

**AUTOMATIC ALIGNMENT OF 3D MULTI-SENSOR
POINT CLOUDS**

RAVI ANCIL PERSAD

A DISSERTATION SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAMME IN EARTH AND SPACE SCIENCE

YORK UNIVERSITY

TORONTO, ONTARIO

AUGUST 2017

© Ravi Ancil Persad, 2017

Abstract

Automatic 3D point cloud alignment is a major research topic in photogrammetry, computer vision and computer graphics. In this research, two keypoint feature matching approaches have been developed and proposed for the automatic alignment of 3D point clouds, which have been acquired from different sensor platforms and are in different 3D conformal coordinate systems.

The first proposed approach is based on 3D keypoint feature matching. First, surface curvature information is utilized for scale-invariant 3D keypoint extraction. Adaptive non-maxima suppression (ANMS) is then applied to retain the most distinct and well-distributed set of keypoints. Afterwards, every keypoint is characterized by a scale, rotation and translation invariant 3D surface descriptor, called the ‘radial geodesic distance-slope histogram’. Similar keypoints descriptors on the source and target datasets are then matched using bipartite graph matching, followed by a modified-RANSAC for outlier removal.

The second proposed method is based on 2D keypoint matching performed on height map images of the 3D point clouds. Height map images are generated by projecting the 3D point clouds onto a planimetric plane. Afterwards, a multi-scale wavelet 2D keypoint detector with ANMS is proposed to extract keypoints on the height maps. Then, a scale, rotation and translation-invariant 2D descriptor referred to as the ‘Gabor, Log-Polar-Rapid Transform’ descriptor is computed for all keypoints. Finally, source and target height map keypoint correspondences are determined using a bi-directional nearest neighbour matching, together with the modified-RANSAC for outlier removal.

Each method is assessed on multi-sensor, urban and non-urban 3D point cloud datasets. Results show that unlike the 3D-based method, the height map-based approach is able to align source and target datasets with differences in point density, point distribution and missing point data. Findings also show that the 3D-based method obtained lower transformation errors and a greater number of correspondences when the source and target have similar point characteristics. The 3D-based approach attained absolute mean alignment differences in the range of 0.23m to 2.81m, whereas the height map approach had a range from 0.17m to 1.21m. These differences meet the proximity requirements of the data characteristics and the further application of fine co-registration approaches.

Acknowledgements

I would like to express the utmost thanks and appreciation to my PhD. supervisor, Dr. Costas Armenakis for his constant support, guidance, patience and mentorship during my time at York University. Under his numerous years of tutelage, I have learnt many things about academics and about life in general. I cannot thank him enough for the opportunity to do so. I'd also like to sincerely thank Dr. Gunho Sohn for his years of support. He and Dr. Armenakis graciously welcomed me into the GeoICT Lab when I first came to Canada. I have thoroughly enjoyed working with and learning from him. I also wish to extend my thanks and appreciation to Dr. Regina Lee, Dr. Burton Ma, Dr. Derek Lichti and Dr. Franz Newland for their valuable time and effort on reviewing this dissertation. I also thank my colleagues at the GEOICT lab with whom I have worked on several interesting research projects over the years.

I would like to sincerely thank Dr. James Elder, Department of Electrical Engineering and Computer Science, York University, for providing the Aeryon Scout UAV video data, and Mike Demuth (Geological Survey of Canada) and Alexander Chichagov (Canada Centre for Mapping and Earth Observation), both from Natural Resources Canada (NRCan) for providing the Columbia Icefield datasets. Teledyne Optech and First Base Solutions are much thanked for providing the urban datasets.

Finally, I dedicate this work to my parents, Sundar and Deokie Persad and thank them for their endless support.

Table of Contents

Abstract	ii
Acknowledgements	iv
Table of Contents	v
List of Tables	ix
List of Figures	xi
List of Acronyms	xv
1 Introduction	1
1.1 Initial alignment versus refined alignment	4
1.2 Initial alignment: global versus local methods	7
1.3 Overview and objectives.....	8
1.4 Contributions	10
1.5 Organization	11
2 Related Works on Initial Point Cloud Alignment	14
2.1 3D Descriptor-based methods	15
2.1.1 3D keypoint extraction.....	15
2.1.2 Matching of 3D keypoints using descriptors	18
2.2 3D Non-descriptor-based methods	24

2.3	2D image-based methods.....	27
2.4	Summary.....	29
3	A 3D-based Approach for Point Cloud Alignment	31
3.1	3D-based Point Cloud Alignment Methodology.....	32
3.2	Extraction of 3D Surface Keypoints	34
3.2.1	Scale invariance for 3D keypoints.....	37
3.2.2	Keypoint refinement by adaptive non-maxima suppression.....	41
3.3	3D Surface Descriptors for Keypoints.....	44
3.3.1	Rigid invariance for local 3D descriptors.....	44
3.3.2	Local 3D surface description.....	46
3.3.3	3D Keypoint matching using RGSB descriptor	50
3.3.4	Removal of 3D keypoint correspondence outliers	51
3.4	Summary.....	54
4	A Height Map-based Approach for Point Cloud Alignment	55
4.1	Height Map-based Point Cloud Alignment Methodology.....	56
4.2	Multi-scale 2D keypoint extraction.....	58
4.2.1	2D keypoint extraction using DTCWT	60
4.3	Scale, rotation and translation invariant 2D keypoint descriptor	66
4.3.1	Log-polar sampling and mapping for 2D scale and rotation invariance	68
4.3.1.1	Generation of Gabor filter-based derivatives	71

4.3.2	Descriptor invariance to 2D cyclic-shifts using the Rapid Transform	73
4.3.3	2D keypoint matching using GLP-RT descriptor	77
4.4	Summary.....	79
5	Results and Analysis	80
5.1	Results for Method 1: 3D-based Point Cloud Alignment	81
5.1.1	Empirical selection of RGSH descriptor bin size.....	82
5.1.2	Case 1: Same sensor datasets, different coordinate systems	88
5.1.3	Case 2: Different sensor datasets, different coordinate systems	94
5.2	Results for Method 2: Height Map-based Point Cloud Alignment.....	102
5.2.1	Experimental datasets	102
5.2.1.1	Dataset 1 (<i>Urban, Loc1</i>).....	103
5.2.1.2	Dataset 2 (<i>Urban, Loc2</i>).....	104
5.2.1.3	Dataset 3 (<i>Non-Urban, Loc3</i>)	106
5.2.1.4	Tuning and testing datasets	107
5.2.2	Empirical tuning: Selection of GLP-RT descriptor parameters	108
5.2.2.1	The minimum radius	110
5.2.2.2	The maximum radius	111
5.2.2.3	The number of rays and number of rings	112
5.2.3	Testing experiment: Assessment of the 2D height map approach with other 2D keypoint detectors and descriptors	114
5.2.4	Accuracy analysis of 2D height map-based point cloud co-registration.....	120

5.3 Overall assessment of the proposed 3D-based and height map-based co- registration methods	128
5.3.1 Observations for <i>real</i> datasets 1 and 2 (<i>Urban, Loc1 and Urban, Loc2</i>)....	129
5.3.2 Observations for <i>real</i> dataset 3 (<i>Non-Urban, Loc3</i>)	132
5.4 Computation time	134
6 Conclusions	136
6.1 Research outcomes.....	137
6.1.1 Summary of the 3D-based point cloud alignment method.....	137
6.1.2 Summary of the Height map-based point cloud alignment method	139
6.2 Recommendations for future work	140
References	143
Appendix A : Bipartite matching using the Hungarian method	155
Appendix B : Rapid Transform	158

List of Tables

5.1 Manually-defined transformation parameters used for generating target point clouds of the 4 training sites in the tuning dataset	85
5.2 Descriptor matching for various keypoints on Figure 5.4	92
5.3 Co-registration result for ‘Case 1’ Urban dataset	93
5.4 Co-registration result for ‘Case 1’ Icefield (Non-Urban) dataset	93
5.5 Average Angular and Translation errors for ‘Case 1’ datasets	94
5.6 Co-registration result for ‘Case 2’ Urban dataset	96
5.7 Co-registration result for ‘Case 2’ Icefield (Non-Urban) dataset	96
5.8 Average Angular and Translation errors for ‘Case 2’ datasets	98
5.9 Simulated and real source and target datasets which are used for the empirical tuning	109
5.10 Simulated and real source and target datasets which are used for the testing experiment.....	109
5.11 Optimal GLP-RT descriptor parameters after tuning.....	112
5.12 Combinations of 2D keypoint detectors and 2D descriptors evaluated on the height map testing datasets	116
5.13 Co-registration result for <i>real</i> test dataset 1 (<i>Urban, Loc1</i>)	122
5.14 Co-registration result for <i>real</i> test dataset 2 (<i>Urban, Loc2</i>)	122
5.15 Co-registration result for <i>real</i> test dataset 3 (<i>Non-Urban, Loc3</i>)	123
5.16 Co-registration errors using <i>proposed multi-scale wavelet</i> 2D keypoint detector	

and <i>GLP-RT</i> descriptor.....	131
5.17 Co-registration errors using <i>proposed surface curvature</i> -based 3D detector and <i>RGSH</i> descriptor.....	131
5.18 Co-registration errors using <i>3D-SIFT</i> 3D keypoint detector and <i>FPFH</i> descriptor	131
5.19 Co-registration errors using <i>3D-SIFT</i> 3D keypoint detector and <i>SHOT</i> descriptor	132
5.20 Comparison of 3D keypoint detectors for ‘ <i>real dataset 3</i> ’ based on localization accuracy and similarity of local keypoint scales.....	134

List of Figures

1.1 Illustration of the co-registration problem for 3D point cloud datasets from multiple sensors	2
1.2 Distinction amongst various 3D point cloud alignment (co-registration) approaches.....	5
1.3 Example of two point cloud datasets from different sensors (left: UAV, right: Mobile laser scanner) with varying point characteristics such as different point density, point distribution and point details.....	9
2.1 Different approaches for the initial alignment of 3D point clouds	15
2.2 Concept of Spin Image point cloud descriptor formation.....	19
2.3 Concept of FPFH formation showing the triplet angular relation (α , θ , ϕ) between \mathbf{p}_s (the keypoint) and \mathbf{p}_t (neighbouring point)	20
2.4 Illustration of the SHOT descriptor	21
3.1 Concept of keypoint matching between source and target point clouds.....	33
3.2 Workflow of the proposed 3D-based point cloud co-registration approach	35
3.3 Workflow for the proposed 3D keypoint extraction process.....	38
3.4 Concept of obtaining scale-invariant keypoints.....	40
3.5 Example of keypoint extraction on point cloud surfaces. a) Before ANMS b) After ANMS.....	43
3.6 Local keypoint neighbourhood on the surface mesh. Geodesic paths running in radial pattern from keypoint (neighbourhood centroid) to all its neighbouring	

points (in black) are shown	47
3.7 Illustration of 1-ring mesh neighbourhood around a point \mathbb{P}_j on the surface mesh and the geometry for obtaining its slope	47
3.8 Illustration of the 2D radial geodesic distance-slope histogram (the gray scale shows binning frequency)	49
3.9 Example of bipartite graph for keypoint point matching	51
4.1 Overview of the height map image point matching approach for co-registering 3D multi-sensor point clouds	57
4.2 Scale-space representation of a height map produced by the dual tree complex wavelet transform at three levels of decomposition	62
4.3 Keypoint energy maps generated at each of the three decomposition levels	64
4.4 Keypoint extraction results. (a) Initial keypoints (before ANMS). (b) Final keypoints (after ANMS)	66
4.5 Example of log-polar sampling and mapping	70
4.6 Concept of applying Rapid Transform to correct cyclical shift between log-polar descriptors on corresponding keypoints	74
4.7 Computation steps of the 1D rapid transform based on the signal flow (or ‘butterfly’) structure when $K = 8$	76
4.8 Concept of bi-directional keypoint descriptor matching showing a successful correspondence (dashed arrows) and an unsuccessful correspondence (solid arrows)	78
5.1 Urban DSMs used for co-registration experiments to evaluate the proposed 3D-	

based point cloud alignment method	83
5.2 Icefield (Non-Urban) DSMs used for co-registration experiments to evaluate the proposed 3D-based point cloud alignment method.	84
5.3 <i>Recall vs. 1-precision</i> graphs for selecting optimal bin size of the RGS descriptor across a range of coarse to dense bin resolutions using the DSM tuning dataset	87
5.4 Keypoint matching under scaling, rotation and translation	90
5.5 Alignment of urban test scene (<i>Urban, Loc2</i>)	99
5.6 Alignment of Saskatchewan Glacier test site (<i>Non-Urban, Loc3</i>)	100
5.7 Alignment differences between source and target point clouds for ‘Case 2’ datasets.	101
5.8 Dataset 1 (<i>Urban, Loc1</i>) used to evaluate the proposed height map-based point cloud alignment method.	104
5.9 Dataset 2 (<i>Urban, Loc2</i>) used to evaluate the proposed height map-based point cloud alignment method	105
5.10 Dataset 3 (<i>Non-Urban, Loc3</i>) used to evaluate the proposed height map-based point cloud alignment method	106
5.11 <i>Recall vs. 1-precision</i> graphs for selecting optimal GLP-RT descriptor parameters using the tuning datasets	113
5.12 <i>Recall vs. 1-precision</i> graphs of the six test datasets using different keypoint detectors/descriptor combinations from Table 5.12	116
5.13 Height map point matching results for <i>real</i> test dataset 1 using proposed multi-	

scale keypoint extraction and GLP-RT descriptor.....	117
5.14 Height map point matching results for <i>real</i> test dataset 2 using proposed multi-scale keypoint extraction and GLP-RT descriptor.....	118
5.15 Height map point matching results for <i>real</i> test dataset 3 using proposed multi-scale keypoint extraction and GLP-RT descriptor.....	119
5.16 Co-registration of point clouds for <i>real</i> test dataset 1	124
5.17 Co-registration of point clouds for <i>real</i> test dataset 2	125
5.18 Co-registration of point clouds for <i>real</i> test dataset 3	126
5.19 Alignment differences between source and target point clouds for the <i>real</i> test datasets	127
B.1 Example showing rapid transform on a pair of synthetic images with translation differences	158

List of Acronyms

RGSH : Radial Geodesic distance-Slope Histogram

GLP-RT: Gabor, Log-Polar-Rapid Transform

UAV: Unmanned Aerial Vehicles

LIDAR : Light Detection and Ranging

ICP: Iterative Closest Point

PCA: Principal Component Analysis

RANSAC: RANdom Sample And Consensus

SIFT: Scale Invariant Feature Transform

DoG : Difference-of-Gaussian

SURF : Speeded Up Robust Features

ISS : Intrinsic Shape Signature

MRI: Magnetic Resonance Imaging

CT : Computed tomography

FPFH: Fast Point Feature Histograms

SHOT : Signature of Histograms of Orientations

TLS : Terrestrial Laser Scanning

MLS: Mobile Laser Scanning

ALS : Airborne Laser Scanning

NDT: Normal Distributions Transform

4PCS: 4-Point Congruent Set

SVD: Singular Value Decomposition

GPS: Global Positioning System

SPDF: Scale Parameter-Dependent Function

KP: Keypoint

ANMS: Adaptive Non-Maxima Suppression

NMS : Non-Maxima Suppression

DSID: Dense Scale Invariant Descriptor

DTCWT: Dual Tree Complex Wavelet Transform

DWT: Discrete Wavelet Transform

FFT: Fast Fourier Transform

RT: Rapid Transform

LP : Log-Polar

DSM: Digital Surface Model

RMSE: Root Mean Square Error

AMRE: Absolute Mean Rotational Error

AMTE: Absolute Mean Translation Error

ASTER: Advanced Spaceborne Thermal Emission and Reflection Radiometer

GDEM: Global Digital Elevation Model

WV-2: WorldView-2

TP: True Positive

FP: False Positive

LS3D: Least Squares 3D Surface Matching

HKS: Heat Kernel Signature

NNDR: Nearest Neighbour Distance Ratio

1. Introduction

Automatic alignment (or co-registration) of 3D point clouds is an active area of research in numerous fields of study including photogrammetry, computer vision, laser scanning, 3D modelling and computer graphics. Co-registration is the process of aligning multiple shapes (two or more) in a common coordinate system. It is typically applied to overlapping pairs of 2D images or 3D point cloud models. This research concentrates on addressing the latter issue of automated 3D pairwise point cloud co-registration.

Typical registration tasks usually require the alignment of 3D point clouds that are: i) multi-temporal (i.e., collected at different epochs) and/or ii) acquired from various sensors (e.g., aerial, terrestrial or mobile laser scanners, satellite systems and unmanned aerial vehicles (UAV)) and/or iii) acquired from different viewpoints of the same or similar sensors. More specifically, the co-registration of point cloud datasets is required for 3D surface completion or reconstruction from partially overlapping 3D points located in different coordinate systems (Figure 1.1). Alignment of multi-sensory data has numerous applications in 3D building and terrain modelling, change detection and map-revision in urban and non-urban environments, cultural heritage, crime scene/accident reconstruction, and mapping of open-pit mines.

The co-registration process is based on the mathematical mapping that projects the ‘source’ point cloud to its ‘target’ point cloud. The mathematical mapping is expressed by the transformation relationship (e.g., scale, rotation, translation and shape deformation) between the coordinate systems of the two datasets. Generally, there are

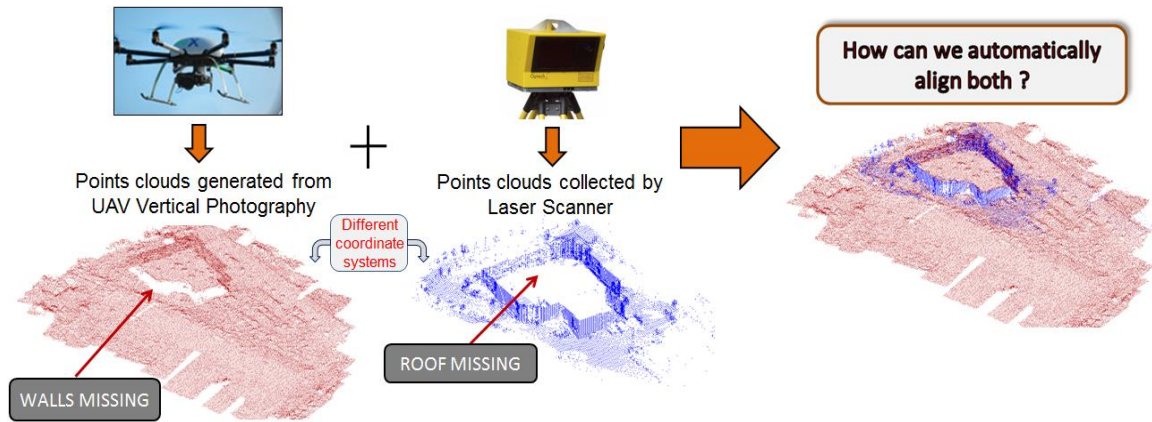


Figure 1.1: Illustration of the co-registration problem for 3D point cloud datasets from multiple sensor platforms.

three categories of 3D coordinate transformations which are commonly utilized for 3D point cloud alignment: i) 3D conformal, ii) 3D rigid, and iii) 3D non-rigid.

The 3D conformal transformation accounts for uniform scale, 3 rotations and 3 translations. The 3D conformal transformation is also referred to as ‘3D similarity transformation’, ‘Helmert transformation’ or ‘7-parameter transformation’ (Andrei, 2006). The 3D rigid transformation estimates 3 rotations and 3 translations. It assumes no scale change between the two datasets. 3D non-rigid transformations such as the affine transformation and spline functions (Jian and Vemuri, 2005) also model the shape deformation between the source and target. There are different types of 3D affine transformation solutions (Lehmann et al., 2014), which vary in terms of the number of estimated transformation parameters, for example: i) 12-parameters (3 rotation angles, 3 translations, 3 skew factors (i.e., shearing along each axis) and different scale factors

along each axis), and ii) 9-parameters (3 rotation angles, 3 translations and different scale factors along each axis).

Traditionally, co-registration is achieved by the manual selection of user-specified corresponding point features, which is then used as input to compute the transformation parameters. However, this is a tedious process particularly when: i) there are a large number of datasets to be co-registered, ii) when datasets contain a large number of points and iii) when the determination of corresponding features is difficult to establish between two point cloud datasets. To overcome these difficulties, an automated process is highly desirable. The challenge in this process includes the automatic extraction and correspondence of the distinct point features. The extraction of point features relates to the automated detection of distinct ‘keypoints’ (e.g., points of sharp topographic variation such as building corners). Correspondence relates to the automatic matching of source keypoints to their corresponding target entities in the 3D space, which are then used to solve for the desired mapping parameters. When there is significant variation between the two point cloud datasets to be aligned, for example, large differences in scale, rotation, translation, and point characteristics (e.g., point density and spatial distribution), it is challenging to establish correct correspondences.

In this dissertation, the source and target point cloud datasets to be aligned differ in terms of a 3D conformal displacement (Equation 1.1) and in terms of point characteristics.

$$Target_{point\ clouds} = sR(Source_{point\ clouds}) + T \quad (1.1)$$

where,

- s is the scaling factor,
- R is a 3x3 orthogonal rotation matrix formed using the 3 rotation angles (ω , φ , κ) about the x , y and z -axes respectively,
- T is a 3x1 translation vector with x , y and z components.

A minimum of three point correspondences are required to determine the scale, rotation and translation 3D conformal parameters. The parameters are commonly estimated through the use of a least squares solution which minimizes the sum of squares of the spatial distances between the source to target point correspondences, thereby estimating the parameters. The solution can be either linear, closed form (Horn, 1987) or non-linear, iterative (Luhmann et al., 2006). Upon estimation of the 3D conformal mapping parameters, the final step for the alignment is to transform the source point clouds into the coordinate system of the target point clouds using Equation 1.1.

1.1 Initial alignment versus refined alignment

There are two main phases for automated, pairwise 3D point cloud co-registration as illustrated in Figure 1.2: i) the *initial* alignment, and ii) the *refined* alignment. The former case handles the co-registration of point cloud datasets in different coordinate systems and there is no proximate matching between the source and target. The latter case assumes that an initial alignment has been applied and there is an existing, approximate

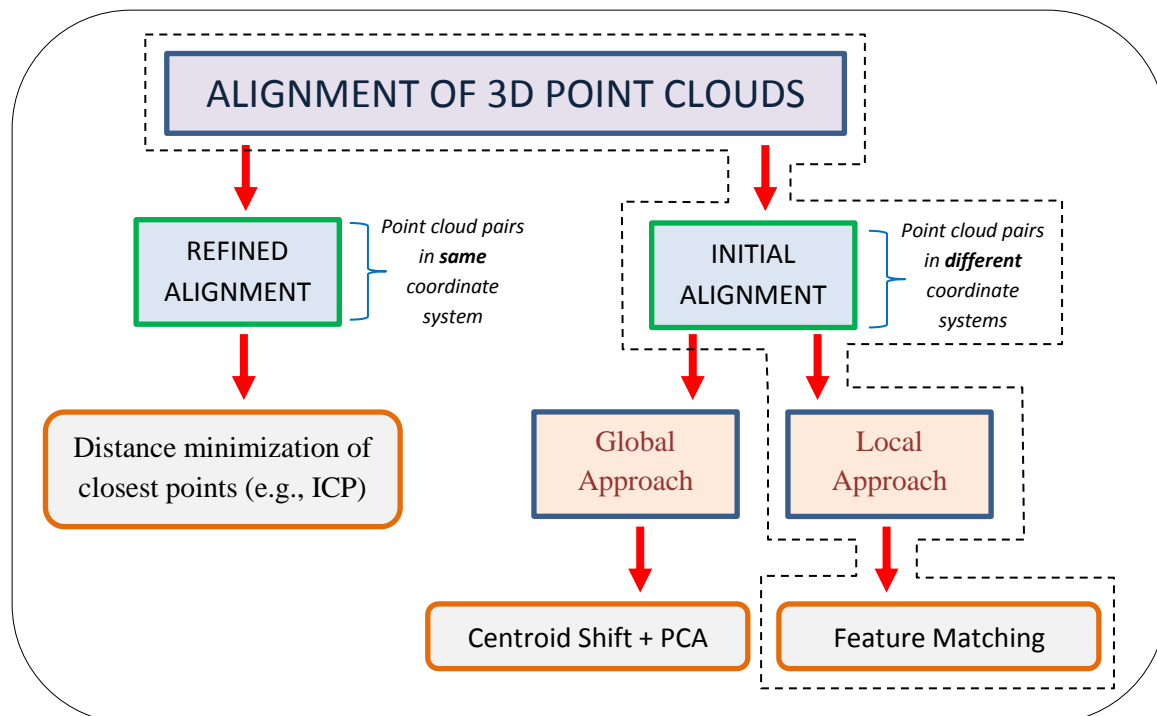


Figure 1.2: Distinction amongst various 3D point cloud alignment (co-registration) approaches (this work concentrates on the framework marked by the dashed outline).

co-registration between the source and target datasets. Both require the computation of a mathematical mapping between two point cloud datasets.

For over two decades, the refined alignment problem has received considerable attention since the development of the influential ‘Iterative Closest Point’ (ICP) algorithm (Besl and McKay, 1992; Chen and Medioni, 1992). Rusinkiewicz and Levoy (2001) provide an overview of many ICP variants. Bouaziz et al. (2013) developed the so-called ‘Sparse ICP’ which is less sensitive to outliers than the classical ICP. In the photogrammetric community, Gruen and Akca (2005) proposed an alternative to the ICP

referred to as ‘Least Squares 3D Surface Matching’ (LS3D). Resembling the ICP approach, LS3D also iteratively minimizes the sum of squares of Euclidean distances between two point cloud datasets. However, LS3D differs from ICP in its formulation. ICP computes the transformation parameters using Horn’s linear least squares closed-form solution (Horn, 1987), whereas LS3D uses the Generalized Gauss-Markov nonlinear model. Instead of using the closest point concept for correspondences as done in ICP, Bae and Lichti (2008) developed the ‘Geometric Primitive ICP’ method, which instead uses the point normal vector information together with change in surface curvature for point cloud matching. In more recent times, another class of refinement techniques are ‘non-rigid’ 3D point cloud alignment approaches (Chui and Rangarajan (2003); Lin et al. (2016)). ‘ICP’-based methods assume that the source and target differ in terms of a 3D conformal or 3D rigid transformation. However, ‘non-rigid’ techniques also handles deformation changes between the pairwise point clouds to be co-registered.

‘Refinement-based’ registration methods strongly depend on a very good initial point cloud alignment with sufficient overlap between the source and target. The ‘refinement’ methods do not require an intricate feature-matching step as they are typically based on minimizing the Euclidean distance between the closest points. If the initial alignment is inaccurate, the refinement-based approaches are prone to various mis-registration factors such as local minima solutions and exhaustive searching in the solution space, which negatively affects computational efficiency. Motivated by these issues, this research work concentrates on addressing the initial 3D point cloud co-registration problem.

1.2 Initial alignment: global versus local methods

As shown in Figure 1.2, there are two known primary approaches for initial 3D-to-3D point cloud alignment: i) global techniques and ii) local techniques, (Castellani and Bartoli, 2012). The global-based initial alignment revolves around the use of the principal component analysis (PCA) of the point clouds. The translation can be estimated by the difference in centroids of the source and target data. Then, PCA is used to approximate the rotation required to align the coordinate systems of the source and target point clouds. The global scale factor can be derived based on the ratio of the respective largest distances between the source and target data.

On the other hand, local techniques are based on the definition of local surface properties (i.e., descriptors) for automatically detected ‘key geometric features’ on both the source and target point clouds (Note: geometric features can include points, lines, curves or planes). The similarities of the descriptors are then assessed for the determination of corresponding key geometric features. The global co-registration approach suffers when there is partial overlap and/or shape deformation between the source and target surfaces. For instance, the centroids of both shapes may differ due to deformations or when the source and target have different coverage. This affects the estimation of translation parameters. Difference in shape creates similar problems when attempting to estimate scale and rotation parameters. Therefore, it can be argued that the local alignment techniques are better suited for co-registering the ‘stable’ parts of the point cloud surfaces, for instance, when dealing with natural terrain datasets which may have undergone deformation, for example, landslides, flow of glaciers, etc.

1.3 Overview and objectives

3D point clouds have varying characteristics and be represented in various ways. They are represented in 3D or 2D formats such as: i) as raw 3D points, or as ii) interpolated, 2D height (or depth) map raster images. As shown in Figure 1.3, source and target point cloud datasets can also differ in terms of characteristics such as: i) point density (e.g., dense versus sparse point spacing), ii) point distribution (e.g., regular, gridded points versus irregular, non-gridded points), and iii) missing point data (i.e., data gaps/holes), possibly caused by occlusions or by different sensor viewpoint perspectives during data acquisition. To handle these different cases (i.e., differences in data representation and characteristics) two independent approaches for the automatic co-registration of point clouds in different 3D conformal coordinate systems are investigated and explored.

Both of the implemented methods are local alignment type techniques which follow an automated feature matching pipeline that includes three main phases: i) *feature extraction*, ii) *feature description* and iii) *feature correspondence*. The proposed methods are based on extracting and matching distinct point landmarks, i.e., *keypoints* on the source and target point clouds.

Although both proposed approaches adopt a similar feature matching workflow, their inherent individual components are unique, i.e., the techniques used for keypoint extraction, keypoint descriptor formation and keypoint matching are different. This stems from the two different ways in which the point clouds can be represented, i.e., either as 3D points or as interpolated, height map 2D raster images.

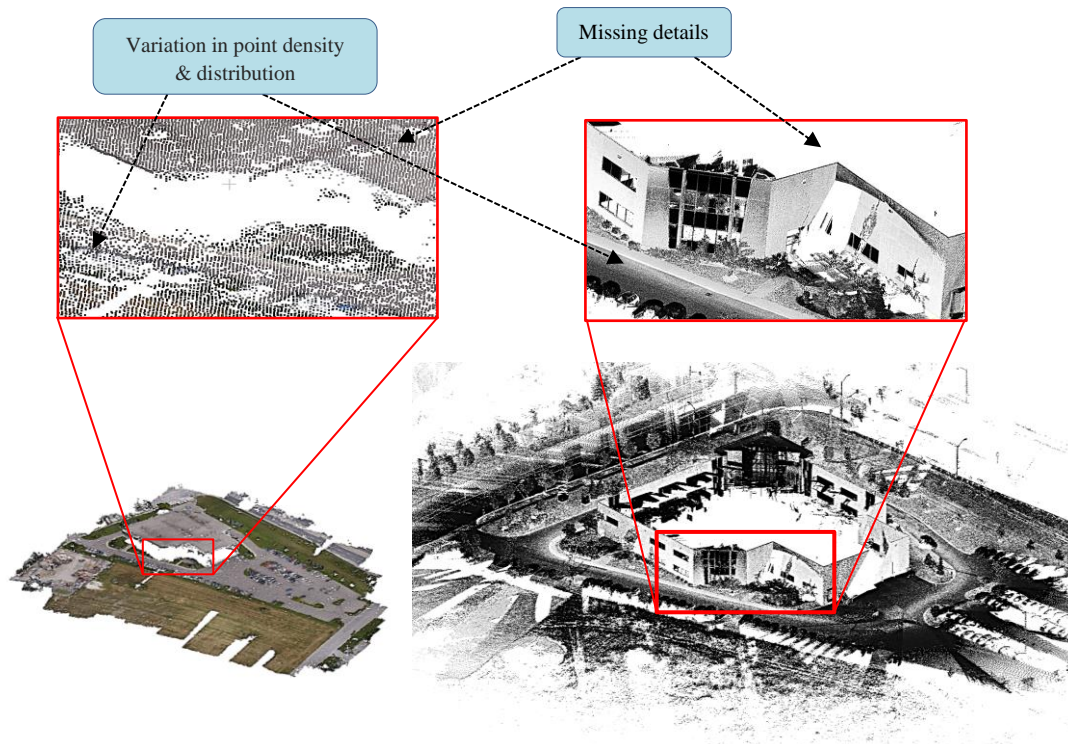


Figure 1.3: Example of two point cloud datasets from different sensors (left: UAV, right: Mobile laser scanner) with varying point characteristics such as different point density, point distribution and point details.

In the first proposed approach, feature matching is performed entirely in the original 3D point cloud space, whilst in the second method, the feature matching process is applied to the planimetric, height map projection (i.e., 2D image representation) of the 3D point clouds. For the latter, even though feature matching is performed in the 2D domain, the resulting matched points also have an associated Z or depth component, thereby facilitating 3D to 3D co-registration. The objectives of this dissertation are:

- i) To develop a 3D-based feature matching approach for co-registering 3D point clouds in different 3D conformal coordinate systems.
- ii) To develop a height map-based feature matching approach for co-registering 3D point clouds in different 3D conformal coordinate systems.
- iii) To individually evaluate the experimental findings of each approach on urban and non-urban datasets with different point cloud characteristics.
- iv) To assess the performance of both methods relative to each other, as well as with existing, state-of-the-art approaches.

1.4 Contributions

This research work contributes to the alignment of 3D point clouds in the geomatics fields of photogrammetry, remote sensing, laser-scanning and geographic information systems and incorporates multi-sensor and multi-temporal, urban and non-urban datasets. In this section, the main contributions in each of the two proposed 3D point cloud alignment methods are listed.

The contributions in the *3D-based* co-registration method are:

- The development of a scale-invariant 3D keypoint feature extraction method using morphological properties, specifically the local surface curvature.
- The development of a scale, rotation and translation invariant 3D keypoint surface descriptor referred to as the radial geodesic distance-slope histogram (RGSH).

- The use of bipartite graph descriptor matching for establishing 3D keypoint feature correspondences without the need for user-specified thresholds. A threshold-free, RANSAC outlier detection algorithm is then used to filter incorrect keypoint correspondences (i.e., outliers).

The contributions in the *Height map-based* co-registration method are:

- The development of a multi-scale, wavelet-based 2D keypoint extraction method on the height map image representations of the 3D point clouds.
- The development of a scale, rotation and translation invariant 2D keypoint descriptor referred to as the Gabor, Log-Polar-Rapid Transform (GLP-RT) descriptor.
- The use of bi-directional, nearest neighbour descriptor matching for establishing height map keypoint correspondences, without the need for user-specified thresholds.

1.5 Organization

The remaining chapters in this dissertation are organized as follows:

Chapter 2: A literature review of relevant works related to initial 3D point cloud alignment techniques is discussed. These include a survey of: i) 3D descriptor-based

point cloud co-registration methods, ii) 3D non-descriptor-based point cloud co-registration methods and iii) 2D-image based point cloud co-registration methods.

Chapter 3: This chapter covers the proposed 3D-based point cloud alignment approach. An automated 3D feature matching approach is presented. This is achieved by extracting scale-invariant 3D keypoints and generating their 3D local surface descriptors. To match the 3D keypoints, a one-to-one correspondence approach based on bipartite graphs is used. To filter outliers (i.e., incorrect keypoint correspondences), a threshold-free modified-RANSAC is applied. Finally, the 3D conformal transformation parameters are determined using the established correspondences.

Chapter 4: The second proposed height map-based automated approach for 3D point cloud alignment is detailed in this chapter. Unlike the first method, whose feature matching process is implemented entirely in the 3D domain, this approach instead uses 2D height map images of the 3D point clouds to find correspondences. Prior to co-registration, source and target height map images are generated directly from the source and target 3D point cloud datasets respectively. This is achieved by projecting and interpolating the 3D point cloud dataset onto the x,y -plane. Afterwards, 2D keypoints are extracted on both height map image pairs using a multi-scale wavelet technique. This is followed by generation of scale, rotation and translation-invariant 2D keypoint descriptors. Source and target descriptors are matched using a bi-directional nearest neighbour search in the feature space. Then, the modified-RANSAC developed in

Chapter 3 is applied to remove keypoint correspondence outliers. Finally, the 3D conformal transformation parameters are determined using the established correspondences.

Chapter 5: This chapter presents experimental results for each of the two proposed co-registration approaches. The methods are evaluated through comparisons with reference data, reference 3D conformal transformation parameters. Experiments are also carried out to directly evaluate the performance of both proposed methods with each other, as well as with existing state-of-the-art 3D point cloud co-registration approaches.

Chapter 6: A summary of the contributions and research findings are outlined in this chapter. Also discussed are suggestions for future work and potential improvements.

2. Related Works on Initial Point Cloud Alignment

This chapter provides an overview of existing work related to the initial 3D point cloud alignment problem. In particular, a review of methods used for solving initial 3D point cloud co-registration is discussed from Sections 2.1 to 2.3.

Automatic estimation of scale and the six 3D rigid parameters between point clouds is a challenging problem. For initial point cloud alignment, it is assumed that there is no prior knowledge of the 3D conformal transformation parameters (i.e., single global scale factor, 3D rotation angles and 3D translations). However, in some of the reviewed literature, the scale factor is assumed to be known and only the six rigid parameters are considered as the unknowns to be computed. Instances of such cases for the reviewed literature will be identified in this chapter. If scale is assumed to be known, the matching (or correspondence) problem is greatly simplified, since geometric elements such as lengths, distance between features and surface area can all be utilized to find correspondences.

There are various approaches one can apply to achieve initial source to target 3D point cloud co-registration. These can be classified into three categories (Figure 2.1): i) 3D descriptor-based methods, ii) 3D non-descriptor-based- methods and iii) 2D image-based methods. There are three general steps to solve the alignment problem: detection/extraction of key geometric features, matching/correspondence of these features

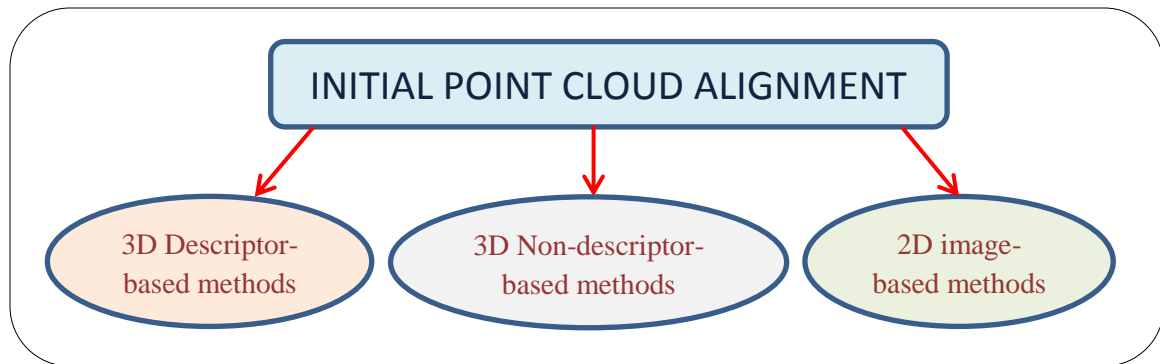


Figure 2.1: Different approaches for the initial alignment of 3D point clouds.

and assessment of the correspondences. These tasks are explicit or implicit depending on the co-registration approach utilized.

2.1 3D Descriptor-based methods

2.1.1 3D keypoint extraction

Descriptor-based methods are typically applied in 3D feature matching workflows. They usually rely on the extraction of salient key-features (e.g., 3D keypoints) on the point cloud surface. For these keypoints, descriptors are formed by utilizing various types of local neighbourhood shape attributes of the point cloud. Similar descriptors on source and target point clouds can then be matched using a similarity cost function to find corresponding keypoints.

Interest points or keypoints are well utilized for matching and registration problems in various 2D image-processing applications such as object recognition (Lowe, 1999; Azad

et al., 2009) and scene reconstruction (Hartley and Zisserman, 2000). Keypoint detectors can be regarded either as: i) a fixed scale detector, where the user has to manually define a local neighbourhood around a candidate point to perform the required checks for keypoint detection, or ii) a scale-invariant detector, where the local scale (i.e., local neighbourhood of interest) around a keypoint is automatically defined by the algorithm. The concept of scale invariance is that the attribute or features of an object should not change when the object is scaled by a multiplicative factor. The definition of a similar local scale for a corresponding source and target keypoint is important since it ensures that they both have the same local neighbourhood regions, which can then be used for computing comparable keypoint descriptors (or attributes). Scale-invariant detectors are typically used for this purpose.

Automated scale selection mechanisms have been popularly applied for 2D keypoint detectors. Examples include the Scale Invariant Feature Transform (SIFT) detector (Lowe, 2004), which uses a ‘Difference-of-Gaussian’ (DoG) framework for estimating the local scale, whereas another detector, i.e., the Harris-Laplacian interest point operator (Mikolajczyk and Schmid, 2004) uses Lindeberg’s automatic scale selection approach (Lindeberg, 1998). The DoG approach smoothes the data with Gaussian kernels of differing standard deviations and then takes the difference of smoothed outputs to build a scale-space representation. The details of Lindeberg’s approach will be discussed in Chapter 3.

With the increasing use of point clouds for 3D object recognition and matching (Lai and Fox, 2010; Tam et al., 2013), there are numerous 3D keypoint detectors including the

intrinsic shape signature (ISS) (Zhong, 2009), the mesh-Difference of Gaussians (mesh-DoG) (Zaharescu et al., 2009), Heat Kernel Signature (HKS) (Sun et al., 2009) and Harris 3D (Sipiran and Bustos 2011). ISS is a fixed scale detector. ISS uses the ratios of the eigenvalues of the local neighbourhood to determine surface variation. Points with large surface variations are marked as keypoints. The mesh-DoG, HKS and Harris 3D detectors operate on mesh representations of the point clouds. The mesh-DoG is a scale-invariant detector which uses a DoG-based scale-space representation. For mesh-DoG, the ratios of eigenvalues from the Hessian matrix of the local mesh neighbourhood are used for keypoint definition. The HKS is related to the surface curvature of a point and is based on the diffusion of heat on a surface mesh using the Laplace-Beltrami operator. This operator is extensively used in 3D shape analysis to describe physical processes such as heat diffusion and wave propagation (Wetzler et al., 2013). Keypoints are determined by searching for local maxima HKSs across the surface mesh (Teran and Mordohai). HKS is not a scale-invariant detector, however, Bronstein and Kokkinos (2010) have presented an approach to address this problem. Harris 3D is a fixed scale detector. It fits a local surface quadratic patch to the point data and computes the so-called ‘Harris-response’ (Harris and Stephens, 1988) for each mesh vertex. Query vertices with large responses are classified as keypoints.

ISS, mesh-DoG, HKS and Harris 3D are examples of detectors which utilize surface geometry for the extraction of 3D keypoints. There are also volume-based methods which utilize 3D voxel representations instead of direct point cloud data for keypoint detection (Yu et al., 2013). These include a 3D extension of the SIFT method (Rusu and Cousins,

2011; Hänsch et al., 2014). 3D-SIFT is scale-invariant and utilizes a ‘Difference-of-Gaussian’ scale-space approach, where a series of downsampling/smoothing is applied to the point data to determine keypoints and their local scale. 3D-SIFT encompasses both keypoint detection, as well as keypoint description (Section 2.1.2). Volume-based approaches operate on voxel representations of the 3D model, whereas surface geometry-based methods use geometric attributes from surface patches, normals or contours of the 3D point clouds.

2.1.2 Matching of 3D keypoints using descriptors

Following the extraction phase, attributes (or descriptors) must be assigned to the keypoints. Then a search strategy is employed to find keypoint descriptors with high similarities. The generation of uniquely discriminable descriptors is an important step since it influences the keypoint matching success rate. Descriptors can be represented in various forms including: 1D vectors, 2D / 3D histograms or multi-dimensional arrays.

Volume-based 3D keypoint descriptors such as the 3D-SIFT implementations have been used for video sequences and 3D medical images (e.g., MRI and CT scans) (Scovanner et al., 2007; Flitton et al., 2010). In these cases, the 3D data is first converted into a 3D array of voxels containing data points and the descriptor is generated based on the gradient magnitude and orientation of these voxels.

There has also been various surface geometry 3D point cloud descriptors developed over the years. Some of these include the Spin Images (Johnson and Hebert, 1999), Fast Point Feature Histograms (FPFH) (Rusu et al., 2009) and Signature of Histograms of

Orientations (SHOT) (Tombari et al., 2010). The HKS described in the previous section can also be used as a descriptor for surface keypoints (Section 2.1.1).

For Spin Images (Figure 2.2), every point within the local keypoint neighbourhood are assigned two coordinates, α and β ; α is the distance from the keypoint to the projection of the neighbourhood point on the local surface tangent plane (i.e., the plane tangent to the normal vector of the keypoint). β is the distance from the neighbourhood point to the local tangent plane. For every point in the local neighbourhood, these pair of coordinates is accumulated into a 2D array, thus forming the descriptor.

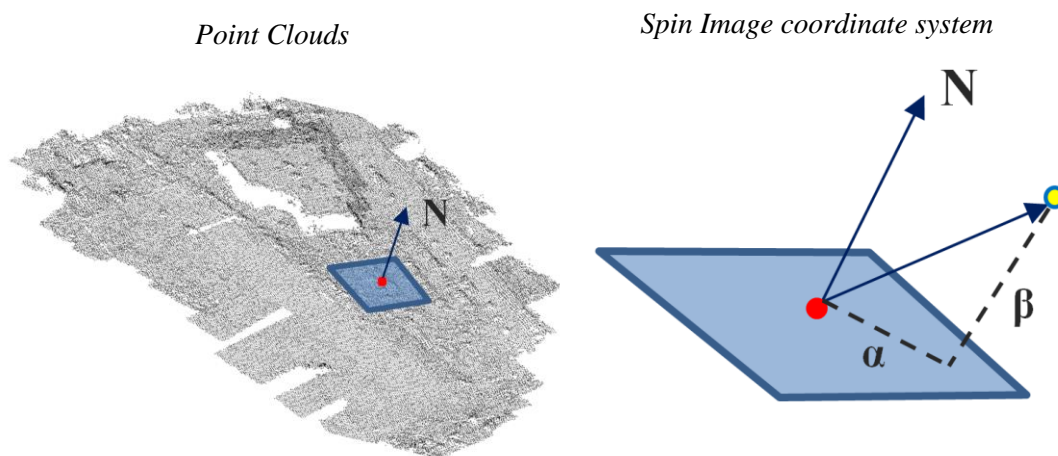


Figure 2.2: Concept of Spin Image point cloud descriptor formation. Left: Keypoint (red point) with its normal vector N and tangent plane to this vector (blue region). Right: Coordinate system of spin image where the coordinate pair (α, β) is defined by the vector projecting from the keypoint (red point) to a neighbouring point cloud (yellow point). (Modified after: Ruiz-Correa et al., 2004)

FPFH is a histogram-based descriptor which bins three angular attributes defined by the relation between every neighbourhood point and the keypoint (Figure 2.3). SHOT is also a histogram-based descriptor which defines a spherical neighbourhood around the

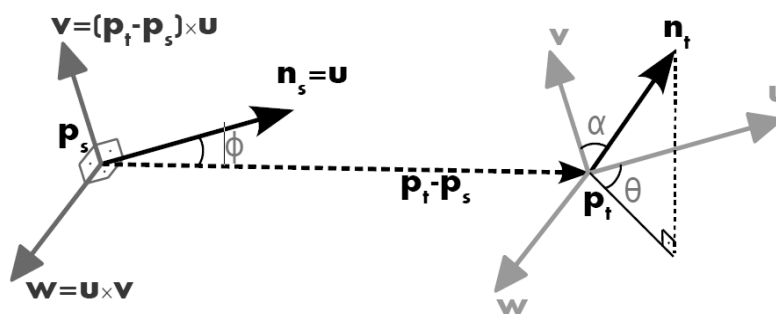


Figure 2.3: Concept of FPFH formation showing the triplet angular relation (α, θ, ϕ) between \mathbf{p}_s (the keypoint) and \mathbf{p}_t (neighbouring point). The \mathbf{u}, \mathbf{v} and \mathbf{w} vectors defines a local coordinate frame of the point cloud and is computed using the normal vector of the keypoint. (From: Rusu, 2009).

keypoint (Figure 2.4). This spherical neighbourhood is then partitioned into spherical grid sectors. For each grid sector, the angles between the normals at the neighbouring points and the normal at the keypoint are accumulated into a local histogram. The local histograms of all grid sectors are then concatenated to form the SHOT descriptor.

Geometry-based descriptors such as Spin Images, FPFH and SHOT require a local point cloud neighbourhood to be defined around the keypoint. A user-specified distance

can be applied to define local neighbourhoods when the source and target point clouds have the same scale. However, in situations where there is a scale difference between the source and target datasets, the descriptors are not scale-invariant and will fail during the feature matching process. As discussed earlier, scale-invariance is typically provided by local keypoint scales estimated from a front-end detector. Mellado et al. (2016) developed an approach for scale-invariant co-registration of multi-sensor point clouds based on a descriptor known as ‘Growing Least Squares’ (GLS). The GLS descriptor is built in a logarithmic scale space by fitting algebraic spheres on the point cloud data.

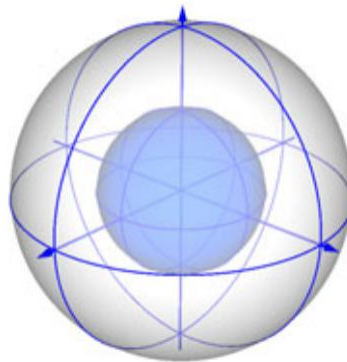


Figure 2.4: Illustration of the SHOT descriptor. It is based on the partitioning of sectors within a spherical grid structure around the keypoint. (From: Tombari et al., 2010).

Descriptor-based matching usually comprises of three main components: i) the design of a cost (or similarity) function to assess the similarities of source and target keypoint descriptors, ii) a searching mechanism to efficiently compare the descriptors in their feature space (i.e., 1D, 2D, 3D or multi-dimensional feature space) for establishing one-

to-one keypoint correspondences and iii) an approach to filter false (or outlier) keypoint matches.

Weber et al. (2015) developed a descriptor-based point-matching framework to automatically align surface point clouds collected from a Microsoft Kinect sensor. The method fuses multiple Kinect-based point clouds of an object or scene. Their approach uses the FPFH point cloud descriptor and does not extract points of interest or keypoints. Instead, the descriptors are computed for every point cloud in the dataset. The local point cloud neighbourhood used to compute the surface descriptors are defined by a user-specified radius value. This indicates that the approach is not scale-invariant as the same radius value is applied on both source and target point clouds. Thus, the approach is unable to handle cases where the point clouds to be co-registered differ by a global scale factor. The descriptor-based point matching is determined using the nearest neighbour distance ratio (NNDR) (Szeliski, 2010) followed by the RANdom SAMple Consensus (RANSAC) (Fischler and Bolles, 1981) for removal of correspondence outliers. The combination of NNDR and RANSAC is a popular strategy for matching keypoints using descriptors.

NNDR is based on searching for the target keypoint that is the ‘nearest neighbour’ for a source keypoint within the descriptor feature space. The nearest neighbour is the target keypoint with the minimum Euclidean distance to the source keypoint in the feature space. In this case, the Euclidean distance metric is the similarity (or cost) function. Efficient nearest neighbour searching is typically achieved using k - d trees (Bentley, 1975)

It is possible that 1st and 2nd nearest neighbour target matches have similar distances to the source keypoint. NNDR compares the ratio of these two distances. A distance ratio that tends to 1 indicates matching ambiguity and thus the source keypoint should not be included in the final set of correspondences. RANSAC is used to further filter wrong matches. It is based on randomly sampling the minimum number of correspondences required to compute the transformation parameters. Then all the source keypoints are back-projected onto the target domain using these parameters. Matches are then established by searching for source to target keypoints which are in close proximity to each other based on a threshold. The number of correspondences are then recorded. RANSAC is iterative and the final set of matches is the sample set which gives the highest amount of correspondences. The disadvantage of utilizing both the NNDR and RANSAC is that user-defined thresholds are required to filter potentially incorrect point matches. If there is no information about the coordinate systems of the source and target point clouds prior to matching, it becomes difficult to determine optimal threshold values without some empirical analysis.

Zeng et al. (2016) proposed a 3D local volumetric patch descriptor algorithm referred to as '3DMatch'. The approach is based on deep learning which requires training the descriptors on large volumes of data. This can be time-consuming and also depends on the availability of training data.

2.2 3D Non-descriptor-based methods

There are also descriptor-free approaches which address the initial 3D point cloud alignment problem based on the data and data-derived geometric primitives. A common approach for global co-registration is the utilization of PCA or SVD (Singular Value Decomposition). PCA (or SVD) is used to approximate the rotation required to align the coordinate systems of the source and target point clouds. The translation can be estimated by the difference in centroids of the source and target data. However, when there is partial overlap and/or shape deformation between the source and target surfaces this approach does not provide the correct transformation parameters (Salvi et al., 2007; Castellani and Bartoli, 2012).

Other non-descriptor based methods utilize various geometric constraints and relationships amongst points, lines or planes. In terms of the plane-based methods, von Hansen (2006) presented a framework for terrestrial laser scanning (TLS) co-registration. Firstly, planes are extracted from point cloud data and this is followed by an exhaustive search for corresponding planes. The method does not cater for scale differences between the point clouds. Brenner et al. (2008) derived two methods for the initial co-registration of TLS data: a plane-based scoring approach and another which uses the normal distributions transform (NDT) (Biber, 2003). In the first method, plane triplet correspondences are scored using the similarity of their normal vector directions, in combination with distances to the plane origin. The second method sliced the 3D scans into 2D layers, and then used the 2D NDT algorithm for co-registration. NDT is an optimization-based co-registration algorithm which tries to maximize a probabilistic

matching score between two 2D scans. Only the 3D roto-translational parameters were accounted for in their work.

Stamos and Leordeanu (2004) used both linear and planar features to align laser scans of buildings. Properties such as length of the lines, in addition to plane sizes were used to discard possible erroneous matches, thus reducing the combinatorial correspondence search space. This was accomplished using a variety of heuristically set thresholds. Their method solved for the six rigid parameters. Yang et al. (2016) used semantic features from urban scenes for automated TLS co-registration. The point cloud data was segmented into ground and non-ground followed by the extraction of vertical linear features. The vertical features were then triangulated to form a network. Then a hashing table with triangular constraints were used to find matching source and target triangles. The method used various Euclidean distance-based constraints and thresholds which can only be applied when source and target point clouds are of the same scale. A geometric object approach was proposed by Chan et al. (2016) where a single, octagonal lamp pole was used for the alignment of terrestrial laser scans in different coordinate systems (i.e., different 3D position and orientation). The premise of the approach is to fit an octagonal pyramid model to the raw point clouds. Then, virtual points from the lamp pole structure are computed and used within an iterative matching strategy to estimate the registration parameters.

Linear features extracted from point clouds have been used to match Airborne Laser Scanning (ALS) and TLS data (von Hansen et al., 2008). This method sequentially computed the 3D rotation and translation parameters. Rotation was derived via the

correlation of line orientation histograms. Afterwards, translation was determined using a ‘generate and test’ scheme, where the quality of all line correspondence combinations are assessed using the proximity of matching between ALS and TLS line midpoints. Yang et al. (2015) presented an approach for ALS to TLS alignment in urban scenes. They employed a spectral graph correspondence approach for matching building outlines. The graph matching utilized scale-variant geometric constraints such as distances together with several other spatial relations derived from the TLS and ALS building outlines. Urban areas typically contain many other rich descriptive details such as road networks, street furniture and vegetation. Therefore, the method may falter in urban datasets where there is a lack of building structures.

Aiger (2008) developed the ‘4-Point Congruent Set’ (4PCS) method for initial rigid alignment of point clouds. The approach begins by sampling four-point coplanar tuples from the source point cloud, followed by a search based on an affine ratio to find corresponding four-point tuples in the target point cloud. The best transformation is then selected from multiple candidate transformations formed by the set of matching quadruples. There have been several extensions/variations of 4PCS. Theiler et al. (2014) combined 3D keypoints with the 4PCS for the alignment of terrestrial laser scans. In other work, Mellado et al. (2014) developed a speeded up version of 4PCS. In context of full initial registration (i.e., solving for scale and rigid parameters), Corsini et al. (2013) presented an extension of 4PCS which can handle scale changes between datasets.

Yang et al. (2013) developed an ICP method referred to as ‘Globally Optimal ICP’. This ICP approach does not require any initial alignment and is based on a branch and

bound optimization search for optimal registration parameters. However, as mentioned in Theiler et al. (2014), the globally-optimized ICP is not efficient when applied to large scale data such as laser scans.

In comparison to non-descriptor based methods, descriptor-dependent approaches take into account local data information, i.e., it considers neighbouring elements for attribute definition. Descriptor-based methods provide semantic context, thus strengthening the matching process with richer information about the local keyfeatures (for example, enabling the elimination of false matches by comparing descriptor similarity).

2.3 2D image-based methods

Another active branch of research which addresses initial point cloud alignment are image-based approaches. The concept revolves around the utilization of image-based representations of the point cloud data collected from various sensor acquisition systems. One type of image representation can be obtained from optical cameras which are mounted to and synchronised with the laser scanners during point cloud data collection. If the transformation between the camera coordinate system and the laser scanning system is established prior to data collection, then the relative orientation of an image pair can be used to derive the transformation parameters between the associated source and target laser scans. Image representations can also be 2D reflectance intensity images formed from the energy of the backscattered laser light on a laser scanning system. Another type of image representation are 2D height maps or range images. The pixels of height maps store the 3D coordinates of a point cloud. Usually each height map image pixel is a

visualization of the point cloud surface elevation. However, depending on the application, the other axes directions of the point cloud's 3D coordinate system can also be used to project the point clouds into the 2D height map/range image domain.

Manasir and Fraser (2006) used the relative orientation of optical image pairs to co-register multiple TLS datasets. However, a significant amount of work instead focuses on TLS point cloud co-registration using reflectance intensity images (Böhm and Becker, 2007; Wang and Brenner, 2008; Kang et al., 2009; Weinmann et al., 2011). These works all follow a similar alignment framework based on 2D keypoint matching between reflectance image pairs.

Various interest point operators can be used for extracting 2D keypoints including the Förstner operator (Förstner and Gülch, 1987), and Moravec and Harris corner detectors (Moravec, 1980; Harris and Stephens 1988). 2D descriptors such as SIFT (Lowe, 2004) and Speeded Up Robust Features (SURF) (Bay et al., 2008) can then be used for matching the 2D keypoints. The SURF descriptor is based on the computation of Haar wavelet filter statistics in both the horizontal and vertical image directions.

The matched 2D feature points from the intensity images also have accompanying 3D point cloud coordinates, therefore 3D transformations can be directly computed for 3D point cloud co-registration. A common trend in these works is the usage of Lowe's SIFT keypoint detector and descriptor algorithm coupled with RANSAC. This due to the scale and rotation invariance properties of SIFT. SIFT has also been applied to match 2D keypoints on range images for the purposes of 3D point cloud alignment (Barnea and Filin, 2007; Sehgal et al., 2010).

Barnea and Filin (2008) proposed a combinatorial keypoint approach for terrestrial point cloud matching using panoramic range images. Range image keypoints are extracted using a so-called ‘min-max’ interest point detector. These keypoints are then subjected to RANSAC. Firstly, a triplet of keypoints is randomly selected. Then differences in 3D Euclidean distances between source and target keypoint pairs from the sample set are used as a check for the verification step within RANSAC. For multi-sensor point clouds which may have scale differences, this verification check will not suffice. Additionally, depending on the amount of keypoints extracted from scene to scene, the correspondence search space can significantly increase and be time consuming.

Novák and Schindler (2013) used height maps for the co-registration of 3D laser scanning and photogrammetric point clouds. Point clouds are firstly converted to height maps by projecting onto a planimetric x, y -plane. Then gradient information and RANSAC are used to match points on source and target height maps. Afterwards, ICP is applied to refine the 3D point cloud registration accuracy.

2.4 Summary

From the reviewed literature, a considerable amount of 3D approaches (both descriptor and non-descriptor based methods) are not scale-invariant and only consider the estimation of 3D rigid transformation parameters. This work addresses the alignment of 3D point clouds which differ in terms of a 3D conformal transformation. In Chapter 3, a modified approach of Lindeberg’s local scale selection mechanism (Lindeberg, 1998) for scale invariant extraction of 3D surface keypoints is proposed.

In terms of 3D descriptors, most current methods utilize geometric relations (e.g., angles) between the 3D points. The proposed 3D descriptor uses surface morphology characteristics for the local neighbouring region around the keypoints.

From the reviewed works on 2D image-based methods for 3D point cloud co-registration, many of the current approaches utilize intensity-based methods for extracting and matching 2D keypoints. The majority of them uses a rectangular grid system for descriptor definition. In Chapter 4, this research studies the wavelet scale-space structure of the height map images for keypoint extraction. In addition, a biological vision-inspired gridding system for space-variant sampling is utilized for generating the 2D descriptors.

3. A 3D-based Approach for Point Cloud Alignment

In this chapter, a 3D keypoint-based feature-matching framework is proposed for co-registering multi-temporal, multi-sensor, natural and anthropogenic (man-made) 3D point clouds. There are four main components: i) the development of a scale-invariant 3D keypoint feature extraction method using morphological properties, specifically the local surface curvature, ii) the development of a rotation, translation and scale invariant 3D keypoint surface descriptor based on surface topography, iii) the application of a bipartite graph descriptor matching method for establishing initial keypoint feature correspondences without the need for user-specified thresholds, and iv) the development of a threshold-free, RANSAC-like outlier detection algorithm to eliminate wrong keypoint correspondences.

Once the final set of keypoint correspondences are found, a 3D conformal transformation is applied to recover the seven parameters (i.e., a global scale factor, three rotation angles and three translations), which will enable source to target point cloud co-registration.

3.1 3D-based Point Cloud Alignment

Methodology

This work follows a surface geometry-based approach, using: i) 3D points for estimating surface curvature in the keypoint extraction process and ii) point surface patches for capturing local 3D surface topography details which are utilized in the descriptor generation process. The proposed approach uses an automated feature-matching pipeline which includes feature extraction, feature description and feature correspondence.

The presented method is based on extracting and matching distinct 3D point landmarks referred to as *keypoints* on the source and target point clouds. A couple of the main challenges lie in the establishment of: i) keypoints which are scale-invariant (i.e., point features which can be used for matching source and target datasets which differ by a global scale factor), and ii) keypoint descriptors, which must be invariant to scale, rotations and translations as a result of the 3D conformal displacement between source and target datasets.

Figure 3.1 is an illustration of the keypoint matching concept, where two point cloud datasets are given and differ by a rotation matrix R , translation T and scale factor s . Distinct keypoints (small blue circles) are extracted on both point cloud datasets. A scale-invariant neighbourhood (large circles) is determined for each keypoint. This neighbourhood is used to compute descriptors D (or attributes) for the keypoints. Source and target descriptors (D_{SOURCE} and D_{TARGET}) are matched using a similarity metric $SimCost$ to find corresponding keypoints. The descriptors, D_{SOURCE} and D_{TARGET} are shown as 1D

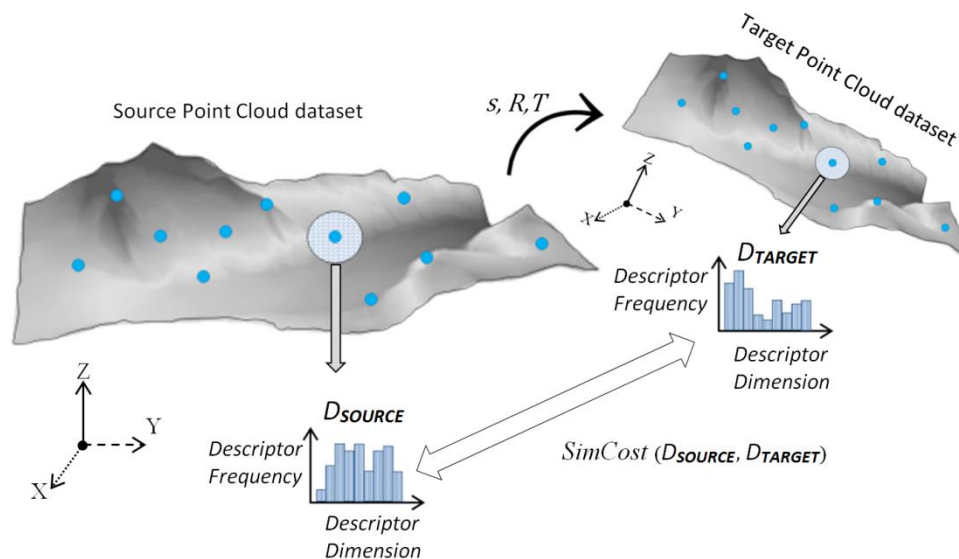


Figure 3.1: Concept of keypoint matching between source and target point clouds.

histograms (y-axis show the descriptor frequency and the x-axis show the descriptor dimensionality which is the descriptor size) for visual purposes. However, descriptors can also be represented in higher orders (e.g., 2D or 3D histograms).

For photogrammetric and mobile mapping applications, there is no guarantee of source and target datasets being in the same coordinate system and in close proximity to each other. For example, un-georeferenced source point clouds versus geo-referenced target point clouds. Common examples of such instances are when Global Positioning System (GPS) signals are lost during a mobile laser scanning operation or the generation of photogrammetric point clouds from platforms such as UAVs using structure-from-motion. In both cases, the resulting point data are in local coordinate systems. Hence, this work focuses on developing a co-registration framework which estimates the seven-parameter 3D conformal transformation between source and target point clouds without

any proximate matching assumptions. The proposed 3D keypoint extraction and descriptor methods utilize various terrain characteristics such as curvature, slope and surface distances, specifically for the co-registration of urban and natural point cloud datasets. Figure 3.2 illustrates the proposed registration framework. In the following sections, each component of the framework is presented.

3.2 Extraction of 3D Surface Keypoints

In this section, the aim is to establish discrete, 3D, stable and repeatable keypoints on the point cloud surface. Repeatable keypoints are those points that can be detected at the same location on both the source and target data, even in the presence of scale changes and rigid motion. To achieve this aim, a 3D detector has been developed which utilizes surface morphology, namely, the curvature of the point cloud surface, to find points of significance. The input datasets used in this 3D-based co-registration pipeline comprise of 3D point clouds with (x, y, z) coordinates. To classify a 3D surface query point cloud $P_{surface}$ as a possible keypoint, the curvature at $P_{surface}$ is computed. Given that $P_{surface}$ is centered on a circular neighbourhood of surface point clouds \mathcal{N} , the local surface curvature is estimated by utilizing the local covariance matrix $Cov_{surface}^{\mathcal{N}}$. \mathcal{N} comprises of the \mathbb{K} -nearest point neighbours around $P_{surface}$ (Equation 3.1).

$$\mathcal{N} = \begin{bmatrix} N_1 \\ N_2 \\ \cdot \\ \cdot \\ N_{\mathbb{K}} \end{bmatrix} \quad (3.1)$$

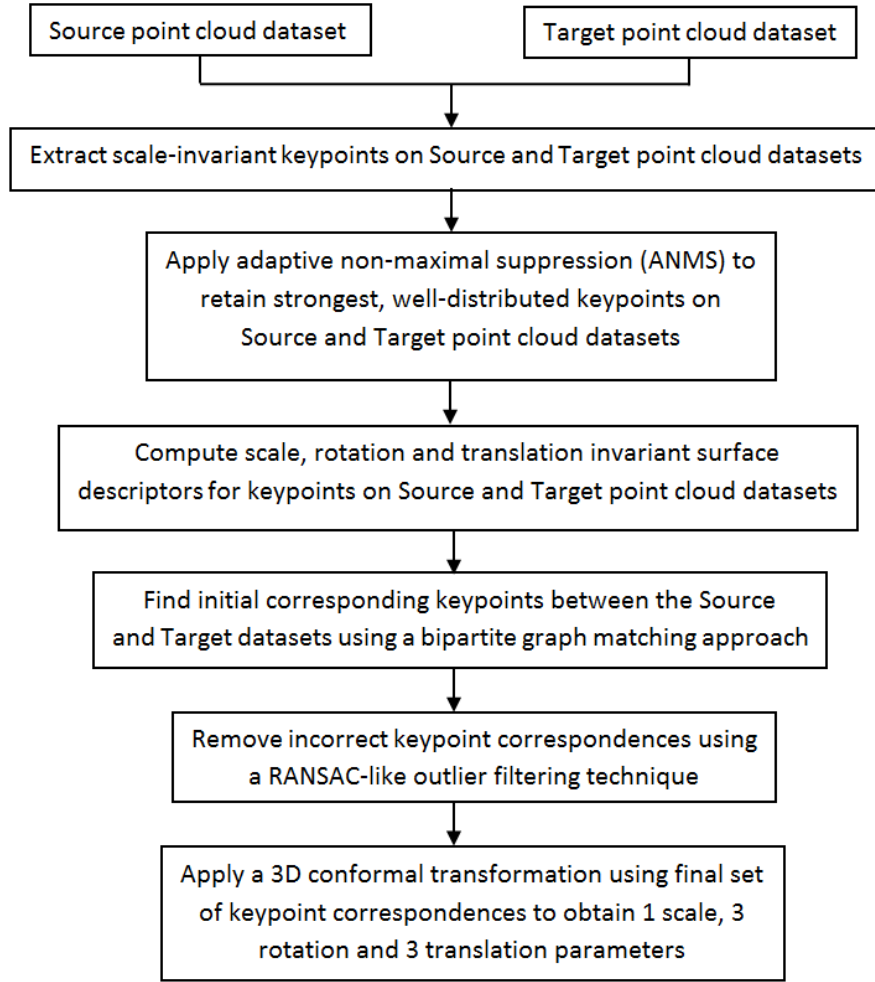


Figure 3.2: Workflow of the proposed 3D-based point cloud co-registration approach.

$P_{surface}$ is the focal point of \mathcal{N} , from which $Cov_{surface}^{\mathcal{N}}$ is estimated using Equation 3.2.

$$Cov_{surface}^{\mathcal{N}} = \frac{1}{\mathbb{K} - 1} \sum_{j=1}^{\mathbb{K}} (N_j - P_{surface}) \cdot (N_j - P_{surface})^T \quad (3.2)$$

Eigen-decomposition of $Cov_{surface}^{\mathcal{N}}$ provides three eigenvalues λ_u and corresponding eigenvectors V_u given in Equation 3.3. V_u represents the three axes of the local, 3D orthogonal coordinate frame \mathcal{F} for $P_{surface}$. λ_u represents the magnitude of the three \mathcal{F} axes and $P_{surface}$ is \mathcal{F} 's origin. The magnitude of λ_u is as a result of the dispersion of \mathcal{N} in each of \mathcal{F} 's 3 orthogonal axis directions.

$$Cov_{surface}^{\mathcal{N}} \cdot V_u = \lambda_u \cdot V_u \quad (3.3)$$

where, $u = (1,2,3)$ is in ascending order of λ 's magnitude

λ_1 , is the minimum eigenvalue of $Cov_{surface}^{\mathcal{N}}$ whose eigenvector is the surface normal for the local region around $P_{surface}$. The surface normal is the orthogonal axis direction, which has the smallest variation relative to the local tangent plane on the point cloud surface. λ_2 and λ_3 indicate the deviation of \mathcal{N} 's points in the other two axes directions on the local neighbourhood's tangent plane. Utilizing the eigenvalues, the surface curvature at $P_{surface}$ is established as the ratio of the surface normal variation λ_1 to the total variance $\sum_{u=1}^3 \lambda_u$ (Equation 3.4; Pauly et al., 2003; Bae and Lichti, 2008).

$$Surface\ Curvature\ (P_{surface}) = \frac{\lambda_1}{\sum_{u=1}^3 \lambda_u} \quad (3.4)$$

A keypoint is a surface point that can be distinguished from its local neighbours \mathcal{N} . The surface curvature is used for this purpose; however, an approach to determine the

boundary extent of \mathcal{N} has yet to be established. These neighbouring points are critical since they also serve as the local region for computing the subsequent keypoint descriptors. A radius can be user-specified to define a circular local region around the keypoint candidate and establish this local point neighbourhood. However, if the source and target point clouds differ by an unknown global scale factor, a manually applied radius value is not feasible for obtaining similar local regions (hence, similar descriptors), which is an important criterion for finding corresponding source and target keypoints. Therefore, a scale-invariant keypoint extraction process is applied, based on ranges of radii, to automatically delineate homogeneous source and target neighbourhood regions under any apparent scale change between the two point cloud datasets to be co-registered (Figure 3.3). Every 3D point belonging to the input source and target data is examined as a possible keypoint location. For each point, multiple curvature values are computed by gradually increasing the size of point neighbourhoods based on a range of radii, thus using a 'scale-space' representation.

3.2.1 Scale invariance for 3D keypoints

SIFT (Lowe, 2004), and its 3D-based extensions use a 'Difference-of-Gaussian' approach for obtaining scale-invariant keypoints. The Harris-Laplacian interest point operator has been shown to have higher discriminative capabilities than Difference-of-Gaussian-based detectors (Mikolajczyk and Schmid, 2004; Grauman and Leibe, 2011). Harris-Laplacian uses Lindeberg's method (Lindeberg, 1998) for automatic keypoint scale selection. Lindeberg's method is based on selecting the optimal scale value (and hence the optimal

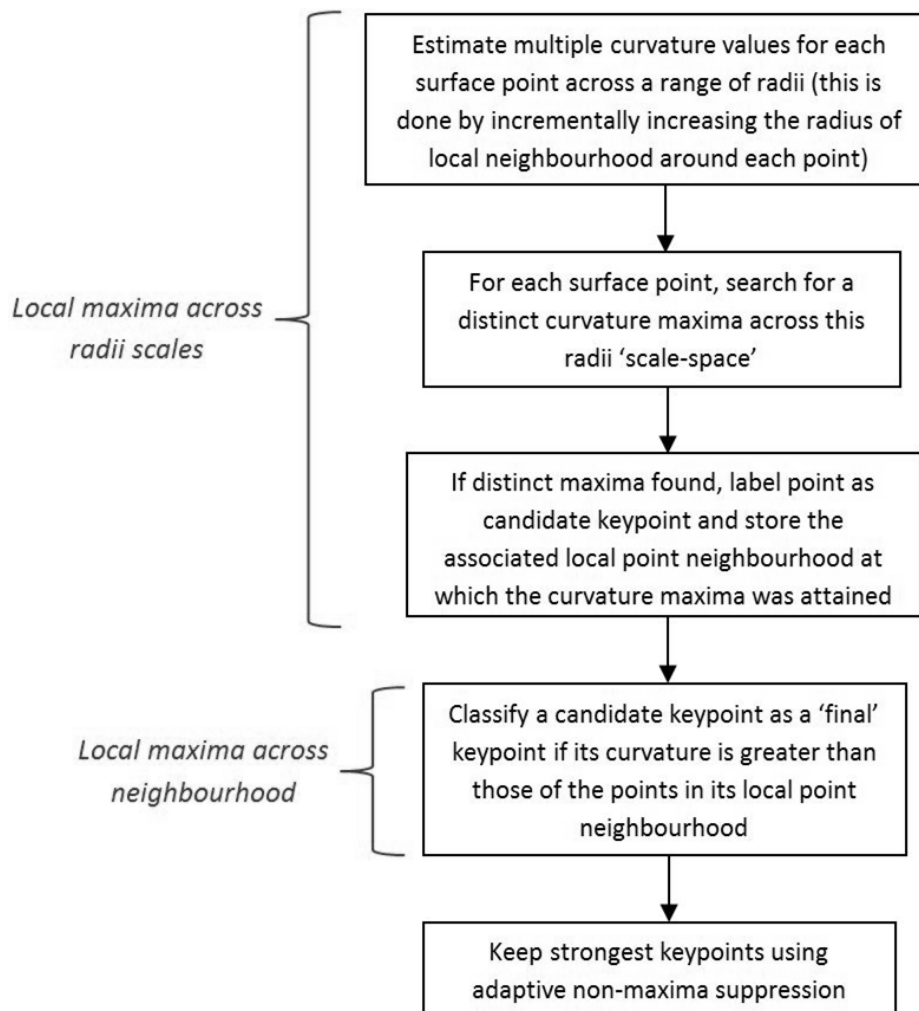


Figure 3.3: Workflow for the proposed 3D keypoint extraction process.

local neighbourhood of a keypoint candidate) as the strongest local maxima of a scale parameter-dependent function (SPDF) across a range of scales. The proposed 3D scale-invariant keypoint extraction approach follows a similar framework to Harris-Laplacian by combining 3D curvature information with Lindeberg's automated scale estimation method. For the presented approach in this dissertation, the scale parameter is the radius, which defines the concentric local neighbourhoods around possible keypoint candidates.

The 3D point clouds within these neighbourhoods are then used to compute the surface curvature of the local region around the keypoint candidate. Since the curvature of a local region varies depending on the radii used, in this context, the curvature measure serves as the SPDF.

The respective SPDF signals for a correspondence pair (i.e., matching source and target keypoints) would have comparable shapes since the two keypoints are focal points defining similar local point cloud regions. However, depending on the magnitude of the global scale factor ‘ s ’, the shapes of source and target SPDF signals can be compressed or stretched versions of each other (Figure 3.4). The SPDF signal may have several local maximas. For a correspondence pair, the distinct local maxima (i.e., global maxima) of the curvature responses on both the source and target SPDF signals would give us the two radii, r_{source} and r_{target} for obtaining the same local point neighbourhoods, regardless of the scale difference between the source and target point clouds. The ratio of r_{source} and r_{target} is equivalent to s .

Scale-invariant candidate keypoints are established when the 3D surface query point cloud $P_{surface}$ exhibits a distinct local maxima across the range of curvature scales. SPDF signals are shown in the plots of Figure 3.4 for a source and target point cloud pair, which differ by a global scale factor. The signals exhibit similar shapes since the same keypoint exists on both point cloud datasets. The distinct SPDF maximas also establish the same local regions around the keypoints on both the source and target point clouds. Therefore these keypoint candidates would now also have an associated local neighbourhood, thus ensuring their keypoint descriptors are also scale-invariant.

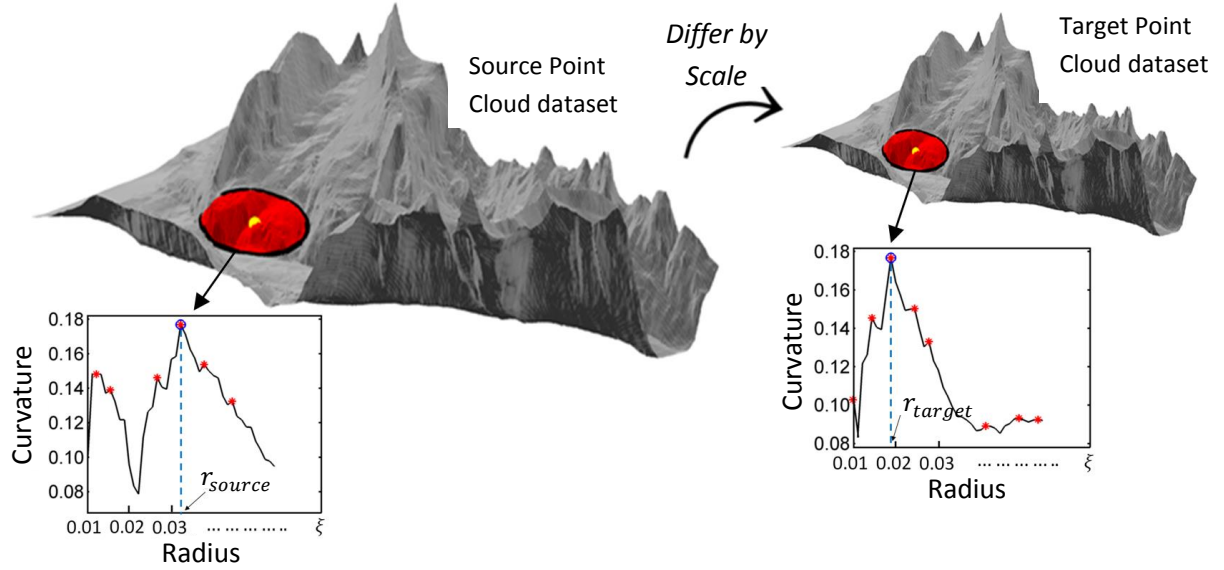


Figure 3.4: Concept of obtaining scale-invariant keypoints.

Both the input source and target point cloud coordinates are normalized between [0~1]. The step of normalized radius values δ_r used for generating the neighbourhoods is set at 0.001 intervals, where $\delta_r \in [0.010, 0.011, 0.012, \dots, \xi]$. δ_r is stopped at a maximum outer radius bound ξ . The value of ξ has been empirically set as 10% of the maximum extent on the point cloud surface. These are the default parameters set in all experiments, however, the δ_r range and its intervals can optionally be changed since they depend on the point spacing resolution of the point clouds datasets to be co-registered.

Until now, the initial set of point cloud keypoints are those that have a ‘distinct local maxima across scales’. In the next step, a search performed to identify the ‘distinct local maxima across the local neighbourhood’, i.e., the surface curvature of the candidate keypoint KP_{cand} is compared with the surface curvature of the points inside its local

neighbourhood. Thus, a point cloud keypoint, KP , is retained if this local-maxima criteria is met (Equation 3.5).

$$\text{Retained } KP; \text{ if } \textit{Surface Curvature}(KP_{\text{cand}}) > \textit{Surface Curvature}(\mathcal{N}) \quad (3.5)$$

3.2.2 Keypoint refinement by adaptive non-maxima suppression

Thus far, the keypoints have been determined using a dominant local maxima approach i.e., comparing the surface curvature strength of a candidate keypoint with respect to the surface curvature strength of local neighbourhood points. However, these initial detections can suffer from poor spatial localization, i.e., there may be multiple keypoint detections which are close to each other. These unwanted additional points increase the computational time for the feature-matching phase and may also cause matching ambiguity due to the closeness of multiple keypoints. To remove spurious keypoints and retain the most distinctive and strongest ones on the point cloud surface, global non-maxima suppression is applied.

The n -th strongest keypoints can simply be chosen based on those with the greatest surface curvature strength. However, this does not guarantee uniform distribution of keypoints across the point cloud surface since the strongest features may be clustered together in certain regions. Therefore, a keypoint suppression approach has been implemented. The approach is similar to the adaptive non-maximal suppression (ANMS) technique originally proposed by Brown et al. (2005). ANMS compromises between the

elimination of relatively weak keypoints and at the same time ensuring a regular distribution of distinct keypoints across the point cloud surface. In contrast to the 2D corner strength function for image keypoints utilized by (Brown et al., 2005), the surface curvature is used as the ‘strength indicator’ for interest points on the point cloud surface.

The suppression process begins by letting KP_{num} ($num = 1, 2, \dots$, number of initial *keypoints*), be the initial set of detected keypoints. For each *keypoint* $\in KP_{num}$, a search is performed to find its closest neighbouring keypoint, KP_c , which is of greater curvature strength. The distances between *keypoints* $\in KP_{num}$ and their respective KP_c are stored and sorted from the largest to smallest. Keypoints found to have a large distance from their nearest, ‘stronger’ neighbour are then retained. A large distance represents a discrete *keypoint* $\in KP_{num}$ that is not suppressed since its KP_c is spatially far away. This criterion encourages a final set of keypoints, which are well-distributed on the point cloud surface. Therefore, the accepted keypoints are those with the largest \mathcal{M} distances. The remaining keypoints are eliminated from KP_{num} , where \mathcal{M} is the maximum number of final keypoints which the user wishes to keep after suppression. The \mathcal{M} parameter is dataset specific, depending on the size and coverage of point cloud datasets used. For the point cloud datasets used in this dissertation, the parameter \mathcal{M} is set as 60% of the total number of initially detected keypoints. Figure 3.5 illustrates sample results of keypoint extraction on a glacial icefield (non-urban) point cloud dataset before and after ANMS is applied.

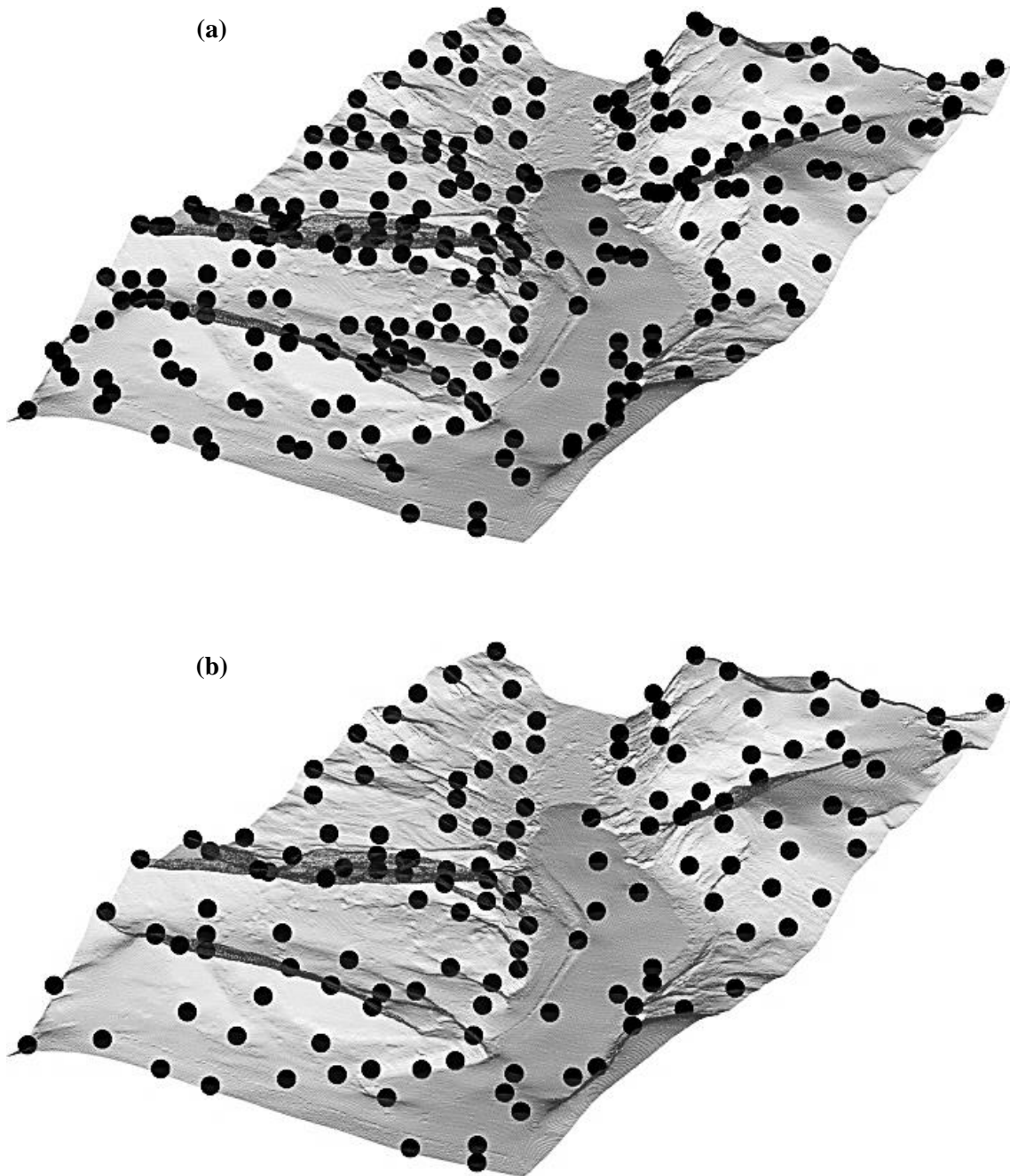


Figure 3.5: Example of keypoint extraction on point cloud surfaces. a) Before ANMS b) After ANMS.

3.3 3D Surface Descriptors for Keypoints

In the previous section, a neighbourhood around each 3D keypoint was defined, based on a scale-space approach. For every source keypoint, its best keypoint match on the target point cloud surface must be obtained. Hence, neighbourhood attributes are used to define descriptors for the source and target keypoints. A corresponding source and target keypoint would have closely matching descriptors. In this section, a descriptor is developed that captures the local surface properties of the neighbourhood around the keypoint to ensure uniqueness during the point-to-point matching phase.

3.3.1 Rigid invariance for local 3D descriptors

The descriptors are computed using the local scale-invariant neighbourhoods established in Section 3.2.1. This means that the descriptors themselves are inherently scale-invariant. However, descriptor invariance with respect to the 3D rigid parameters (i.e., 3D rotation and 3D translation) is also required. To achieve this, the local orthogonal coordinate frame \mathcal{F} (defined in Section 3.2) of each KP is utilized. The eigenvector with the maximum eigenvalue is the x -axis of \mathcal{F} and the eigenvector with the minimum eigenvalue is the z -axis (in the direction of the local surface normal). The y -axis is the remaining eigenvector. However, the directions of the eigenvectors are not always repeatable if the surface point clouds undergo a rotation (Tombari et al, 2010).

To overcome this ambiguous effect and ensure consistency in axes directions, the directions between $vecs(KP, \mathcal{N})$ and $vec(Ortho-Axis)$ are compared; where

i) $vecs(KP, \mathcal{N})$ are all the vectors formed from the KP to its neighbouring points belonging to \mathcal{N} and, ii) $vec(Ortho-Axis)$ is the vector for one of the 3 major axes. This is done by utilizing the sign of the dot product between $vecs(KP, \mathcal{N})$ and $vec(Ortho-Axis)$. For instance, if the dot product between $vec(Ortho-Axis)$ and a vector, $vec(KP, \mathcal{N})$ is negative, then they have opposite directions. Likewise, if their dot product is positive, then they share similar directions. The idea behind the choice of axes directions is that each eigenvector forming the local coordinate frame should be in the same main direction as the majority of keypoint-to-neighbourhood point vectors. Hence, the number of positive and negative signs as a result of the dot product between $vecs(KP, \mathcal{N})$ and each of the three $vec(Ortho-Axis)$ are counted. If the amount of positive signs is greater than the amount that are negative, then an eigenvector direction remains as is, otherwise, the eigenvector is flipped by changing its sign (i.e., positive to negative or vice versa). This procedure is applied for both the x -axis and z -axis of \mathcal{F} . The unambiguous, repeatable y -axis is therefore the cross product of the x -axis and z -axis, whose directions have already been verified.

After forming the rotation and translation-invariant, repeatable \mathcal{F} , \mathcal{N} is transformed from its original, global coordinate frame to the local \mathcal{F} (Equation 3.6). This is to ensure the subsequent descriptors are also rotation and translation-invariant. First, the coordinates of \mathcal{N} are translated relative to the KP . Then \mathcal{N} is rotated with respect to $R_{\mathcal{F}}$, which is the 3×3 rotation matrix comprising the repeatable, direction-verified eigenvectors forming \mathcal{F} .

$$\hat{\mathcal{N}} = R_{\mathcal{F}}(\mathcal{N} - KP) \quad (3.6)$$

where, $\hat{\mathcal{N}}$ is the scale-invariant and rigid-invariant local point neighbourhood used to compute KP 's descriptor.

3.3.2 Local 3D surface description

The 3D surface keypoint descriptor must capture the local topographic morphology of the surrounding surface structure. The descriptor is developed by utilizing point cloud surface information. Specifically, the descriptor is referred to as the radial geodesic distance-slope histogram (RGSH). RGSH encodes the joint distribution of: i) the geodesic distance (i.e., shortest path travelled between two points along the point cloud surface) from the keypoint to all other points in the local neighbourhood, and ii) the slope around each point belonging to $\hat{\mathcal{N}}$.

Delaunay triangulation (Li et al., 2005) is applied on the point cloud surface to generate a mesh representation. Local keypoint descriptor regions now consist of points, edges (point-to-point connections) and triangular faces (formed from three closed edges).

For a given KP with local region $\hat{\mathcal{N}}$, the RGSH descriptor is constructed as follows:

1. Geodesic distances are computed between KP and every point $\mathbb{P}_j \in \hat{\mathcal{N}}$ using Kimmel and Sethian's Fast Marching algorithm (Kimmel and Sethian, 1998). This results in a set of geodesic paths resembling a radial pattern emanating from KP (Figure 3.6).

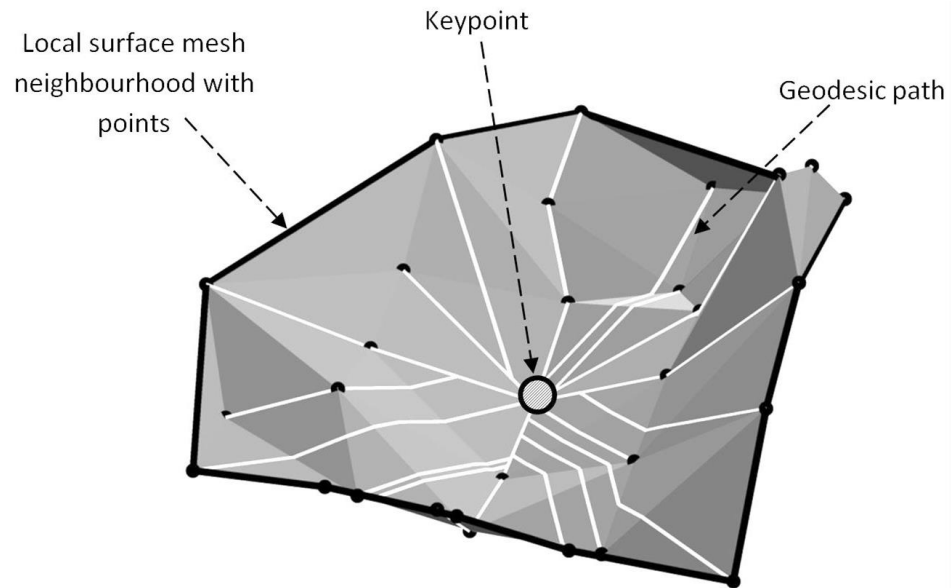


Figure 3.6: Local keypoint neighbourhood on the surface mesh. Geodesic paths running in radial pattern from keypoint (neighbourhood focal point) to all its neighbouring points (in black) are shown.

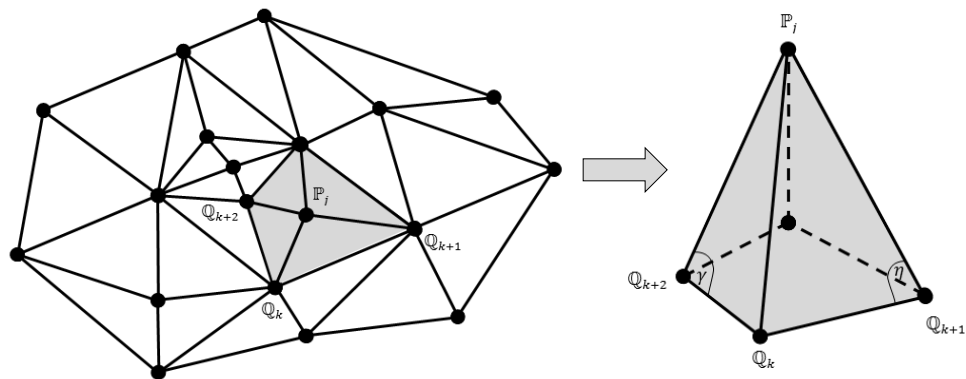


Figure 3.7: Illustration of 1-ring mesh neighbourhood around a point \mathbb{P}_j on the surface mesh and the geometry for obtaining its slope.

2. For every \mathbb{P}_j , the slope around each of their 1-ring mesh neighbourhoods are estimated. The 1-ring neighbourhood consists of the mesh faces formed from the surrounding point clouds \mathbb{Q} , which share an edge with \mathbb{P}_j and also share edges between themselves (Figure 3.7). $Slope_{1-ring}$ is the magnitude of the 1-ring area gradient (Equation 3.7).

$$Slope_{1-ring} = \left\| \nabla A_{1-ring} \right\| \quad (3.7)$$

where,

- A_{1-ring} is the 1-ring area,
- ∇A_{1-ring} is the gradient of A_{1-ring} relative to \mathbb{P}_j ,
- $\| \ \|$ is the magnitude.

A_{1-ring} is the sum of each triangular mesh face area. ∇A_{1-ring} is computed using the cotangents of the angles in the two triangles opposite the edge formed by \mathbb{P}_j and its neighbour \mathbb{Q}_k (where $k = 1, 2, 3, \dots$, # of 1-ring neighbourhood points belonging to \mathbb{P}_j) (Equation 3.8; Pinkall and Polthier, 1993; Desbrun et al., 1999).

$$\nabla A_{1-ring} = \frac{1}{2} \sum_{k=1}^{\# \text{ 1-ring points } \in \mathbb{P}_j} (\cot \eta_k + \cot \gamma_k) (\mathbb{Q}_k - \mathbb{P}_j) \quad (3.8)$$

After obtaining a radial geodesic distance and slope measure for every \mathbb{P}_j , both pairs of values are normalized relative to their maximum values within the local neighbourhoods of each keypoint. Then both sets of values are projected into a 2D histogram \mathcal{H} . \mathcal{H} is divided into a space of uniform $B \times B$ bins (where bin intervals are: $b = 1, 2, \dots, B$). Figure 3.8 is an example of the RGS descriptor with 6 x 6 bins (the size of B is determined experimentally using a ‘tuning’ dataset and details are provided in Chapter 5, Section 5.1.1).

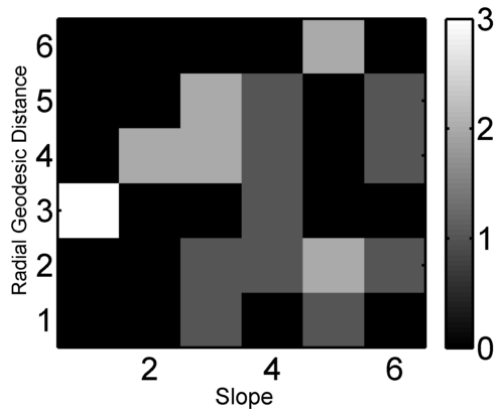


Figure 3.8: Illustration of the 2D radial geodesic distance-slope histogram (the gray scale shows binning frequency).

The Chi-square distance χ^2 (Berretti et al., 2013) is used to measure the similarity cost *SimCost* between a source histogram descriptor \mathcal{H}_s with a target histogram descriptor \mathcal{H}_t (Equation 3.9). Lower costs indicate higher similarity between a source and target keypoint.

$$SimCost_{\chi^2}(\mathcal{H}_s, \mathcal{H}_t) = \frac{1}{2} \sum_{b=1}^B \frac{(\mathcal{H}_s(b) - \mathcal{H}_t(b))^2}{\mathcal{H}_s(b) + \mathcal{H}_t(b)} \quad (3.9)$$

3.3.3 3D Keypoint matching using RGSF descriptor

The objective here is to establish optimal one-to-one, source to target *KP* correspondences with a minimum total matching cost. This is a combinatorial optimization problem, i.e., where bijective correspondences are sought at the lowest possible cost. The bipartite graph matching approach is applied to address this problem. This method for finding point-to-point feature correspondences has been utilized in other related point matching works such as by Belongie et al. (2002). Alternative approaches for matching keypoints with descriptors include the ‘Nearest Neighbour Distance Ratio’ (Szeliski, 2010). However, unlike bipartite graph matching, nearest-neighbour based descriptor matching methods are dependent on user-defined matching acceptance thresholds. To solve for the correspondences via bipartite graph matching, firstly, a $(m \times n)$ cost matrix $SimCost_{ij}$ is formed using Equation 3.10 for every permutation, i.e., every source and target *KP* combination pair.

$$\min \sum_{ij} SimCost_{ij} \quad (3.10)$$

Let u_i be the source *KPs* and v_j the target *KPs*, where $i = 1, \dots, m$ and $j = 1, \dots, n$, (m is the total number of source *KPs* and n is the total number of target *KPs*). In cases

where $m \neq n$, ‘slack’ (or ‘dummy’) nodes are used to ensure a square $SimCost_{ij}$. Each entry into the $SimCost_{ij}$ matrix is essentially a weight associated with a bipartite graph edge (u_i, v_j) (Figure 3.9). To solve the bipartite graph matching optimization by minimizing the total cost of Equation 3.10, the Hungarian algorithm (Bourgeois and Lassalle, 1971) is used (Appendix A). During the optimization, if a source KP has a large cost with respect to the target KPs , i.e., it has no existing point correspondence; it is assigned to a dummy node and recorded as a non-match.

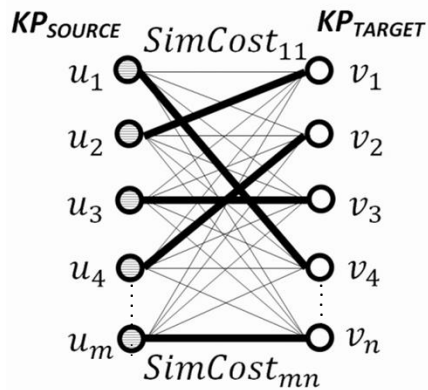


Figure 3.9: Example of bipartite graph for keypoint point matching (Thick lines are the bipartite edges which show the final one-to-one source to target correspondences).

3.3.4 Removal of 3D keypoint correspondence outliers

The output of the Hungarian algorithm is the point correspondences giving the least total cost. To co-register the source point clouds to the target point clouds, transformation parameters have to be computed via a 3-D conformal transformation. However, before

the final transformation is computed, false source to target KP correspondences are filtered (i.e., outliers). An approach similar to RANSAC (Fischler and Bolles, 1981) is employed for this purpose. However, a slight modification is made to the typical RANSAC framework at the threshold-based inlier-checking phase, by instead employing a threshold-free approach.

The classical RANSAC approach begins by randomly selecting the minimum number of point matches required to compute the transformation parameters. This is the ‘hypothesis generation’ phase. Afterwards, the estimated parameters must be validated via a ‘hypothesis verification’ step. In the first stage of hypothesis verification, all the source points are projected into the target space using the estimated parameters. In the second stage, correct/inlying matches are counted by checking the Euclidean distance between target points and the projected source points. Inliers are accepted if the Euclidean distance is less than a user-defined distance threshold.

Source and target point clouds may be in different coordinate systems, and in this research, no prior information about their respective coordinate systems is assumed to be known. Therefore, it becomes difficult to manually set an appropriate user threshold for inlier-checking. Additionally, thresholds used for inlier counting is subjective and an ‘acceptable’ threshold may vary from one user to another.

Instead, this problem is eliminated by employing a threshold-free, inlier consistency check. The concept behind the proposed inlier-checking is based on the verification of initial descriptor-based point matches using the spatial nearest neighbour between source and target points. For example, consider a correct descriptor-based match, $SourcePt_A$

Algorithm 3.1. REMOVAL OF OUTLYING KEYPOINT CORRESPONDENCES

1. Randomly select a triplet of point correspondences from the initial correspondence set acquired via descriptor matching.
 2. Using the randomly sampled triplet set, compute the 3D conformal transformation parameters via Horn's closed form solution (Horn, 1987).
 3. Project all source points to the target dataset using the estimated parameters.
 4. For all initial source to target point correspondences acquired from the descriptor matching phase, determine how many of these source points (when projected to the target) are also the spatial nearest neighbours to their corresponding target points. This is recorded as the total inlier count.
 5. Repeat steps 1 through 4 for a maximum of L iterations (L is determined using the approach from Fischler and Bolles (1981)). At each iteration, check the total inlier count. Update the set of inliers if it is greater than those found at previous iterations.
 6. After exiting the loop (steps 1 to 5), re-estimate (i.e., refine) the 3D conformal parameters by a non-linear least squares adjustment (Luhmann et al., 2006) using all the verified inliers.
-

and $TargetPt_A$. When $SourcePt_A$ is projected to the target point cloud dataset, then the transformed $SourcePt_A$ and $TargetPt_A$ should also be nearest neighbours on the target point cloud domain. Specifically, the transformed $SourcePt_A$ should have minimal spatial distance with $TargetPt_A$ when compared to the other target points. Therefore, this consistency check, which utilizes both descriptor and spatial domains, will accept inlying matches if estimated parameters generated via the random sampling-based hypothesis generation are correct. The overall procedure is presented in Algorithm 3.1. After step 6 in Algorithm 3.1 is completed, the source and target point clouds are co-registered (aligned) using the estimated parameters.

3.4 Summary

A 3D approach for aligning 3D point clouds has been proposed. First, a method for automatically extracting scale-invariant keypoints was developed. The keypoint detector used surface curvature as a measure to identify points of sharp topographic variation. Surface attributes such as the local slope around points and geodesics distances between points were used to form a histogram-based keypoint descriptor. The descriptors provided a unique identifier for the keypoints. The similarities of the descriptors were assessed and matched using the Hungarian algorithm (bipartite graph matching). Outliers were filtered using a threshold-free RANSAC. In the next chapter, an independent, alternative co-registration approach based on the 2D height map representation of the 3D point clouds is presented.

4. A Height Map-based Approach for Point Cloud Alignment

A 2D height map keypoint matching framework is proposed to address the alignment of 3D point clouds from multiple data acquisition platforms. The approach uses height map image pairs as input (i.e., a source and target height map). These height maps are generated directly from 3D point cloud data. This is done by projecting the 3D point cloud dataset along its the z -axis direction onto the x,y -plane, followed by inverse distance weighting interpolation (Childs, 2004).

Similar to the 3D-based co-registration method presented in Chapter 3, the following 2D approach does not require any approximate matching between the source and target and it assumes that the point cloud datasets to be aligned are in different coordinate systems. First, distinct 2D keypoints on the source and target height maps are extracted using a multi-scale wavelet approach. Afterwards, scale, rotation and translation invariant height map-based 2D descriptors are generated and utilized for keypoint matching. The proposed 2D descriptor is inspired by the dense scale invariant descriptor (DSID) originally developed by Kokkinos et al. (2012). It is based on two modifications to the DSID, which include the use of Gabor filter derivatives and the Rapid Transform.

4.1 Height Map-based Point Cloud Alignment

Methodology

The height map-based 2D keypoint correspondence pipeline has three main phases as illustrated in Figure 4.1: i) *Multi-scale 2D keypoint extraction*, where a wavelet transform is adopted to create a multi-scale representation of the height map image. This supports the extraction of distinct 2D keypoints across the height map image scale-space using an energy function. Adaptive non-maxima suppression is then applied to retain strong and well-distributed keypoints. Extraction is performed on both the source and target height maps, ii) *Generation of scale, rotation and translation-invariant 2D keypoint descriptors*, where attributes / descriptors are assigned to the detected keypoints. The descriptor for each keypoint is generated in two phases. It begins with log-polar sampling and mapping of derivatives computed from local height map patches around the keypoint. The log-polar strategy enables scale and rotation invariance. However, corresponding source and target log-polar descriptors are prone to cyclic shifts depending on the magnitude of their scale and rotation differences. To make the descriptor translation-invariant, the Rapid Transform (Reitboeck and Brody, 1969) is utilized, and iii) *Height map image keypoint matching*, where a bi-directional (i.e., source to target and vice versa) descriptor matching is used to find corresponding keypoints. Outliers are then filtered using the modified, threshold-free RANSAC method proposed in Chapter 3. Finally, the matched keypoints are used to compute a 3D conformal transformation for source to target point cloud alignment.

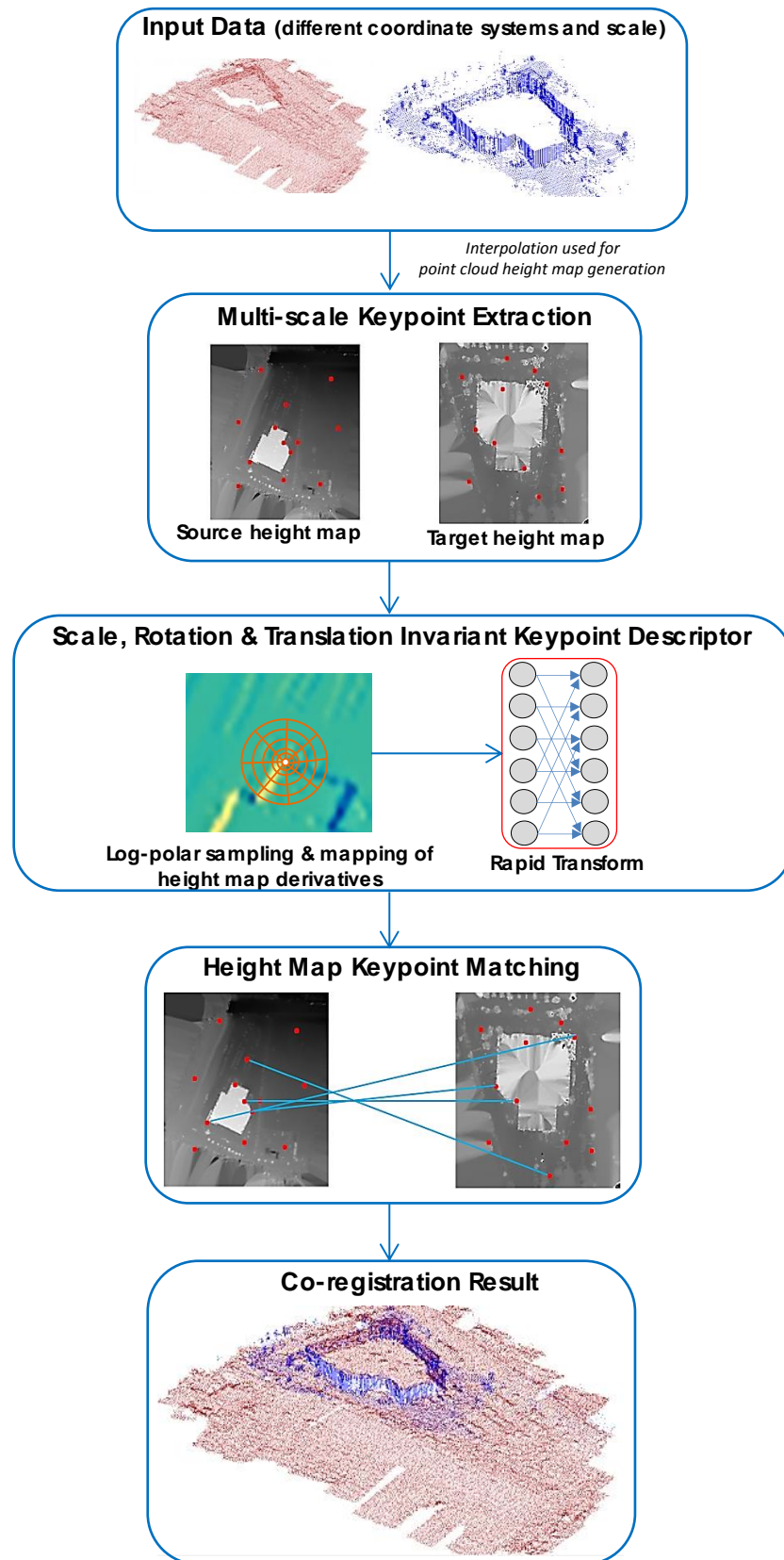


Figure 4.1: Overview of the height map image point matching approach for co-registering 3D multi-sensor point clouds.

4.2 Multi-scale 2D keypoint extraction

In the first step of the height map matching framework, 2D keypoints are automatically extracted. The proposed multi-scale keypoint extraction approach is based on the Dual Tree Complex Wavelet Transform (DTCWT) (Kingsbury, 1998). The utilization of the DTCWT for extracting keypoints was inspired by the method in Fauqueur et al. (2006). They used a DTCWT-based keypoint energy function to determine the points of interest on images. Their function required two user-specified scale space-related parameters. The proposed keypoint extraction framework in this dissertation is similar to Fauqueur et al. (2006). However, an alternative parameter-free keypoint energy function is utilized in combination with an adaptive non-maxima suppression (Brown et al., (2005)) to acquire salient, well-distributed keypoints on the height map images.

There are several existing scale-space extrema-based keypoint detection methods one can utilize. In addition to 'Difference-of-Gaussian' and Lindeberg's scale-space method, wavelets also provide an approach for scale-space representation. Wavelets are well developed in the field of scale-space theory for multiscale feature detection (Mallat and Zhong, 1992).

For addressing feature matching problems, keypoint descriptors typically achieve scale-invariance (i.e., the capability to perform matching between datasets which differ by a scale factor) through the use of a front-end keypoint detector such as SIFT or the Harris-Laplacian operator. However, the estimation of local scales from front-end keypoint detectors such as SIFT can be unstable (Dorkó and Schmid, 2006; Kokkinos and Yuille, 2008).

For typical image matching problems where methods such as SIFT are usually employed, the general assumption is that the source and target images are captured from the same sensor (i.e., cameras). However, this dissertation uses source and target point clouds collected from different viewpoints (e.g., airborne vs. terrestrial platforms), as well as with different point sampling densities and distributions to form the height map image pairs via interpolation. Therefore, the resultant source and target height map images are heterogeneous since they have different texture variation and noise from each other. This is caused by rasterization during the interpolation process and particularly significant along object boundary edges (e.g., building boundaries) in the urban datasets. Hence, source and target keypoints detected on identical structures (e.g., building corners) may have dissimilar contextual details within their respective local regions of interest as defined by the local scale estimation procedure from a front-end detector such as SIFT. This will negatively impact the descriptor matching process as the source and target keypoint descriptors will be different.

Instead, scale, as well as rotation invariance is achieved directly during the descriptor generation phase through the use of log-polar sampling and mapping around the detected keypoints (Section 4.3). The primary objective during the keypoint detection phase is to detect the most salient points of interest using multi-scale image analysis. Multi-scale or multi-resolution image analyses are particularly useful when trying to identify the strongest interest points of the most prominent structures across the image scale-space (e.g., image pyramids). It is used for simulating the scale-space representation of real world objects as typically perceived by human vision. That is, as one physically moves

away from an object, the finer details are lost whilst ‘stronger’ and more prominent features remain visible. Wavelet transforms are utilized because these transforms provide a natural, multi-scale representation of an image through a series of smoothing and down-sampling. This supports the extraction of distinct keypoints across the scale-space using the proposed energy function.

4.2.1 2D keypoint extraction using DTCWT

Wavelet transforms are popular in the areas of computer vision (Mallat, 1996; Tang 2011) and remote sensing (Ranchin and Wald, 1993; Martínez and Gilabert, 2009). The discrete wavelet transform (DWT) (Mallat, 1989) is the most commonly applied wavelet transform. The DWT is not shift-invariant and has limited directional selectivity. At each scale level, the 2D DWT provides directional details in three major directions: horizontal, vertical and diagonal (Ranchin and Wald, 1993). However, images naturally contain features in various random orientations and may not be optimally represented via the 2D DWT. For keypoint extraction, it is critical that blobs and multi-oriented edge structures which form corners are well defined. Kingsbury (1998) introduced the DTCWT to overcome some of the disadvantages of DWT. The DTCWT comprises of six complex-valued wavelet functions defined at six different orientations and is approximately shift-invariant. The increased angular resolution with the real and imaginary components captures more image content than the regular 2D DWT (Hill et al., 2005).

As its name implies, the DTCWT uses two wavelet filter trees, one tree produces real coefficients and the other gives imaginary coefficients. However, this is for the one-

dimensional case (e.g., 1D signals). The two-dimensional case is required. For 2D image decomposition, the 2D-DTCWT has a pair of trees for generating real coefficients ($Tree_A, Tree_B$) and another pair for imaginary coefficients ($Tree_C, Tree_D$). In combination, the two pairs of trees form a single set of complex coefficients. The size of the trees is defined by the number of image decomposition (i.e., scale) levels specified by the user. Across the various levels on each tree, a series of high-pass and low-pass filters are used. At each level, the input image is down-sampled and the wavelet coefficients generated by the high-pass and low-pass filters of the four trees are used to form six complex-valued sub-band images (Selesnick et al., 2005). $Tree_A$ and $Tree_B$ produce six real-valued sub-bands, whereas $Tree_C$ and $Tree_D$ generate six imaginary-valued sub-bands. The real and imaginary sub-band images are combined to give the final six complex-valued sub-bands. Each sub-band image ξ corresponds to one of the six directions of the wavelets, i.e., $\{-75^\circ, -45^\circ, -15^\circ, 15^\circ, 45^\circ, 75^\circ\}$ (Coria et al., 2008).

The number of decomposition levels used for the DTCWT depends on the size of the height maps. At extremely low decomposition levels, structural details are lost due to the continuous down-sampling of the height-map and hence provide no benefit for the keypoint extraction process. As a result, no more than three levels of decomposition are exceeded, i.e., the first level of decomposition down-samples the height map at 50%, the second level down-samples at 25% and the final, third level down-samples at 12.5%. Figure 4.2 illustrates the result of DTCWT when applied to a point cloud height map.

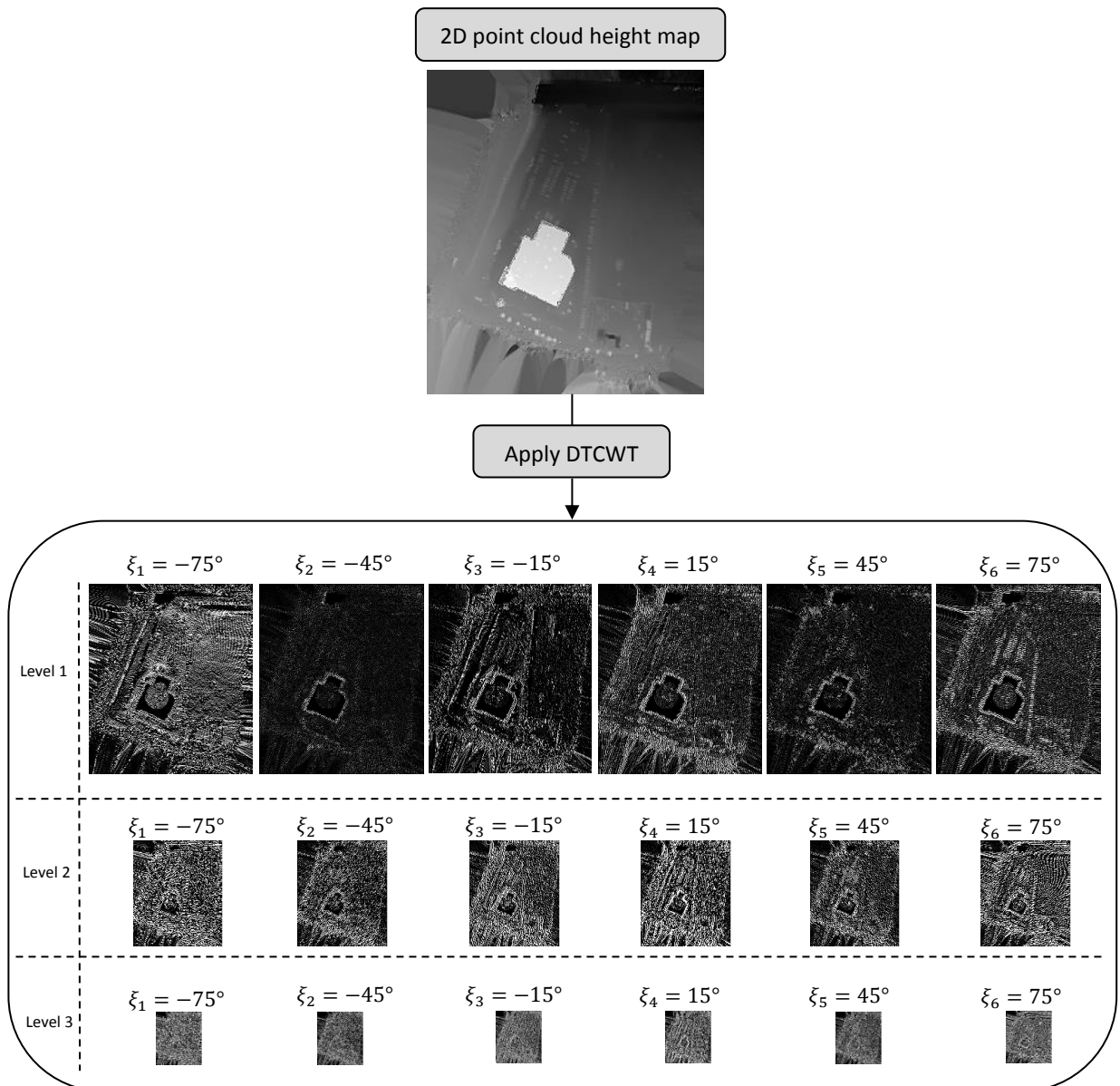


Figure 4.2: Scale-space representation of a height map produced by the dual tree complex wavelet transform at three levels of decomposition. Each level shows the six sub-band images.

Following the generation of DTCWT coefficients, a keypoint energy map KP_{energy} is computed using the harmonic mean of the six sub-band images at each decomposition level (Equation 4.1). Since three decomposition levels were applied, three keypoint energy maps are generated. The use of the harmonic mean as a measure for establishing keypoints has also been applied by Brown et al. (2005).

$$KP_{energy} = \frac{S}{\sum_{b=1}^S \xi_b^{-1}} \quad (4.1)$$

where,

- S is the number of sub-band images (i.e., $S=6$ at each decomposition level) and $b = 1, 2, 3, \dots, S$.

Figure 4.3 shows the keypoint energy maps generated for the three decomposition levels. From each energy map, a search is performed to determine the various local maxima (i.e., keypoints) using the non-maxima suppression (NMS) algorithm (Neubeck and Van Gool, 2006). The use of NMS for directly establishing interest points from a saliency measure has also been applied in previous works by Tuytelaars and Van Gool (2004) and Tombari and Di Stefano (2014). The concept of NMS is that a query location on the energy map is selected as a keypoint if its KP_{energy} is greater than those of its neighbours. A 3x3 neighbourhood similar to Fauqueur et al. (2006) is used to define the set of neighbours around the query pixel. A small neighbourhood was used to ensure that

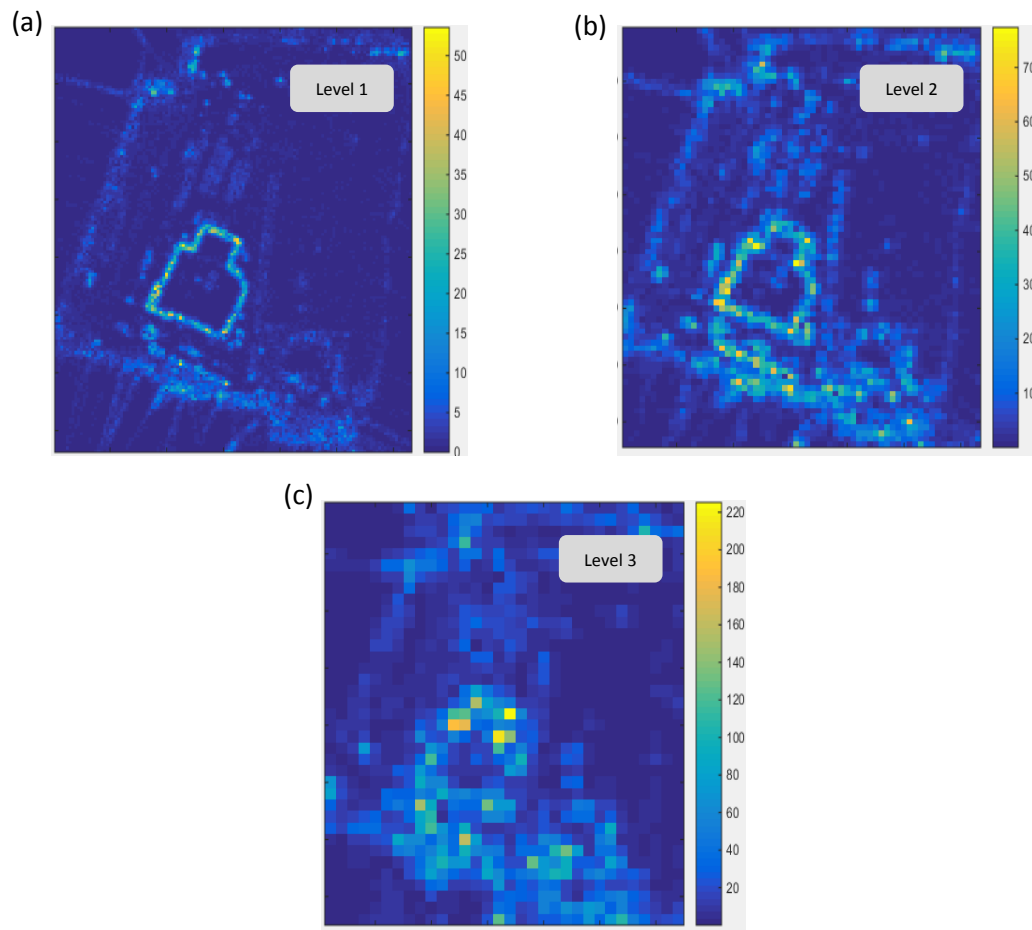


Figure 4.3: Keypoint energy maps generated at each of the three decomposition levels.

(a) Level 1, (b) Level 2, (c) Level 3. Colour bar indicates keypoint energy value for a point on the energy map. Higher values indicate stronger keypoint candidate locations.

local variation is captured as the closer points have higher influence in detecting salient interest points.

After keypoints from all three energy maps have been acquired by NMS, the next step is to retain the strongest keypoints based on their energy responses and remove spurious

keypoints which are in very close spatial proximity or overlap with other keypoints. This is done using an adaptive non-maxima suppression (ANMS) algorithm (Brown et al., 2005). This algorithm prevents an uneven distribution of keypoints by keeping those whose energy (Equation 4.1) is greater than those of its neighbouring keypoints. ANMS compromises between the elimination of relatively weak keypoints and at the same time ensuring a regular distribution of distinct keypoints throughout the height map. Rescaling to the original image scale is applied to ensure keypoint locations from the down-sampled energy maps are in the same pixel coordinate system as the original height map image before ANMS is applied. In contrast to the 2D corner strength function utilized by (Brown et al., 2005), the KP_{energy} measure (Equation 4.1) is used as the ANMS ‘strength indicator’ for filtering interest points on the height map.

To begin the ANMS process, let KP_{num} ($num = 1, 2, \dots$, number of initial *keypoints*) be the set of detected keypoints combined from all three KP_{energy} maps. For each *keypoint* $\in KP_{num}$, a search is performed to find its closest neighbouring keypoint, KP_c which is of greater energy strength. The distances between *keypoints* $\in KP_{num}$ and their respective KP_c are stored and sorted from the largest to smallest. The algorithm then retains those keypoints which have a large distance from their nearest, ‘stronger’ neighbour. A large distance represents a distinct *keypoint* $\in KP_{num}$ that is not suppressed since its KP_c is spatially far away. This criterion encourages a final set of keypoints which are well-distributed on the height map. Therefore, the accepted keypoints are those with the \mathcal{T} largest distances, where \mathcal{T} is the maximum number of final keypoints which the user wishes to keep after suppression. The remaining keypoints

are eliminated from KP_{num} . The parameter \mathcal{T} is dataset specific and depends on the size and coverage of point cloud height map image. For the height map datasets used in the experiments (Chapter 5), $\mathcal{T} = 60\%$ of the total number of detected keypoints accumulated from all three DTCWT levels. This value is used because it was empirically observed that it led to higher true positive keypoint matching rates. Figure 4.4 illustrates sample results of keypoint extraction on a height map before and after ANMS is applied.

4.3 Scale, rotation and translation invariant 2D keypoint descriptor

In this section, a scale, rotation and translation invariant 2D keypoint descriptor referred to as the Gabor, Log-Polar-Rapid Transform (GLP-RT) descriptor is proposed. The descriptor is inspired by the approaches developed in Tola et al. (2010) and Kokkinos et

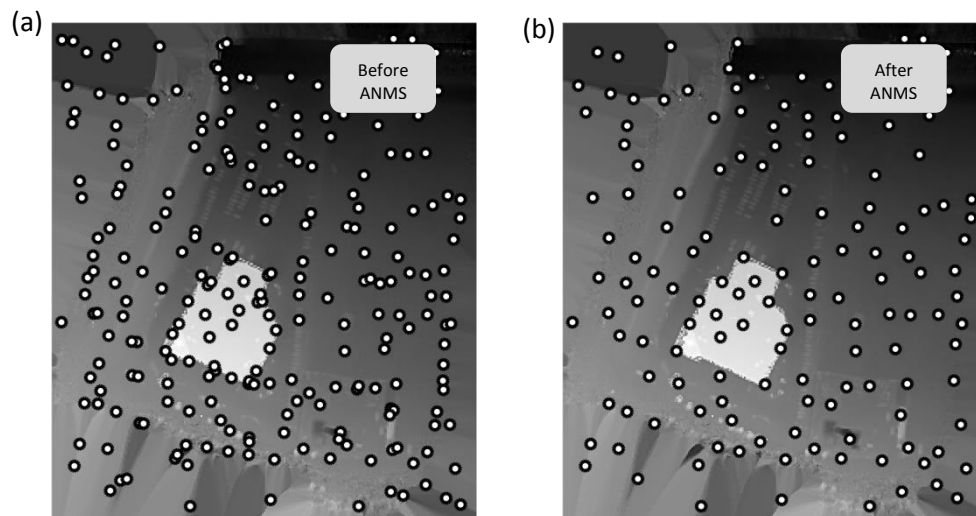


Figure 4.4: Keypoint extraction results. (a) Initial keypoints (before ANMS).

(b) Final keypoints (after ANMS).

al. (2012). Tola et al. (2010) used Gaussian kernel-based directional derivatives sampled on a polar-grid to efficiently compute the so-called dense ‘DAISY’ descriptor. However, the DAISY descriptor is not scale and rotation invariant. Kokkinos et al. (2012) addressed this by applying local log-polar grid sampling and mapping of the DAISY-like, Gaussian-based directional derivatives around image points to achieve scale and rotation invariance. The log-polar transform of an image and its scaled and rotated version is the same (Zokai and Wolberg, 2005). However, the magnitude of the scale and rotation differences between the image and its scaled, rotated version are represented as a cyclical translational shift between their respective log-polar images. Kokkinos et al. (2012) utilized the Fast Fourier Transform (FFT) (Cooley and Tukey, 1965) to achieve shift-invariance.

A similar descriptor framework is utilized, with some variations. The proposed descriptor algorithm consists of the following two general steps: i) log-polar sampling and mapping of Gabor filter-based directional derivatives, and ii) transformation of the preliminary, scale and rotation invariant log-polar-based descriptors formed in i) into a cyclic-shift invariant descriptor using the 2D Rapid Transform (RT) (Reitboeck and Brody, 1969). The following sections provide details on the construction of proposed GLP-RT keypoint descriptor, as well as, the motivation for using the Gabor filter-based derivatives and the 2D RT.

4.3.1 Log-polar sampling and mapping for 2D scale and rotation invariance

In the first step of the descriptor formation, a log-polar grid is applied around the local neighbourhood of a keypoint to determine descriptors characterizing the keypoint based on local height changes. Log-polar grid systems represent the height image information with a space-variant resolution inspired by the visual system of mammals (Traver and Bernadino, 2010). The log-polar grid is a series of concentric rings with exponentially increasing size which are split into various sectors by a set of radial rays projecting from the keypoint.

For any scale and rotation differences between regions around corresponding keypoints on the source and target height map, the log-polar transform is utilized to form source and target height map descriptors which manifest these differences as a translational shift between the two descriptors. The log-polar transform is well-known for its scale and rotation invariant characteristics (Zokai and Wolberg, 2005). It has been used for various image processing applications such as automatic, global image registration (Reddy and Chatterji, 1996), face detection and tracking (Jurie, 1999) and image-based texture classification (Pun and Lee, 2003). Similar to Gabor filters, the use of the log-polar transform is also biologically motivated. Its logarithmic space-variant sampling scheme is reminiscent of the retina as represented in the visual cortex of humans (Schwartz, 1994).

The log-polar transformation is done relative to the center point of the log-polar grid, i.e., the keypoint. The log-polar grid around the keypoint is defined by four parameters:

the minimum ring radius $minR$ and maximum ring radius $maxR$ (both in pixels), the number of bisecting rays M on the grid, as well as, the number of specified concentric rings N . N logarithmically, equally spaced radii values, \mathcal{R}_n (where, $n = 1, 2, 3, \dots, N$) are computed between log-decades 10^{minR} and 10^{maxR} . These logarithmically-scaled radii serve as the radius values used to generate each of the N concentric rings on the log-polar grid. Each concentric ring is partitioned into M uniformly spaced radial rays with angles $\alpha_j = \frac{2\pi j}{M}$ (where, $j = 1, 2, 3, \dots, M$). Sampled points on the log-polar grid are the points of intersection formed by the M radial rays and N concentric rings.

For each ring on the log polar grid, smoothed Gabor-filter based derivatives are generated at four orientations θ_v (where, $v = 1, 2, 3, 4$) and recorded for each sampled point. Orientations are computed in the horizontal (180°), vertical (90°), positive (45°) and negative (-45°) diagonal directions. The number of orientations can be increased but based on experimental analysis there are no significant benefits of increased descriptor performance. However, the trade-offs are disadvantageous, with an increase in descriptor dimensionality and longer computation times. Therefore, four derivative orientations (the procedure for derivative computation is provided in the next section) are used. The locally oriented derivatives of each sampled point on the log-polar grid are then mapped to the log-polar descriptor domain (Equation 4.2; Kokkinos and Yuille, 2008; Kokkinos et al., 2012). The log-polar sampling is done on the Gabor-based derivatives of the height map and not the height map itself since the derivatives provide the intensity-invariant structural information which is useful for the descriptor formation. The log-polar

coordinate system (Figure 4.5, right) comprises of two axes, with each being defined by M rays and N concentric rings. Therefore, the log-polar descriptor domain I_{LP} is a 2D $M \times N$ array. The $minR$, $maxR$, M and N parameters are empirically determined in Chapter 5, Section 5.2.2.

$$I_{LP}[M, N] = [I_G(\mathcal{R}_n \cos \alpha_j + x_{kp}), I_G(\mathcal{R}_n \sin \alpha_j + y_{kp})] \quad (4.2)$$

where,

- I_{LP} is the 2D log-polar descriptor,
- I_G is a directional derivative image for one of the 4 specific orientations,
- (x_{kp}, y_{kp}) is the keypoint on the height map,
- \mathcal{R}_n is the logarithmically-scaled radius ($n = 1, 2, 3, \dots, N$),
- α_j is the sector angle for the log-polar grid ($j = 1, 2, 3, \dots, M$).

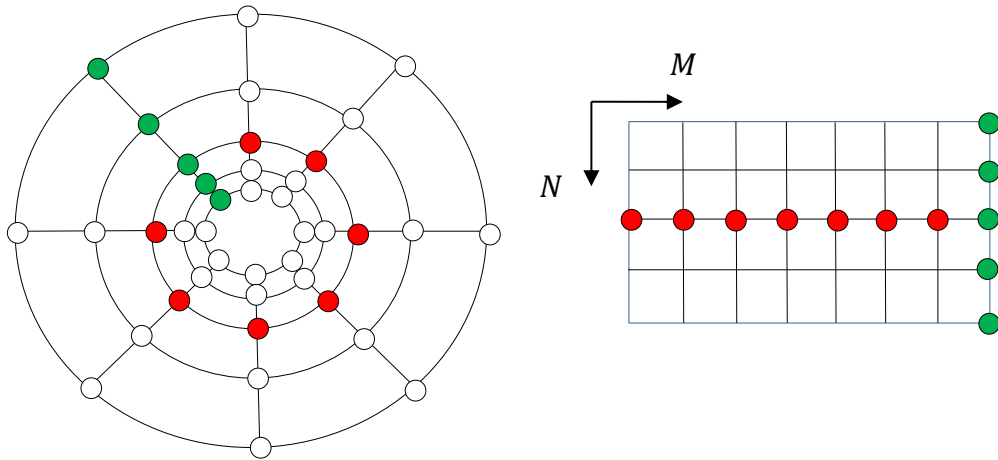


Figure 4.5: Example of log-polar sampling and mapping. Left: Exponential log-polar grid sampling that is applied to height map derivatives. Right: mapping of gridded points into uniformly-spaced log-polar domain, forming a log-polar descriptor (right diagram). Red circles are grids located on the 3rd ring and green circles are located on the 8th ray.

4.3.1.1 Generation of Gabor filter-based derivatives

The motivation for using Gabor filter-based derivatives is due to their robustness to illumination changes, image noise and natural image background variations (Kamarainen et al., 2006). This is important as the height map image pairs to be matched are generated from the different data collection platforms and contain significant texture variations from one dataset to another. The use of the Gabor filter is also mathematically motivated. The Gabor function has greater flexibility in terms of the number of ‘free’ parameters which can be modified to define the function shape, in comparison to the Gaussian function (Jones and Palmer, 1987; Zambanini and Kampel, 2013). The 2D Gabor filter \mathcal{G} , (Equation 4.3), is a sinusoidal plane wave with a defined wavelength and orientation that is modulated by a Gaussian kernel (Hamamoto et al., 1998; Haghigat et al., 2013).

$$\mathcal{G}(x, y, \theta, \sigma_n) = e^{-0.5\left(\frac{W_1^2 + W_2^2}{\sigma_n^2}\right)} \times e^{[i\frac{2\pi W_1}{\lambda}]}$$
 (4.3)

where,

- $W_1 = x\cos\theta + y\sin\theta$ and $W_2 = -x\sin\theta + y\cos\theta$,
- λ is the wavelength of the sinusoidal plane wave and controls the frequency of the \mathcal{G} (where, $\lambda = 2\sigma_n$ as in Konishi et al. (2003)),
- θ is the orientation of \mathcal{G} ,
- i is the imaginary unit,
- σ_n is the scale and is a function of the varying radii for each circle on the

concentric log-polar grid (where, $\sigma_n = \frac{\mathcal{R}_n}{N}$; ($n = 1, 2, 3, \dots, N$)). Note: this is similar to the σ setting in Tola et al. (2010).

The Gabor filter is a complex valued filter (i.e., the filter has a real and imaginary component). In this work, the priority is to capture the structural details from edge features in various orientations on the height map to generate highly discriminative descriptors for the keypoints. Therefore, the imaginary part of the Gabor filter (Equation 4.4) is used since it has been shown to efficiently provide robust edges (Jiang et al., 2009).

$$\mathcal{G}(x, y, \theta, \sigma_n)_{imaginary} = e^{\left[-0.5 \left(\frac{w_1^2 + w_2^2}{\sigma_n^2}\right)\right]} \times \sin\left(\frac{2\pi w_1}{\lambda}\right) \quad (4.4)$$

Two dimensional Gabor-filter derivatives are computed via convolution of the height map image with $\mathcal{G}(x, y, \theta, \sigma_n)_{imaginary}$ in each of the four v directions. For each of these v oriented directions, a form of multi-scale smoothing is applied to the derivatives generated at each ring (Tola et al., 2010). With the increasing radius value for each ring on the log-polar grid, the scale σ of the Gabor filter is also incrementally increased as the concentric rings become larger, i.e., smoothing increases as the ring size increases. This low-pass filtering of the height map image is done to prevent any aliasing effects when computing the derivatives and to ensure the source and target descriptors are as similar as possible. Aliasing is caused by the rasterization of object boundaries (e.g., building edges in urban datasets) as a result of the 3D point cloud to height map interpolation process.

Aliasing also arises due to the exponential space-variant pattern of log-polar sampling (Taberner et al., 1999; Palander and Brandt, 2008).

4.3.2 Descriptor invariance to 2D cyclic-shifts using the Rapid Transform

The log-polar descriptors generated in Section 4.3.1 convert scale and rotation changes between local regions of corresponding keypoints on the source and target height maps into a representation which differs by a cyclical translation (or cyclical shift). This translation difference between source log-polar descriptors and target log-polar descriptors can occur along the horizontal or vertical axes of the log-polar domain and will lead to incorrect point correspondences. Therefore, the cyclic shift is addressed by applying the translation-invariant 2D Rapid Transformation (RT) versus the FFT as used by Kokkinos et al. (2012). The RT was developed by Reitboeck and Brody (1969) for pattern recognition applications. They showed that the RT was computationally more efficient and 10-100 times faster than the translation-invariant FFT. RT was also able to outperform FFT for hand-printed letter recognition in the presence of inclinations and small rotations. In more recent work, Li et al., (2014) developed an RT-based descriptor for texture classification again citing speed advantages over the FFT as the motivation for its usage. In terms of computational efficiency, the RT variables are real numbers, whereas FFT variables are complex numbers, therefore RT requires twice as less storage capacity than FFT.

In this section, the 2D RT algorithm is applied to the 2D descriptors which were initially formed in the log-polar domain (Figure 4.6). Even though the RT is inherently a one-dimensional algorithm, it is extended to two dimensions by applying 1D RT twice. Specifically, the 1D RT is first applied on each row of I_{LP} , thereby generating a ‘row-transformed’ 2D coefficient array. The 1D RT is then used again on each column of the row-transformed 2D array. The output of the 2D RT is the final form of the proposed GLP-RT descriptor, which has a dimensionality similar to Kokkinos et al. (2012), i.e., number of derivative gradient orientations $\times M \times N$.

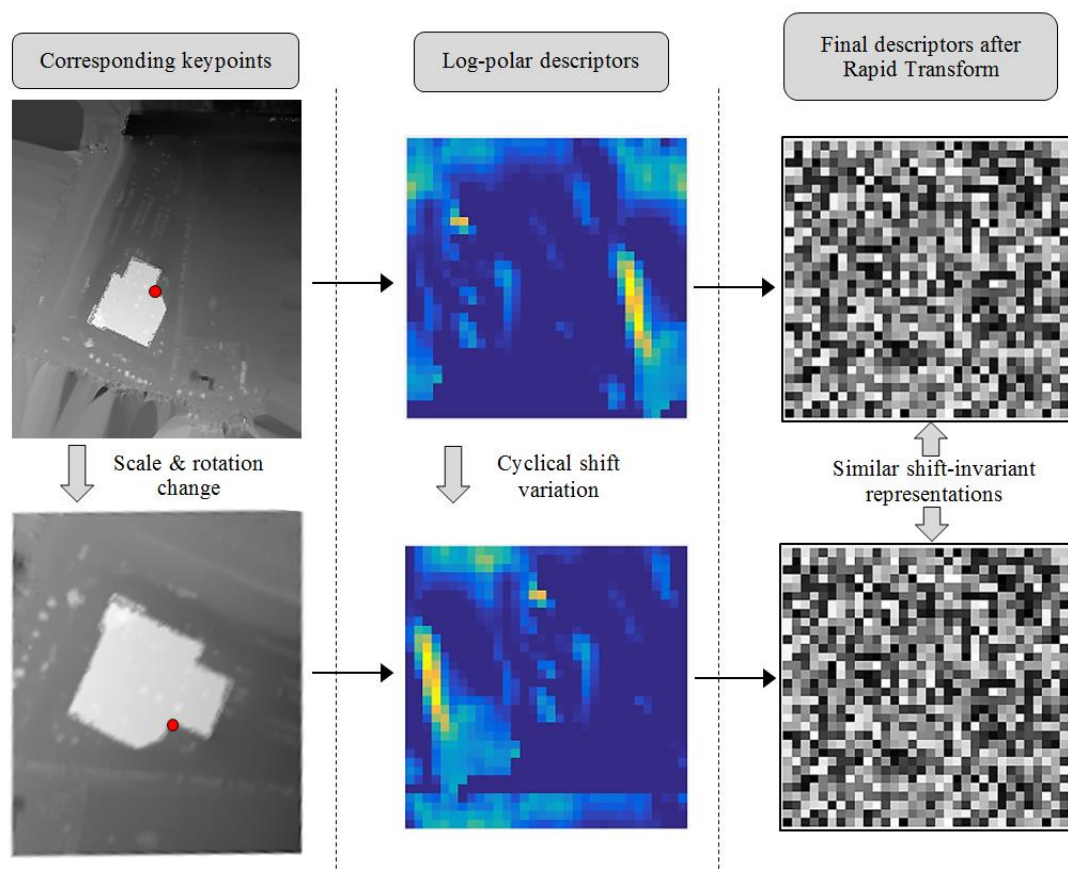


Figure 4.6: Concept of applying Rapid Transform to correct cyclical shift between log-polar descriptors on corresponding keypoints (i.e., red dots on left-most figure).

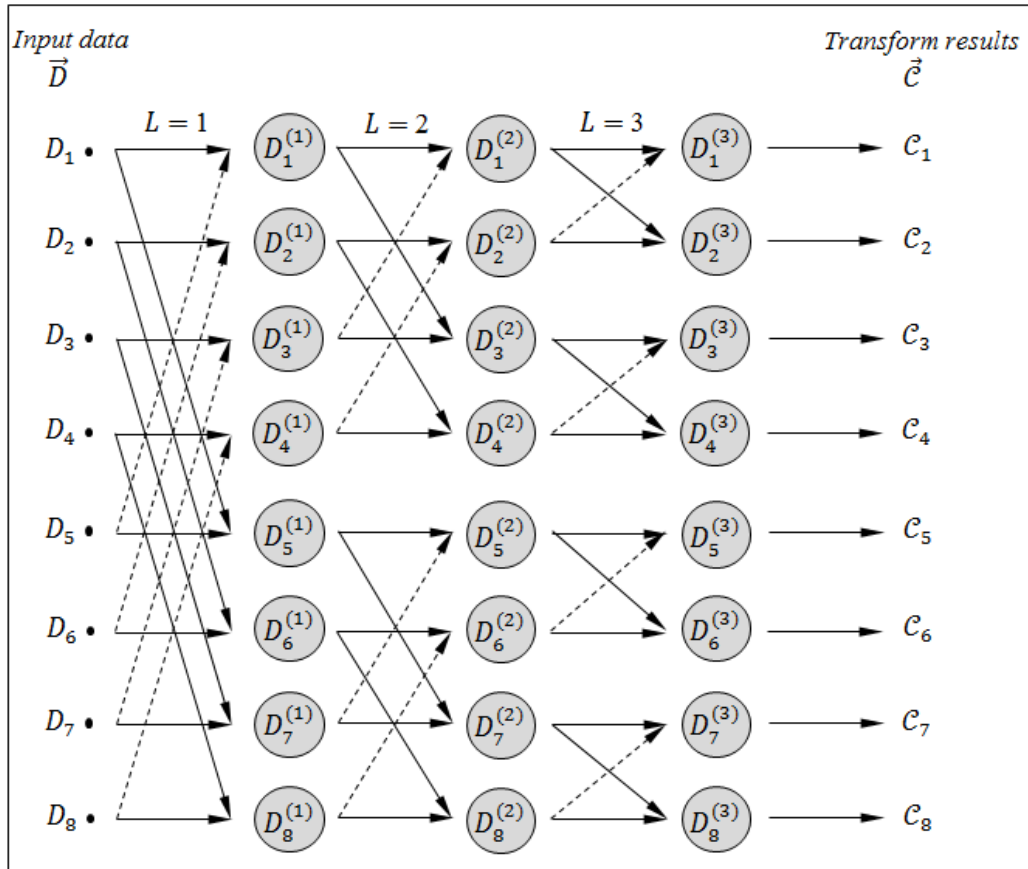
The FFT computation process can be represented in the form of a signal flow (or ‘butterfly’) structure which is based on a divide and conquer approach (Herman, 2013). The computation of RT is also determined using a butterfly structure. An example is presented illustrate the concept of the 1D RT algorithm. For an input 1D data vector $\vec{D} = [D_1, D_2, D_3, \dots, D_K]$ of size K , RT computes the transform coefficients $\vec{C} = [C_1, C_2, C_3, \dots, C_K]$ using the signal flow structure similar to the steps illustrated in Figure 4.7. K is assumed to be of the form $K = 2^p$ where p is a positive integer.

This requirement that K must be a power of 2 does not limit the generality of the RT algorithm, as a zero-padding (Stoica and Moses, 2005) is applied to the next power of 2 for data which requires it. Similarly to FFT, the RT algorithm comprises of a total L transformation stages (where, $L = \log_2 K$ (Herman, 2013)) that are required to convert the original data into the transformed coefficients. Figure 4.7 is an example of the RT with \vec{D} comprising of 8 data points (i.e., $K = 8$) and a total of $L = 3$ transformation stages. At each stage, a pair of commutative functions D (Equation 4.5) is applied on each data element. In the first stage (i.e., $L = 1$), these operators are applied to the initial values of the input data elements whereas, in the subsequent stages (i.e., $L = 2$ and $L = 3$), the operators are applied to the output data elements from the previous stage.

$$\begin{aligned} D_i^{(L)} &= D_i^{(L-1)} + D_{i+K/2}^{(L-1)} \\ D_{i+K/2}^{(L)} &= \left| D_i^{(L-1)} - D_{i+K/2}^{(L-1)} \right| \end{aligned} \quad (4.5)$$

where,

$$- \quad i = 1, 2, 3, \dots, K.$$



Key	
Input data element	•
Output data element	○
Commutative function 1	$D_i^{(L-1)} \xrightarrow{+} D_i^{(L)}$ $D_{i+K/2}^{(L-1)} \xrightarrow{\text{dashed}} D_i^{(L)}$
Commutative function 2	$D_i^{(L-1)} \xrightarrow{ - } D_{i+K/2}^{(L)}$ $D_{i+K/2}^{(L-1)} \xrightarrow{ - } D_{i+K/2}^{(L)}$

Figure 4.7: Computation steps of the 1D rapid transform based on the signal flow (or 'butterfly') structure when $K = 8$.

As shown in Figure 4.7, after the first stage, the value of K is continuously halved and its value is updated at each of the remaining stages. This is due to divide and conquer approach used in the signal flow process which splits the output data sequence at each stage into two individual sequences. Further details on the RT are given in Appendix B.

4.3.3 2D keypoint matching using GLP-RT descriptor

For typical nearest neighbour-based matching, a source keypoint is compared to all the target keypoints by computing the Euclidean distance between their descriptors. The Euclidean distance serves as a measure of descriptor similarity. The corresponding target keypoint (i.e., the nearest neighbour) is chosen as the one giving the smallest Euclidean distance relative to the source keypoint descriptor. However, to increase the robustness of descriptor correspondence determination, the nearest neighbour matching is also applied in the opposite direction to assess the bi-directional similarities of source and target GLP-RT descriptors. That is, a target descriptor is compared with all source descriptors to find its nearest neighbour match. A check is then performed to determine the same point correspondences which are obtained in both directions. Another alternative approach is the ‘nearest neighbour distance ratio’ (Szeliski, 2010). However, this measure was not utilized since it is dependent on a user-defined matching acceptance threshold, which can vary amongst different datasets.

To illustrate the approach, assume $Source_{KPA}$ and $Target_{KPA}$ are true point correspondences. In the first step of the matching process, a nearest neighbour search is applied to obtain the closest target descriptor match $Target_{KPN}$ for $Source_{KPA}$.

Similarly, nearest neighbour search is applied again to obtain the closest source descriptor match $Source_{KP_{NN}}$ for $Target_{KP_A}$. A final point to point correspondence is established if the $Target_{KP_{NN}}$ and $Target_{KP_A}$ are the same points and if $Source_{KP_{NN}}$ and $Source_{KP_A}$ are the same points (Figure 4.8). This process is applied for all source and target keypoints to determine a set of point correspondence pairs. However, outliers (i.e. false correspondences) are a possibility. Therefore, to prune these initial point matches, the RANSAC-based outlier detection method (Algorithm 3.1) developed in Chapter 3, Section 3.3.4 was applied. Recall that the 2D height map coordinates also have an associated elevation (i.e., Z coordinate component). Thus, the inputs for Algorithm 3.1 are 3D keypoint coordinates (i.e., X , Y and Z) of the source and target correspondences, since the objective is to find the most optimal 3D conformal transformation parameters.

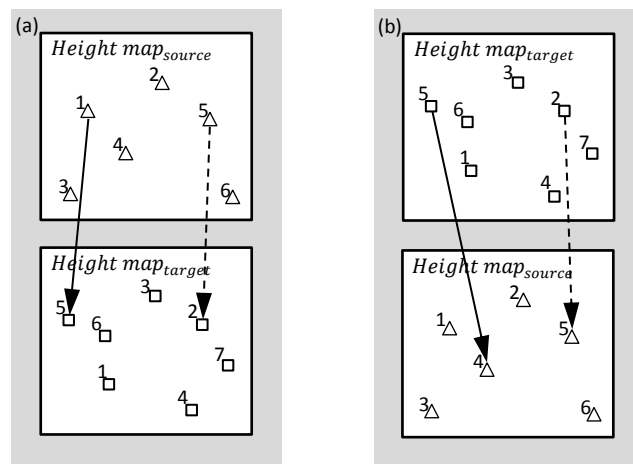


Figure 4.8: Concept of bi-directional keypoint descriptor matching showing a successful correspondence (dashed arrows) and an unsuccessful correspondence (solid arrows).

4.4 Summary

A 2D height map-based keypoint matching approach has been proposed for the alignment of 3D point clouds. First, a Dual Tree Complex Wavelet Transform-based keypoint extraction method was implemented to detect salient interest points on the height maps. After, a 2D keypoint descriptor was developed to characterize the keypoints with a unique identifier. The descriptor was based on log polar sampling and mapping, and the 2D Rapid Transformation to be scale, rotation and translation invariant. A bi-directional matching strategy was then employed to assess and match keypoints based on their descriptor similarities. The threshold-free RANSAC was used to filter outliers. In the next chapter, the results and analysis of the two proposed co-registration methods are presented.

5. Results and Analysis

This chapter presents the results and analysis of the two proposed point cloud alignment approaches. Experiments are performed on multi-sensor, urban and non-urban datasets to individually assess the accuracy of each of the two proposed methods. A comparative study on the two approaches is also done, as well as comparisons with state-of-the-art algorithms. The source and target datasets used differ in terms of scale, 3D rotation and 3D translation. Various experiments are also performed on source and target datasets with different overlapping coverage, point density, spatial point distribution and point details (i.e., missing data gaps). Specifically, the datasets are from three different locations: two different urban areas in Ontario, Canada and one non-urban area in Western Canada. The two urban areas are referred to as “*Loc1*” and “*Loc2*” respectively, whilst the non-urban area is identified as “*Loc3*”.

Also included are experiments which are used to select the respective parameter settings for the developed 3D-based RGSH keypoint descriptor and height map-based GLP-RT keypoint descriptor. Section 5.1 provides details of the experiments and analysis for the 3D-based alignment method. Section 5.2 presents the experiments and analysis for the height map-based alignment method. Finally, Section 5.3 evaluates the two proposed methods relative to each other, as well as with state-of-the-art 3D keypoint-based co-registration methods.

5.1 Results for Method 1: 3D-based Point Cloud Alignment

In this section, results from the first proposed matching framework developed in Chapter 3 are presented, i.e., the 3D-based point cloud co-registration approach. In particular, results are illustrated from the keypoint extraction and descriptor generation phases for datasets that vary in terms of scale, rotation and translation. The capability of the co-registration framework is assessed under two different cases: i) using a ‘controlled’ setting, where the source and target point cloud datasets are from the same sensor acquisition system and time period and also have the same point density, and overlap and ii) using a ‘varied’ setting, where the source and target point cloud dataset are collected at different time periods and generated from different sensor acquisition systems with different point density, partial overlap and deformation. The quality of the scale (s), 3 rotation angles (ω , φ , κ) and 3 translation (T_x , T_y , T_z) parameters are analyzed by means of: i) results provided by least squares adjustment residual statistics from estimation of 3D conformal transformation parameters, and ii) differences in results obtained by the proposed automated method versus those from known reference parameters.

The presented approach is assessed using data from urban and non-urban digital surface models (DSMs). The 3D (x , y , z) point clouds from the DSMs are directly used. Figure 5.1 illustrates a pair of urban DSMs representing coverage over York University, Toronto, Ontario, Canada (*Loc2*). The DSM in Figure 5.1(a) was generated using aerial photos acquired in 2005, whilst Figure 5.1(b) shows a DSM produced from airborne

LIDAR data collected in 2009. Figure 5.2 shows natural (non-urban) DSMs of the Columbia Icefield, situated along the border of Alberta and British Columbia, Canada (*Loc3*). The DSM in Figure 5.2(a) was generated using aerial photos from 1950 and the DSM in Figure 5.2(b) was produced using WorldView-2 satellite imagery acquired in 2010. In addition to co-registration experiments, also presented are the empirical results used for selecting the number of bins for the RGSB descriptor. The planimetric and vertical positioning accuracy of the urban dataset was in the range of 0.2m to 0.5m. The planimetric and vertical positioning accuracy of the non-urban dataset was in the range of 2.0m to 5.0m.

5.1.1 Empirical selection of RGSB descriptor bin size

The bin size B is a critical parameter for the co-registration experiments as it defines the RGSB descriptor's discriminability (i.e., the descriptor's uniqueness for each keypoint). The number of bins was experimentally determined using a 'tuning' dataset based on the bipartite graph descriptor matching. The use of 'tuning' datasets to set the parameters of 3D feature detectors and descriptors has also been applied in similar works, such as Salti et al. (2012). The tuning dataset comprises of 4 arbitrarily selected 'training' sites from each of the 4 DSMs in Figures 5.1 and 5.2 (these sites are labeled as 'Training area 1', 'Training area 2', 'Training area 3' and 'Training area 4'). For each of the 4 sites, manually defined transformation parameters were applied to generate scaled, rotated and translated versions of the original point clouds. In this way, each of the 4 training sites

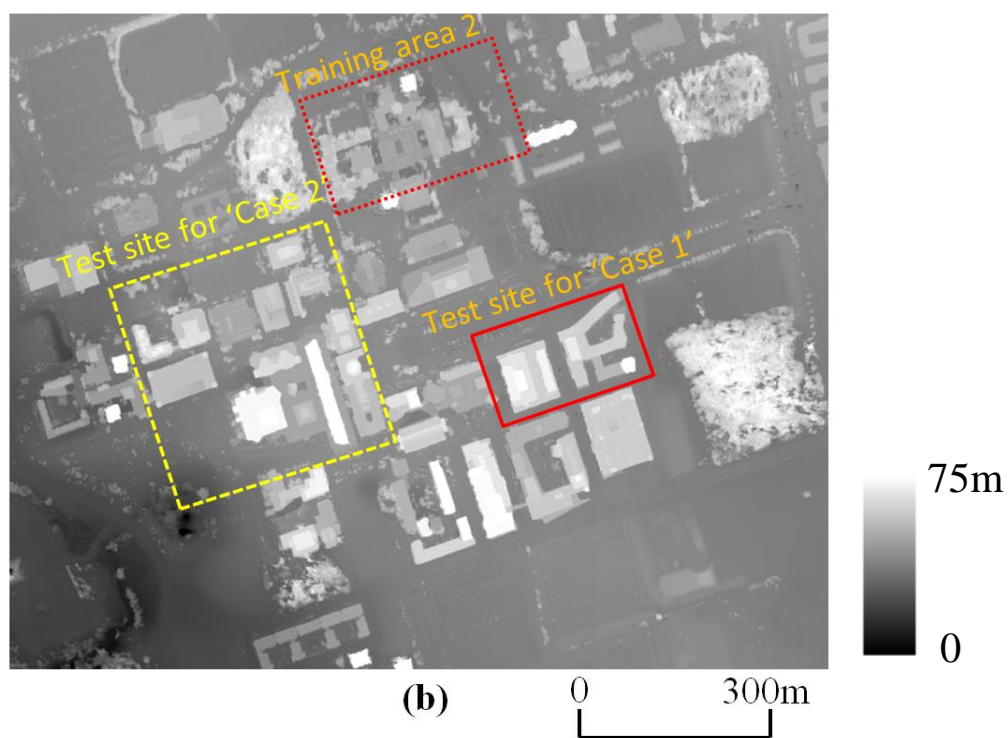
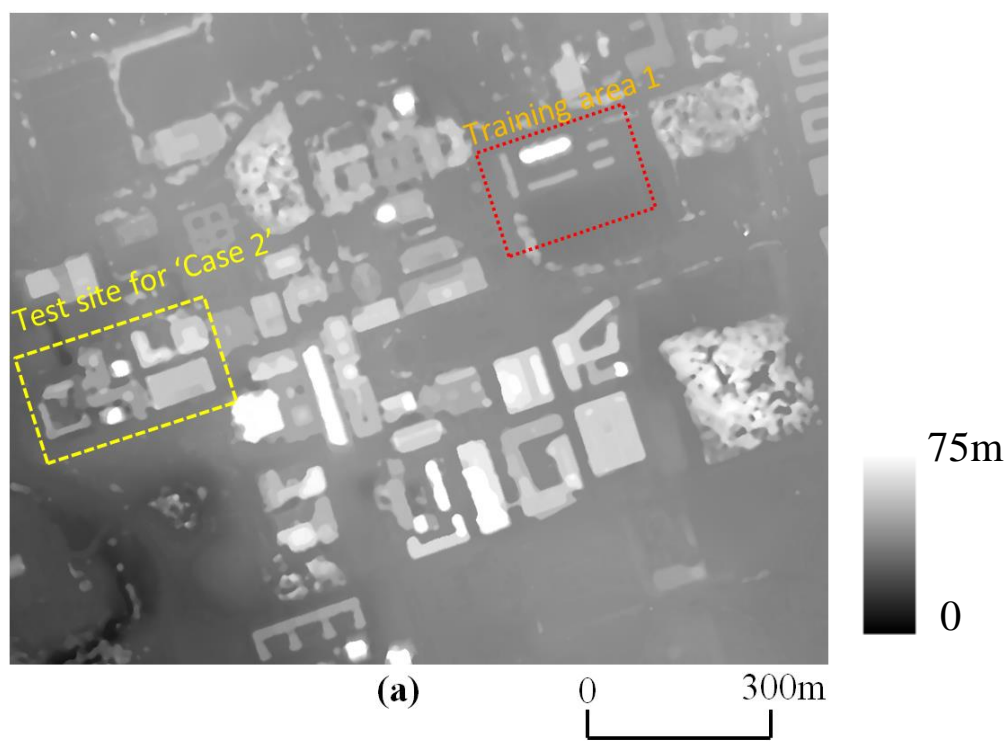


Figure 5.1: Urban DSMs used for co-registration experiments to evaluate the proposed 3D-based alignment method. (a) Aerial photo DSM (b) Aerial LIDAR DSM.

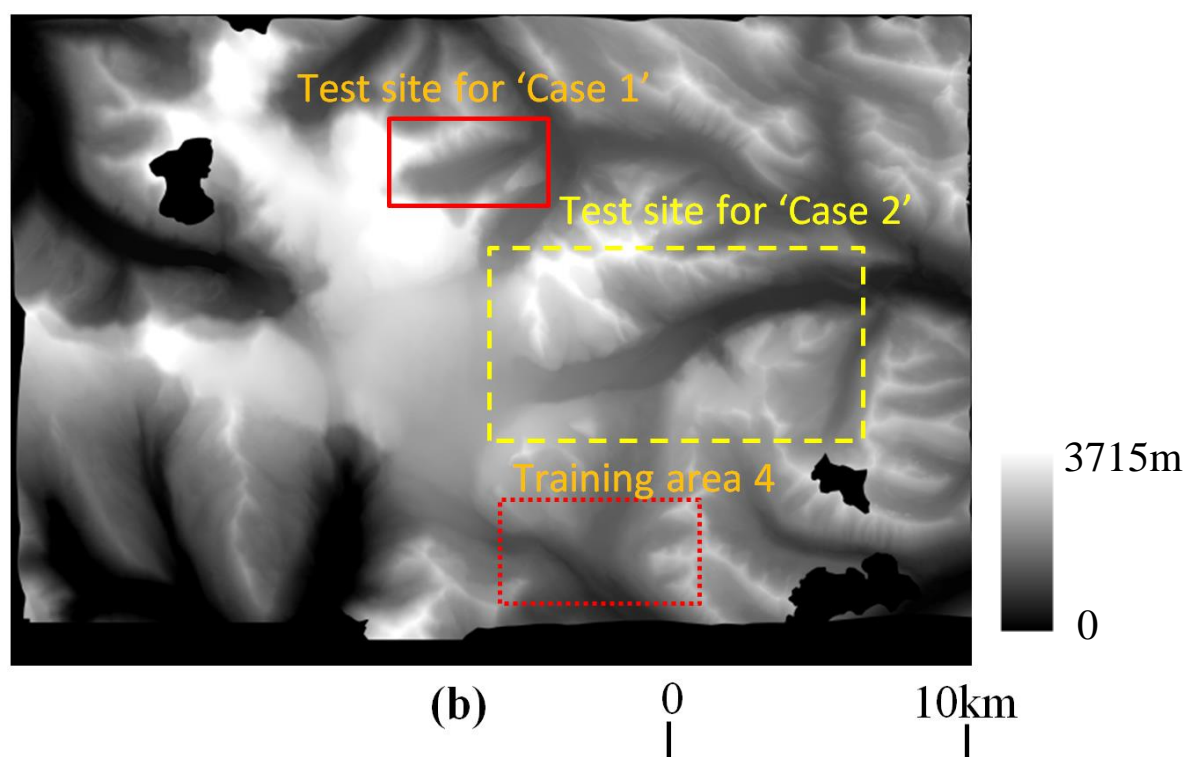
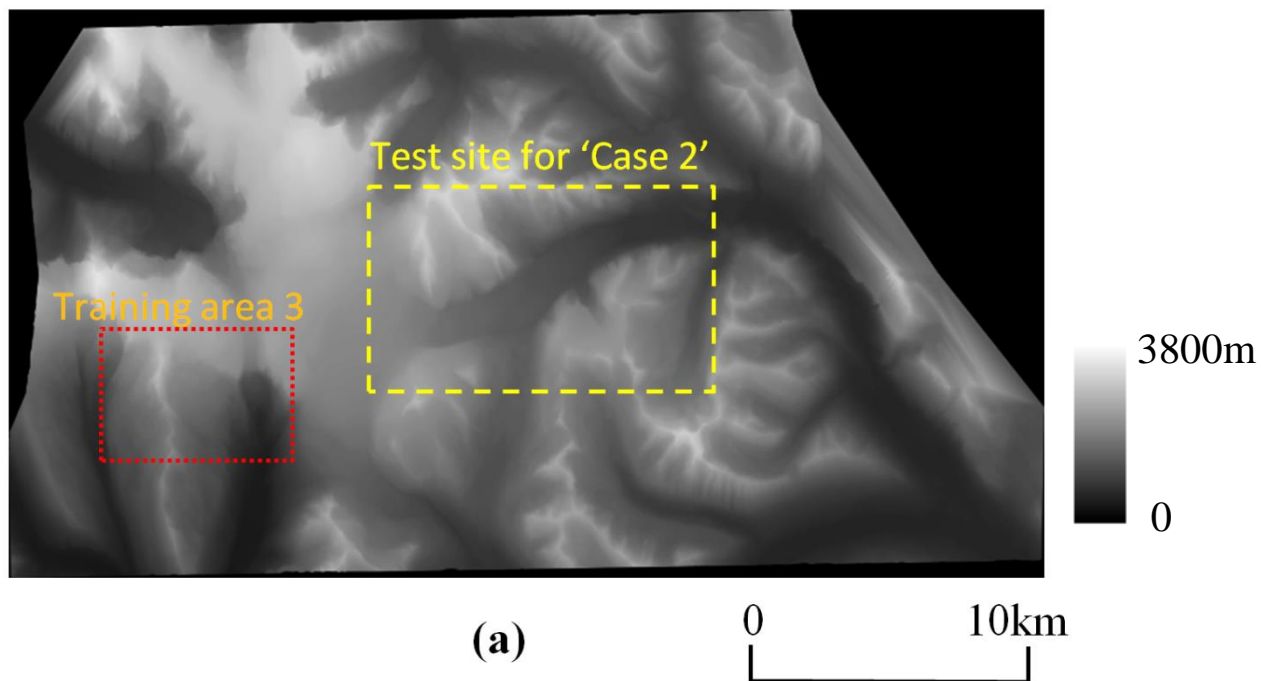


Figure 5.2: Icefield (Non-Urban) DSMs used for co-registration experiments to evaluate the proposed 3D-based alignment method. (a) Aerial photo DSM (b) WorldView-2 DSM.

has source and target point clouds to be used for matching. The manual parameters applied to generate the respective target point cloud datasets for Figures 5.1(a), 5.1(b), 5.2(a) and 5.2(b) are shown in Table 5.1.

A low value of B (i.e., coarse bin resolution) can lower the discriminative power of the descriptor. This would lead to wrong point matching results, since the descriptors would lose some of their uniqueness due to the coarse bin cell partitioning. On the other hand, a dense bin resolution with fine bin cell partitioning will have the opposite effect and ‘over-sensitize’ the descriptor, thus making it difficult to find similar matching source and target descriptors.

Table 5.1: Manually-defined transformation parameters used for generating target point clouds of the 4 training sites in the tuning dataset.

Transformation Parameter	Training area 1	Training area 2	Training area 3	Training area 4
s	0.3	0.4	0.5	0.6
ω ($^\circ$)	3	5	7	9
φ ($^\circ$)	2	4	6	8
κ ($^\circ$)	1	4	7	10
T_x (m)	10	15	20	25
T_y (m)	12	14	16	18
T_z (m)	14	18	22	26

To measure the performance of various bin sizes and its effect on the descriptor’s matching performance, *recall vs. 1-precision* graphs (Ke and Sukthankar, 2004) were utilized. The *recall* (Re) metric (Equation 5.1) provides an indication of the number of true positive (TP) matches found after matching relative to the total number of actual

correct matches (the total number of correct matches are manually checked and known a priori). The *1-precision* ($1 - P$) metric (Equation 5.2) is the number of false positive (FP) matches relative to the total number of recovered point matches (including both TP and FP matches). High *recall* and low *1-precision* will indicate optimal bin size. A TP is considered to be two matching keypoints from the same corresponding positions on the source and target point cloud surfaces. Likewise, a FP is recorded when two matching keypoints come from different positions on the source and target point cloud surfaces.

$$Re = \frac{\text{number of TPs found after matching}}{\text{number of correct matches known a priori}} \quad (5.1)$$

$$1 - P = \frac{\text{number of FPs found after matching}}{\text{total number of FPs and TPs found after matching}} \quad (5.2)$$

The RGS descriptor was evaluated at the following coarse-to-dense bin sizes: $B = 2, 4, 6, 8, 10, 12, 14$. Figure 5.3 shows the *recall vs. 1-precision* graphs. Individual *recall vs. 1-precision* graphs was generated for the urban and non-urban training sites respectively. In Figure 5.3(a), the best performance was achieved at $B = 6$ for the urban training sites. For the non-urban training, the best performance was attained at $B = 8$ (Figure 5.3(b)). However, this was closely followed by $B = 6$. This was reflected upon observation of the matching results from two non-urban training sites, where there was minimal disparity between the number of correspondences at both of these bin resolutions. When $B = 8$, there were 147 TP matches and 4 FP matches. Whilst when

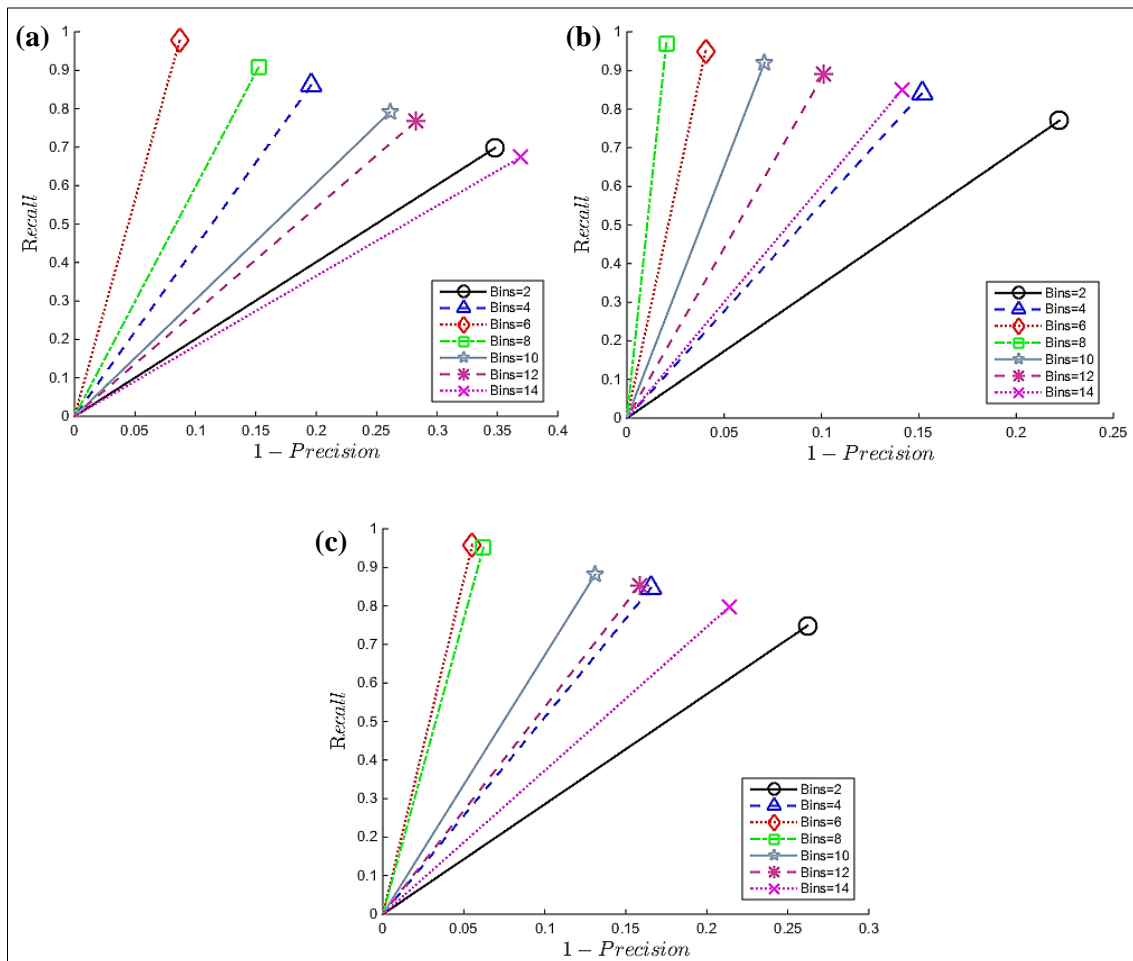


Figure 5.3: *Recall vs. 1-precision* graphs for selecting optimal bin size of the RGS descriptor across a range of coarse to dense bin resolutions using the DSM tuning dataset. (a) Plot for 2 urban training sites. (b) Plot for 2 non-urban training sites. (c) Plot for entire tuning dataset (2 urban and 2 non-urban training sites combined).

$B = 6$, 145 TP matches and 6 FP matches were found. Therefore, to get an overall indication of a suitable bin size across urban and non-urban scenes, a *recall vs. 1-*

precision plot was also generated, when the TPs and FPs of the urban and non-urban training sites are combined (Figure 5.3(c)). The highest *recall* rate and the lowest *1-precision* rate occurred at $B = 6$. Therefore, a histogram bin resolution of 6x6 was used for the RGS descriptor in the experiments as this produced the highest matching success rate based on empirical observations.

5.1.2 Case 1: Same sensor datasets, different coordinate systems

The method is assessed using a ‘controlled’ environment. In this case, a target point cloud dataset was generated by applying manually defined 3D conformal parameters to a source point cloud dataset. Hence, both the source and target data to be co-registered are from the same sensor with the same point density and overlap. In this way, the keypoint extraction, descriptor correspondence and co-registration results were analyzed between a source dataset and target dataset without the influence of data noises and artificial geometric deformations/distortions, which may arise when trying to match multi-sensor and multi-temporal datasets.

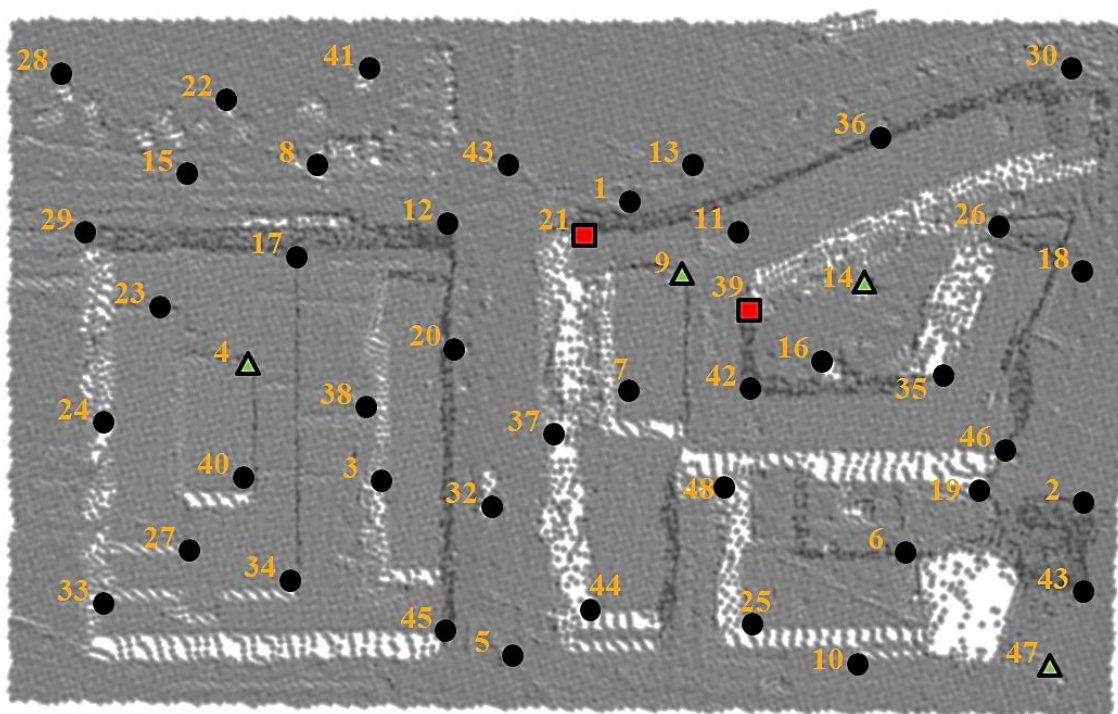
To demonstrate a sample result of the keypoint extraction and descriptor matching processes, the area labeled as ‘Test site for Case 1’ in Figure 5.1(b) was used. This urban site is the source point cloud dataset and comprises of two buildings with a coverage of 44,055m². The target point clouds were generated by applying the following scale,

rotation and translation parameters to the source point cloud dataset: $s = 0.7$, $\omega = 15^\circ$, $\varphi = 30^\circ$, $\kappa = 45^\circ$, $T_x=3\text{m}$, $T_y=5\text{m}$, $T_z=7\text{m}$. These are the ‘reference’ parameters.

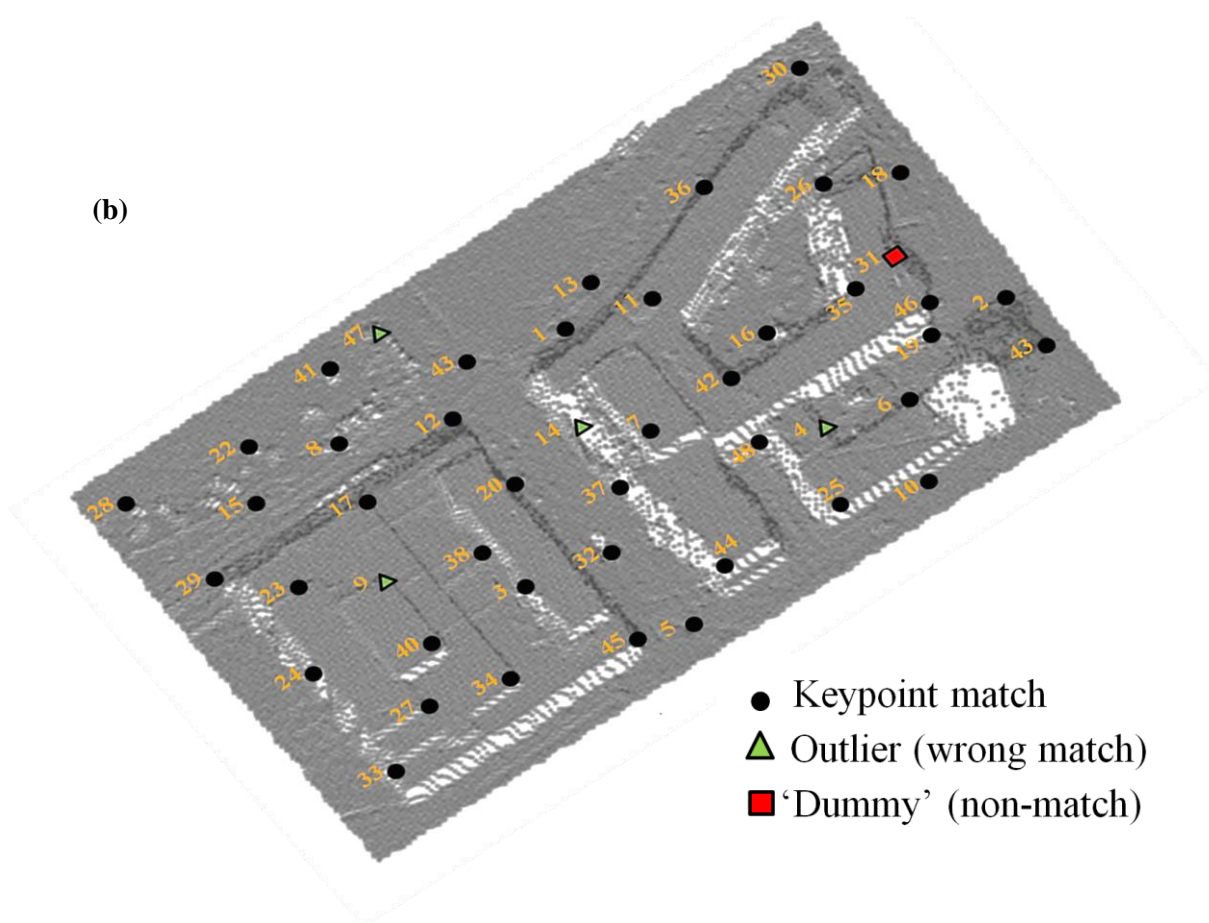
Keypoints were extracted on the source and target datasets, their descriptors were generated and initial point correspondences along with ‘dummy’ matches were found. The dummy matches were those keypoints that have no existing point correspondence, and which were automatically determined by the bipartite graph matching. Outlying matches were then automatically identified using the approach outlined in Algorithm 3.1 (Chapter 3, Section 3.3.4). Figure 5.4 illustrates the final set of keypoint matching results. In Figure 5.4, keypoints with circles and identical numbers indicate inlying matches. Keypoints with the same numbers and triangles are the detected outliers. Keypoints with the squares are the dummy matches.

In Table 5.2, the source descriptors \mathcal{H}_s , target descriptors \mathcal{H}_t , and the similarity score $SimCost_{\chi^2}(\mathcal{H}_s, \mathcal{H}_t)$ are shown for various keypoint correspondences on the source and target point clouds of Figure 5.4. Rows (a) and (b) of Table 5.2 show the results for two inlying matches, i.e., keypoints with IDs 16 and 43 on both the source and target datasets. Row (c) of Table 5.2 is an outlier match (keypoint ID 9). On visual inspection of Figure 5.4(a) and (b), keypoint ID 9 on the source and target are non-corresponding, different keypoint locations. From another visual check, the true match should be keypoint ID 4 on the source dataset and keypoint ID 9 on the target dataset (i.e., row (d) of Table 5.2). The smaller the value of $SimCost_{\chi^2}(\mathcal{H}_s, \mathcal{H}_t)$, the greater the similarity between a source and target keypoint. From Table 5.2, the incorrect match is due to the similarity score of row (d) being larger than the score of row (c). Keypoints 4 and 9 on the source point cloud

(a)



(b)



- Keypoint match
- ▲ Outlier (wrong match)
- 'Dummy' (non-match)

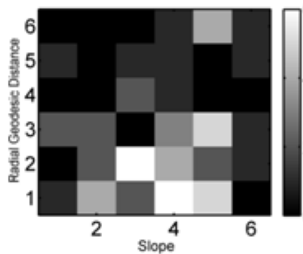
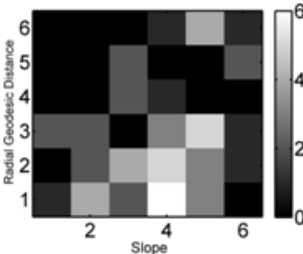
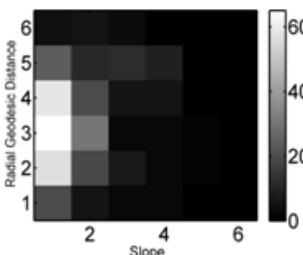
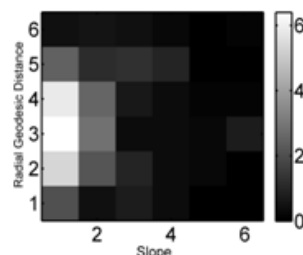
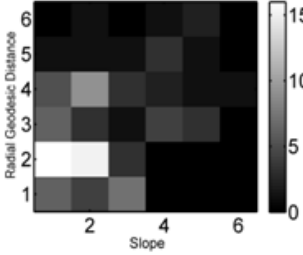
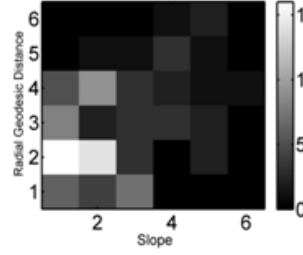
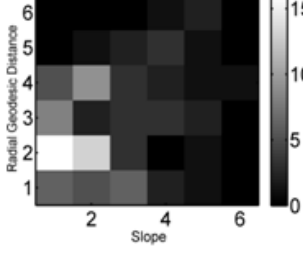
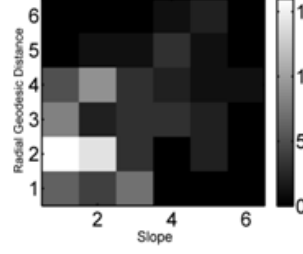
Figure 5.4: Keypoint matching under scaling, rotation and translation. Same number IDs on both the source and target datasets indicate keypoint correspondences (a) Original point clouds (source dataset), (b) Scale, rotated and translated point clouds (target dataset). (Note: surface points for (a) and (b) are: i) illustrated in planar-like views for visualization purposes, and ii) shown in their individual coordinate systems).

dataset were located on similar structures, thereby resulting in similar descriptors and matching ambiguity.

Table 5.3 illustrates the results of transformation parameters obtained using the proposed approach versus the reference parameters. Additionally, the difference between the reference and proposed parameters (Δ_{param}), the precision of the proposed parameters ($\sigma_{Proposed}$) and the root mean square error (RMSE) of least squares adjustment residuals in the X, Y and Z directions are also reported in Table 5.3. These least squares adjustment statistics are derived from the computation of the 3D conformal transformation parameters using inlying keypoint matches. After the outlier removal algorithm (Algorithm 3.1, Chapter 3, Section 3.3.4) is applied, 4 false point matches were removed and 42 valid ones were retained for this urban scene.

A similar ‘controlled’ co-registration experiment was carried out for the non-urban, Icefield scene. The area labeled as ‘Test site for Case 1’ in Figure 5.2(b) was used as the source point cloud dataset. Manually defined ‘reference’ scale, rotation and translation values were applied to generate a target point cloud dataset (see Table 5.4). The site chosen on the Columbia Icefield is the Dome Glacier with coverage of 21.32km². Co-

Table 5.2: Descriptor matching for various keypoints on Figure 5.4.

	\mathcal{H}_s	\mathcal{H}_t	$SimCost_{\chi^2}(\mathcal{H}_s, \mathcal{H}_t)$
(a)	 <p>Keypoint ID: <i>16</i></p>	 <p>Keypoint ID: <i>16</i></p>	0.138
(b)	 <p>Keypoint ID: <i>43</i></p>	 <p>Keypoint ID: <i>43</i></p>	0.097
(c)	 <p>Keypoint ID: <i>9</i></p>	 <p>Keypoint ID: <i>9</i></p>	0.114
(d)	 <p>Keypoint ID: <i>4</i></p>	 <p>Keypoint ID: <i>9</i></p>	0.123

registration results of the Icefield are shown in Table 5.4, and for this dataset there were 2 false keypoint matches and 79 correct keypoint matches. For the ‘Case 1’ urban and non-

Table 5.3: Co-registration result for ‘Case 1’ Urban dataset.

Transformation Parameter	Reference	Proposed Approach	$\sigma_{Proposed Approach}$	Δ_{param}
s	0.7	0.6893	0.051	0.0107
ω (°)	15	14.93	0.137	0.0700
φ (°)	30	29.93	0.087	0.0700
κ (°)	45	44.85	0.045	0.1500
Tx (m)	3	3.030	0.030	-0.0300
Ty (m)	5	5.010	0.074	-0.0100
Tz (m)	7	7.020	0.049	-0.0200
RMSE _x (m)	-	0.013	-	-
RMSE _y (m)	-	0.147	-	-
RMSE _z (m)	-	0.052	-	-

Table 5.4: Co-registration result for ‘Case 1’ Icefield (Non-Urban) dataset.

Transformation Parameter	Reference	Proposed Approach	$\sigma_{Proposed Approach}$	Δ_{param}
s	0.85	0.8514	0.019	-0.0014
ω (°)	6	5.923	0.093	0.0770
φ (°)	12	12.15	0.055	-0.1500
κ (°)	18	18.14	0.068	-0.1400
Tx (m)	9	9.011	0.009	-0.0110
Ty (m)	18	17.89	0.005	0.1100
Tz (m)	27	26.87	0.010	0.1300
RMSE _x (m)	-	1.539	-	-
RMSE _y (m)	-	1.963	-	-
RMSE _z (m)	-	1.746	-	-

urban datasets, the Δ_{param} changes are equivalent to an absolute mean alignment difference of $0.23(\pm 0.05)\text{m}$ and $2.81(\pm 0.16)\text{m}$ respectively. The absolute mean rotational error (AMRE) (i.e., average value of the absolute differences between the automatically-derived and reference angular parameters), as well as the absolute mean translation error (AMTE) (i.e., average value of the absolute differences between the automatically-derived and reference translation parameters) for each dataset is given in Table 5.5. The

urban scene had minimum and maximum residuals of -0.87m and 1.05m. The non-urban scene had minimum and maximum residuals of -4.77m and 4.15m.

Table 5.5: Average Angular and Translation errors for ‘Case 1’ datasets.

Error Measure	Urban Co-registration	Glacier Co-registration
AMRE (°)	0.097	0.122
AMTE (m)	0.020	0.084

5.1.3 Case 2: Different sensor datasets, different coordinate systems

Compared to the ‘Case 1’ tests, this section utilizes multi-sensor datasets, which introduce new challenges to the co-registration process. These include matching points between source and target point clouds which i) have been generated from different sensor data sources, ii) have partial overlap, as a result of less coverage in case of the urban scene or as a result of deformation in the glacial regions of the icefield, iii) have been generated using multi-temporal datasets, iv) have been geo-referenced using different ground control points during the DSM generation process (causing mis-registration errors and requiring a refined alignment), and v) have different point density. To assess the developed co-registration method on the multi-sensor datasets, the respective regions labeled as ‘Test site for Case 2’ on Figures 5.1 and 5.2 were used. That is, the aerial photo point clouds are matched with the aerial LIDAR point clouds for the urban scene. Likewise, the aerial photo point clouds are matched with the WorldView-2

point clouds for the Icefield scene. The observations used in the least squares adjustment were of equal weights assuming similar accuracies.

The urban scene contains buildings, trees, shrubs and bare terrain. The aerial photo urban test site has an area of $56,416\text{m}^2$, whilst the urban aerial LIDAR data covers $152,460\text{m}^2$. The urban aerial photo dataset has a point spacing of 1m and the airborne LIDAR has a point spacing of 0.78m. The non-urban test site is the Saskatchewan Glacier located on the Columbia Icefield. Both of the point cloud datasets to be co-registered have an equivalent point spacing of 1m. The Saskatchewan glacier has an area of 53.55km^2 and comprises of the glacial ice cap in addition to surrounding snowy mountainous regions. Given the 60-year time lapse between the aerial photo and WorldView-2 data collection periods, deformation has occurred on the icefield. The glacier cap has been subjected to severe ice ablation over time where some parts of the upper mountains are snow accumulation areas. This dataset highlights the importance of co-registration for possible change detection applications.

The source and target point clouds of the urban and non-urban multi-sensor datasets were already pre-processed by the data providers and referenced in the same coordinate system. Therefore, to validate the approach, significant transformation parameters were applied for scale, rotation and translation. These serve as the 'reference' parameters. Tables 5.6 and 5.7 show the reference parameters in comparison to those estimated via the proposed automated method.

The automated keypoint matching resulted for the urban dataset resulted in 9 false point correspondences, which were filtered via the outlier removal algorithm, as well as

Table 5.6: Co-registration result for ‘Case 2’ Urban dataset.

Transformation Parameter	Reference	Proposed Approach	$\sigma_{Proposed Approach}$	Δ_{param}
s	0.5	0.4986	4.0e-11	0.0014
ω (°)	13	13.76	0.003	-0.7600
φ (°)	17	18.51	0.012	-1.5100
κ (°)	21	21.28	0.009	-0.2800
Tx (m)	200	200.01	0.013	-0.0100
Ty (m)	400	400.03	0.020	-0.0300
Tz (m)	600	600.00	0.007	0.0000
RMSE _x (m)	-	0.515	-	-
RMSE _y (m)	-	0.820	-	-
RMSE _z (m)	-	0.682	-	-

Table 5.7: Co-registration result for ‘Case 2’ Icefield (Non-Urban) dataset.

Transformation Parameter	Reference	Proposed Approach	$\sigma_{Proposed Approach}$	Δ_{param}
s	0.6	0.5998	2.3e-09	0.0002
ω (°)	30	30.12	6.3e-04	-0.1200
φ (°)	45	45.09	0.031	-0.0900
κ (°)	60	59.99	9.6e-03	0.0100
Tx (m)	1100	1100.01	4.4e-04	-0.0100
Ty (m)	1500	1500.02	3.7e-04	-0.0200
Tz (m)	1900	1899.99	0.001	-0.0100
RMSE _x (m)	-	0.902	-	-
RMSE _y (m)	-	0.934	-	-
RMSE _z (m)	-	0.232	-	-

72 inlying, correct point correspondences. The inlying matches were used to compute the final transformation parameters. For the Saskatchewan glacier dataset, 11 false correspondences were eliminated by the outlier removal algorithm and 141 correct correspondences were used to compute the automated parameters. For the ‘Case 2’ urban and non-urban datasets, the Δ_{param} changes are equivalent to an absolute mean alignment difference of 1.35(\pm 0.29)m and 1.88(\pm 0.91)m respectively. Table 5.8

illustrates the AMRE and AMTE errors relative to the reference parameters for each ‘Case 2’ dataset.

Relative to the coverage of the study areas, there is a dense network of keypoints (i.e., approximately 1 point per $28 \times 28 \text{m}^2$ for the urban dataset and 3 points per 1km^2 for the non-urban dataset). Therefore, the degrees of freedom are large resulting in estimating the transformation parameters with high precision (Tables 5.6 and 5.7). The minimum and maximum of the correspondence residuals from the least squares adjustment for the urban scene were -3.05m and 2.11m respectively with mean of 0.78m and standard deviation of 1.19m . The non-urban scene had minimum and maximum residuals of -2.89m and 3.47m with mean of 0.22m and standard deviation of 1.34m . The alignment errors from the proposed 3D co-registration method met the proximity requirements of the data characteristics. Specifically, for the urban dataset with planimetric and vertical positioning accuracies in the range of 0.2 to 0.5m , the 3D approach obtained errors in the range of 0.5 to 0.8m . For the non-urban data with a positioning accuracies in the range 2.0 to 5.0m , the 3D approach obtained errors in the range of 0.2 to 0.9m .

Figures 5.5 and 5.6 show the co-registration results produced by the developed 3D-based alignment method for the urban and glacier scenes respectively. Noticeably, Figure 5.6(c) (dashed lines) shows an area of significant ice loss on the glacier after automated alignment. Figure 5.6(c) is visualized from a side-view for illustration of alignment of the glacier. Source and target point clouds in Figures 5.5 and 5.6 are shown at 1:1 scaling in their individual coordinate systems and as triangulated meshes for visualization purposes.

It is important to note that refinement-based algorithms such as the ICP can now be applied to possibly improve the co-registration results and overall accuracy statistics.

Figure 5.7 illustrates the alignment differences (i.e., displacement) between the source and target datasets for the ‘Case 2’ urban and non-urban scenes respectively. In Figure 5.7 (b), the maximum distances of approximately 184m are due to the changes of the glacier (red and green areas), while the majority of displacements for the rigid portions were several meters (blue areas). ‘Non-rigid’ refinement algorithms (e.g., Li et al., 2008) can be applied to morph (or warp) the deformed regions for full alignment.

Table 5.8: Average Angular and Translation errors for ‘Case 2’ datasets.

Error Measure	Urban Co-registration	Glacier Co-registration
AMRE (°)	0.850	0.073
AMTE (m)	0.013	0.013

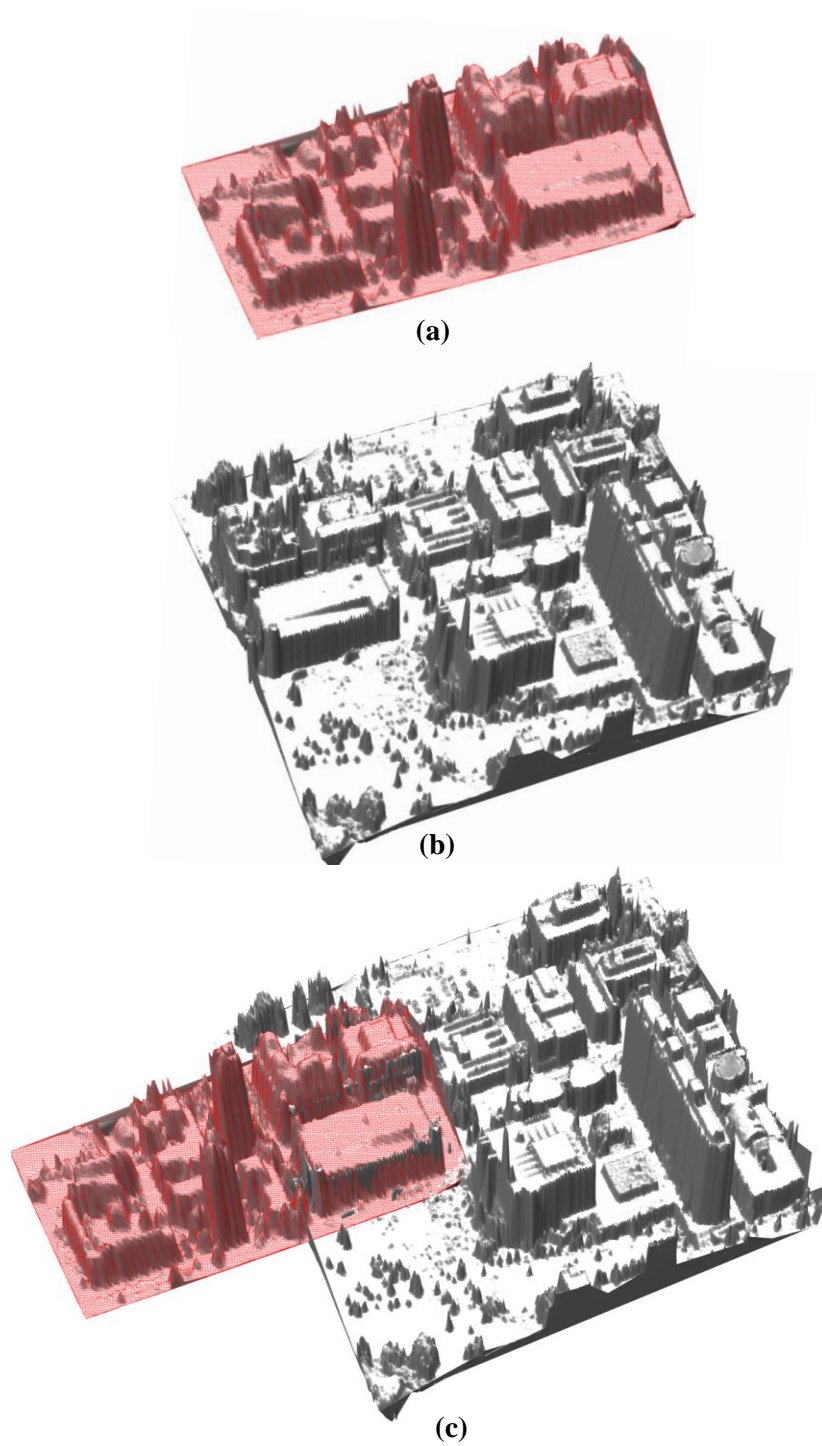


Figure 5.5: Alignment of urban test scene (*Urban, Loc2*). (a) 2005 Aerial photo point cloud surface, (b) 2009 Airborne LIDAR point cloud surface, (c) Co-registration result.

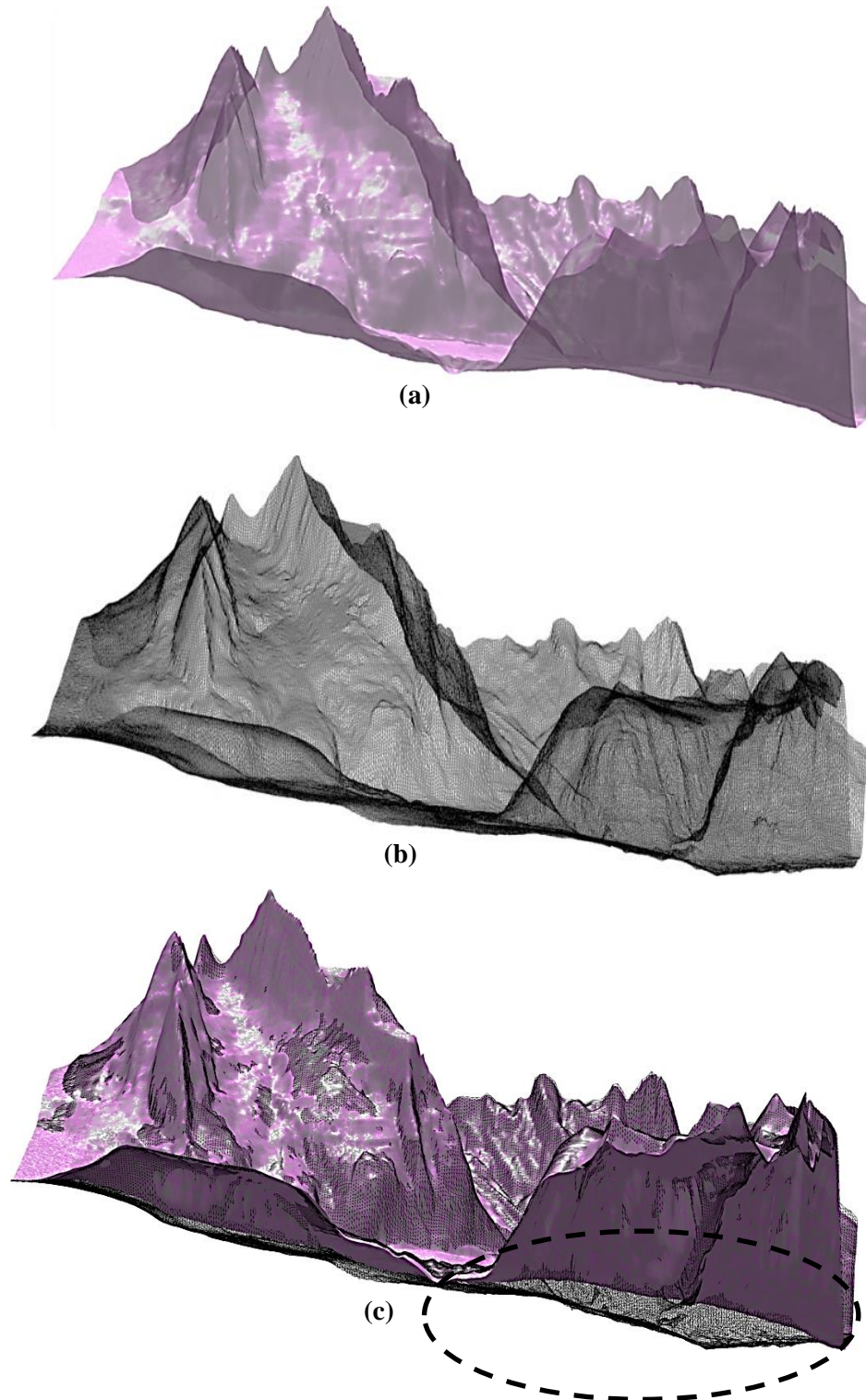


Figure 5.6: Alignment of Saskatchewan Glacier test site (*Non-Urban, Loc3*). (a) 1950 Aerial photo point cloud surface, (b) 2010 WorldView-2 point cloud surface, (c) Co-registration result (dotted line shows region of significant ice loss on glacier).

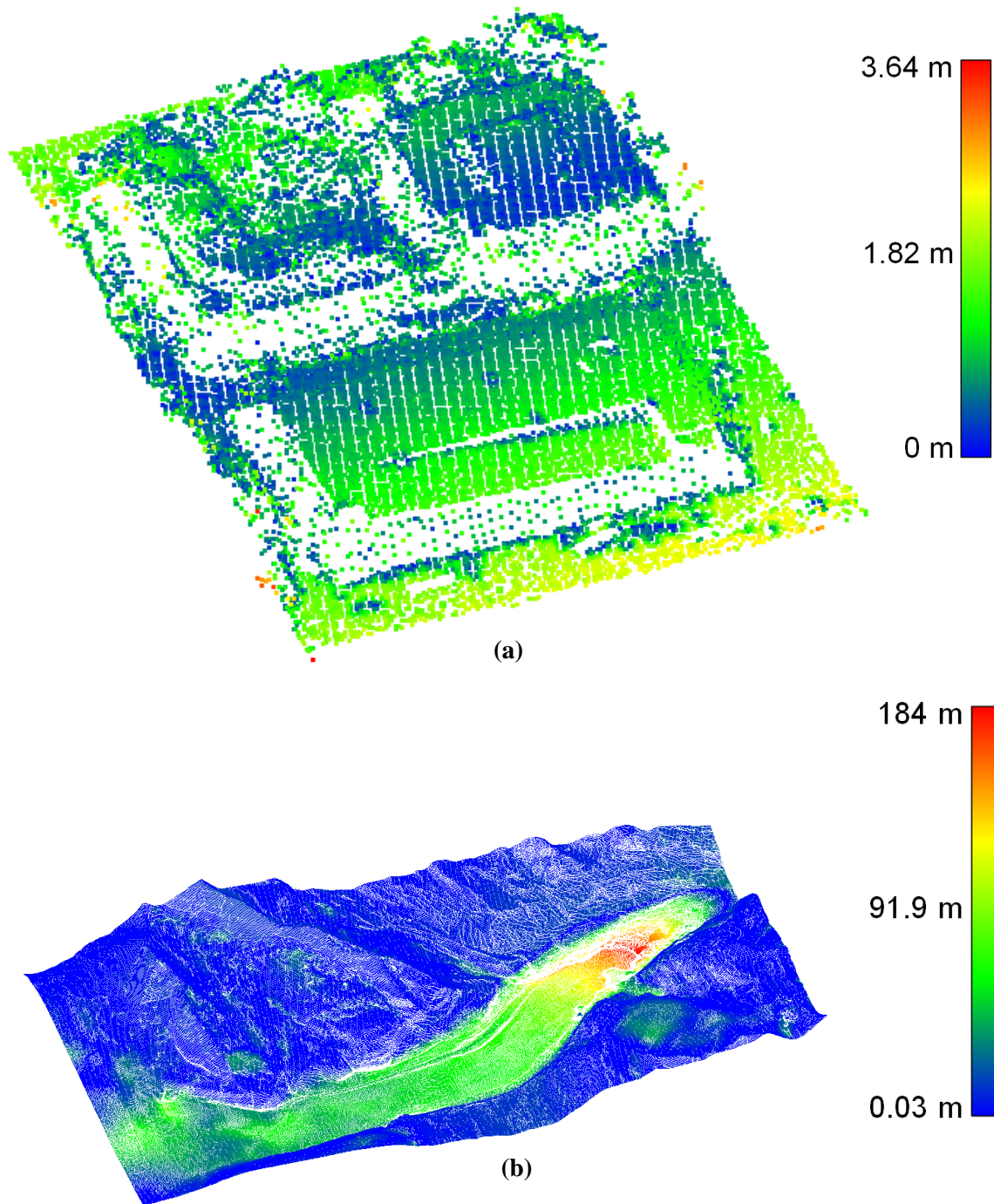


Figure 5.7: Alignment differences between source and target point clouds for ‘Case 2’ datasets. (a) Urban test scene, (b) Non-Urban scene.

5.2 Results for Method 2: Height Map-based Point Cloud Alignment

In this section, results from the second proposed matching framework developed in Chapter 4 are presented, i.e., the height map-based point cloud co-registration. Various experimental results are used to evaluate the proposed multi-scale keypoint detector and the GLP-RT descriptor for height map point matching and 3D point cloud co-registration. In the first experiment, the performance of the 2D keypoint correspondence framework is compared with existing 2D keypoint detection and descriptor methods including SURF (Bay et al., 2008) and SIFT (Lowe, 2004). The second experiment assesses the quality of the automatically estimated 3D conformal transformation parameters for source to target point cloud co-registration. This is done by comparing against known, reference transformation parameters. However, prior to any experiments, the evaluation datasets are introduced and empirical tuning is performed to determine the optimal parameters for the GLP-RT descriptor.

5.2.1 Experimental datasets

To demonstrate the capability of this matching and co-registration framework, various urban (*Loc1* and *Loc2*) and non-urban (*Loc3*), multi-sensor 3D point clouds were used. Point cloud pairs used for matching: i) have different point distributions (e.g., the source point clouds can be uniformly distributed while the target point clouds have non-uniform distribution), ii) have different overlapping coverage, iii) have varying point densities

between them, and iv) are in different coordinate systems (i.e., source and target point clouds to be matched differ by a 3D conformal transformation). Three different datasets (i.e., Figures 5.8, 5.9 and 5.10) are used for experimental analysis. Prior to keypoint extraction, descriptor generation and matching, point clouds are converted to 2D height map images using inverse distance weighting interpolation (Childs, 2004).

5.2.1.1 Dataset 1 (*Urban, Loc1*)

The first dataset (Figure 5.8) includes non-uniform point clouds generated from: i) aerial images collected by a UAV platform, ii) a mobile laser scanner, and iii) a terrestrial laser scanner. The study area is located in Toronto, Ontario, Canada (*Loc1*). This test site comprises of a single building surrounded by vegetation, bare land, paved roadways and a parking lot. Vertical (nadir-looking) images (6000 x 4000 resolution) were acquired from a 19mm Sony Nex-7 camera mounted on a Geo-X8000 UAV. Afterwards, 657,829 points (Figure 5.8(a)) were generated by structure from motion using the Agisoft Photoscan (Agisoft, 2016) photogrammetric software. Mobile laser scanning (MLS) point clouds (75,105,924 points) were also acquired from Optech's Lynx mobile mapping vehicle (Figure 5.8(b)) and an Optech ILRIS long range terrestrial laser scanner collected 57,338,771 points (Figure 5.8(c)). The UAV-based point clouds are generated in a non-georeferenced, local image coordinate system, whilst both the mobile laser and terrestrial point clouds are georeferenced.

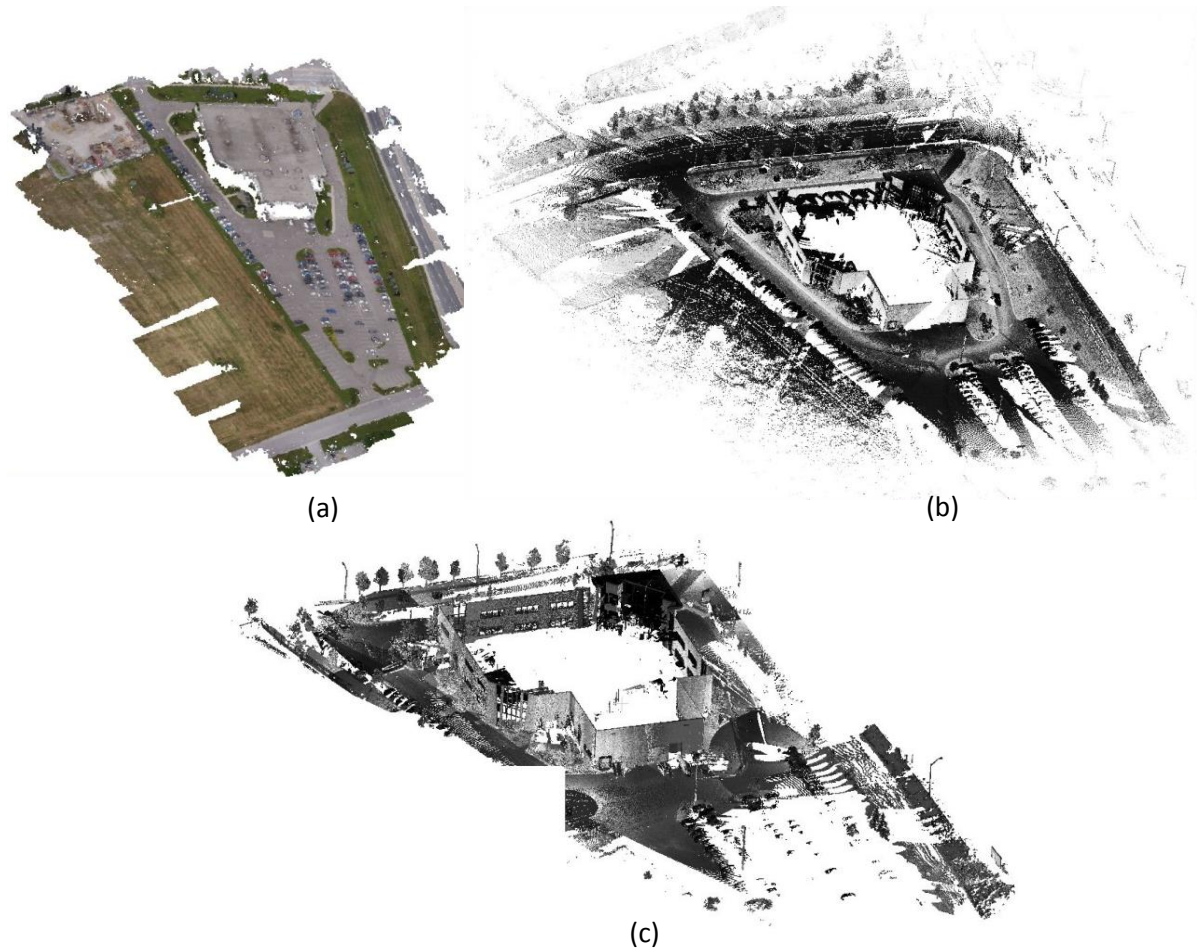


Figure 5.8: Dataset 1 (*Urban, Loc1*) used to evaluate the proposed height map-based point cloud alignment method. (a) UAV-based point clouds (points visualized with RGB texture). (b) Mobile laser scanning (MLS) point clouds. (c) Terrestrial laser scanning (TLS) point clouds.

5.2.1.2 Dataset 2 (*Urban, Loc2*)

The second dataset (Figure 5.9) comprises of point clouds derived from i) aerial images acquired from a UAV platform, ii) aerial photos collected from a manned aircraft and iii) an airborne laser scanner. This test site is located at York University, Toronto, Ontario, Canada (*Loc2*). This dataset is mainly populated with buildings, pedestrian walkways

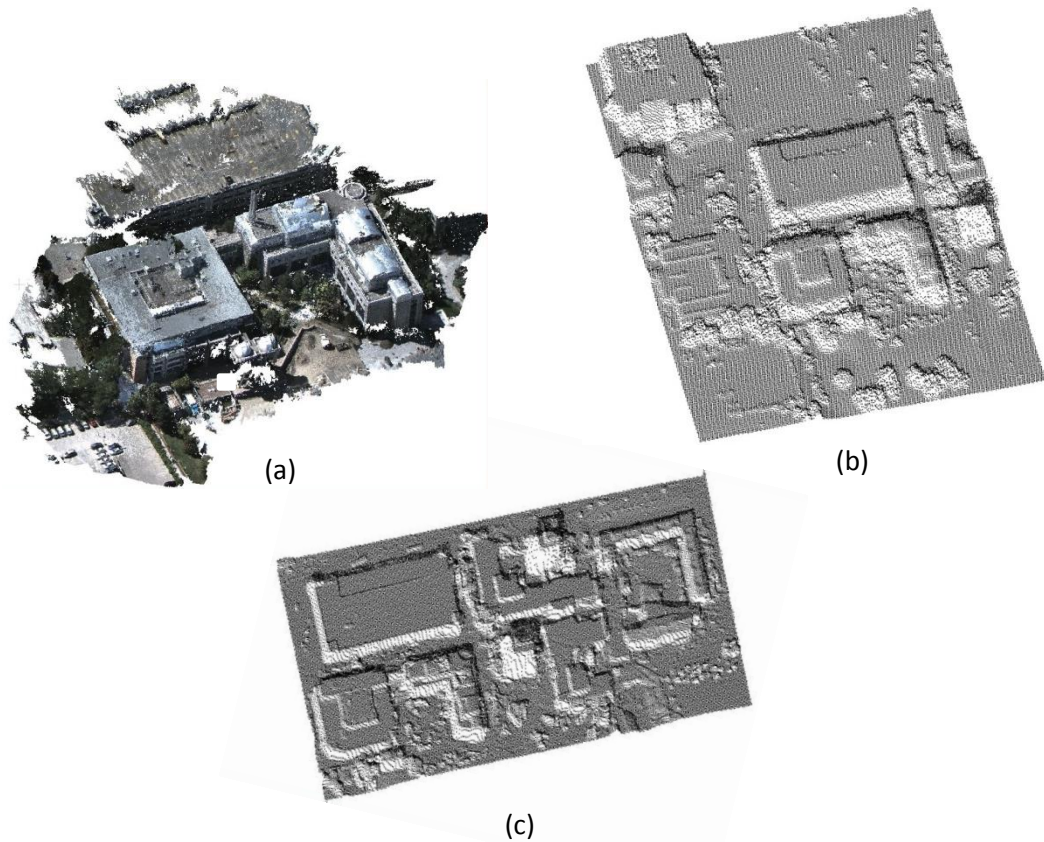


Figure 5.9: Dataset 2 (*Urban, Loc2*) used to evaluate the proposed height map-based point cloud alignment method. (a) UAV-based point clouds (points visualized with RGB texture). (b) Airborne laser scanning (ALS) point clouds. (c) Photogrammetric point clouds from nadir-looking aerial images.

and vegetation. The non-uniform point clouds (7,144,275 points) in Figure 5.9(a) were generated using a combination of oblique and nadir-looking video images (640 x 480 resolution) captured from a Photo3S camera on-board an Aeryon Scout UAV. Agisoft Photoscan was used to generate the point clouds. The 57,911 points in Figure 5.9(b) were generated from nadir-looking, vertical, aerial digital images captured at 0.15m digital resolution and provided by First Base Solutions. The 76,226 points in Figure 5.9(c) were obtained from an Optech airborne LIDAR system (ALS) flown from an altitude of

2300m and 0.78m grid spacing. Similar to Dataset 1, the UAV point clouds are in a non-georeferenced local image coordinate system, whereas the other two datasets are georeferenced.

5.2.1.3 Dataset 3 (*Non-Urban, Loc3*)

In comparison to the first two datasets, this third dataset is non-urban. The study site is the Columbia Icefield situated in Western Canada (*Loc3*). The icefield comprises snowy

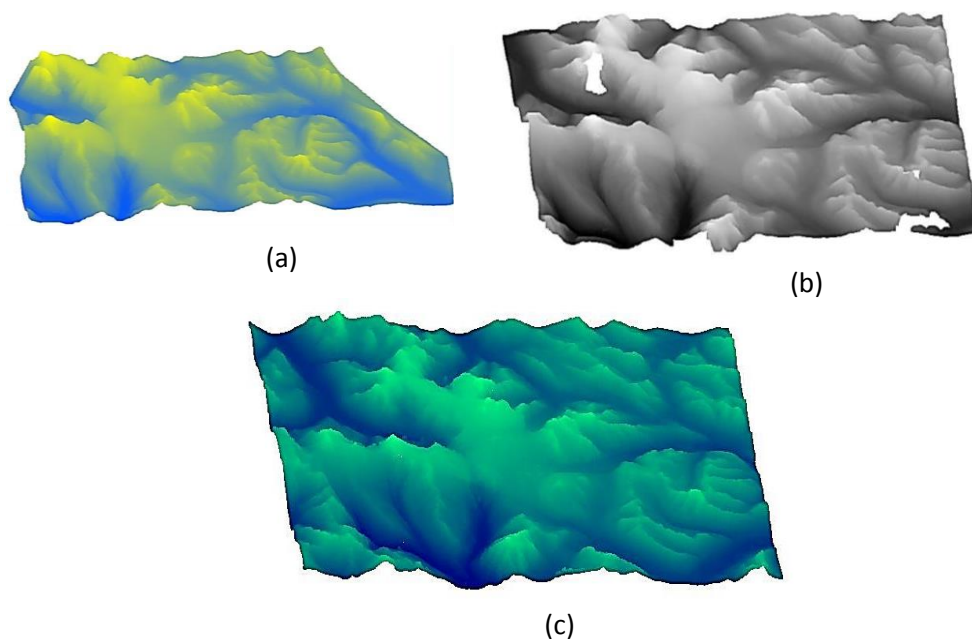


Figure 5.10: Dataset 3 (*Non-Urban, Loc3*), the Columbia Icefield, used to evaluate the proposed height map-based point cloud alignment method. (Note: elevation-based colour ramps are used here for visualization purposes). (a) Photogrammetric point cloud surface model from aerial photos of the icefield. (b) WorldView-2 point cloud surface model of the icefield. (c) Point cloud surface model from ASTER.

mountainous regions, glaciers and rivers. The dataset consists of three gridded digital surface models of the icefield, which were photogrammetrically generated from imagery data collected by different platforms and at different epochs. Figure 5.10(a) has 5,636,140 points and was generated using aerial photographs collected in 1950. The data in Figure 5.10(b) was generated from 2010 WorldView-2 (WV-2) satellite imagery and contains 6,225,640 points. The data in Figure 5.10(c) was acquired from the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) Global Digital Elevation Model (GDEM) and has 9,484,360 points. The 1950 aerial photo point clouds were referenced in a local coordinate system, whilst the WV-2 and ASTER point clouds were georeferenced.

5.2.1.4 Tuning and testing datasets

The three presented datasets are split into two separate categories: a tuning dataset group and a testing dataset group. The tuning dataset are source and target height map pairs used to empirically select the optimal parameters for the GLP-RT descriptor based on sensitivity analysis performed in the next section. The testing datasets are independent source and target height map pairs not included in the tuning process and are instead used for evaluating the accuracies of the height map point matching and 3D point cloud co-registration. The tuning and testing datasets comprise of source and target height map images which differ by scale and a rotation around the Z-axis. The scale ranges from 0 to 1, for example an applied scale of 0.5 represents downsampling of the original image by one-half in each of the two respective image dimensions. There is only one-directional

rotation since the simulated data are 2D height map images. These images are rotated around an axis perpendicular to the image plane and passing through the image center.

Beside the real datasets, additional tuning and training height map datasets were also generated to increase the sample size for: i) descriptor parameter selection, and ii) evaluating the developed keypoint matching approach in comparison to state-of-the-art methods. The datasets are created by applying a rotation and scale change to a source height map to produce a simulated target height map. Table 5.9 shows the respective source and target combinations which were used for the empirical tuning. Table 5.10 are the datasets used in the testing experiment. In total, there are nine tuning height map pairs (i.e., three from the ‘*real*’ datasets and six from the ‘*simulated*’ datasets) and six testing height map pairs (i.e., three from the ‘*real*’ datasets and three from the ‘*simulated*’ datasets).

5.2.2 Empirical tuning: Selection of GLP-RT descriptor parameters

Empirical tuning for setting the parameter values of feature descriptors has been applied in related works such as Guo et al. (2013) and Huang et al. (2014). The GLP-RT keypoint descriptor has four parameters: i) the minimum radius $minR$ for the log-polar sampling area, ii) the maximum radius $maxR$ for the log-polar sampling area, iii) the number of subdividing rays M for the log-polar grid, and iv) the number of concentric rings N for the log-polar grid. In this section, the impact of the individual descriptor parameters on

the keypoint matching process is investigated. This is done by varying the values of each parameter across heuristically set ranges and evaluating the descriptor on the nine tuning datasets from Table 5.9. The objective is to select the parameter values which yield the best matching performance based on the *recall vs. 1-precision* metric (Ke and Sukthankar, 2004) defined in Equation 5.1 and Equation 5.2.

Table 5.9: Simulated and real source and target datasets which are used for the empirical tuning. (Note: The simulated dataset column includes the rotation and scale values used to generate the respective simulated target height maps).

	Simulated tuning dataset (<i>Source, Target</i>)	Real tuning dataset (<i>Source, Target</i>)
<i>Dataset 1</i>	i) UAV, UAV _{rotation=10°, scale=0.8} ii) TLS, TLS _{rotation=15°, scale=0.75}	UAV, TLS
<i>Dataset 2</i>	i) UAV, UAV _{rotation=20°, scale=0.7} ii) Aerial image, Aerial image _{rotation=25°, scale=0.65}	UAV, Aerial image
<i>Dataset 3</i>	i) Aerial photo, Aerial photo _{rotation=30°, scale=0.6} ii) ASTER, ASTER _{rotation=35°, scale=0.55}	Aerial photo, ASTER

Table 5.10: Simulated and real source and target datasets which are used for the testing experiment. (Note: The simulated dataset column includes the rotation and scale values used to generate the respective simulated target height maps).

	Simulated testing dataset (<i>Source, Target</i>)	Real testing dataset (<i>Source, Target</i>)
<i>Dataset 1</i>	i) MLS, MLS _{rotation=20°, scale=0.7}	UAV, MLS
<i>Dataset 2</i>	i) ALS, ALS _{rotation=30°, scale=0.6}	UAV, ALS
<i>Dataset 3</i>	i) WV-2, WV-2 _{rotation=40°, scale=0.5}	Aerial photo, WV-2

A TP is recorded when two matching keypoints are from the same corresponding positions on the source and target height maps. Similarly, a FP occurs when two matching keypoints are from different positions on the source and target height maps.

The *recall vs. 1-precision* graphs are generated by alternately varying one parameter while keeping the others fixed. For the source and target height maps in each of the nine tuning datasets, keypoints are extracted using the proposed multi-scale detection method, their GLP-RT descriptors are computed and correspondences are found via bi-directional matching. Their combined *recall* and *1-precision* results are illustrated in Figure 5.11 for each GLP-RT parameter. Optimal parameter values are those with high *recall* and low *1-precision* rates.

5.2.2.1 The minimum radius

The *minR* parameter is the radius value of the smallest concentric circle on the log-polar sampling grid. The *minR* should ideally have a value (in pixels) to ensure important features at smaller scales are modelled by the descriptor. The *minR* was tested in following range: 0.4% to 1.8% of the maximum dimension of the height map image (in pixels). Testing is done at 0.2% intervals for this range. The height, width and cross-directional dimension of the height map image are considered when choosing the maximum dimension. On observing Figure 5.11(a), when the *minR* values increase beyond 1.4%, the descriptor's performance degrades since critical structural information existing at finer scales is not captured / sampled by the log-polar grid. Therefore, the

recall vs. 1-precision plots in Figure 5.11(a) indicate that the descriptor performs best at a value of $minR = 1.4\%$ of the maximum height map dimension (in pixels).

5.2.2.2 The maximum radius

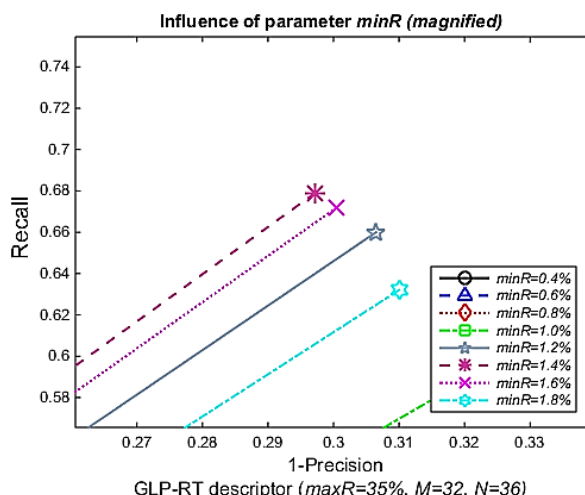
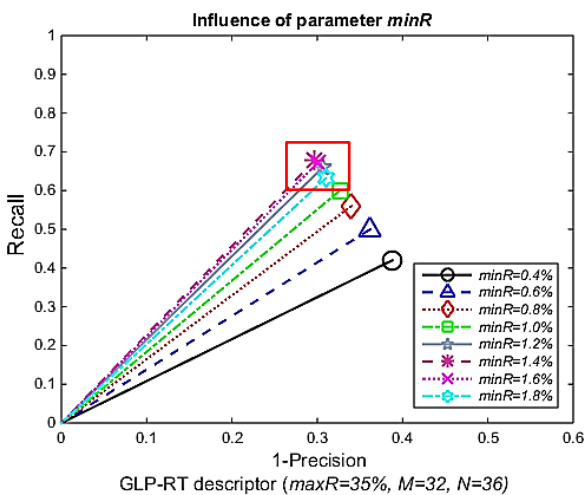
The $maxR$ parameter (in pixels) defines the radius of the outermost circle on the log-polar grid. If its value is too small, the descriptor will have insufficient contextual information to disambiguate keypoints which exist on similar structures, for instance, similarly shaped building corners. On the other hand, when there are keypoints in the vicinity of the height map boundaries, exceedingly large $maxR$ values will go beyond the height map image limits. This will cause the descriptor to include sampled grid points in regions where no useful information exists, thus, distorting the descriptor. The $maxR$ parameter was tested between the ranges of 15% to 50% of the maximum dimension of the height map image (in pixels). Testing was done at 5% intervals for this range. Figure 5.11(b) illustrates that from values 15% to 35%, there is a gradual rise in descriptor performance. As the value increases from 35% to 50% there is degradation in the accuracy. On analysing the point matching results, this is due to the larger $maxR$ values, which cause the majority of the sampled points on the log-polar grid to be outside the height map image, thus reducing descriptor's discriminative ability. Based on Figure 5.11(b), $maxR$ was set to be 35% of the maximum height map dimension (in pixels).

5.2.2.3 The number of rays and number of rings

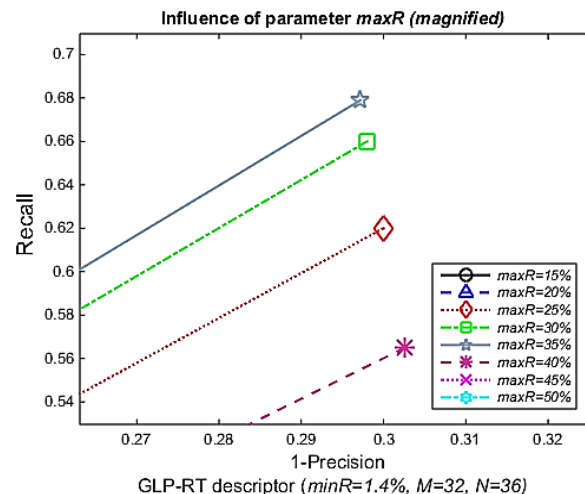
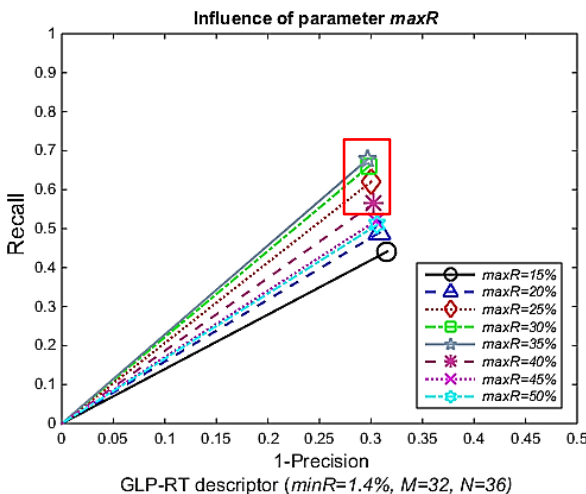
Both the M and N parameters influence the descriptor dimensionality and discriminability (i.e., the descriptor's ability to distinguish one keypoint from another). If the number of rays and rings are too small, the descriptor's discriminative power will be lowered, thus increasing the likelihood of wrong keypoint correspondences. Alternatively, if the number of rays and rings are too large, the descriptor can become 'over-sensitized', i.e., the descriptors will be too unique. This will make it difficult to establish true source to target point matches. In addition to this, there are also increased computations with higher values of M and N . The values of M and N were tested for the range: 20 to 40, over intervals of 4. Observing Figure 5.11(c) and Figure 5.11(d) respectively, the descriptor is most optimal when $M=32$ and $N=36$. Table 5.11 provides a summary of the GLP-RT parameter values found by empirical tuning. These descriptor parameters are used for the remaining experiments conducted in the dissertation.

Table 5.11: Optimal GLP-RT descriptor parameters after tuning.

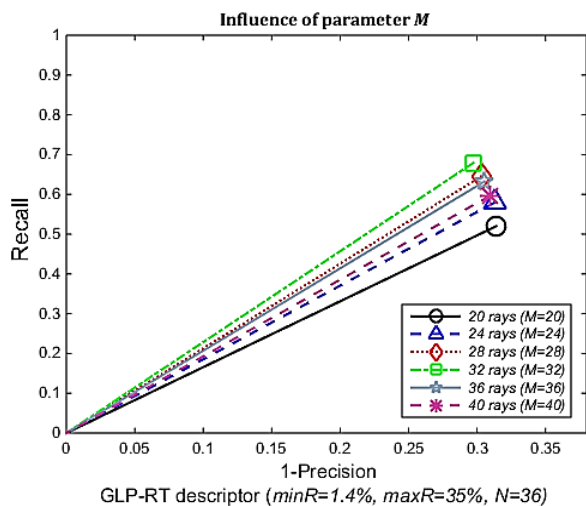
<i>Parameter</i>	<i>Value</i>
Minimum radius $minR$	1.4% (of max. height map dimension)
Maximum radius $maxR$	35% (of max. height map dimension)
Number of rays M	32
Number of rings N	36



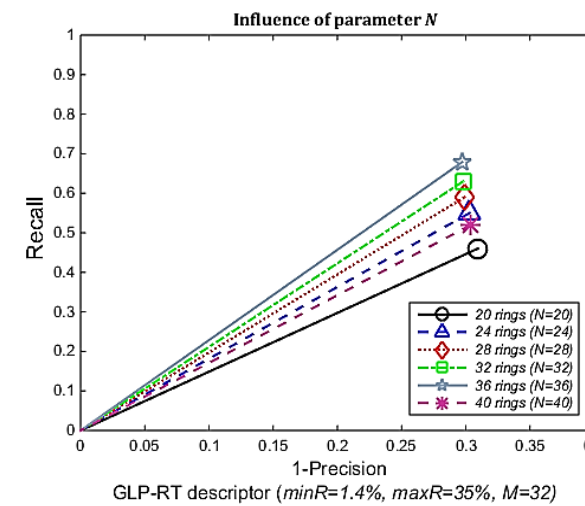
(a)



(b)



(c)



(d)

Figure 5.11: *Recall vs. 1-precision* graphs for selecting optimal GLP-RT descriptor parameters using the tuning datasets. (a) The minimum radius $minR$ (right plot is a magnification of the rectangle in the left plot). (b) The maximum radius $maxR$ (right plot is a magnification of the rectangle in the left plot). (c) The number of rays N . (d) The number of rings M .

5.2.3 Testing experiment: Assessment of the 2D height map approach with other 2D keypoint detectors and descriptors

In this section, the performance of the proposed multi-scale keypoint detector and GLP-RT descriptor are compared to other state-of-the-art 2D keypoint detectors, as well as, with state-of-the-art scale-and rotation-invariant 2D descriptors. This is done using the six simulated and real test height map datasets in Table 5.10. Table 5.12 depicts the various combinations of detectors and descriptors which are evaluated. For the source and target height maps in each of the six test datasets, keypoints are extracted, their descriptors are computed and correspondences are found via bi-directional matching. The *recall vs. 1-precision* criterion is used for evaluation. The plots of Figure 5.12 are the results showing the average recall and 1-precision value of all 6 datasets for each combination. The proposed multi-scale keypoint detection with the GLP-RT descriptor (i.e., Combination 8) outperformed the other combinations. The only instance of inferior GLP-RT performance was in combination with the SURF detector (i.e., Combination 5),

where DSID (i.e., Combination 3) achieved higher matching rates. In terms of keypoint detection methods, it was also observed that the proposed wavelet-based detector has comparable matching accuracies with the SIFT and SURF detectors. However, this occurs when the SIFT and SURF detectors are used in combination with the proposed GLP-RT or the DSID descriptors.

Figure 5.12 reveals a noticeable disparity. The descriptors (i.e., the SIFT and SURF descriptors as used for Combinations 1, 2, 6 and 7) relying on local keypoint scales estimated from the detectors have lower matching accuracies in comparison to those not relying on scales from detectors (i.e., the GLP-RT and DSID descriptors as used for Combinations 3, 4, 5 and 8). On examining the keypoint matching results, it was observed that the local estimated scales from the front-end detectors negatively affected the matching accuracy due to two main factors.

First, the estimated scales provide an insufficient level of local neighbourhood context to ensure descriptor discriminability. That is, at the defined scales, the local descriptor neighbourhoods were too small and did not capture enough local image content. Second, matching between source and target keypoints could not be established because their corresponding local neighbourhoods, as defined by their respective keypoint scales, were not consistent. That is, source and target descriptor neighbourhoods did not contain similar local regions. The lack of similar local scales around keypoints to establish correspondence was associated with differences in texture variation and noise between the source and target multi-sensor height map images, particularly around object boundaries (e.g., building corners). Visual point matching results for the three *real test*

datasets (from Table 5.10) based on the proposed feature matching framework are shown in Figures 5.13, 5.14 and 5.15.

Table 5.12: Combinations of 2D keypoint detectors and 2D descriptors evaluated on the height map testing datasets.

	<i>Detector</i>	<i>Descriptor</i>
<i>Combination 1</i>	Proposed multi-scale approach	SIFT
<i>Combination 2</i>	Proposed multi-scale approach	SURF
<i>Combination 3</i>	Proposed multi-scale approach	DSID
<i>Combination 4</i>	SIFT	Proposed GLP-RT
<i>Combination 5</i>	SURF	Proposed GLP-RT
<i>Combination 6</i>	SIFT	SIFT
<i>Combination 7</i>	SURF	SURF
<i>Combination 8</i>	Proposed multi-scale approach	Proposed GLP-RT

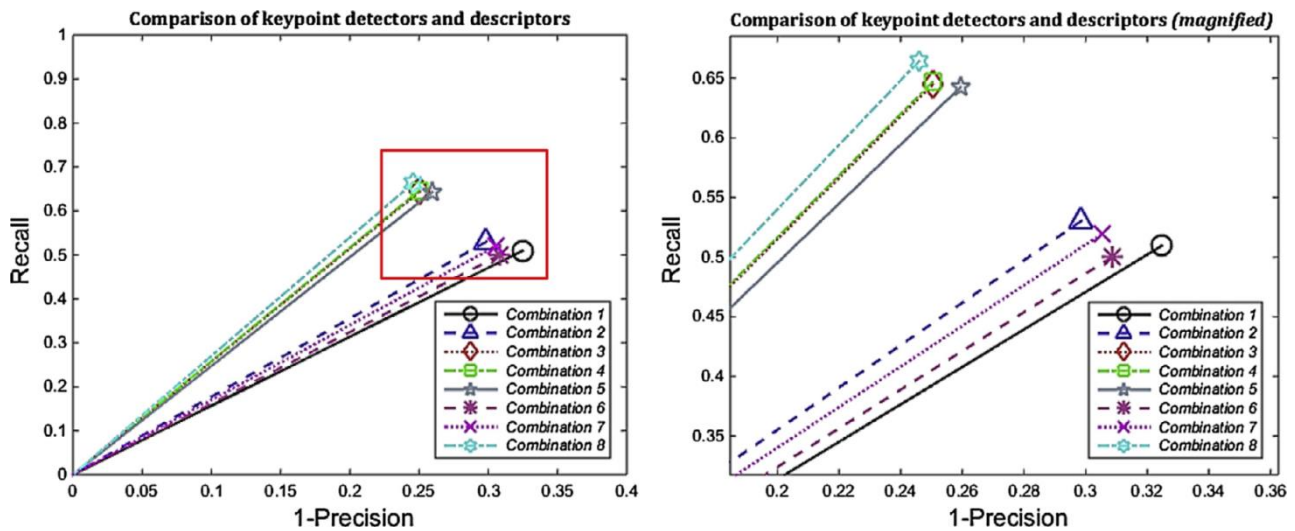


Figure 5.12: Recall vs. 1-precision graphs of the six test datasets using different keypoint detectors/descriptor combinations from Table 5.12 (right plot is a magnification of the square in the left plot).

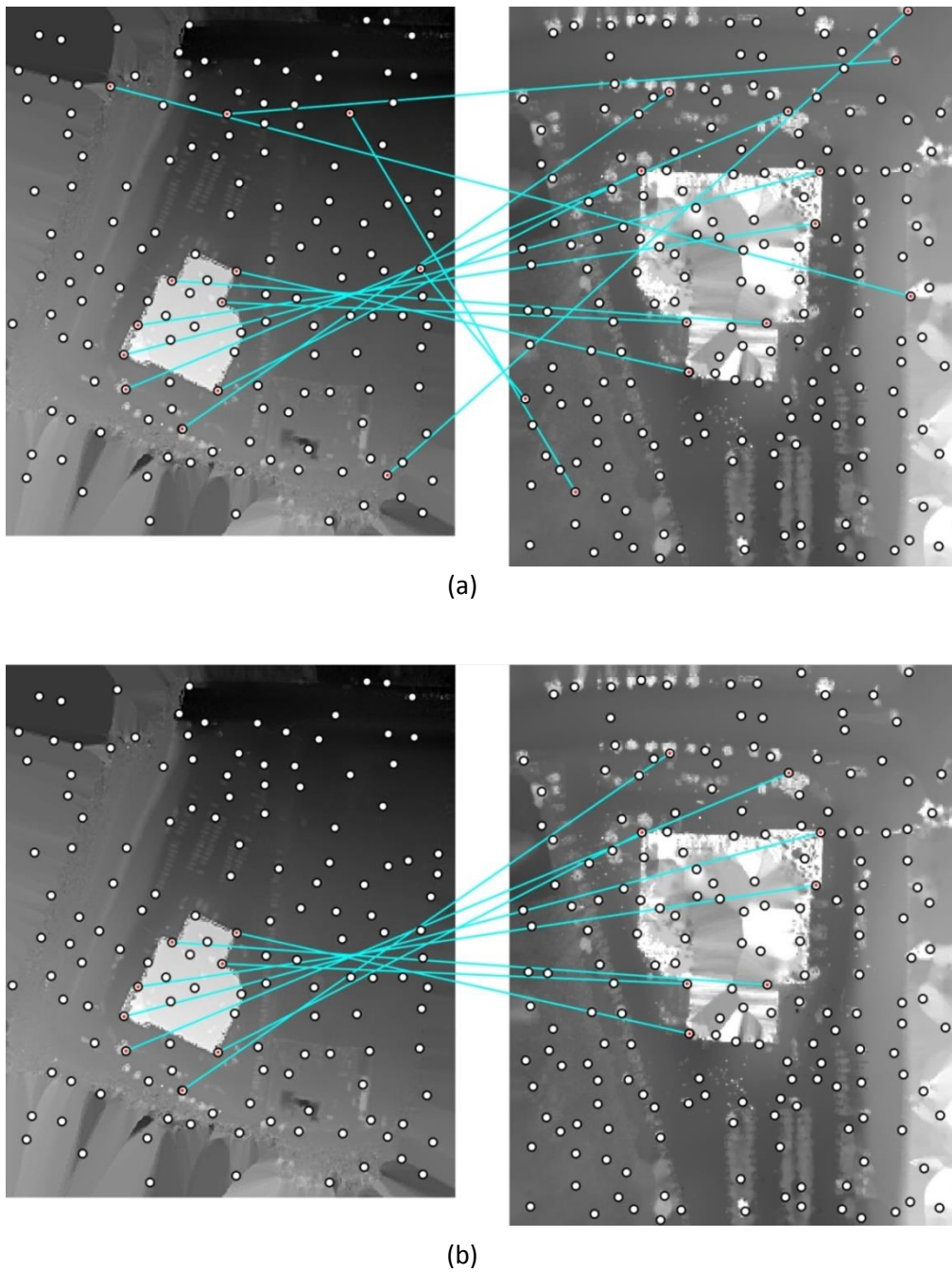


Figure 5.13: Height map point matching results for *real* test dataset 1 using proposed multi-scale keypoint extraction and GLP-RT descriptor (Left: UAV height map, Right: MLS height map). (a) After bi-directional matching. (b) After modified-RANSAC.

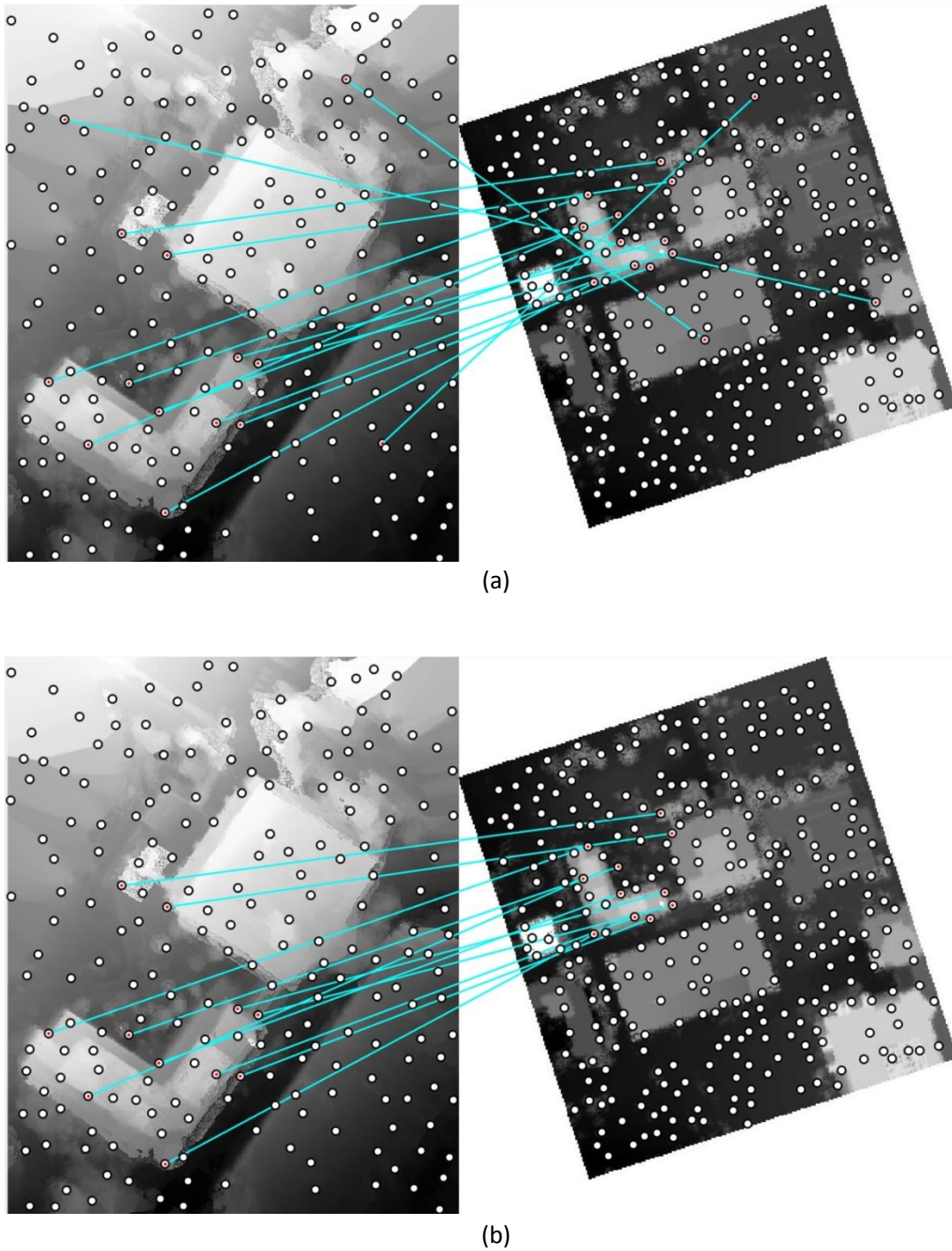


Figure 5.14: Height map point matching results for *real* test dataset 2 using proposed multi-scale keypoint extraction and GLP-RT descriptor (Left: UAV height map, Right: ALS height map). (a) After bi-directional matching. (b) After modified-RANSAC.

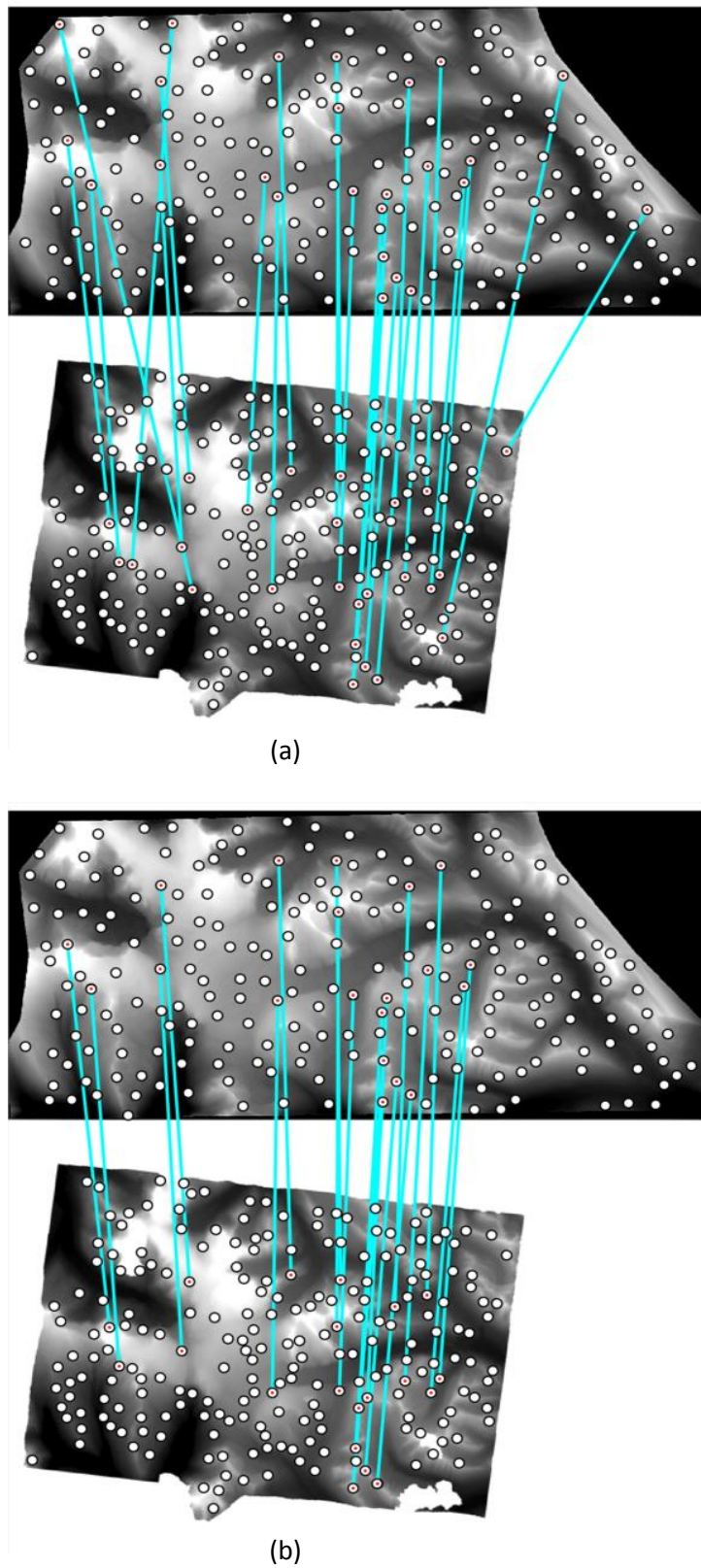


Figure 5.15: Height map point matching results for *real* test dataset 3 using proposed multi-scale keypoint extraction and GLP-RT descriptor (Top: Aerial photo height map,

Bottom: WorldView-2 height map). (a) After bi-directional matching. (b) After modified-RANSAC.

5.2.4 Accuracy analysis of 2D height map-based point cloud co-registration

3D point-cloud co-registration results based on the height map keypoint matching approach for the three *real* test datasets from Table 5.10 are presented and discussed. The accuracy of the scale (s), 3 rotation angles (ω , ϕ , κ) and 3 translation (T_x , T_y , T_z) parameters are assessed in two ways based on the: i) results provided by least squares adjustment statistics and ii) differences between the seven transformation parameters obtained using the proposed automated point matching method versus those from known ‘reference’ parameters. The reference parameters for the real datasets were computed by manually selecting distinct, well distributed corresponding landmark points on both the source and target point clouds (8 correspondences were selected for *real* test datasets 1 and 2, whilst 27 correspondences were selected for *real* test dataset 3). Afterwards, a 3D conformal transformation least squares adjustment is used to obtain the seven parameters. A larger number of corresponding points than the minimum three points required to estimate the 3D transformation was used to ensure redundancy of feature point observations in the non-linear least squares optimization. This non-linear minimization was initialized using Horn’s linear closed-form 3D conformal solution (Horn, 1987). From the least squares adjustment, the precision of the parameters (σ) and the root mean

square error (RMSE) of the least squares adjustment point observation residuals in the X, Y and Z directions were computed. It is noted that: i) when computing the co-registration parameters, the point clouds in the local coordinate systems are set as the source point cloud, whilst the georeferenced point clouds are set as the target point cloud and ii) large georeferenced 'X' and 'Y' coordinate values are shifted to a local system to avoid numerical instabilities during the least squares adjustment. The observations used in the least squares adjustment were of equal weights assuming similar accuracies.

The minimum and maximum residuals for *real* test dataset 1 was -0.62m and 1.07m with mean of 0.04m and standard deviation of 0.53m. For *real* test dataset 2, the maximum and minimum residuals were -0.96m and 1.58m respectively, with mean of 0.12m and standard deviation of 1.10m. For *real* test dataset 3, the maximum and minimum residuals were -4.12m and 3.80m with mean of 0.26m and standard deviation of 1.35m. The alignment errors from the proposed height map co-registration method met the proximity requirements of the data characteristics. Specifically, for the two urban datasets with planimetric and vertical positioning accuracies in the range of 0.2 to 0.5m, the proposed height map approach obtained errors in the range of 0.3 to 0.6m. For the non-urban data with positioning accuracies in the range 2.0 to 5.0m, the height map method obtained errors in the range of 0.4 to 1.0m.

Tables 5.13, 5.14 and 5.15 contain results of the reference parameters in comparison to those estimated using the automated approach. The difference between both are reflected by Δ_{param} . For *real* test datasets 1, 2 and 3, the Δ_{param} changes are equivalent to an absolute mean alignment difference of 0.17(\pm 0.09)m, 0.48(\pm 0.13)m and 1.21(\pm 0.37)m

respectively. On observing the RMSE in the three directional components, the automated method obtained errors of approximately 1m from the three tested datasets. Figures 5.16, 5.17 and 5.18 show the visual co-registration results for each of the three *real* tested datasets.

Table 5.13: Co-registration result for *real* test dataset 1 (*Urban, Loc1*).

Transformation Parameter	Reference	$\sigma_{Reference}$	Proposed Approach	$\sigma_{Proposed Approach}$	Δ_{param}
s	45.17	0.065	45.25	0.043	0.080
ω (°)	39.25	0.026	38.94	0.012	0.310
ϕ (°)	-2.86	0.019	-3.01	0.036	0.150
κ (°)	-34.61	0.032	-34.45	0.005	0.160
Tx (m)	750.29	0.027	750.40	0.040	0.110
Ty (m)	418.75	0.031	418.72	0.009	0.030
Tz (m)	126.11	0.014	125.98	0.006	0.130
RMSE _x (m)	0.267	-	0.190	-	-
RMSE _y (m)	0.458	-	0.243	-	-
RMSE _z (m)	0.509	-	0.414	-	-

Table 5.14: Co-registration result for *real* test dataset 2 (*Urban, Loc2*).

Transformation Parameter	Reference	$\sigma_{Reference}$	Proposed Approach	$\sigma_{Proposed Approach}$	Δ_{param}
s	5.74	0.033	5.73	0.048	0.010
ω (°)	2.76	0.037	2.92	0.022	0.160
ϕ (°)	-15.89	0.027	-16.32	0.031	0.430
κ (°)	4.46	0.035	4.15	0.029	0.310
Tx (m)	170.92	0.049	171.31	0.022	0.390
Ty (m)	796.35	0.047	795.89	0.018	0.460
Tz (m)	162.57	0.059	162.21	0.030	0.360
RMSE _x (m)	0.526	-	0.619	-	-
RMSE _y (m)	1.003	-	0.744	-	-
RMSE _z (m)	0.281	-	0.460	-	-

Table 5.15: Co-registration result for *real* test dataset 3 (*Non-Urban, Loc3*).

Transformation Parameter	Reference	$\sigma_{Reference}$	Proposed Approach	$\sigma_{Proposed Approach}$	Δ_{param}
s	11.65	0.001	11.63	1.4e-03	0.019
ω (°)	-9.10	0.014	-9.09	0.011	0.009
ϕ (°)	7.70	0.006	7.72	0.009	0.020
κ (°)	18.45	0.005	18.44	0.004	0.009
Tx (m)	835.59	0.209	836.72	0.138	1.130
Ty (m)	1184.33	0.183	1183.29	0.151	1.039
Tz (m)	2530.65	0.281	2531.05	0.096	0.400
RMSE _x (m)	0.326	-	0.391	-	-
RMSE _y (m)	0.607	-	0.753	-	-
RMSE _z (m)	0.728	-	1.011	-	-

Figure 5.19 illustrates the visual alignment differences (i.e., displacement) for points which were common on both the source and target real test datasets. For Figure 5.19 c), the maximum distances of approximately 400m were due to the unaligned deformation areas existent on the icefield.

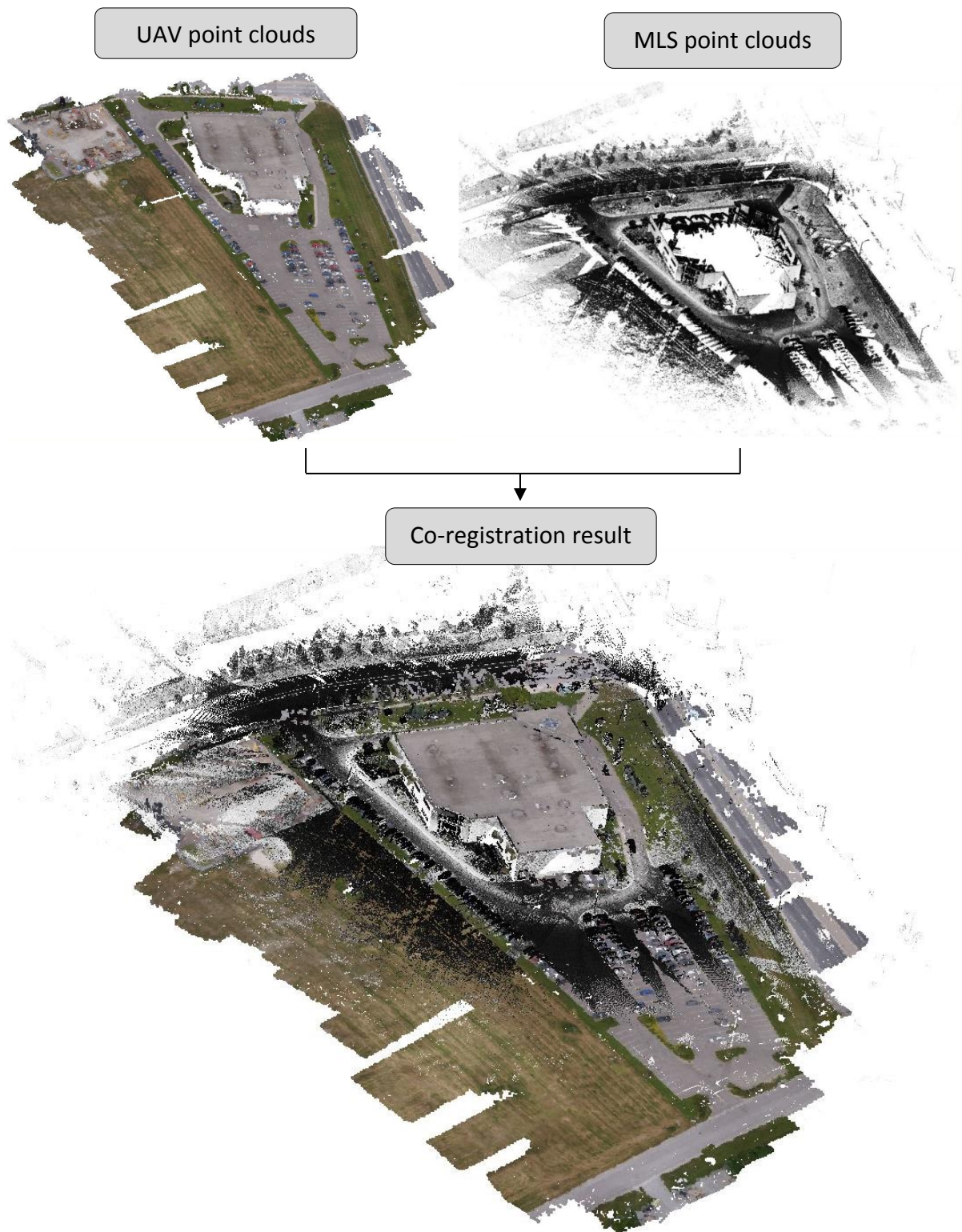


Figure 5.16: Co-registration of point clouds for *real* test dataset 1(*Urban, Loc1*).

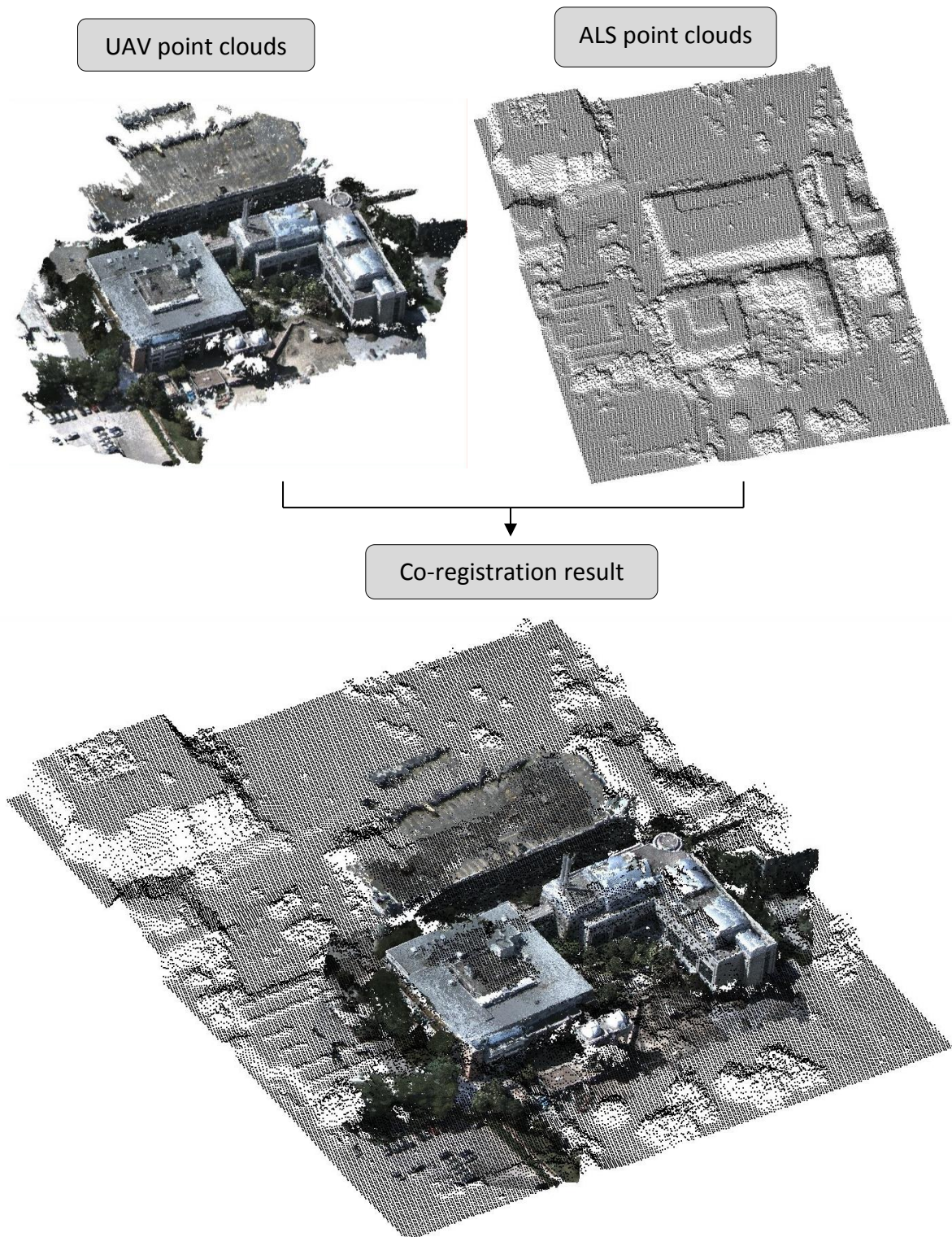


Figure 5.17: Co-registration of point clouds for *real* test dataset 2 (*Urban, Loc2*).

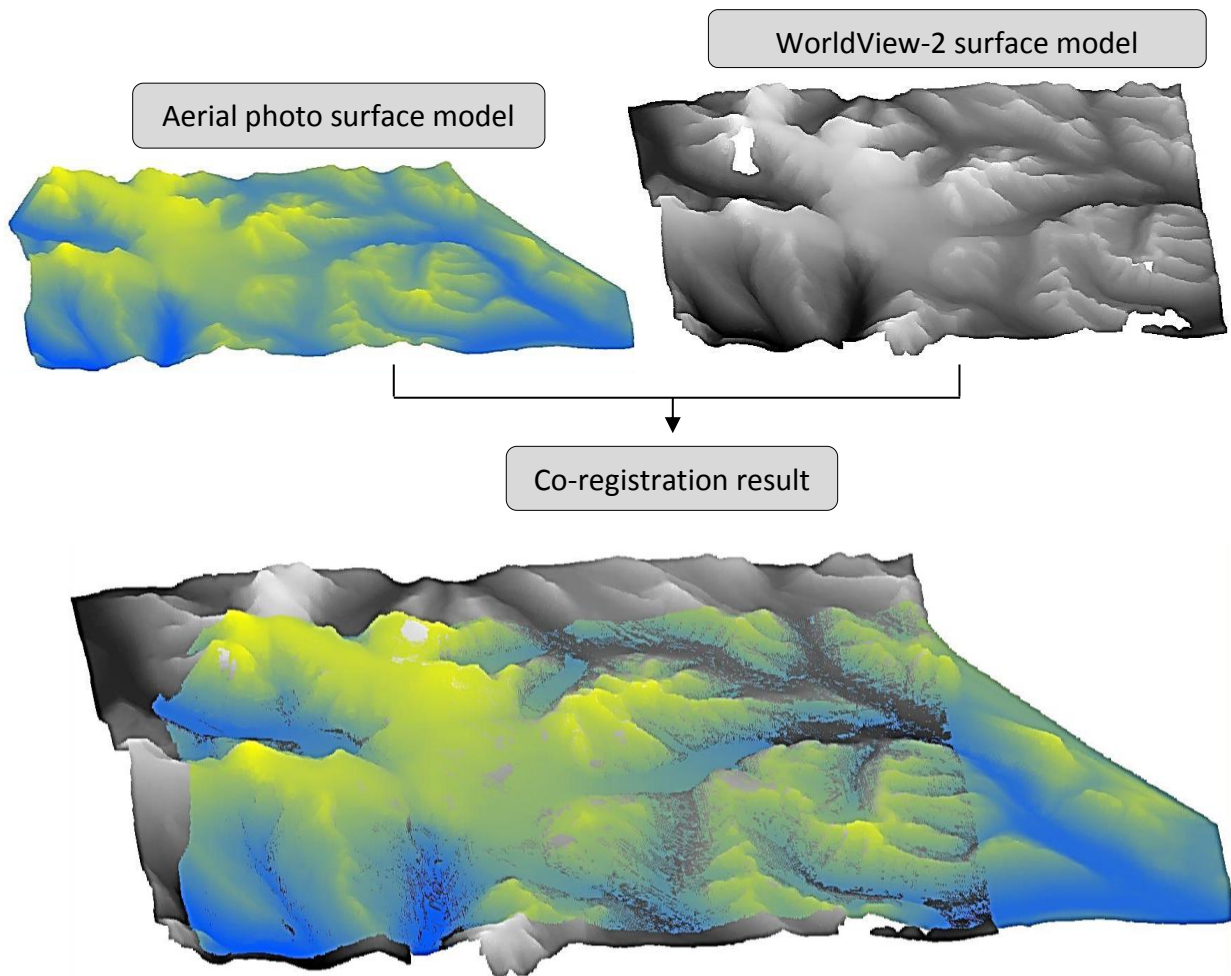


Figure 5.18: Co-registration of point clouds for *real* test dataset 3 (*Non-Urban, Loc3*).

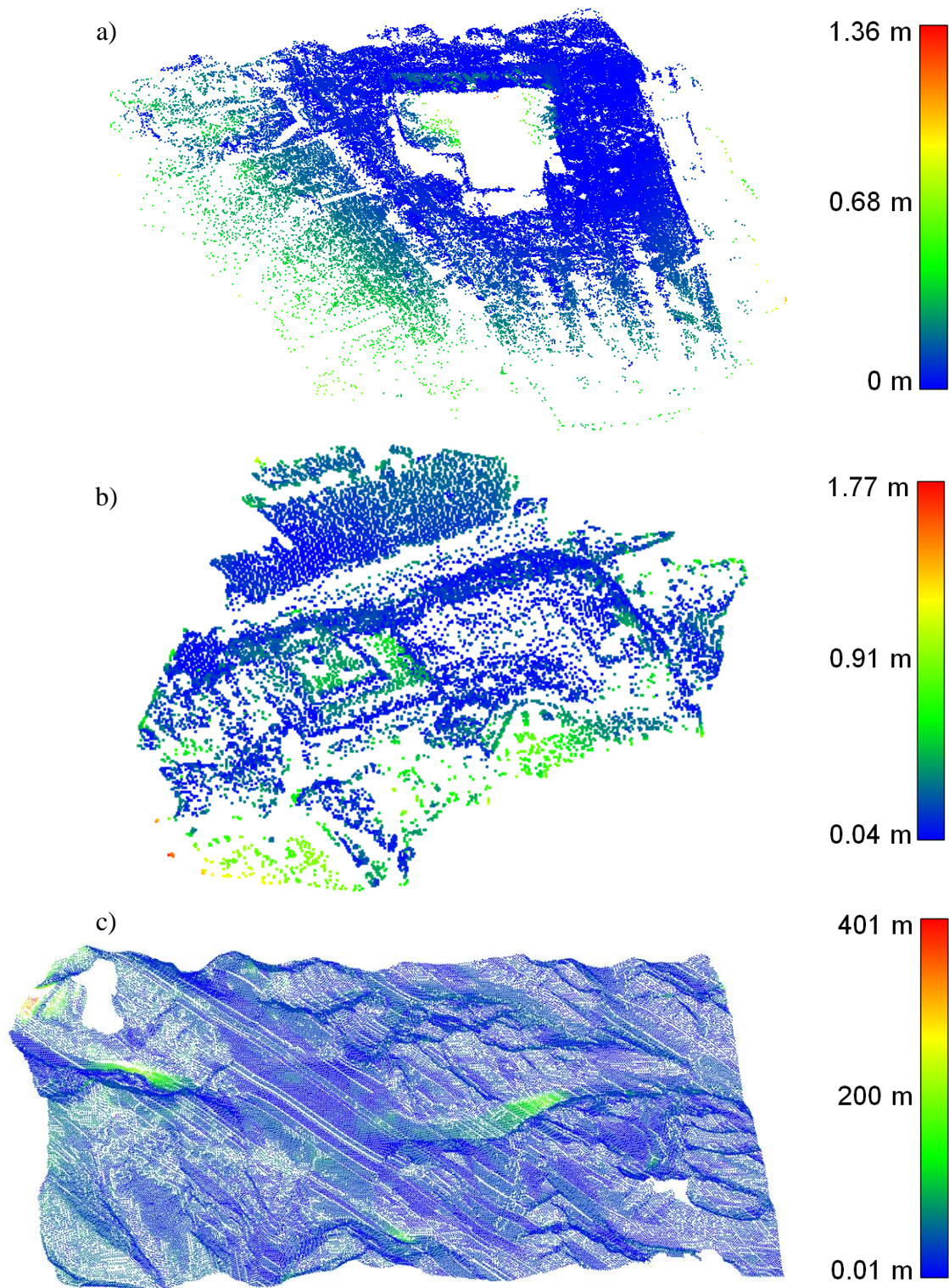


Figure 5.19: Alignment differences between source and target point clouds for the *real*

test datasets. a) *(Urban, Loc1)*, b) *(Urban, Loc2)*, c) *(Non-Urban, Loc3)*.

5.3 Overall assessment of the proposed 3D-based and height map-based co-registration methods

In this section, the quality of co-registration results obtained from the two proposed methods is compared with each other for the three *real* test datasets in Table 5.10. Comparisons are also made with two existing 3D keypoint matching pipelines. For scale-invariant 3D keypoint detection, the 3D-SIFT algorithm (Rusu and Cousins, 2011; Hänsch et al., 2014) was used. For local attribute assignment around the scale-invariant keypoints, two rotation-invariant, histogram-based 3D point cloud descriptors were evaluated: Fast Point Feature Histograms (FPFH) (Rusu et al., 2009), and Signature of Histograms of Orientations (SHOT) (Tombari et al., 2010). The 3D-SIFT, FPFH and SHOT implementations from the Point Cloud Library (Rusu and Cousins, 2011) were used. The number of bins for FPFH and SHOT were empirically tuned in a similar manner to the RGSF and also using the same tuning dataset defined in Section 5.1.1.

Using the ‘reference’ parameters, three measures to indicate co-registration quality were computed: (i) the absolute value of the difference between automatically computed scale and the reference scale value, $|s_{error}|$, (ii) the absolute mean rotational error (AMRE) (i.e., average value of the absolute differences between the automatically-derived and reference angular parameters), and iii) the absolute mean translation error (AMTE) (i.e., average value of the absolute differences between the automatically-derived and reference translation parameters).

From Table 5.16, when compared to reference data, the height map co-registration method produced: (i) scale errors range from 0.010 to 0.080, (ii) rotation errors range from approximately 0.013° to 0.300° and (iii) translation errors range from 0.090m to 0.856m. Tables 5.17, 5.18, 5.19 show the co-registration errors for the 3D keypoint matching approaches: (i) the proposed surface curvature-based 3D keypoint detector and the proposed RGSH point cloud descriptor, (ii) the 3D-SIFT keypoint detector and the FPFH point cloud descriptor, and (iii) the 3D-SIFT keypoint detector and the SHOT point cloud descriptor.

5.3.1 Observations for *real* datasets 1 and 2 (*Urban, Loc1 and Urban, Loc2*)

For the *real* test datasets 1 and 2, the 3D keypoint descriptor methods (i.e., FPFH, SHOT and the proposed RGSH) did not retrieve any inlying point correspondences and hence co-registration was unsuccessful. From the 3D keypoint detector phase, for both the 3D-SIFT and the proposed surface curvature-based detector, it was observed that the scale-invariant local neighbourhoods around ‘true corresponding’ source and target keypoints were generally dissimilar. This resulted in the non-matching of the source and target descriptors. For the 3D descriptor formation phase, the FPFH, SHOT and RGSH were all affected by dissimilar number of points between local source and target keypoint neighbourhoods.

Two main factors were primarily responsible for unsuccessful matching when using the 3D-descriptor based methods on *real* test datasets 1 and 2 (*Urban, Loc1 and Urban,*

Loc2). The first factor relates to missing 3D point clouds due to different viewing perspectives while collecting data. For example, in *real* test dataset 1, the building façade details are missing from the UAV point clouds, whilst the roof structure details are nonexistent on the MLS point clouds. For instance, the lack of points on the building walls from the source UAV, and the absence of points on the roof for the target MLS datasets produced dissimilar descriptors between ‘corresponding’ source and target keypoints on building corners. A similar issue was also present for *real* test dataset 2, where there was a sparsity of building façade details for the ALS point clouds. The second factor for unsuccessful matching relates to significant differences in 3D point cloud density and point distribution between the source and target datasets. In particular, this affects the matching of lower resolution to higher resolution point clouds or vice versa, as well as, the matching of regularly gridded, uniform point clouds to those which have an irregular, non-gridded distribution or vice versa. In Section 5.1.3, the proposed surface curvature-based 3D keypoint detector and the RGS point cloud descriptor successfully co-registered urban datasets with slight differences in point density. However, both of these source and target datasets had similar point distribution patterns which were uniform and regularly gridded.

The experiments on the urban *real* test datasets 1 and 2 highlight the weaknesses of 3D co-registration methods to remain robust when applied to source and target 3D point clouds with different characteristics. These include different 3D point cloud sampling density and distribution (amount of detail), as well as absence of 3D point clouds caused

Table 5.16: Co-registration errors using *proposed multi-scale wavelet* 2D keypoint detector and *GLP-RT* descriptor.

<i>Error measure</i>	<i>real test dataset 1 (Urban, Loc1)</i>	<i>real test dataset 2 (Urban, Loc2)</i>	<i>real test dataset 3 (Non-Urban, Loc3)</i>
# source KPs/target KPs	148/189	215/314	216/225
# correspondences	8	11	25
$ s_{error} $	0.080	0.010	0.019
AMRE (°)	0.207	0.300	0.013
AMTE (m)	0.090	0.403	0.856

Table 5.17: Co-registration errors using *proposed surface curvature*-based 3D keypoint detector and *RGSH* descriptor.

<i>Error measure</i>	<i>real test dataset 1 (Urban, Loc1)</i>	<i>real test dataset 2 (Urban, Loc2)</i>	<i>real test dataset 3 (Non-Urban, Loc3)</i>
# source KPs/target KPs	363/641	330/552	576/608
# correspondences	0	0	384
$ s_{error} $	-	-	0.008
AMRE (°)	-	-	0.006
AMTE (m)	-	-	0.439

Table 5.18: Co-registration errors using *3D-SIFT* 3D keypoint detector and *FPFH* descriptor.

<i>Error measure</i>	<i>real test dataset 1 (Urban, Loc1)</i>	<i>real test dataset 2 (Urban, Loc2)</i>	<i>real test dataset 3 (Non-Urban, Loc3)</i>
# source KPs/target KPs	301/527	278/491	187/251
# correspondences	0	0	46
$ s_{error} $	-	-	0.017
AMRE (°)	-	-	0.011
AMTE (m)	-	-	0.733

Table 5.19: Co-registration errors using *3D-SIFT* 3D keypoint detector and *SHOT* descriptor.

<i>Error measure</i>	<i>real test dataset 1 (Urban, Loc1)</i>	<i>real test dataset 2 (Urban, Loc2)</i>	<i>real test dataset 3 (Non-Urban, Loc3)</i>
# source KPs/target KPs	301/527	278/491	187/251
# correspondences	0	0	49
$ s_{error} $	-	-	0.017
AMRE (°)	-	-	0.009
AMTE (m)	-	-	0.684

by different data collection viewpoints (which produces ‘holes’ in the 3D dataset). Similar observations have been also been reported by Mahiddine et al. (2015) and Mellado et al. (2016). The influence of such heterogeneous point data properties is minimized when feature matching is applied on the height map image pairs.

5.3.2 Observations for *real dataset 3 (Non-Urban, Loc3)*

In the non-urban, *real test dataset 3* experiments, where the source and target datasets have the same point density and point distribution characteristics, the 3D-based methods (Tables 5.17 to 5.19) produced a larger number of inlying correspondences compared to the height map approach (Table 5.16). This is associated with the strength of the 3D-based co-registration methods which directly utilize the point cloud’s 3D surface structure information for extracting keypoints and for defining descriptors. In contrast, the height map-based descriptors are limited to morphological terrain attributes from a 2D perspective.

For *real* test dataset 3, there is a notable disparity in the number of detected and matched keypoints between the two existing 3D co-registration methods (i.e., 3D-SIFT/FPFH and 3D-SIFT/SHOT in Tables 5.18 and 5.19) and the proposed 3D co-registration pipeline (Table 5.17). Both the keypoint detection and descriptor generation phases determine the amount of valid matches. Specifically, these factors include: i) the number of detected keypoints, ii) detection of keypoints at similar locations on both the source and target with similarly defined local scales (i.e., similar descriptor neighbourhood regions), and iii) descriptor discriminability (i.e., uniqueness).

For the proposed 3D detector, keypoint density is controlled by the adaptive non-maxima suppression parameter \mathcal{M} . For complex surfaces such as the icefield where deformation has taken place, it is preferable to have more keypoints to increase correspondence rates. Therefore, $\mathcal{M} = 60\%$ was used for all experiments (urban and non-urban cases). For *real* test dataset 3, to achieve similar keypoint density to 3D-SIFT, \mathcal{M} should be in the range of 25% (Table 5.20).

It was observed for similar keypoint densities, the proposed 3D detector had 112 more occurrences of similar keypoint locations (with similar local neighbourhood regions) on both the source and target in comparison to 3D-SIFT (Table 5.20). This is associated with the different local scale-space extrema detection approaches used by each method. Furthermore, the 3D-SIFT computes keypoints using a voxel grid representation of the data versus the raw 3D points used by the proposed method.

The number of matches obtained for *real* test dataset 3 was also influenced by the 3D descriptor used. For further evaluation, the proposed 3D detector ($\mathcal{M} = 25\%$) was used in

Table 5.20: Comparison of 3D keypoint detectors for ‘real dataset 3’ based on localization accuracy and similarity of local keypoint scales.

<i>3D Keypoint detector</i>	<i># of detected keypoints (source/target)</i>	<i># of keypoints at same locations on source & target with similar local scales</i>	<i>Density (point/km²)</i>
3D-SIFT	187/251	64	≈ 1
Proposed surface curvature approach ($\mathcal{M} = 25\%$)	241/254	176	≈ 1
Proposed surface curvature approach ($\mathcal{M} = 60\%$)	576/608	429	≈ 2

combination with the FPFH and SHOT descriptors, producing 104 and 123 inlying matches respectively. In comparison, a higher number of correspondences (i.e., 134) were obtained when the proposed RGSF descriptor was used. This highlights the discriminative strength of the RGSF, i.e., its capability to provide a unique set of attributes for matching keypoints on 3D point cloud surfaces.

5.4 Computation time

Based on height map matching performed on the 15 datasets (listed in Tables 5.9 and 5.10), the average computation time for the pipeline (i.e., from keypoint detection to modified-RANSAC) is 2 minutes and 17 seconds using MATLAB code on an Intel CPU at 3.4 GHz. Processing times for the 3D co-registration framework depend on the density and size of the point cloud datasets. From all the evaluated datasets for the proposed 3D

keypoint matching method, the Columbia icefield dataset had the largest coverage and greatest number of point clouds (i.e., ‘Real dataset 3’ in Section 5.3). On an Intel CPU at 3.4 GHz using MATLAB code, the total processing time for alignment of this scene was 4 hours and 47 minutes.

6. Conclusions

This research has investigated and proposed two approaches for automating the alignment of 3D source and target point clouds collected from various data acquisition systems (e.g., UAVs, LIDAR and satellite imagery). The developed methods do not require approximate alignment between the point cloud datasets to be co-registered. The first approach is a *3D-based point cloud co-registration* method and the second approach is a *2D height map-based point cloud co-registration* method.

Both of these methods follow a feature matching workflow which includes three main steps: keypoint extraction, keypoint descriptor generation and matching of keypoint descriptors. The proposed alignment methods can be used for co-registering source and target point clouds which differ in terms of a global scale factor, 3D rotation, 3D translation and having overlapping coverage without the need for initial transformation parameters. The first method is carried out entirely in the 3D point cloud domain whereas the second method uses height map images of the 3D point clouds for 2D-based keypoint feature matching.

Experimental analysis showed that the selection of using one co-registration method instead of the other depends strongly on the characteristics of the point cloud dataset. The 3D-based method for point cloud co-registration relies on local neighbourhood patches with similar point characteristics to facilitate strong local region matching. However, point cloud pairs to be aligned can have different point densities, different point distributions and different level of details (i.e., missing point clouds due to data being

collected from different viewing perspectives). These factors increase the fragility of 3D descriptors. Such issues can be addressed by using the continuous 2D height map image representation of the point clouds to perform feature matching operations. While both methods are automated approaches with respect to the extraction and matching of keypoints, prior knowledge of the dataset characteristics is required for the selection of the method.

The work and contents of this dissertation have contributed to the following publications (Persad and Armenakis 2015, 2016, 2017a, 2017b, 2017c; Persad et al., 2017).

6.1 Research outcomes

This dissertation has presented several research contributions towards solving the 3D point cloud alignment problem. For each of the two developed approaches, these contributions are summarized, as well as the various findings from their respective experimental tests.

6.1.1 Summary of the 3D-based point cloud alignment method

An automated 3D-based point cloud alignment method for urban and non-urban scenes has been presented. The approach can be used for aligning point cloud pairs in different 3D conformal coordinate systems. There are several components within this framework which has been proposed for automatically extracting and matching 3D point features.

These include the development of: (a) a scale invariant, curvature-based keypoint extraction method with adaptive non-maxima suppression, (b) a scale, rotation and translation invariant 3D surface keypoint descriptor, and (c) an approach which uses bipartite graph descriptor matching and a RANSAC-type outlier detection method to find corresponding keypoints independent of any user-specified thresholds.

Experiments conducted in Section 5.1 showed that the automated approach recovered 3D conformal transformation parameters which were comparable to the known reference parameters, even in the presence of significant scale, rotation, and translation changes. The approach was tested under two different scenarios. In the first scenario, the co-registration method was assessed under a “controlled”, noise-free setting, whereby the source and target pairs are from the same sensor data acquisition system but with different reference systems. In the second case, the method was evaluated using source and target point clouds which were in different reference systems and generated from different sensors, thereby introducing additional challenges to the matching process such as different overlap, different point density, geometric noise/distortions, and deformation. The experiment for the first case, using an urban scene, produced an absolute scale factor error of 0.0107, an average rotation error of 0.097° , and an average translation error of 0.020 m relative to the reference parameters. The experiment for the non-urban, glacier dataset resulted in an absolute scale factor error of 0.0014, an average rotation error of 0.122° , and an average translation error of 0.084m. In the second case, the results for the urban scene showed an absolute scale factor error of 0.0014, an average rotation error of 0.850° , and an average translation error of 0.013m. For the non-urban scene, the co-

registration of the glacier point cloud pair had an absolute scale factor error of 0.0002, an average rotation error of 0.073° , and an average translation error of 0.013m. On these evaluated datasets, the absolute mean alignment differences relative to reference transformation parameters are in the range of 0.23m to 2.81m. The alignment errors from the proposed 3D co-registration method met the proximity requirements of the data characteristics.

The developed method was also assessed on the entire Columbia icefield dataset (Section 5.3). For this experiment, the proposed 3D co-registration approach produced the highest number of correspondences and the lowest parameter transformation errors in comparison to the other evaluated approaches.

6.1.2 Summary of the Height map-based point cloud

alignment method

A height map-based approach for the automatic co-registration of multi-sensor 3D point clouds in different 3D conformal coordinate systems has been presented. The method uses height maps formed from the 3D point clouds for the extraction of keypoints, formation of keypoint descriptors and their subsequent matching. Specific contributions are in the development of (i) a wavelet-based, multi-scale 2D keypoint detector, (ii) a 2D scale, rotation and translation invariant keypoint descriptor utilizing Gabor derivatives and the Rapid transform, and (iii) a bidirectional nearest neighbor approach to find matching keypoints.

Based on experiments performed in Section 5.2, the method overcomes some of the limitations faced by 3D descriptor-based co-registration approaches and is able to automatically align multi-sensor, urban and non-urban 3D point clouds which differ in terms of overlap, point distribution and density, sensor viewpoint variations (i.e., missing data), scale, 3D rotation and 3D translation. Co-registration experiments with urban and non-urban scenes produced scale errors ranging from 0.010 to 0.080, 3D rotation errors in the order of 0.013° to 0.300° and 3D translation errors from 0.090m to 0.856m. On these evaluated datasets, the absolute mean alignment differences relative to reference transformation parameters are in the range of 0.17m to 1.21m. The alignment errors from the proposed height map-based co-registration method met the proximity requirements of the data characteristics.

The proposed 2D detector and 2D descriptor obtained higher true positive and lower false positive height map keypoint matching accuracies when compared to existing 2D-based keypoint correspondence methods (Section 5.2.3).

6.2 Recommendations for future work

There are several aspects of each approach which can be explored for future research. The 3D-based co-registration method currently utilizes two surface attributes for the RGSF descriptor formation. Expansion of this descriptor with additional 3D surface attributes can potentially improve keypoint correspondence results by introducing supplementary shape information into the matching process. Furthermore, the proposed 3D co-registration approach is not robust to variations in point density and point

distribution. Future work can investigate the generation surface mesh descriptors which are independent of any point-based attributes. The computational efficiency of the 3D-based co-registration pipeline can also be improved by conversion to a low-level language (e.g., C++) and through the use of parallel programming for subtasks such as keypoint detection and descriptor generation.

The GLP-RT descriptor for the height map-based co-registration method is multi-dimensional. In future research, descriptor dimensionality reduction can also be explored using methods such as PCA. A more compact and compressed descriptor representation can potentially improve overall discriminability and the matching results.

For both the 3D-based and height map-based approaches, expansion of the respective descriptor tuning databases is highly recommended. The use of more tuning data will refine the descriptor's parameters and accommodate overall descriptor generalization for use with other datasets.

Both of the proposed co-registration methods utilize keypoints for feature correspondences. Alternatively, other geometric primitives can be extracted and used for matching, namely keylines or keyplanes. Such higher order features provide more geometrical and structural information about the scene in comparison to point features (Fan et al., 2010). For instance, linear or planar features can provide additional geometrical attributes such as line or plane orientation similarity, which can potentially strengthen the matching process when used in combination with local point descriptors. Curvilinear features such as ridges or crestlines as used in medical image analysis (Pennec et al., 2010) can also be explored.

The descriptors used in this work can handle changes in scale, rotation and translation between the source and target. For the icefield dataset, surface deformation changes were prevalent in certain regions. The developed RGSH and GLP-RT descriptors were not invariant to such deformations and thus, were unable to match keypoints in these areas. Therefore, future work can explore the development of surface deformation-invariant descriptors.

References

Agisoft, 2016. <http://www.agisoft.com/> (Accessed on 21.3.2016).

Aiger, D., N.J. Mitra, and D. Cohen-Or, 2008. 4-points congruent sets for robust pairwise surface registration. *ACM Transactions on Graphics (TOG)*, 27(3), p. 85.

Al-Manasir, K. and C.S. Fraser, 2006. Registration of terrestrial laser scanner data using imagery. *The Photogrammetric Record*, 21(115):255-268.

Andrei, C.O., 2006. 3D affine coordinate transformations. MSc Thesis in Geodesy No. 3091. School of Architecture and the Built Environment Royal Institute of Technology (KTH) Stockholm, Sweden.

Azad, P., T. Asfour, and R. Dillmann, 2009. Combining Harris interest points and the SIFT descriptor for fast scale-invariant object recognition. In *Intelligent Robots and Systems, 2009 (IROS'2009) Proceedings*. IEEE, pp. 4275 – 4280.

Bae, K.H. and D.D. Lichti, 2008. A method for automated registration of unorganized point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(1):36-54.

Barnea, S. and S. Filin, 2007. Registration of terrestrial laser scans via image based features. *International Archives of Photogrammetry and Remote Sensing*, 36(3/W52), pp. 26-31.

Barnea, S. and S. Filin, 2008. Keypoint based autonomous registration of terrestrial laser point-clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(1):19-35.

Bay, H., A. Ess, T. Tuytelaars, and L. Van Gool, 2008. Speeded-up robust features (SURF). *Computer vision and image understanding*, 110(3):346-359.

Belongie, S., J. Malik, and J. Puzicha, 2002. Shape matching and object recognition using shape contexts, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 24(4):509-522.

Bentley, J.L., 1975. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509-517.

Berretti, S., N. Werghi, A. Del Bimbo, and P. Pala, 2013. Matching 3D face scans using interest points and local histogram descriptors. *Computers & Graphics*, 37(5):509-525.

- Besl, P.J. and N. D. McKay, 1992. A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14 (2):239– 256.
- Biber, P., 2003. The normal distributions transform: A new approach to laser scan matching. Tech. Rep., vol. 3. Wilhelm Schickard Institute for Computer Science, Graphical-Interactive Systems (WSI/GRIS), University of Tübingen.
- Böhm, J., and S. Becker, 2007. Automatic marker-free registration of terrestrial laser scans using reflectance features. In: Gruen, A., Kahmen, H. (Eds.), *Optical 3-D Measurement Techniques VIII*, pp. 338–344.
- Bouaziz, S., A. Tagliasacchi, and M. Pauly, 2013. Sparse iterative closest point. In *Computer graphics forum*, 32(5):113-123. Blackwell Publishing Ltd.
- Bourgeois, F. and J. C. Lassalle, 1971. An extension of the Munkres algorithm for the assignment problem to rectangular matrices. *Communications of the ACM*, 14(12):802-804.
- Brenner, C., C. Dold, and N. Ripperda, 2008. Coarse orientation of terrestrial laser scans in urban environments. *ISPRS journal of photogrammetry and remote sensing*, 63(1):4-18.
- Bronstein, M.M. and Kokkinos, I., 2010. Scale-invariant heat kernel signatures for non-rigid shape recognition. *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010*, pp. 1704-1711.
- Brown, M., Szeliski, R., Winder, S., 2005. Multi-image matching using multi-scale oriented patches. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, vol. 1, pp. 510–517.
- Bruno, E. and D. Pellerin, 2002. Robust motion estimation using spatial Gabor-like filters. *Signal Processing*, 82(2), pp.297-309.
- Castellani, U. and A. Bartoli, 2012. 3d shape registration. In *3D Imaging, Analysis and Applications*, (pp. 221-264). Springer London.
- Chan, T.O., Lichti, D.D., Belton, D. and Nguyen, H.L., 2016. Automatic Point Cloud Registration Using a Single Octagonal Lamp Pole. *Photogrammetric Engineering & Remote Sensing*, 82(4):257-269.
- Chen, Y. and G. Medioni, 1992. Object modelling by registration of multiple range images. *Image and vision computing*, 10(3):145-155.

- Childs, C., 2004. Interpolating surfaces in ArcGIS spatial analyst. *ArcUser*, July-September, 3235.
- Chui, H. and Rangarajan, A., 2003. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2):114-141.
- Cooley, J. W. and Tukey, O. W. 1965. An Algorithm for the Machine Calculation of Complex Fourier Series. *Math. Comput.* 19, 297-301.
- Coria, L.E., Pickering, M.R., Nasiopoulos, P., Ward, R.K., 2008. A video watermarking scheme based on the dual-tree complex wavelet transform. *IEEE Trans. Inf. Forensics Secur.* 3(3):466–474.
- Corsini, M., M. Dellepiane, F. Ganovelli, R. Gherardi, A. Fusiello, and R. Scopigno, 2013. Fully automatic registration of image sets on approximate geometry. *International journal of computer vision*, 102(1-3):91-111.
- Desbrun, M., M. Meyer, P. Schröder and A. H. Barr, 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (pp. 317-324). ACM Press/Addison-Wesley Publishing Co.
- Dorkó, G. and C. Schmid, 2006. Maximally stable local description for scale selection. In *European Conference on Computer Vision* (pp. 504-516). Springer Berlin Heidelberg.
- Fan, B., F. Wu, Z. Hu, 2010. Line matching leveraged by point correspondences, In: *Computer Vision and Pattern Recognition, CVPR 2010*, pp. 390–397.
- Fauqueur, J., Kingsbury, N., Anderson, R., 2006. Multiscale keypoint detection using the dual-tree complex wavelet transform. In: *2006 IEEE International Conference on Image Processing*, pp. 1625–1628.
- Fischler, M. A. and R. C. Bolles, 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381-395.
- Förstner, W. and Gülch, E., 1987. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Proc. ISPRS intercommission conference on fast processing of photogrammetric data* (pp. 281-305).
- Flitton, G.T., T.P. Breckon, and N.M., Bouallagu, 2010. Object Recognition using 3D SIFT in Complex CT Volumes. *Proceedings of the British Machine Vision Conference*. pp. 11.1–12.

Ge, X., and T. Wunderlich, 2016. Surface-based matching of 3D point clouds with variable coordinates in source and target system. *ISPRS Journal of Photogrammetry and Remote Sensing*, 111, 1-12.

Grauman, K. and Leibe, B., 2011. Visual object recognition. *Synthesis lectures on artificial intelligence and machine learning*, 5(2), pp.1-181.

Gruen, A. and D. Akca, 2005. Least squares 3D surface and curve matching. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(3):151-174.

Guo, Y., F. Sohel, M. Bennamoun, M. Lu, and J. Wan, 2013. Rotational projection statistics for 3D local surface description and object recognition. *International journal of computer vision*, 105(1), pp. 63-86.

Haghighat, M., S. Zonouz, and M. Abdel-Mottaleb, 2013. Identification using encrypted biometrics. In *Computer analysis of images and patterns* (pp. 440-448). Springer Berlin Heidelberg.

Hamamoto, Y., S. Uchimura, M. Watanabe, T. Yasuda, Y. Mitani, and S. Tomita, 1998. A Gabor filter-based method for recognizing handwritten numerals. *Pattern Recognition*, 31(4):395-400.

Hänsch, R., Weber, T. and Hellwich, O., 2014. Comparison of 3D interest point detectors and descriptors for point cloud fusion. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3):57.

Harris, C. and Stephens, M., 1988. A combined corner and edge detector. In *Alvey vision conference*, 15(50):10-5244.

Hartley, R. and A. Zisserman, 2000. *Multiple View Geometry in Computer Vision*. Cambridge University Press.

Herman, R.L., 2013. An Introduction to Fourier and Complex Analysis With Application to the Spectral Analysis of Signals. *University of North Carolina Wilmington, Wilmington, NC, online publication*, http://people.uncw.edu/hermanr/mat367/FCABook/FCA_Main2015.pdf. (Accessed on 1 August 2017).

Hill, P.R., Bull, D.R., Canagarajah, C.N., 2005. Image fusion using a new framework for complex wavelet transforms. *IEEE International Conference on Image Processing*, vol. 2, pp. II-1338.

Horn, B. K., 1987. Closed-form solution of absolute orientation using unit quaternions. *JOSA A*, 4(4):629-642.

- Huang, D., Zhu, C., Wang, Y., Chen, L., 2014. HSOG: a novel local image descriptor based on histograms of the second-order gradients. *IEEE Trans. Image Process.* 23 (11):4680–4695.
- Jian, B. and Vemuri, B.C., 2005. A robust algorithm for point set registration using mixture of Gaussians. *IEEE ICCV*, vol. 2, Oct. 2005, pp. 1246–1251.
- Jiang, W., Lam, K.M., Shen, T.Z., 2009. Efficient edge detection using simplified Gabor wavelets. *IEEE Transactions on Cybernetics Systems, Man, and Cybernetics, Part B* 39 (4):1036–1047.
- Johnson, A.E., Hebert, M., 1999. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* 21(5):433–449.
- Jones, J.P. and L.A. Palmer, 1987. An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *Journal of neurophysiology*, 58(6), pp. 1233-1258.
- Jurie, F., 1999. A new log-polar mapping for space variant imaging.: Application to face detection and tracking. *Pattern Recognition*, 32(5), pp. 865-875.
- Kamarainen, J.K., Kyrki, V., Kälviäinen, H., 2006. Invariance properties of Gabor filter-based features-overview and applications. *IEEE Trans. Image Process.* 15(5):1088–1099.
- Kang, Z., J. Li, L. Zhang, Q. Zhao, and S. Zlatanova, 2009. Automatic registration of terrestrial laser scanning point clouds using panoramic reflectance images. *Sensors*, 9(4):2621-2646.
- Ke, Y., Sukthankar, R., 2004. PCA-SIFT: A more distinctive representation for local image descriptors. Proceedings of the 2004 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. CVPR 2004, vol. 2, pp. II-506.
- Kimmel, R., and J. A. Sethian, 1998. Computing geodesic paths on manifolds. *Proceedings of the National Academy of Sciences*, 95(15):8431-8435.
- Kingsbury, N., 1998. The dual-tree complex wavelet transform: a new efficient tool for image restoration and enhancement. *IEEE 9th European Signal Processing Conference (EUSIPCO 1998)*, pp. 1–4.
- Kokkinos, I., M. Bronstein, and A. Yuille, 2012. Dense Scale Invariant Descriptors for Images and Surfaces. *Research Report RR-7914*, INRIA, March 2012.
- Kokkinos, I., Yuille, A., 2008. Scale invariance without scale selection. *IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 2008, pp. 1–8.

Konishi, S., A.L. Yuille, J.M. Coughlan, and S.C. Zhu, 2003. Statistical edge detection: Learning and evaluating edge cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):57-74.

Lai, K. and D. Fox, 2010. Object recognition in 3D point clouds using web data and domain adaptation. *The International Journal of Robotics Research*, 29(8):1019-1037.

Lehmann, R., 2014. Transformation model selection by multiple hypotheses testing. *Journal of Geodesy*, 88(12):1117-1130.

Li, Z., Q. Zhu, C. Gold, 2005. *Digital Terrain Modeling – Principles and Methodology*. CRC Press, ISBN: 0-415-32462-9.

Li, H., Sumner, R.W. and Pauly, M., 2008. Global Correspondence Optimization for Non-Rigid Registration of Depth Scans. In *Computer graphics forum*, 27(5):1421-1430. Blackwell Publishing Ltd.

Li, C., J. Li, D. Gao, and B. Fu, 2014. Rapid-transform based rotation invariant descriptor for texture classification under non-ideal conditions. *Pattern Recognition*, 47(1):313-325.

Lin, S., Lai, Y.K., Martin, R.R., Jin, S. and Cheng, Z.Q., 2016. Color-aware surface registration. *Computers & Graphics*, 58, pp.31-42.

Lindeberg, T., 1998. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79-116.

Lowe, D., 1999. Object recognition from local scale-invariant features. *Inter. Conf. Computer Vision*, pages 1150–1157.

Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91-110.

Luhmann, T., Robson, S., Kyle, S.A. and Harley, I.A., 2006. *Close range photogrammetry: principles, techniques and applications*. Whittles.

Mahiddine, A., Iguernaissi, R., Merad, D., Drap, P. and Boï, J.M., 2015. 3D Registration of multi-modal data using surface fitting. *ICPRAM (2)*, pp. 71–78.

Mallat, S.G., 1989. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (7):674–693.

Mallat, S., Zhong, S., 1992. Characterization of signals from multiscale edges. *IEEE Trans. Pattern Anal. Mach. Intell.* 14 (7):710–732.

- Mallat, S., 1996. Wavelets for a vision. *Proceedings of the IEEE*, 84(4), pp. 604-614.
- Martínez, B. and M.A. Gilabert, 2009. Vegetation dynamics from NDVI time series analysis using the wavelet transform. *Remote Sensing of Environment*, 113(9):1823-1842.
- Mellado, N., D. Aiger, and N.J. Mitra, 2014. Super 4pcs fast global point cloud registration via smart indexing. In *Computer Graphics Forum*, 33(5):205-215.
- Mellado, N., Dellepiane, M. and Scopigno, R., 2016. Relative scale estimation and 3D registration of multi-modal geometry using Growing Least Squares. *IEEE transactions on visualization and computer graphics*, 22(9):2160-2173.
- Mikolajczyk, K. and C. Schmid, 2004. Scale & affine invariant interest point detectors. *International journal of computer vision*, 60(1):63-86.
- Moravec, H.P., 1980. *Obstacle avoidance and navigation in the real world by a seeing robot rover* (No. STAN-CS-80-813). STANFORD UNIV CA DEPT OF COMPUTER SCIENCE.
- Neubeck, A., Van Gool, L., 2006. Efficient non-maximum suppression. *18th International Conference on Pattern Recognition, ICPR 2006*, vol. 3, pp. 850–855.
- Novák, D. and K. Schindler, 2013. Approximate registration of point clouds with large scale differences. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1(2):211-216.
- Palander, K., Brandt, S.S., 2008. Epipolar geometry and log-polar transform in widebaseline stereo matching. *19th International Conference on Pattern Recognition, ICPR 2008*, pp. 1–4.
- Pauly, M., R. Keiser and M. Gross, 2003. Multi-scale Feature Extraction on Point-Sampled Surfaces. In *Computer graphics forum*, 22(3):281-289. Blackwell Publishing, Inc.
- Persad, R.A. and Armenakis, C., 2015. Alignment of point cloud DSMs from TLS and UAV Platforms. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(1), p.369.
- Persad, R.A. and Armenakis, C., 2016. Co-registration of DSMs generated by UAV and terrestrial laser scanning systems.. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 41.

Persad, R.A. and Armenakis, C., 2017a. Automatic 3D Surface Co-Registration Using Keypoint Matching. *Photogrammetric Engineering & Remote Sensing*, 83(2):137-151.

Persad, R.A. and Armenakis, C., 2017b. Automatic co-registration of 3D multi-sensor point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 130, pp.162-186.

Persad, R.A. and Armenakis, C., 2017c. Comparison of 2D and 3D approaches for the alignment of UAV and LIDAR point clouds. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* (Accepted for publication).

Persad, R.A., Armenakis, C. Hopkinson, C., and Brisco, B, 2017. Automatic registration of 3-D point clouds from UAS and airborne LiDAR platforms. *Journal of Unmanned Vehicle Systems*, (doi: 10.1139/juvs-2016-0034).

Pennec, X., Ayache, N., Thirion, J.-P., 2000. Landmark-based Registration using Features Identified through Differential Geometry, In: *Handbook of medical imaging - Processing and Analysis*. I., Academic Press. pp. 499–513.

Pinkall, U. and K. Polthier, 1993. Computing discrete minimal surfaces and their conjugates. *Experimental mathematics*, 2(1):15-36.

Pun, C.M. and M.C. Lee, 2003. Log-polar wavelet energy signatures for rotation and scale invariant texture classification. , *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):590-603.

Ranchin, T. and L. Wald, 1993. The wavelet transform for the analysis of remotely sensed images. *International Journal of Remote Sensing*, 14(3):615-619.

Reddy, B.S. and B.N. Chatterji, 1996. An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE transactions on image processing*, 5(8):1266-1271.

Reitboeck, H. and T.P. Brody, 1969. A transformation with invariance under cyclic permutation for applications in pattern recognition. *Information and Control*, 15(2):130-154.

Ruiz-Correa, S., Shapiro, L.G., Meila, M. and Berson, G., 2004. Discriminating deformable shape classes. *Advances in Neural Information Processing Systems* (pp. 1491-1498).

Rusinkiewicz, S., M. Levoy, 2001. *Efficient Variants of the ICP Algorithm*; IEEE Computer Soc.: Los Alamitos, CA, USA; pp. 145–152.

Rusu, R.B., N. Blodow, and M. Beetz, 2009. Fast point feature histograms (FPFH) for 3D registration. *IEEE International Conference on Robotics and Automation*, 2009 (ICRA'09). IEEE, pp. 3212–3217.

Rusu, R.B., 2009. Semantic 3d object maps for everyday manipulation in human living environments. *Ph.D. dissertation, Inst. fur Informatik, Technische Univ. Munchen, Munich, Germany, 2009.*

Rusu, R.B., Cousins, S., 2011. 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China.

Salti, S., A. Petrelli, F. Tombari, and L.D. Stefano, 2012. On the affinity between 3D detectors and descriptors. *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, IEEE, pp. 424-431.

Salvi, J., C. Matabosch, D. Fofi, and J. Forest, 2007. A review of recent range image registration methods with accuracy evaluation. *Image and Vision computing*, 25(5):578-596.

Scovanner, P., S. Ali, and M., Shah, 2007. A 3-dimensional sift descriptor and its application to action recognition. *Proceedings of the 15th International Conference on Multimedia*, ACM, pp. 357-360.

Schwartz E. L., 1994. Topographical mapping in primate visual cortex: history, anatomy and computation, in *Visual Science and Engineering: Models and Applications*, Ed. D H Kelly (New York: Marcel Dekker), pp. 293-359.

Sehgal, A., D. Cernea, and M. Makaveeva, 2010. Real-time scale invariant 3D range point cloud registration. *International Conference of Image Analysis and Recognition* (pp. 220-229). Springer Berlin Heidelberg.

Selesnick, I.W., R.G. Baraniuk, and N.G. Kingsbury, 2005. The dual-tree complex wavelet transform. *Signal Processing Magazine, IEEE*, 22(6):123-151.

Sipiran, I., and B. Bustos, 2011. "Harris 3D: a robust extension of the Harris operator for interest point detection on 3D meshes." *The Visual Computer*, 27(11):963-976.

Stamos, I., M. Leordeanu, 2003. Automated feature-based range registration of urban scenes of large scale. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. II, IEEE CS Press 2003, pp. 555–561.

Stoica, P., Moses, R.L., 2005. *Spectral Analysis of Signals*. Pearson/Prentice Hall, Upper Saddle River, NJ.

Sun, J., M. Ovsjanikov and L. Guibas, 2009. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer graphics forum*, 28(5):1383-1392. Blackwell Publishing Ltd.

Szeliski, R., 2010. *Computer Vision: Algorithms and Applications*. Springer, New York.

Tabernero, A., Portilla, J., Navarro, R., 1999. Duality of log-polar image representations in the space and spatial-frequency domains. *IEEE Trans. Signal Process.* 47 (9):2469–2479.

Tam, G.K., Z.Q. Cheng, Y.K. Lai, F.C. Langbein, Y. Liu, D. Marshall, R.R. Martin, X.F. Sun and P.L. Rosin, 2013. Registration of 3D point clouds and meshes: A survey from rigid to nonrigid, *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1199–1217.

Tang, H., Joshi, N., Kapoor, A., 2011. Learning a blind measure of perceptual image quality. *IEEE Conference on Computer Vision and Pattern Recognition CVPR 2011*, pp. 305–312.

Teran, L. and P., Mordohai, 2014. 3d interest point detection via discriminative learning. In *European Conference on Computer Vision* (pp. 159-173). Springer, Cham.

Theiler, P.W., J.D. Wegner, and K. Schindler, 2014. Keypoint-based 4-Points Congruent Sets—Automated marker-less registration of laser scans. *ISPRS Journal of Photogrammetry and Remote Sensing*, 96, pp. 149-163.

Tola, E., Lepetit, V., Fua, P., 2010. Daisy: an efficient dense descriptor applied to wide-baseline stereo. *IEEE Trans. Pattern Anal. Mach. Intell.* 32 (5):815–830.

Tombari, F., S. Salti and L. Di Stefano, 2010. Unique signatures of histograms for local surface description. In *Computer Vision—ECCV 2010* (pp. 356-369). Springer Berlin Heidelberg.

Tombari, F. and L. Di Stefano, 2014. Interest Points via Maximal Self-Dissimilarities. In *Computer Vision—ACCV 2014* (pp. 586-600). Springer International Publishing.

Traver, V.J. and Bernardino, A., 2010. A review of log-polar imaging for visual perception in robotics. *Robotics and Autonomous Systems*, 58(4):378-398.

Tuytelaars, T. and L. Van Gool, 2004. Matching widely separated views based on affine invariant regions. *International journal of computer vision*, 59(1):61-85.

von Hansen, W., 2006. Robust automatic marker-free registration of terrestrial scan

data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 36 (Part 3), pp. 105–110.

von Hansen, W., Gross, W., Thoennessen, U. 2008. Line-based registration of terrestrial and airborne LIDAR data. *International Archives of Photogrammetry and Remote Sensing*, 37, pp. 161–166.

Wang, Z. and C. Brenner, 2008. Point based registration of terrestrial laser data using intensity and geometry features. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(Part B5), pp. 583-590.

Weber, T., R. Hänsch, and O. Hellwich, 2015. Automatic registration of unordered point clouds acquired by Kinect sensors using an overlap heuristic. *ISPRS Journal of Photogrammetry and Remote Sensing*, 102, 96-109.

Weinmann, Ma., Mi. Weinmann, S. Hinz, and B. Jutzi, 2011. Fast and automatic image-based registration of TLS data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6):S62-S70.

Wetzler, A., Aflalo, Y., Dubrovina, A. and Kimmel, R., 2013. The Laplace-Beltrami operator: a ubiquitous tool for image and shape processing. In *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing* (pp. 302-316). Springer, Berlin, Heidelberg.

Yang, J., Li, H. and Jia, Y., 2013. Go-icp: Solving 3d registration efficiently and globally optimally. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1457-1464).

Yang, B., Y. Zang, Z. Dong, and R. Huang, 2015. An automated method to register airborne and terrestrial laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 109, pp. 62-76.

Yang, B., Z. Dong, F. Liang, and Y. Liu, 2016. Automatic registration of large-scale urban scene point clouds based on semantic feature points. *ISPRS Journal of Photogrammetry and Remote Sensing*, 113, pp. 43-58.

Yu, T.H., O.J. Woodford, and R. Cipolla, 2013. A performance evaluation of volumetric 3D interest point detectors, *International Journal of Computer Vision*, 102(1-3):180–197.

Zaharescu, A., E. Boyer, K. Varanasi, and R. Horaud, 2009. Surface feature detection and description with applications to mesh matching, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009*, pp. 373–380).

Zambanini, S. and M. Kampel, 2013. A local image descriptor robust to illumination changes. In *Image analysis* (pp. 11-21). Springer Berlin Heidelberg.

Zeng, A., Song, S., Nießner, M., Fisher, M. and Xiao, J., 2016. 3DMatch: Learning the matching of local 3D geometry in range scans. *arXiv preprint arXiv:1603.08182*.

Zhong, Y. 2009. “Intrinsic shape signatures: A shape descriptor for 3D object recognition,” *International Conference on Computer Vision Workshops*, pp. 689–696.

Zokai, S., Wolberg, G., 2005. Image registration using log-polar mappings for recovery of large-scale similarity and projective transformations. *IEEE Trans. Image Process.* 14 (10):1422–1434.

Appendix A

Bipartite matching using the Hungarian method

This appendix demonstrates the general procedure for solving the Bipartite matching (one-to-one correspondence problem) using the Hungarian method. Assume that the following cost matrix is given, where KP^S and KP^T are the source and target keypoints respectively (the values within the matrix are the matching scores):

$$\begin{array}{c}
 KP_1^S \quad KP_2^S \quad KP_3^S \\
 \left[\begin{array}{ccc}
 KP_1^T & 0.805 & 0.569 & 0.835 \\
 KP_2^T & 0.539 & 0.522 & 0.562 \\
 KP_3^T & 0.825 & 0.529 & 0.845 \\
 KP_4^T & 0.536 & 0.525 & 0.535
 \end{array} \right]
 \end{array}$$

In the cost matrix above, there are 4 target keypoints and 3 source keypoints, so a dummy column is added to the cost matrix. The values within the dummy column are assigned the highest cost value from the cost matrix as follows:

$$\begin{array}{c}
 KP_1^S \quad KP_2^S \quad KP_3^S \\
 \left[\begin{array}{ccc}
 KP_1^T & 0.805 & 0.569 & 0.835 \\
 KP_2^T & 0.539 & 0.522 & 0.562 \\
 KP_3^T & 0.825 & 0.529 & 0.845 \\
 KP_4^T & 0.536 & 0.525 & 0.535
 \end{array} \right]
 \Rightarrow
 \begin{array}{c}
 KP_1^S \quad KP_2^S \quad KP_3^S \quad dummy \\
 \left[\begin{array}{cccc}
 KP_1^T & 0.805 & 0.569 & 0.835 & 0.845 \\
 KP_2^T & 0.539 & 0.522 & 0.562 & 0.845 \\
 KP_3^T & 0.825 & 0.529 & 0.845 & 0.845 \\
 KP_4^T & 0.536 & 0.525 & 0.535 & 0.845
 \end{array} \right]
 \end{array}
 \end{array}$$

Then, each element in a row is subtracted from the minimum value of the row to which it belongs:

$$\begin{array}{c}
 KP_1^S \quad KP_2^S \quad KP_3^S \quad dummy \\
 \left[\begin{array}{cccc}
 KP_1^T & 0.805 & 0.569 & 0.835 & 0.845 & (-0.569) \\
 KP_2^T & 0.539 & 0.522 & 0.562 & 0.845 & (-0.522) \\
 KP_3^T & 0.825 & 0.529 & 0.845 & 0.845 & (-0.529) \\
 KP_4^T & 0.536 & 0.525 & 0.535 & 0.845 & (-0.525)
 \end{array} \right]
 \Rightarrow
 \begin{array}{c}
 KP_1^S \quad KP_2^S \quad KP_3^S \quad dummy \\
 \left[\begin{array}{cccc}
 KP_1^T & 0.236 & 0 & 0.266 & 0.276 \\
 KP_2^T & 0.017 & 0 & 0.040 & 0.323 \\
 KP_3^T & 0.296 & 0 & 0.316 & 0.316 \\
 KP_4^T & 0.011 & 0 & 0.010 & 0.320
 \end{array} \right]
 \end{array}
 \end{array}$$

After, a similar procedure is applied in a column-wise manner, i.e., each element in a column is subtracted from the minimum value of that column:

$$\begin{array}{cccc}
 & KP_1^S & KP_2^S & KP_3^S & dummy \\
 KP_1^T & 0.236 & 0 & 0.266 & 0.276 \\
 KP_2^T & 0.017 & 0 & 0.040 & 0.323 \\
 KP_3^T & 0.296 & 0 & 0.316 & 0.316 \\
 KP_4^T & 0.011 & 0 & 0.010 & 0.320 \\
 & (-0.011) & (-0) & (-0.010) & (-0.276)
 \end{array}
 \Rightarrow
 \begin{array}{cccc}
 & KP_1^S & KP_2^S & KP_3^S & dummy \\
 KP_1^T & 0.225 & 0 & 0.256 & 0 \\
 KP_2^T & 0.006 & 0 & 0.030 & 0.047 \\
 KP_3^T & 0.285 & 0 & 0.306 & 0.040 \\
 KP_4^T & 0 & 0 & 0 & 0.044
 \end{array}$$

The objective of the next step is to cover all zeros with the least amount of lines possible:

$$\begin{array}{cccc}
 & KP_1^S & KP_2^S & KP_3^S & dummy \\
 KP_1^T & 0.225 & 0 & 0.256 & 0 \\
 KP_2^T & 0.006 & 0 & 0.030 & 0.047 \\
 KP_3^T & 0.285 & 0 & 0.306 & 0.040 \\
 KP_4^T & 0 & 0 & 0 & 0.044
 \end{array}$$

Then, subtract all uncovered matrix elements with the minimum uncovered value, i.e., 0.006. Also, add the minimum value, i.e., 0.006 to those elements covered by two lines:

$$\begin{array}{cccc}
 & KP_1^S & KP_2^S & KP_3^S & dummy \\
 KP_1^T & 0.225 & 0.006 & 0.256 & 0 \\
 KP_2^T & 0 & 0 & 0.024 & 0.041 \\
 KP_3^T & 0.279 & 0 & 0.300 & 0.034 \\
 KP_4^T & 0 & 0.006 & 0 & 0.044
 \end{array}$$

Repeat step 4, i.e., cover all zeros with the least amount of lines possible. The overall objective is to stop this procedure when the number of lines is equivalent to the number of rows (or columns) of the matrix. As shown below, there are 4 lines and the number of rows (or columns) is 4, so this iterative process stops:

	KP_1^S	KP_2^S	KP_3^S	<i>dummy</i>
KP_1^T	0.225	0.006	0.256	0
KP_2^T	0	0	0.024	0.041
KP_3^T	0.279	0	0.300	0.034
KP_4^T	0	0.006	0	0.044

Finally, select a set of zeros such that it unique to only one row and one column:

	KP_1^S	KP_2^S	KP_3^S	<i>dummy</i>
KP_1^T	0.225	0.006	0.256	0
KP_2^T	0	0	0.024	0.041
KP_3^T	0.279	0	0.300	0.034
KP_4^T	0	0.006	0	0.044

Thus, based on the locations of zeros, the one to one correspondence list is given as follows with a minimum matching cost of 1.603 (i.e., $0.539+0.529+0.535$, which is the sum of values from the input cost matrix for the selected correspondence row and column locations):

- a) $KP_1^T, dummy$
- b) KP_2^T, KP_1^S
- c) KP_3^T, KP_2^S
- d) KP_4^T, KP_3^S

Appendix B

Rapid Transform

The rapid transform can be used for identifying the similarity between a pair of images if there is a cyclic shift between them (i.e., translation-invariant pattern matching). Rapid transform takes the pixel values of the images as input and applies a pair of commutative functions (Equation 4.5) on the rows and columns of the images. These functions are independent of the position of the pattern contents of the image and its shifted version.

The following example (Figure B.1) illustrates the rapid transform as applied on a synthetic image and a cyclic-shifted version of the same image. The shifts are in both horizontal and vertical directions. The output of both the original image and its shifted version are the rapid transform coefficients which are both identical.

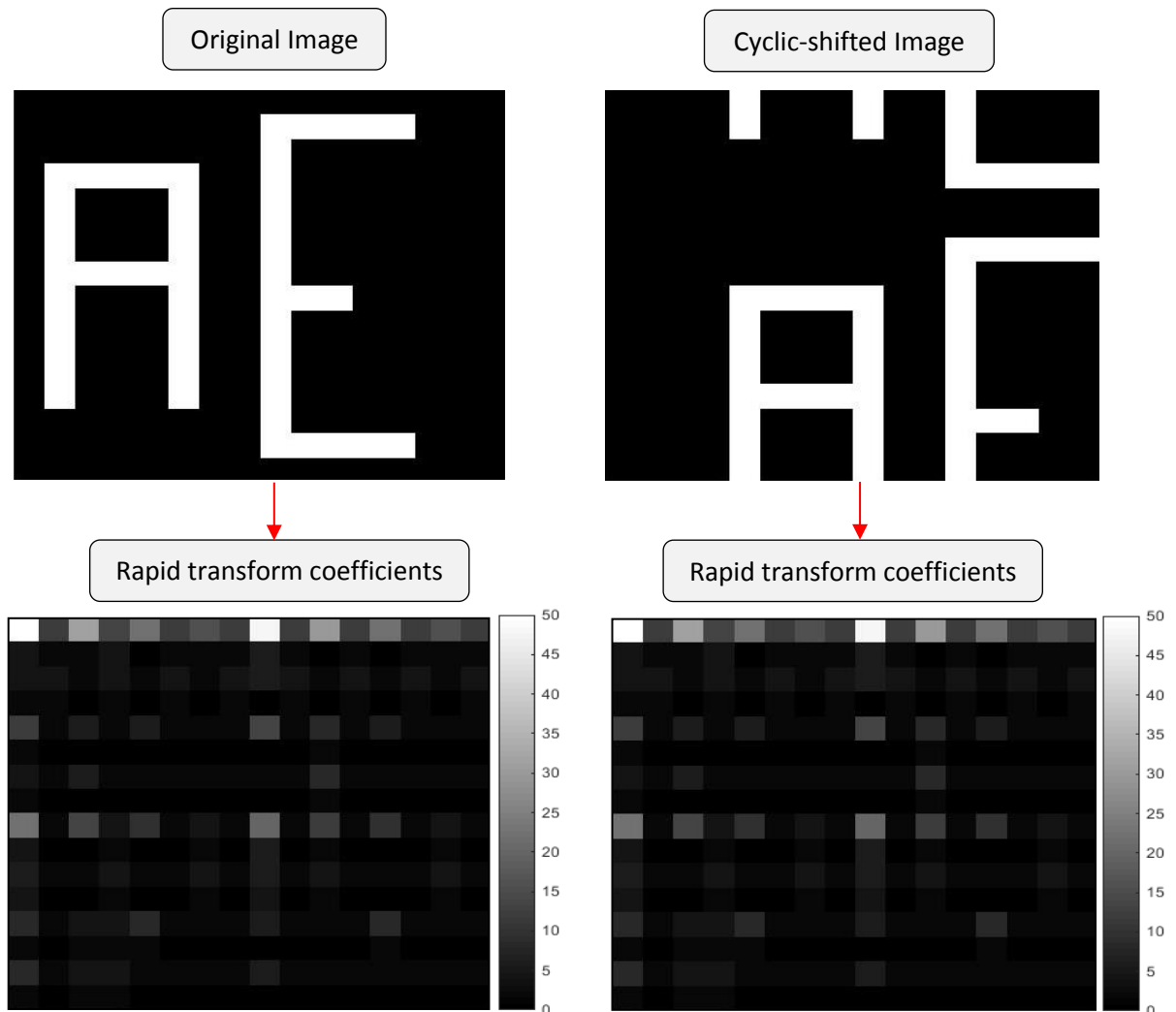


Figure B.1: Example showing rapid transform on a pair of synthetic images with translation differences. Top left: Original Image, Top right: Cyclic-shifted version of the original image. Bottom: Rapid transform coefficients indicating the similarity of the two images regardless of the cyclic shifts.