into each coefficient in the new model.

3D affine

$$u = a_1x + a_2y + a_3z + a_4,$$

$$v = a_1x + a_2y + a_3z + a_4.$$

3D projective

$$u = \frac{a_1x + a_2y + a_3z + a_4}{a_5x + a_6y + a_7z + 1},$$

$$v = \frac{a_8x + a_9y + a_{10}z + a_{11}}{a_5x + a_6y + a_7z + 1}.$$

separated 3D projective

$$u = \frac{a_1x + a_2y + a_3z + a_4}{a_5x + a_6y + a_7z + 1},$$

$$v = \frac{a_8x + a_9y + a_{10}z + a_{11}}{a_{12}x + a_{13}y + a_{14}z + 1}. \quad (5.27)$$

where:

a_1 to a_{14} are the 14 parameters for translation, scaling and rotation

The u and v values are acquired by Least Squares Image Matching method (*equation 5.24*) for the registration transformation model.

5.7.1 Result of image registration

In the registration, the locations of selected GCP and CP are well distributed on the image; on the other hand, the deviation between LSM results and initial values is minimized by visually selecting. When 10 GCP and 3 CP are selected to investigate the transformation result, Root Mean Squares errors are presented in *table 5.4*. The calculation of RMSE is demonstrated in (*equation 5.30*).

The RMSE for a collection of N values in x direction x_1, x_2, \dots, x_i is:

$$X_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} = \sqrt{\frac{x_1^2 + x_2^2 + \dots + x_i^2}{N}} \quad (5.28)$$

The RMSE for a collection of N values in y direction y_1, y_2, \dots, y_i is:

$$Y_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N y_i^2} = \sqrt{\frac{y_1^2 + y_2^2 + \dots + y_N^2}{N}} \quad (5.29)$$

The total RMSE for a collection of N values in two dimensional direction X_{RMS} , Y_{RMS} is:

$$RMS_{total} = \sqrt{X_{RMS}^2 + Y_{RMS}^2} \quad (5.30)$$

Table 5.3: Result of transformation models using LSM generated GCP

10 GCP & 3 CP	GCP (unit in pixel)		CP (unit in pixel)	
	RMSE in X	RMSE in Y	RMSE in X	RMSE in Y
3D Affine	1.468258282	1.6864345	1.089768388	0.963807802
3D Projective	1.345640394	1.061573792	1.46624605	0.606574587
Separated Projective 3D	1.156916535	1.065971395	0.68356141	0.715608595

The result produced very good registration accuracy; though some errors are contaminated x-directions. It can be original error of block noises from image compression jpeg artifact (ALOS CAL/VAL 2006).

Table 5.4: Registration results for each GCP and CP (Unit: Pixel)

Separated 3D Projective Transformation			3D Projective Transformation		3D Affine Transformation	
Residual Error of around GCP						
GCP 1	0.123118	0.099725	0.099765	0.033348	-0.356873	-1.167675
GCP 2	-1.451566	0.381426	-1.490621	0.369404	-0.811113	2.647025
GCP 3	0.699352	-1.46621	0.176255	-1.321265	1.377664	-2.278831
GCP 4	-0.694298	0.185213	-1.214247	0.199552	-1.816975	-1.247894
GCP 5	2.398085	-0.43142	3.081114	-0.140154	2.964214	1.313065
GCP 6	0.738066	-0.62379	0.708687	-0.420552	1.820123	1.449194
GCP 7	0.053154	1.506187	0.685071	1.275968	-0.825451	0.469720
GCP 8	-1.578302	-1.86057	-1.454666	-2.218795	-0.440030	-2.556952
GCP 9	-0.998334	1.476882	-1.194119	1.365053	-1.576975	1.633666
GCP 10	0.710723	0.732589	0.602763	0.857456	-0.334584	-0.261317
RMSE : u	1.156916535		1.345640394		1.468258282	
RMSE : v	1.065971395		1.061573792		1.686434500	
Residual Error of around CP						
CP 1	-0.516858	-0.75581	-1.993423	-0.600243	-0.106683	0.788102
CP 2	-0.695570	-0.49091	-0.101953	-0.506032	-0.839086	0.416441
CP 3	0.806727	-0.85091	1.570192	-0.698168	1.687406	-1.41147

RMSE : u	0.683561410		1.466246050		1.089768388	
RMSE : v	0.715608595		0.606574587		0.963807802	

Separated 3D Projective Transformation			3D Projective Transformation		3D Affine Transformation	
Residual Error of around GCP						
GCP 1	-0.617301	0.070970	-0.633809	0.052392	-0.575607	-2.353290
GCP 2	0.627864	-0.00679	0.630441	-0.037964	0.445254	2.852101
GCP 3	0.187565	-1.79992	-0.039377	-1.199986	1.050278	-2.212775
GCP 4	-0.349701	-0.01647	-1.403730	0.322549	-1.990805	-0.965905
GCP 5	1.494828	0.292400	2.499303	0.714837	2.990194	0.992306
GCP 6	0.748222	-0.63759	0.844040	-0.543671	1.680647	1.585493
GCP 7	-0.397708	1.889440	0.279839	1.325734	-0.906075	0.825665
GCP 8	-1.936053	-1.58508	-1.623403	-2.616106	-0.592028	-2.680077
GCP 9	-0.492225	1.044456	-1.004172	0.622729	-1.684159	1.357204
GCP 10	0.734321	0.748600	0.450857	1.359496	-0.417700	0.599279
RMSE : u	0.917337296		1.168566575		1.467791975	
RMSE : v	1.071020963		1.148579924		1.818404051	
Residual Error of around CP						
CP 1	-0.896857	-1.37325	-1.373610	-1.253310	-0.449404	0.817206
CP 2	-0.644459	-0.32987	-0.519876	-0.669844	-0.855728	-0.05953
CP 3	-0.203288	-1.82258	0.602819	-1.228084	1.322472	-1.65096
RMSE : u	0.648332602		0.916599650		0.945721275	
RMSE : v	1.331220453		1.084384408		1.064117158	

5.8 Error propagation for geometric models observation

The geometric transformation results from three models are added with random errors (*section 4.4*). Random error is generated using Box-Muller method to get a normally distribution population. In this method the geometric model result pixel value is used as seed of desire mean and standard deviation of the distribution (*equation 5.3 1, 5.32 & 5.33*).

$$\sigma = \text{Random} () \quad (5.31)$$

where,

σ = standard deviation and

if $\text{rand}() > 0.5$ then $\sigma=1$ else $\text{rand}() < 0.5$ then $\sigma=-1$ for orientation of the error arrow

$$\Delta e = (\text{SQRT}(-2 * \text{LN}(\text{RAND}())) * \text{SIN}(2 * \text{PI} * \text{RAND}()) * \sigma) \quad (5.32)$$

$$f(u, v)_e = f(u, v) + \Delta e \quad (5.33)$$

where,

Δe = the generated random error

$f(u, v)$ = the geometric model generated pixel value

$f(u, v)_e$ = the error added pixel value

The result is random error propagation to the models are presented (table 5.5). Generated random errors (equation 5.33) are added to geometric transformation values (Org_X , Org_Y). The error contaminated pixel values (Err_X , Err_Y) are generated and both of values are used to draw error vectors (figure 5.14).

Table 5.5: Error propagation to geometric transformation results (Unit: Pixel)

Control	Org_X	Org_Y	Err_X	Err_Y	Models
GCP1	2294.218082	876.447231	2293.861209	875.279556	3D Affine with Least Squares Matching values
GCP2	2267.229992	1899.832932	2266.418879	1902.479957	
GCP3	2370.935694	2732.387915	2372.313358	2730.109084	
GCP4	3370.599162	2389.322119	3368.782187	2388.074225	
GCP5	1475.983667	2929.657740	1478.947881	2930.970805	
GCP6	3211.878837	684.335257	3213.698960	685.784451	
GCP7	3558.467598	1952.631823	3557.642147	1953.101543	
GCP8	1210.492061	1984.075516	1210.052031	1981.518564	
GCP9	731.922626	2934.947463	730.345651	2936.581129	
GCP10	307.307190	1159.360909	306.972606	1159.099592	
CP1	3211.000000	671.000000	3210.893317	671.788102	3D Projective with Least Squares Matching values
CP2	895.000000	2888.000000	894.160914	2888.416441	
CP3	830.000000	1091.000000	831.687406	1089.588530	
GCP1	2294.218082	876.447231	2294.317847	876.480579	
GCP2	2267.229992	1899.832932	2265.739371	1900.202336	
GCP3	2370.935694	2732.387915	2371.111949	2731.066650	
GCP4	3370.599162	2389.322119	3369.384915	2389.521671	
GCP5	1475.983667	2929.657740	1479.064781	2929.517586	
GCP6	3211.878837	684.335257	3212.587524	683.914705	
GCP7	3558.467598	1952.631823	3559.152669	1953.907791	
GCP8	1210.492061	1984.075516	1209.037395	1981.856721	separated 3D projection
GCP9	731.922626	2934.947463	730.728507	2936.312516	
GCP10	307.307190	1159.360909	307.909953	1160.218365	
CP1	3211.000000	671.000000	3209.006577	670.399757	
CP2	895.000000	2888.000000	894.898047	2887.493968	
CP3	830.000000	1091.000000	831.570192	1090.301832	
GCP1	2294.218082	876.447231	2294.341200	876.546956	
GCP2	2267.229992	1899.832932	2265.778426	1900.214358	
GCP3	2370.935694	2732.387915	2371.635046	2730.921709	

GCP4	3370.599162	2389.322119	3369.904864	2389.507332	with Least Squares Matching values
GCP5	1475.983667	2929.657740	1478.381752	2929.226318	
GCP6	3211.878837	684.335257	3212.616903	683.711458	
GCP7	3558.467598	1952.631823	3558.520752	1954.138010	
GCP8	1210.492061	1984.075516	1208.913759	1982.214942	
GCP9	731.922626	2934.947463	730.924292	2936.424345	
GCP10	307.307190	1159.360909	308.017913	1160.093498	
CP1	3211.000000	671.000000	3210.483142	670.244188	
CP2	895.000000	2888.000000	894.304430	2887.509090	
CP3	830.000000	1091.000000	830.806727	1090.149093	
GCP1	2294.218082	876.447231	2293.642475	874.093941	3D Affine with VRS-GPS GCP
GCP2	2267.229992	1899.832932	2267.675246	1902.685033	
GCP3	2370.935694	2732.387915	2371.985972	2730.175140	
GCP4	3370.599162	2389.322119	3368.608357	2388.356214	
GCP5	1475.983667	2929.657740	1478.973861	2930.650046	
GCP6	3211.878837	684.335257	3213.559484	685.920750	
GCP7	3558.467598	1952.631823	3557.561523	1953.457488	
GCP8	1210.492061	1984.075516	1209.900033	1981.395439	
GCP9	731.922626	2934.947463	730.238467	2936.304667	
GCP10	307.307190	1159.360909	306.889490	1159.960188	
CP1	3211.000000	671.000000	3210.550596	671.817206	3D Projective with VRS-GPS GCP
CP2	895.000000	2888.000000	894.144272	2887.940473	
CP3	830.000000	1091.000000	831.322472	1089.349041	
GCP1	2294.218082	876.447231	2293.584273	876.499623	
GCP2	2267.229992	1899.832932	2267.860433	1899.794968	
GCP3	2370.935694	2732.387915	2370.896317	2731.187929	
GCP4	3370.599162	2389.322119	3369.195432	2389.644668	
GCP5	1475.983667	2929.657740	1478.482970	2930.372577	
GCP6	3211.878837	684.335257	3212.722877	683.791586	
GCP7	3558.467598	1952.631823	3558.747437	1953.957557	
GCP8	1210.492061	1984.075516	1208.868658	1981.459410	
GCP9	731.922626	2934.947463	730.918454	2935.570192	
GCP10	307.307190	1159.360909	307.758047	1160.720405	
CP1	3211.000000	671.000000	3209.626390	669.746690	Separated 3D Projective with VRS-GPS GCP
CP2	895.000000	2888.000000	894.480124	2887.330156	
CP3	830.000000	1091.000000	830.602819	1089.771916	
GCP1	2294.218082	876.447231	2293.600781	876.518201	
GCP2	2267.229992	1899.832932	2267.857856	1899.826136	
GCP3	2370.935694	2732.387915	2371.123259	2730.587993	
GCP4	3370.599162	2389.322119	3370.249461	2389.305652	
GCP5	1475.983667	2929.657740	1477.478495	2929.950140	
GCP6	3211.878837	684.335257	3212.627059	683.697665	
GCP7	3558.467598	1952.631823	3558.069890	1954.521263	
GCP8	1210.492061	1984.075516	1208.556008	1982.490432	

GCP9	731.922626	2934.947463	731.430401	2935.991919
GCP10	307.307190	1159.360909	308.041511	1160.109509
CP1	3211.000000	671.000000	3210.103143	669.626750
CP2	895.000000	2888.000000	894.355541	2887.670127
CP3	830.000000	1091.000000	829.796712	1089.177416

5.9 Error vectors for geometric transformation models

Error vectors are generated using values of each geometric transformation. Original start point of the vector is geometric transformation values and the end value of the vector is error added values.

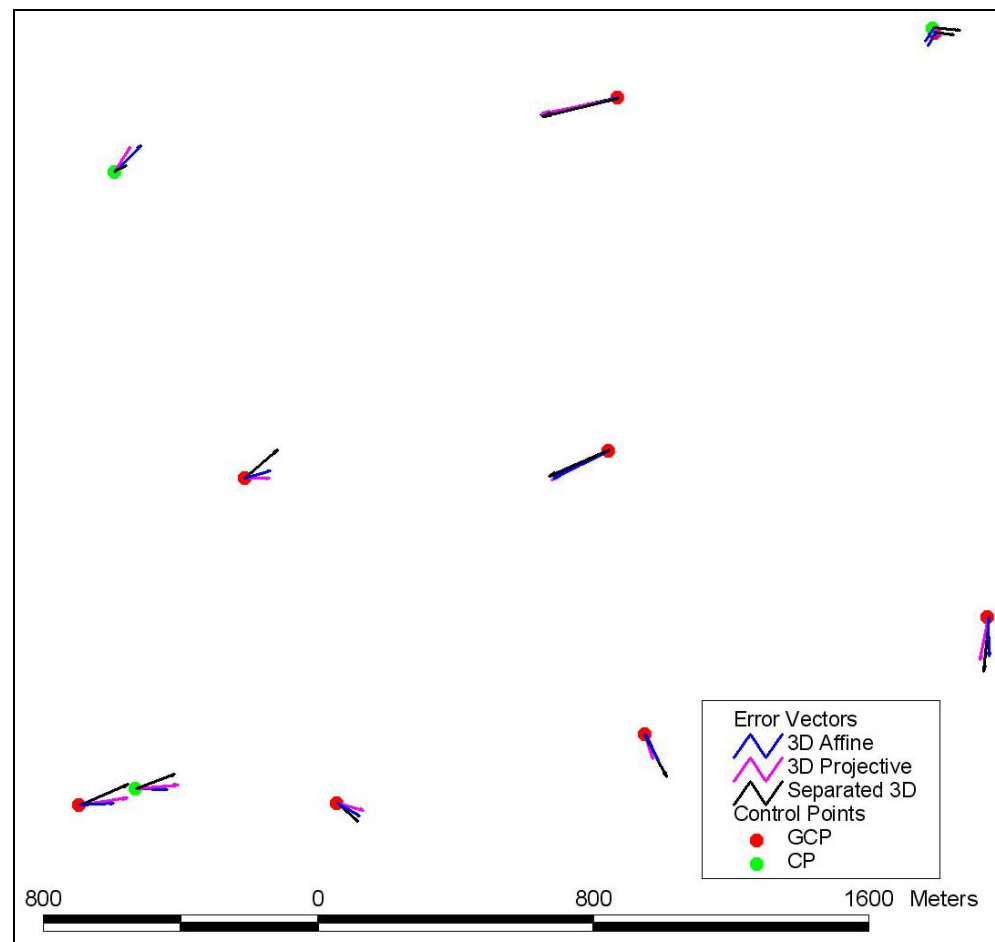


Figure 5.15: Error vector of GCP and CP (5000 times magnification)

Error vectors showed that the perspective geometric models produced better accurate than parallel geometric model for narrow study area.

DISCUSSION

6.1 Result of Transformation Models

Three transformation models are observed in the study. At the same time that 3D affine is observed as parallel model, 3D projective model and separated 3D projective model are observed as perspective transformation models. In this section, conjugated GCP are interpreted visual from subset (4000 x 3000 pixels) PRISM nadir image and IKONOS image. The transformation results are presented in the table (*table 3.3*). In this step, 10 GCP and 3 CP are used to register the raw image. The control points are pickup interactively from both of reference and target image.

In this transformation models, the study used 10 well distributed GCP which are collected visually from conjugated images. On the other hand, the check points are used very precise VRS-GPS control points.

The result shows less than 5 meters in all models. Moreover, the parallel transformation model produced global accuracy improvement while the perspective models are giving more accurate response around control points.

6.2 Error trends based on the distribution of GCP

A mount four selected patterns (*figure 4.11*) Pattern 1, well distributed control points, had more precision consistency than others for both of parallel and perspective transformations. Pattern 2, GCPs from non-elevated area, precision consistency is high around GCPs when CPs has low precision. Pattern 3, narrow distribution GCPs, produced high accuracy around GCPs. Unexpectedly; 3D affine transformation had more potential around CPs. Pattern 4, GCPs from lower part of selected area, issued low precision for some GCP located in other part with elevated area.

Apart from this, the reference IKONOS image may be intruded by original errors. Accuracy may be increased whilst the accurate original reference image is considered.

Geometric transformation results for three models are listed in the tables (*table 4.1 and table 4.2*) for four selected pattern. RMS errors around the GCP show perspective models with high accuracy.

Parallel geometric model, 3D affine, produced less RMS errors around the CP. The errors show consistency of image global accuracy.

6.3 Result of image matching

ICP accuracy is improved when artistic structures are selected; though, several ICP are collected from simulated reference image. Selected ICP are well distributed in the study area and the number ICP is minimized to 10 based on the window sizes and matching quality (*table 6.1*).

The least squares matching result values are presented in the table and the selected 10 image control points are listed with LSM values (*table 5.2*).

Each ICP result RMS errors for each transformation model are presented in the table (*table 5.3*).

Table 6.1: Selection of widow size

Window Size	ICP 1	ICP 2	ICP 3	ICP 4	ICP 5	ICP 6	ICP 7	ICP 8	ICP 9	ICP 10	Registered
51 x 51	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	
55 x 55	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	
61 x 61	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	
65 x 65	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	
71 x 71	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	
75 x 75	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	
81 x 81		LSM				LSM			LSM	LSM	
85 x 85	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	OK
91 x 91	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	OK
95 x 95	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	OK
101x101	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	OK
105x105				LSM		LSM			LSM		
111x111	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	OK
115x115	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	OK
121x121	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	OK
125x125	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	LSM	
131x131							LSM				
135x135		LSM					LSM				
141x141							LSM		LSM		
145x145	LSM			LSM			LSM		LSM		
151x151	LSM					LSM		LSM	LSM		

	Least Squares Matching impossible
	Registration impossible by LSM result
	Registration impossible by big errors
LSM	Least Squares Matching
OK	Registration

6.4 Result of image registration

The results of three different GCP are compares using different transformation models. Amount three GCP datasets the GCP collected by visual interpretation produced the worst; at the same time that the simulated GCP produced similar result with VRS-GPS GCP (table 6.1). Based on the result, the finding is applicable as high resolution satellite image registration. Simulated GCP (image control points) could be used as very precise GCP.

Table 6.2:GCP/CP RMSE comparison between visual collection, simulation and VRS-GPS

Unit: Pixel in pixel	Visual Interpretation				VRS-GPS				Simulated GCP IKONOS + DSM			
	GCP		CP		GCP		CP		GCP		CP	
	RMS X	RMS Y	RMS X	RMS Y	RMS X	RMS Y	RMS X	RMS Y	RMS X	RMS Y	RMS X	RMS Y
3D Affine	1.63	1.88	1.20	1.05	1.47	1.82	0.95	1.06	1.47	1.69	1.09	0.96
3D Projective	1.47	1.25	1.80	1.04	1.17	1.15	0.92	1.08	1.35	1.06	1.47	0.61
Separated 3D Projective	1.34	1.25	1.44	1.19	0.92	1.07	0.65	1.33	1.16	1.07	0.68	0.71

In the comparison result, perspective transformation model demonstrated that the error is contaminated in original reference data. The observation confirmed the potential of ICP referring to previous image matching result (RMS XY 7.2 meter) between image to image (Kersten et al., 2000; Gruen, 2000).

The proposed study is demonstrated four important factors of satellite image registration using differently observed GCP. The required GCP are observed from three conditions:

- (1) GCP acquired by image simulation (red color in graphs),
- (2) GCP collected visual interpretation (blue color in graphs) and
- (3) Measured GCP using VRS-GPS.

The study demonstrated the possibility new idea on using simulated ICP as automated collected GCP. The result shows there is not big difference between VRS-GPS GCP and simulated-ICP GCP, whilst the GCP collected by visual interpretation produced differences.

On the other hand, these data are applied to investigate transformation models (3D affine, 3D projective and separated 3D projective). The transformation root mean squares errors presented that parallel transformation model (3D affine) has most stability in overall area of the image. This condition is confirmed by the less RMS error around the check points (CPs). In the case of perspective transformation models (3D projective and separated 3D projective) the RMS

error around the CP presented instability; however, there is very high accuracy around the GCP.

Thus, the study demonstrated that the potential of GCP selection method. Moreover, the result could be suggestion on the selection of transformation model for satellite sensor.

The study observed the use of simulated ICP as of GCP with three geometric transformation models follow by investigating the result using error propagated vector.

Radiometric and geometric differences on ordinary satellite images are the major difficulties in image matching for acquiring conjugated points. The proposed algorithm demonstrated a possibility to use simulated ICP. A reference image is simulated giving orbital acquisition parameters of satellite sensors to very high resolution DSM considering geometric differences elimination while radiometric differences are generated from very high resolution multispectral image. A problem found in the step of reference image simulation that is the pre-geocoded between DSM and multispectral image. It could be a factor of accuracy reduction in this research. These DSM and multispectral image should be accurately co-registered before generating reference image. As suggestion for future research, those two datasets should be product of the same sensor. This problem can define as original error in the reference dataset.

A panchromatic spectral is generated from Red band and green band of reference multispectral image based on the similarity between these two bands and target panchromatic image spectral range. Once another satellite image is uses as target image the part of calculation of panchromatic band generation should be changed.

6.5 Role of Research

The research demonstrated that the solution of major problems in satellite remote sensing in image registration. After the problems are solved, the role the research draws possibility of automatic GCP selection and automatic satellite image registration with high accuracy.

Moreover future role of this study can be oriented to pattern recognition and automatic target recognition in robotic industry by well understands upon simulated 3D image control point.

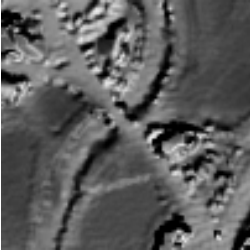
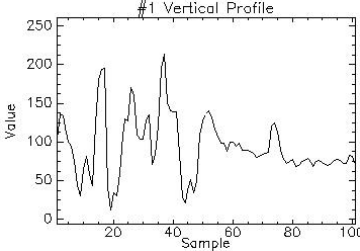
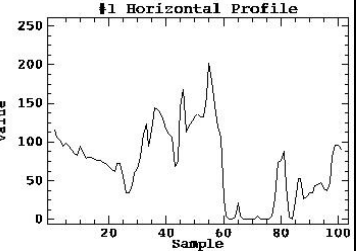
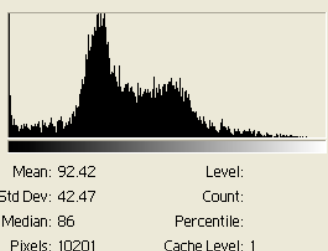
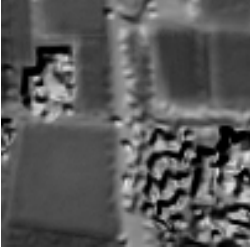
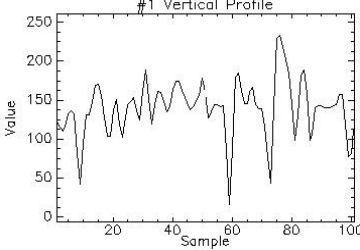
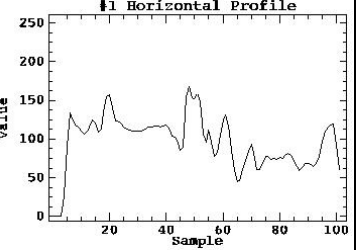
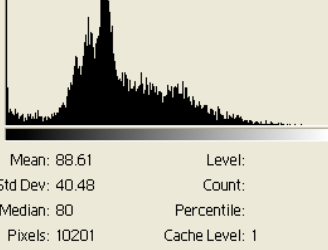

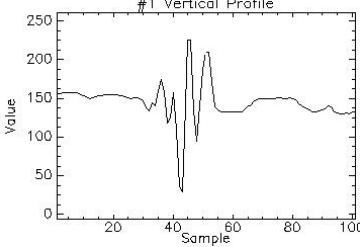
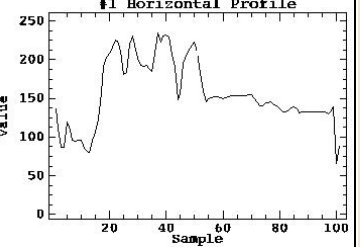
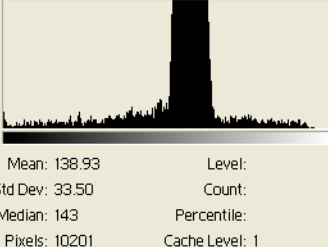
Another study on ICP which investigated the ground features. In this study, 10 features are selected as ICPs (template patches) from a simulated reference image including building, road crossing, bridge and dike to cover investigation on all type of artistic structures. LSM generated values, IKONOS geocoded values, VRS-GPS survey values initial values and deviation of LSM from initial values are listed (*table 6.2*).

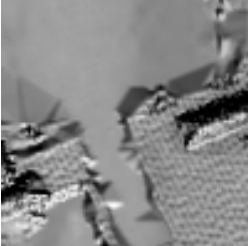
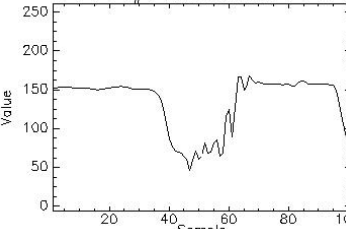
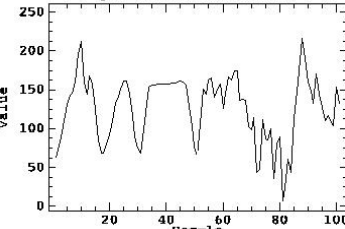
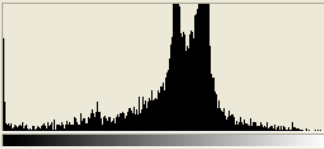
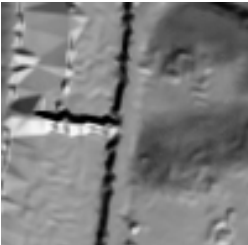
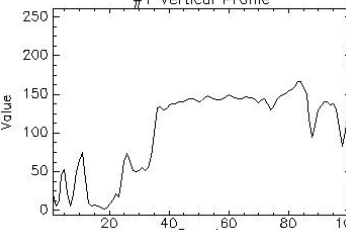
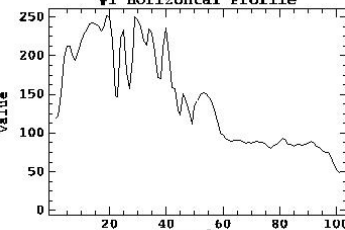
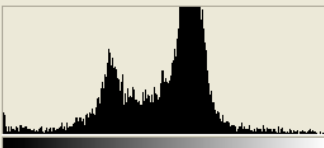
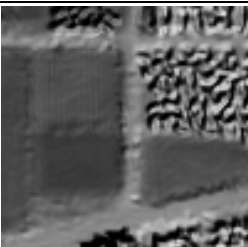
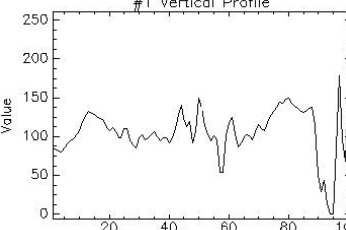
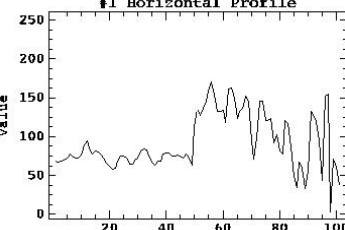
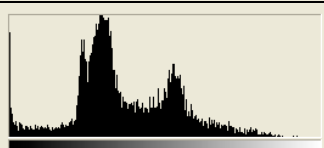
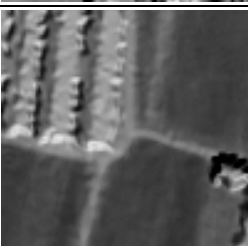
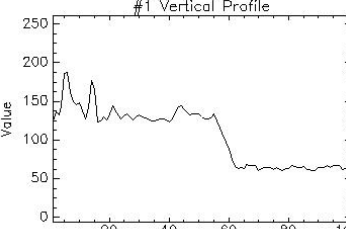
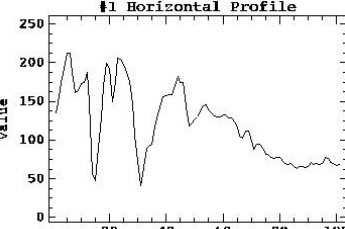
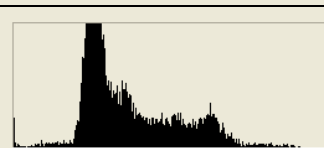
Histogram, horizontal profile and vertical profile of each feature are compared for target image subsets (*table 6.4*) and for ICPs (*table 6.3*). Finally, selected ICPs are compared using the relationship between window sizes and residual (*table 7.4*).

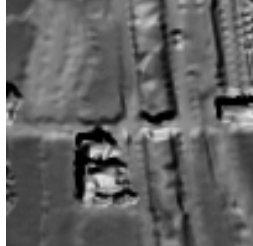
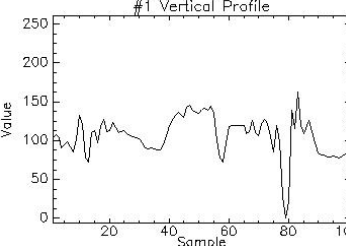
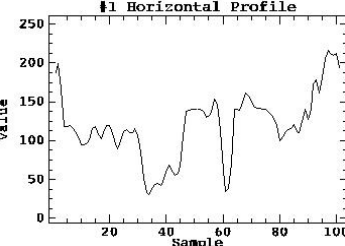
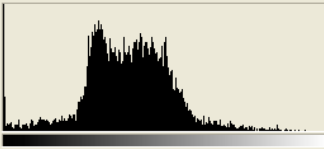
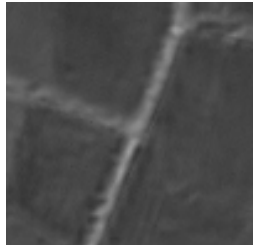
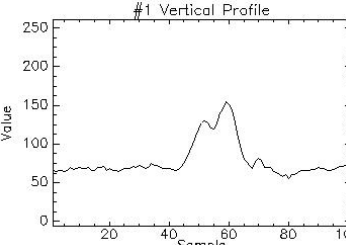
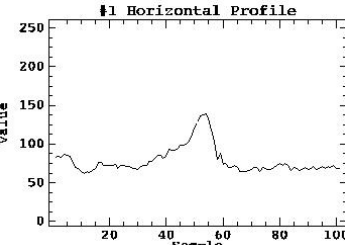
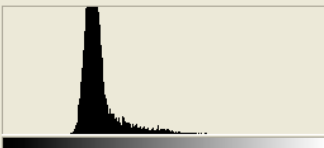

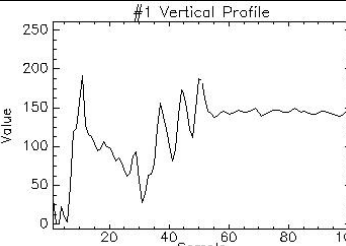
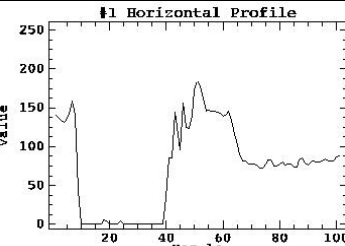
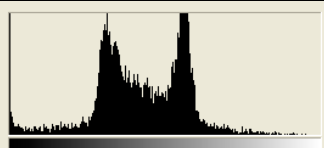
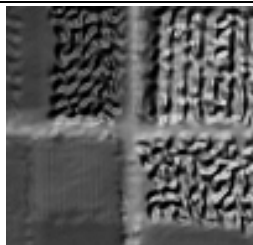
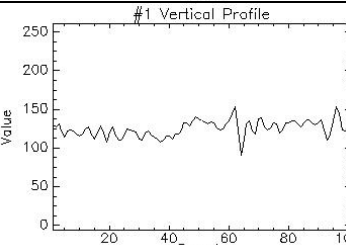
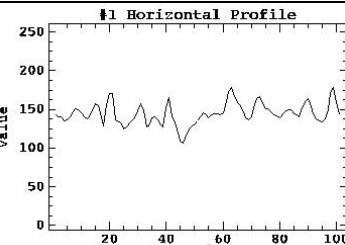
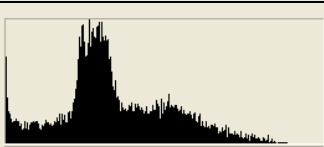
Table 6.3: Observed ICPs, LSM results and deviation of LSM results from initial values

GCP/CP	LSM ICP		Center of Image Patch			VRS-GPS			Initial Value		Deviation (LSM-Initial)	
	U	V	X	Y	Z	X	Y	Z	U	V	ΔU	ΔV
GCP 1	2294.218082	876.447231	18808.5	64034.5	31.181	18802.870	64035.018	68.269	2293	874	1.218082	2.447231
GCP 2	2267.229992	1899.832932	18169.5	61540.5	12.877	18167.078	61539.723	50.197	2268	1900	-0.770008	-0.167068
GCP 3	2370.935694	2732.387915	17966.5	59460.5	1.421	17965.617	59460.534	1.882	2372	2732	-1.064306	0.387915
GCP 4	3370.599162	2389.322119	20586.5	59736.5	1.091	20586.083	59736.805	2.152	3371	2389	-0.400838	0.322119
GCP 5	1475.983667	2929.657740	15676.5	59471.5	3.677	15676.354	59471.203	4.303	1479	2933	-3.016333	-3.342260
GCP 6	3211.878837	684.335257	21157.5	63976.5	15.027	21157.932	63976.774	16.278	3214	685	-2.121163	-0.664743
GCP 7	3558.467598	1952.631823	21289.5	60691.5	2.842	21289.686	60691.948	3.159	3558	1957	0.467598	-4.368177
GCP 8	1210.492061	1984.075516	15550.5	61937.5	8.648	15550.528	61937.336	8.822	1210	1983	0.492061	1.075516
GCP 9	731.922626	2934.947463	13848.5	59876.5	3.245	13848.156	59876.205	3.586	732	2935	-0.077374	-0.052537
GCP 10	307.307190	1159.360909	13807.5	64449.5	16.094	13807.500	64446.466	16.391	306	1161	1.307190	-1.639091
CP 1	3211.315858	671.805769	20720.5	63015.5	8.751	21158.34	64012.91	17.282	3211	671	0.315858	0.805769
CP 2	898.03302	2887.606037	14277.5	59899.5	4.72	14274.66	59902.51	4.101	895	2888	3.033020	-0.393963
CP 3	816.110404	1089.628904	15121.5	64322.5	15.781	15125.41	64325.32	15.988	830	1091	-13.88959	-1.371096

Table 6.4: Observed ICPs with vertical profiles, horizontal profiles and histograms

No	Simulated ICP	Vertical profile	Horizontal profile	Histogram
ICP 1				 Mean: 92.42 Level: Std Dev: 42.47 Count: Median: 86 Percentile: Pixels: 10201 Cache Level: 1
ICP 2				 Mean: 88.61 Level: Std Dev: 40.48 Count: Median: 80 Percentile: Pixels: 10201 Cache Level: 1
ICP 3				 Mean: 138.93 Level: Std Dev: 33.50 Count: Median: 143 Percentile: Pixels: 10201 Cache Level: 1

ICP4		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 135.89 Level: Std Dev: 38.49 Count: Median: 145 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP5		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 126.11 Level: Std Dev: 38.89 Count: Median: 139 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP6		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 94.11 Level: Std Dev: 42.60 Count: Median: 83 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP7		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 90.33 Level: Std Dev: 38.27 Count: Median: 74 Percentile: Pixels: 10201 Cache Level: 1</p>

ICP8				 <p> Mean: 97.46 Level: Std Dev: 35.93 Count: Median: 98 Percentile: Pixels: 10201 Cache Level: 1 </p>
ICP9				 <p> Mean: 74.67 Level: Std Dev: 13.78 Count: Median: 71 Percentile: Pixels: 10201 Cache Level: 1 </p>
ICP10				 <p> Mean: 106.56 Level: Std Dev: 40.41 Count: Median: 107 Percentile: Pixels: 10201 Cache Level: 1 </p>
CP 1				 <p> Mean: 87.75 Level: Std Dev: 44.87 Count: Median: 79 Percentile: Pixels: 10201 Cache Level: 1 </p>


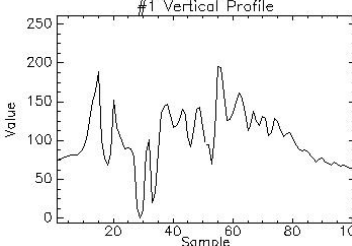
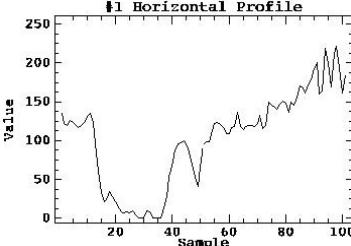
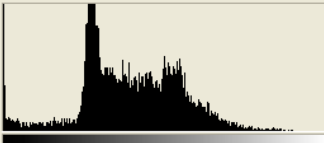
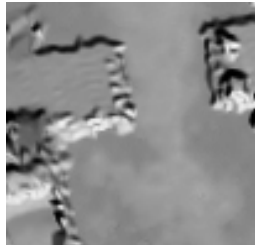
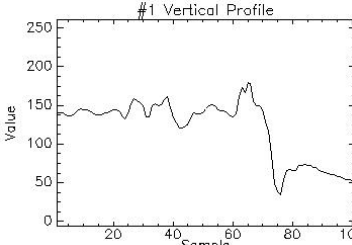
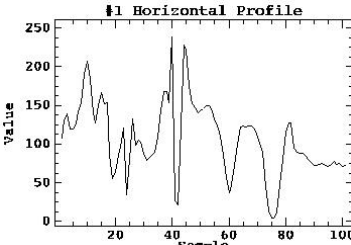
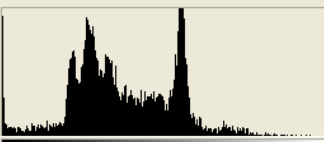

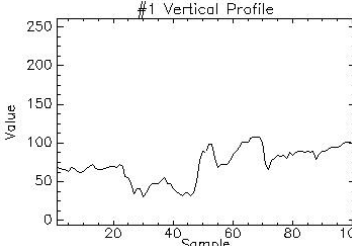
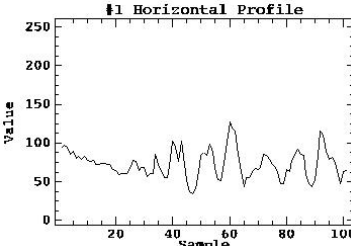
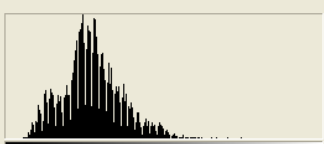

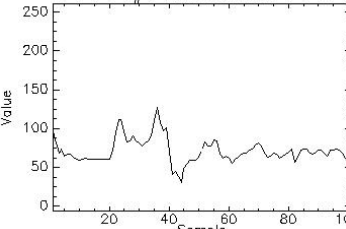
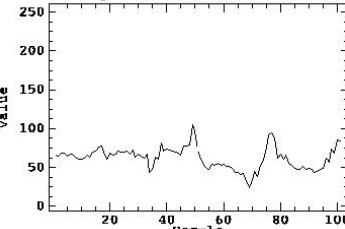
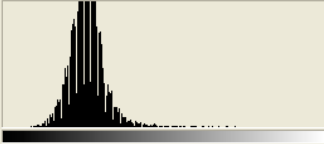

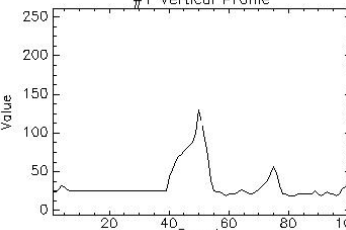
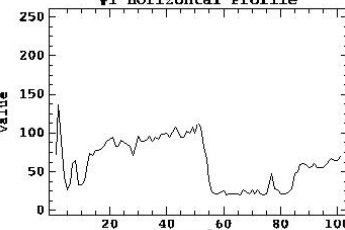
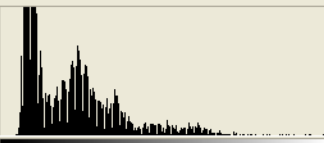

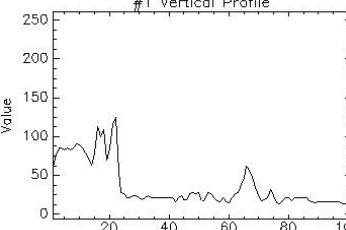
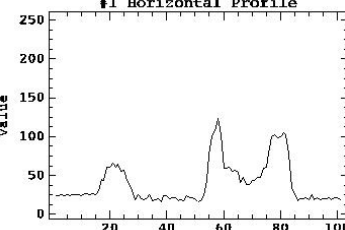
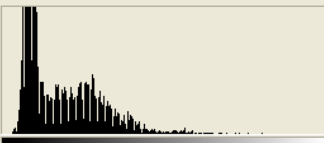

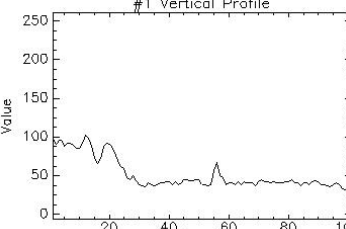
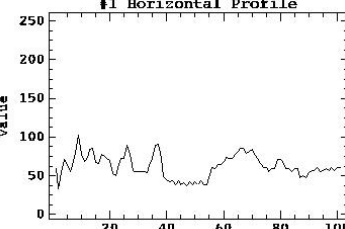
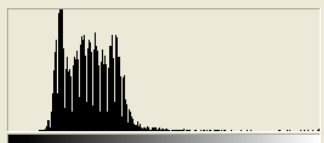

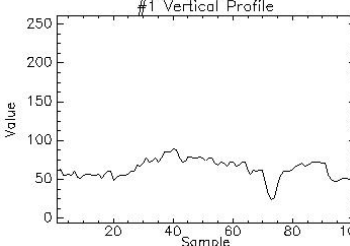
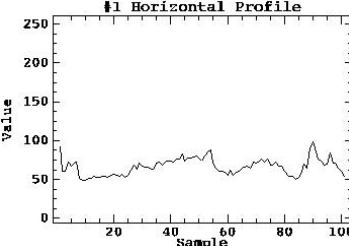
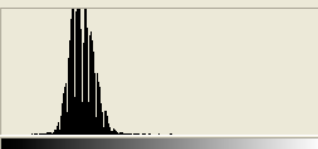

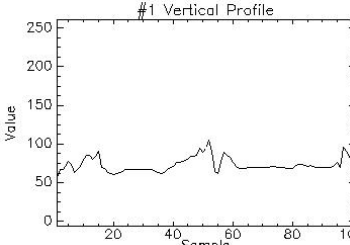
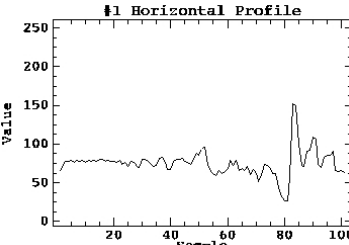
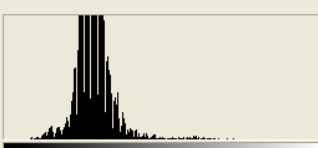

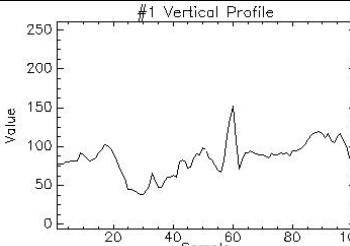
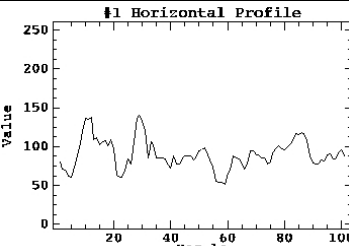
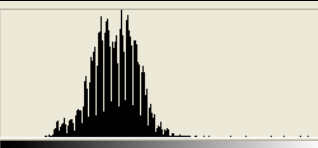
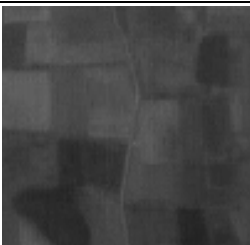
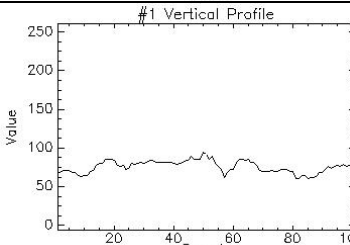
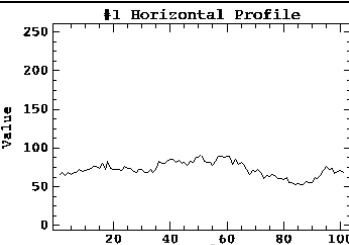
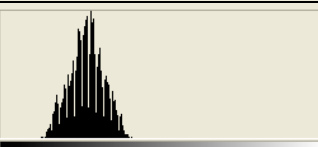
CP 2		 <p>#1 Vertical Profile</p>	 <p>#1 Horizontal Profile</p>	 <p>Mean: 97.84 Level: Std Dev: 41.47 Count: Median: 95 Percentile: Pixels: 10201 Cache Level: 1</p>
CP 3		 <p>#1 Vertical Profile</p>	 <p>#1 Horizontal Profile</p>	 <p>Mean: 94.91 Level: Std Dev: 41.16 Count: Median: 88 Percentile: Pixels: 10201 Cache Level: 1</p>

Table 6.4: Subsets of PRISM image with horizontal profiles, vertical profiles and histograms

No	PRISM Image	Vertical profile	Horizontal profile	Histogram
ICP 1		 <p>#1 Vertical Profile</p>	 <p>#1 Horizontal Profile</p>	 <p>Mean: 69.38 Level: Std Dev: 23.83 Count: Median: 68 Percentile: Pixels: 10201 Cache Level: 1</p>

ICP2		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 66.45 Level: Std Dev: 14.12 Count: Median: 66 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP3		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 50.36 Level: Std Dev: 35.39 Count: Median: 36 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP4		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 43.32 Level: Std Dev: 29.09 Count: Median: 28 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP5		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 66.65 Level: Std Dev: 21.86 Count: Median: 65 Percentile: Pixels: 10201 Cache Level: 1</p>

ICP6		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 63.93 Level: Std Dev: 9.95 Count: Median: 63 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP7		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 71.77 Level: Std Dev: 15.90 Count: Median: 71 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP8		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 89.53 Level: Std Dev: 18.25 Count: Median: 90 Percentile: Pixels: 10201 Cache Level: 1</p>
ICP9		<p>#1 Vertical Profile</p> 	<p>#1 Horizontal Profile</p> 	 <p>Mean: 67.56 Level: Std Dev: 13.35 Count: Median: 68 Percentile: Pixels: 10201 Cache Level: 1</p>


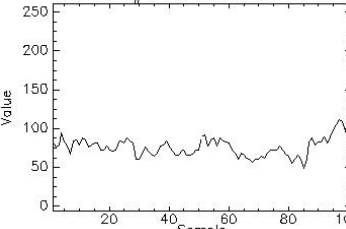
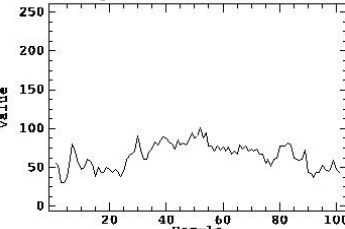
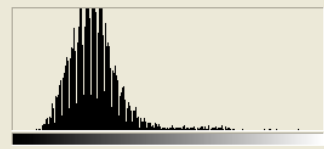
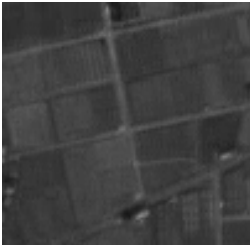
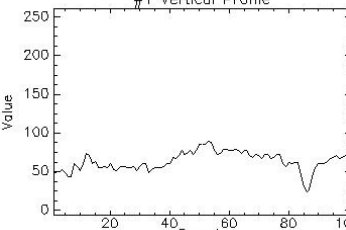
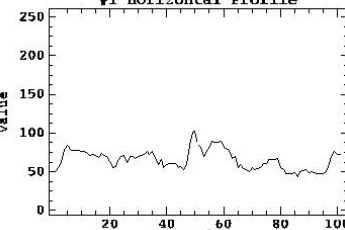
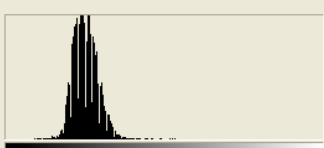

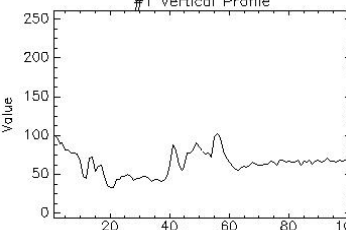
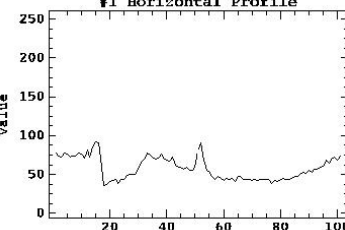
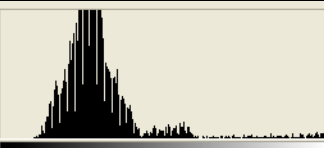
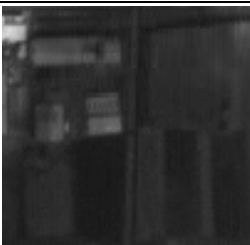
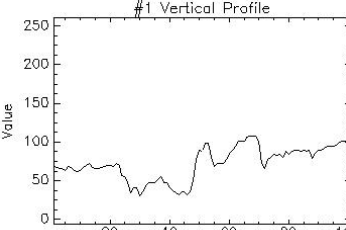
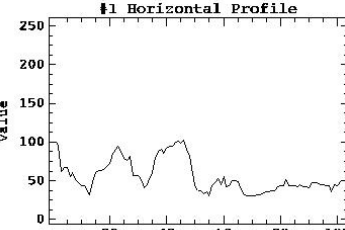
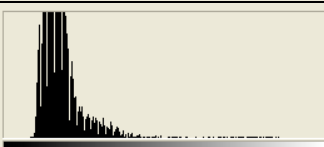
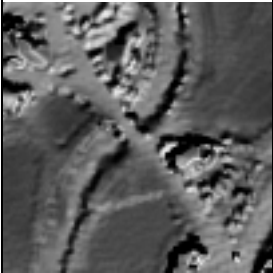
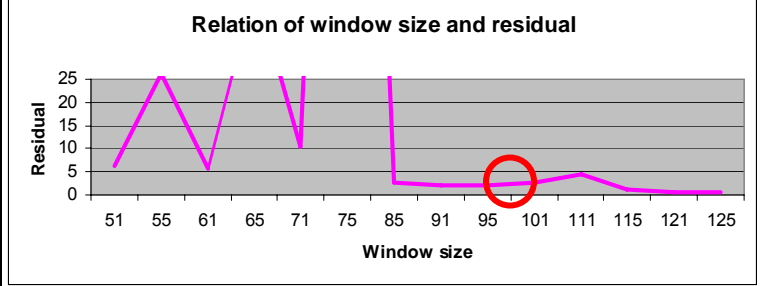
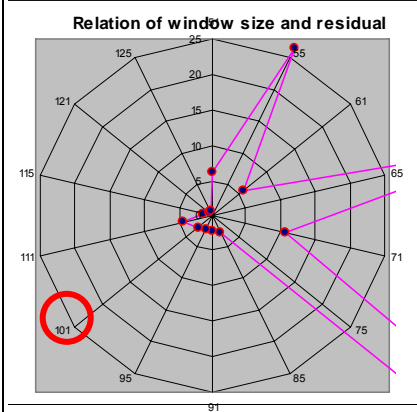
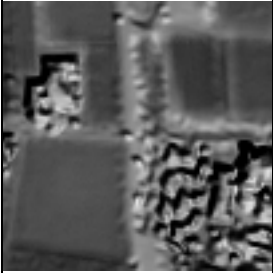
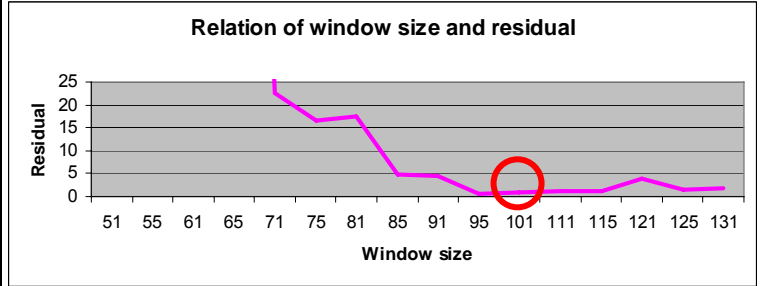
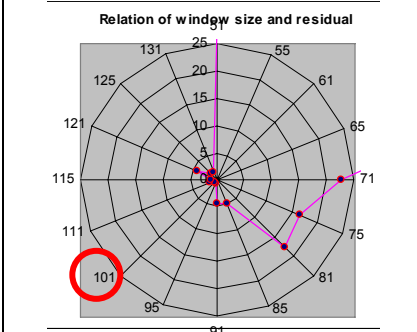
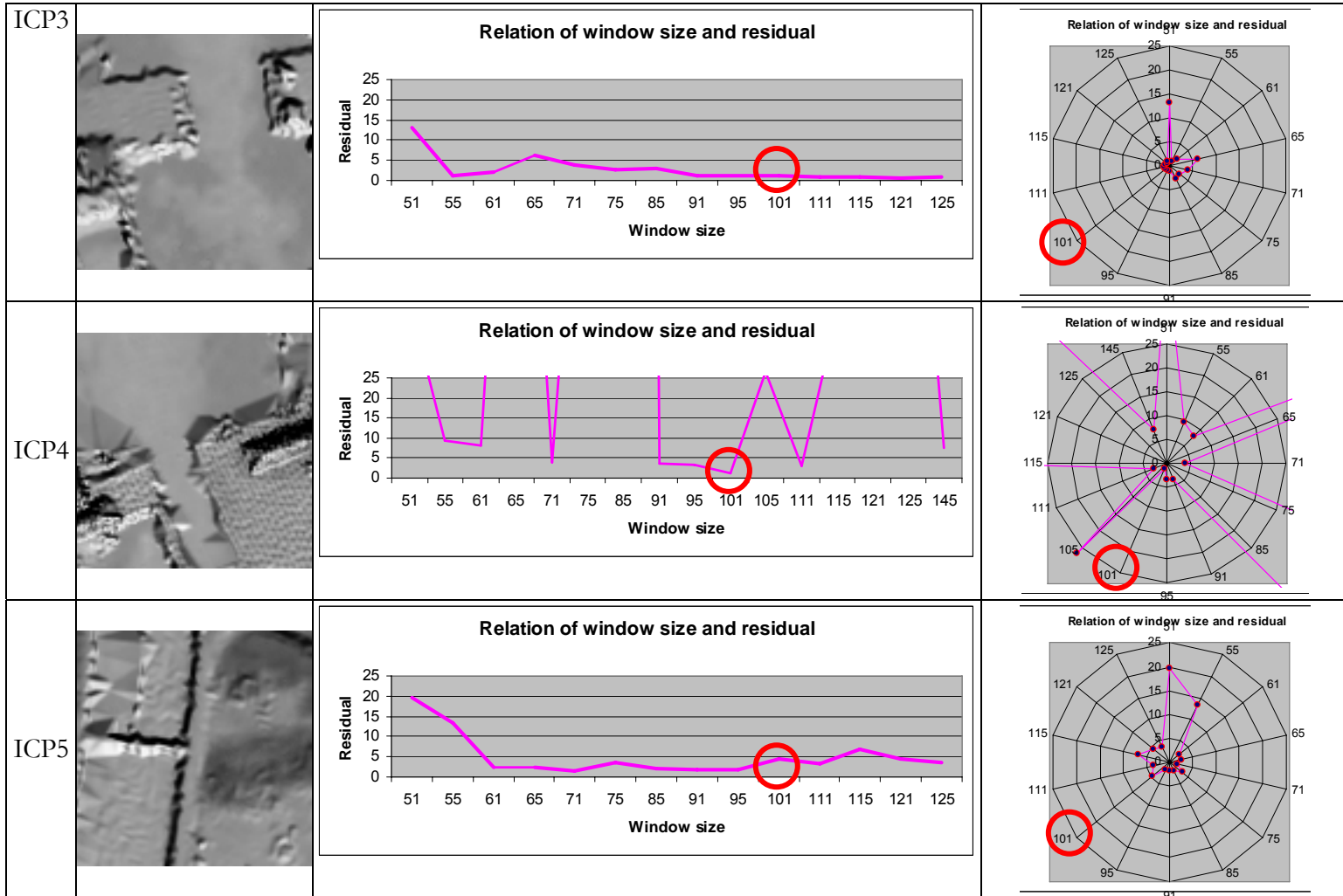
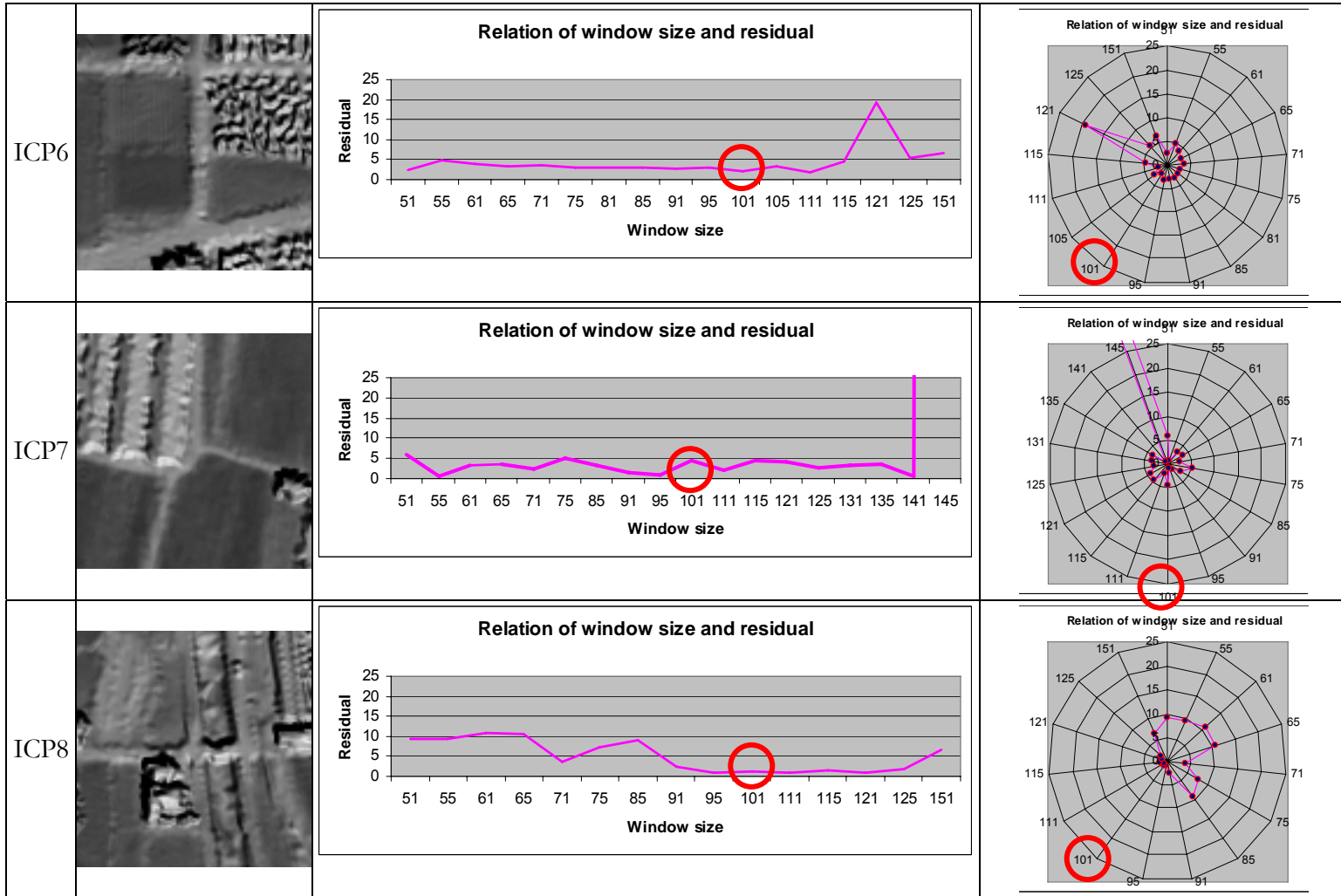
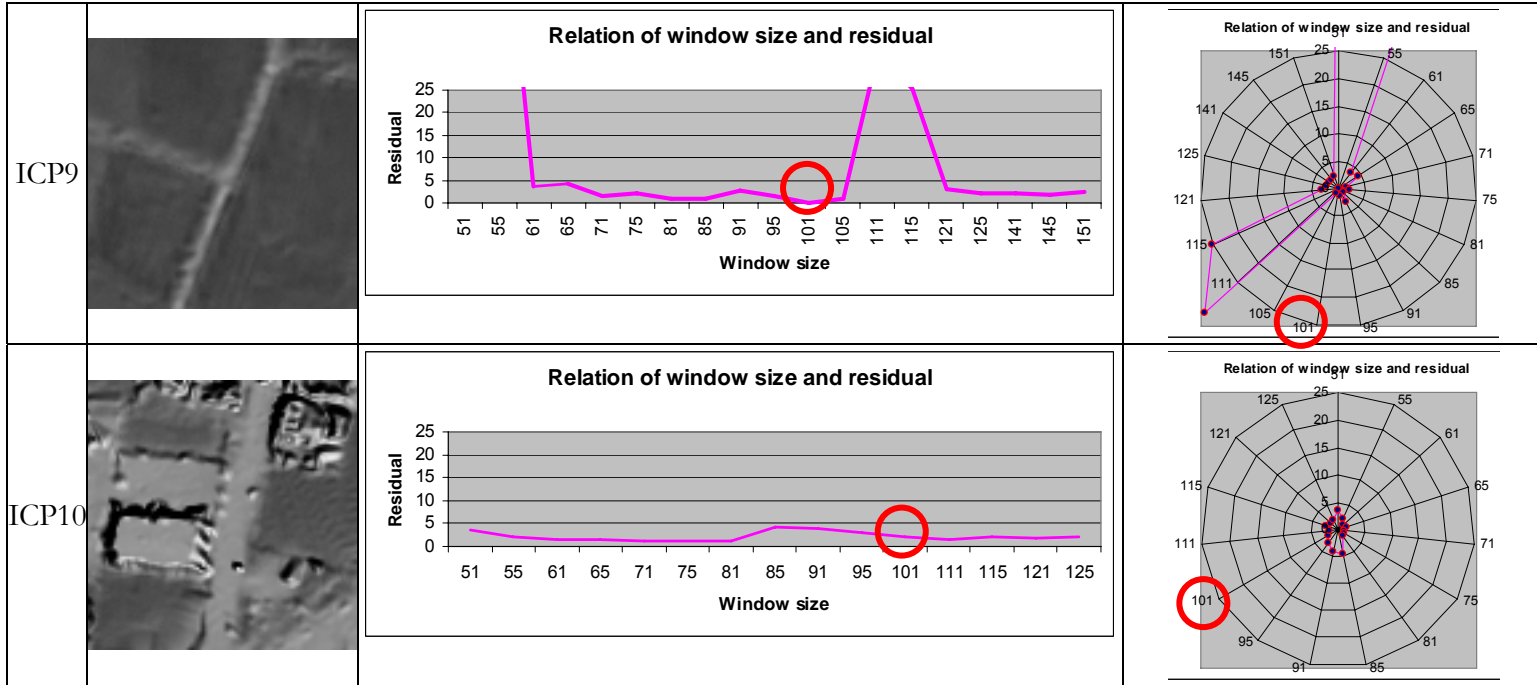
ICP10		 <p>#1 Vertical Profile</p>	 <p>#1 Horizontal Profile</p>	 <p>Mean: 67.37 Level: Std Dev: 22.34 Count: Median: 65 Percentile: Pixels: 10201 Cache Level: 1</p>
CP 1		 <p>#1 Vertical Profile</p>	 <p>#1 Horizontal Profile</p>	 <p>Mean: 63.94 Level: Std Dev: 10.26 Count: Median: 63 Percentile: Pixels: 10201 Cache Level: 1</p>
CP 2		 <p>#1 Vertical Profile</p>	 <p>#1 Horizontal Profile</p>	 <p>Mean: 85.41 Level: Std Dev: 49.53 Count: Median: 71 Percentile: Pixels: 10201 Cache Level: 1</p>
CP 3		 <p>#1 Vertical Profile</p>	 <p>#1 Horizontal Profile</p>	 <p>Mean: 45.82 Level: Std Dev: 18.19 Count: Median: 43 Percentile: Pixels: 10201 Cache Level: 1</p>

Table 6.5: Relationship of residual and window sizes and selected ICP 101x 101 windows ICP

No	Simulated ICP	Search Window Sizes	Search Window Sizes (in other charts)
ICP 1		<p style="text-align: center;">Relation of window size and residual</p> 	<p style="text-align: center;">Relation of window size and residual</p> 
ICP2		<p style="text-align: center;">Relation of window size and residual</p> 	<p style="text-align: center;">Relation of window size and residual</p> 







Even though ICP matching procedure was done on Nadir image of ALOS-PRISM, off-nadir images also have high possibility to apply the algorithm.

Research presented that simulated ICP can apply in various type of high resolution satellite image registration with less expense in GCP selection while GCP collection is cost in conventional survey. The GCP can be the roof of buildings or the top of vegetations while it is impossible to ground true in GPS survey.

In the case of seasonal vegetation changes, this study demonstrated that the related spectral pattern can integrate to image simulation depend on the season of image acquisition. For this purpose, a database for seasonal vegetation spectral patterns should develop in advance. Recent developments between two acquisition times can reduce matching accuracy; on the other hand, the changes such as development of infrastructure should list as achieve information to accommodate in selection of ICP location.

Next, the study demonstrated that spectral reflectance differences on satellite images by different sensors can be adjusted by band ratios averaging. Thus, different satellite images can use a common set reference data.

Possibility of using Image Control Points is demonstrated. Registration accuracy will be improved when we used accurate ICP. Less number of GCP is possible when we have very high accurate GCP. 3D projective model is more accurate than 3D affine for ALOS PRISM L1B2 data with narrow extent (8 km x 6 km).

Chapter 7

FUTURE WORKS

As future work, the study will continue to investigate transformation models and consideration of error trends to conclude possibility of accurate transformation model in high resolution satellite images registration using other sensor images.

As we found that it is require developing season spectral reflectance database, the spectral database will develop in future. Similar to GCP database, the spectral data can also be distributed through internet as a contribution to remote sensing, GIS and photogrammetric communities.

Finally, A ICP database can develop for all kind of satellite images registration, while the places are difficult to do ground true measurement with GPS.

Appendix A

GEOMETRIC TRANSFORMATION MODEL CODE (EXAMPLE: 3D AFFINE)

The coding is developed with c-compiler in Mac OS. If Windows platform is used to compile the program, cygwin compiler is commended.

Program could be compiled simply in the cygwin using following command.

```
..... $ affine3D_err.c -o affine3D_err.exe
```

Then, the (executable) application file affine3D_err.exe will be output to the same directory.

The user can calculate the geometric transformation with the executable file and two other text files (GCP.txt and CP.txt) using following command.

```
..... $ ./affine3D_err.exe GCP.txt CP.txt output.txt
```

```
=====/affine3D_err.c/=====  
#include <stdio.h>  
#include <stdlib.h>  
#include <math.h>  
  
#define SIZE_OF_COE 9  
#define SIZE_OF_DATA 100  
  
void read_coordinates_file(char *filename, double **org, int  
*n, double *u_data, double *v_data, double *x_data, double  
*y_data, double *z_data, double *a_org)  
{  
  
FILE *fp;  
fp = fopen(filename, "r");  
  
if(fp == NULL){  
fprintf(stderr, "Error :File couldn't read . File name : %s  
\n", filename);  
exit(-1);  
}  
  
int i, j;  
for(i = 0; i < SIZE_OF_COE; i++){
```

```

    for(j = 0; j < SIZE_OF_COE; j++){
        org[i][j] = 0;
    }
}

for(i = 0; i < SIZE_OF_COE - 1; i++){
    a_org[i] = 0;
}

double x, y, z, u, v;
while(fscanf(fp, "%d %lf %lf %lf %lf %lf", &i, &u, &v, &x, &y,
&z) > 0){
    u_data[(int)(*n)] = u;
    v_data[(int)(*n)] = v;
    x_data[(int)(*n)] = x;
    y_data[(int)(*n)] = y;
    z_data[(int)(*n)] = z;

    *n += 1;

    org[0][0] += x * x;
    org[0][1] += x * y;
    org[0][2] += x * z;
    org[0][3] += x;

    org[1][0] += x * y;
    org[1][1] += y * y;
    org[1][2] += y * z;
    org[1][3] += y;

    org[2][0] += x * z;
    org[2][1] += y * z;
    org[2][2] += z * z;
    org[2][3] += z;

    org[3][0] += x;
    org[3][1] += y;
    org[3][2] += z;
    org[3][3] += 1;

    org[4][4] += x * x;
    org[4][5] += x * y;
    org[4][6] += x * z;
    org[4][7] += x;

    org[5][4] += x * y;
    org[5][5] += y * y;
    org[5][6] += y * z;
    org[5][7] += y;

    org[6][4] += x * z;

```

```

        org[6][5] += y * z;
        org[6][6] += z * z;
        org[6][7] += z;

    org[7][4] += x;
    org[7][5] += y;
    org[7][6] += z;
    org[7][7] += 1;

    a_org[0] += x * u;
    a_org[1] += y * u;
    a_org[2] += z * u;
    a_org[3] += u;
    a_org[4] += x * v;
    a_org[5] += y * v;
    a_org[6] += z * v;
    a_org[7] += v;

    }
    fclose(fp);
}

void set_array(double **org, double *tmp_org){

    org[0][8] = tmp_org[0];
    org[1][8] = tmp_org[1];
    org[2][8] = tmp_org[2];
    org[3][8] = tmp_org[3];
    org[4][8] = tmp_org[4];
    org[5][8] = tmp_org[5];
    org[6][8] = tmp_org[6];
    org[7][8] = tmp_org[7];
}

void cal_gauss(int num, double **src, double *dst)
{
    int i, j, k;
    double tmp[num + 1][num + 1];
    double mm;

    for(i = 0; i < num; i++){
        for(j = 0; j < num + 1; j++){
            tmp[i][j] = src[i][j];
        }
    }

    for(i = 0; i < num; i++){
        for(j = 0; j < num; j++){
            if(i != j){
                mm = tmp[j][i] / tmp[i][i];
                for(k = 0; k < num + 1; k++)

```

```

        tmp[j][k] -= tmp[i][k] * mm;
    }
}

for(i = 0; i < num; i++){
    dst[i] = tmp[i][num] / tmp[i][i];
}
}

void print_rmse(double *a_dst, double *u_data, double *v_data,
double *x_data,
double *y_data, double *z_data, int n){
    double d_u = 0, d_v = 0, u = 0, v = 0, d_uz = 0, d_vz = 0,
uz = 0, vz = 0;
    int i;
    for(i = 0; i < n; i++){
        d_u = (a_dst[0] * x_data[i] + a_dst[1] * y_data[i] +
a_dst[2] * z_data[i] + a_dst[3]);
        d_v = (a_dst[4] * x_data[i] + a_dst[5] * y_data[i] +
a_dst[6] * z_data[i] + a_dst[7]);

        u += pow(u_data[i] - d_u, 2);
        v += pow(v_data[i] - d_v, 2);

        fprintf(stdout, "[%d] -> %lf, %lf, %lf, %lf\n", i, d_u -
u_data[i], d_v - v_data[i], d_u, d_v);
    }
    fprintf(stdout, "Number of point : %d\n", n);
    fprintf(stdout, "RMSE : u -> %6.9lf\n", sqrt(u / n));
    fprintf(stdout, "RMSE : v -> %6.9lf\n", sqrt(v / n));
}

void read_cp_file(char *filename, int *n, double *u, double *v,
double *x, double *y, double *z)
{
    FILE *fp = fopen(filename, "r");
    int i = 0, gomi;
    while(fscanf(fp, "%d %lf %lf %lf %lf %lf", &gomi, &u[i],
&v[i], &x[i], &y[i], &z[i]) > 0){
        i++;
    }
    fclose(fp);
    *n = i;
}

void print(double *a_dst, char *filename){
    FILE *fpo = fopen(filename, "w");
    fprintf(fpo, "%6.13lf\n", a_dst[0]);
    fprintf(fpo, "%6.13lf\n", a_dst[1]);
    fprintf(fpo, "%6.13lf\n", a_dst[2]);
}

```

```

    fprintf(fpo, "%6.13lf\n", a_dst[3]);
    fprintf(fpo, "%6.13lf\n", a_dst[4]);
    fprintf(fpo, "%6.13lf\n", a_dst[5]);
    fprintf(fpo, "%6.13lf\n", a_dst[6]);
    fprintf(fpo, "%6.13lf\n", a_dst[7]);
    fclose(fpo);
}

int main(int argc, char *argv[]){
    if(argc < 2){
        fprintf(stderr, "./projection [GCP.text] [CP.text] [out
Coefficient.text]\n");
        return -1;
    }

    double **org = (double **)malloc(sizeof(double) *
SIZE_OF_COE * SIZE_OF_COE);
    int i;
    for(i = 0 ; i < SIZE_OF_COE; i++){
        org[i] = (double *)malloc(sizeof(double) * SIZE_OF_COE);
    }

    double *a_org = (double *)malloc(sizeof(double) *
(SIZE_OF_COE - 1));
    double *a_dst = (double *)malloc(sizeof(double) *
(SIZE_OF_COE - 1));

    double *u_data = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *v_data = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *x_data = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *y_data = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *z_data = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);

    int n = 0;

    fprintf(stdout, "3D Affine Transformation\n");
    read_coordinates_file(argv[1], org, &n, u_data, v_data,
x_data, y_data, z_data, a_org);
    set_array(org, a_org);
    cal_gauss(SIZE_OF_COE - 1, org, a_dst);
    fprintf(stdout, "-----\n");
    fprintf(stdout, "Residual Error of around GCP\n");
    print_rmse(a_dst, u_data, v_data, x_data, y_data, z_data,
n);

    if(argc == 2) return 0;
}

```

```

    double *check_x = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *check_y = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *check_z = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *check_u = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    double *check_v = (double *)malloc(sizeof(double) *
SIZE_OF_DATA);
    read_cp_file(argv[2], &n, check_u, check_v, check_x, check_y,
check_z);

    fprintf(stdout, "-----\n");
    fprintf(stdout, "Residual Error of around CP\n");
    print_rmse(a_dst, check_u, check_v, check_x, check_y,
check_z, n);

    fprintf(stdout,
"%lf, %lf, %lf, %lf, %lf, %lf, %lf, %lf, %lf, %lf, %lf\n",
a_dst[0], a_dst[1], a_dst[2], a_dst[3],
a_dst[4], a_dst[5], a_dst[6], a_dst[7], a_dst[8], a_dst[9], a_dst[10]
, a_dst[11]);

    if(argc == 3) return 0;
    print(a_dst, argv[3]);
}

```

=====/END/====
Sample of Ground Control Points file

GCP.txt
1 2294.218082 876.447231 18808.5 64034.5 31.20
3 2267.229992 1899.832932 18169.5 61540.5 12.88
4 1398.898577 1357.396148 16390.5 63347.5 15.53
5 2370.935694 2732.387915 17966.5 59460.5 1.42
6 3124.349024 1099.349301 20720.5 63015.5 8.75
7 3370.599162 2389.322119 20586.5 59736.5 2.15
8 1475.983667 2929.657740 15676.5 59471.5 3.71

Sample of Check Points file

CP.txt
9 1824 2679 16657.5 59895.5 3.53
10 1731 2151 16727.5 61232.5 7.14
11 3303 1989 20648.5 60750.5 11.65
12 1031 2756 14682.5 60146.5 7.20
14 3016 2413 19707.5 59876.5 1.77

ERROR VECTOR GENERATION

Error vectors are generated using avenue script of ArcView GIS. The script is available on ESRI website. The script is developed by Jianping Hu and I would like to credit to him for his work on vector generation. The GCP and CP database (dbf file) should be included vector original (geometric transformation result values are used in this case) and vector end (error added geometric transformation values are used in this case). When user creates this dbf file, the file can be opened in ArcView windows. The script can be run from table menu by compiling it. Then, user need to follow the instruction.

```
=====Avenue Script (DispVector.AVE)=====
'DispVector.AVE
'Created by Jianping Hu, Aug 14, 2001
'
'This script must be run on a Table document. The output will be a polyline
shapefile.

'This script will read a table of points, and convert the two points to a straight line
with arrow in the
'end of the line. It is useful to convert ground deformation data point to
displacement vector.
'
'The input point table should be one the following three format
'I. Fields containing numeric data
' 1 x-coordinate of first point
' 2 y-coordinate of first point
' 3 x-coordinate of second point
' 4 y-coordinate of second point
' 5 a polyline identifier field (i.e. points in the same polyline must have the
same ID)
'II. Fields containing numeric data
' 1 x-coordinate of first point
' 2 y-coordinate of first point
' 3 a polyline identifier field (i.e. points in the same polyline must have the
same ID)
' 4 displacement value
' 5 displacement direction or aspect (degree)
'III. Fields containing numeric data
```

```
' 1 x-coordinate of first point
' 2 y-coordinate of first point
' 3 a polyline identifier field (i.e. points in the same polyline must have the
same ID)
' 4 displacement value
' for this type, you will ask to input the displacement direction as degree
'
```

"The displacement vector can be extended to certain scale by inputting scale factor

```
' Get the active table and choose the table type, then choose the related fields.
```

```
theVTab = av.GetActiveDoc.GetVTab
theFields = theVTab.GetFields
chosenfields={}
theType={"Start - End point", "Start point & Direction", "Start point"}
theAnalysisType=MsgBox.ListAsString(theType,"Data Type","Select the type")
if (theAnalysisType="Start - End point") then
    type=1
    X1Field = MsgBox.List(theFields, "Easting or X values of first point", "Select
X-coordinates field")
    if (X1Field=nil) then
        exit
    else
        chosenfields.add(X1Field)
    end
    Y1Field = MsgBox.List(theFields, "Northing or Y values of first point",
"Select Y-coordinates field")
    if (Y1Field=nil) then
        exit
    else
        chosenfields.add(Y1Field)
    end
    X2Field = MsgBox.List(theFields, "Easting or X values of second point",
"Select X-coordinates field")
    if (X2Field=nil) then
        exit
    else
        chosenfields.add(X2Field)
    end
    Y2Field = MsgBox.List(theFields, "Northing or Y values of second point",
"Select Y-coordinates field")
    if (Y2Field=nil) then
        exit
```

```

else
    chosenfields.add(Y2Field)
end
IDField = MsgBox.List(theFields, "Each vector must be identified by a unique
number", "Select Unique ID field in table")
if (IDField=nil) then
    exit
else
    chosenfields.add(IDField)
end

elseif (theAnalysisType="Start point & Direction") then
    type=2
    X1Field = MsgBox.List(theFields, "Easting or X values of first point", "Select
X-coordinates field")
    if (X1Field=nil) then
        exit
    else
        chosenfields.add(X1Field)
    end
    Y1Field = MsgBox.List(theFields, "Northing or Y values of first point",
"Select Y-coordinates field")
    if (Y1Field=nil) then
        exit
    else
        chosenfields.add(Y1Field)
    end
    IDField = MsgBox.List(theFields, "Each vector must be identified by a unique
number", "Select Unique ID field in table")
    if (IDField=nil) then
        exit
    else
        chosenfields.add(IDField)
    end
    DispField = MsgBox.List(theFields, "Displacement value", "Select the
displacement field")
    if (DispField=nil) then
        exit
    else
        chosenfields.add(DispField)
    end
    AspectField=MsgBox.List(theFields, "The direction of the displacement",
"Select the direction field")

```

```

if (AspectField=nil) then
  exit
else
  chosenfields.add(AspectField)
end

elseif (theAnalysisType="Start point") then
  type=3
  X1Field = MsgBox.List(theFields, "Easting or X values of first point", "Select
X-coordinates field")
  if (X1Field=nil) then
    exit
  else
    chosenfields.add(X1Field)
  end
  Y1Field = MsgBox.List(theFields, "Northing or Y values of first point",
"Select Y-coordinates field")
  if (Y1Field=nil) then
    exit
  else
    chosenfields.add(Y1Field)
  end
  IDField = MsgBox.List(theFields, "Each vector must be identified by a unique
number", "Select Unique ID field in table")
  if (IDField=nil) then
    exit
  else
    chosenfields.add(IDField)
  end
  DispField = MsgBox.List(theFields, "Displacement value", "Select the
displacement field")
  if (DispField=nil) then
    exit
  else
    chosenfields.add(DispField)
  end
  DispDir=MsgBox.Input("Please input the displacement direction
(degree)","Displacement Direction","0")
  DispDir=(DispDir.AsNumber)
end
  Scal=MsgBox.Input("Please input the scale that enlarge the
displacement","Scale factor","100")
  scale=Scal.AsNumber

```

```

' Arrow rotation angle
RA=10*3.1415926/180
RB=-RA

' check to make sure that the x, y, ID and vertex fields are numeric, the attributes
can be anything
'

for each i in 0 .. (chosenfields.count-2)
  stringTest=chosenfields.get(i)
  if (stringTest.IsTypeNumber.Not) then
    MsgBox.Info(stringTest.asstring++" field must be numeric!","Correct Point
Table")
    exit
  end
end

** Create the new shapefile
'

fnDefault = FileName.Make("$HOME").MakeTmp("shape","shp")
fnOutput = FileDialog.Put( fnDefault,"*.shp","Output Shape File" )
if (fnOutput = nil) then exit end
fnOutput.SetExtension("shp")

'

' Create the field list and see if fields from the input table are to be appended to
the output table
'

ftbOutput = FTab.MakeNew( fnOutput, POLYLINE )
'

X=chosenFields.get(3)          'get the displacement value attribute field in
the source table
FldList={"Shape","ID","Disp"} 'create the field names in the output table
'

ftbOutput.AddFields({Field.Make("ID", #FIELD_LONG, 8, 0)})
ftbOutput.AddFields({Field.make("Disp",#FIELD_DOUBLE,6,2)})

' ** Get a list of all polyline ID values. These are the polylines that you are going
to create
' The ID number will be assigned to the ID field of each polyline that is
created.

```

```

' Also checks to see if there are selected records in the source table.

IDList = List.Make
if (theVTab.GetSelection.Count = 0) then
  theSet = theVTab
else
  theSet = theVTab.GetSelection
end

for each rec in theSet
  IDList.Add(theVTab.ReturnValueString(IDField, rec))
end

**
' Check for errors, remove duplicate ID numbers, and sorts the list.
'
if (IDList.Count > 0) then
  IDList.RemoveDuplicates
  IDList.Sort(true)
else
  exit
end

theBitMap = theVTab.GetSelection
theBitMap.Clearall
theVTab.SetSelection(theBitMap)

**
'
for each ID in IDList
  theVTab.query("[ " ++ IDField.AsString ++ "] = " ++ ID.asString, theBitMap,
#VTAB_SELTYPE_NEW)
  theVTab.UpdateSelection

' **Create a point list
pointList = {}

' ** Make the point and polyline, also add the arrow for it

for each rec in theVTab.GetSelection
if (type=1) then

```

```

    a1Point = point.make( theVTab.ReturnValue (X1Field, rec),
theVTab.ReturnValue (Y1Field, rec) )
    a2Point = point.make( theVTab.ReturnValue (X2Field, rec),
theVTab.ReturnValue (Y2Field, rec) )
    x1=a1Point.GetX
    y1=a1Point.GetY
    pointList.Add(x1@y1)
    x2=a2Point.GetX
    y2=a2Point.GetY
    detx=x2-x1
    dety=y2-y1
    disp=((detx^2)+(dety^2)).sqrt
    detx=detx*scale
    dety=dety*scale
    x2=x1+detx
    y2=y1+dety
    pointList.Add(x2@y2)
    x3=x2-(((detx*(RA.cos))-(dety*(RA.sin)))/10)
    y3=y2-(((detx*(RA.sin))+(dety*(RA.cos)))/10)
    x4=x2-(((detx*(RB.cos))-(dety*(RB.sin)))/10)
    y4=y2-(((detx*(RB.sin))+(dety*(RB.cos)))/10)
    pointList.Add(x3@y3)
    pointList.Add(x4@y4)
    pointList.Add(x2@y2)
end

if (type=2) then
    a1Point = point.make( theVTab.ReturnValue (X1Field, rec),
theVTab.ReturnValue (Y1Field, rec) )
    disp=theVTab.ReturnValue(DispField,rec)
    DispDir=theVTab.ReturnValue(AspectField,rec)

    x1=a1Point.GetX
    y1=a1Point.GetY

    detx=0
    dety=0
    pointList.Add(x1@y1)

    if (DispDir<=90) then
        detx=-disp*(((90-DispDir)*3.1415926/180).cos)
        dety=disp*(((90-DispDir)*3.1415926/180).sin)
    elseif (DispDir<=180) then

```

```

    detx=-disp*(((DispDir-90)*3.1415926/180).cos)
    dety=-disp*(((DispDir-90)*3.1415926/180).sin)
elseif (DispDir<=270) then
    detx=disp*(((DispDir-180)*3.1415926/180).sin)
    dety=-disp*(((DispDir-180)*3.1415926/180).cos)
elseif (DispDir<=360) then
    detx=disp*(((DispDir-270)*3.1415926/180).cos)
    dety=disp*(((DispDir-270)*3.1415926/180).sin)
end

x2=x1+detx
y2=y1+dety
a2Point= point.make( x2,y2 )
x2=a2Point.GetX
y2=a2Point.GetY
detx=x2-x1
dety=y2-y1
detx=detx*scale
dety=dety*scale

x2=x1+detx
y2=y1+dety
pointList.Add(x2@y2)
x3=x2-(((detx*(RA.cos))- (dety*(RA.sin)))/10)
y3=y2-(((detx*(RA.sin))+ (dety*(RA.cos)))/10)
x4=x2-(((detx*(RB.cos))- (dety*(RB.sin)))/10)
y4=y2-(((detx*(RB.sin))+ (dety*(RB.cos)))/10)
pointList.Add(x3@y3)
pointList.Add(x4@y4)
pointList.Add(x2@y2)

end

if (type=3) then
    a1Point = point.make( theVTab.ReturnValue (X1Field, rec),
theVTab.ReturnValue (Y1Field, rec) )
    disp=theVTab.ReturnValue(DispField,rec)

    x1=a1Point.GetX
    y1=a1Point.GetY

    detx=0
    dety=0

```



```

pointList.Add(x1@y1)

if (DispDir<=90) then
  detx=-disp*(((90-DispDir)*3.1415926/180).cos)
  dety=disp*(((90-DispDir)*3.1415926/180).sin)
elseif (DispDir<=180) then
  detx=-disp*(((DispDir-90)*3.1415926/180).cos)
  dety=-disp*(((DispDir-90)*3.1415926/180).sin)
elseif (DispDir<=270) then
  detx=disp*(((DispDir-180)*3.1415926/180).sin)
  dety=-disp*(((DispDir-180)*3.1415926/180).cos)
elseif (DispDir<=360) then
  detx=disp*(((DispDir-270)*3.1415926/180).cos)
  dety=disp*(((DispDir-270)*3.1415926/180).sin)
end

x2=x1+detx
y2=y1+dety
a2Point= point.make( x2,y2 )
x2=a2Point.GetX
y2=a2Point.GetY
detx=x2-x1
dety=y2-y1
detx=detx*scale
dety=dety*scale

x2=x1+detx
y2=y1+dety
pointList.Add(x2@y2)
x3=x2-(((detx*(RA.cos))-(dety*(RA.sin)))/10)
y3=y2-(((detx*(RA.sin))+(dety*(RA.cos)))/10)
x4=x2-(((detx*(RB.cos))-(dety*(RB.sin)))/10)
y4=y2-(((detx*(RB.sin))+(dety*(RB.cos)))/10)
pointList.Add(x3@y3)
pointList.Add(x4@y4)
pointList.Add(x2@y2)

end

newRec = ftbOutput.AddRecord
newline = Polyline.make({pointList})
fldShapeOut=ftbOutput.FindField("shape")

```

```

        fldIDOut=ftbOutput.findField("id")
        fldDataOut=ftbOutput.findField("DISP")
        ftbOutput.SetValue(fldShapeOut, newRec, newLine) '.AsPolyLine)
        ftbOutput.SetValue(fldIDOut, newRec, ID.AsString)
        ftbOutput.SetValue(fldDataOut, newRec, disp)
    end
end
theBitMap.Clearall
theVTab.UpdateSelection
'
' ** Add the output shapefile to a View
'
viewList = list.make
for each d in av.GetProject.GetDocs
    if (d.Is(View)) then
        viewList.Add(d)
    end
end
viewList.Add("<New View>")
addToView = MsgBox.ListAsString( viewList,"Add Theme to:", "Convert
Points to Polylines")
'
' ** Get the specified view, make the theme, and add it...
if (addToView <> nil) then
    if (addToView = "<New View>") then
        addToView = View.Make
        addToView.GetWin.Open
    end
    newTheme = FTheme.Make( ftbOutput )
    addToView.AddTheme( newTheme )
    addToView.GetWin.Activate
end

=====END=====

```

Appendix C

LEAST SQUARES MATCHING

LSM executable program can be produced using three c program files (lsmj.c, lsm_util.c and image.h) which developed by Jeong, J. H. , 2007 in tabagi laboratory of Kochi University of Technology, JAPAN.

The programs can executes by following command in cygwin compiler.

```
..... $ gcc -g lsmj.c lsm_util.c image.h -o lsm.exe
```

And three files for coding are listed below.

```
=====/lsmj.c/====  
#include "image.h"
```

```
int main(int argc, char*argv[])  
{  
  
    int num = NUM_PARAM+1;  
    int i,j;  
    long loop;  
  
    //coefficient  
    //coefficient[0] -> p  
    //coefficient[1] -> q  
    //coefficient[2] -> a11  
    //coefficient[3] -> a12  
    //coefficient[4] -> a21  
    //coefficient[5] -> a22  
    //coefficient[6] -> Uic  
    //coefficient[7] -> Vic  
    double coefficient[NUM_PARAM];  
    //double A_tmp[NUM_PARAM*NUM_PARAM];  
    //double tmp_B[NUM_PARAM];  
    double tmp_A[NUM_PARAM][NUM_PARAM+1];  
    double tmp_B[NUM_PARAM];  
    double work[NUM_PARAM+NUM_PARAM];  
  
    //for(i=0;i<NUM_PARAM;i++) coefficient[i]=0;  
    //double A[1000][8], B[1000];  
    struct dmatrix A,normal_A;  
    struct dvector B,normal_B;  
    long img_size;  
    long n,kl,ku,inc,ldab,info,piv[NUM_PARAM];
```

```

long M,N,lda,ldb,ldw,nrhs;
struct Image target,templ;
//double *work;
double std, old_std, diff_std;
FILE *ouput_residual, *output_result;

if(argc != 2) {
    fprintf(stderr, "least square matching. program \n");
    fprintf(stderr, "Usage: %s [data File]\n", argv[0]);
    fprintf(stderr, "\n data file format\n-----\n");
    fprintf(stderr, "target file name\n");
    fprintf(stderr, "template row \n");
    fprintf(stderr, "template column \n");
    fprintf(stderr, "template file name\n");
    fprintf(stderr, "target row\n");
    fprintf(stderr, "target column\n");
    fprintf(stderr, "p(0) q(0) a11(0) a12(0) a21(0) a22(0)
Uic(0) Vic(0) //(0)-> means initial value \n");
    fprintf(stderr, "-----\n");
    exit(1);
}

read_dat(argv[1],&target, &templ, coefficient);

printf("target name:\t%s\n", target.name);
printf("target row:\t%d\n", target.Rows);
printf("target col:\t%d\n", target.Cols);
printf("target col:\t%d\n", target.Band);
printf("template name:\t%s\n", templ.name);
printf("target row:\t%d\n", templ.Rows);
printf("target col:\t%d\n", templ.Cols);
printf("target col:\t%d\n", templ.Band);
printf("#p(0)\tq(0)\ta11(0)\ta12(0)\ta21(0)\ta22(0)\tUic(0)\t
Vic(0) \n");

for(i=0;i<NUM_PARAM;i++) printf("%lf\t",coefficient[i]);

printf("\n");

A.Cols=NUM_PARAM;
A.Rows=templ.Rows*templ.Cols;
A.Mat=dmatrix(A.Rows,A.Cols);

B.Rows=A.Rows*A.Cols;
B.Vec=dvector(B.Rows);

target.Type=BASIC;
templ.Type=BASIC;
Assign_memory(&target);

```

```

Assign_memory(&templ);

Set_operation(&target, READ);
Set_operation(&templ, READ);
printf("target operation %d\n", target.Operation);
printf("templ operation %d\n", templ.Operation);

Image_IO(&target, target.name);
Image_IO(&templ, templ.name);

Set_LSM_MAT(target, templ, A, B, coefficient);

normal_A.Cols=NUM_PARAM;
normal_A.Rows=NUM_PARAM;
normal_A.Mat=dmatrix(A.Rows*A.Cols, NUM_PARAM);

normal_B.Rows=NUM_PARAM;
normal_B.Vec=dvector(normal_B.Rows);

cal_ATA(normal_A, A);

/* printf("normal_A \n");
for(j=0;j<NUM_PARAM; j++)
{
for(i=0;i<NUM_PARAM;i++)
{
printf("%lf\t", normal_A.Mat[j][i]);
}
printf("\n");
}*/

cal_ATB(normal_B, A, B);
/* printf("normal_B \n");
for(i=0;i<NUM_PARAM;i++) printf("%lf\n", normal_B.Vec[i]);*/

for (i=0;i<NUM_PARAM;i++)
{
tmp_A[i][NUM_PARAM]=normal_B.Vec[i];
// printf("\t %d %lf\n", i, tmp_A[i][NUM_PARAM]);
}

printf("\nmatrixt[A|B]\n");
for (j=0;j<NUM_PARAM;j++)
{
for (i=0;i<NUM_PARAM;i++)
{
tmp_A[j][i]=normal_A.Mat[j][i];
//printf("%lf\t", tmp_A[j][i]);
}
}

```

```

    //printf("\n");
}

for (j=0;j<NUM_PARAM;j++)
{
    for (i=0;i<NUM_PARAM+1;i++)
    {
        printf("%lf\t", tmp_A[j][i]);

    }

    printf("\n");
}

cal_gauss(NUM_PARAM, tmp_A,tmp_B);
update_initial(coefficient, tmp_B, NUM_PARAM);
std=cal_std_error(target,templ,coefficient);
printf(" \n\nstd %lf\n", std);

loop=0;
old_std=0.0;

while(loop !=1000)
{
    Set_LSM_MAT(target,templ,A,B,coefficient);
    for(i=0;i<NUM_PARAM;i++)
    {
        for(j=0;j<NUM_PARAM;j++)
        {
            normal_A.Mat[i][j]=0.0;
        }
        normal_B.Vec[i]=0.0;
    }

}

cal_ATA(normal_A, A);
cal_ATB(normal_B,A,B);
//Set_matrix_for_lapack(tmp_A,normal_A);

for (i=0;i<NUM_PARAM;i++)
{
    tmp_A[i][NUM_PARAM]=normal_B.Vec[i];
    // printf("\t  %d  %lf\n",i, tmp_A[i][NUM_PARAM]);
}

for (j=0;j<NUM_PARAM;j++)
{
    for (i=0;i<NUM_PARAM;i++)
    {
        tmp_A[j][i]=normal_A.Mat[j][i];
    }
}

```

```

        //printf("%lf\t", tmp_A[j][i]);
    }

    //printf("\n");
}

cal_gauss(NUM_PARAM, tmp_A,tmp_B);

update_initial(coefficient, tmp_B, NUM_PARAM);

std=cal_std_error(target,tmp1,coefficient);

for (i=0;i<M;i++) printf("%lf\t", tmp_B[i]);

//printf("%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t\t%l
f\t%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t\n", tmp_B[0], tmp_B[1],
tmp_B[2], tmp_B[3], tmp_B[4], tmp_B[5], tmp_B[6], tmp_B[7],std,
//
coefficient[0],coefficient[1],coefficient[2],coefficient[3],co
efficient[4],coefficient[5],coefficient[6],coefficient[7]);
//printf(" std %lf\n", std);

printf("u %lf v %lf\n", coefficient[6],coefficient[7]);

diff_std=std-old_std;

if (diff_std <0.0) diff_std = diff_std*-1.0;

//printf(" diff std %lf\n", diff_std);

if(diff_std<0.0001)
{
    break;
}
else
{
    old_std=std;
}

loop++;
}

return 0;
}

===== /END/ =====

```

```

===== /lsm_util.c /=====
#include "image.h"

void read_dat(char *fname, struct Image *target, struct Image
*templ, double coefficient[])
{ /*
  fprintf(stderr, "Usage: %s [data File]\n", argv[0]);
  fprintf(stderr, "\n data file format\n-----
-----\n");
  fprintf(stderr, "target file name\n");
  fprintf(stderr, "template Rows \n");
  fprintf(stderr, "template Colsumn \n");
  fprintf(stderr, "template file name\n");
  fprintf(stderr, "target Rows\n");
  fprintf(stderr, "target Colsumn\n");
  fprintf(stderr, "p(0) q(0) a11(0) a12(0) a21(0) a22(0)
Vic(0) Vic(0) //(0)-> means initial value ");
  */

  int i, j, k;
  FILE *fpi;
  char   fname_tmplate[CHR_SIZE];
  char   fname_target[CHR_SIZE];
  char   ibuf[CHR_SIZE];
  char   ctmp[CHR_SIZE];

  if((fpi = fopen(fname, "r")) == NULL) {
    fprintf(stderr, "Can not Open %s", fname);
    exit(1);
  }

  fgets(ibuf, CHR_SIZE, fpi);          //get fname
target
  sscanf(ibuf, "%s %s\n", target->name, ctmp);
// printf("target Rows %s \n", target.name);

  fgets(ibuf, CHR_SIZE, fpi);          //get focal
length
  sscanf(ibuf, "%d %s\n", &target->Rows, ctmp);
//printf("target Rows %d \n", target.Rows);

  fgets(ibuf, CHR_SIZE, fpi);
  sscanf(ibuf, "%d %s\n", &target->Cols, ctmp);

  fgets(ibuf, CHR_SIZE, fpi);
  sscanf(ibuf, "%d %s\n", &target->Band, ctmp);

  fgets(ibuf, CHR_SIZE, fpi);          //get fname
template

```



```

    sscanf(ibuf, "%s %s\n", templ->name, ctmp);

    fgets(ibuf, CHR_SIZE, fpi); //get focal
length
    sscanf(ibuf, "%d %s\n",&templ->Rows, ctmp);

    fgets(ibuf, CHR_SIZE, fpi);
    sscanf(ibuf, "%d %s\n", &templ->Cols, ctmp);

    fgets(ibuf, CHR_SIZE, fpi);
    sscanf(ibuf, "%d %s\n", &templ->Band, ctmp);
    //fscanf(fpi, "%d %d %d %d %d %d %d \n", &coefficient[0],
&coefficient[1], &coefficient[2], &coefficient[3],
&coefficient[4], &coefficient[5], &coefficient[6],
&coefficient[7]);

    fgets(ibuf, CHR_SIZE, fpi);
    sscanf(ibuf, "%lf %lf %lf %lf %lf %lf %lf %lf %s\n",
&coefficient[0], &coefficient[1], &coefficient[2],
&coefficient[3], &coefficient[4], &coefficient[5],
&coefficient[6], &coefficient[7], ctmp);

    //input initial value
    /*for(i=0;i<NUM_PARAM; i++)
    {
    fgets(ibuf, CHR_SIZE, fpi);
    sscanf(ibuf, "%d %s\n", &coefficient[i], ctmp);
    }

    //printf("target Rows %d \n", coefficient[0]);*/

    fclose(fpi);
}

int Image_IO(struct Image *Img, char *filename)
{
    FILE *iofile;
    int i;
    int size;
    unsigned char c;
    if((iofile = fopen (filename, "rb"))== NULL) {
        fprintf(stderr, "Can not Open %s", filename);
        exit(1);
    }

    size=Img->Rows*Img->Band;

```

```

// Decide operation
switch (Img->Operation){

    case READ:
        printf("start reading\n");
        //printf("Image type Basic\n");
        if (Img->Type == BASIC){
            printf("Image type Basic\n");
            for(i=0; i<size; ++i)
                fread(Img->Data+i*Img->Cols,Img->Cols,1,iofile);
        }

        if (Img->Type == INTEGER){
            printf("Image type INTEGER\n");

            for(i=0; i<size; ++i)
                fread(Img->Data_int+i*Img->Cols,Img->Cols,Img-
>sz_pix,iofile);
        }

        if (Img->Type == REAL){
            printf("Image type REAL\n");
            for(i=0; i<size; ++i)
                fread(Img->Data_float+i*Img->Cols,Img->Cols,Img-
>sz_pix,iofile);
        }

        printf("end reading\n");
        break;

    case WRITE:
        printf("start writing\n");
        for(i=0; i<Img->Rows; ++i)
            fread(Img->Data+i*Img->Cols,Img->Cols,Img->sz_pix,iofile);
        printf("end writing\n");
        break;

}

fclose(iofile);
return 0;
}

int Set_operation (struct Image *Img, int operation)
{

```

```

    Img->Operation = operation;
    printf("set operation %d\n", Img->Operation );

    return 0;
}

int Assign_memory(struct Image *Img)
{

    /*// Decide data size
    switch (Img->Type){

        case BASIC:
            Img->sz_pix = sizeof(unsigned char);
            printf("read_data_size %d\n", Img->sz_pix);
            break;

        case INTEGER:
            Img->sz_pix = sizeof(unsigned int);
            printf("read_data_size %d\n", Img->sz_pix);
            break;

        case REAL:
            Img->sz_pix = sizeof(float);
            printf("read_data_size %d\n", Img->sz_pix);
            break;

    }*/

    //Assign memory

    switch (Img->Type){
        case BASIC:
            Img->Data=(unsigned char *)calloc(Img->Cols, Img->Rows);
            if(!Img->Data){
                printf(" out of memory\n");
                exit(1);
            }

            printf("Assign memory: unsigned char\n");
            break;

        case INTEGER:
            Img->Data_int=(unsigned int *)calloc(Img->Cols, Img->Rows);
            if(!Img->Data_int){
                printf(" out of memory\n");
                exit(1);
            }
    }
}

```

```

    }
    printf("Assign memory: int\n");
    break;

    case REAL:
    Img->Data_float=(float *)calloc(Img->Cols, Img->Rows);
    if(!Img->Data_float){
        printf(" out of memory\n");
        exit(1);
    }
    printf("Assign memory:float\n");
    break;

    }

return 0;
}

int Set_pixel_size(struct Image *Img)
{
    // Decide data size
    switch (Img->Type){

        case BASIC:
        Img->sz_pix = sizeof(unsigned char);
        printf("read_data_size %d\n", Img->sz_pix);
        break;

        case INTEGER:
        Img->sz_pix = sizeof(unsigned int);
        printf("read_data_size %d\n", Img->sz_pix);
        break;

        case REAL:
        Img->sz_pix = sizeof(float);
        printf("read_data_size %d\n", Img->sz_pix);
        break;

    }

return 0;
}

int Set_struct(struct Image *Img, int Cols, int Rows, int
type,int band )
{

```

```

    Img->Cols =Cols;
    Img->Rows =Rows;
    Img->Type =type;
    Img->Band =band;
        printf("Cols %d Rows %d Type %d\n", Img->Cols,Img-
>Rows,Img->Type);

    return 0;
}

void Set_LSM_MAT(struct Image target, struct Image
templ,struct dmatrix A, struct dvector B, double
coefficient[])
{
    int i,j;
    double u0, v0;
    int  int_u0, int_v0;

    double  Iu0, Iv0;
    double mid_Cols;
    double mid_Rows;
    double ut_m_utc,vt_m_vtc;

    mid_Cols= (double)(templ.Cols)/2.0;
    mid_Rows= (double)(templ.Rows)/2.0;
        // printf("mid_cols %lf mid_row %lf\n", mid_Cols,
mid_Rows);

    // printf("bilinear(target, coefficient,ut_m_utc+1,
vt_m_vtc) %lf bilinear(target, coefficient,ut_m_utc-1,
vt_m_vtc) %lf\n", bilinear(target, coefficient,ut_m_utc+1,
vt_m_vtc),bilinear(target, coefficient,ut_m_utc-1,
vt_m_vtc));

    for(j=0;j<templ.Rows; j++)
    {
        for(i=0;i<templ.Cols;i++)
        {
            ut_m_utc=(double)i-mid_Cols;
            vt_m_vtc=(double)j-mid_Rows;

            Iu0 = ( bilinear(target, coefficient,ut_m_utc+1, vt_m_vtc)
- bilinear(target, coefficient,ut_m_utc-1, vt_m_vtc))/2.0;
            Iv0 = ( bilinear(target, coefficient,ut_m_utc,
vt_m_vtc+1) - bilinear(target, coefficient,ut_m_utc,
vt_m_vtc-1))/2.0;
            // printf("Iu0 %lf Iv0 %lf\n", Iu0, Iv0);

```

```

// printf(" bilinear(target, coefficient,ut_m_utc,
vt_m_vtc) %lf\n",bilinear(target, coefficient,ut_m_utc,
vt_m_vtc));

    A.Mat[j*templ.Cols+i][0]= bilinear(target,
coefficient,ut_m_utc, vt_m_vtc);
    A.Mat[j*templ.Cols+i][1]=1.0;

    A.Mat[j*templ.Cols+i][2]=coefficient[0]*Iu0*ut_m_utc;
    A.Mat[j*templ.Cols+i][3]=coefficient[0]*Iu0*vt_m_vtc;

    A.Mat[j*templ.Cols+i][4]=coefficient[0]*Iv0*ut_m_utc;
    A.Mat[j*templ.Cols+i][5]=coefficient[0]*Iv0*vt_m_vtc;

    A.Mat[j*templ.Cols+i][6]=coefficient[0]*Iu0;
    A.Mat[j*templ.Cols+i][7]=coefficient[0]*Iv0;

    B.Vec[j*templ.Cols+i]=templ.Data[j*templ.Cols+i]-
(coefficient[0]*bilinear(target, coefficient,ut_m_utc,
vt_m_vtc)+coefficient[1]);

    //printf(" %d - (%lf * %lf +%lf) = %lf\n",
templ.Data[j*templ.Cols+i],coefficient[0],bilinear(target,
coefficient,ut_m_utc,
vt_m_vtc),coefficient[1],B.Vec[j*templ.Cols+i]);

}
}
}

double **dmatrix(long Rows, long Column)
{
    double **m;
    long i;

    //m=(double **) malloc(((Rows)*sizeof(double*)));
    m=malloc(((Rows)*sizeof(double*)));

    for (i=0; i<Rows; i++)
m[i]=malloc(((Column)*sizeof(double)));//m[i]=(double *)
malloc(((Column)*sizeof(double)));

    if (!m)
    {
        printf("allocation failure 1 in matrix()\n");
        exit(1);
    }
}

```

```

    }

    return m;
}

double *dvector(long Rows)
{
    double *m;
    long i;

    //m=(double *)malloc(((Rows)*sizeof(double)));
    m=calloc(Rows,sizeof(double));
    //for (i=0; i<Rows; i++) m[i]=(double *)
    malloc(((Column)*sizeof(double)));

    if (!m)
    {
        printf("allocation failure 1 in matrix()\n");
        exit(1);
    }

    return m;
}

int test( struct Image *Img)
{
    int i,j;

    for(j=0 ; j<Img->Rows; j++)
    for(i=0 ; i<Img->Cols; i++)
    {
        printf("test %d\n", *(Img->Data+j*Img->Cols+i));
    }

    return 0;
}

void Set_matrix_for_lapack(double *A_tmp, struct dmatrix A)
{
    long i,j,n;

    n=0;

    for (i=0;i<A.Cols;i++)
    {
        for(j=0;j<A.Rows;j++)
        {

```

```

        A_tmp[n]=A.Mat[j][i];
        n++;
    }
}
}

void cal_ATA(struct dmatrix out_normal_matrix, struct dmatrix
input_matrix)
{
    long i,j,k,l;

    if(out_normal_matrix.Cols != input_matrix.Cols)
    {
        printf("in Function cal_ATA\n");
        printf("out_normal_matrix.Cols %d is not same as
input_matrix.Cols %d\n", out_normal_matrix.Cols ,
input_matrix.Cols );
        exit(0);
    }

    if(out_normal_matrix.Rows != input_matrix.Cols)
    {
        printf("in Function cal_ATA\n");
        printf("out_normal_matrix.Rows %d is not same as
input_matrix.Cols %d\n", out_normal_matrix.Rows ,
input_matrix.Cols );
        exit(0);
    }

    // transpose and multiply A^T*A
    for(j=0;j<input_matrix.Cols;j++){
        for(i=0;i<input_matrix.Cols;i++){
            for(k=0;k<input_matrix.Rows;k++){
                out_normal_matrix.Mat[j][i] +=
input_matrix.Mat[k][j]*input_matrix.Mat[k][i];
            }
        }
        //printf("%lf\t", out_normal_matrix.Mat[j][i]);
    }
    // printf("\n");
}
}

```



```

void cal_ATB(struct dvector out_normal_vector,struct dmatrix
input_matrix, struct dvector input_vector)
{
    long i,j,k,l;

    if(out_normal_vector.Rows != input_matrix.Cols)
    {
        printf("in Function cal_ATB\n");
        printf("out_normal_vector.Rows %d is not same as
input_matrix.Cols %d\n", out_normal_vector.Rows ,
input_matrix.Cols );
        exit(0);
    }

    if(input_vector.Rows != input_matrix.Rows)
    {
        printf("in Function cal_ATB\n");
        printf("input_vector.Rows %d is not same as
input_matrix.Rows %d\n", input_vector.Rows ,
input_matrix.Rows );
        exit(0);
    }

    // transpose and multiply A^T*B
    for(j=0;j<input_matrix.Cols;j++){
        for(k=0;k<input_matrix.Rows;k++){
            out_normal_vector.Vec[j] +=
input_matrix.Mat[k][j]*input_vector.Vec[k];
            //printf("j %d , k %d\n",j,k);
        }
    }

    //printf("input_matrix.Cols %d ,input_matrix.Rows %d\n",input
_matrix.Cols,input_matrix.Rows);

}

void update_initial(double *initial, double *correction, int
num_initial)
{
    int i;

    //printf("\n\n");
    for (i=0;i <num_initial; i++)
    {
        //printf("initial befor correction\t after correction\n");
        // printf("%lf\t",initial[i]);
        // printf("+\t%lf\t", ( correction[i]));
        initial[i] += correction[i];
        // printf("=%lf\n", (initial[i]));
    }
}

```

```

    }
}

double cal_std_error(struct Image target, struct Image templ,
double coefficient[])
{
    int i,j;
    double u0, v0;
    double std;
    int int_u0, int_v0;
    double Iu0, Iv0;
    double mid_Cols;
    double mid_Rows;
    double ut_m_utc,vt_m_vtc;
    double sum_residual_square=0;

    mid_Cols= templ.Cols/2.0;
    mid_Rows= templ.Rows/2.0;
    // printf("mid_cols %lf mid_row %lf\n", mid_Cols, mid_Rows);

    for(j=0;j<templ.Rows; j++)
    {
        for(i=0;i<templ.Cols;i++)
        {
            ut_m_utc=(double)i-mid_Cols;
            vt_m_vtc=(double)j-mid_Rows;

            u0=
            (coefficient[2]*(ut_m_utc)+coefficient[3]*(vt_m_vtc)+coefficient[6]);
            v0=
            (coefficient[4]*ut_m_utc+coefficient[5]*vt_m_vtc+coefficient[7]);
            // printf("u0 %lf v0 %lf \n", u0,v0);

            int_u0=(int)(u0+0.5);
            int_v0=(int)(v0+0.5);

            //printf("%d\n",target.Data[0]);

            Iu0=((double)target.Data[int_v0*target.Cols+int_u0+1]-
            (double)target.Data[int_v0*target.Cols+int_u0-1])/2.0;
            Iv0=((double)target.Data[(int_v0+1)*target.Cols+int_u0]-
            (double)target.Data[(int_v0-1)*target.Cols+int_u0])/2.0;

            //printf("Iu0 %lf Iv0 %lf \n", Iu0,Iv0);

```

```

        sum_residual_square +=(templ.Data[j*templ.Cols+i]-
(coefficient[0]*target.Data[int_v0*target.Cols+int_u0]+coeffic
ient[1]))*(templ.Data[j*templ.Cols+i]-
(coefficient[0]*target.Data[int_v0*target.Cols+int_u0]+coeffic
ient[1]));

    }
}
//printf("sum_residual_square %lf\n", sum_residual_square);

std= sqrt(sum_residual_square / ((double)
templ.Cols*(double)templ.Rows-(double)NUM_PARAM));

// printf("std %lf\n", std);

return std;
}

void cal_gauss(int num, double aa[][NUM_PARAM+1],double *xx)

{
long i, j, k;
double cc[NUM_PARAM][NUM_PARAM+1];
double mm;

for(i=0; i<num; i++)
for(j=0; j<num+1; j++)
cc[i][j] = aa[i][j];

for(i=0; i<num; i++) {
for(j=0; j<num; j++) {
if(i != j) {
mm = cc[j][i] / cc[i][i];
//printf( " %lf\t %d \t %d \n ", mm, cc[j][i],cc[i][i]);
for(k=0; k<num+1; k++)
cc[j][k] -= cc[i][k]*mm;
}
}
}

for(i=0; i<num; i++)
xx[i] = cc[i][num] / cc[i][i];

// for(i=0;i < num ; i++) printf("%lf \t", xx[i]);

}

double bilinear(struct Image target, double
coefficient[],double u, double v)
{

```

```

double normal_u, normal_v;
int u0,v0;
// u0 integer part of u
// v0 integer part of v
// u0,v0 (i,j)
// u p 1 i+1
// v_p1 j+1
// normal u float part of u
// normal v float part of v

normal_u= (coefficient[2]*u+coefficient[3]*v+coefficient[6])-
(double) ( (int)
(coefficient[2]*u+coefficient[3]*v+coefficient[6]) );
normal_v= (coefficient[4]*u+coefficient[5]*v+coefficient[7])-
(double) ( (int)
(coefficient[4]*u+coefficient[5]*v+coefficient[7]) );

u0= ( (int)
(coefficient[2]*u+coefficient[3]*v+coefficient[6]) );
v0= ( (int)
(coefficient[4]*u+coefficient[5]*v+coefficient[7]) );

//printf(" u %lf v%lf normal_u %lf normal_v %lf u0 %d v0 %d
\n",u,v, normal_u,normal_v, u0, v0);

return (1.0-normal_u) * (1.0-normal_v) * target.Data[v0 *
target.Cols+u0] + (1.0 - normal_u) * normal_v *
target.Data[(v0+1) * target.Cols + u0] + normal_u * (1.0 -
normal_v) *
target.Data[v0*target.Cols+u0+1]+normal_u*normal_v*
target.Data[(v0+1)*target.Cols+u0+1];

}
=====/END/====

=====/image.h/====
#include<stdio.h>
#include<stdlib.h>
#include<memory.h>
#include<math.h>
struct Image{

int Rows;
int Cols;

int sz_pix;
char name[256];

unsigned char *Data;

```

```

unsigned int *Data_int;
float *Data_float;

unsigned char *Data_tmp;
unsigned int *Data_int_tmp;
float *Data_float_tmp;

int Band;
int Type;
int Operation;

};

struct dmatrix{
    long Rows;
    long Cols;
    double **Mat;
};

struct dvector{
    long Rows;
    double *Vec;
};

#define BASIC 0
#define INTEGER 1
#define REAL 2
#define CHR_SIZE 256
#define NUM_PARAM 8

//***** operation *****
#define WRITE 3
#define READ 4
//*****

void read_dat(char *fname, struct Image *target, struct Image
*templ, double coefficient[]);
int Image_IO(struct Image *Img, char *filename);
int Set_operation(struct Image *Img, int operation);
int Set_struct(struct Image *Img, int cols, int rows, int
type, int band );
int Set_pixel_size(struct Image *Img);
int Assign_memory(struct Image *Img);
int Matching_2img(struct Image *Img_source, struct Image
*Img_template);
int Assign_memory_tmp(struct Image *Img, int Type);
double *dvector(long Rows);
double **dmatrix(long Rows, long Column);

```

```

void Set_LSM_MAT(struct Image target, struct Image
templ,struct dmatrix A, struct dvector B, double
coefficient[]);
void Set_matrix_for_lapack(double *A_tmp, struct dmatrix A);
void cal_ATA(struct dmatrix out_normal_matrix, struct dmatrix
input_matrix);
void cal_ATB(struct dvector out_normal_vector,struct dmatrix
input_matrix, struct dvector input_vector);
void update_initial(double *initial, double *correction, int
num_initial);
//double cal_std_error(double *correction,struct dmatrix
normal_A, struct dvector normal_B);
double cal_std_error(struct Image target, struct Image templ,
double coefficient[]);
void cal_gauss(int num, double aa[][NUM_PARAM+1],double *xx);
double bllinear(struct Image target, double
coefficient[],double u, double v);
===== /END /=====

```

Appendix D

DEFINITIONS OF REMOTE SENSING

Remote sensing has been variously defined but basically it is the art or science of telling something about an object without touching it. (Fischer et al. 1976, p. 34)

Remote sensing is the acquisition of physical data of an object without touch or contact. (Lintz & Simonett 1976, p. 1)

...imagery is acquired with a sensor other than (or in addition to) a conventional camera through which a scene is recorded, such as by electronic scanning, using radiations outside the normal visual range of the sensor and camera—microwave, radar, thermal, infrared, ultraviolet, as well as multispectral, special techniques are applied to process and interpret remote sensing imagery for the purpose of producing conventional maps, thematic maps, resources surveys, etc., in the fields of agriculture, archaeology, forestry, geography, geology, and others. (American Society of Photogrammetry)

Remote sensing is the observation of a target by a device separated from it by some distance. (Barrett & Curtis, 1976, p. 3)

The term “remote sensing” in its broadest sense merely means “reconnaissance at a distance.” (Colwell, 1966, p. 71)

Remote sensing, though not precisely defined, includes all methods of obtaining pictures or other forms of electromagnetic records of the Earth’s surface from a distance, and the treatment and processing of the picture data.... Remote sensing then in the widest sense is concerned with detecting and recording electromagnetic radiation from the target areas in the field of view of the sensor instrument. This radiation may have originated directly from separate components of the target area; it may be solar energy reflected from them; or it may be reflections of energy transmitted to the target area from the sensor itself. (White, 1977, pp. 1-2)

“Remote sensing” is the term currently used by a number of scientists for the study of remote objects (earth, lunar, and planetary surfaces and atmospheres, stellar and galactic phenomena, etc.) from great distances. Broadly defined..., remote sensing denotes the joint effects of employing modern sensors, data-processing equipment, information theory and processing methodology, communications theory and devices, space and airborne vehicles, and large-

systems theory and practice for the purposes of carrying out aerial or space surveys of the earth's surface. (National Academy of Sciences, 1970, p.1)

Remote sensing is the science of deriving information about an object from measurements made at a distance from the object, i.e., without actually coming in contact with it. The quantity most frequently measured in present-day remote sensing systems is the electromagnetic energy emanating from objects of interest, and although there are other possibilities (e.g. seismic waves, sonic waves, and gravitational force), our attention...is focused upon systems which measure electromagnetic energy. (D.A. Landgrebe, in Swain & Davis, 1978, p.1)

Remote sensing is the practice of deriving information about earth's land and water surfaces using images acquired from an overhead perspective, using electromagnetic radiation in one or more regions of the electromagnetic spectrum, reflected or emitted from the earth's surface. (Campbell, James B., 1996, p.5)

Appendix E

MILESTONES IN THE HISTORY OF REMOTE SENSING

1800	Discovery of infrared by Sir William Hershel
1839	Beginning of practice of photography
1847	Infrared spectrum shown by A. H. L. Fizeau and J. B. L. Foucault to share properties
1850-1860	Photography from balloons
1873	Theory of electromagnetic energy developed by James Clerk Maxwell
1909	Photography from airplanes
1910-1920	World War I: aerial reconnaissance
1920-1930	Developmental and initial applications of aerial photography and photogrammetry
1930-1940	Development of radar in Germany, United States, and United Kingdom
1940-1950	World War II: applications of nonvisible portions of electromagnetic spectrum; training of persons in acquisition and interpretation of airphotos
1950-1960	Military research and development
1956	Colwell's research on disease detection with infrared photography
1960-1970	First use of term "remote sensing" TIROS weather satellite Skylab remote sensing observations from space
1972	Launch of Landsat 1
1970-1980	Rapid advances in digital image processing
1980-1990	Landsat 4: new generation of Landsat sensors
1986	(SPOT) French Earth Observation Satellite
1980s	Development of hyperspectral sensors
Campbell, James B., 1996, p.6	
1999	IKONOS: The first commercially available high resolution imagery at 1 and 4 meter.
1996	MOM-2P: second generation instrument with high performance along track stereo capability
1992	JERS-1: Asia first and dual sensors for optical and microwave remote sensing
2006	ALOS: Asia first and multi sensors onboard satellite with multi purpose capabilities

Appendix F

KEY FACTORS OF REMOTE SENSING

Spectral Differentiation: Remote sensing depends on observed spectral differences in the energy reflected or emitted from features of interest.

Radiometric Differentiation: Satellite images depend on detection of differences in brightness of objects and features.

Spatial Differentiation: Satellite sensors are limited in respect to the size of the data sampling area that can be separately recorded as a pixel on an image.

Geometric Transformation: Image represents a landscape in a specific geometric relationship determined by the design of the remote sensing instrument, specific operating conditions, terrain relief, and the other factors.

Interchangeability of Pictorial and Digital Formats: many remote sensing systems generate, as their primary output, digital arrays that represent brightness of areas of the earth's surface.

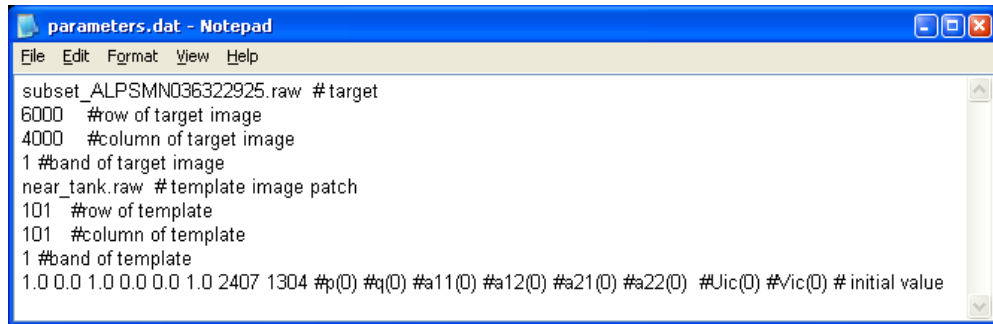
Remote Sensing Instrumentation Acts as a System: The image analyst must always be conscious of the fact that the many components of the remote sensing process act as a system and cannot be isolated from one another.

Role of the Atmosphere: All energy reaching the remote sensing instrument must pass through a portion of the earth's atmosphere.

Appendix G

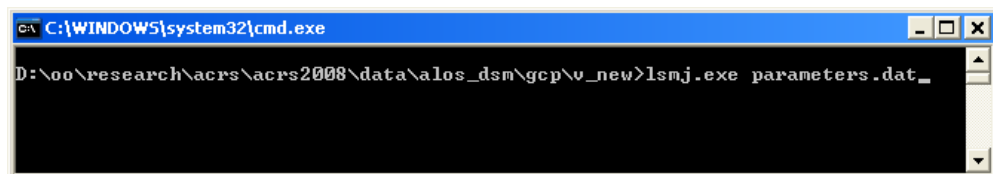
APPLICATION USAGES

After generating the reference image from DSM and spectral reference data, GCP template patches are selected. The extent of patches is 101 x 101 in pixels. These patches are matched with least squares matcher program including parameter file. The parameters of matcher program are target image in raw format, the initial point, and the patch file in raw format. The program is written in C program and it can be run under the DOS command prompt. At least, a Least Squares Matching value is produced by iterating the initial point. The LSM values are the generated GCP in this study.



```
parameters.dat - Notepad
File Edit Format View Help
subset_ALPSMN036322925.raw #target
6000 #row of target image
4000 #column of target image
1 #band of target image
near_tank.raw #template image patch
101 #row of template
101 #column of template
1 #band of template
1.0 0.0 1.0 0.0 0.0 1.0 2407 1304 #p(0) #q(0) #a11(0) #a12(0) #a21(0) #a22(0) #Uic(0) #Vic(0) # initial value
```

Figure: Sample LSM parameters file



```
C:\WINDOWS\system32\cmd.exe
D:\oo\research\acrs\acrs2008\data\alos_dsm\gcp\v_new>lsmj.exe parameters.dat_
```

Figure: Sample DOS command

```

C:\WINDOWS\system32\cmd.exe
target row:      6000
target col:      4000
target col:      1
template name:   near_tank.raw
target row:      101
target col:      101
target col:      1
#p(0)    q(0)    a11(0)  a12(0)  a21(0)  a22(0)  Uic(0)  Uic(0)
1.000000  0.000000  0.000000  1.000000  1.000000  0.000000  0.000000  0.000000
1.000000  2407.000000  1304.000000
Assign memory: unsigned char
Assign memory: unsigned char
set operation 4
set operation 4
target operation 4
templ operation 4
start reading
Image type Basic
end reading
start reading
Image type Basic
end reading

matrix[A;B]
67175726.750000  815745.500000  -2602345.281250  2243273.531250  2915860.328125
3890341.171875  24347.875000  -125213.906250  -21267364.750000
815745.500000  10201.000000  -25517.500000  29480.000000  33632.437500
39891.562500  330.500000  -1250.375000  -244605.500000
-2602345.281250  -25517.500000  52492531.757813  8675262.320313  13077352.746094
-1342565.441406  267803.234375  219253.476563  2592021.781250
2243273.531250  29480.000000  8675262.320313  60742425.820313  -1342565.441406
18013194.683594  -1109465.390625  -283955.710938  -894979.281250
2915860.328125  33632.437500  13077352.746094  -1342565.441406  68230045.207031
8129315.308594  219253.476563  642592.648438  745264.359375
3890341.171875  39891.562500  -1342565.441406  18013194.683594  8129315.308594
82910921.300781  -283955.710938  -1278251.023438  -1013828.109375
24347.875000  330.500000  267803.234375  -1109465.390625  219253.476563
-283955.710938  154228.406250  57962.359375  28983.375000
-125213.906250  -1250.375000  219253.476563  -283955.710938  642592.648438
-1278251.023438  57962.359375  216040.078125  48353.281250

std 45.432568
u 2410.091048 v 1302.021591
u 2410.433183 v 1302.425781
u 2409.648404 v 1302.871030
u 2410.521592 v 1302.855483
u 2410.001311 v 1303.445653
u 2410.333167 v 1303.356450
u 2410.308560 v 1303.552296
u 2410.416599 v 1303.647244
u 2410.456250 v 1303.814523
u 2410.522466 v 1304.039151
u 2410.573268 v 1304.383551
u 2410.649412 v 1304.779075
u 2410.737387 v 1305.104604
u 2410.750371 v 1305.376968
u 2410.761826 v 1305.602735
u 2410.743680 v 1305.806362
u 2410.716028 v 1305.991284
u 2410.698634 v 1306.140689
u 2410.687142 v 1306.251444
u 2410.650994 v 1306.341433
u 2410.601139 v 1306.415000
u 2410.529747 v 1306.482229
u 2410.426633 v 1306.551969
u 2410.313491 v 1306.619765
u 2410.195978 v 1306.687500
u 2410.085833 v 1306.744038
u 2410.007296 v 1306.778063
u 2409.957036 v 1306.799268
u 2409.918414 v 1306.810279
u 2409.887886 v 1306.815221
u 2409.863425 v 1306.816430
u 2409.842286 v 1306.815228
u 2409.822859 v 1306.813835
u 2409.805217 v 1306.813515
u 2409.787880 v 1306.813694
D:\oo\research\acrs\acrs2008\data\alos_dsm\gcp\u_new>

```

Figure: Iterated GCP result

HOW TO ACCESS GCP DATABASE THROUGH INTERNET?

Direct web address to GCP database is copied below.

http://www.infra.kochi-tech.ac.jp/takalab/gcp_correction/GCPDB/index.html

Users can also go through step by step from Takagi laboratory main page as follow:

- a) Takagi laboratory (<http://www.infra.kochi-tech.ac.jp/takalab/index.html>). Takagi lab is a laboratory of Professor Masataka Takagi and he is professor in Department of Infrastructure Systems Engineering, Kochi University of Technology, JAPAN.
- b) Construction of GCP database is one of his research projects on RS&GIS. GCP DB research page can be reached from menu of the lab page through research menu option.

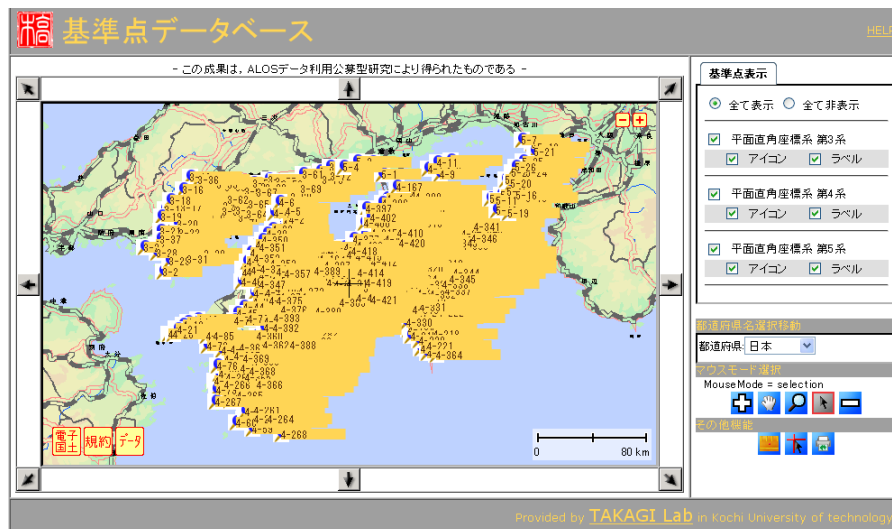


- c) The GCP database can be opened by clicking through text of こちら (this Japanese Hiragana characters mean HERE in English).

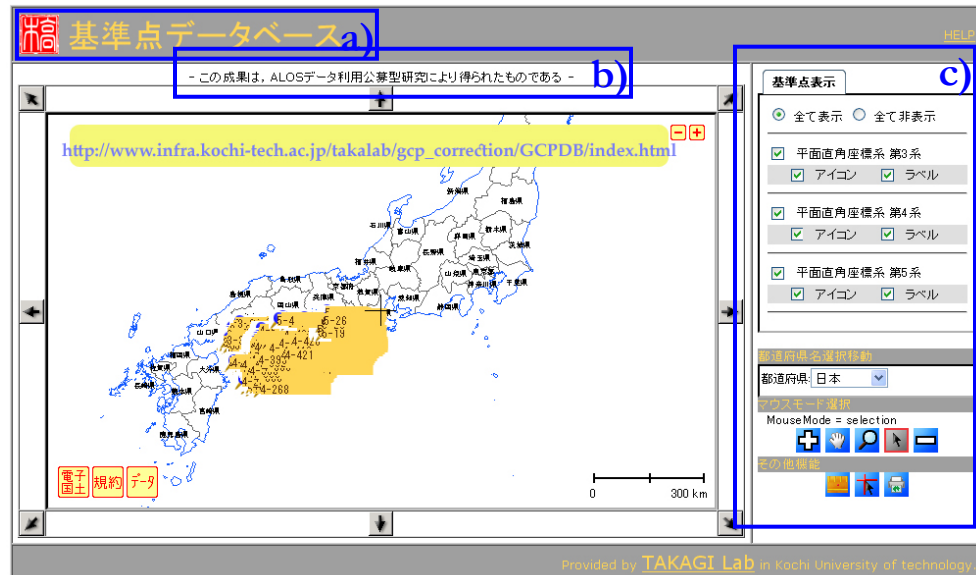
When you click on こちら, research introduction page will be appeared (<http://www.infra.kochi-tech.ac.jp/takalab/Information/research/GCPDB/GCPDB.html>).

Under the introduction page there is a link 同意する(電子国土検索画面へ) (the meaning of this text link is “if you agree with the above data policy, here you can access digital country interactive page”) to WebGIS. (http://www.infra.kochi-tech.ac.jp/takalab/gcp_correction/GCPDB/index.html).

Finally, user will be arrived to GCP database as show below. Figure is the most zoom out view of this database because recently the database is covered Shikoku island of Japan.

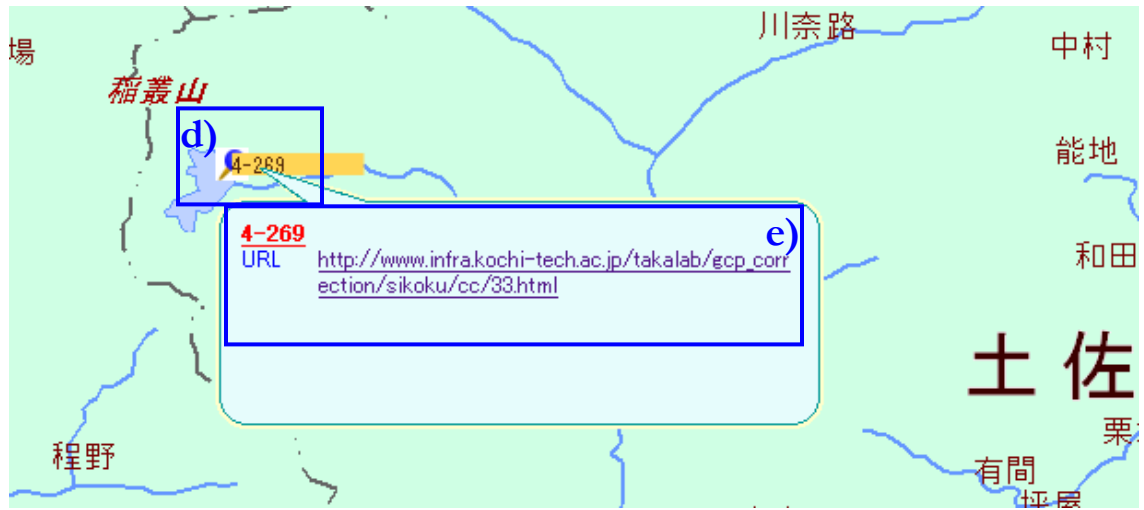


Explanation of Japanese text website (shown in figure)



- The red Japanese character is the seal of professor Takagi. Other characters mean “GCP database”.
- The second characters might mean “This result is a product of ALOS data for public access”.
- The right side textbox are data on/off options and WebGIS functions.

When users zoom in to the database through + symbols from right corner of the database main view, the webmap will zoom in. User may see as show below.



- d) When user point the mouse pointer on a GCP, a pop-up window which linked to detail data page will be appeared (e.g. GCP 4-269 is pointing in above figure).
- e) In the pop-up window, there are short description about GCP and a URL to detail GCP data.

After user clicked on the URL link, the detail information for the GCP 4-269 will be appeared.

基準点座標詳細

- この成果は、ALOSデータ利用公募型研究により得られたものである -

名称	4-269	画像をクリックすると別ウィンドウで拡大表示されます。	
種類			
住所			
電話番号			
URL	http://www.infra.kochi-tech.ac.jp/takalab/gcp_correction/sikoku/cc/33.html		
ひとこと			
測地系	世界測地 第4系	<div style="border: 1px solid black; height: 100px; width: 100%;"></div>	
X座標	-11740.538		
Y座標	82078.393		
標高	1128.019		
緯度	33.740047		
経度	133.373289		
測量日	2009/5/2		
観測箇所	ダムの角		

- f) The title texts mean “GCP Database - this result is a product of ALOS data for public access-”. It will appear on all GCP pages.
- g) The text mean “when the user click on the picture, it will expend to another window with full extent”. All photographs and image subsets are clickable.
- h) The left most fields are descriptions of GCP such as Name, Type, Address, Telephone number, URL, Personal remark, Geographic coordinate system (e.g. state plane region 4 for Japanese datum), X coordinate, Y coordinate, height, Latitude, Longitude, Observation date, and Observation place (e.g. corner of dam).
- i) Right region has clickable photographs and subset of ALOS satellite images.

Some useful links for the GCP database information:

- 1) Core GIS data is acquired from Geographical Survey Institute
(<http://www.gsi.go.jp/ENGLISH/index.html>)
- 2) About VRS technology and GPS device in Trimble
(<http://www.trimble.com/vrsnow.shtml>).

BIBLIOGRAPHY

- Ackermann, F., 1984. *Digital Image Correlation: Performance and Potential Application in Photogrammetry*. Photogrammetry Record, 11(64): 429-439.
- Ackermann, F., 1984. *High Precision Digital Image Correlation*. In Proceedings of the 39th Photogrammetric Week, volume 9 of Schriftenreihe der Universität Stuttgart, Stuttgart.
- ALOS, 2006. *Product format description (PRISM)*. NEC/TOSHIBA, NEC TOSHIBA Space Systems, Ltd.
- ALOS CAL/VAL, 2006. *PRISM consolidated verification report*. GAEL-P237-DOC-005. Issue 1, revision 0 – Dec 20th, 2006. GAEL Consultant.
- Argialas, D. P. and Harlow, C. A., 1990. *Computational image interpretation models: an overview and a perspective*. Photogrammetric Engineering and Remote Sensing, 56, 6, 871-886.
- Barrett, E. C., and L. F. Curtis, 1976. *Introduction to Environmental Remote Sensing*. New York: Wiley, p.3.
- Campbell, J. B., 1996. *Introduction to Remote Sensing*. 2nd ed., the Guilford Press, USA, p.5.
- Colwell, Robert N., 1966. *Uses and Limitations of Multispectral Remote Sensing*. In reocceedings of the fourth Symposium on Remote Sensing of Environment. Ann Arbor: University of Michigan, p.71.
- Cox, M. G., Dainton, M.P., Harris, P.M. 2001. *Software Specifications for Unvertainty Calculation and Associated Statistical Analysis*. NPL Report CMSC 10/01.
- Dial, G., J. Grodecki, 2005. *RPC Replacement Camera Models*. ASPRS 2005 Annual Conference on “Geospatial Goes Global: From Your Neighborhood to the Whole Planet”. March 7-11, 2005, Baltimore, Maryland.
- Ebner, H., G. Strunz, 1988. *Combined Point Determination Using Digital Terrain Models as Control Information*. Int. Arch. of Photogrammetry and Remote Sensing, 27(Part B11): 578-587.
- Ebner, H., W. Kornus, G. Strunz, O. Hoffman, and F. Muller, 1991. *A Simulation Study on Point Determination Using MOMS-02/D2 Imagery*. Photogrammetric Engineering & Remote Sensing, 57(10): 1315-1320.
- Ebner, H., T. Ohlhof, and E. Putz, 1996. *Orientation of MOMS-02/D2 and MOMS-2P Imagery*. Int. Arch. of Photogrammetry and Remote Sensing, 31(Part B2): 158-164.
- Ebner, H., G. Strunz, 1988. *Combined Point Determination Using Digital Terrain Models as Control Information*. International Archive of Photogrammetry and Remote Sensing, 27(Part B11): 578-587.
- EORC-JAXA (2006). PRISM: Panchromatic Remote-sensing Instrument for Stereo Mapping: About ALOS. <http://www.eorc.jaxa.jp/ALOS/en/about/prism.htm>

- Fischer, William A., W. R. Hemphill, and A. Kover, 1976. *Progress in Remote Sensing*. Photogrammetria, Vol. 32, pp. 34.
- Förstner, W., 1986. *A Feature Based Correspondence Algorithm for Image Matching*. In Proceedings ISPRS Commission III Symposium, volume XXVI-3/3 of International Archives of Photogrammetry and Remote Sensing, p. 150-166, Rovaniemi, Finland.
- Ghilani, Charles D., and Paul R. Wolf. 2006. *Addustment Computation: spatial data analysis*, 4th ed., John Wiley & Sons, Inc., Hoboken, NJ.
- Grodecki, J., (2001). *IKONOS Stereo Feature Extraction – RPC approach*. Proceedings of 2001 ASPRS Annual Convention (CD ROM) 23-27 April, St. Louis, Missouri, unpaginated.
- Grodecki, J., G. Dial, (2002). *IKONOS Geometric Accuracy Validation*. Proceedings of 2002 PECORA / ISPRS Commission I, Denver, CO (on CD ROM).
- Gruen, A.W., 1985. *Adaptive least squares correlation: a powerful image matching technique*. South African Journal of Photogrammetry, Remote Sensing and Cartography, 14(3), p. 175-187.
- Gruen, A., 2000. *Potential and Limitations of Highresolution Satellite Imagery*. Keynote Address on 21st Asian Conference on Remote Sensing, December 4-8, 2000, Taipei, Taiwan.
- Hashimoto, T., 2006. *Geometric Properties of ALOS/PRISM and AVNIR-2 Images*. Asian Journal of Geoinformatics. Vol. 6, No. 4, October 2006. p. 21-27.
- Hastedt, H., 2006. (2006). *Monte-Carlo-Simulation in close-range photogrammetry*, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34, Part XXX[CD-ROM].
- Holben, B. N., Justice, C. O., 1980. *The topographic effects on spectral response from nadir-point sensors*. Photogrammetric Engineering & Remote Sensing, 46(10): p. 1191-1200.
- Horn, B. K. P., 1982. *Hill shading and the reflectance map*. Geo-Processing, v. 2, p. 65-146.
- IKONOS. Imagery Sources, Geoeye, USA <http://www.geoeeye.com/CorpSite/products/imagery-sources/Default.aspx> (accessed 10 Jan. 2009).
- Imhof, E., 1982. *Cartographic Relief Representation*. Berlin and New York, Walter de Gruyter.
- Jacobsen, K., 2001. *Automatic matching and generation of orthophotos from airborne and spaceborne line scanner images*. Proceeding of Joint ISPRS Workshop on “High Resolution Mapping from Space 2001”, University of Hannover (on CD).
- Jacobsen, K. 2002a. *Generation of Orthophotos with CARTERRA Geo Images without Orientation Information*. ASPRS annual convention, Washington.
- Jacobsen, K. 2002b. *Mapping with IKONOS images*. EARSeL 2002, Prague, Geoinformation for European-wide Integration, Millpress ISBN 90-77017-71-2, p. 149-156.

- Jacobsen, K. 2003. *DEM generation from satellite data*. EARSeL 2003, Ghent.
- Jacobsen, K. 2003. *Orthoimages and DEMs by QuickBird and IKONOS*. EARSeL 2003, Ghent.
- Jacobsen, K. 2006. *Comparison of Image Orientation by IKONOS, QuickBird and OrbView-3*. EARSeL symposium 2006.
- JAXA EORC. Earth Observation satellite, JAXA, JAPAN
http://www.eorc.jaxa.jp/hatoyama/satellite/sat_menu_e.html (accessed 10 Jan. 2009).
- Kojima, et al., 2008. *Accuracy of 3 Dimensional Measurement by ALOS AVNIR-2*. Asian Conference on Remote Sensing 2008, 10-14 Nov 2008, Colombo, Sri Lanka.
- Kojima et al., 2009. *Establishment of Ground Control Points Database for Satellite Remote Sensing and the Evaluation*, 30th Asian Conference on Remote Sensing 2009, 18-23 Oct 2009, Beijing, CHINA.
- Kornus, W., Lehner, M., Blechinger, F., Putz, E., 1996. *Geometric Calibration of the Stereoscopic CCD-Linescanner MOMS-2P*. International Archives of Photogrammetry and Remote Sensing, Vol. 31, Part B1, 90-98, Vienna, Austria.
- Kornus, W., 1996. *MOMS-2P Geometric Calibration Report (Version 1.1)-Results of Laboratory Calibration*. Institute of Optoelectronics, DLR, Wessling, Germany, 16 p.
- Kornus, W., 1997. *MOMS-2P Geometric Calibration Report (Version 1.3)-Results of Channel 5A/5B Registration, Institute of Optoelectronics*. DLR, Wessling, Germany, 40 p.
- Kornus, W., and M. Lehner, 1998. *Photogrammetric Point Determination and DEM Generation Using MOMS-2P/PRIRODA Three-Line Imagery*. Int. Arch. of Photogrammetry and Remote Sensing. 32 (Part 4): 321-328.
- Kornus, W., Lehner, M., Schroeder, M., 1999. *Geometric Inflight-Calibration by Block Adjustment Using MOMS-2P Imagery of Three Intersecting Stereo-strips*. ISPRS Workshop on 'Sensors and Mapping from Space 1999', September 27-30, 1999, Hanover.
- 國友 達也 (Tatuya Kunitomo), 2009. *ベクトルデータベース基準点の構築と幾何補正手法の開発*. 社会システム工学科 4 年.
- Laben et al., 2000. *Process for Enhancing the Spatial Resolution of Multispectral Imagery Using Pan-Sharpning*. U.S. Patent number 6011875, Jan. 4, 2000.
- Lintz, Joseph, and D. S. Simonett, 1976. *Remote Sensing of Environment*. Reading, MA. Addison-Wesley, p.1.
- Lillesand, T.M., and Kiefer, R.W., 1994. *Remote Sensing and Image interpretation (3rd Ed.)*. John Wiley & Sons, New York p. 533, 536-541, 567-572.
- Majumdar, J., Y. Dilip, (2002). *Implementation of Image Registration Algorithms for Real-time Target Tracking Through Video Sequences*. Defence Science Journal, Vol. 52, No. 3, July 2002, p. 227-242.
- Meyer, P., Itten, K.I., Kellenberger, K.J., Sandmeier, S., and Sandmeier, R., (1993). *Radiometric corrections of*

- topographically induced effects on Landsat TM data in alpine terrain.* Photogrammetric Engineering and Remote Sensing, 48 p. 17.
- Murai, S., (1998). *GIS Work book – Technical Course.* Japan Association of Surveyors, JAPAN
- Murai, S., et al., 1998. *Generation for 3D View Map Using Polygon Shift Method.* ISPRS Commission IV Symposium on GIS – Between Visions and Applications, IAPRS, Vol. 32/4, p. 417-420, Stuttgart, Germany.
- 濱田哲伸, 2004. 標高データを用いた不可視判定プログラム開発とその応用. 社会システム工学科 4年.
- Masataka TAKAGI, Lecture on Survey 1, section 7, “*Vector*”.
- Masataka TAKAGI, Lecture on Survey 2, section 10, “*Transformation*”.
- Mitsuhiro Kojima, Takahiro Nakagawa and Masataka Takagi, 2008. *Accuracy of 3 Dimensional Measurement by ALOS AVNIR-2.* Asian Conference on Remote Sensing 2008, 10-14 Nov 2008, Colombo, Sri Lanka.
- MOMS-2P. Modular Optoelectronic Multispectral Scanner 2 on PRIRODA, DLR, DEUTSCHES <http://www.nz.dlr.de/moms2p/> (accessed 10 Jan. 2009).
- National Academy of Sciences, 1970. *Remote Sensing with Special Reference to Agriculture and Forestry.* Washington, DC: American Geographical Society. P. 1.
- NEC/TOSHIBA, 2006. *ALOS: Product Format Description (PRISM).* NEC TOSHIBA Space Systems, Ltd.
- Proy, C., Tanre, D., and Deschamps, P.Y., 1989. *Evaluation of Topographic Effects in Remotely Sensed Data.* Remote Sensing of the Environment, 30 p.21.
- QuickBird. DigitalGlobe product guide “Quickbird Imagery Products”, USA <http://www.digitalglobe.com/index.php/6/DigitalGlobe+Products> (accessed 9 Nov.2008)
- RAMS-e DSM, (2000), Kokusai Kogyo Co., Ltd. “RAMS-e”, JAPAN <http://www.ramse3d.com/laser/index.html> (accessed 6 Nov. 2008).
- Richards, J. A., *Remote Sensing Digital Image Analysis*, Springer-Verlag, Berlin, 1999.
- Rosenholm, D., 1987. *Empirical investigation of optimal window size using the least squares image matching method.* Photogrammetria, Volume 42, Issue 3, December 1987, Pages 113-125.
- Russ, John C., *The image processing handbook*, CRC Press, Inc., USA, 1995.
- Swain, P. H., and S. M. Davis, 1978. *Remote Sensing: The Quantitative Approach.* New York: McGraw Hill, p.1.
- Tadono, et al. 2006. *Initial Results of Calibration and Validation for PRISM and AVNIR-2.* Asian Journal of geoinformatics, vol. 6, No. 4, October 2006. p. 11-20.
- Takeshi MIYATA (2005). *Establishment of Methodology for Accurate Geometric Coorection by using Centroid as Control Point*, Graduation thesis to fullfull master degree,

- Kochi University of Technology,
JAPAN.
- Takeshi MIYATA, Jong Hyeok
JEONG and Masataka TAKAGI,
(2006). *Extracting Method of Mudflow
Disaster by using Arial Photography*,
Proceedings of International
Symposium on Management
Systems for Disaster Prevention,
JAPAN.
- Tanaka, K., 1932. *The orthographic relief
method of representing hill features on a
topographic map*. Geographical
Journal, v. 79, p. 213-219.
- Tanaka, K., 1950. *The relief contour
method of representing topography on
maps*. Geographical Review, v. 40, p.
444-456.
- Tao, C.V., and Y., Hu (2001). *A
Comprehensive Study of the
Rational Function Model for
Photogrammetric Processing*. PE &
RS, Vol. 67, No. 12, p. 1347-1357.
- Touton, Th., 2003. *Error Tracking in
IKONOS Geometric Processing using a
3D Parametric Model*.
Photogrammetric Engineering &
Remote Sensing, Vol. 69, No. 1, p.
43-51.
- Tzelepis, N., and Nakos, B., 2003. *A
Study on the Lighting Factors Affecting
Relief Presentation*. Proceedings of
21st International Cartographic
Conference, p. 1343.
- Usery, E.L. and P. Altheide, 1988.
*Knowledge-Based GIS Techniques
Applied to Geological Engineering*.
Photogrammetric Engineering and
Remote Sensing, 54 (11): 1623-
1628.
- White, L. P. 1977. *Aerial Photography
and Remote Sensing for Soil Survey*.
Oxford: Clarendon Press. P.1-2.
- WorldView-1. DigitalGlobe product guide
"WorldView-1 specification", USA
<http://www.digitalglobe.com/index.php/86/WorldView-1> (accessed 9 Nov.2008)
- Yoeli, P., 1976. *Computer-aided relief
presentation by traces of inclined planes*.
The American Cartographer, v. 3, p.
75-85.
- Yoeli, P. 1983. *Shadowed contours with
computer and plotter*. The American
Cartographer 10:101-110.
- Zhou, G., and R. Li, (2000). *Accuracy
Evaluation of Ground Control Points
from IKONOS High-Resolution Satellite
Imagery*. Photogrammetric
Engineering and Remote Sensing,
Vol. 66, No. 9, p. 1103-1112.

INDEX

A

Aerial photo, ii, 3, 7, 9, 10, 52, 53, 132, 151
Affine, ii, iii, vii, ix, xi, xv, 3, 14, 18-21, 24-27, 31, 32, 36, 37, 40, 42, 43, 45, 46, 61, 70-73, 75, 77, 92, 94, 98
Application, xv, 1, 3, 5, 53, 94, 132, 134, 146, 150

B

Backward, iv, 12, 13, 24, 25, 27
Band ratio, 17, 57-58, 92
Blank value, 57

C

CP, iv, x, xi, xii, xiii, xvi, 24-27, 31, 32, 36-41, 46, 62, 63, 66, 67, 69-75, 77, 80, 87, 94, 97-100

D

DGPS, xvi, 2, 6
Distribution, ii, iii, iv, vii, viii, ix, x, xi, xiii, 2, 3, 5, 7, 9, 24-26, 28-30, 34-46, 54, 63, 66, 71, 75
DSM, iv, xvi, 4, 7, 9, 10, 15-17, 36, 38, 53-56, 59, 63, 77, 78, 134, 150

E

Error propagation, iv, vii, viii, xiii, 36, 37, 40, 71, 72
Error vector, ii, iv, viii, xi, xii, xv, 4, 7, 36, 42-45, 60, 72, 74, 100

F

Forward look, 3

G

GCP, ii-v, vii-xiv, xvi, 2-9, 14-16, 24-42, 45, 46, 53, 55, 62-67, 69-75, 77, 78, 80, 92-94, 98, 99, 100, 134-140
GPS, xvi, 2, 6, 33, 34, 92, 93, 140
Geometric transformation, ii, iii, viii, xv, 7, 46, 71, 72, 74, 75, 78, 94, 100, 133

I

ICP, ii, iv, viii, xi, xiv, xvi, 3, 4, 6, 7, 8, 9, 16, 53, 64-68, 76-93
IKONOS, ix, xii, xvi, 1, 3, 4, 9-11, 17, 31, 32, 38, 46, 56-58, 63, 66, 75, 77, 79, 132, 148, 149, 151
Image registration, ii, iii, iv, vi, viii, ix, 1-6, 14-16, 32, 33, 35, 36, 54, 59, 65, 69, 77, 78, 92, 149
Image simulation, viii, ix, xi, xii, 4, 8, 55, 58, 59, 77, 78, 92

L

Least squares, xiii, xv, xvi, 3, 4, 7, 15, 19, 53, 60, 61, 65, 66, 69, 72, 73, 76, 110, 134, 148, 150

N

Nadir look, iv, 3, 11
Number of GCP, ii, x, xi, xiii, 3, 4, 26, 28-30, 35, 36, 63, 92

P

PRISM, ii, ix, x, xii, xiii, xvi, 1, 3-5, 9, 11-13, 24-31, 33, 37, 38, 58, 59, 63, 64, 65, 75, 84, 92, 146, 148, 150
Projective, 12, 15-17, 21, 25, 27-28, 49-51, 55-57, 63-65, 92

R

Random error, vi, viii, 36, 37, 39, 71, 72
Reference image, ii, iv, viii, xi, xii, 4, 5, 7, 10, 12, 15-17, 38, 46, 53-57, 59, 61-63, 66, 75, 76, 78, 79, 134

S

Shade and Shadow, viii, 4, 6, 7, 10, 15-17, 53-57, 59
Seperated, iii, vii, x, xi, 15, 18, 23-27, 31, 45, 46, 59, 69, 70-73, 75, 77, 130
Spectral reflectance, 1, 7, 49, 58, 92, 93
Sun, xi, xii, 2, 6, 7, 10, 11, 13, 15, 17, 48, 50, 51-57

T

Target image, ii, viii, 4, 5, 7, 9-13, 15-17, 53-55, 59-63, 75, 78, 79, 134

V

VRS-GPS, ii, iii, iv, x, xvii, 2, 6, 8, 24, 32, 33, 34, 67, 73, 75, 77, 79, 80

