Department of Spatial Sciences

Automated Registration Of Unorganised Point Clouds From Terrestrial Laser Scanners

Kwang-Ho Bae

This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University of Technology

August 2006

Declaration

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgement has been made.

Signature : .....

Date : .....

To my good friends, Dong-Dong and Shortdari.

Abstract

Laser scanners provide a three-dimensional sampled representation of the surfaces of objects. The spatial resolution of the data is much higher than that of conventional surveying methods. The data collected from different locations of a laser scanner must be transformed into a common coordinate system. If good a priori alignment is provided and the point clouds share a large overlapping region, existing registration methods, such as the Iterative Closest Point (ICP) or Chen and Medioni's method, work well. In practical applications of laser scanners, partially overlapping and unorganised point clouds are provided without good initial alignment. In these cases, the existing registration methods are not appropriate since it becomes very difficult to find the correspondence of the point clouds.

A registration method, the Geometric Primitive ICP with the RANSAC (GP-ICPR), using geometric primitives, neighbourhood search, the positional uncertainty of laser scanners, and an outlier removal procedure is proposed in this thesis. The change of geometric curvature and approximate normal vector of the surface formed by a point and its neighbourhood are used for selecting the possible correspondences of point clouds. In addition, an explicit expression of the position uncertainty of measurement by laser scanners is presented in this dissertation and this position uncertainty is utilised to estimate the precision and accuracy of the estimated relative transformation parameters between point clouds.

The GP-ICPR was tested with both simulated data and datasets from close range and terrestrial laser scanners in terms of its precision, accuracy, and convergence region. It was shown that the GP-ICPR improved the precision of the estimated relative transformation parameters as much as a factor of 5. In addition, the rotational convergence region of the GP-ICPR on the order of 10°, which is much larger than the ICP or its variants, provides a window of opportunity to utilise this automated registration method in practical applications such as terrestrial surveying and deformation monitoring.

Acknowledgements

First of all, I would like to express my gratitude to my supervisor, Dr. Derek D. Lichti for his advice and guidance. Derek has also provided an excellent environment for me to pursue a Ph.D. degree. I am grateful to my associate supervisor, Associate Professor Michael P. Stewart, for his encouragement and proof-reading of this thesis.

I thank Dr. Stuart Gordon and Jochen Franke for their friendship. I extend my thanks to David Belton with whom I have had many interesting and valuable discussions. Lori Patterson is appreciated for reading a draft of this dissertation. In addition, I would like to thank all the members of the Western Australia Centre for Geodesy and Department of Spatial Sciences for their friendship and hospitality.

Curtin University of Technology and Department of Spatial Sciences are appreciated for providing the Curtin International Student Scholarship (CISS) and the John Szorenyi Scholarship. I thank the Australian Research Council (ARC) for supporting this project with a Discovery grant (DP0342887) and a Linkage-International grant (LX0347905). I also thank Dr. Maria Tsakiri and the National Technical University Athens for supporting the collection of the Agia Sanmarina church datasets in Greece.

I am grateful to Murray in the University of Adelaide for showing me the joy of pursuing science. I would like to thank my parents, Jae-Seong Bae and Young-Hee Lee, and my sister, In-Sook Bae. Last, but not least, I would like to express my great gratitude and love to my wife, Dong-Sook Kim.

Contents

List of Figures

List of Tables

Notation

- $\max(i, j)$ and $\min(i, j)$: the maximum and minimum between i and j.
- $\max(i)$ and $\min(i)$: the maximum and minimum of a variable, i.
- Normal and bold letters are scalars and vectors or matrices, respectively.
- $|a|$: the absolute value of a scalar a.
- $\mathbf{I}_n$: an unit matrix with $\text{rank}(\mathbf{I}_n) = n$.
- $\| \mathbf{A} \|$: the norm of a vector $\mathbf{A}$.
- $a^{-1} = \frac{1}{a}$
- $\mathbf{A}^{-1}$: the inverse of $\mathbf{A}$.
- $\mathbf{A}^{-}$: the Penrose-Moore generalised inverse.
- $(\mathbf{c}^T, 1)^T \in \Re^{4\times1}$: The homogeneous coordinate representation of a vector $\mathbf{c} \in \Re^{3\times1}$.
- $(\mathbf{A}, \mathbf{B})$: dot or inner product between $\mathbf{A}$ and $\mathbf{B}$.
- $\hat{a}$: an optimal estimate of a.
- $\hat{\mathbf{A}}$:
  1. an optimal estimate of $\mathbf{A}$.
  2. the unit vector of a vector $\mathbf{A}$.
- $\bigoplus$: the direct sum of matrices
- $\text{sgn}(a)$: the sign of a.
- $diag(\cdot)$: a diagonal matrix.
- $\mathbf{P}_{\hat{\mathbf{A}}} = \mathbf{I}_n - \hat{\mathbf{A}}\hat{\mathbf{A}}^T$: the projection operator of $\hat{\mathbf{A}}$ where $\text{rank}(\hat{\mathbf{A}}) = n$.

- $!p$: the inverse of a boolean operator p.
- $\nabla \equiv \left(\frac{\partial}{\partial x}\ \frac{\partial}{\partial y}\ \frac{\partial}{\partial z}\right)$: the gradient operator expressed in a three-dimensional Cartesian coordinate.
- D^{pp} : The distances between of points and their corresponding points.
- D^{pp}_{mean} and D^{pp}_{std} : The mean and standard deviation of D^{pp}.
- D^{ps} : The distances between of points and their corresponding surfaces.
- D^{ps}_{mean} and D^{ps}_{std} : The mean and standard deviation of D^{ps}.

Chapter 1

Introduction

Terrestrial laser scanners provide a three-dimensional sampled representation of the surfaces of objects resulting in a very large number of points. The spatial resolution of the data is much higher than that of conventional surveying methods. Since laser scanners have a limited line of sight, in order to obtain a complete representation of an object, it is necessary to collect data from several locations. These data must be transformed into a common coordinate system and this procedure is called the registration of point clouds. Existing registration methods, such as the Iterative Closest Point (ICP) by Besl and McKay (1992) or Chen and Medioni's (1992) method, work well if good a priori alignment is provided. However, in the case of the registration of partially overlapping and unorganised point clouds without good initial alignment, these methods are not appropriate since it becomes very difficult to find the correspondence of the point clouds.

In terms of input data, registration methods can be classified into two categories: one is the registration of two point clouds from different scanner locations, so-called pair-wise registration (Rusinkiewicz and Levoy, 2001), and the other is the simultaneous registration of multiple point clouds. This thesis aims to develop a method for the automated pair-wise registration of laser scanner data.

1.1 Review of previous work

A considerable amount of work for the automated registration of point clouds from range finders has been conducted over the last few decades by researchers from different fields such as computer vision, photogrammetry, and artificial intelligence. One can find reviews on existing registration methods from Haralick et al. (1989), Rusinkiewicz and Levoy (2001), Campbell and Flynn (2001), Rodrigues et al. (2002), and Gruen and Akca (2005). In particular, Gruen and Akca (2005) presented an excellent review and bibliography on available registration methods. Rodrigues et al. (2002) and

Rusinkiewicz and Levoy (2001) produced a good bibliography on finding corresponding points in an automated manner. Furthermore, Rodrigues et al. (2002) stated that no method can be clearly regarded as superior to the other.

Chen et al. (1999) classified the existing registration methods into two classes: iterative approaches (Besl and McKay, 1992; Chen and Medioni, 1992; Dorai et al., 1998; Turk and Levoy, 1994; Blais and Levine, 1995) and feature based methods (Stein and Medioni, 1992; Higuchi et al., 1995). They are based on the original ICP, although the latter focuses on finding a set of transformation-invariant features from geometric or radiometric information, e.g. geometric curvature or returning laser intensity.

The Iterative Closest Point algorithm is a reliable and popular idea for point cloud registration (Horn, 1987; Besl and McKay, 1992). Horn (1987) developed a closed-form adjustment method with quaternions in order to estimate unknown relative transformation parameters of two point clouds with knowledge of the correspondence of the point clouds. This method is called a point-to-point adjustment method, which is the same as a network adjustment method in the context of surveying (Wolf and Dewitt, 1999). Besl and McKay (1992) developed the ICP using Horn's (1987) algorithm in conjunction with a neighbourhood search algorithm. If a priori information about the point-to-point correspondence of two point clouds is provided, then the ICP can iteratively recover the relative transformation of the point clouds. It converges monotonically to a local minimum, which may or may not be the global minimum. However, in the ICP, all points in a point cloud are assumed to have corresponding points in the other cloud.

Basically, the original ICP is a general idea for the automated registration of point clouds using both an adjustment algorithm, e.g. Horn (1987), Umeyama (1991), or Wolf and Dewitt (1999), and a method for finding corresponding points. In the context of the original ICP, as indicated by its name, the closest point of a point in a point cloud in terms of Euclidean distance is assumed to be its corresponding point. However, a set of correct correspondences between point clouds can not be recovered in this manner since the closest point of a point based on Euclidean distance may not be its corresponding point (Eggert et al., 1998).

Chen and Medioni (1992) proposed an idea of minimising the distance between points and their corresponding surfaces rather than between corresponding point pairs as proposed by either Horn (1987) or Umeyama (1991). In addition, they proposed to find corresponding points by the direction of the estimated surface normal vector, so-called normal shooting, in conjunction with Euclidean distance. Chen and Medioni's (1992) method has been popular since there is a better chance in finding corresponding point-to-surface pairs than to discover corresponding point pairs (Rusinkiewicz and Levoy, 2001).

Zhang (1994) reported a variant of the ICP using several thresholds for the distance between a set of corresponding points in each iteration. Masuda and Yokoya (1995)

proposed a random sampling method for the ICP and introduced the least median of squares (LMedS) to the adjustment algorithm of the ICP. Turk and Levoy (1994) introduced a variant of the ICP for the registration of two meshes, i.e. triangulated point clouds, and named it a method for zipped polygon meshes.

Sharp et al. (2002) proposed a method based on Euclidean invariant features: curvature, second order moments, and spherical harmonics. Stamos and Leordeanu (2003) reported a registration method using extracted features, e.g. lines and planes from terrestrial laser scanner point clouds. Recently Dold (2005) proposed a method for initial alignment using Gaussian image and covariance analysis.

Following Rusinkiewicz and Levoy (2001), the ICP variants can be summarised as follows. First, these strategies of selecting sample points have been proposed.

- Using all available points from a point cloud (Besl and McKay, 1992).
- Uniform subsampling of the available points (Turk and Levoy, 1994).
- Random subsampling and using different points in each iteration (Masuda et al., 1996).
- Using the points with high intensity gradient around its neighbourhood (Weik, 1997).

In addition, the following ways have been proposed for finding possible corresponding points between two point clouds.

- Find the closest point in the other point cloud (Besl and McKay, 1992; Simon, 1996).
- Find the intersection point of the ray originated from a point to the direction of the surface normal vector of the point (Chen and Medioni, 1992).
- Project the points measured from a laser scanner into the other laser scanner coordinate (Blais and Levine, 1995; Neugebauer, 1997).
- Using a threshold value to avoid the incorrect selection of corresponding points, e.g. distance threshold (Zhang, 1994), angle between the surface normal vectors (Pulli, 1999), or difference in either intensity or color (Weik, 1997).

1.2 Objective of the dissertation

An automated method for the registration of partially overlapping point clouds from terrestrial laser scanners without good a priori alignment has still to be developed, mainly because it is very difficult to recover the correspondence between point clouds

without good a priori alignment. Three limitations of the existing techniques for the registration of partially overlapping point clouds, briefly reviewed in the previous section, can be quoted from Chen et al. (1999) as follows. First, the existing methods can not ensure a correct solution even for noiseless cases (Besl and McKay, 1992; Chen and Medioni, 1992; Dorai et al., 1998; Turk and Levoy, 1994). Second, the existing methods require a good initial estimate of the rigid transformation between the two datasets (Besl and McKay, 1992; Chen and Medioni, 1992). Third, they can only be used if the datasets contain sufficient local features (Stein and Medioni, 1992).

Another limitation of the existing methods is a size of the convergence region (Rodrigues et al., 2002). For example, if the rotational convergence region of a registration algorithm around an axis, e.g. x-axis, is 20° in the presence of a relative translation, then the registration algorithm can consistently find an optimal solution within this relative rotational range around the axis. A greater rotational or translational convergence is preferable but it is very difficult not only to achieve a large convergence region but also to evaluate the convergence region of a registration algorithm since the size of the convergence region of the registration algorithm depends on the method of sampling or finding correspondence.

In this thesis, a method for the registration of two partially overlapping point clouds, measured from different locations, is proposed using geometric primitives and neighbourhood search without good a priori alignment. In addition, the position uncertainty of laser scanners is explicitly expressed in this dissertation. This position uncertainty provides an optimal weight for an adjustment algorithm, a lower bound of the registration error between partially overlapping point clouds, and an evaluation of the accuracy of the solution.

The objectives of this dissertation are summarised as follows:

- The main objective of this thesis is to develop an automated registration algorithm for terrestrial laser scanner data and to overcome three limitations stated by Chen et al. (1999) and highlighted above, that is: inability to produce an evaluation of the accuracy of the estimated transformation, necessity of a good initial alignment, and requirement of sufficient local features, i.e. geometric primitives.

- An order of ten degrees rotational convergence region in case of a small relative translation error, which is large enough for a practical application of terrestrial or close-range laser scanners, is to be achieved by the proposed method.

- The proposed method is to be extensively tested with data from different kinds of laser scanners in terms of its precision, accuracy, and convergence region.

In addition, the scope of this thesis is limited by three constraints as follows:

- The proposed registration algorithm in this dissertation is for two partially overlapping point clouds, i.e. a pair-wise registration algorithm. In other words, the proposed registration method is not designed for multiple point cloud registration.
- The scale factor of the point clouds are assumed to be unity. Gruen and Akca (2005) judged this to be a reasonable assumption. However, a test for the estimation of the scale factor will be presented in Chapter 5.
- An analysis on the computational performance or efficiency of the proposed method, i.e. execution time, is not the objective of this thesis.

1.3 Organisation of the thesis

The characteristics of data from laser scanners and the Iterative Closest Point (ICP) are introduced in Chapter 2. The derivation of an explicit expression of the position uncertainty of laser scanner measurement is presented in Chapter 3 and it is frequently utilised throughout the thesis. Chapter 4 presents an automated registration method of two partially overlapping point clouds from laser scanners using geometric curvature and the results of Chapter 3. The thesis then presents extensive tests of the proposed method with both simulated data and real data from close-range and terrestrial laser scanners in terms of the precision, accuracy, and convergence region of the proposed registration method in Chapters 5, 6, and 7. Finally, conclusions and future directions are presented in Chapter 8.

Chapter 2

Background

In this section, some facts about laser scanner data will be briefly introduced. First the local spatial resolution of points from laser scanners will be discussed. Then, a description and the limitations of the original ICP algorithm (Besl and McKay, 1992) and its variants will be presented. One can find a good summary and discussion on the background of laser scanner technology in Gordon (2005, Chs. 2 - 3).

2.1 Unorganised point clouds from laser scanners

In the last few decades, there has been much development of two-dimensional image analysis in terms of research and practical applications (McGlone et al., 2004; Haralick and Shapiro, 1993). Airborne laser scanner datasets are generally referred as 2.5D point clouds since they can be treated as two dimensional data by projecting the datasets to a global plane such as ground (Maas and Vosselman, 1999). On the other hand, datasets from terrestrial laser scanners are usually called 3D point clouds. Many photogrammetric and computer graphical techniques and theories can be, with a small modification, applied to airborne laser scanner data (Maas, 2000; Pfeifer, 2005). In fact, in principle, any problem in three-dimensional data can be solved by a method for the two-dimensional equivalent problem. However, it is not practical to treat the measurement of terrestrial or close-range laser scanners as two-dimensional data since finding a good global surface, equivalent to the ground in the airborne case, for the projection of the data is not easy or sometimes is impossible. In addition, it is difficult to find the neighbourhood of a point that provides an acceptable level of precision or accuracy in the estimation of geometric properties such as surface normal vectors, mainly because a grid structure does not exist.

The local spatial resolution of points from laser scanners is not uniform over a point cloud as shown in Figure 2.1. In other words, a laser scanner provides a set of three-dimensional unorganised point clouds. When the laser beam divergence is relatively

small compared to the radial distance of an object from a laser scanner, as shown in Figure 2.1, the laser scanner has a spatial scanning interval, d. But the sampled positions of the surface of an object from the different locations of the laser scanner are different. In case of Figure 2.1(a), $\widehat{P_i^1 P_{i+1}^1}$ is invisible to the laser scanner. To overcome this local resolution variation in a point cloud, a data structure is required for finding a set of nearest neighbour points. One can find several candidates for the data structure such as the kd-tree and octree (Sedgewick, 1988; Samet, 1989, 1990).

The kd-tree data structure is utilised in the proposed method for the registration of point clouds from laser scanners in this thesis. For implementation, a kd-tree library developed by Arya et al. (1998) is used. In order to increase the efficiency of the kd-tree, one can modify the original kd-tree structure taking into account the local spatial resolution (Jost and Hügli, 2002). However, the original kd-tree structure is used in this thesis since it is more important with the scope of this dissertation to find reliable corresponding points than to increase efficiency of finding a set of neighbourhood points.

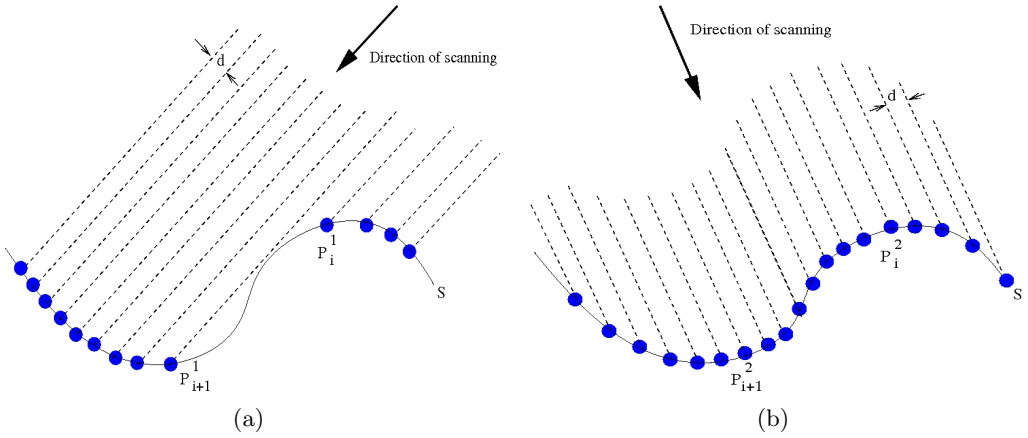


FIGURE 2.1: Local resolution variation on laser scanner data. S is the true surface of an object to be measured by a laser scanner. In the cases of (a) and (b), the object is measured by a laser scanner in different locations.

2.2 The Iterative Closest Point (ICP) algorithm

Suppose there are two point clouds, C^1 and C^2, and they are measured from different locations. A point cloud C^1 has a set of N_{C^1} points, $\{p_1^1, \cdots, p_{N_{C^1}}^1\}$. Bold and normal letters represent vectors and scalars, respectively. Let $||\mathbf{p}_i^1 - \mathbf{p}_j^2||$ be the Euclidean distance between point p_i^1 of point cloud C^1 and p_j^2 of C^2. Let $CP(p_i^1, C^2)$ be the corresponding point in C^2 of a point p_i^1. The ICP algorithm (Besl and McKay, 1992) is briefly described as follows.

1. Assume that the point of C^2 closest to a point of C^1 in terms of the Euclidean distance is the corresponding point.

2. Using a nearest neighbour search algorithm such as the kd-tree, find the correspondence of two point clouds, $\mathtt{C} = \cup_{i=1}^{N_{C^1}} \{T_{iter=k}(p_i^1), CP(T_{iter=k}(p_i^1), C^2)\}$, where $\mathtt{C}$ is the set of all pairs of corresponding points, $T_{iter=k}$ is the estimated transformation - usually rigid body - of the kth iteration, and $T_{iter=0}$ is an initial transformation.

3. Compute the new transformation $T_{iter=k+1}$ that minimises the sum of squared distances between the corresponding point pairs:

$$\sum_{i=1}^{n_{iter=k}} ||\mathbf{p}_i^1 - \boldsymbol{CP}(T_{iter=k}(p_i^1), C^2)||^2$$

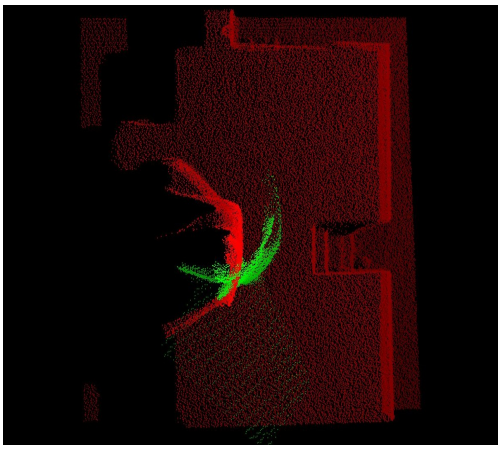
   where $n_{iter=k}$ is the number of selected samples in the kth iteration.

Another set of popular algorithms, named point-to-surface methods such as Chen and Medioni's (1992), minimises the sum of the square distances of a point to its corresponding surface. A mathematical derivation of Chen and Medioni's (1992) algorithm will be presented in Chapter 4. This algorithm is generally faster than the ICP since it deals with points and surfaces rather than point pairs. However, the point clouds need to be much more closely aligned to each other initially than with the ICP.

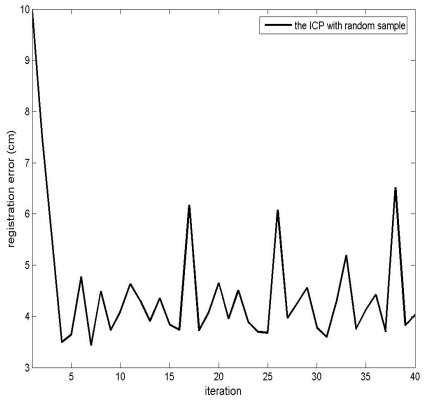
2.3 Limitation of the ICP and its variants

The ICP and its variants assume that the closest point in point cloud C^2 is a good estimate of the correct corresponding point to a point in C^1. If two point clouds are not approximately aligned using a priori information on the relative transformation of the point clouds, this assumption is not correct. Although the initial alignment can be provided from other means such as surveying of the laser scanner locations, it is not always possible or requires an extra cost. Finding corresponding points and a good estimation of the relative transformation of the point clouds is more difficult if the point clouds only partially overlap, since the original ICP was developed and tested with point clouds sharing approximately more than ninety percent overlapping region (Besl and McKay, 1992).

An example of the limitation of the ICP with random sampling by Masuda and Yokoya (1995) is presented in Figure 2.2 where Masuda and Yokoya (1995)'s method were tested with good a priori information on the relative transformation. Two datasets from the Ayuthaya Buddha statue in Thailand scanned by a Riegl LMS-Z210 (Riegl, 2006) were used for this test.



(a) The ICP with random sampling



(b)

FIGURE 2.2: Registration results of point clouds by the ICP with random sampling by Masuda and Yokoya (1995) and the proposed method in this dissertation. Note that a good initial alignment was provided using a priori information on the relative orientation between the point clouds.

Note that the sole purpose of this simple experiment was to present one of the main limitations of a variant of the ICP: convergence to a local minimum of the registration error. Therefore, the algorithm was terminated at the 40th iteration regardless of the resultant precision or an evaluation of the accuracy of the estimated transformation parameters. In addition, since the point clouds are initially very close in this test, it is expected for a registration algorithm to converge into a global minimum or possibly a set of transformation parameters which is close to the true transformation. However, the ICP with random sampling did not reach a global minimum as shown in Figure 2.2(b).

What causes this instability of a registration algorithm converging to a local minimum? One of the main problems of the original ICP and its variants is how to find the correspondence of point clouds. The Euclidean distance between possible corresponding points used by most of the ICP variants, is not good enough for finding the correct correspondence (Eggert et al., 1998). Therefore, a set of possible corresponding points in an iteration must be carefully selected with the notion that any geometric parameter for the selection of the corresponding points is an estimate.

Another limitation of the ICP and its variants is that they can not provide any rigorous evaluation of the precision and accuracy of the estimated relative transformation as stated in Chapter 1. This is the most important information in the applications of terrestrial laser scanner data in fields such as surveying, geomatics, and other spatial sciences. Therefore, this thesis presents a method overcoming the limitations of the original ICP and its variants for laser scanner data in geomatic and surveying applications.

Chapter 3

Position uncertainty : Significance and Derivation

In this chapter, an explicit expression for the approximate position uncertainty of laser scanner measurements will be formulated. This will be used in subsequent chapters to evaluate both a lower bound for the registration error of point clouds and an optimal size of neighbourhood for the estimation of surface normal vectors.

3.1 Significance

Position uncertainty of a point in an unorganised point cloud measured by a range sensor is an important quantity since it provides the confidence level of any parametric estimation such as the estimation of the surface normal vector and the registration of point clouds from laser scanners. In addition, an optimal weight for an estimation process can be found from the knowledge of the variance of each element in a dataset. However, in practice, evaluating position uncertainty from data is very difficult and more so is its derivation in an explicit form, mainly because position uncertainty depends on the design of the measurement system, i.e. laser and detector in a laser scanner. For example, the position uncertainties of a measured point from two different kinds of laser scanners may or may not be the same. It is analogous to measuring the metric length of an object with different kinds of rulers.

One of the important factors in position uncertainty is the incidence angle of the laser beam to the surface of objects. For example, the incidence angles of the laser beam from two different laser scanner locations are shown in Figures 3.1 and 3.2 for two different sites, respectively. The incidence angle of the laser beam is defined as the angle between the estimated surface normal vector and the line of sight of the laser beam. In Figure 3.1, the difference in incidence angle in the overlapping regions of the point clouds is about 40°. This large difference implies that position uncertainties

of those regions are likely to be quite different. This is an important factor which we need to take into account for either the estimation of the surface normal vector or the registration of point clouds. In Figure 3.2, a small incidence angle difference is observed in the overlapping region around the body of the Buddha statue. However, a larger incidence angle difference, about 40°, is observed in a flat curvature region and the magnitude of the position uncertainty in this region is expected to be greater than that of Figures 3.1(a) and Figures 3.1(b) since the radial and angular variance of the Riegl LMS-Z210 is much greater than the Leica HDS2500.

Some research about the estimation of position uncertainty in range sensors has been conducted, for example, registration of three-dimensional data (Johnson and Kang, 1997; Dorai et al., 1997; Pennec and Thirion, 1997; Stoddart et al., 1996), uncertainty of sampled three-dimensional surfaces (Tasdizen and Whitaker, 2003; Pauly et al., 2004; Whaite and Ferrie, 1991), and range error analysis for different kinds of laser scanners (Blais et al., 2000). Except for Tasdizen and Whitaker (2003) and Blais et al. (2000), the others regard the measurement of the position of a point using range sensors as a stochastic process with a three-dimensional, isotropic, and zero-mean Gaussian probability density function. Tasdizen and Whitaker (2003) found an explicit form of the Cramer-Rao lower bound (CRLB) of measurements by a range sensor with the assumption that the variance of the estimated geometric surface normal vector is small.

As presented in Figures 3.1 and 3.2, two point clouds of the same region of an object from different locations of the laser scanner may have a large difference in the incidence angle of the laser beam from the scanner. Although we do not have an explicit form of the variance of estimated surface normal vectors to be derived in Section 3.5, it is obvious that the variance of the estimated surface normal vector is dependent on the incidence angle of the laser beam. Therefore, it is expected to be related to the difference between the variance of estimated surface normal vectors in the point clouds of the same region of the object from different locations of the laser scanner. Furthermore, an explicit expression of the variance of laser scanner data is useful for an outlier procedure for a least-squares method since a set of the outliers must be removed by a criterion made by an independently derived or evaluated variance information on the data.

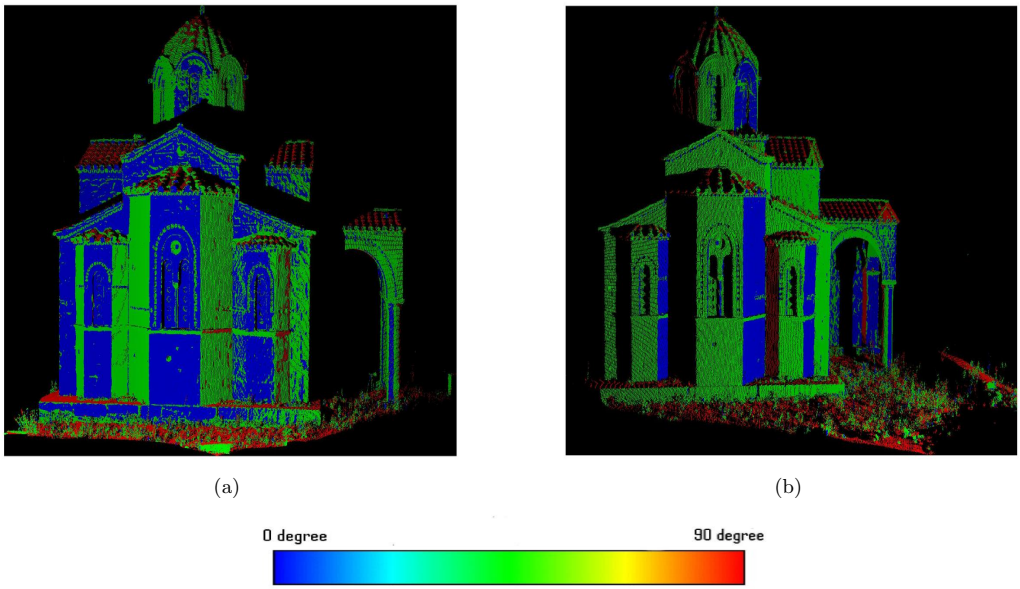


FIGURE 3.1: Colour maps of incidence angles of the Agia Sanmarina church in Greece scanned from two different locations using a Leica HDS2500. The radial and angular variances of the scanner are $\sigma_r^2 = (0.004)^2$ (m^2) and $\sigma_a^2 = (6 \times 10^{-5})^2$ (rad^2), respectively. The number of points for (a) and (b) are 486340 and 453142, respectively. The radial distance between the church and the scanner is approximately 20m in both cases. The dimension of the church is approximately $(L, W, H) = (25.0\mathrm{m}, 15.0\mathrm{m}, 10.0\mathrm{m})$ where L, W, and H are the length, width, and height, respectively, of the object.

3.2 Overview

In this chapter, several coordinate systems related to laser scanners will be first introduced. Using these coordinate systems and linear algebra, a new coordinate system named the uvz coordinate system* will be formulated in order to make it much simpler to transform matrices or vectors in one coordinate system to an other. An explicit form of position uncertainty or variance in unorganised point clouds from laser scanners will be derived.

3.3 Coordinate systems

Let us introduce four coordinate systems: the scanner coordinate system ($\mathbf{O}_{scanner}$), the laser beam coordinate system ($\mathbf{O}_{laser}$), the measured point coordinate system ($\mathbf{O}_{point}$) and the uvz coordinate system ($\mathbf{O}_{uvz}$) as shown in Figure 3.3. A laser scanner measures

*Tasdizen and Whitaker (2003) first used the uvz coordinate system with a less complete derivation. In this dissertation, complete mathematical derivations such as the orthogonality of the basis vectors of the uvz coordinate system will be presented.

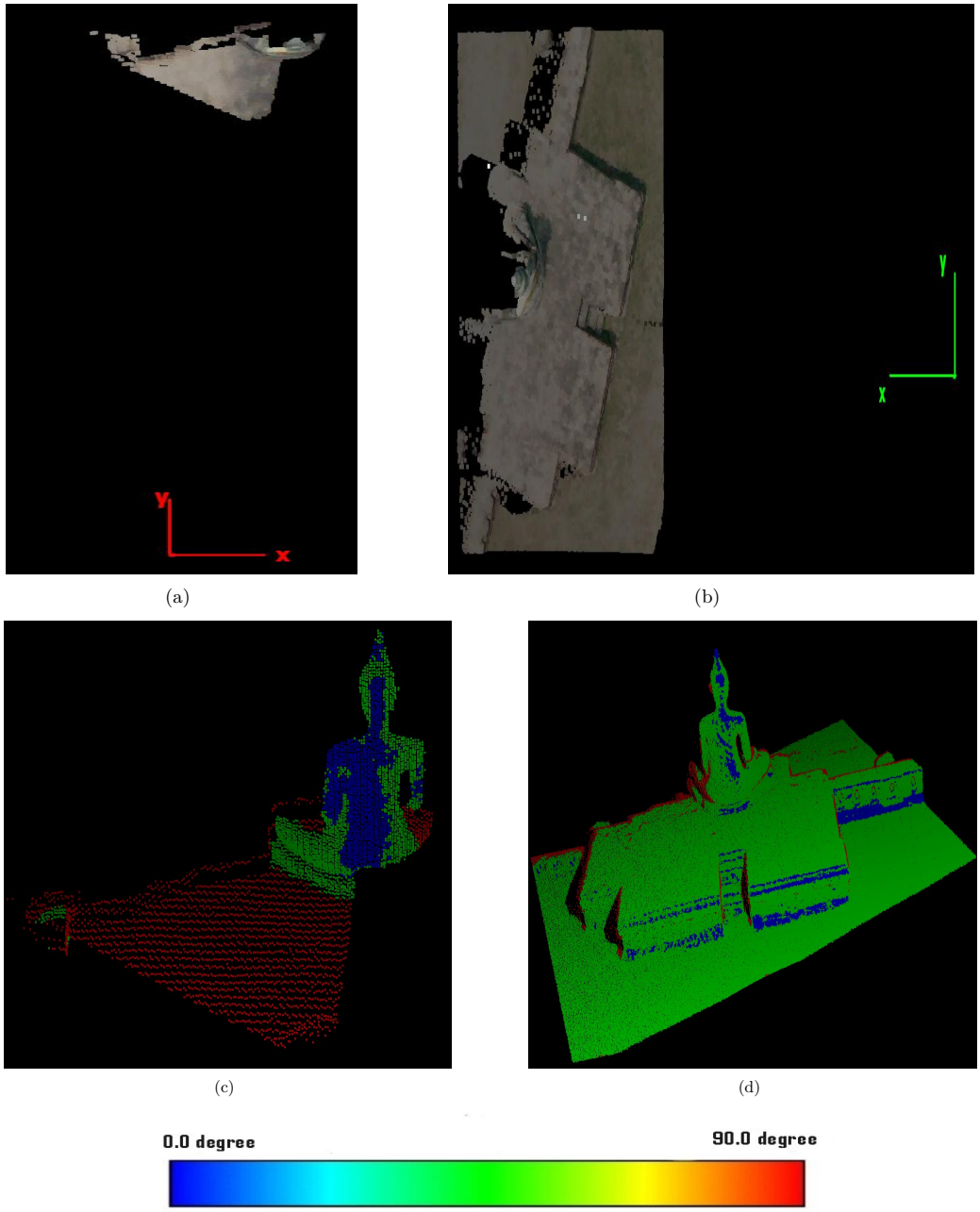


FIGURE 3.2: Colour maps of incidence angles of the Ayuthaya Buddha statue in Thailand are presented in (c) and (d). The point clouds were scanned from different locations using a Riegl LMS-Z210. The radial and angular variances of the scanner are $\sigma_r^2 = (0.025)^2$ (m^2) and $\sigma_a^2 = (4.7 \times 10^{-4})^2$ (rad^2), respectively. The locations of the scanners for each point cloud are presented in (a) and (b), respectively. The numbers of the points for (a) and (b) are 5640 and 149987, respectively. The radial distances between the Buddha and the scanner are approximately 25m and 15m for (a) and (b), respectively. The dimension of the Buddha statue is $(L, W, H) \simeq$ (6.0m, 8.0m, 4.0m).

the radial distance in $\mathbf{O}_{laser}$ with fixed horizontal and vertical angular intervals, i.e. two angular components of a spherical coordinate system. The measured position recorded in the spherical coordinate system by a laser scanner is transformed to a Cartesian coordinate system, i.e. $\mathbf{O}_{scanner}$. In addition, the measured point coordinate system $\mathbf{O}_{point}$ is frequently used for calculation of the local orthogonal basis using covariance analysis (Linsen, 2001; Pauly et al., 2004; Bae and Lichti, 2004; Mitra et al., 2004) since the scanner coordinate system $\mathbf{O}_{scanner}$ is not always convenient for the estimation of geometric parameters. For example, $\mathbf{O}_{point}$ is a better choice for the estimation of geometric surface normal or tangential vectors, which are the most important quantities for registration of point clouds or automated classification of geometric features in point clouds. The relationships between these coordinate systems are simply given as

$$\mathbf{O}_{scanner} = \mathbf{R}_{laser}^{scanner}\mathbf{O}_{laser} + \mathbf{Tr}_{laser}^{scanner} = \mathbf{R}_{point}^{scanner}\mathbf{O}_{point} + \mathbf{Tr}_{point}^{scanner}, \tag{3.1}$$

where $\mathbf{R}_b^a$ is the relative rotation matrix from b to a and $\mathbf{Tr}_b^a$ is the relative translation between a and b. $\mathbf{R}_{laser}^{scanner}$ and $\mathbf{Tr}_{laser}^{scanner}$ are dependent on the hardware design of a laser scanner and can possibly be provided by the manufacturer. In addition, using the eigenvectors of the covariance matrix of a point, $\mathbf{R}_{point}^{scanner}$ can be expressed as

$$\mathbf{R}_{point}^{scanner} = \begin{pmatrix} \hat{\mathbf{e}}_0 & \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 \end{pmatrix}^T = \begin{pmatrix} \hat{\mathbf{e}}_0^T \\ \hat{\mathbf{e}}_1^T \\ \hat{\mathbf{e}}_2^T \end{pmatrix}, \tag{3.2}$$

where $\hat{\mathbf{e}}_i$ is the eigenvector of the $(i+1)$th smallest eigenvalue of the covariance matrix and $\hat{\mathbf{e}}_i \in \Re^{3\times 1}$ (McGlone et al., 2004, pp. 434). Note that $\mathbf{R}_{point}^{scanner}(\mathbf{R}_{point}^{scanner})^T = \mathbf{I}_3$ since $(\hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) = \delta_{ij}$ where δ_{ij} is the Kronecker-delta symbol† and $\mathbf{r}$ is the position vector of a point. Let $\mathbf{V}(\mathbf{r})_{scanner}$, $\mathbf{V}(\mathbf{r})_{laser}$ and $\mathbf{V}(\mathbf{r})_{point}$ be the covariance matrices of a point measured by a laser scanner in terms of $\mathbf{O}_{scanner}$, $\mathbf{O}_{laser}$, and $\mathbf{O}_{point}$ coordinate systems, respectively. Then $\mathbf{V}(\mathbf{r})_{scanner}$ and $\mathbf{V}(\mathbf{r})_{point}$ is written as

$$\mathbf{V}(\mathbf{r})_{scanner} = \mathbf{R}_{point}^{scanner}\mathbf{V}(\mathbf{r})_{point}(\mathbf{R}_{point}^{scanner})^T \tag{3.3}$$

and

$$\begin{aligned}\mathbf{V}(\mathbf{r})_{point} &= (\mathbf{R}_{point}^{scanner})^T\mathbf{V}(\mathbf{r})_{scanner}\mathbf{R}_{point}^{scanner} \\ &= \mathbf{R}_{scanner}^{point}\mathbf{V}(\mathbf{r})_{scanner}(\mathbf{R}_{scanner}^{point})^T,\end{aligned}$$

† $\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$

respectively. Therefore, if the position uncertainty of a laser scanner is defined in any of these three coordinate systems, $\mathbf{O}_{scanner}$, $\mathbf{O}_{laser}$, and $\mathbf{O}_{point}$, then its expression in the other coordinate systems can be also derived. If $\mathbf{V}(\mathbf{r})_{laser}$ is known, there are two ways to obtain the variance matrices in the other coordinate systems for $\mathbf{V}_{laser}$. One is to first calculate $\mathbf{V}(\mathbf{r})_{scanner}$ using $\mathbf{R}_{laser}^{scanner}$ and then evaluate $\mathbf{V}(\mathbf{r})_{point}$ by $\mathbf{R}_{scanner}^{point}$. The other way is to first calculate $\mathbf{V}(\mathbf{r})_{point}$ and then go to $\mathbf{V}(\mathbf{r})_{scanner}$ later. The latter was chosen since knowledge of $\mathbf{R}_{laser}^{scanner}$ is necessary for the first method. Furthermore, the latter makes it possible to obtain $\mathbf{V}(\mathbf{r})_{point}$ as a function of the angles between the line of sight of laser and the axes of $\mathbf{O}_{point}$, i.e. θ, α, and β as shown in Figure 3.3 with the assumption that $\mathbf{Tr}_{laser}^{scanner}$ is small enough to be $\mathbf{r}_{laser} \simeq \mathbf{r}_{scanner}$ where $\mathbf{r}_{laser}$ and $\mathbf{r}_{scanner}$ are the radial distances in $\mathbf{O}_{laser}$ and $\mathbf{O}_{scanner}$, respectively.

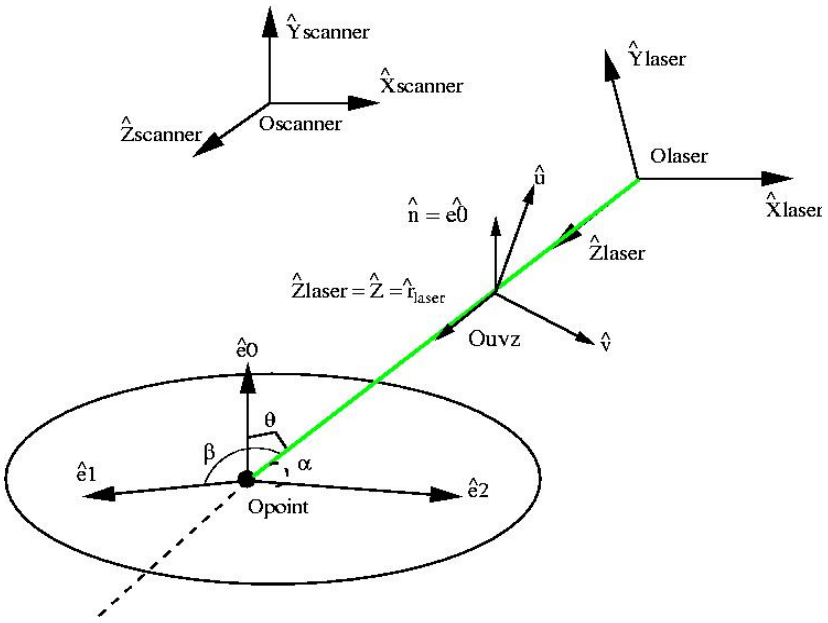


FIGURE 3.3: θ, α, and β are the direction cosine angles of $\hat{\mathbf{z}}_{laser}$ relative to the local surface of an object in the scene. Note that $\hat{\mathbf{r}}_{laser} = \hat{\mathbf{z}} = \hat{\mathbf{z}}_{laser}$ where $\hat{\mathbf{r}}_{laser}$ is a radial distance in $\mathbf{O}_{laser}$. In addition, note that $\hat{\mathbf{e}}_0$ is used as the x axis of $\mathbf{O}_{point}$.

3.4 uvz coordinate system

In this section a Cartesian coordinate system constructed by both the surface normal vector and $\hat{\mathbf{z}}_{laser}$, the line of sight of a scanner, as shown in Figure 3.3 will be introduced. It is named the uvz coordinate system that Tasdizen and Whitaker (2003) also used for the evaluation of the Cramer-Rao lower bound of measurement by laser scanners. This coordinate system is important since it acts as a bridge between $\mathbf{O}_{laser}$ and $\mathbf{O}_{point}$ as illustrated in Figure 3.3. Any vector or matrix in $\mathbf{O}_{laser}$ can be approximately expressed

in the uvz coordinate system without knowledge of the rotation between $\mathbf{O}_{laser}$ and $\mathbf{O}_{uvz}$, i.e. $\mathbf{R}_{laser}^{uvz}$. For example, suppose that the components of a matrix $\mathbf{A}$ in the direction of $\hat{\mathbf{x}}_{laser}$ and $\hat{\mathbf{y}}_{laser}$ are the same, i.e. $(\hat{\mathbf{x}}, \mathbf{A}) = (\hat{\mathbf{y}}, \mathbf{A})$. Let $\hat{\mathbf{z}}_{laser}$ and $\hat{\mathbf{e}}_0$ be $\hat{\mathbf{z}}$ and $\hat{\mathbf{n}}$, respectively. Then the unit vectors of the uvz coordinate system are expressed as follows:

$$\begin{aligned} \hat{\mathbf{v}} &= \hat{\mathbf{n}} \times \hat{\mathbf{z}} \\ \hat{\mathbf{u}} &= \hat{\mathbf{z}} \times \hat{\mathbf{v}} \end{aligned} \tag{3.4}$$

and

$$\begin{aligned} \hat{\mathbf{x}}_{\mathbf{laser}} \times \hat{\mathbf{v}} &= -\hat{\mathbf{z}}(\hat{\mathbf{x}}_{\mathbf{laser}}, \hat{\mathbf{n}}) \\ \hat{\mathbf{y}}_{\mathbf{laser}} \times \hat{\mathbf{v}} &= -\hat{\mathbf{z}}(\hat{\mathbf{y}}_{\mathbf{laser}}, \hat{\mathbf{n}}) \\ \hat{\mathbf{x}}_{\mathbf{laser}} \times \hat{\mathbf{u}} &= \hat{\mathbf{z}}(\hat{\mathbf{x}}_{\mathbf{laser}}, \hat{\mathbf{v}}) \\ \hat{\mathbf{y}}_{\mathbf{laser}} \times \hat{\mathbf{u}} &= \hat{\mathbf{z}}(\hat{\mathbf{y}}_{\mathbf{laser}}, \hat{\mathbf{v}}) \end{aligned} \tag{3.5}$$

using $\mathbf{A} \times (\mathbf{B} \times \mathbf{C}) = \mathbf{B}(\mathbf{A}, \mathbf{C}) - \mathbf{C}(\mathbf{A}, \mathbf{B})$ with vectors $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$. This shows that two planes whose tangential vectors are $\{\hat{\mathbf{u}}, \hat{\mathbf{v}}\}$ and $\{\hat{\mathbf{x}}_{laser}, \hat{\mathbf{y}}_{laser}\}$, respectively, are parallel. Then the transformed form of $\mathbf{A}$ in the uvz coordinate system $\mathbf{A}^{uvz}$ also has the same components in the direction of $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$, i.e. $(\hat{\mathbf{u}}, \mathbf{A}^{uvz}) = (\hat{\mathbf{v}}, \mathbf{A}^{uvz})$, which is proven from Eqs. 3.5. The main purpose of introducing the uvz coordinate system is to make it simple to transform quantities in $\mathbf{O}_{point}$ to that in $\mathbf{O}_{laser}$, or the reverse. The relationships between $\mathbf{O}_{laser}$, $\mathbf{O}_{point}$, and $\mathbf{O}_{scanner}$ are simply given as

$$\begin{aligned} \mathbf{O}_{\mathbf{laser}} &\xrightarrow{\mathbf{uvz}} \mathbf{O}_{\mathbf{point}} \xrightarrow{\mathbf{R}_{\mathbf{point}}^{\mathbf{scanner}}} \mathbf{O}_{\mathbf{scanner}} \\ \mathbf{O}_{\mathbf{laser}} &\xleftarrow{\mathbf{uvz}} \mathbf{O}_{\mathbf{point}} \xleftarrow{(\mathbf{R}_{\mathbf{point}}^{\mathbf{scanner}})^{-1}} \mathbf{O}_{\mathbf{scanner}}. \end{aligned}$$

These basic relations between the axes of the uvz coordinate system will be used throughout the chapter. Let $\hat{\mathbf{e}}_i^{uvz}$ be the counterpart of $\hat{\mathbf{e}}_{i=0..2}$ in the uvz coordinate system. In addition, let θ, α, and β in Figure 3.3 be the angles of the direction cosines of $\hat{\mathbf{z}}_{laser}$ to the local surface in $\mathbf{O}_{point}$ so they have a simple relationship (Arfken, 1995, Ch. 3):

$$\cos^2\theta + \cos^2\alpha + \cos^2\beta = 1. \tag{3.6}$$

Using the dot products between the eigenvectors of the covariance matrix and each axis of the uvz coordinate system, $(\hat{\mathbf{e}}_0, \hat{\mathbf{u}})$ is expressed as follows:

$$\begin{aligned} (\hat{\mathbf{e}}_0, \hat{\mathbf{u}}) &= (\hat{\mathbf{e}}_0, \frac{\hat{\mathbf{z}} \times \hat{\mathbf{v}}}{|\hat{\mathbf{z}} \times \hat{\mathbf{v}}|}) = (\hat{\mathbf{z}}, \frac{\hat{\mathbf{v}} \times \hat{\mathbf{e}}_0}{|\hat{\mathbf{v}} \times \hat{\mathbf{e}}_0|}) \\ &= (\hat{\mathbf{z}}, \frac{1}{|\sin\theta|}[\cos\theta\hat{\mathbf{e}}_0 + \hat{\mathbf{z}}]) \\ &= \frac{1}{|\sin\theta|}\left(1 - \cos^2\theta\right) = |\sin\theta| \end{aligned} \tag{3.7}$$

since

$$\begin{aligned}\hat{\mathbf{e}}_0 \times \hat{\mathbf{v}} &= \hat{\mathbf{e}}_0 \times (\hat{\mathbf{e}}_0 \times \hat{\mathbf{z}}) \\ &= \hat{\mathbf{e}}_0(\hat{\mathbf{e}}_0, \hat{\mathbf{z}}) - \hat{\mathbf{z}}(\hat{\mathbf{e}}_0, \hat{\mathbf{e}}_0) \\ &= -\cos\theta\hat{\mathbf{e}}_0 - \hat{\mathbf{z}} \end{aligned} \tag{3.8}$$

and $(\mathbf{A}, \mathbf{B} \times \mathbf{C}) = (\mathbf{B}, \mathbf{C} \times \mathbf{A})$ with vectors $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$. In a similar manner, one can calculate $(\hat{\mathbf{e}}_{i=1\ldots2}, \hat{\mathbf{u}})$ as follows:

$$(\hat{\mathbf{e}}_1, \hat{\mathbf{u}}) = \frac{\cos\alpha}{|\cos\alpha|}\cos\theta = \operatorname{sgn}(\cos\alpha)\cos\theta \tag{3.9}$$

$$(\hat{\mathbf{e}}_2, \hat{\mathbf{u}}) = -\frac{\cos\beta}{|\cos\beta|}\cos\theta = -\operatorname{sgn}(\cos\beta)\cos\theta \tag{3.10}$$

where $\operatorname{sgn}(a)$ is the sign of a. Note that $\operatorname{sgn}(0)$ is defined to be zero, regardless of its mathematical incorrectness, to make sure that $(\hat{\mathbf{e}}_{i=1\ldots2}, \hat{\mathbf{u}}) = 0$ with $\theta = 0°$, in which the line of sight of the laser beam is exactly aligned with the estimated surface normal vector. It must be also noted that, in the case of $\theta = 0°$, $\mathbf{R}_{uvz}^{point} = \mathbf{I}_3$. Using these results, $\hat{\mathbf{e}}_{i=0}^{uvz}$ is expressed as

$$\hat{\mathbf{e}}_0^{uvz} = \begin{pmatrix}(\hat{\mathbf{e}}_0, \hat{\mathbf{u}}) \\ (\hat{\mathbf{e}}_0, \hat{\mathbf{v}}) \\ (\hat{\mathbf{e}}_0, \hat{\mathbf{z}})\end{pmatrix} = \begin{pmatrix}|\sin\theta| \\ 0 \\ -\cos\theta\end{pmatrix}, \tag{3.11}$$

and in a similar manner,

$$\hat{\mathbf{e}}_1^{uvz} = \begin{pmatrix}-\operatorname{sgn}(\cos\alpha)\cos\theta \\ -\cos\beta \\ -\cos\alpha\end{pmatrix} \tag{3.12}$$

and

$$\hat{\mathbf{e}}_2^{uvz} = \begin{pmatrix}-\operatorname{sgn}(\cos\beta)\cos\theta \\ \cos\alpha \\ -\cos\beta\end{pmatrix}. \tag{3.13}$$

It is observed that all of $\hat{\mathbf{e}}_{i=0\ldots2}^{uvz}$ satisfy the normality condition for the basis of a coordinate system. However, unlike $\mathbf{R}_{point}^{scanner}$ in Eq. 3.2, $\mathbf{R}_{uvz}^{point}$ does not automatically satisfy the orthogonal condition since the uvz coordinate system is dependent on $\hat{\mathbf{e}}_0$. Without the orthogonal condition of $\hat{\mathbf{e}}_{i=0\ldots2}^{uvz}$, $\mathbf{R}_{uvz}^{point}$ is approximately expressed as

$$\mathbf{R}_{uvz}^{point} \simeq \begin{pmatrix}\hat{\mathbf{e}}_0^{uvz} & \hat{\mathbf{e}}_1^{uvz} & \hat{\mathbf{e}}_2^{uvz}\end{pmatrix}^T \tag{3.14}$$

where $\mathbf{A} \simeq \mathbf{B}$ means $\parallel \mathbf{A} - \mathbf{B} \parallel \simeq 0$ with vectors $\mathbf{A}$ and $\mathbf{B}$.

To investigate the orthogonality of $\hat{\mathbf{e}}^{uvz}_{i=0\ldots2}$, the dot products of these eigenvectors in the uvz coordinate can be calculated as follows:

$$\begin{aligned}(\hat{\mathbf{e}}^{uvz}_0, \hat{\mathbf{e}}^{uvz}_1) &= -|\sin\theta|\,\mathrm{sgn}\,(\cos\alpha)\cos\theta + \cos\alpha\cos\theta \\ &= \cos\alpha\cos\theta\left[-\frac{|\sin\theta|}{|\cos\alpha|} + 1\right], \\ (\hat{\mathbf{e}}^{uvz}_0, \hat{\mathbf{e}}^{uvz}_2) &= \cos\theta\left[-|\sin\theta|\,\mathrm{sgn}\,(\cos\beta) + \cos\beta\right],\end{aligned}$$

and

$$(\hat{\mathbf{e}}^{uvz}_1, \hat{\mathbf{e}}^{uvz}_2) = \mathrm{sgn}\,(\cos\alpha)\,\mathrm{sgn}\,(\cos\beta)\cos^2\theta.$$

The orthogonality between $\hat{\mathbf{e}}^{uvz}_{i=0..2}$ is always preserved by rotating both $\hat{\mathbf{e}}^{uvz}_1$ around $\hat{\mathbf{e}}^{uvz}_0$ until α satisfies the equation:

$$\begin{aligned}\cos\alpha &= \sin\theta \\ \therefore \alpha &= \frac{\pi}{2} - \theta\end{aligned}$$

and this guarantees $(\hat{\mathbf{e}}^{uvz}_0, \hat{\mathbf{e}}^{uvz}_1) = 0$. Furthermore, it also assures $(\hat{\mathbf{e}}^{uvz}_0, \hat{\mathbf{e}}^{uvz}_2) = (\hat{\mathbf{e}}^{uvz}_1, \hat{\mathbf{e}}^{uvz}_2) = 0$ using the fact of $\cos\beta = 0$ from Eq. 3.6. In addition, Eq. 3.14 will be numerically proven to be acceptable in Section 3.5.2 by estimating the position uncertainty in the scanner coordinate system, i.e. in a conventional Cartesian coordinate system.

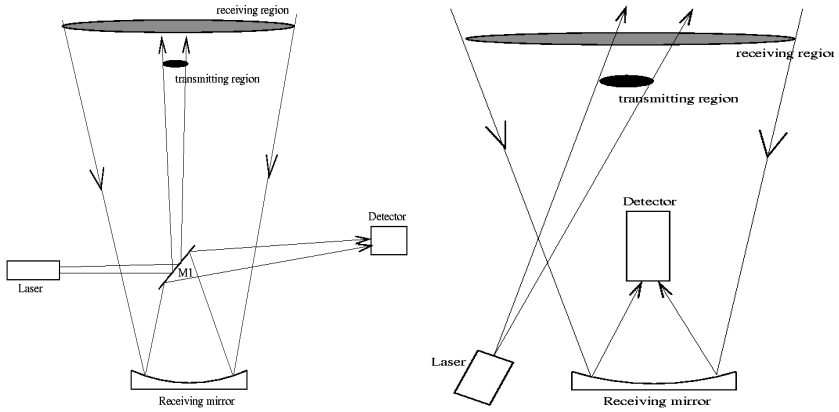
3.5 Position uncertainty

A laser scanner is a kind of LIDAR (LIght Detection And Ranging) system and can be classified as either a monostatic or a bistatic system according to the locations of the laser scanner transmitter, i.e. laser, and receiver (Devara, 1998). Monostatic lidar systems have their transmitters and detectors in the almost same location and can be divided into two different sub-classes, co-axial and bi-axial, as shown in Figure 3.4. A near-monostatic lidar is defined as a system which is close to a monostatic system and such that the assumption of $\mathbf{r}_{laser} \simeq \mathbf{r}_{scanner}$ is valid.

Let $diag(\cdot)$ represent a diagonal matrix. In near-monostatic laser scanners, the position uncertainty of laser scanner measurement in the $\mathbf{O}_{laser}$ is approximately expressed as

$$\mathbf{V}(\mathbf{r})_{laser} \simeq diag(r^2\sigma_a^2,\ r^2\sigma_a^2,\ \sigma_r^2) \tag{3.15}$$

where σ_a^2 and σ_r^2 are the angular and radial uncertainties of the laser scanner, respec-



(a) Co-axial monostatic system (b) Bi-axial monostatic system

FIGURE 3.4: Two kinds of monostatic lidar systems. M1 is a double-sided mirror and solid lines represent transmitting and receiving laser beams.

tively, their units being radians squared and distance squared, respectively. In addition, r is the radial distance of a point from a laser scanner.

Except for a few close-range scanners, many laser scanners are monostatic or at least near-monostatic, in which Eq. 3.15 can be utilised to describe the position uncertainty of measurement. The transmitting and receiving regions of a lidar system including a laser scanner are graphically defined in Figure 3.4. Since the absolute size of either the transmitting or receiving region is meaningless, the divergence angle, an angular measurement of its divergence, is utilised. The size of transmitting region depends on the divergence angle of the laser beam and that of the receiving region is decided by the size and optics of the detector. In practice, the receiving region is larger than the transmitting region so it is reasonable to use the divergence angle of the receiving region as σ_a. In addition, the angular uncertainty of the laser scanner is proportional to the ratio between the divergence angles of transmitting and receiving parts, i.e. the ratio of the sizes of two regions (Blais et al., 2000). Radial uncertainty is how well the system can detect the returning signal and depends on the quality of the detector and optics within the laser scanner. Therefore, radial uncertainty may be dependent on the radial distance of a point from the laser scanner. Since it is not the goal of this dissertation to investigate the relationship between the radial uncertainty, σ_r, and $\mathbf{r}$, the receiving region of a laser scanner is assumed to be a circle whose radius is $r\sigma_a$.

From Eq. 3.5 and the fact that $\mathbf{V}(\mathbf{r})_{laser}$ is assumed to be symmetric in the directions of $\hat{\mathbf{x}}_{laser}$ and $\hat{\mathbf{y}}_{laser}$, i.e. Eq. 3.15, the relationship between the variance matrices in the uvz coordinate system and $\mathbf{O}_{laser}$ is given as

$$\mathbf{V}(\mathbf{r})_{uvz} = \mathbf{V}(\mathbf{r})_{laser}.$$

Bearing in mind that the final goal of this chapter is to transform $\mathbf{V}(\mathbf{r})_{laser}$ into $\mathbf{O}_{point}$,

let $\mathbf{V}(\mathbf{r})_{point}$ be the transformed $\mathbf{V}(\mathbf{r})_{laser}$ in $\mathbf{O}_{point}$. Recalling that $\hat{\mathbf{e}}_{i=0...2}$ is the orthogonal basis of $\mathbf{O}_{point}$ and $\hat{\mathbf{e}}^{uvz}_{i=0...2}$ is $\hat{\mathbf{e}}_{i=0...2}$ transformed to the uvz coordinate system, then $\mathbf{V}(\mathbf{r})_{point}$ is expressed as

$$\begin{aligned}
\mathbf{V}(\mathbf{r})_{point} &= \mathbf{R}^{point}_{uvz}\mathbf{R}^{uvz}_{laser}\,\mathbf{V}(\mathbf{r})_{laser}\,(\mathbf{R}^{point}_{uvz}\mathbf{R}^{uvz}_{laser})^T \\
&= \mathbf{R}^{point}_{uvz}\left[\mathbf{R}^{uvz}_{laser}\,\mathbf{V}(\mathbf{r})_{laser}\,(\mathbf{R}^{uvz}_{laser})^T\right](\mathbf{R}^{point}_{uvz})^T \\
&= \mathbf{R}^{point}_{uvz}\,\mathbf{V}(\mathbf{r})_{uvz}(\mathbf{R}^{point}_{uvz})^T \\
&\simeq \begin{pmatrix} \hat{\mathbf{e}}^{uvz}_0 & \hat{\mathbf{e}}^{uvz}_1 & \hat{\mathbf{e}}^{uvz}_2 \end{pmatrix}^T \mathbf{V}(\mathbf{r})_{uvz} \begin{pmatrix} \hat{\mathbf{e}}^{uvz}_0 & \hat{\mathbf{e}}^{uvz}_1 & \hat{\mathbf{e}}^{uvz}_2 \end{pmatrix}
\end{aligned} \tag{3.16}$$

with the assumption that the variances of $\hat{\mathbf{e}}^{uvz}_{i=0...2}$ are small. Let us take into account the variance of $\hat{\mathbf{e}}^{uvz}_0$ and let $V(r)^{ij}_{point}$ be the ith row and jth column of $\mathbf{V}(\mathbf{r})_{point}$. Bearing in mind that $\mathbf{V}(\mathbf{r})_{laser} = \mathbf{V}(\mathbf{r})_{uvz}$, then $V(r)^{ij}_{point}$ is expressed as

$$\begin{aligned}
V(r)^{11}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_0, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_0) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r}) \\
V(r)^{22}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_1, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_1) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_1)\mathbf{r}) \\
V(r)^{33}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_2, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_2) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_2)\mathbf{r}) \\
V(r)^{12}_{point} = V(r)^{21}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_0, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_1) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r}) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_1)\mathbf{r}) \\
V(r)^{13}_{point} = V(r)^{31}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_0, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_2) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r}) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_2)\mathbf{r}) \\
V(r)^{23}_{point} = V(r)^{32}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_1, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_2) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_1)\mathbf{r}) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_2)\mathbf{r}),
\end{aligned}$$

or simply

$$V(r)^{ij}_{point} \simeq (\hat{\mathbf{e}}^{uvz}_{i-1}, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_{j-1}) + \sum_{k=min(i,j)}^{max(i,j)} \left(\frac{1}{2}\right)^{\delta_{ij}} (\delta_{ik} + \delta_{jk})(\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_{k-1})\mathbf{r}) \tag{3.17}$$

where $\max(i,j)$ and $\min(i,j)$ are the maximum and minimum of i and j. Note that the symbol of approximation is used in the above equation for the case that Eq. 3.14 is utilised without rotating $\mathbf{e}^{uvz}_1$ around $\mathbf{e}^{uvz}_0$ in order to achieve the orthogonality of $\mathbf{R}^{point}_{uvz}$ described at the end of Section 3.4. In Eq. 3.17, the first term represents the component of $\mathbf{V}(\mathbf{r})_{laser}$ in the direction of $\hat{\mathbf{e}}_i$ and the second represents the component of $\mathbf{V}(\hat{\mathbf{e}}_i)$ to the direction of $\hat{\mathbf{r}}$. If the variance of the estimated vector is zero, so is the second term of Eq. 3.17. This term is expected to be smaller than the first term in a flat region, i.e. a low curvature region since the variance of $\hat{\mathbf{e}}_{i=0..2}$ is relatively small in this region. In cases of higher curvature regions, a much larger variance of the estimated $\hat{\mathbf{e}}_{i=0...2}$ is expected since a higher variance exists in the directions of the surface normal vector, i.e. $\hat{\mathbf{e}}_0$, and two tangential vectors, i.e. $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$. It is difficult to precisely estimate the eigenvectors of the covariance matrix of a point in a higher curvature region. Using Eq. 3.11, Eq. 3.12, and Eq. 3.13, the approximate diagonal

terms of $\mathbf{V}(\mathbf{r})_{point}$ is found by ignoring the second term of Eq. 3.17 as

$$\begin{aligned} V(r)^{ii}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_0, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}^{uvz}_0) \\ &= \cos^2\theta_i\sigma_r^2 + \sin^2\theta_i r^2\sigma_a^2 \end{aligned}$$

where $\theta_{i=0..2} = \{\theta, \alpha, \beta\}$. In addition, the off-diagonal terms are:

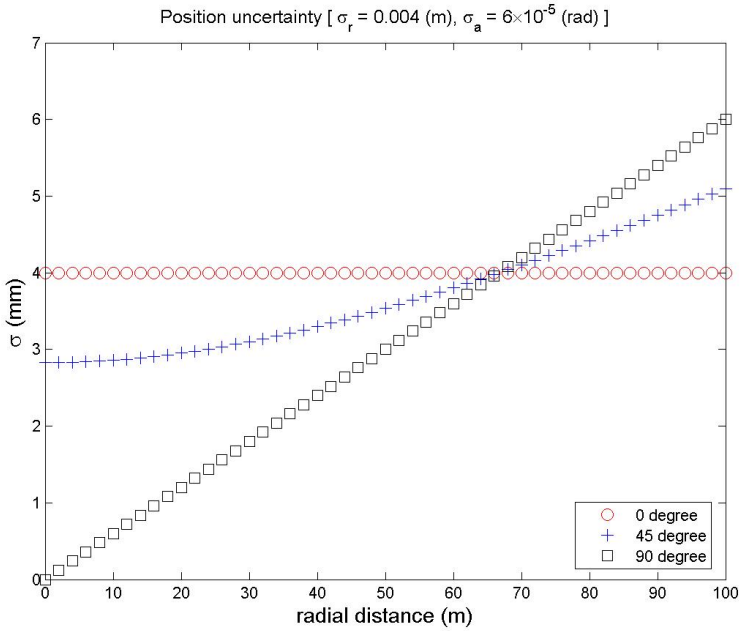
$$\begin{aligned} V(r)^{12}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_0, \mathbf{V}(\mathbf{r})_{laser}\hat{\mathbf{e}}^{uvz}_1) \\ &= \cos\theta\cos\alpha\sigma_r^2 - \mathrm{sgn}(\cos\alpha)|\sin\theta|\cos\theta\ r^2\sigma_a^2 \\ V(r)^{13}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_0, \mathbf{V}(\mathbf{r})_{laser}\hat{\mathbf{e}}^{uvz}_2) \\ &= \cos\theta\cos\beta\sigma_r^2 - \mathrm{sgn}(\cos\alpha)|\sin\theta|\cos\theta\ r^2\sigma_a^2 \\ V(r)^{23}_{point} &\simeq (\hat{\mathbf{e}}^{uvz}_1, \mathbf{V}(\mathbf{r})_{laser}\hat{\mathbf{e}}^{uvz}_2) \\ &= \cos\alpha\cos\beta\sigma_r^2 - \left(\frac{\cos^2\theta}{|\cos\alpha\cos\beta|} + 1\right)\cos\alpha\cos\beta\ r^2\sigma_a^2. \end{aligned}$$

The mathematical forms of the diagonal components of $\mathbf{V}(\mathbf{r})_{point}$ are as expected since θ, α, and β are the incidence angles of a laser beam to the vectors $\hat{\mathbf{e}}_{i=0...2}$, respectively. The diagonal term of $\mathbf{V}(\mathbf{r})_{point}$ is plotted in Figure 3.5(a) and Figure 3.5(b) against radial distance from a HDS2500 (Leica, 2006) scanner and the incidence angles of laser beam to the surface of objects, respectively.

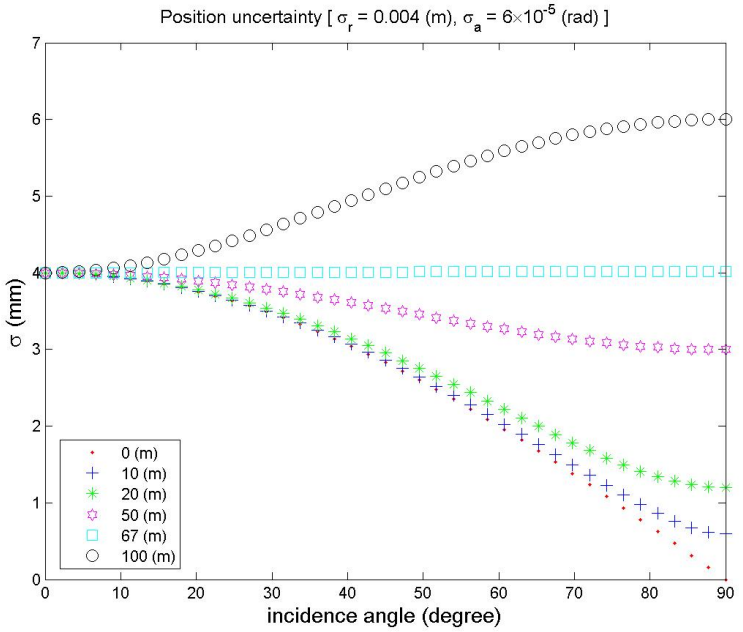
Consider the case that the line of sight of the laser beam is exactly aligned in the direction of the surface normal vector, in which $\theta = 0°$ and $\alpha = \beta = 90°$. In this case, $\mathbf{V}(\mathbf{r})^{11}_{point}$ is a constant, σ_r^2, and $\mathbf{V}(\mathbf{r})^{22}_{point}$ and $\mathbf{V}(\mathbf{r})^{33}_{point}$ follow $r^2\sigma_a^2$. Let a radial distance be $r_{Gaussian}$ in which the diagonal components are the same. It may simply be expressed as

$$r_{Gaussian} = \frac{\sigma_r}{\sigma_a}.$$

Importantly, the measurement of a laser scanner is regarded as a stochastic process with homogeneous variance, only if the distance between objects and the laser scanner is $r_{Gaussian}$. In other words, noise in laser scanner data is Gaussian only in this particular case. For example, at $r_{Gaussian} \simeq 67$m for a Leica HDS2500, these three diagonal components are the same. After 67m, the tangential components of the estimated surface are larger than the surface normal component, i.e. $\mathbf{V}(\mathbf{r})^{11}_{point}$. In addition, the diagonal components of $\mathbf{V}(\mathbf{r})_{point}$ decrease monotonically as the incidence angle increases up to 67m as shown in Figure 3.5(b). However, beyond 67m, if the incidence angle increases, then so does the diagonal component of $\mathbf{V}(\mathbf{r})$. The similar behaviour in the case of a Riegl LMS-Z210 is observed in Figure 3.6 except that three diagonal components are the same at about $r_{Gaussian} \simeq 53$m.

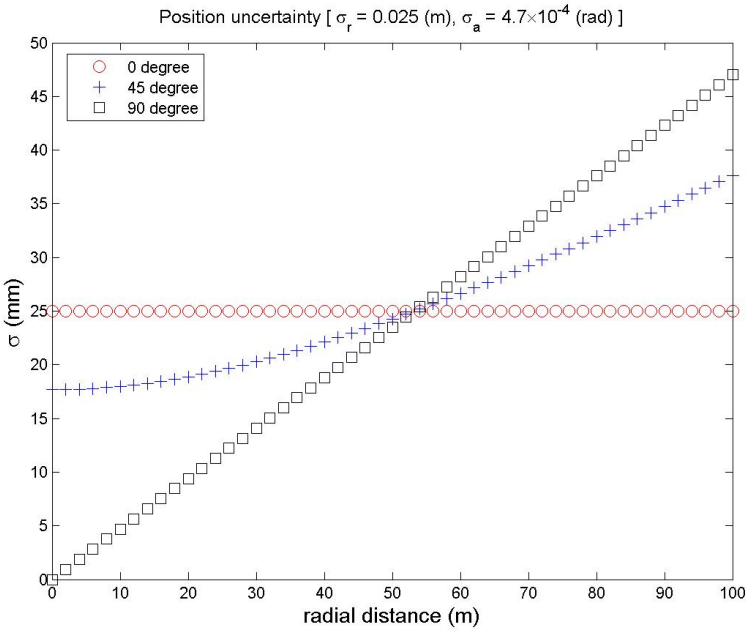


(a)

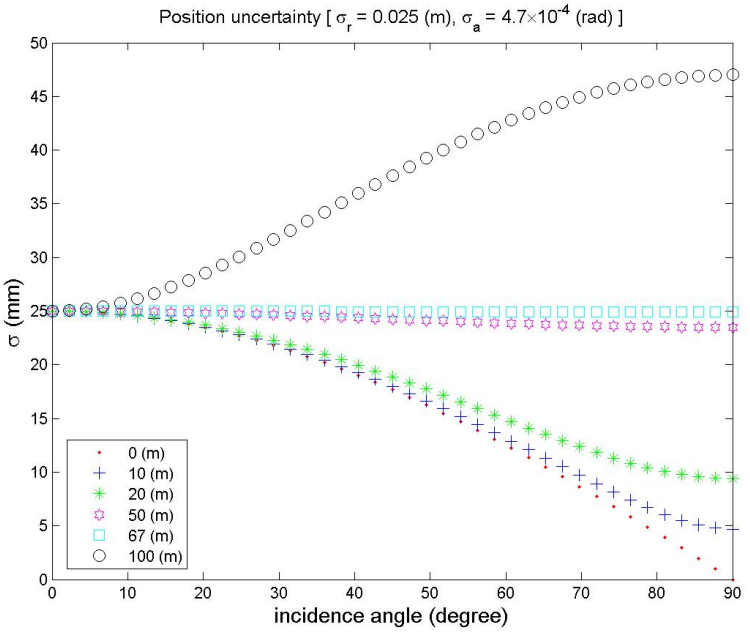


(b)

FIGURE 3.5: Position uncertainty in cases of different incidence angles and radial distances for a Leica HDS2500, i.e. the diagonal component of $V(r)_{point}$.

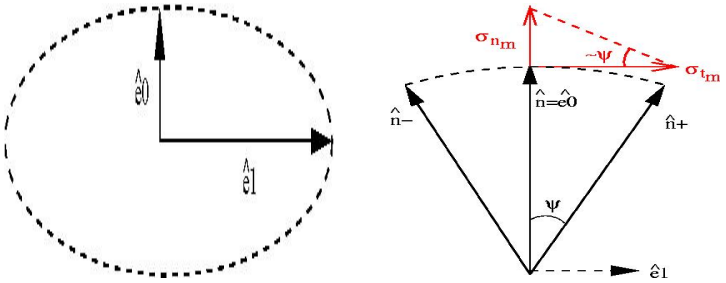


(a)



(b)

FIGURE 3.6: Position uncertainty in cases of different incidence angles and radial distances for a Riegl LMS-Z210 (Riegl, 2006; Lichti et al., 2005), i.e. the diagonal component of $V(r)_{point}$.



(a) (b)

FIGURE 3.7: Normal vector variance in a two-dimensional case.

3.5.1 Variance of the estimated surface normal vector

In this section, the variance of the estimated surface normal vector by the covariance method will be explicitly derived and then the position uncertainty will be rigorously expressed. Let $\mathbf{V}(\hat{\mathbf{e}}_i)_{point}$ and $\mathbf{V}(\mathbf{r})_{point}$ be $\mathbf{V}(\hat{\mathbf{e}}_i)$ and $\mathbf{V}(\mathbf{r})$, respectively. Consider a point and its neighbourhood which are unorganised and distributed within a two dimensional elliptical region whose semi-major axes are $\hat{\mathbf{e}}_0$ and $\hat{\mathbf{e}}_1$ as shown in Figure 3.7(a). The component of the variance matrix of the point, $\mathbf{V}(\mathbf{r}_m)$, is expressed as

$$\mathbf{V}(\mathbf{r}_m) = \mathbf{r}_m {\mathbf{r}_m}^T$$

where $\mathbf{r}_m = \mathbf{r} - \mathbf{r}_{centroid}$ and $\mathbf{V}(\mathbf{r}_m) \in \Re^{2\times 2}$ in two-dimensional cases. To evaluate the variance of $\hat{\mathbf{e}}_0$, let $\sigma^2_{t_m}$ and $\sigma^2_{n_m}$ be the variances of $\hat{\mathbf{e}}_0$ in the directions of $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_0$, respectively. They are presented in Figure 3.7(b) for the two-dimensional case. They can be expressed as follows:

$$\sigma^2_{t_m} \simeq \left(\hat{\mathbf{e}}_1, \mathbf{r}_m \mathbf{r}_m^T \hat{\mathbf{e}}_1\right) = \left(\mathbf{P}_{\hat{\mathbf{e}}_0}\hat{\mathbf{e}}_1, \mathbf{r}_m \mathbf{r}_m^T \mathbf{P}^T_{\hat{\mathbf{e}}_0}\hat{\mathbf{e}}_1\right) = \left(\hat{\mathbf{e}}_1, \left[\mathbf{P}_{\hat{\mathbf{e}}_0}\mathbf{r}_m\right]\left[\mathbf{P}_{\hat{\mathbf{e}}_0}\mathbf{r}_m\right]^T \hat{\mathbf{e}}_1\right)$$
$$\sigma^2_{n_m} \simeq \left(\hat{\mathbf{e}}_0, \mathbf{r}_m \mathbf{r}_m^T \hat{\mathbf{e}}_0\right)$$

where $\mathbf{P}_{\hat{\mathbf{e}}_i} = \mathbf{I}_2 - \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T$ is the projection operator of $\hat{\mathbf{e}}_i$, $\mathbf{r}_m = \mathbf{r} - \mathbf{r}_{centroid}$, and $\mathbf{P}_{\hat{\mathbf{e}}_i}\hat{\mathbf{e}}_j = \mathbf{P}^T_{\hat{\mathbf{e}}_i}\hat{\mathbf{e}}_j = \hat{\mathbf{e}}_j$. Note that the variance of $\hat{\mathbf{e}}_0$, $V(\hat{\mathbf{e}}_0)$, is a scalar since we are dealing with a two-dimensional case. The Cramer-Rao lower bound for $V(\hat{\mathbf{e}}_0)$ is expressed as

$$V(\hat{\mathbf{e}}_0) = \frac{1}{\sum_{m=1}^{k} \frac{1}{V(\hat{\mathbf{e}}_0^m)}} \tag{3.18}$$

where $\hat{\mathbf{e}}_0^m$ is the estimated normal vector using of the kth neighbour point. Its mathematical proof can be found in Appendix B or Kay (1993, Ch. 3). If an unbiased scalar estimator of a least-squares problem is obtained, then Eq. 3.18 is the lower bound for the variance of the estimator. Bearing in mind that the variance of $\hat{\mathbf{e}}_0^m$ is inversely

proportional to $\sigma_{t_m}^2$, it is estimated as

$$V(\hat{\mathbf{e}}_0^m) \simeq \frac{\sigma_{n_m}^2}{\sigma_{t_m}^2},$$

therefore

$$V(\hat{\mathbf{e}}_0) \simeq \left[\sum_{m=1}^{k} \frac{\left(\hat{\mathbf{e}}_1, [\mathbf{P}_{\hat{\mathbf{e}}_0} \mathbf{r}_m] [\mathbf{P}_{\hat{\mathbf{e}}_0} \mathbf{r}_m]^T \hat{\mathbf{e}}_1 \right)}{(\hat{\mathbf{e}}_0, \mathbf{r}_m \mathbf{r}_m^T \hat{\mathbf{e}}_0)} \right]^{-1}.$$

In general, the covariance matrix of an estimated vector is written

$$\mathbf{V}(\hat{\mathbf{e}}_i) = \left(\sum_{m=1}^{k} \frac{(\mathbf{P}_{\hat{\mathbf{e}}_i} \mathbf{r}_m)(\mathbf{P}_{\hat{\mathbf{e}}_i} \mathbf{r}_m)^T}{(\hat{\mathbf{e}}_i, \mathbf{V}(\mathbf{r}_m) \hat{\mathbf{e}}_i)} \right)^{-} \tag{3.19}$$

where $\mathbf{A}^-$ is the Moore-Penrose generalised inverse of a matrix $\mathbf{A}$ (Kanatani, 1996, Ch. 7; Kanazawa and Kanatani, 1996). Let $\hat{\mathbf{n}}$ be $\hat{\mathbf{e}}_0$, the estimated surface normal vector of a point and its neighbourhood. Then the variance of $\hat{\mathbf{n}}$ can be decomposed into $\mathbf{n}_+$ and $\mathbf{n}_-$, which are expressed as

$$\mathbf{n}_\pm = \hat{\mathbf{n}} \pm \sum_{i=1}^{2} \mathbf{n}^i$$

where $\mathbf{n}^{i=1..2} = \sqrt{(\hat{\mathbf{e}}_i, \mathbf{V}(\hat{\mathbf{n}}) \hat{\mathbf{e}}_i)} \hat{\mathbf{e}}_i$ and is shown graphically in Figure 3.8. The variance angles of the estimated surface normal vector, $\Psi^{i=1...2}$, is defined

$$\begin{aligned} \cos \Psi^i &= \left(\hat{\mathbf{n}}, \frac{\hat{\mathbf{n}} + \hat{\mathbf{n}}^i}{\parallel \hat{\mathbf{n}} + \hat{\mathbf{n}}^i \parallel} \right) = \frac{1}{\parallel \hat{\mathbf{n}} + \hat{\mathbf{n}}^i \parallel} \\ &= \frac{1}{\sqrt{1 + (\hat{\mathbf{e}}_i, \mathbf{V}(\hat{\mathbf{n}}) \hat{\mathbf{e}}_i)}} \end{aligned}$$

so

$$\tan^2 \Psi^i = (\hat{\mathbf{e}}_i, \mathbf{V}(\hat{\mathbf{n}}) \hat{\mathbf{e}}_i) \tag{3.20}$$

using $\cos^2 \Psi^i = \left[1 + \tan^2 \Psi^i \right]^{-1}$ where $i = 1..2$. The variance angles represent indirectly the size of the variance region of the estimated surface normal vectors as shown in Figure 3.8. In three-dimensional cases, we have two variance angles and if they are the same, then the variance region of the estimated normal vector is a circular cone. Otherwise, it is an elliptical cone. A more explicit form of Eq. 3.19 needs to be derived since we can not investigate any further the characteristics of the second term of Eq. 3.17 with Eq. 3.19 in unorganised point clouds from laser scanners. Note that we can not

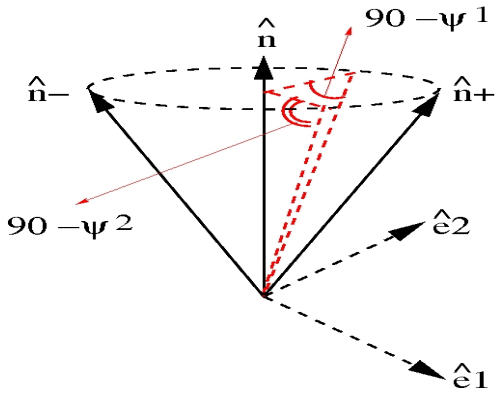


FIGURE 3.8: Variance region and the variance angles of the estimated surface normal vector in a three dimensional case.

directly prove if Eq. 3.19 is correct. However, the correctness of Eq. 3.19 will be indirectly proved by investigating the components of an explicit form of the equation.

With the assumptions that $\mathbf{V}(\mathbf{r}_m)$ is uniform and the change of the true normal vector within the neighbourhood of a query point is small, then

$$\begin{aligned}\sum_{m=1}^{k}\frac{(\mathbf{P}_{\hat{\mathbf{n}}}\mathbf{r}_m)(\mathbf{P}_{\hat{\mathbf{n}}}\mathbf{r}_m)^T}{(\hat{\mathbf{n}},\mathbf{V}(\mathbf{r}_m)\hat{\mathbf{n}})} &= \sum_{m=1}^{k}\frac{(\mathbf{P}_{\hat{\mathbf{n}}}\mathbf{r}_m)(\mathbf{r}_m^T\mathbf{P}_{\hat{\mathbf{n}}}^T)}{(\hat{\mathbf{n}},\mathbf{V}(\mathbf{r}_m)\hat{\mathbf{n}})}\\ &\simeq \mathbf{P}_{\hat{\mathbf{n}}}\left[\sum_{m=1}^{k}\frac{\mathbf{r}_m\mathbf{r}_m^T}{(\hat{\mathbf{n}},\mathbf{V}(\mathbf{r}_m)\hat{\mathbf{n}})}\right]\mathbf{P}_{\hat{\mathbf{n}}}^T\\ &\simeq \frac{1}{(\hat{\mathbf{n}},\mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}\mathbf{P}_{\hat{\mathbf{n}}}\left[\sum_{m=1}^{k}\mathbf{r}_m\mathbf{r}_m^T\right]\mathbf{P}_{\hat{\mathbf{n}}}^T\end{aligned}$$

where $\mathbf{r}_m = \mathbf{r}-\mathbf{r}_{centroid}$ and $\mathbf{r}_{m=0}$ is a query point. Using the definition of the covariance matrix,

$$\begin{aligned}\frac{1}{(\hat{\mathbf{n}},\mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}\mathbf{P}_{\hat{\mathbf{n}}}\left[\sum_{m=1}^{k}\mathbf{r}_m\mathbf{r}_m^T\right]\mathbf{P}_{\hat{\mathbf{n}}}^T &= \frac{k}{(\hat{\mathbf{n}},\mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}\mathbf{P}_{\hat{\mathbf{n}}}\left[\sum_{i=0}^{2}\lambda_i\hat{\mathbf{e}}_i\hat{\mathbf{e}}_i^T\right]\mathbf{P}_{\hat{\mathbf{n}}}^T\\ &= \frac{k}{(\hat{\mathbf{n}},\mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}\left[\sum_{i=1}^{2}\lambda_i\hat{\mathbf{e}}_i\hat{\mathbf{e}}_i^T\right] \qquad (3.21)\end{aligned}$$

since $\mathbf{P}_{\hat{\mathbf{e}}_0}\hat{\mathbf{e}}_0 = \mathbf{P}_{\hat{\mathbf{n}}}\hat{\mathbf{n}} = \mathbf{0}$. The rank of Eq. 3.21 is two in the case of three-dimensional point clouds and its Moore-Penrose generalised inverse is the variance of the estimated

normal vector:

$$\mathbf{V}(\hat{\mathbf{n}}) = \left\{ \frac{k}{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})} \left[\sum_{i=1}^{2} \lambda_i \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T \right] \right\}^{-} = \frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}{k} \left[\sum_{i=1}^{2} \frac{1}{\lambda_i} \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T \right]. \tag{3.22}$$

The angular variance defined in Eq. 3.20 is rewritten as

$$\Psi^{i=1\ldots2} = \tan^{-1}\left[(\hat{\mathbf{e}}_i, \mathbf{V}(\hat{\mathbf{n}})\hat{\mathbf{e}}_i)^{\frac{1}{2}} \right] = \tan^{-1}\left\{ \left[\frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}{k\lambda_i} \right]^{\frac{1}{2}} \right\}. \tag{3.23}$$

Given the variance matrix of the estimated normal vector in Eq. 3.22, the variance matrix of a n-dimensional eigenvector is expressed as

$$\mathbf{V}(\hat{\mathbf{e}}_i) = \frac{(\hat{\mathbf{e}}_i, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{e}}_i)}{k} \left[\sum_{j=0, j\neq i}^{n} \frac{1}{\lambda_j} \hat{\mathbf{e}}_j \hat{\mathbf{e}}_j^T \right]. \tag{3.24}$$

Note that the variance matrix of the eigenvectors in $\mathbf{O}_{point}$ has been evaluated using $\mathbf{r}_m = \mathbf{r} - \mathbf{r}_{centroid}$. One diagonal component of Eq. 3.17 in the direction of $\hat{\mathbf{n}}$ is rewritten as

$$\begin{aligned} V_{point}^{11} \equiv V^{normal}(\mathbf{r}) &= \frac{1}{2}\left\{ (\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_m)\hat{\mathbf{n}}) + (\mathbf{r}_m, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r}_m) \right\} \\ &= (\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r})\hat{\mathbf{n}}) + \frac{1}{2}(\mathbf{r}_m, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r}_m). \end{aligned} \tag{3.25}$$

using $\mathbf{V}(\mathbf{r}_m) = 2\mathbf{V}(\mathbf{r})$ and $V(a+b) = V(a) + V(b)$ with stochastic variables a and b. In a similar manner, the components of $\mathbf{V}(\mathbf{r})_{point}$ are expressed as

$$\begin{aligned} V(r)_{point}^{ij} &= (\hat{\mathbf{e}}_{i-1}^{uvz}, \mathbf{V}(\mathbf{r})_{uvz}\hat{\mathbf{e}}_{j-1}^{uvz}) + \sum_{k=min(i,j)}^{\max(i,j)} \left(\frac{1}{2}\right)^{\delta_{ij}+1} (\delta_{ik} + \delta_{jk}) (\mathbf{r}_m, \mathbf{V}(\hat{\mathbf{e}}_k)\mathbf{r}_m) \\ &= (\hat{\mathbf{e}}_{i-1}, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_{j-1}) + \sum_{k=min(i,j)}^{\max(i,j)} \left(\frac{1}{2}\right)^{\delta_{ij}+1} (\delta_{ik} + \delta_{jk}) (\mathbf{r}_m, \mathbf{V}(\hat{\mathbf{e}}_k)\mathbf{r}_m) \end{aligned} \tag{3.26}$$

since $(\hat{\mathbf{e}}_{i-1}, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_{j-1}) = (\hat{\mathbf{e}}_{i-1}^{uvz}, \mathbf{V}(\mathbf{r})_{uvz}\ \hat{\mathbf{e}}_{j-1}^{uvz})$ and $\mathbf{V}(\mathbf{r})_{laser} = \mathbf{V}(\mathbf{r})_{uvz}$. Note that the only reason for the revision of Eq. 3.17 into Eq. 3.26 is that the variance of the normal vector has been calculated using $\mathbf{r}_m$ as shown in Eq. 3.24. Assuming $\frac{\mathbf{r}}{|\mathbf{r}|} \simeq \hat{\mathbf{z}} \simeq \frac{\mathbf{r}_{centroid}}{|\mathbf{r}_{centroid}|}$, $\mathbf{V}(\mathbf{r}) \simeq \mathbf{V}(\mathbf{r}_{centroid})$, and $\mathbf{V}(\mathbf{r}_m) \simeq 2\mathbf{V}(\mathbf{r})$, the component of $\mathbf{V}(\hat{\mathbf{n}})$ in the direction of

$\frac{\mathbf{r}_m}{\|\mathbf{r}_m\|}$ is approximately expressed as follows

$$\begin{aligned}(\mathbf{r}_m, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r}_m) &\simeq (\mathbf{r}, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r}) - (\mathbf{r}_{centroid}, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r}_{centroid}) \\ &= r^2(\hat{\mathbf{z}}, \mathbf{V}(\hat{\mathbf{n}})\hat{\mathbf{z}}) - r^2_{centroid}(\hat{\mathbf{z}}, \mathbf{V}(\hat{\mathbf{n}})\hat{\mathbf{z}})\end{aligned}$$

and using Eq. 3.24,

$$\begin{aligned}(\mathbf{r}_m, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r}_m) &\simeq (\mathbf{r}^2 - \mathbf{r}^2_{centroid})(\hat{\mathbf{z}}, \mathbf{V}(\hat{\mathbf{n}})\hat{\mathbf{z}}) \\ &= (\mathbf{r}^2 - \mathbf{r}^2_{centroid}) \left(\hat{\mathbf{z}}, \frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}{k} \left[\sum_{i=1}^{2} \frac{1}{\lambda_i} \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T \right] \hat{\mathbf{z}} \right) \\ &= (\mathbf{r}^2 - \mathbf{r}^2_{centroid}) \frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}{k} \left(\hat{\mathbf{z}}, \left[\sum_{i=1}^{2} \frac{1}{\lambda_i} \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T \right] \hat{\mathbf{z}} \right) \\ &= (\mathbf{r}^2 - \mathbf{r}^2_{centroid}) \frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}{k} \sum_{i=1}^{2} \frac{1}{\lambda_i} (\hat{\mathbf{z}}, \hat{\mathbf{e}}_i)^2 \\ &\simeq \left(r^2 - r^2_{centroid}\right) \frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}_{m=0})\hat{\mathbf{n}})}{k} \left[\frac{1}{\lambda_1} \cos^2\alpha + \frac{1}{\lambda_2} \cos^2\beta \right] \\ &= 2\left(r^2 - r^2_{centroid}\right) \frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r})\hat{\mathbf{n}})}{k} \left[\frac{1}{\lambda_1} \cos^2\alpha + \frac{1}{\lambda_2} \cos^2\beta \right]. \end{aligned} \tag{3.27}$$

A complete and explicit form of $\mathbf{V}(\mathbf{r})_{point}$ and its ith row and jth column element, $V(r)^{ij}_{point}$, is expressed using Eqs. 3.26 and 3.27 as follows

$$\begin{aligned}V(r)^{11}_{point} &\simeq (\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r}) \\ &= \cos^2\theta\sigma_r^2 + \sin^2\theta\ r^2\sigma_a^2 + \left(r^2 - r^2_{centroid}\right) \frac{(\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0)}{k} \left[\frac{1}{\lambda_1} \cos^2\alpha + \frac{1}{\lambda_2} \cos^2\beta \right]\end{aligned} \tag{3.28}$$

$$\begin{aligned}V(r)^{22}_{point} &\simeq (\hat{\mathbf{e}}_1, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_1) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_1)\mathbf{r}) \\ &= \cos^2\alpha\sigma_r^2 + \sin^2\alpha\ r^2\sigma_a^2 + \left(r^2 - r^2_{centroid}\right) \frac{(\hat{\mathbf{e}}_1, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_1)}{k} \left[\frac{1}{\lambda_0} \cos^2\theta + \frac{1}{\lambda_2} \cos^2\beta \right]\end{aligned} \tag{3.29}$$

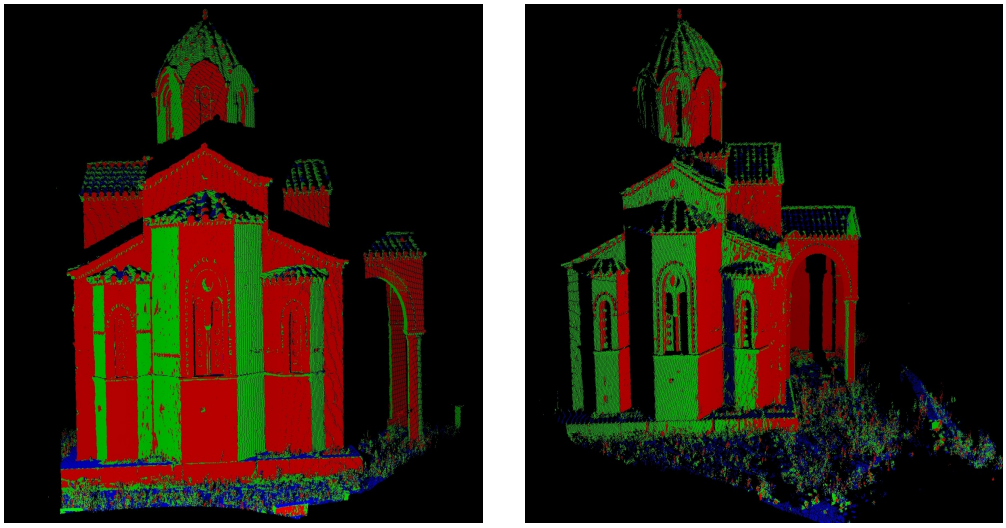
$$\begin{aligned}V(r)^{33}_{point} &\simeq (\hat{\mathbf{e}}_2, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_2) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_2)\mathbf{r}) \\ &= \cos^2\beta\sigma_r^2 + \sin^2\beta\ r^2\sigma_a^2 + \left(r^2 - r^2_{centroid}\right) \frac{(\hat{\mathbf{e}}_2, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_2)}{k} \left[\frac{1}{\lambda_0} \cos^2\theta + \frac{1}{\lambda_1} \cos^2\alpha \right]\end{aligned} \tag{3.30}$$

and

$$
\begin{aligned}
V(r)^{12}_{point} &= V(r)^{21}_{point} \simeq (\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_1) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r}) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_1)\mathbf{r}) \\
&= \cos\theta\cos\alpha\sigma_r^2 - \mathrm{sgn}(\cos\alpha)|\sin\theta|\cos\theta\ r^2\sigma_a^2 \\
&+ \left(r^2 - r^2_{centroid}\right)\frac{(\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0)}{k}\left[\frac{1}{\lambda_1}\cos^2\alpha + \frac{1}{\lambda_2}\cos^2\beta\right] \\
&+ \left(r^2 - r^2_{centroid}\right)\frac{(\hat{\mathbf{e}}_1, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_1)}{k}\left[\frac{1}{\lambda_0}\cos^2\theta + \frac{1}{\lambda_2}\cos^2\beta\right] \\
V(r)^{13}_{point} &= V(r)^{31}_{point} \simeq (\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_2) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r}) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_2)\mathbf{r}) \\
&= \cos\theta\cos\beta\sigma_r^2 - \mathrm{sgn}(\cos\alpha)|\sin\theta|\cos\theta\ r^2\sigma_a^2 \\
&+ \left(r^2 - r^2_{centroid}\right)\frac{(\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0)}{k}\left[\frac{1}{\lambda_1}\cos^2\alpha + \frac{1}{\lambda_2}\cos^2\beta\right] \\
&+ \left(r^2 - r^2_{centroid}\right)\frac{(\hat{\mathbf{e}}_2, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_2)}{k}\left[\frac{1}{\lambda_0}\cos^2\theta + \frac{1}{\lambda_1}\cos^2\alpha\right] \\
V(r)^{23}_{point} &= V(r)^{32}_{point} \simeq (\hat{\mathbf{e}}_1, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_2) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_1)\mathbf{r}) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_2)\mathbf{r}) \\
&= \cos\alpha\cos\beta\sigma_r^2 - \left(\frac{\cos^2\theta}{|\cos\alpha\cos\beta|} + 1\right)\cos\alpha\cos\beta\ r^2\sigma_a^2 \\
&+ \left(r^2 - r^2_{centroid}\right)\frac{(\hat{\mathbf{e}}_1, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_1)}{k}\left[\frac{1}{\lambda_0}\cos^2\theta + \frac{1}{\lambda_2}\cos^2\beta\right] \\
&+ \left(r^2 - r^2_{centroid}\right)\frac{(\hat{\mathbf{e}}_2, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_2)}{k}\left[\frac{1}{\lambda_0}\cos^2\theta + \frac{1}{\lambda_1}\cos^2\alpha\right].
\end{aligned}
$$

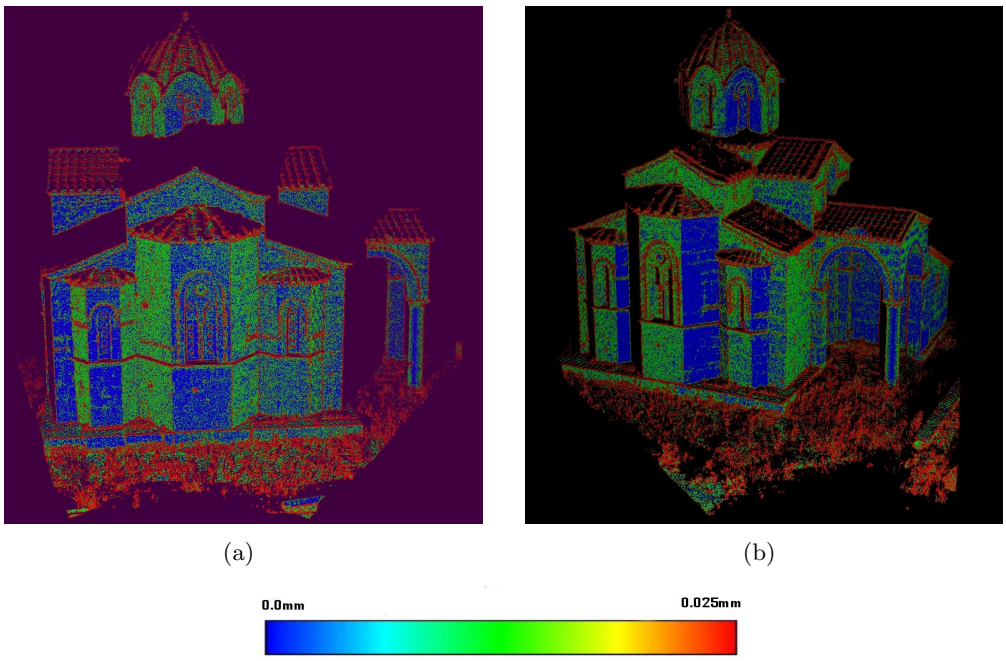
The second term of Eq. 3.26 is expected to be small since it is proportional to the differences between the squared radial distances of the query point and the centroid of its neighbourhood, i.e. $\left(r^2 - r^2_{centroid}\right)$. Let δ^{11} be the ratio between the first and second term of the diagonal component of $V(r)^{11}$ expressed as

$$
\begin{aligned}
\delta^{11} = \frac{(\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r})}{(\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0)} &= \left(r^2 - r^2_{centroid}\right)\frac{1}{k}\left[\frac{1}{\lambda_1}\cos^2\alpha + \frac{1}{\lambda_2}\cos^2\beta\right] \\
&\leq \left(r^2 - r^2_{centroid}\right)\frac{1}{k}\left[\frac{1}{\lambda_1} + \frac{1}{\lambda_2}\right] \qquad (3.31)
\end{aligned}
$$



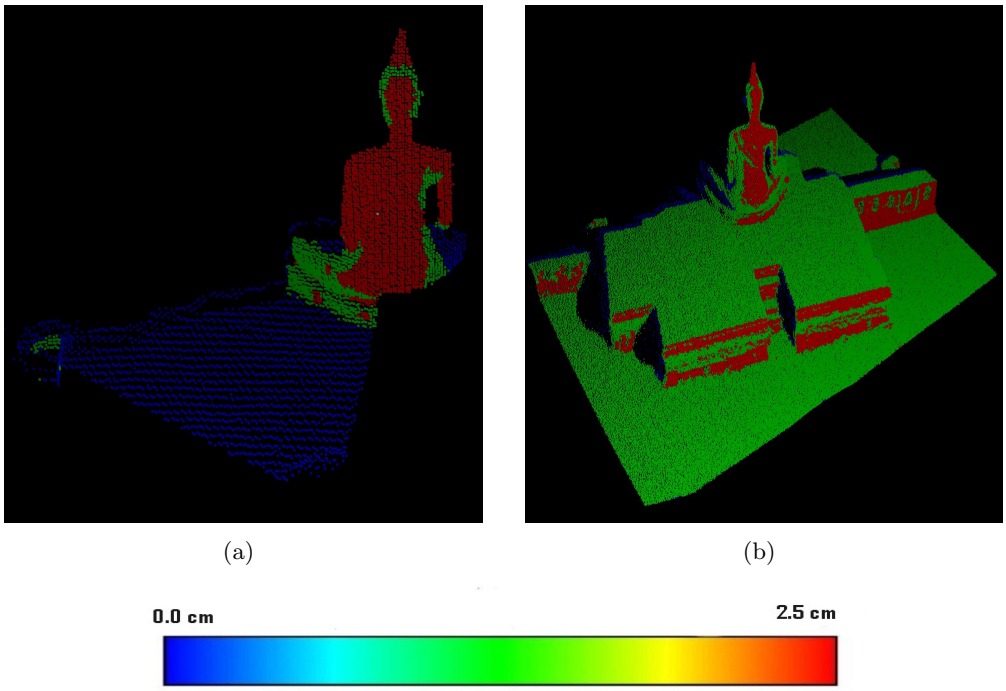
(a) (b)

FIGURE 3.9: (a) and (b) are $\sqrt{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}))\hat{\mathbf{n}})}$, i.e. the square root of the first term of Eq. 3.26, of the Agia Sanmarina church in Greece scanned from different locations using a Leica HDS2500.



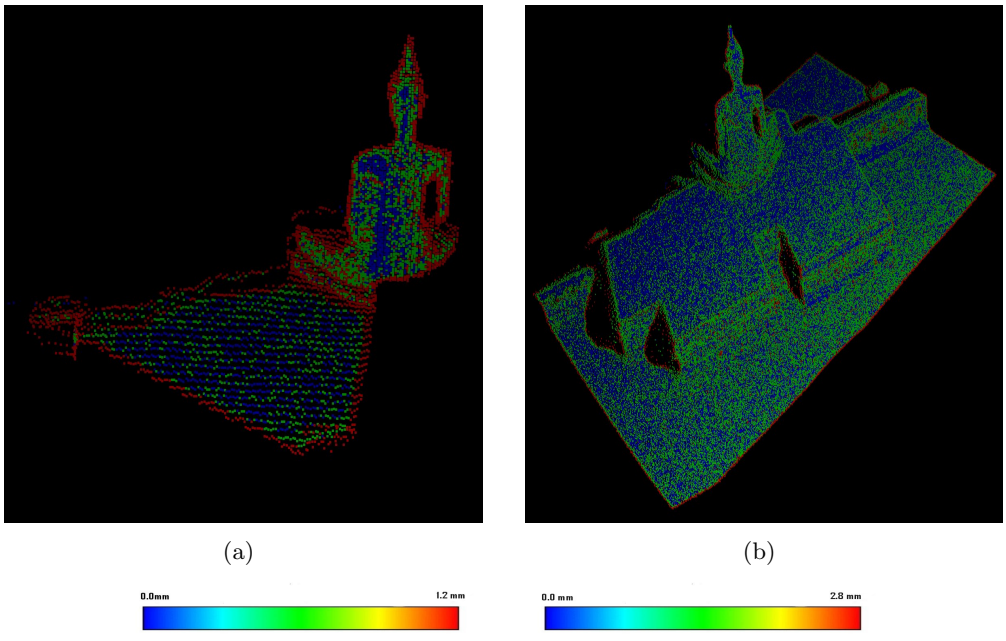
(a) (b)

FIGURE 3.10: (a) and (b) are $\sqrt{(\hat{\mathbf{r}}, \mathbf{V}(\mathbf{n}))\hat{\mathbf{r}})}$, i.e. the square root of the second term of Eq. 3.26, of the Agia Sanmarina church in Greece scanned from different locations using a Leica HDS2500.



(a) (b)

FIGURE 3.11: (a) and (b) are $\sqrt{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}))\hat{\mathbf{n}}}$, i.e. the square root of the first term of Eq. 3.26, of the Ayuthaya Buddha statute in Thailand scanned from different locations using a Riegl LMS-Z210.



(a) (b)

FIGURE 3.12: (a) and (b) are $\sqrt{(\hat{\mathbf{r}}, \mathbf{V}(\mathbf{n}))\hat{\mathbf{r}}}$, i.e. the square root of the second term of Eq. 3.26, of the the Ayuthaya Buddha statute in Thailand scanned from different locations using a Riegl LMS-Z210.

The length, d, of the maximum major axis of an ellipse, where a query point and its k-neighbourhood are distributed, can be estimated as

$$\frac{1}{d^2} \simeq \max\left[\frac{1}{k\lambda_1}, \frac{1}{k\lambda_2}\right]$$

and the upper bound of δ^{11} can be given as

$$\delta^{11} \leq \mu\frac{1}{d^2}.$$

where $\mu = \left(r^2 - r^2_{centroid}\right)$. Both d and μ are dependent on the distribution of points and the behaviour of these two variables is stated as follows:

- Suppose that points in a point cloud are distributed evenly enough to assume that μ is a constant in a low curvature region. In a higher curvature region, μ will be greater than in a low curvature region since points in the high curvature are not distributed as evenly as points in the lower curvature region.
- In a similar manner, we can find that d in a low curvature region is greater than in a higher curvature region.
- δ^{ij} can be deduced to be smaller in a lower curvature region than in the higher curvature region and can perform as an edge detector as shown in Figures 3.10 and 3.12.

The colour maps of $\sqrt{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}))\hat{\mathbf{n}})}$ and $\sqrt{(\mathbf{r}, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r})}$ of two point clouds scanned from two different locations are presented in Figures 3.9 − 3.12. For the Agia Sanmarina data, the effect of the radial variance in the direction of the surface normal vector is about a hundred times larger than that of $\sqrt{(\mathbf{r}, \mathbf{V}(\hat{\mathbf{n}})\mathbf{r})}$. In Figure 3.9, it is also observed that there is about 2mm of the difference in $\sqrt{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}))\hat{\mathbf{n}})}$ between the overlapping regions of the point clouds. In the case of the Ayuthaya Buddha statue, it is observed that its square root of the first term of Eq. 3.26 is about fifty times greater than that of Agia Sanmarina. In addition, the magnitude of the square root of the second term of Eq. 3.26 is about hundred times greater than that of Agia Sanmarina and is on the order of millimetres.

3.5.2 Position uncertainty in $O_{scanner}$

In this section, it will be numerically proved that the approximation of the rotation matrix between $\mathbf{O}_{point}$ and $\mathbf{O}_{uvz}$, i.e. Eq. 3.14, is a reasonable estimation by evaluating $\mathbf{V}(\mathbf{r})_{scanner}$ from $\mathbf{V}(\mathbf{r})_{uvz}$. Using the position uncertainty in $\mathbf{O}_{point}$, $\mathbf{V}(\mathbf{r})_{scanner}$ is

simply expressed as

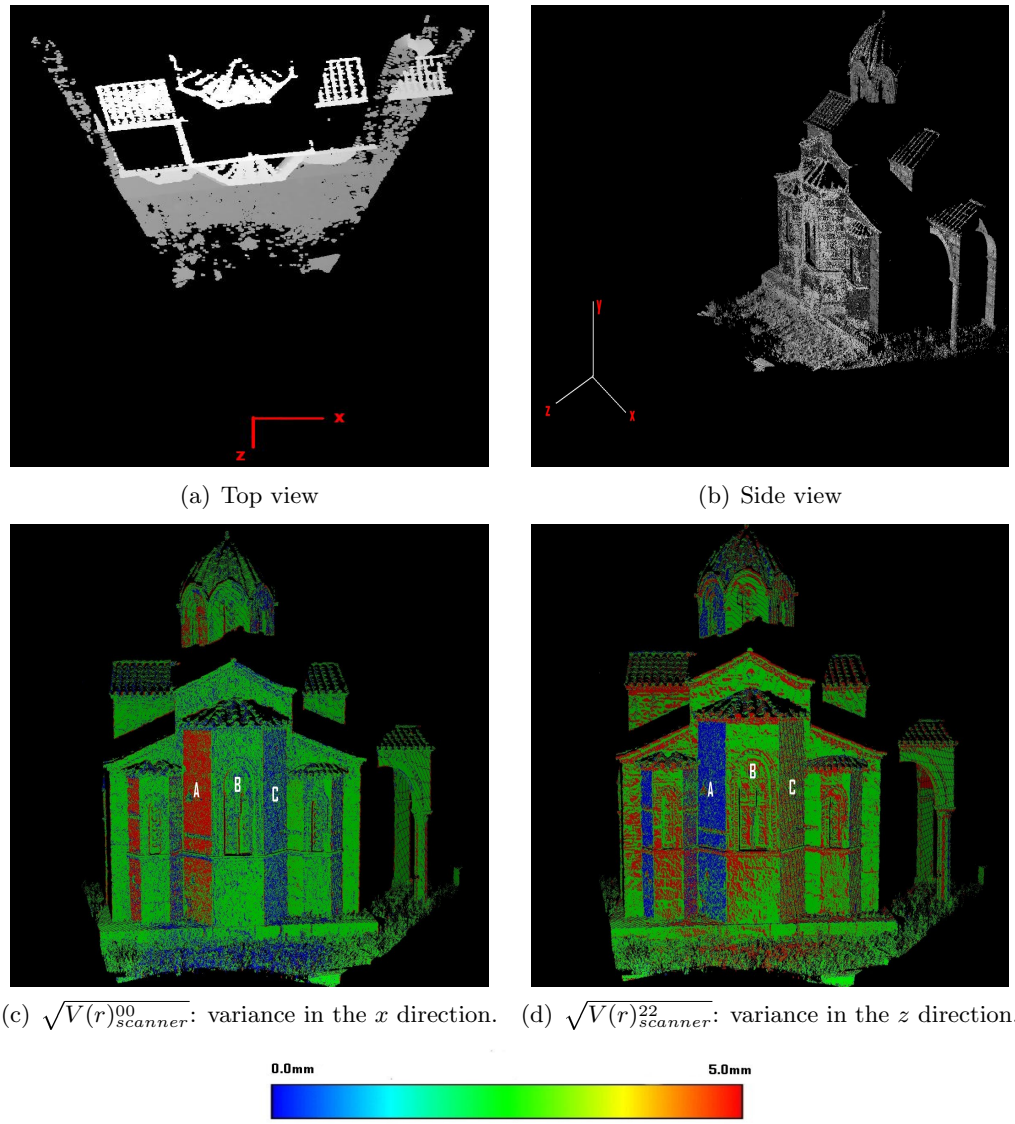
$$\begin{aligned}\mathbf{V}(\mathbf{r})_{scanner} &= \mathbf{R}_{point}^{scanner} V_{point} (\mathbf{R}_{point}^{scanner})^T \\ &= \begin{pmatrix} \hat{\mathbf{e}}_0 & \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 \end{pmatrix}^T \mathbf{V}(\mathbf{r})_{point} \begin{pmatrix} \hat{\mathbf{e}}_0 & \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 \end{pmatrix}. \end{aligned} \tag{3.32}$$

There is one thing to be cautious of in the construction of $\mathbf{R}_{point}^{scanner}$ using the eigenvectors of the covariance matrix. The rotation matrix constructed by Eq. 3.3 may not be a proper rotation matrix, i.e. $\det(\mathbf{R}) = -1$. Umeyama (1991) proposed a method to correct an improper rotation matrix using singular value decomposition. However, this problem is simply fixed by flipping one of the eigenvectors until a proper rotation matrix is obtained.

The colour maps of the square root of the diagonal components of $\mathbf{V}(\mathbf{r})_{scanner}$ in two point clouds of the Agia Sanmarina church are presented in Figures. 3.13 − 3.15. The y-direction components of these two point clouds in $\mathbf{O}_{scanner}$ are expected to be similar, as shown in Figure 3.15 since the surfaces of the object are approximately symmetric in the direction of the y axis in $\mathbf{O}_{scanner}$.

Region A in Figures 3.13(c) and Figure 3.13(d) is expected to have a larger variance in the x-direction than in the z-direction and this is easily deduced from Figure 3.13(a) using the incidence angle of the laser beam to the surface of region A. In a similar manner, it is expected that the square root of the variances in both the x and z directions in region B will be similar. On the other hand, z-direction variance in region C in Figures 3.13 is larger than that in the x-direction. For example, if the laser scanner is placed in a position where the scanner can view the church in a symmetric manner in the direction of the x axis of the laser scanner, then the positional uncertainties in regions A and B are similar. In addition, the same kinds of results are expected for Figures 3.14(c) and 3.14(d).

Regions B and C in Figure 3.13 are the corresponding regions of E and F in Figure 3.14, respectively. Note that region A is invisible from the location of the scanner in Figure 3.14. There is about 2mm difference in the square roots of the variances of region B in Figure 3.13(c) and region E in Figure 3.14(c). Regions C of Figure 3.13(c) and F of Figure 3.14(c) have similar amounts of the difference in the square roots of the variance. Also note that, in Figure 3.9, similar differences in the square roots of the variances have been observed in the direction of the estimated surface normal vector in the same regions. These results indicate that $\mathbf{R}_{point}^{scanner}$ in Eq. 3.3 is a reasonable estimation of the rotation matrix from the measured point coordinate system to the scanner coordinate system.



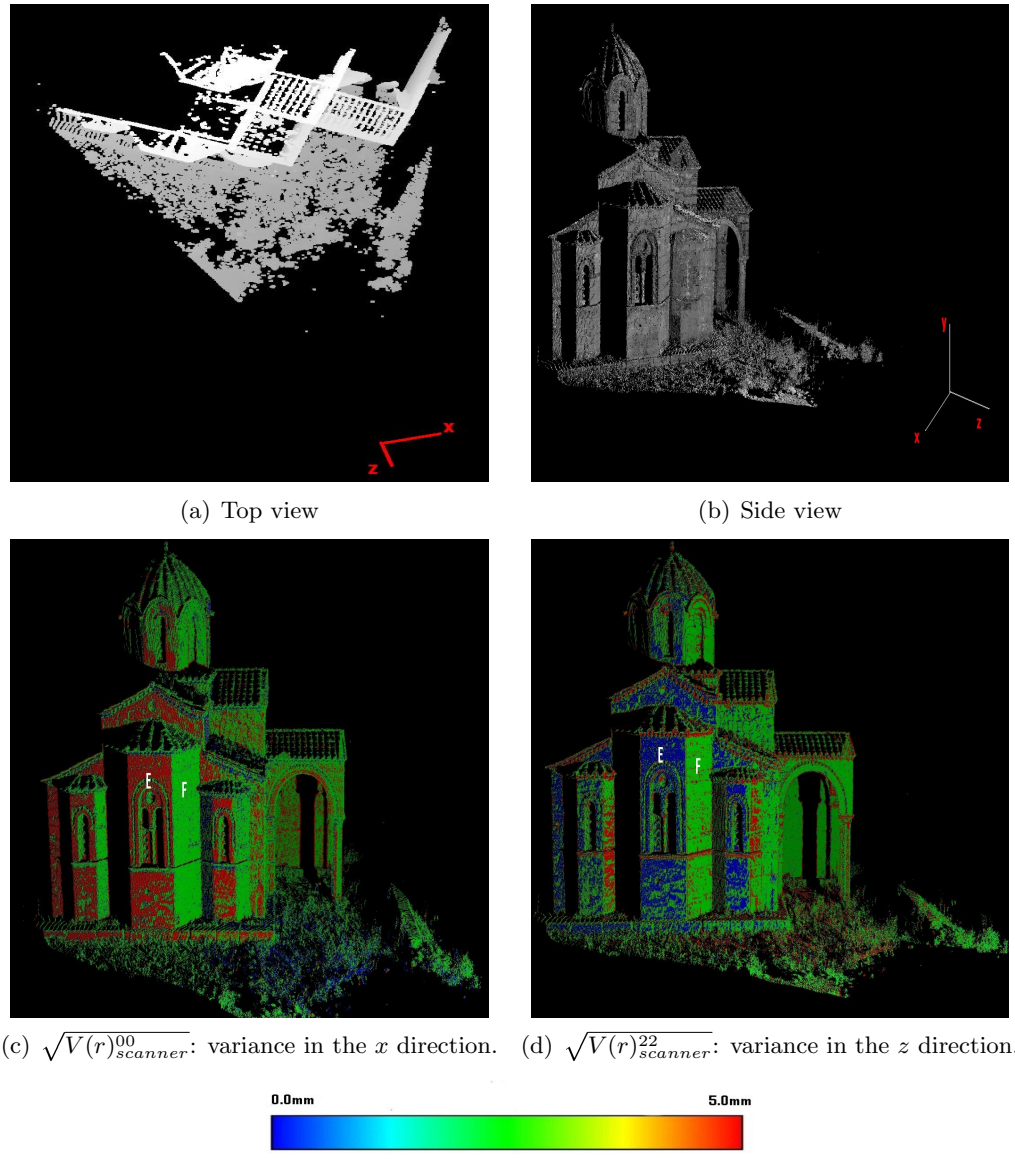
(a) Top view

(b) Side view

(c) $\sqrt{V(r)^{00}_{scanner}}$: variance in the x direction.

(d) $\sqrt{V(r)^{22}_{scanner}}$: variance in the z direction.

FIGURE 3.13: (a) is the top view of a point cloud, respectively, of the Agia Sanmarina church. The locations of the laser scanner and two axes of $\mathbf{O}_{scanner}$ are drawn in (a). The position uncertainties in the direction of the x and z axes calculated from Eq. 3.32 are presented in (c) and (d), respectively.



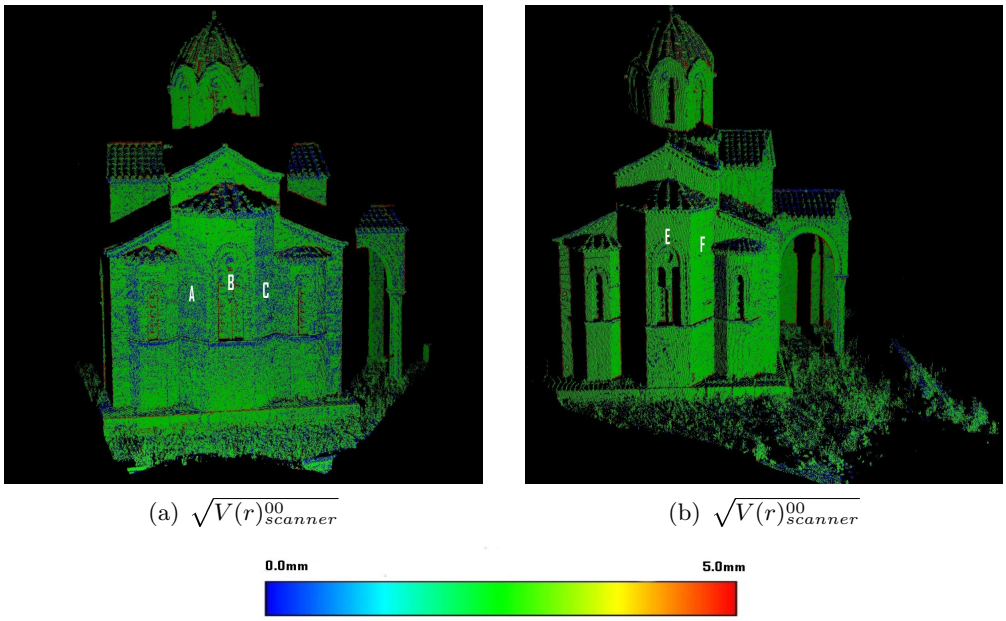
(a) Top view

(b) Side view

(c) $\sqrt{V(r)^{00}_{scanner}}$: variance in the x direction.

(d) $\sqrt{V(r)^{22}_{scanner}}$: variance in the z direction.

FIGURE 3.14: (a) is the top view of a point cloud of the Agia Sanmarina church scanned from the different location from Figure 3.13. The locations of the laser scanner and two axes of $\mathbf{O}_{scanner}$ are drawn in (a). The position uncertainties in the direction of the x and z axes calculated from Eq. 3.32 are presented in (c) and (d), respectively.



(a) $\sqrt{V(r)^{00}_{scanner}}$ (b) $\sqrt{V(r)^{00}_{scanner}}$

FIGURE 3.15: (a) and (b) are the position uncertainties of two point clouds of the Agia Sanmarina church in the direction of the y axis.

3.6 Summary

An explicit form of position uncertainty in unorganised point clouds from laser scanners is presented by means of the covariance analysis of a point. An explicit form of the variance of the estimated surface normal vector was also found. The results of this chapter will be used for an optimal weight of a registration algorithm in Section 4.4.2, finding a lower bound for the registration error of two point clouds in Section 4.4.1, and statistical inferences for evaluating the accuracy of the estimated relative transformation parameters in Section 4.6.

Chapter 4

Automated registration of point clouds : algorithms

4.1 Overview

The proposed method for the automated registration of two point clouds is briefly summarized as follows:

1. **Adjustment algorithms**: Algorithms for estimating the relative transformation parameters between point clouds. The scale factor between the point clouds is assumed to be unity in this thesis. Therefore, the number of unknown transformation parameters is six. Adjustment algorithms can be classified in two categories: point-to-point and point-to-surface based algorithms. Both of them will be utilised for the proposed method, although a point-to-surface based method is the main algorithm since the probability of finding a set of good corresponding points and their surfaces is much higher than that of finding a set of good corresponding point pairs.

2. **Finding correspondence**: The correspondence, either corresponding point pairs or point-surface pairs, is recovered using geometric primitives such as estimated surface normal vectors and geometric curvature.

3. **A lower bound of measurement error by laser scanners** : Using the results of Chapter 3 and the definition of the Cramer-Rao lower bound, a lower bound of measurement error by laser scanners will be evaluated. In addition, the lower bound for the registration error of point clouds from the laser scanner will be also evaluated.

4. **An optimal weight for a point-to-surface adjustment algorithm**: An optimal weight for a point-to-surface based registration method will be found using the position uncertainty of measurement by laser scanners discussed in Chapter 3.

5. **An outlier removal procedure and statistical tests**: An outlier removal procedure for the registration of point clouds from a laser scanner will be proposed since the estimated variance of measurement by a laser scanner was derived in the results of Chapter 3. In addition, statistical inferences for evaluating the accuracy of the estimated relative transformation parameters between point clouds will be formulated.

6. **Resampling based on clustering**: A method for finding an optimal size of neighbourhood in a point cloud will be presented that minimises the average of the variance angles of the estimated surface normal vector in a point cloud. Furthermore, using a curvature weighted resampling method, a smaller variance angle with a smaller size of neighbourhood is to be achieved without the loss of the original geometric properties of the objects.

4.2 Adjustment algorithms

4.2.1 Point-to-point based algorithms

In many applications of photogrammetry, computer vision, and remote sensing, it is a fundamental problem to recover the transformation parameters between two datasets using an adjustment algorithm, e.g. point-to-point based algorithms which minimise the sum of the distances between corresponding points (Haralick and Shapiro, 1993, Ch. 14; Wolf and Dewitt, 1999; Horn, 1987; Arun et al., 1987; Umeyama, 1991). Although Eggert et al. (1997) reported the small differences of some major adjustment algorithms in terms of stability with the presence of noise, this thesis does not attempt to develop another point-to-point based algorithm since the most important and difficult part of automatic registration is not finding an adequate adjustment algorithm but finding corresponding points. The existing point-to-point based algorithms use different mathematical methods such as Euler's representation of rotation, quaternions, linearised least-square methods, or singular value decomposition (SVD) and so on. Although Horn's (1987) method has been popular since Besl and McKay (1992) used it to develop the original ICP, Umeyama's (1991) method is used in this dissertation in order to avoid the usage of the quaternions, which makes mathematical derivations

unnecessarily complicated.

Let $\mathbf{p}_i \in P$ and $\mathbf{q}_i \in Q$, with $i = 1 \dots n$, usually called control points or corresponding points, be two point sets in m-dimensional space. It is preferable to use corresponding points or correspondence between the datasets so these terminologies will be used in this thesis. For example, the corresponding point of $\mathbf{p}_5$ is $\mathbf{q}_5$. It is necessary to estimate the transformation parameters that provide the minimum value of the mean squared distance of these two point sets. Umeyama's (1991) method is introduced from his paper without proof as follows:

Lemma 4.1. *Umeyama's method*
The mean square distance between P and Q is expressed as

$$\epsilon^2(\mathbf{R}, \mathbf{Tr}, c) = \frac{1}{n} \sum_{i=1}^{n} \parallel \mathbf{q}_i - (c\mathbf{R}\mathbf{p}_i + \mathbf{Tr}) \parallel^2 \tag{4.1}$$

where $\mathbf{R}$, $\mathbf{Tr}$, and c are rotation, translation, and scale factor, respectively. Let $\boldsymbol{\mu}_x$ and $\boldsymbol{\mu}_q$ be mean vectors of P and Q, σ_x and σ_q be variances of P and Q, and $\boldsymbol{\Sigma}_{pq}$ be the covariance matrix of X and Y. In addition, let the singular value decomposition of $\boldsymbol{\Sigma}_{xy}$ be $\mathbf{UDV}^T$ where $\mathbf{D} = \mathrm{diag}(d_i)$ and $d_1 \geq d_2 \geq \cdots d_m \geq 0$. The minimum of the mean square distance, ϵ_{min}, is expressed as

$$\epsilon_{min} = \sigma_q^2 - \frac{\mathrm{tr}(\mathbf{DS}')^2}{\sigma_p^2} \tag{4.2}$$

where

$$\mathbf{S}' = \begin{cases} \mathbf{I} & \text{if } \det(\boldsymbol{\Sigma}_{pq}) \geq 0 \\ \mathrm{diag}(1, 1, 1, \cdots, -1) & \text{if } \det(\boldsymbol{\Sigma}_{pq}) < 0 \end{cases}.$$

When $rank(\boldsymbol{\Sigma}_{pq}) \geq m - 1$, the optimal estimation of the transformation parameters that provide Eq. 4.2 as the minimum of the mean squared distance between P and Q, is uniquely determined as follows:

$$\mathbf{R} = \mathbf{USV}^T \tag{4.3}$$

$$\mathbf{Tr} = \boldsymbol{\mu}_q - c\mathbf{R}\boldsymbol{\mu}_p \tag{4.4}$$

$$c = \frac{1}{\sigma_p^2} \mathrm{tr}(\mathbf{DS}) \tag{4.5}$$

where $\mathbf{S} = \mathrm{diag}\left(1, 1, 1, \cdots, \det\left(\mathbf{UV}^T\right)\right)$.

For the implementation of Umeyama's method, LAPACK (Linear Algebra PACKage, 1999) was used for singular value decomposition and LAPACK was also utilised for the other parts of the proposed method such as finding the eigenvalues and eigen-

vectors of the covariance matrix. Note that the minimum of the mean square distance in Lemma 4.1 is valid only if two point sets have noise with the same statistical properties. As shown in Chapter 3, this assumption is unrealistic in the cases of points clouds from laser scanners so this theorem needs to be modified. However, the original form of Umeyama's method is used without modification since this point-to-point based algorithm is only used in the early stages of iterations in the proposed method. In other words, Umeyama's method is only utilised when the registration error of point clouds is larger than a threshold value.

4.2.2 Point-to-surface based algorithms

The objective of a point-to-surface based algorithm is to estimate the relative transformation parameters that minimise the distances between points and their corresponding surfaces (Chen and Medioni, 1992; Dorai et al., 1997; Gruen and Akca, 2004). In this section, a simplified version of Chen and Medioni's (1992) method will be introduced and it will be modified using the position uncertainty of measurement by laser scanners.

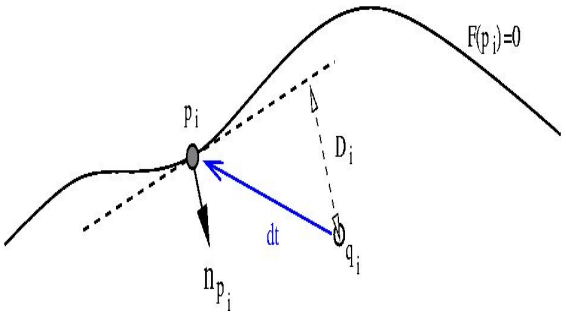


FIGURE 4.1: Chen and Medioni's algorithm for estimating the optimal relative transformation parameters that minimise the distance between points and their corresponding surfaces. $\mathbf{p}_i$ is a point on the surface $F(p_i) = 0$ and $\mathbf{n}_{p_i}$ is the true surface normal vector at $\mathbf{p}_i$. The true transformation between p_i and q_i is $d\mathbf{t}$.

The mean square distance between two corresponding points, i.e. Eq. 4.1, can be rewritten as

$$\epsilon^2(\mathbf{T}) = \frac{1}{n}\sum_{i=1}^{n} \parallel \mathbf{q}_i - \mathbf{T}\mathbf{p}_i \parallel^2 \tag{4.6}$$

where

$$\mathbf{T} = \begin{bmatrix} c\mathbf{R} & \mathbf{Tr} \\ \mathbf{0}^T & 1 \end{bmatrix} \tag{4.7}$$

and $\mathbf{p}_i$ and $\mathbf{q}_i$ are expressed in homogeneous coordinates*, $\mathbf{T} \in \Re^{4\times4}$, and c is the scale factor. The complete expression of $\mathbf{R}$ in Euler's representation is found in Appendix A. First, the idea of a point-to-surface base algorithm will be briefly reviewed and then a solution for the estimation of the transformation parameters that minimise the distances between points and their corresponding surfaces will be derived. Note that in this chapter, vectors are expressed in either three by one column matrices or matrices in the homogeneous coordinate systems, i.e. four by one column matrices. Chen and Medioni (1992) proposed an idea to iteratively minimise the sum of the distances from points to the corresponding surfaces as shown in Figure 4.1:

$$\epsilon^2_{ps\_k} = \sum_{i=1}^{n} d^2_{ps}(\mathbf{T}^k\mathbf{p}_i, S^k_i) \tag{4.8}$$

where

1. d^2_{ps} is the square distance between a point and its corresponding surface.
2. $\mathbf{T}^k$ is the transformation matrix, Eq. 4.7, at kth iteration. For example, $\mathbf{T}^{k=0}$ is expressed as

$$\mathbf{T}^{k=0} = \begin{bmatrix} c\mathbf{I}_3 & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \tag{4.9}$$

3. $S^k_i = \left\{\mathbf{s} | \left(\hat{\mathbf{n}}_{\mathbf{q}^k_i}, \mathbf{q}^k_i - \mathbf{s}\right) = 0\right\}$ is the surface at $\mathbf{q}^k_i$ in Q where $\mathbf{s}$ is a three dimensional vector.
4. $\mathbf{q}^k_i$ is the intersection point in Q with the transformed line $\mathbf{T}^k\mathbf{l}_i$.
5. $\hat{\mathbf{n}}_{\mathbf{q}^k_i}$ is the estimated surface normal vector at $\mathbf{q}^k_i$ which is on the surface S^k_i.
6. $\mathbf{l}_i$ is the line at $\mathbf{p}_i$ whose normal vector is the estimated surface normal vector at $\mathbf{p}_i$.

Note that Chen and Medioni's method is based on the assumption, which will be shown in the course of the derivation, as follows:

$$\hat{\mathbf{n}}_{p_i} \simeq \hat{\mathbf{n}}_{q_i}. \tag{4.10}$$

It states that the estimated normal vectors of corresponding points and its surface are similar. In Chen and Medioni's method, the corresponding point of $\mathbf{p}_i$ is found

* $\begin{bmatrix} p^p_i & p^y_i & p^z_i & 1 \end{bmatrix}$ where p^p_i, p^y_i, and p^z_i are x, y, and z components of $\mathbf{p}_i$. See Förstner (2004) or Kanatani (1996, Ch. 4).

using $\mathbf{l}_i$ and this method is called normal shooting in the context of computer graphics (Rusinkiewicz and Levoy, 2001). When the initial registration error, either Eq. 4.1 or Eq. 4.8, is small, Chen and Medioni's method provides better results than point-to-point based algorithms in terms of the probability of convergence to a global minimum, a smaller resultant registration error, and even the number of required iterations (Rusinkiewicz and Levoy, 2001). However, if the initial registration error is large, it tends to converge into a local minimum since the normal shooting method can no longer provide reliable corresponding point and surface pairs. In addition, the assumption of the Chen and Medioni's method, i.e. Eq. 4.10, is not valid when the initial registration error is not small. Therefore, it is proposed to remove the routine for normal shooting from Chen and Medioni's original method and to find the correspondence of point clouds by means of the geometric properties of the point cloud explained in Section 4.3 for the estimation of the transformation parameters.

Now, a least-squares equation and the solution for estimating transformation parameters which minimise the sum of square distance of points and their corresponding surfaces will be derived. Suppose that ω, ϕ and κ are the rotations around the x, y, and z axes, respectively, and t_x, t_y, and t_z are the translation parameters to the x, y, and z axes, respectively. An arbitrary transformation is then expressed as either

$$\mathbf{t} = [\omega \ \phi \ \kappa \ t_x \ t_y \ t_z]^T, \quad \mathbf{t} \in \Re^{6\times 1}$$

or $\mathbf{T} \in R^{4\times 4}$ as defined in Eq. 4.7. Note that $\mathbf{T}$ and $\mathbf{t}$ are not values but variables representing a transformation. Let $d\mathbf{t}$ be the true transformation between $\mathbf{p_i}$ and $\mathbf{q}_i$. Let $F(\mathbf{p}_i) = 0$ be an implicit equation of an arbitrary surface, $S = \{\mathbf{s} | \, (\mathbf{n}_{\mathbf{p}_i}, \mathbf{p}_i - \mathbf{s}) = 0\}$ where $\mathbf{n}_{\mathbf{p}_i}$ is the true surface normal vector at $\mathbf{p}_i$. These relationships between $\mathbf{p}_i$, $\mathbf{q}_i$, $\mathbf{t}$, and $\mathbf{T}$ are valid:

$$\mathbf{t}\mathbf{p}_i|_{\mathbf{t}=d\mathbf{t}} \equiv \mathbf{T}\mathbf{p}_i|_{\mathbf{t}=d\mathbf{t}} = \mathbf{q}_i$$

and

$$\mathbf{t} = \mathbf{0} \longrightarrow \mathbf{T} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

where the scale factor is assumed to be one. Note that the equivalent sign is used since $\mathbf{t}\mathbf{p}_i$ is not valid in terms of matrix calculus. A linearised least-squares equation for the registration between a set of the corresponding points and surfaces is expressed as

$$\mathbf{H}d\mathbf{t} = \mathbf{D} \tag{4.11}$$

where

$$\mathbf{H} = \left.\frac{\partial \mathbf{D}}{\partial \mathbf{t}}\right|_{\mathbf{t}=d\mathbf{t}} = \left.\left[\frac{\partial \mathbf{D}}{\partial \omega}\frac{\partial \mathbf{D}}{\partial \phi}\frac{\partial \mathbf{D}}{\partial \kappa}\frac{\partial \mathbf{D}}{\partial t_x}\frac{\partial \mathbf{D}}{\partial t_y}\frac{\partial \mathbf{D}}{\partial t_z}\right]\right|_{\mathbf{t}=d\mathbf{t}}, \quad \mathbf{H} \in \Re^{n\times 6}$$
$$\mathbf{D} = \left[d_{ps}(\mathbf{q}_1, S_1), \ldots, d_{ps}(\mathbf{q}_n, S_n)\right], \quad \mathbf{D} \in \Re^{n\times 1}.$$

Simply speaking, we would like to find an optimal estimate, $\hat{d\mathbf{t}}$, of the true transformation $d\mathbf{t}$. First $\mathbf{H}$, i.e. the partial derivatives of $\mathbf{D}$ with respective to $\mathbf{t}$, need to be calculated. The distance between an arbitrary point, $\mathbf{Tp}_i$, and $F(\mathbf{p}_i)$ is defined as the length of the shortest line between the point and the surface. Before proceeding, the following symbols are defined:

$$f(a)|_{a=c} = f(a=c)$$
$$f(a)\,g(a)|_{a=c} = f(a=c)\,g(a=c)$$
$$bf(a)\,g(a)|_{a=c} = bf(a=c)\,g(a=c)$$

where $f(a)$ and $g(a)$ are the functions of a and both b and c are the constants.

It is impossible to explicitly express the distance between a point and an arbitrary surface, but an approximate distance can be defined implicitly

$$D_i(\mathbf{Tp}_i) = \frac{F(\mathbf{Tp}_i)}{\parallel \nabla F(\mathbf{Tp}_i) \parallel} \tag{4.12}$$

and the distance between $\mathbf{q}_i$ and $F(\mathbf{p}_i)$ is written as

$$D_i = D_i(\mathbf{Tp}_i)|_{\mathbf{t}=d\mathbf{t}} = \left.\frac{F(\mathbf{Tp}_i)}{\parallel \nabla F(\mathbf{Tp}_i) \parallel}\right|_{\mathbf{t}=d\mathbf{t}} \tag{4.13}$$

where D_i is the estimate of the ith row of $\mathbf{D}$ (Taubin, 1991). Its partial derivative with respect to $\mathbf{t}$ is

$$\begin{aligned}\left.\frac{\partial}{\partial \mathbf{t}}\mathbf{D}_i\right|_{\mathbf{t}=d\mathbf{t}} &= F(\mathbf{Tp}_i)\frac{\partial}{\partial \mathbf{t}}\left.\left[\frac{1}{\parallel \nabla F(\mathbf{Tp}_i) \parallel}\right]\right|_{\mathbf{t}=d\mathbf{t}} + \left.\frac{1}{\parallel \nabla F(\mathbf{Tp}_i) \parallel}\frac{\partial}{\partial \mathbf{t}}F(\mathbf{Tp}_i)\right|_{\mathbf{t}=d\mathbf{t}} \\ &\simeq \left.\frac{1}{\parallel \nabla F(\mathbf{Tp}_i) \parallel}\frac{\partial}{\partial \mathbf{t}}F(\mathbf{Tp}_i)\right|_{\mathbf{t}=d\mathbf{t}}\end{aligned}$$

with the assumption that $F(\mathbf{Tp}_i)|_{\mathbf{t}=d\mathbf{t}} = F(\mathbf{q}_i) \simeq 0$. In other words, $\mathbf{q}_i$ is close enough to the surface $F(\mathbf{p}_i) = 0$ to assume that $F(\mathbf{q}_i) = 0$. By the chain rule, it is expressed

as

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{t}}\mathbf{D}_i\Big|_{\mathbf{t}=d\mathbf{t}} &= \frac{1}{\parallel \nabla F\left(\mathbf{T}\mathbf{p}_i\right) \parallel} \frac{\partial\left(\mathbf{T}\mathbf{p}_i\right)}{\partial \mathbf{t}} \nabla F\left(\mathbf{T}\mathbf{p}_i\right)\Big|_{\mathbf{t}=d\mathbf{t}} \\
&= \frac{1}{\parallel \nabla F\left(\mathbf{T}\mathbf{p}_i\right) \parallel} \frac{\partial \mathbf{T}}{\partial \mathbf{t}} \frac{\partial\left(\mathbf{T}\mathbf{p}_i\right)}{\partial \mathbf{T}} \nabla F\left(\mathbf{T}\mathbf{p}_i\right)\Big|_{\mathbf{t}=d\mathbf{t}} \\
&= \frac{1}{\parallel \nabla F\left(\mathbf{T}\mathbf{p}_i\right) \parallel} \frac{\partial \mathbf{T}}{\partial \mathbf{t}} \mathbf{p}_i \nabla F\left(\mathbf{T}\mathbf{p}_i\right)\Big|_{\mathbf{t}=d\mathbf{t}} \\
&= \frac{\partial \mathbf{T}}{\partial \mathbf{t}} \mathbf{p}_i \frac{\nabla F\left(\mathbf{T}\mathbf{p}_i\right)}{\parallel \nabla F\left(\mathbf{T}\mathbf{p}_i\right) \parallel}\Big|_{\mathbf{t}=d\mathbf{t}} \\
&\simeq \frac{\partial \mathbf{T}}{\partial \mathbf{t}} \mathbf{p}_i\Big|_{\mathbf{t}=d\mathbf{t}} \hat{\mathbf{n}}_{p_i} \\
&= \frac{\partial\left(\mathbf{T}\mathbf{p}_i\right)}{\partial \mathbf{t}}\Big|_{\mathbf{t}=d\mathbf{t}} \hat{\mathbf{n}}_{p_i}
\end{aligned} \tag{4.14}$$

since

$$\begin{aligned}
\frac{\partial\left(\mathbf{T}\mathbf{p}_i\right)}{\partial \mathbf{T}} &= \mathbf{p}_i \\
\frac{\nabla F\left(\mathbf{T}\mathbf{p}_i\right)}{\parallel \nabla F\left(\mathbf{T}\mathbf{p}_i\right) \parallel}\Big|_{\mathbf{t}=d\mathbf{t}} &= \hat{\mathbf{n}}_{q_i} \simeq \hat{\mathbf{n}}_{p_i}.
\end{aligned}$$

Let the Jacobian of $\mathbf{T}\mathbf{p}_i$ with respect to $\mathbf{t}$ be $\mathbf{J} \in \Re^{6\times 4}$. A sub-matrix of the Jacobian of $\mathbf{T}\mathbf{p}_i$ at $\mathbf{t} = d\mathbf{t}$ is given as

$$\mathbf{J}_{\{6,3\}}\big|_{\mathbf{t}=d\mathbf{t}} = \left[\frac{\partial\left(\mathbf{T}\mathbf{p}_i\right)}{\partial \mathbf{t}}\Big|_{\mathbf{t}=d\mathbf{t}}\right]_{\{6,3\}} = \left[\frac{\partial\left(\mathbf{T}\mathbf{q}_i\right)}{\partial \mathbf{t}}\Big|_{\mathbf{t}=\mathbf{0}}\right]_{\{6,3\}} = \begin{bmatrix} 0 & -q_i^z & q_i^y \\ q_i^z & 0 & -q_i^x \\ -q_i^y & q_i^x & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where q_i^x, q_i^y, and q_i^z are the components of $\mathbf{q}_i$ to the direction of the x, y, and z axes (Simon, 1996, p. 185). Note that $\mathbf{J}_{\{6,3\}}$ is introduced instead of $\mathbf{J}$ in order to express the final result in a three-by-one matrix rather than in the homogeneous coordinate system. Then Eq. 4.14 is expressed as

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{t}}\mathbf{D}_i\Big|_{\mathbf{t}=d\mathbf{t}} &= \mathbf{J}_{\{6,3\}}\big|_{\mathbf{t}=d\mathbf{t}}\, \hat{\mathbf{n}}_{p_i} \\
&= \begin{bmatrix} \mathbf{q}_i \times \hat{\mathbf{n}}_{p_i} \\ \hat{\mathbf{n}}_{\mathbf{p}_i} \end{bmatrix} \in \Re^{6\times 1}
\end{aligned}$$

and also Eq. 4.11 is rewritten as

$$\mathbf{H}d\mathbf{t} = \begin{bmatrix} \frac{\partial}{\partial \mathbf{t}}\mathbf{D}_1\big|^T_{\mathbf{t}=d\mathbf{t}} \\ \vdots \\ \frac{\partial}{\partial \mathbf{t}}\mathbf{D}_n\big|^T_{\mathbf{t}=d\mathbf{t}} \end{bmatrix} d\mathbf{t} = \mathbf{D} \tag{4.15}$$

with $\hat{\mathbf{t}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T d\mathbf{t}$ being the optimal estimation of the true transformation $\mathbf{t}$. Finally the solution of Chen and Medioni's (1992) method was obtained as above. It should be noted that two approximations which have used in the derivation are written as follows:

$$\hat{\mathbf{n}}_{p_i} \simeq \hat{\mathbf{n}}_{q_i} \tag{4.16}$$

$$F(\mathbf{q}_i) \simeq 0 \quad \text{or} \quad F(\mathbf{q}_i) \simeq F(\mathbf{p}_i) \tag{4.17}$$

which is why Chen and Medioni's (1992) method requires that the initial registration error be small.

In principle, Chen and Medioni's method can be extended to multiple point cloud registrations with a priori corresponding points and surfaces. Suppose m point clouds is given and each point cloud has n_m number of points. Assume that these point clouds were already registered by a registration algorithm like Eq. 4.15 and more precise transformation parameters can be obtained by registering all point clouds simultaneously.

Let $d\mathbf{T}$ be a matrix whose element is the transformation parameters for ith point cloud and $d\mathbf{T} = (d\mathbf{t}_1 d\mathbf{t}_2 \dots d\mathbf{t}_m)^T \in \Re^{6m\times 1}$. Let $\mathbf{D}_{total}$ be a matrix with the distances between corresponding points and surfaces as elements and $\mathbf{D}_{total} \in \Re^{N\times 1}$ where $N = \sum n_i$. In addition, let $\mathbf{H}_i$ be the matrix defined in Eq. 4.15 with corresponding points and their surface normal vectors between two point clouds and $\mathbf{H}_i \in \Re^{n_i\times 6m}$. Let $\mathbf{H}_{total}$ be defined

$$\mathbf{H}_{total} = \bigoplus_{i=1}^{m} \mathbf{H}_i$$

where $\bigoplus$ is the direct sum of matrices and $\mathbf{H}_{total} \in \Re^{N\times 6m}$. Then a least-squares equation for the simultaneous registration of m point clouds is given as

$$\mathbf{H}_{total} d\mathbf{T} = \mathbf{D}_{total} \tag{4.18}$$

or

$$\begin{bmatrix} \mathbf{H}_1 & & & \\ & \mathbf{H}_2 & & \\ & & \ddots & \\ & & & \mathbf{H}_m \end{bmatrix} \begin{bmatrix} d\mathbf{t}_1 \\ d\mathbf{t}_2 \\ \vdots \\ d\mathbf{t}_m \end{bmatrix} = \begin{bmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \\ \vdots \\ \mathbf{D}_m \end{bmatrix}.$$

with which a solution minimising the total sum of the square distances between points and corresponding surfaces in a set of point clouds may be found.

4.3 Metrics for finding correspondence

4.3.1 Surface normal vector estimation using covariance analysis

The most simple method of estimating the surface normal vector is first order three-dimensional plane fitting (Berkmann and Caelli, 1994; Sharkarji, 1998). Let $D(r_i, r)$ be the signed distance between a point, $\mathbf{r}_i = (x_i, y_i, z_i)^T$, and its plane in three dimensional space. Let $\mathbf{r} = (x, y, z)^T$ be a point on the plane whose equation is $ax + by + cz = d$. In addition, let $\hat{\mathbf{n}}$ be the estimated surface normal vector to the plane. Then $D(r_i, r)$ is given from Eq. 4.12 with $\mathbf{T} = \mathbf{I}_4$ as follows

$$\begin{aligned} D(r_i, r) &= (\hat{\mathbf{n}}, \mathbf{r}_i - \mathbf{r}) \\ &= \frac{1}{\sqrt{a^2 + b^2 + c^2}} \left[(ax_i + by_i + cz_i) - (ax + by + cz) \right] \\ &= \frac{1}{\sqrt{a^2 + b^2 + c^2}} \left[(ax_i + by_i + cz_i) - d \right] \end{aligned} \tag{4.19}$$

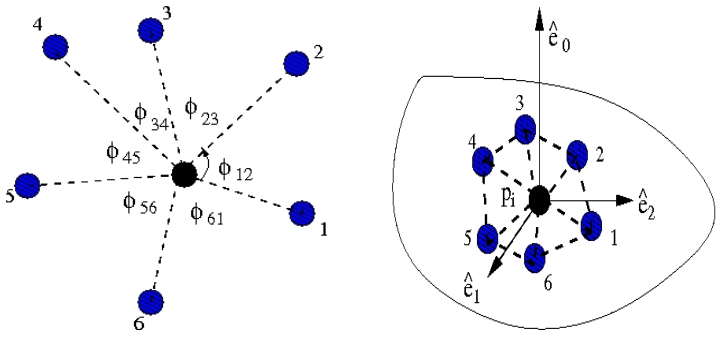
since $\mathbf{n} = (a, b, c)^T$. The cost function J to be minimised is given as

$$\begin{aligned} J &= \sum D^2(r_i, r) \\ \therefore \nabla J &= 2 \sum D(r_i, r) \end{aligned}$$

and the solution of a least-squares equation which minimises J must satisfy $\nabla J = 0$. The average of $D(r_i, r)$ that also satisfies $\nabla J = 0$, is expressed as

$$\begin{aligned} \frac{1}{k} \sum D(r_i, r) &= \frac{1}{k} \sum \left[a(x_i - x) + b(y_i - y) + c(z_i - z) \right] \\ &= a(\bar{x} - x) + b(\bar{y} - y) + c(\bar{z} - z) \\ &= 0 \end{aligned}$$

where $\bar{x}_i$ is the centroid of $x_i = \{x, y, z\}$. This means that the centroid of the data must be on the plane that minimises the sum of the distance between a point and the



(a) (b)

FIGURE 4.2: (a) Blue and black points are the projected points to the plane whose normal vector is the eigenvector of the smallest eigenvalue of its covariance matrix. Blue points are the neighbour points of a point p_i. ϕ_{ij} is the angle between two neighbouring points. (b) the eigenvectors of $COV(p_i)$, $\hat{\mathbf{e}}_{i=0\ldots2}$, are represented.

plane in the case of an unweighted least-squares method. In addition, the cost function can be rewritten as

$$J = \frac{(\mathbf{n}, \mathbf{V}\mathbf{n})}{\| \mathbf{n} \|^2} \tag{4.20}$$

where $\mathbf{V} = \sum [\mathbf{r} - \bar{\mathbf{r}}]^T [\mathbf{r} - \bar{\mathbf{r}}]$. Since J is a Rayleigh quotient (Golub and Loan, 1989), the following inequality is valid for any nonzero vector $\mathbf{n}$:

$$\lambda_{min} \leq \frac{(\mathbf{n}, \mathbf{V}\mathbf{n})}{\| \mathbf{n} \|^2} \leq \lambda_{max} \tag{4.21}$$

and the left inequality holds if $\hat{\mathbf{n}}$ is the eigenvector for the minimum eigenvalue of $\mathbf{V}$. In other words, the first order plane fitting is equivalent to the eigenvalue problem of $\mathbf{V}$. For the estimation of the surface normal vector in a point cloud from laser scanners, the covariance matrix, sometimes called the variance-covariance matrix, that is simply $\mathbf{V}$ divided by the number of data points, will be utilised in this thesis. Both the covariance matrix and $\mathbf{V}$ provide the same eigenvectors although the eigenvalues are different. Figure 4.2(a) presents a point and its neighbour points for estimating surface normal vector. Let p_j^i be the jth point of the ith point cloud. The covariance of a point and its k neighbour points, $\mathbf{COV}(p_i) \in \Re^{3\times3}$, is expressed as

$$\mathbf{COV}(p_i) = \frac{1}{k} \sum_{m=1}^{k} \mathbf{r}_m {\mathbf{r}_m}^T = \sum_{i=0}^{2} \lambda_i \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T$$

where $\mathbf{p}_{centroid}\left(p_i^1\right)$ is the centroid of p_i and its k neighbour points, $\mathbf{r}_m = \mathbf{p}_i^1 - \mathbf{p}_{centroid}\left(p_i^1\right)$, and $\hat{\mathbf{e}}_i$ is the eigenvector of the $(i+1)$th smallest eigenvalue λ_i. Since

$\mathbf{COV}(p_i)$ is a real, positive, semi-definite matrix, its eigenvalues are always greater than or equal to zero (Golub and Loan, 1989). The eigenvector of the minimum eigenvalue is the estimated normal vector of the surface formed by p_i and its k neighbour points as shown in Eq. 4.21. The other eigenvectors are the tangential vectors of the surface as shown in Figure 4.2(b). If the minimum eigenvalue is close to zero, then the surface consisting of a point and its neighbourhood is a plane.

4.3.2 Change of geometric curvature estimation

There are many ways to define curvature, e.g. through Gaussian and mean curvature methods or by using the eigenvalues of the covariance matrix (Dyn et al., 2001). The Gaussian curvature K is defined as

$$\mathrm{K} = \frac{2\pi - \sum_{i=1}^{k} \phi_{ij}}{A}$$

where ϕ_{ij} is the angle between two neighbour points and A is the average area of each triangle as shown in Figure 4.2. Note that ϕ_{ij} is the angle between the projected points to the plane whose normal vector is $\hat{\mathbf{e}}_0$. In addition, there are a number of different ways of defining the Gaussian curvature since it is dependent on the definition of A (Dyn et al., 2001). It is preferable to estimate curvature directly by using points without any preprocess such as triangulation since it is faster to use the neighbourhood of a point than to utilise the connectivity information provided by triangulation. Hoppe et al. (1992) proposed a covariance analysis method for the estimation of the normal vector with consistent orientation. The covariance analysis method has been also utilised for the estimation of local curvature estimation using the ratio between the minimum eigenvalue and the sum of the eigenvalues. It has been called by different names, for example, Pauly et al. (2002) called it the surface variance. A definition of local curvature proposed by Hoppe et al. (1992) is used in this thesis and named as the change of (geometric) curvature since this method estimates the first order differential of local curvature rather than local curvature itself (Linsen, 2001). Note that the unit of Gaussian curvature K is the inverse of the square of distance, but the unit of the change of curvature is dimensionless.

Each eigenvalue of the covariance matrix represents the spatial variation along the direction of the corresponding eigenvector. The change of curvature quantifies how much the surface formed by a point and its neighbourhood deviates from the tangential plane formed by $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$. The ratio of the minimum eigenvalue and the sum of the eigenvalues approximates the change of curvature, $M_{cc}(p_i)$, as follow

$$M_{cc}(p_i) = \frac{\lambda_0}{\sum_{i=0}^{2} \lambda_i}. \tag{4.22}$$

In addition, the local curvature, $M_{curv}(p_i)$, of p_i is estimated by the normal vectors of the point and its neighbourhood:

$$M_{curv}(p_i) = \frac{1}{k} \sum_{j=1}^{k} ||\hat{\mathbf{n}}_{p_i} - \hat{\mathbf{n}}_{neighbour\{j,p_i\}}|| \tag{4.23}$$

where $\hat{\mathbf{n}}_{p_i}$ and $\hat{\mathbf{n}}_{neighbour\{j,p_i\}}$ are the normal vectors of p_i and its jth neighbourhood point, respectively, and k is the number of the neighbour points of p_i. The unit of Eq. 4.23 is dimensionless.

Before proceeding further, terminologies used for describing the geometric properties of a point and its neighbourhood, i.e. curvature and the change of curvature will be briefly discussed. Basically it is necessary to distinguish between Eq. 4.22 and Eq. 4.23. The change of curvature, $M_{cc}(p_i)$, is defined as the ratio between the variance in the direction of $\hat{\mathbf{e}}_0$ and total variance within a point and its neighbourhood. Local curvature, $M_{curv}(p_i)$, is defined by the norm of average difference in the normal vectors in a point and its neighbourhood. Furthermore, the change of curvature can be also expressed using $M_{curv}(p_i)$ as

$$M_{cc}(p_i) = \frac{1}{k} \sum_{j=1}^{k} |M_{curv}(p_i) - M_{curv}(p_{neighbour\{j,p_i\}})|. \tag{4.24}$$

The quality of geometric primitive estimation depends on how well the neighbour points are distributed and also on the level of position uncertainty in the point cloud. Since we are dealing with unorganised point clouds, there is no guarantee that every point of a cloud has a set of evenly-distributed neighbour points. This problem may be overcome by using the angle criterion, e.g. using a threshold for the minimum angle between the neighbour points, as Linsen (2001) did for the triangulation of point clouds.

4.4 Lower bound of registration error using position uncertainty

4.4.1 Cramer-Rao lower bound of measurement by laser scanners

A method to provide a lower bound of the variance of any unbiased estimator of measurement is very useful and important since no measurement can be observed without error. Therefore, it is required to have information on the statistical quality of the measurement, e.g. variance, in order to investigate the accuracy of an estimation procedure. In this section, the Cramer-Rao lower bound of measurements by laser scanners using the results of Sections 3.5 and 3.5.1 will be explicitly expressed.

In general, a laser scanner is interpreted as an instrument which measures the position of a point in a Cartesian coordinate with the position uncertainty defined as Eq. 3.15. This approach can provide information about the position uncertainty of measurement that can be used to optimise weights for point-to-point based algorithms. In addition, this approach can be used to make a reasonable statistical inference for evaluating the quality of parameters in any estimation problem using point clouds from laser scanners.

However, regarding the registration of unorganised point clouds measured by terrestrial laser scanners, do we have a reasonably good probability of having reliable pairs of corresponding points for the registration of two point clouds scanned from different locations? A point cloud from a laser scanner is distributed in an unorganised manner and the state of its spatial organisation depends mainly on the incidence angles of the laser beam to the surfaces of objects scanned by the laser scanner. Although a flat wall is scanned from different locations of a laser scanner, a quite different density of points will be observed for the same region of the wall.

Therefore, it is desirable to interpret a laser scanner as a instrument which measures a distance between a point and its true or ideal corresponding surface. For example, if a perfect laser scanner is used, i.e. both σ_a^2 and σ_r^2 are zero, then the variance of the distance between the point and the true corresponding surface is also zero. In order to find the optimal weight for a point-to-surface algorithm and a statistical inference for the accuracy evaluation of the estimated parameters, we only need to find the variance of position in the direction of the surface normal vector.

The variance of position in the direction of $\hat{\mathbf{e}}_0$ as V^{11} has already been derived in Eq. 3.25. Furthermore, using Eq. B.13 in Theorem B.2, the variance of a point

measured by a laser scanner from n different locations, $V(p)$, is expressed as

$$\begin{aligned} V(p) &\geq \frac{1}{\sum_{i=1}^{n} \frac{1}{V^{11}(p_i)}} \\ &\simeq \left[\sum_{i=1}^{n} \frac{1}{(\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0) + (\mathbf{r}, \mathbf{V}(\hat{\mathbf{e}}_0)\mathbf{r})} \right]^{-1} \\ &= \left[\sum_{i=1}^{n} \frac{1}{\left\{ \cos^2\theta\sigma_r^2 + \sin^2\theta\ r^2\sigma_a^2 + \rho_r \frac{(\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0)}{k} \left[\frac{1}{\lambda_1}\cos^2\alpha + \frac{1}{\lambda_2}\cos^2\beta \right] \right\}} \right]^{-1} \qquad (4.25) \\ &\simeq \left[\sum_{i=1}^{n} \frac{1}{\cos^2\theta\sigma_r^2 + \sin^2\theta\ r^2\sigma_a^2} \right]^{-1}, \qquad (4.26) \end{aligned}$$

where θ is the incidence angle defined as the angle between the estimated surface normal vector and the line of sight of the laser beam, $\rho_r = \left(r^2 - r_{centroid}^2\right)$, and both α and β were defined in Figure 3.3. Note that both θ and ρ_r are variables depending on the geometric properties of a point and index i was not used for them in the above equations. Therefore, the right side of Eq. 4.25 is the Cramer-Rao lower bound of a measurement by a laser scanner $V_{CRLB}(p)$. One can find that Eq. 4.26 is the same as Eq. 22 in Tasdizen and Whitaker (2003) after correcting some minor mathematical errors and assuming that the variance of the estimated normal vector is small.

If it is assumed that the variances of position, σ^2, in each measurement from different locations of a scanner are the same, then using Eq. B.18 the CRLB of the measurement by a laser scanner is estimated as

$$V(p) \geq \frac{\sigma^2}{N}$$

where N is the numbers of measurements at a point p. In other words, the optimal measurement uncertainty is simply given as the average of measurement uncertainty rather than one which satisfies Eq. B.15. This approximation does not perform well since it does not take into account the geometry of measurement by a laser scanner (Tasdizen and Whitaker, 2003).

4.4.2 Modified Chen and Medioni's algorithm

In this section, a modified version of Chen and Medioni's algorithm will be proposed using the Cramer-Rao lower bound of the measurement by a laser scanner which was formulated in the previous section. Simply speaking, a weighted version of Chen and Medioni's algorithm will be introduced. Note that a weighted least squared method, e.g. the modified Chen and Medioni's method, is important since it provides not only the lowest possible registration error, i.e. the CRLB of the registration error of point

clouds from laser scanners, but also some statistical evidence of its closeness to the truth presented in the next section.

The Gauss-Markov theorem and linear least-squares methods will be briefly revisited, since statistical inferences or tests need to be constructed to evaluate the accuracy or unbiasness of the estimation for laser scanner data based on the results of the previous sections and Chapter 3. Detailed discussion on the Gauss-Markov theorem and linear least-squares methods can be found in textbooks such as Rao (1973, Ch. 4) and Koch (1999, Ch. 3).

Consider uncorrelated observations, $y_{i=1..n}$ with $E(y_i) = \sum_{j=1}^{m} x_{ij}\beta_j$ and $V(y_i) = \sigma^2$, where $E(\cdot)$ and $V(\cdot)$ are the expectation and variance operators, respectively, x_{ij} are known coefficients, and n is the number of observations. Let $\mathbf{Y}$, $\mathbf{X}$, and $\boldsymbol{\beta}$ be the matrices of the variables, y_i, the known coefficients, x_{ij}, and the parameters, β_i, respectively. The setup for the Gauss-Markov theorem for observation, $y_{i=1..n}$, is,

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

with $E(\boldsymbol{\epsilon}) = 0$ and $V(\boldsymbol{\epsilon}) = E(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T) = \sigma^2\mathbf{I}_n$ where $\mathbf{Y} \in \Re^{n\times 1}$, $\mathbf{X} \in \Re^{n\times m}$, $\boldsymbol{\beta} \in \Re^{m\times 1}$, and $\boldsymbol{\epsilon} \in \Re^{n\times 1}$. The cost function J can be expressed as

$$J = (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$$

and the optimal estimation of $\boldsymbol{\beta}$ minimizing the cost function is $\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}(\mathbf{X}^T\mathbf{Y})$ in the case of $V(\boldsymbol{\epsilon}) = E(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T) = \sigma^2\mathbf{I}_n$. Then a statistical inference to verify the goodness of the estimated solution is given as

$$\hat{\sigma_0}^2 = \frac{\mathbf{V}^T\mathbf{V}}{n-m}$$

and

$$\frac{\hat{\sigma_0}^2}{\sigma^2} \sim \frac{\chi^2_{df=n-m}}{n-m}, \tag{4.27}$$

where $\mathbf{V} = \mathbf{X}\boldsymbol{\beta} - \mathbf{Y}$, $\hat{\sigma_0}^2$ is the unbiased estimation of variance and $\chi^2_{df=n-m}$ is the Chi-square distribution with $n-m$ degree of freedom. Although Eq. 4.27 looks simple, it provides a strong statistical method to evaluate the quality of estimation. However, as we aware, in most practical cases, this statistical inference can not be utilised since σ^2 is unknown. Kanatani (1996) proposed a method to iteratively estimate σ^2, which he named the noise level, to remove biased estimation, and to utilise this statistical inference. Dorai et al. (1997) also proposed a method for estimating the optimal weight for the registration of point clouds using the covariance analysis of the estimated parameters of local surfaces.

If the variance of observation is a matrix $\mathbf{\Sigma} \in \Re^{n \times n}$ instead of $\sigma^2 \mathbf{I}_n$, then the cost function, the solution, and the statistical inference are given as

$$\begin{aligned} J &= (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T \mathbf{\Sigma}^{-1} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}), \\ \hat{\beta} &= (\mathbf{X}^T \mathbf{\Sigma}^{-1} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{\Sigma}^{-1} \mathbf{Y}), \end{aligned} \tag{4.28}$$

and

$$\hat{\sigma_0}^2 = \frac{\mathbf{V}^T \mathbf{\Sigma}^{-1} \mathbf{V}}{n-m} \sim \frac{\chi^2_{df=n-m}}{n-m}, \tag{4.29}$$

respectively. In addition, the minimum covariance matrix of $\mathbf{C}_{\hat{\boldsymbol{\beta}}}$ is given as

$$\mathbf{C}_{\hat{\boldsymbol{\beta}}} = (\mathbf{X}^T \mathbf{\Sigma}^{-1} \mathbf{X})^{-1}. \tag{4.30}$$

This is an equivalent expression to Eq. 3.19 except that the above equation is a matrix with rank($\mathbf{X}$) but the rank of Eq. 3.19 is rank($\sum_{i=0}^{2} \hat{\mathbf{e}}_i \hat{\mathbf{e}}_i^T$) -1. Suppose we would like to estimate a parameter using a linear or linearised least-squares method. If the variance of every datum is known, then Eq. 4.29 can be utilised as a reasonable validation method for the estimated parameters. If the estimated variances of the data are not adequate, the estimated variance will not pass the statistical test in Eq. 4.29.

However, it is more preferable to directly estimate the variance of measurements from the characteristics of an instrument, e.g. a laser scanner, than to estimate the variance indirectly from the data, e.g. Kanatani (1996) and Dorai et al. (1997). Of course, this approach can not be used in every application since it is virtually impossible to obtain an explicit expression of the variance of the measurement in some cases. However, in the case of laser scanners, a good estimation of the position uncertainty of measurement was explicitly formulated as presented in Chapter 3.

It is desirable to modify the original idea of Chen and Medioni using the optimal variances obtained from careful consideration of the geometry of the laser scanner measurement system as presented in Chapter 3 to provide some validation for the quality of the estimated transformation parameters. Chen and Medioni's point-to-surface algorithm proved in Section 4.2.2 can be rewritten as

$$\mathbf{H} d\mathbf{t} = \begin{bmatrix} \frac{\partial}{\partial \mathbf{t}} \mathbf{D}_1 \Big|^T_{\mathbf{t}=d\mathbf{t}} \\ \vdots \\ \frac{\partial}{\partial \mathbf{t}} \mathbf{D}_n \Big|^T_{\mathbf{t}=d\mathbf{t}} \end{bmatrix} d\mathbf{t} = \mathbf{D}$$

where

$$\frac{\partial}{\partial \mathbf{t}} \mathbf{D}_i \Big|_{\mathbf{t}=d\mathbf{t}} = \begin{bmatrix} \mathbf{q}_i \times \hat{\mathbf{n}}_{\mathbf{p}_i} \\ \hat{\mathbf{n}}_{\mathbf{p}_i} \end{bmatrix}$$

$\mathbf{p}_i$ and $\mathbf{q}_i$ are a pair of corresponding points and $\hat{\mathbf{n}}_{p_i}$ is the estimated surface normal vector at $\mathbf{p}_i$. The variance matrix of n corresponding point pairs, $\mathbf{\Sigma}$, is written as

$$\mathbf{\Sigma}^{-1} = \text{diag}\left(\frac{1}{V(p_1)}, \frac{1}{V(p_2)}, \cdots, \frac{1}{V(p_n)}\right) \tag{4.31}$$

where $V(p_i)$ was defined in Eq. 4.25. Then the optimal estimator of the true transformation and a statistical test are given by Eqs. 4.28 and 4.29, respectively. This is a simple least-squares method with an optimal weight for an datum.

The use of a diagonal matrix, i.e. no correlation between the corresponding point pairs, is justified by recalling that corresponding points from two separate point clouds are statistically independent. The Cartesian coordinates of a point calculated from the initial spherical coordinate measurements of a laser scanner have some degrees of correlation between the coordinates, i.e. the x, y, and z coordinates. An empirical measurement of the true correlation between the observed Cartesian coordinates from the laser scanner is possible from a least-squares method, although it is difficult to recover the true correlation between the spherical coordinate measurements without a procedure for the calibration of the laser scanner. To justify the use of Eq. 4.31, a low correlation between the x, y, and z components of one measurement is not required. Instead a low correlation between the corresponding point pairs from different point clouds is required, which can be easily justified.

4.5 An outlier removal procedure: the RANSAC

When the registration error is small, it is expected to have a set of very reliable corresponding point and surface pairs for the modified Chen and Medioni's algorithm. However, it does not mean that there are no outliers, which precludes an adjustment algorithm from estimating accurate relative transformation parameters of point clouds. Therefore, a method for removing outliers is required, e.g. the RANSAC (Random Sample Consensus) proposed by Fischler and Bolles (1981). Many procedures of outlier detection use as large a number of data points as possible and then detect outliers among them. However, the RANSAC procedure tries to use as small a size of data as possible if the problem allows. More discussion on the RANSAC procedure can be found in the original paper of Fischler and Bolles (1981), Hartley and Zisserman (2001), and Bretar and Roux (1999).

Basically the RANSAC procedure, an iterative outlier removal procedure, first uses a subset of data, S_i, for estimation as shown in Figure 4.3. A consensus set is defined so that a set represents both the data and a subset of the data, i.e. S_i. One can always find a consensus set of a subset with a threshold, T_{ls}. However, the selected consensus set must be a proper consensus set which represents the statistical properties

of the data. After selecting another subset, the consensus set of S_i, whose elements are within the variance from the estimated results, it decides whether the current iteration is large enough to declare the validity of the current estimated parameters based on the number of outliers in the data. If the ratio of outliers in the data is unknown, it can be iteratively recovered (Hartley and Zisserman, 2001).

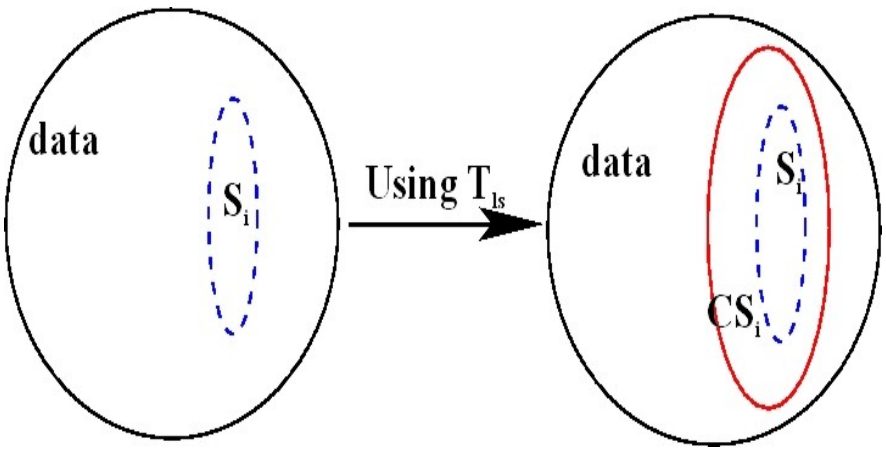


FIGURE 4.3: S_i and CS_i are a subset of the data and its consensus set, respectively. In addition, T_{ls} is a threshold for finding the consensus set of S_i, i.e. CS_i.

The conventional RANSAC procedure can be also restated as follows:

1. Given a least-squares problem which requires at least n data points[†], randomly select a subset of the data S_i in the ith iteration with the condition that the number of points in the subset, n_S, needs to be greater than n. Then estimate the parameters of the least-squares problem using S_i.

2. Determine a set CS_i whose errors is smaller than a threshold T_{ls} usually given by the variance of the estimation with S_i. The set CS_i is named the consensus set of S_i after Fischler and Bolles (1981). It means that CS_i represents S_i in terms of estimating the parameters of the least-squares problem.

3. Let i_{max} be the maximum number of iterations that is required to find at least a proper consensus set, which can be calculated from a priori information on the ratio of outliers in the data. If i_{max} is smaller than the current iteration, then select a new subset S_{i+1} and repeat the above processes.

4. If i_{max} is greater than the current number of iterations, then recalculate the parameters using the proper consensus set CS_i and the new parameters are the final result of the RANSAC procedure.

[†]For example, two points are required for two-dimensional line fitting.

There are three unspecified thresholds: (1) n_S, the number of points in a subset, (2) T_{ls}, the error tolerance for finding the consensus set of a subset, and (3) i_{max}, the maximum number of iterations to find a proper consensus set. The second and third thresholds need to be carefully selected since finding a proper consensus greatly depends on these thresholds.

First, let us discuss the maximum number of iteration i_{max}, supposing that w is the probability that any selected point is within T_{ls} of the estimated parameter. In addition, let $b = w^n$, which is the probability of selecting a proper consensus set with n points: the minimum number of points required for the estimation of the problem. The required number of iterations for the RANSAC procedure that provides at least one proper consensus set for the data with a certain level of confidence needs to be calculated. This can be deduced from considering the probability to obtain l tails of a coin from m trials, $P(l, m)$, is given as

$$P(l, m) = c^l(1 - c)^{m-l}\frac{m!}{l!(l - m)!}$$

where c is the probability for the tail of the coin. Let us calculate the average iteration for selecting at least a proper consensus set. This is the same case to find the probability of obtaining one tail of the coin from $m = 1 \ldots \infty$ trials with the probability of b. Therefore, the expected value of iteration to find a proper consensus is expressed as

$$\begin{aligned} E(i) &= P(1,1) + P(1,2) + \cdots + P(1,i) + \cdots \\ &= \sum_{i=0}^{\infty} b(1-b)^{i-1} i \\ &= b\sum_{i=0}^{\infty} i(1-b)^{i-1} \\ &= b\frac{1}{b^2} = \frac{1}{b} = w^{-n}, \end{aligned} \tag{4.32}$$

using $\frac{1}{b^2} = \sum_{i=0}^{\infty} i(1-b)^{i-1}$.

In a similar manner, the number of iterations in which we have at least one proper consensus with probability of z can be evaluated. It is simply given by

$$\begin{aligned} (1-b)^{i_{max}} &= 1 - z \\ \therefore i_{max} &= \frac{\log(1-z)}{\log(1-b)} \end{aligned} \tag{4.33}$$

For example, if $w = 0.2$ and $n = 4$, then we need about 1438 iterations to obtain a ninety percent confidence level.

Now, the error tolerance for finding the consensus set of a subset, T_{ls}, will be

discussed. Bearing in mind that the RANSAC procedure is used for selecting reliable corresponding point and surface pairs for a point-to-surface algorithm, i.e. the modified Chen and Medioni's, the error metric for the modified Chen and Medioni's algorithm is the distance between a point and its corresponding surface which has one degree of freedom as suggested by Hartley and Zisserman (2001, pp. 101 - 113) and Kanatani (1996, Ch. 5). Assuming that the distances between points and their corresponding surfaces follow a Gaussian distribution with zero mean and standard deviation from the selected samples, the error tolerance for finding the consensus set of a subset, T_{ls}, is expressed as

$$\frac{{T_{ls}}^2}{\sigma^2_{sample}} = \chi^2_{1,a} \tag{4.34}$$

where $\chi^2_{r,a}$ is the Chi-square distribution with significance level of $a\%$ and r degrees of freedom and σ^2_{sample} is the variance of the distances of the selected samples, i.e. the corresponding point pairs.

In the RANSAC procedure, a priori knowledge of data, i.e. the variance of the error in the data, is required in order to determine whether a member of the data is an inlier [‡] or not. Furthermore, the ratio of inliers, w, must be updated with independently evaluated variance of the data. The variance of measurement by a laser scanner was already formulated in Chapter 3. In addition, the combined variance of more than one measurement by a laser scanner was explicitly estimated in Eq. 4.25. Therefore, a threshold for inlier detection in the proposed RANSAC procedure, $T_{\text{RANSAC\_inlier}}$, is expressed as follows

$$T_{\text{RANSAC\_inlier}} = \left[\frac{\sum_{j=1}^{n_S} var\_n(j)}{n_S - 1} \right]^{\frac{1}{2}} \tag{4.35}$$

where

$$var\_n(j) = \left[\sum_{i=1}^{n} \frac{1}{\left\{ \cos^2\theta\sigma_r^2 + \sin^2\theta\ r^2\sigma_a^2 + \rho_r \frac{(\hat{\mathbf{e}}_0, \mathbf{V}(\mathbf{r})\hat{\mathbf{e}}_0)}{k} \left[\frac{1}{\lambda_1}\cos^2\alpha + \frac{1}{\lambda_2}\cos^2\beta \right] \right\}} \right]^{-1},$$

j is the index for a selected subset in jth iteration, n is the number of measurement by a laser scanner, θ is the incidence angle as defined in Eq. 4.25. Note again that both θ and ρ_r are variables which depend on the geometric properties of a point and are used without index i. The proposed RANSAC procedure is presented in Algorithm 1. This

[‡] A set of inliers is defined as a group of observation which seems to belong to the pattern of variability produced by the majority of data (Johnson and Wichern, 1998, pp. 200-204)

outlier removal procedure can be utilised not only for the registration of point clouds from laser scanners but only for any estimation algorithm, if some information on the radial and angular uncertainties of measurement by laser scanners is known. The most important fact in the proposed RANSAC procedure is that $T_{\text{RANSAC\_inlier}}$ is used as a threshold for evaluating the ratio of inliers, i.e. w. Note that the required number of iterations to provide at least a proper consensus set, i_{max}, is a function of w as given in Eq. 4.33.

Algorithm 1 RANSAC procedure for estimating the transformation parameters using modified Chen and Medioni's algorithm

Require: Corresponding point pairs are $data1$ and $data2$. A set of the surface normal vectors of $data1$ is $data1\_normal$.

$iteration \leftarrow 0$
$stat\_test \leftarrow$ FALSE
repeat
  Select a subset S_i and save their coordinates and normal vectors in temporary storage such as t_data1, t_data2, and t_data1_normal.
  Estimate the transformation parameters, **qd**, between the corresponding point pairs using modified Chen and Medioni's method.
  Transform $data2$ and t_data2 using the estimated transformation parameters.
  Evaluate the distances between $data2$ and the surfaces described by $data1$ and $data1\_normal$ and the standard deviation of them.
  $T_{ls} \leftarrow$ Eq. 4.34
  Count the number of inliers using $T_{\text{RANSAC\_inlier}}$ defined in Eq. 4.35 and assign it n_{inlier}.
  $w \leftarrow \frac{n_{inlier}}{n_S}$
  $i_{max} \leftarrow$ Eq. 4.33
  if The estimated parameters satisfy both Eqs. 4.36 and 4.37 **then**
    $stat\_test \leftarrow$ TRUE
  end if
  $i \leftarrow i + 1$
  if $i > T_{iteration}$ **then**
    break
  end if
until $i_{max} < iteration$ AND $stat\_test ==$ TRUE
if $stat\_test ==$ TRUE **then**
  qd $\leftarrow$ The estimated transformation parameters with all inliers, i.e. the consensus set.
  return(qd, TRUE)
else
  qd $\leftarrow$ **0**
  return(qd, FALSE)
end if

4.6 Statistical inferences for registration accuracy evaluation

In the final stage of the proposed method for the registration of point clouds, two statistical inferences for evaluating the quality of the estimated transformation parameters are utilised (Koch, 1999, Ch. 4; Kanatani, 1996, Ch. 7). Briefly, recall what kinds of statistical properties can be estimated from point clouds or a set of corresponding point and surface pairs. They are summarised as follows:

- The Cramer-Rao lower bound of a measurement by a laser scanner, $V_{CRLB}(p)$.
- The mean or standard deviation of the distances between a set of corresponding point and surface pairs with N_c numbers of points.
- The mean or standard deviation of the distances between a subset of the corresponding point and surface pairs, i.e. S_i, with n_S numbers of points.
- The standard deviation of the weighted residual $S0$, i.e. Eq. 4.29, in the modified Chen and Medioni's method:

$$S0 = \left[\frac{\left(\mathbf{V}^T \mathbf{\Sigma} \mathbf{V}\right)}{N_c - 6} \right]^{\frac{1}{2}}$$

  where $\mathbf{V}$ and $N_c - 6$ are the matrix of the residuals and the degrees of freedom in the modified Chen and Medioni's method, respectively. Note that six transformation parameters, i.e. three for rotation and three for translation, need to be estimated.

Using $V_{CRLB}(p)$ and the standard deviation of the distances between a set of corresponding point and surface pairs σ_{ps}^2, a statistical inference is expressed as

$$\frac{\max\left(V_{CRLB}(p), \sigma_{ps}^2\right)}{\min\left(V_{CRLB}(p), \sigma_{ps}^2\right)} \sim \mathbb{F}_{N_c-1, N_c-1, c} \tag{4.36}$$

where $\mathbb{F}_{a,b,c}$ is the Fisher distribution of a and b degrees of freedom with significance level of $c\%$. This statistical inference demonstrates that the registration error calculated from a set of the selected corresponding point and surface pairs is close enough to the theoretical lower bound of the registration error, i.e. $V_{CRLB}(p)$. One can use S_i for the calculation of the σ_{ps}^2 and it can provide another statistical test with the Fisher distribution between the corresponding point and surface pairs and S_i. Note that $V_{CRLB}(p)$ and σ_{ps}^2 are statistically independent since $V_{CRLB}(p)$ are obtained from the theoretical results of Chapter 3 whereas σ_{ps}^2 is the standard deviation of the distances between selected corresponding point and surface pairs, i.e. the sample.

The second statistical inference is a standard statistical test for weighted least-square problems:

$$S0^2 \sim \frac{\chi^2_{n_S-6,c}}{n_S-6}. \tag{4.37}$$

The first inference is a global reliability test of the estimated transformation parameters since it uses all selected corresponding point and surface pairs whereas the second test utilises only a subset of the selected corresponding point and surface pairs. In addition, the second inference is a statistical test within the least-squares method but the first can be viewed as the comparison between two samples which may follow different kinds of statistics.

4.7 A resampling method based on clustering

4.7.1 Angular uncertainty of surface normal vector estimation

The variance angle of the estimated surface normal vector was defined and derived in Section 3.5.1 and it is rewritten as

$$\begin{aligned}\Psi^{i=1...2} &= \tan^{-1}\left\{\left[\frac{(\hat{\mathbf{n}}, \mathbf{V}(\mathbf{r}^{query})\hat{\mathbf{n}})}{k\lambda_i}\right]^{\frac{1}{2}}\right\} \\ &= \tan^{-1}\left\{\left[\frac{\cos^2\theta\sigma_r^2 + \sin^2\theta r^2\sigma_a^2}{k\lambda_i}\right]^{\frac{1}{2}}\right\}\end{aligned} \tag{4.38}$$

where $\mathbf{r}_{query}$ is a point we are interested in. Note that we have two variance angles in the case of three-dimensional point clouds, as shown in Figure 3.8. The variance angles are inversely proportional to both the size of neighbourhood and the eigenvalues of $\hat{\mathbf{e}}_{i=1..2}$. In case of points in a low curvature region, if a size of neighbourhood is large, then the eigenvalues of $\hat{\mathbf{e}}_{i=1..2}$ are large as well. Therefore, we have generally a smaller variance angle in a low curvature region. In the case of points in a high curvature region, things become more complicated. In addition, a point on an edge region has the same properties as a point in a low curvature region. However, the eigenvalues of a point close to the edge region have a nonlinear relationship with the size of neighbourhood (Mitra et al., 2004). We can still evaluate the precision of estimated normal vectors in both low and high curvature regions by investigating the behaviour of the eigenvalues of the covariance matrix (Bae et al., 2005).

An optimal size of neighbourhood can be defined either globally or locally. An optimal size of neighbourhood means that it guarantees the maximum variance angle, Eq. 4.38, of a point cloud is smaller than a threshold. The global optimal size of neighbourhood minimises the average variance angle of points in a point cloud. On the other

hand, the local optimal size of neighbourhood minimises the variance angle of a point so every point in a point cloud has its own local optimal size of neighbourhood. Bae et al. (2005) investigated both the global and local optimal sizes in a point cloud and proposed an algorithm to evaluate a local optimal size of neighbourhood in a robust manner. Since using the global optimal size of neighbourhood makes the implementation of the proposed method much simpler, an optimal size for a point cloud will be used rather than using a different size of neighbourhood for every point.

The average variance angles of the Agia Sanmarina church point clouds used in Chapter 3 are presented in Figure 4.4, in which East and northeast represent the data presented in Figure 3.1(a) and Figure 3.1(b), respectively. In both point clouds, we have about a degree of average variance angle with the size of 50 neighbour points and variance angle is getting smaller as the size of neighbourhood increases. A specified tolerance can be set for the global optimal neighbourhood size which provides the tolerance as the maximum variance angle over a point cloud, although there is still a chance to have less precise surface normal vectors in high curvature regions.

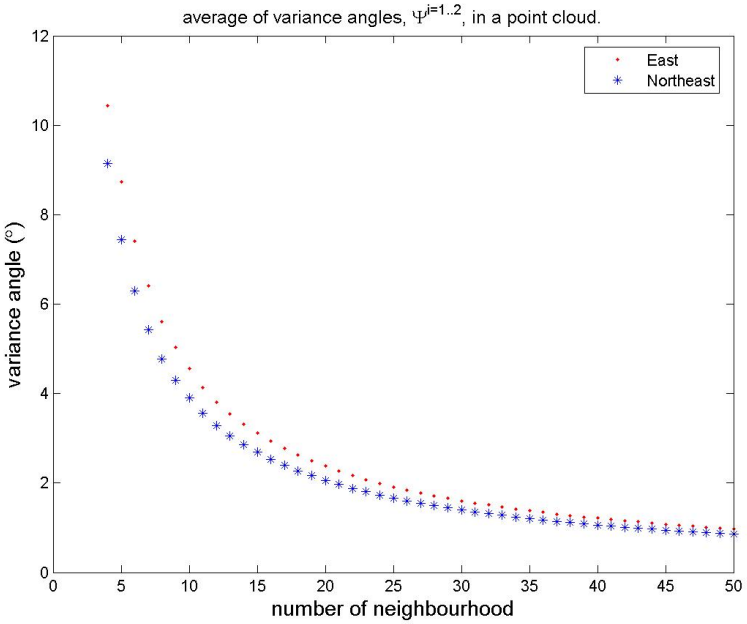


FIGURE 4.4: Average variance angles in two point clouds of the Agia Sanmarina church.

The colour maps of the variance angles of the estimated surface normal vector with 30 neighbour points in the cases of the Agia Sanmarina church and the Ayuthaya Buddha statue are presented in Figures. 4.5 and 4.6, respectively. In Figure 4.5, about one degree difference in the variance angle is observed within the overlapping regions of the point clouds. On the other hand, the point clouds of the Ayuthaya Buddha statue have about 2° difference in the variance angle in the overlapping regions. This is not

surprising if we remember that the Riegl LMS-Z210's radial and angular uncertainties are much greater than the Leica HDS2500's. As deducing from Figure 4.4, a neighbourhood size larger than thirty is required in order to ensure a smaller average variance angle than 2° in these point clouds. However, using a large size of neighbourhood will decrease the efficiency of a registration algorithm. A tree structure for neighbourhood search is a process with $O(n \log n)$ efficiency for finding the nearest neighbour with n data. From my experience with a kd-tree library developed by Arya et al. (1998), less than about 30 neighbours for a point is reasonable for a point cloud with about one million points. Therefore, to reduce the density of a point cloud by using the variance angle of the estimated surface normal vector is useful to achieve a smaller average variance angle in the point cloud than the original which is shown in Figure 4.4. By doing so, a point and its neighbourhood which occupy a much larger area than the original point cloud can be utilised with the same size of neighbourhood.

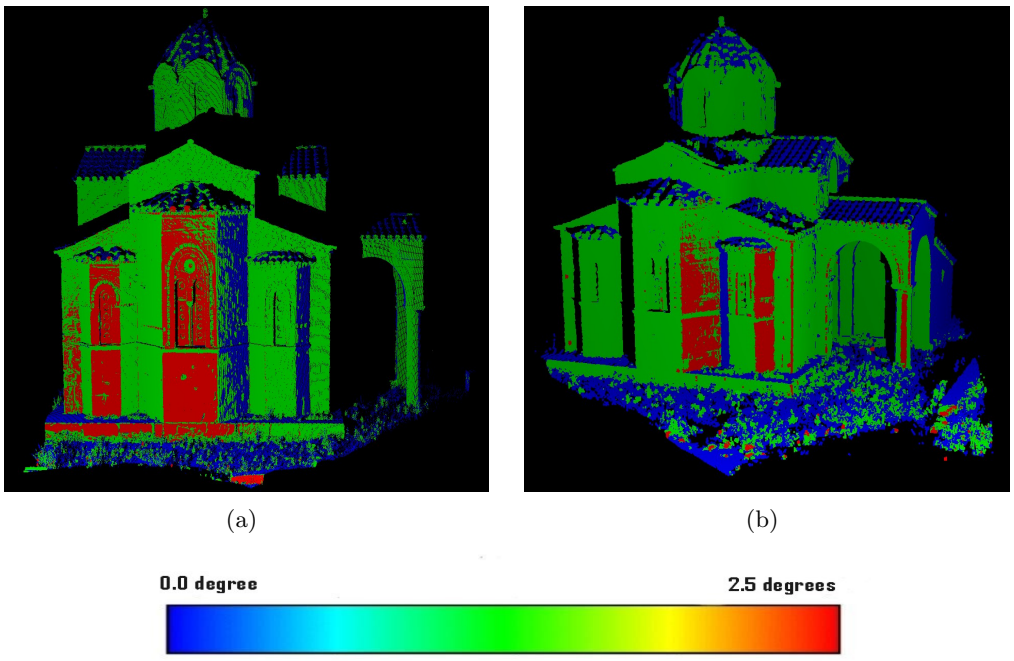


FIGURE 4.5: Variance angles of the Agia Sanmarina church in Greece scanned from different locations. The size of the neighbourhood was 30.

4.7.2 Resampling method for raw point clouds

A good method for resampling, usually down-sampling, can increase efficiency in terms of storage and processing of point clouds for laser scanners (Garland and Heckbert, 1997; Pauly et al., 2002). The most important benefit is that a smaller variance angle in the estimation of surface normal vector is obtained by resampling. Note that a smaller

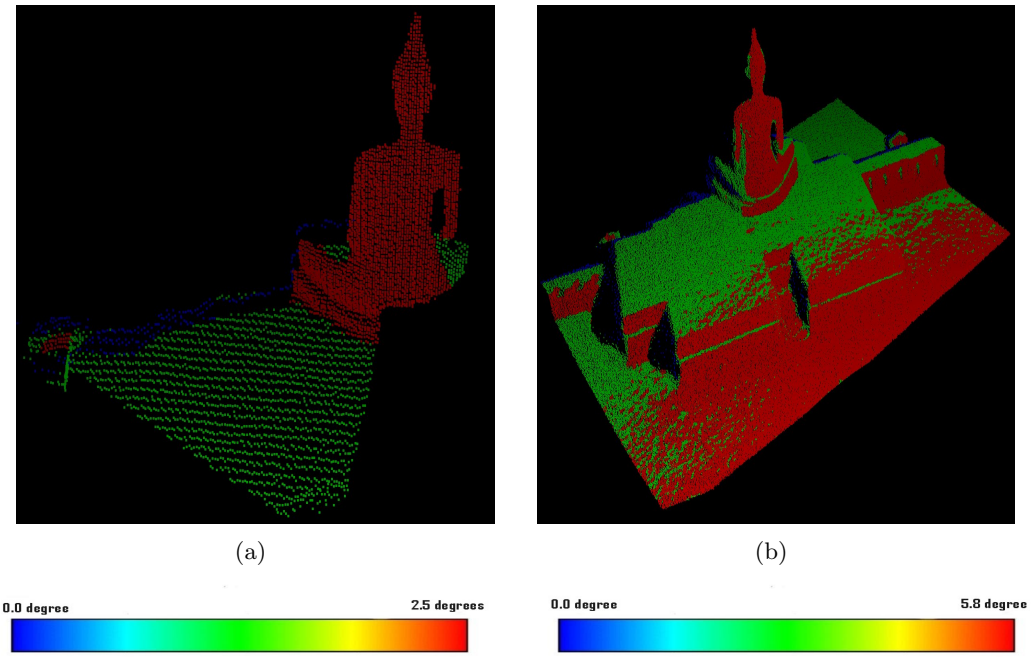


FIGURE 4.6: Variance angles of a Buddha statue, Ayuthaya, in Thailand scanned from different locations. The size of the neighbourhood was 30.

variance angle of the estimated surface normal vector can be made by an up-sampling method, although it does not increase efficiency in processing of point clouds for the registration of point clouds. Furthermore, a resampling method can provide a greater probability to select the correct corresponding point and surface pairs by estimating the surface normal vector with higher precision. In this section, a simple method for the resampling of a point cloud from laser scanners using the change of curvature will be presented. In addition, it will be shown that it provides a lower variance angle than the original point cloud or the uniformly resampled point cloud.

The proposed resampling method is similar with a curvature-adaptive clustering method proposed by Pauly et al. (2002), except that the variance angle in the estimation of surface normal vector is used and the centroid of a cluster is weighted by the changes of curvature of the members of the cluster. Note that Pauly et al. (2002) resampled the original point cloud only based on the local spatial density with less consideration on the change in the variance angle after resampling of the point cloud. A brief description of the proposed method for resampling is as follows:

1. Divide a point cloud into a number of subsets, i.e. clusters, based on the change of curvature as shown in Figure 4.7.

2. The size of a cluster is limited by a maximum difference in the change of curvature from a query point as well as a defined maximum size of clusters. The size

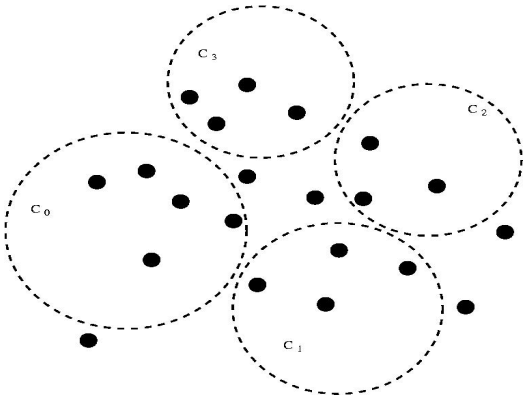


FIGURE 4.7: A clustering method using region-growing for resampling of point clouds from laser scanners.

of a cluster is iteratively increased. Let $C_i^{iter=k}$ be a cluster at the kth iteration for the cluster C_i and $C_i^{iter=0}$ be an initial cluster for C_i defined by either the number of neighbourhood or its radius. $\langle M_{cc}(C_i^{iter=k-1})\rangle + \langle M_{cc}(C_i^{iter=k-1})\rangle_{std}$ is recommended as a maximum difference in the change of curvature for a cluster $C_i^{iter=k}$, where $\langle M_{cc}(C_i^{iter=k-1})\rangle$ and $\langle M_{cc}(C_i^{iter=k-1})\rangle_{std}$ are the mean and standard deviation in a cluster $C_i^{iter=k-1}$, respectively.

3. Assume that the distribution of M_{cc} in a cluster follows a Gaussian probability density function with $\langle M_{cc}(C_i)\rangle$ and $\langle M_{cc}(C_i)_{std}\rangle$ as the mean and standard deviation. Then the centroid of C_i weighted by the change of curvature is expressed as

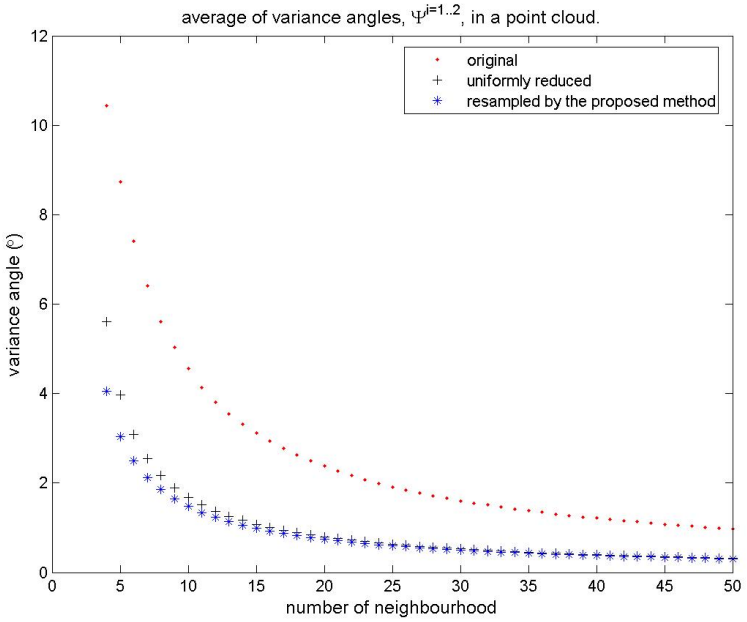
$$\textbf{Centroid}\,(C_i) = \sum_{i=1}^{m} \omega_i \mathbf{r}(p_i)$$

where

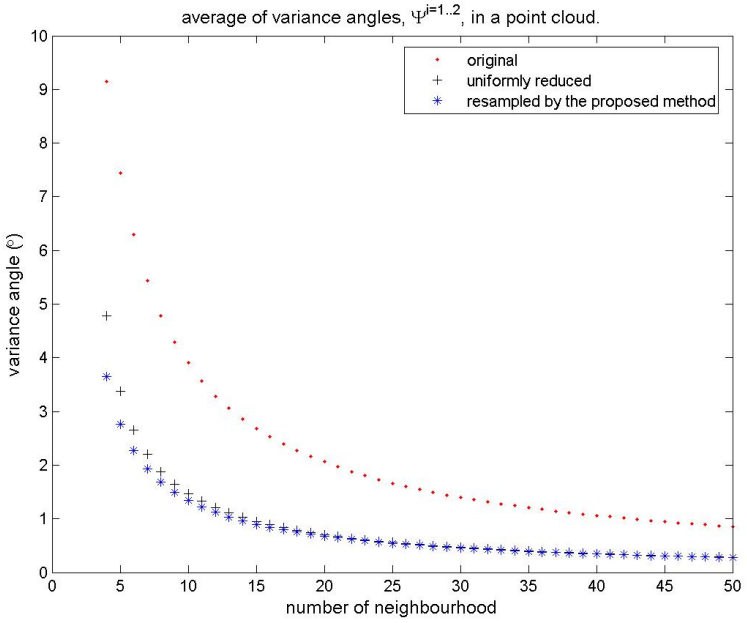
$$\omega_i = \frac{\exp\{-c_i\}}{\sum_{j=1}^{m}\exp\{-c_i\}}$$
$$c_i = \frac{[M_{cc}(p_i) - \langle M_{cc}(C_i)\rangle]^2}{2\langle M_{cc}(C_i)_{std}\rangle^2}.$$

4. Replace each cluster by its centroid weighted by the changes of curvature.

5. There may be some points which do not belong to any cluster as shown in Figure 4.7. Although they can be assigned to a closest cluster as suggested by Pauly et al. (2002), these points are left as clusters in the proposed resampling method in order to avoid unnecessary smoothing effect.



(a) East



(b) Northeast

FIGURE 4.8: Average variance angles of a point cloud plotted against the number of neighbourhood points for the originals, the uniformly reduced point clouds, and the point clouds resampled by the proposed resampling method.

point cloud	number of points	average variance angle(°)
original	486340	1.60
uniformly reduced	48634	0.53
resampled by clustering (1)	45505	0.50
resampled by clustering (2)	14212	0.29

method	k	maximum k	maximum radius (cm)
resampled by clustering (1)	40	30	20
resampled by clustering (2)	100	100	50

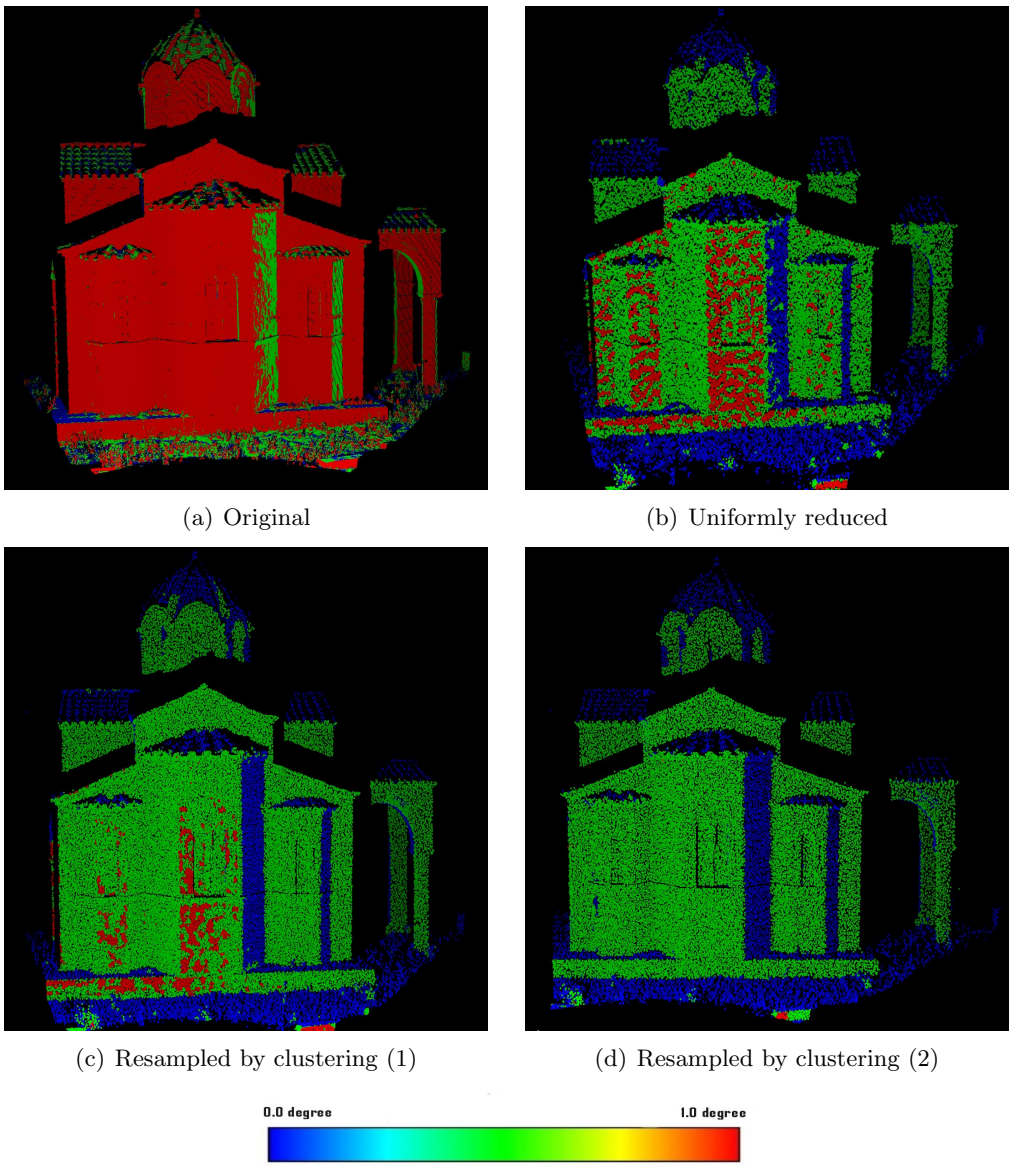
TABLE 4.1: Information about the point cloud, East, resampled by various methods. k is the size of neighbourhood points.

The average variance angle over a point cloud is given, from Eq. 4.38, as

$$\langle \Psi \rangle = \frac{1}{N} \sum_{j=1}^{N} max(\Psi_{p_j}^{i=1}, \Psi_{p_j}^{i=2})$$

where N is the total number of points in a point cloud and $\Psi_{p_j}^{i=1\ldots2}$ is the variance angles of the jth point, p_j, in the point cloud. The average variance angles of the Agia Sanmarina church dataset are presented in Figure 4.8 in the cases of the original, the uniformly reduced, and the resampled point clouds. The average variance angles in both the uniformly reduced and resampled point clouds with the same size of neighbourhood are smaller than that of the original, because a point in either the uniformly reduced or resampled point clouds occupies a larger area than in the original. The difference in variance angles between the uniformly reduced and the resampled point clouds is not significant except of the small sizes of neighbourhood since the average variance angle over an entire point cloud is presented. The colour maps of the average variance angles of East and Northeast are presented in Figures 4.9 and 4.10, respectively. In other words, a smaller number of a higher variance angle can be observed by using the proposed resampling method. Some details of the average variance and angles are presented in Tables 4.1 and 4.2. The advantage of the resampled point clouds is observed in Figures 4.9 and 4.10. They show that the resampled point clouds have a more regular structure, i.e. less unorganised, than either the originals or the uniformly reduced point clouds.

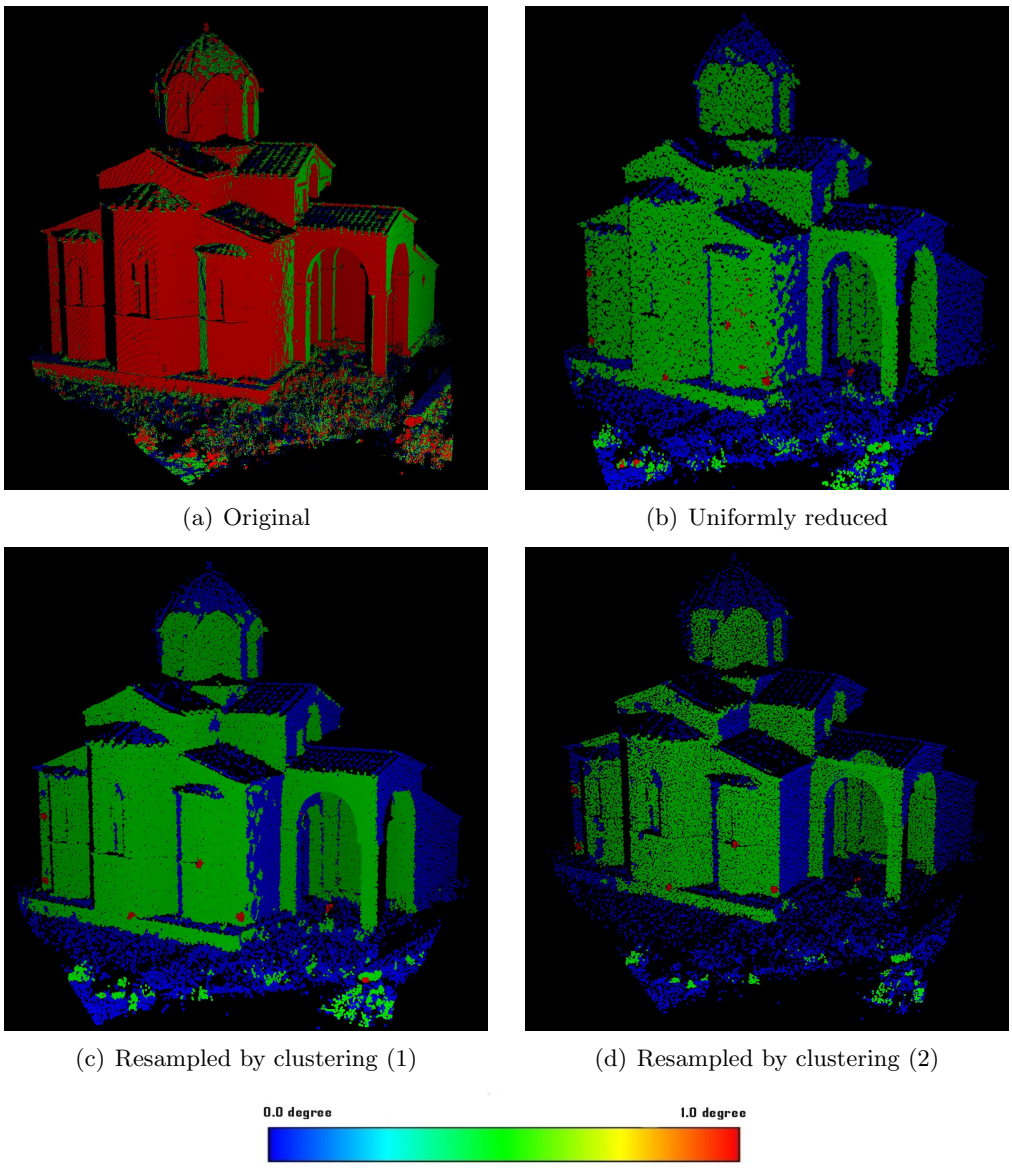
As mentioned earlier, at least one threshold value for either the maximum size or radius of allowed clusters is needed. Since these threshold values are not sensitive to datasets as shown in the bottom of Tables 4.1 and 4.2, the properties of these thresholds was not investigated any further. Furthermore, the main purpose of introducing this resampling method is providing the low variance angle of selected corresponding points and it was demonstrated to be achieved as shown in Figures 4.9 and 4.10.



(a) Original (b) Uniformly reduced

(c) Resampled by clustering (1) (d) Resampled by clustering (2)

FIGURE 4.9: Colour maps of the variance angle of a point in East of the Agia Sanmarina church data.



(a) Original

(b) Uniformly reduced

(c) Resampled by clustering (1)

(d) Resampled by clustering (2)

FIGURE 4.10: Colour maps of the variance angle of a point in Northeast of the Agia Sanmarina church data.

point cloud	number of points	average variance angle(°)
original	453142	1.40
uniformly reduced	45315	0.47
resampled by clustering (1)	42549	0.46
resampled by clustering (2)	29509	0.38

method	k	maximum k	maximum radius (cm)
resampled by clustering (1)	40	20	20
resampled by clustering (2)	30	30	20

TABLE 4.2: Information about the point cloud, Northeast, resampled by various methods. k is the size of neighbourhood points.

4.8 Descriptions of the proposed registration algorithms

A description of the proposed method for the registration of two point clouds is presented in Algorithm 2 where ϵ is the registration error and ! is the inverse of a boolean. Algorithm 2 is named the Geometric Primitive ICP with the RANSAC (GP-ICPR) in this dissertation. In addition, a simplified version of Algorithm 2 is presented in Algorithm 3, the Geometric Primitive ICP (GP-ICP).

The list of threshold values used in either GP-ICPR or GP-ICP is shown in Table 4.3. Some threshold values in GP-ICPR are increased or decreased in a linear manner by an user-defined amount:

$$T^{iter=i+1}_{cc,\ normal,\ or\ distance} = T^{iter=i}_{cc,\ normal,\ or\ distance} \pm \Delta T_{cc,\ normal,\ or\ distance}. \tag{4.39}$$

$T^{iter=0}_{cc}$, ΔT_{cc}, and $T^{iter=0}_{normal}$ are the important and critical thresholds for GP-ICPR and GP-ICP. $T^{iter=0}_{normal}$ represents an approximate relative rotation between the point clouds and is a critical value to the convergence rate of GP-ICPR and GP-ICP. The other threshold values are not as critical as them for the success rate of GP-ICPR and GP-ICP, although they affect the robustness of the registration. The suggestions of $T^{iter=0}_{cc}$ and ΔT from experience with GP-ICPR and GP-ICP are

$$T^{iter=0}_{cc} = \frac{\langle M^1_{cc}\rangle + \langle M^2_{cc}\rangle}{2}$$

and

$$\Delta T_{cc} = \left[\langle M^1_{cc}\rangle^2_{std} + \langle M^2_{cc}\rangle^2_{std}\right]^{\frac{1}{2}}$$

where $\langle M^i_{cc}\rangle$ and $\langle M^i_{cc}\rangle_{std}$ are the mean and standard deviation of the change of curvature of a point cloud, respectively.

Algorithm 2 Geometric Primitive ICP with the RANSAC (GP-ICPR) : An algorithm for the registration of two point clouds

Require: Remove the tree and some erroneous points using Algorithm 5 in Appendix C and resample the point cloud by the method explained in Section 4.7, if it is necessary. Find an optimal size of neighbourhood k for each point cloud using the method presented in Section 4.7.1

Find k neighbour points of every point in C^1 and C^2.
Calculate the eigenvalues and eigenvectors of the covariance matrix of every point in the point clouds.
Estimate the geometric primitives such as surface normal vectors, changes of curvature, and variance angles of the estimated normal vectors for every point in the point clouds.
$passtestornot$ = FALSE
repeat
  if $numberofsample \geq T_{sample}$ **then**
    if $\epsilon > T_{\epsilon_{CM}}$ **then**
      Estimate transformation parameters using Umeyama's (1991) method.
    else
      if $\epsilon \geq T_{\text{RANSAC}}$ **then**
        Estimate transformation parameters using Chen and Medioni's (1992) method.
      else
        Estimate transformation parameters using the modified Chen and Medioni's (1992) method with the RANSAC procedure, i.e. Algorithm 1.
        if The sample satisfies both statistical inferences proposed in Eqs. 4.36 and 4.37 **then**
          $passtestornot$ = TRUE
        else
          $passtestornot$ = FALSE
        end if
      end if
    end if
    Apply the estimated transformation parameters to C^2 and update the surface normal vectors of C^2.
    Evaluate ϵ and other error metrics.
    Update the threshold values in order to apply a more strict criterion for determination of possible corresponding points as shown in Eq. 4.39.
    Select a sample with the three updated thresholds shown in Table 4.3.
    $iteration = iteration + 1$
  else
    Update the threshold values in order to apply a less strict criterion for determination of possible corresponding points as shown in Eq. 4.39.
    Select a sample with the three updated thresholds shown in Table 4.3.
    $iteration = iteration + 1$
  end if
until $\epsilon \geq T_\epsilon$ OR !$passtestornot$

Algorithm 3 Geometric Primitive ICP (GP-ICP) : A simplified version of Algorithm 2

Require: Remove the tree and some erroneous points using Algorithm 5 in Appendix C and resample the point cloud by the method explained in Appendix 4.7, if it is necessary. Find an optimal size of neighbourhood k for each point cloud using the method presented in Section 4.7.1

1: Find k neighbour points of every point in C^1 and C^2.
2: Calculate the eigenvalues and eigenvectors of the covariance matrix of every point in the point clouds.
3: Estimate the geometric primitives such as surface normal vectors, changes of curvature, and variance angles of the estimated normal vectors for every point in the point clouds.
4: **repeat**
5: **if** $number of sample \geq T_{sample}$ **then**
6: **if** $\epsilon > T_{\epsilon_{CM}}$ **then**
7: Estimate transformation parameters using Umeyama's (1991) method.
8: **else**
9: Estimate transformation parameters using Chen and Medioni's (1992) method.
10: **end if**
11: Apply the estimated transformation parameters to C^2 and update the surface normal vectors of C^2.
12: Evaluate ϵ and other error metrics.
13: Update the threshold values in order to apply a more strict criterion for determination of possible corresponding points as shown in Eq. 4.39.
14: Select a sample with the three updated thresholds shown in Table 4.3.
15: $iteration = iteration + 1$
16: **else**
17: Update the threshold values in order to apply a less strict criterion for determination of possible corresponding points as shown in Eq. 4.39.
18: Select a sample with the three updated thresholds shown in Table 4.3.
19: $iteration = iteration + 1$
20: **end if**
21: **until** $\epsilon \geq T_\epsilon$ OR $!passtestornot$

threshold	**description**
k	number of neighbourhood points
T_{sample}	size of a sample
$T_{cc}^{iter=0}$	initial threshold for the difference in the changes of curvature
$T_{normal}^{iter=0}$	initial threshold for the angle between normal vectors
$T_{distance}^{iter=0}$	initial threshold for the distance between possible corresponding points
ΔT_{cc}	increment for $T_{cc}^{iter=k}$
ΔT_{normal}	increment for $T_{normal}^{iter=k}$
$\Delta T_{distance}$	increment for $T_{distance}^{iter=k}$
$T_{\epsilon_{CM}}$	threshold for starting Chen and Medioni's method
T_{ϵ}	threshold for stopping the registration
T_{RANSAC}	threshold for starting the RANSAC procedure

TABLE 4.3: Threshold values are used in either GP-ICPR or GP-ICP.

4.9 Summary

For the improvement of the existing methods such as the ICP and its variants, registration methods, GP-ICP and GP-ICPR, with several key components discussed below were developed.

First, an optimal weight for a point-to-surface based registration method using the position uncertainty of measurement by laser scanners discussed in Chapter 3 was proposed.

Second, both a point-to-point based algorithm and a point-to-surface based algorithm for the estimation of the transformation parameters between two point clouds were proposed to be used in GP-ICPR and GP-ICP. Umeyama (1991) and Chen and Medioni's (1992) methods were used as the point-to-point and point-to-surface based algorithms, respectively, for the registration of the point clouds.

Third, for the selection of points and their corresponding surfaces, the local geometric properties such as the estimated surface normal vector, the change of curvature, the variance angle of the estimated surface normal vector defined as Eq. 4.38, or the variance of position to the direction of the estimated normal vector were utilised in GP-ICPR and GP-ICP. In addition, it was proposed to select points with high change of curvature for an initial sample in the early stages of iteration. We will gradually, in a linear manner, select more points as an initial sample and finally in the last iterations, all points are considered as possible candidates for corresponding points.

Fourth, the lower bound of the registration error and the Cramer-Rao lower bound of measurement by laser scanners were evaluated. Furthermore, two statistical inferences were established using the lower bound of the registration error and they provided some assurance of the unbiasness of the estimated transformation parameters. In addition, a confidence level of the resultant registration errors was provided from the position uncertainty of measurement.

Fifth, it was proposed to use the RANSAC procedure in the last stages of iterations. Since we can only use the estimated geometric properties in order to select corresponding point pairs or point and surface pairs, it is always possible to select the incorrect pairs. Therefore, an outlier removal procedure such as the RANSAC was developed and its proposed implementation was described in Algorithm 1.

Chapter 5

Experiment I : Simulated data

In this chapter, the Geometric Primitive ICP with the RANSAC (GP-ICPR) for the registration of point clouds from laser scanners will be tested in terms of its precision, accuracy, convergence region, and effect of initial threshold values with simulated data. Note that the Geometric Primitive ICP (GP-ICP) is a simplified version of GP-ICPR, both of which were introduced in Chapter 4. Brief summaries of GP-ICPR and GP-ICP are presented in Table 5.1. In addition, regarding the proposed statistical inferences, i.e. Eqs. 4.36 and 4.37, the 95% confidence level will be used throughout this and the following two chapters.

Before presenting the results of the experiments, both the importance of the properties for the evaluation of a registration algorithm and some difficulties which can be encountered in their application will be briefly discussed. First, a smaller precision of an estimation procedure does not mean that either a true solution or a solution which is close to the truth is obtained (Chen et al., 1999). In other words, an independent evaluation of the accuracy of the parametric estimation is required without a priori information on the truth. However, to obtain the true relative transformation parameters of two point clouds from a laser scanner is almost impossible. One can argue that the true relative transformation parameters can be obtained using targets coordinated by an instrument for measuring the position of a point, e.g. a total station, which is regarded as the standard method in surveying and geomatics. Let this be called the direct georeferencing method for the registration of point clouds. Since a set of laser scanner data and the target observation by the direct georeferencing method are two different datasets, the extracted coordinates from the targets may not have corresponding points in the point cloud from the laser scanner. Furthermore, even assuming that all the coordinates of the targets have a set of corresponding points, it does not mean that the variances of these points are the same since the two instruments have different variances. For example, the variance of measurement by a total station is usually much smaller than that of a terrestrial laser scanner. Therefore, the estimated transforma-

GP-ICP ≡ The Geometric Primitive Iterative Closest Point (ICP) **≡ Algorithm 3 in Chapter 4**

GP-ICPR ≡ The Geometric Primitive ICP with the RANSAC **≡ Algorithm 2 in Chapter 4** **≡ GP-ICP + the position uncertainty of a laser scanner** **+ the RANSAC procedure using the position uncertainty,** **proposed in Algorithm 1 in Chapter 4**

TABLE 5.1: Automated registration methods developed in this dissertation.

tion parameters using the direct georeferencing method may or may not provide an estimation which is closer to the truth for laser scanner data. That is why the position uncertainty or variance of the measurement by a laser scanner needs to be explicitly expressed, as derived in Chapter 3, although it is difficult to mathematically prove whether or not an explicit expression of the position uncertainty is absolutely correct in every situation.

Second, a wider convergence region in terms of rotation and translation, especially a wider rotational convergence region, is required for the fully automated registration of point clouds from range finders (Eggert et al., 1997; Chen et al., 1999; Rusinkiewicz and Levoy, 2001; Huber and Hebert, 2003; Fitzgibbon, 2003). One of the important facts and difficulties is that the rotational convergence region depends on both the initial registration error of two point clouds and an initial distance threshold. In addition, if threshold values such as an initial distance threshold value of GP-ICP or GP-ICPR is utilised within a registration algorithm, the convergence region is also dependent on them.

5.1 Overview of the experiments

The tests that will be conducted in this and the following two chapters are briefly summarised as follows. GP-ICPR will be first tested with a set of the simulated data of which the true relative transformation is known. Both the precision and accuracy of the estimation by GP-ICPR will also be properly evaluated and GP-ICPR will be

shown to provide more precise and accurate results than other registration methods such as the direct georeferencing method and a commercial software, Cyclone (Leica, 2006, version 5.0). Note that, in Chapter 5, the direct georeferencing method will be referred as 'a method with a priori correspondence' since simulated datasets are tested.

In Chapter 6, a set of close-range scanner data will be tested with GP-ICP, though information on the accuracy on the estimation will not be provided since the positional and angular uncertainties of the laser scanner, i.e. σ_r and σ_a, are unknown. However, a comparison between GP-ICP and a registration method based on the original ICP (Turk and Levoy, 1994) will be provided in terms of registration error and absolute difference between the estimated transformation parameters. In Chapter 7, a set of terrestrial laser scanner data will be tested and GP-ICPR will be shown to be superior to the other registration methods such as the direct georeferencing method. Additionally, an evaluation of the convergence region of GP-ICPR will be obtained using all three sets of the data, i.e. the simulated, close-range scanner data, and a set of data from terrestrial laser scanners.

5.2 Registration error analysis and accuracy tests

In this section, the precision of the relative transformation parameters by GP-ICPR will be evaluated with a set of simulated point clouds. As mentioned earlier, the accuracy of the estimated parameters can be properly evaluated only with simulated data since the true relative transformation parameters are not available in the other real cases. In addition, the convergence region of GP-ICPR with simulated point clouds in various situations, e.g. with different kinds of spatial noise or different relative transformations, will be evaluated. Furthermore, the effect of the initial threshold values in GP-ICPR will be briefly presented.

It must be noted that simulated datasets are unitless. However, in order to help understanding the magnitude of registration or translational errors in a registration algorithm, the pixel of a point cloud is defined as the average distance from a point to its neighbours in the point cloud as follows:

$$\text{pixel} = \frac{1}{n}\sum_{i=1}^{n}\langle D^{pp}(p_i)\rangle \tag{5.1}$$

$$\langle D^{pp}(p_i)\rangle = \frac{1}{k}\sum_{j=1}^{k} D_j^{pp}(p_i) \tag{5.2}$$

where n is the number of points, k is the number of neighbours, $D_j^{pp}(p_i)$ is the distance between p_i and its jth neighbour, and $\langle D^{pp}(p_i)\rangle$ is the average distance between a point and its neighbourhood. Note that the dimension of the pixel of a point cloud is

equivalent to that of distance. In addition, the pixel of a three-dimensional point cloud is a relative unit since the size of a pixel is dependent on the spatial characteristics of a point cloud such as point density. However, two partially overlapping point clouds simulated from computer-aided design (CAD) models have almost the same size of a pixel since they have similar spatial point densities.

Three simulated point clouds are presented for GP-ICPR precision and accuracy tests: two datasets have been generated from the CAD models, 'cactus' and 'golf club' * and one dataset has been mathematically generated, a cube. The points from the CAD models are taken from the three corners of the triangles constituting the CAD models and the cube consists of a set of random points on the surface of a cube.

These datasets can be regarded as noise-free datasets. Two types of noise, such as zero-mean Gaussian and a noise derived from the results of Chapter 3, were added to a dataset in order to ensure that every point has a different local sampling resolution. Therefore, a simulated dataset with noise does not have perfect correspondence with the other simulated dataset. The simulated point clouds for each dataset have an identical global sampling resolution and no systematic difference in spatial sampling resolution. This situation is different from real laser scanner point clouds.

Two point clouds which share a certain amount of overlapping regions with each other were manually cut from the complete point clouds, e.g. the cactus, the golf club, and the cube. The total number of point clouds and the number of points in the overlapping regions for each point cloud can be found in Table 5.2. Note that the methods of finding corresponding points in either GP-ICPR or GP-ICP are the same.

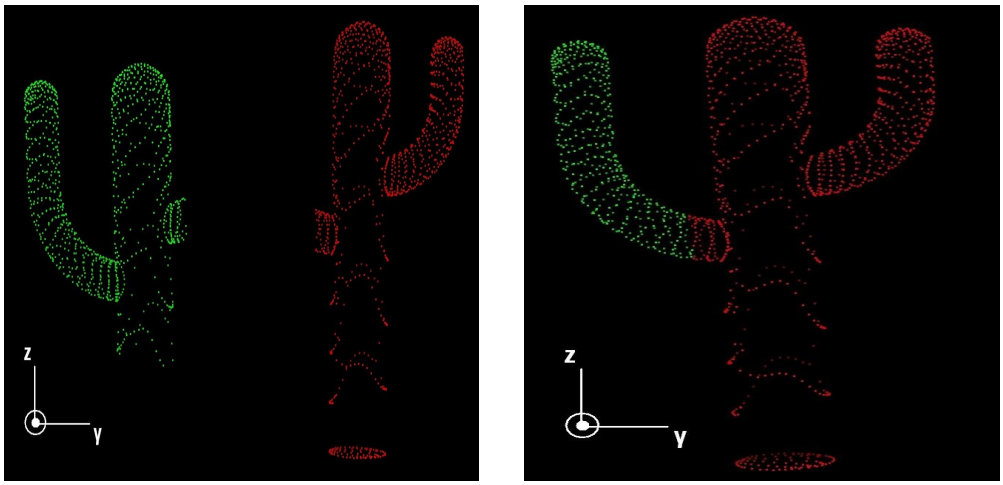
	point cloud 1	point cloud 2
cactus	179/519 = 34%	179/545 = 33%
golf club	622/1371 = 45%	622/1861 = 33%
cube	278/2640 = 11%	278/4048 = 7%

TABLE 5.2: Sizes of the overlapping regions are represented as percentages of the whole point cloud, i.e. the denominator is the total number of points in the point cloud and the numerator is the number of points in the overlapping region.

5.2.1 Registration error analysis with noise-less simulated data

For the accuracy test in absence of noise, two point clouds of an object were perfectly aligned and then one of them was relatively rotated by $\pm 10^\circ$ in a 1° increments around the y axis. The average estimated transformation errors of GP-ICP are presented in Tables 5.3 and 5.4. The average estimated rotation and translation errors are on the

*These data were used in Hoppe et al. (1992) and are available through Hugues Hoppe's website: ftp://ftp.research.microsoft.com/users/hhoppe/data/thesis/phase2_meshes/, which was accessed by the author on 16th April 2006. The file names of the cactus and the golf club are cactus.crep1e-5.m.gz and club71.crep1e-5.m.gz, respectively.



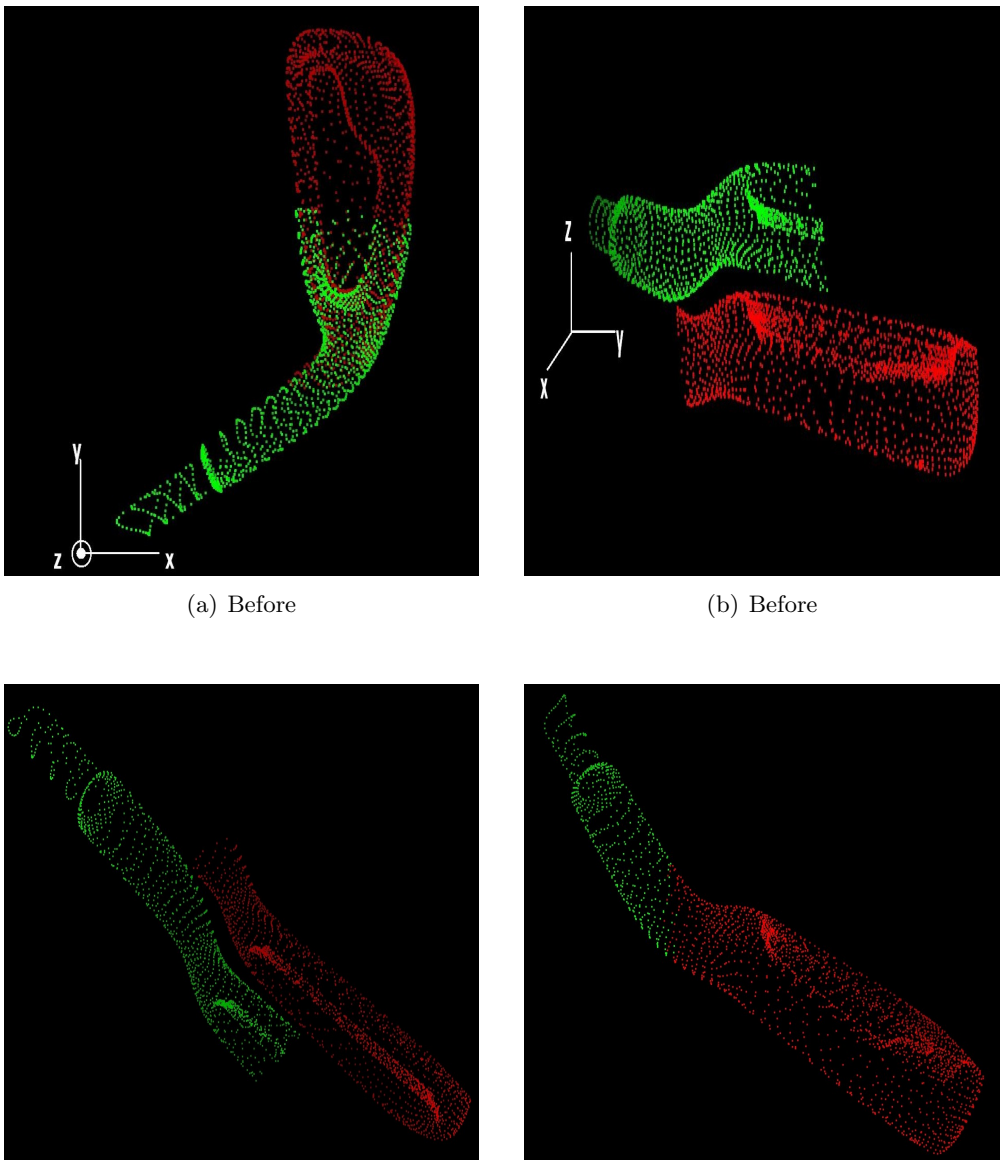
(a) (b)

FIGURE 5.1: Two point clouds from a CAD model named the cactus. The dimension of the cactus is $(H, L, W) \simeq (0.80, 0.50, 0.13)$. The pixel size defined as the average distance from a point to its neighbours over an entire point cloud is 0.01.

order of 10^{-5° and 10^{-5} pixel, respectively. It means that GP-ICP provides a good set of possible corresponding points in the case of the simulated dataset without noise. In addition, a set of good corresponding points will be selected by GP-ICPR since the method for finding corresponding points of GP-ICPR is the same as that of GP-ICP. However, since these data are free from noise and a point in the overlapping region has a perfect corresponding point in the other point cloud, further tests are required in the presence of realistic noise.

As observed in Figures 5.1 and 5.3, the rotation axes of the cactus and the cube, i.e. the y axis, is approximately perpendicular to the longitudinal or medial axis of either the cactus or the cube and so it may increase the chance of convergence to a solution. However, in the case of the golf club shown in Figure 5.2, the rotation axis of the tests is not close to the medial axis of the object. Note that these tests are only for an evaluation of the accuracy of the proposed algorithm. In the case of an evaluation of the convergence region of GP-ICPR, which will be shown in Section 5.3, a translation will be first applied and a rotation will be applied later in order to prevent applying a rotation only around an axis which is close to the medial axis of an object.

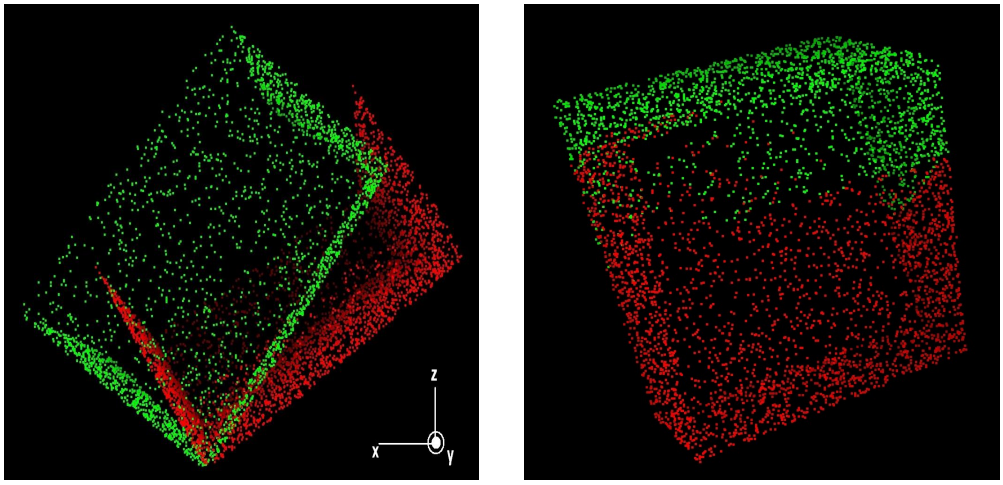
Now consider the registration errors of the tests presented in Table 5.5 in the case of the registration of the original simulated data, i.e. noise-free data. Note that D^{pp}_{mean} is the mean distance between the corresponding points in a point cloud and D^{ps}_{mean} and D^{ps}_{std} are the mean and standard deviation of the distances between points and their corresponding surfaces in a point cloud, respectively. Since these data have a set of perfect corresponding points, D^{pp}_{mean} is small: on the order of 10^{-2} pixel. D^{ps}_{mean}



(a) Before (b) Before

(c) Before (d) After

FIGURE 5.2: Two point clouds from a CAD model 'golf club'. The dimension of the golf club is $(H, L, W) \simeq (0.80, 0.62, 0.13)$. The pixel size defined as the average distance from a point to its neighbours over an entire point cloud is 0.008.



(a) Before, top view (b) After

FIGURE 5.3: Two point clouds from the cube are presented. The dimension of the cube is $(H, L, W) \simeq (2.0, 2.0, 0.45)$. The pixel size defined as the average distance from a point to its neighbours over an entire point cloud is 0.02.

average rotation error [degree]	cactus	golf club	cube
$d\omega$	-1.0×10^{-5}	-0.3×10^{-5}	0.2×10^{-5}
$d\phi$	0.2×10^{-5}	-0.4×10^{-5}	0.2×10^{-6}
$d\kappa$	0.6×10^{-4}	0.4×10^{-5}	0.4×10^{-4}

rotation error [degree]	cactus [1.0×10^{-4}]		golf club [1.0×10^{-4}]		cube [1.0×10^{-5}]	
	max	min	max	min	max	min
$d\omega$	0.5	-0.4	0.3	-1.1	0.8	-0.2
$d\phi$	0.2	-0.4	0.7	-0.8	0.1	-0.1
$d\kappa$	3.3	-1.5	1.4	-0.9	3.8	3.0

TABLE 5.3: Average, maximum, and minimum rotation errors for the accuracy tests of GP-ICP using the original simulated data. One of the point clouds from each set of data is relatively rotated from the original position, i.e. perfectly registered, by $\pm 10°$ with a $1°$ increment. No translation was applied and the same threshold values were used for all three data.

average translation error [pixel]	cactus	golf club	cube
dx	5.2×10^{-5}	6.0×10^{-6}	0.0
dy	-6.0×10^{-5}	-1.2×10^{-5}	0.0
dz	1.4×10^{-5}	6.0×10^{-6}	0.0

translation error [pixel]	cactus [1.0×10^{-4}]		golf club [1.0×10^{-6}]		cube [1.0×10^{-6}]	
	max	min	max	min	max	min
dx	3.0	-1.0	1.0	-1.0	1.0	1.0
dy	1.0	-2.0	1.0	-2.0	1.0	-1.0
dz	1.0	0.0	1.0	-1.0	0.0	0.0

TABLE 5.4: Average, maximum, and minimum translation errors for the accuracy test of GP-ICP using the original simulated data, i.e. noise-free data. One of the point clouds from each set of data is relatively rotated from the original position by $\pm 10^\circ$ with a 1° increment. No translation was applied and the same threshold values were used for all three datasets.

and D^{ps}_{std} are relatively large, although neither of them is greater than 0.5 pixel. As mentioned earlier, D^{pp}_{mean} is not always a good parameter for the evaluation of the quality of the registration of point clouds since we usually do not have the corresponding point of every point in a overlapping region with the presence of either noise or different sampling intervals. Either D^{ps}_{mean} or D^{ps}_{std} is a good substitute for D^{pp}_{mean} since there is a better chance to have a correct set of corresponding point and surface pairs (Simon, 1996, Ch. 2). One disadvantage of D^{ps} is that it depends on the geometric shape of objects we have scanned. Although D^{pp}_{mean} is zero, either D^{ps}_{mean} or D^{ps}_{std} can be non-zero as shown in the second column of Table 5.5.

registration error [pixel]	cactus	golf club	cube
D^{pp}_{mean}	0.00	0.45×10^{-2}	0.23×10^{-1}
D^{ps}_{mean}	0.58	0.45	0.28
D^{ps}_{std}	0.40	0.40	0.55

TABLE 5.5: Average registration errors for the accuracy test of GP-ICP using the simulated datasets. One of the point clouds of each set of data is relatively rotated from the original position by $\pm 10^\circ$ with a 1° increment. No translation was applied and the same threshold values were used for all three data.

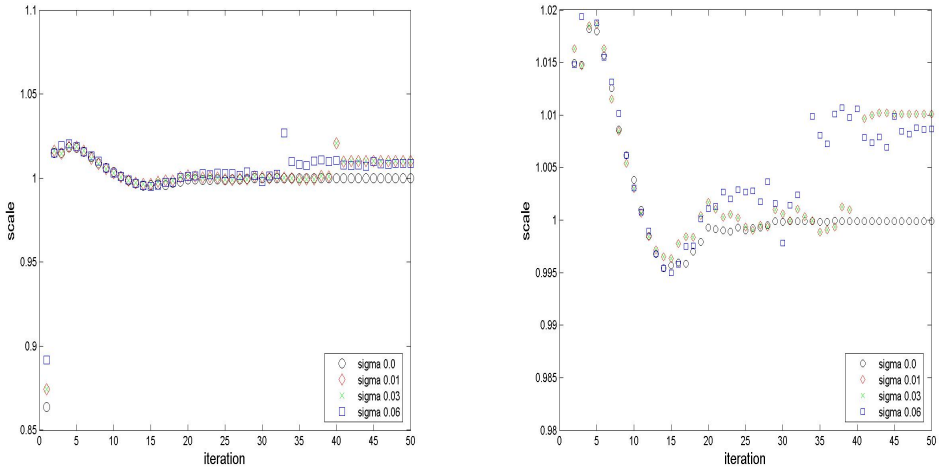
5.2.2 Scale estimation with zero-mean Gaussian noise

The scale factor between two point clouds will be briefly discussed, although, as mentioned in Section 1.2, this thesis does not intend to determine the scale factor of laser scanner datasets to a good degree of accuracy. The scale of selected corresponding points is usually assumed to be unity and this assumption is reasonable in most

cases (Horn, 1987; Gruen and Akca, 2005). However, the estimated scale can be also used as an error metric to represent the quality of registration (Crosilla and Beinat, 2002). For example, if we have incorrect correspondence information, then the scale is not close to unity. The scale factor in the kth iteration, $s_{iter=k}$, can be expressed as

$$s_{iter=k} = \frac{\sum_{i=1}^{n_{iter=k}} \left(\mathbf{p}_i^1, T_{iter=k}(\boldsymbol{CP}(p_i^1, C^2))\right)}{\sum_{i=1}^{n_{iter=k}} ||T_{iter=k}(\boldsymbol{CP}(p_i^1, C^2))||^2} \tag{5.3}$$

where p_i^1 is the ith point in the point cloud 1, $T_{iter=k}$ is the estimated transformation of the kth iteration, $\boldsymbol{CP}(p_i^1, C^2)$ is the position vector of the corresponding point of p_i^1, and $n_{iter=k}$ is the number of corresponding points in the kth iteration. Although it is difficult to determine the scale between two point clouds with good precision and accuracy, even a roughly estimated scale, e.g. Eq. 5.3, provides an insight of the quality of the selected corresponding points in the kth iteration. In other words, if the scale is close enough to unity, then it assures that the selected corresponding points in that iteration are good and so an outlier removal procedure, e.g. the proposed RANSAC procedure within GP-ICPR presented in Algorithm 1, can be started.



(a) (b)

FIGURE 5.4: Scale of selected corresponding points in each iteration of the registration of the cube with zero-mean Gaussian noise. "sigma" in (a) represents the standard deviation of zero-mean Gaussian error. (b) is a magnified figure of (a).

The estimated scale of selected corresponding points at each iteration of the registration of the cube by GP-ICP with various standard deviations of zero-mean Gaussian noise is shown in Figure 5.4. Note that all selected corresponding points used for the estimation of the relative transformation were again utilised for the estimation of the scale using Eq. 5.3. The number of the selected corresponding points in an iteration

σ [pixel]	dω [degree]	dϕ [degree]	dκ [degree]	dx [pixel]	dy [pixel]	dz [pixel]
0.01	0.00	0.00	-0.02	-0.02	0.03	0.01
0.03	-0.09	0.00	-0.06	-0.05	0.00	0.11
0.05	0.04	0.04	0.33	0.22	-0.24	-0.28
0.40	-1.35	0.58	-1.54	-1.06	1.61	0.68

TABLE 5.6: Difference between the true and the estimated transformation parameters with zero-mean Gaussian noise using a set of a priori correspondence.

σ [pixel]	dω [degree]	dϕ [degree]	dκ [degree]	dx [pixel]	dy [pixel]	dz [pixel]
0.01	0.00	0.00	-0.02	0.01	-0.01	0.00
0.03	-0.09	0.03	-0.06	0.08	-0.16	0.06
0.05	0.04	0.17	0.33	0.22	-0.14	-0.13
0.40	-1.35	0.59	-1.54	-6.05	6.23	-0.17

TABLE 5.7: Difference between the true and the estimated transformation parameters with zero-mean Gaussian noise using GP-ICP.

is different, since the number of the selected corresponding points is dependent on the threshold values of GP-ICP. In early stages of registration, scales are greater than unity since a good a priori alignment is not available and a selected set of corresponding points is not correct. After about fifteen iterations in this test, all estimated scales of the data with different levels of noise are close to unity, which is a good indication of success in finding correspondence.

5.2.3 Registration error analysis with zero-mean Gaussian noise

The estimated transformation parameters of two point clouds from the cactus, to which zero-mean Gaussian noise with different variances has been added, are presented in Tables 5.6 and 5.7. A priori information on the corresponding points is used to estimate the transformation parameters given in Table 5.6. In the case of Table 5.7, GP-ICP is utilised to estimate the relative transformation between two point clouds with zero-mean Gaussian noise with different variances. Note that a priori information on the correspondence of the point clouds is never known in real cases. Therefore, the sole purpose of the tests with a priori correspondence is to observe the effect of noise with the best possible selection of corresponding points.

As observed in Table 5.6, as the variance of zero-mean Gaussian noise increases, the accuracy of the estimated transformation decreases, even with a priori correspondence of two point clouds. When the square root of the variance of zero-mean Gaussian noise is smaller than 0.05, the bias in translation is on the order of 0.01 pixel and the rotational parameters' bias is on the order of 0.01°. However, in the case of $\sigma = 0.40$,

about on the order of a degree and a pixel errors are observed in terms of rotation and translation, respectively. This is because a priori corresponding points were collected before adding the noise. A priori correspondence is not correct any more once the noise is added. In the case of $\sigma = 0.01$, it is observed that the bias in the result of GP-ICP is better than with a priori correspondence. In the cases of $\sigma = 0.03$ and $\sigma = 0.05$, both a priori correspondence method and GP-ICP provide a similar bias in the estimation of the relative transformation. It is clearly observed that GP-ICP in the case of $\sigma = 0.40$ provides a much larger bias than with a priori information on correspondence from Table 5.7 simply because it is more difficult to find a set of correct corresponding points in the presence of noise with a large variance.

If the point clouds from either a laser scanner or a range sensor contain zero-mean Gaussian noise with a certain variance, then the variance of the noise can be estimated using covariance analysis of the entire point cloud (Dorai et al., 1997; Kanatani, 1996). Then, using this estimated variance, the bias estimation in the registration of the point clouds can be removed or reduced. In addition, a statistical inference for zero-mean Gaussian noise with the estimated variance can be constructed to verify the quality and unbiasness of the estimated parameters as done in Chapter 3. However, as mentioned earlier, the noise of a point cloud from a laser scanner does not follow zero-mean Gaussian knowing from an explicit expression of the position uncertainty of laser scanners, i.e. Eqs. 3.28 and 3.29. Therefore, GP-ICPR needs to be tested in the presence of a realistic noise for laser scanners.

5.2.4 A realistic noise model for laser scanners

What kind of spatial noise is reasonable to describe the realistic noise in the case of point clouds from laser scanners? Using the results of Chapter 3, a position vector, $\mathbf{r}$, with the realistic laser scanner measurement uncertainties is expressed as

$$\mathbf{r} = \bar{\mathbf{r}} + \sum_{j=1}^{3} \sqrt{V(r)^{jj}_{point}}\hat{\mathbf{e}}_{j-1} \tag{5.4}$$

where $\bar{\mathbf{r}}$ is the true position of a point and $V(r)^{jj}_{point}$ is the variance to the direction of the jth smallest eigenvector of the covariance matrix, $\hat{\mathbf{e}}_{j-1}$, as derived in Eqs. 3.28 and 3.29. If $V(r)^{jj}_{point}$ is a random variable which follows the Gaussian probability density function, then the spatial noise of the scanner is a zero-mean Gaussian. However, as learned from Chapter 3, $V(r)^{jj}_{point}$ is not a random variable. It mainly depends on either the incident angle of laser beam to the surface of objects or the local distribution of a point around its neighbour as derived in Chapter 3.

5.2.5 Registration error analysis with the realistic noise of laser scanners

The registration accuracy tests of two point clouds will be discussed with realistic noise as described in Eq. 5.4. Two different radial and angular uncertainties, σ_r and σ_a, are used for the tests and the results of the accuracy tests are presented in Tables 5.8 and 5.9. Note that, for a reference, the second columns of Tables 5.8 and 5.9 present the results of the estimated transformation parameter errors with noise-free dataset using GP-ICP.

First, a significant amount of error in the transformation parameters is observed in both a priori correspondence and GP-ICP. The results from a priori correspondence imply that a registration algorithm based on a point-to-point based algorithm does not provide a good estimation of the relative transformation of two point clouds since a better set of corresponding points than a set of a priori corresponding points can not be provided. In addition, the results from GP-ICP suggest that, in the presence of a large amount of noise, finding a set of corresponding points with the geometric primitives is not good enough since these primitives are the estimates of the true geometric primitives as well. Furthermore, they also show us that an outlier removal procedure, e.g. the proposed RANSAC procedure in Algorithm 1, is required to estimate the relative transformation parameters of two point clouds as accurately as possible. In Tables 5.8 and 5.9, it is clearly observed that the estimation and outlier removal procedures in GP-ICPR is superior than the others: a method with a priori correspondence and GP-ICP.

rotation error [degree]	noiseless using GP-ICP	with a priori correspondence	GP-ICP	GP-ICPR
$d\omega$	-5.20×10^{-5}	-0.85	-2.15	-0.28
$d\phi$	1.20×10^{-5}	-0.37	-0.36	-0.18
$d\kappa$	-1.10×10^{-4}	-0.34	-2.5	-0.01

translation error [pixel]	noiseless using GP-ICP	with a priori correspondence	GP-ICP	GP-ICPR
dx	-1.00×10^{-4}	-0.25	-1.60	-0.03
dy	1.00×10^{-4}	0.08	1.10	-0.13
dz	0.00	-0.53	0.11	0.40

TABLE 5.8: Test I: Accuracy test of the registration of two point clouds (the cactus) with realistic noise with $\sigma_r = 0.4$ [pixel] and $\sigma_a = 3.4 \times 10^{-3}$ [degrees] using three methods: a method with a priori correspondence, GP-ICP, and GP-ICPR. The distance of the point cloud from an imaginary location of a laser scanner is approximately 70 pixel.

The registration errors of the above tests in the cases of GP-ICP and GP-ICPR are shown in Table 5.10. Let test I and test II be the tests involved by Table 5.8

rotation error [degree]	noiseless using GP-ICP	with a priori correspondence	GP-ICP	GP-ICPR
$d\omega$	-5.20×10^{-5}	-0.16	0.57	0.14
$d\phi$	1.20×10^{-5}	-0.11	0.09	0.04
$d\kappa$	-1.10×10^{-4}	0.96	0.80	0.36

translation error [pixel]	noiseless using GP-ICP	with a priori correspondence	GP-ICP	GP-ICPR
dx	-1.00×10^{-4}	0.69	0.37	0.12
dy	1.00×10^{-4}	-0.84	-0.41	-0.33
dz	0.00	-0.53	0.01	0.08

TABLE 5.9: Test II: Accuracy test of the registration of two point clouds (the cactus) with realistic noise with $\sigma_r = 0.5 \times 10^{-1}$ [pixel] and $\sigma_a = 3.4 \times 10^{-3}$ [degrees] using three methods: a method with a priori correspondence, GP-ICP, and GP-ICPR. The distance of the point cloud from an imaginary location of a laser scanner is approximately 70 pixel.

	GP-ICP		GP-ICPR			
registration error	D^{ps}_{mean} [pixel]	D^{ps}_{std} [pixel]	ratio of inliers inliers/total	D^{ps}_{mean} [pixel]	D^{ps}_{std} [pixel]	CRLB [pixel]
test I	0.61	0.69	30/31 = 97%	0.21	0.15	0.15
test II	0.47	0.47	36/44 = 82%	0.08	0.11	0.08

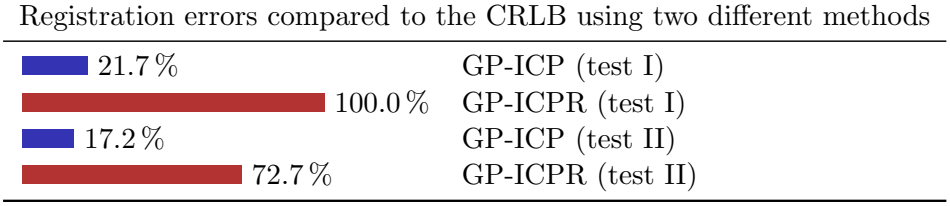


TABLE 5.10: Top table shows the mean and standard deviation of the distance between points and their corresponding surfaces using two different methods, i.e. GP-ICP and GP-ICPR. It also presents the ratio of inliers and the Cramer-Rao lower bound (CRLB) of the registration errors estimated from Eq. 4.25. The bottom bar graph represents how much the values of D^{ps}_{std} in two different proposed methods, are close to the Cramer-Rao lower bound of the registration error of these point clouds. For example, if it is 100%, then the resultant registration error is as small as it can be, i.e. the CRLB.

and Table 5.9, respectively. The Cramer-Rao lower bound (CRLB) of D_{std}^{ps} represents the minimum possible standard deviation of the distance between points and their corresponding surfaces as estimated from Eq. 4.25. In the case of GP-ICP, the final registration error is to the order of half a pixel but that of GP-ICPR is to the order of 0.2 pixel, although a smaller registration error does not guarantee the accuracy of the estimated transformation parameters. However, we have the Cramer-Rao lower bound (CRLB) and two statistical inferences based on the CRLB, i.e. Eqs. 4.36 and 4.37, which are a set of good indicators to assure the accuracy of the estimation.

However, we can not still be sure about the accuracy of estimation. Fortunately, the true transformation is known in these cases. Figures 5.5 - 5.8 show the estimated transformation parameters in three different situations: without noise using GP-ICP, with realistic noise using GP-ICP, and with realistic noise using GP-ICPR, versus iteration of the methods. Note that, in the case of data with realistic noise registered by GP-ICPR, GP-ICP was applied up to the 110th iteration and then GP-ICPR was applied to the end, i.e. the 120th iteration.

From Figures 5.7 - 5.8, it is observed that GP-ICPR helps the results of GP-ICP to reach the truth, although the z components of the estimation move slightly away from the true translation. In the tests with the golf club, GP-ICP also yields a set of the estimated transformation parameters that is very close to the truth as shown in Figures 5.7 - 5.8. Therefore, GP-ICPR provides a set of results which is not much different from that of GP-ICP, as expected.

In the case of the cactus, the rotation axis, i.e. the y axis, is approximately perpendicular to its medial axis. Therefore, a faster convergence than the other rotation parameters is observed in Figures 5.5 and 5.6. On the other hand, this tendency is not observed in the case of the golf club as presented in Figures 5.7 - 5.8 since the medial axis of the golf club is different from the rotation axis of these tests, i.e. the y axis.

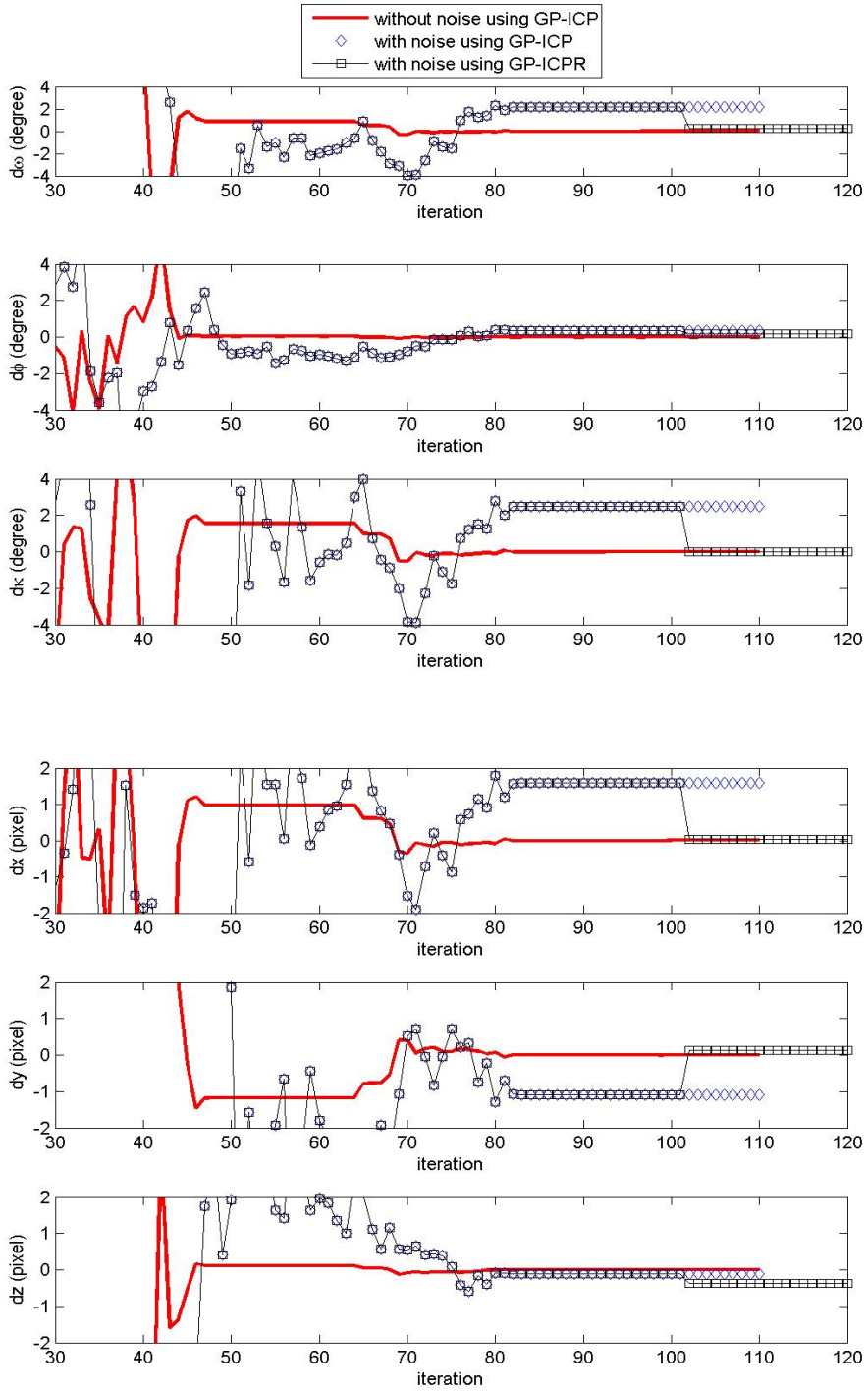


FIGURE 5.5: Accuracy test of the registration of two point clouds from the cactus with a realistic noise ($\sigma_r = 0.4$ [pixel] and $\sigma_a = 0.6 \times 10^{-4}$ [rad]) with three methods: without noise using GP-ICP, with noise using GP-ICP, and with noise using GP-ICPR.

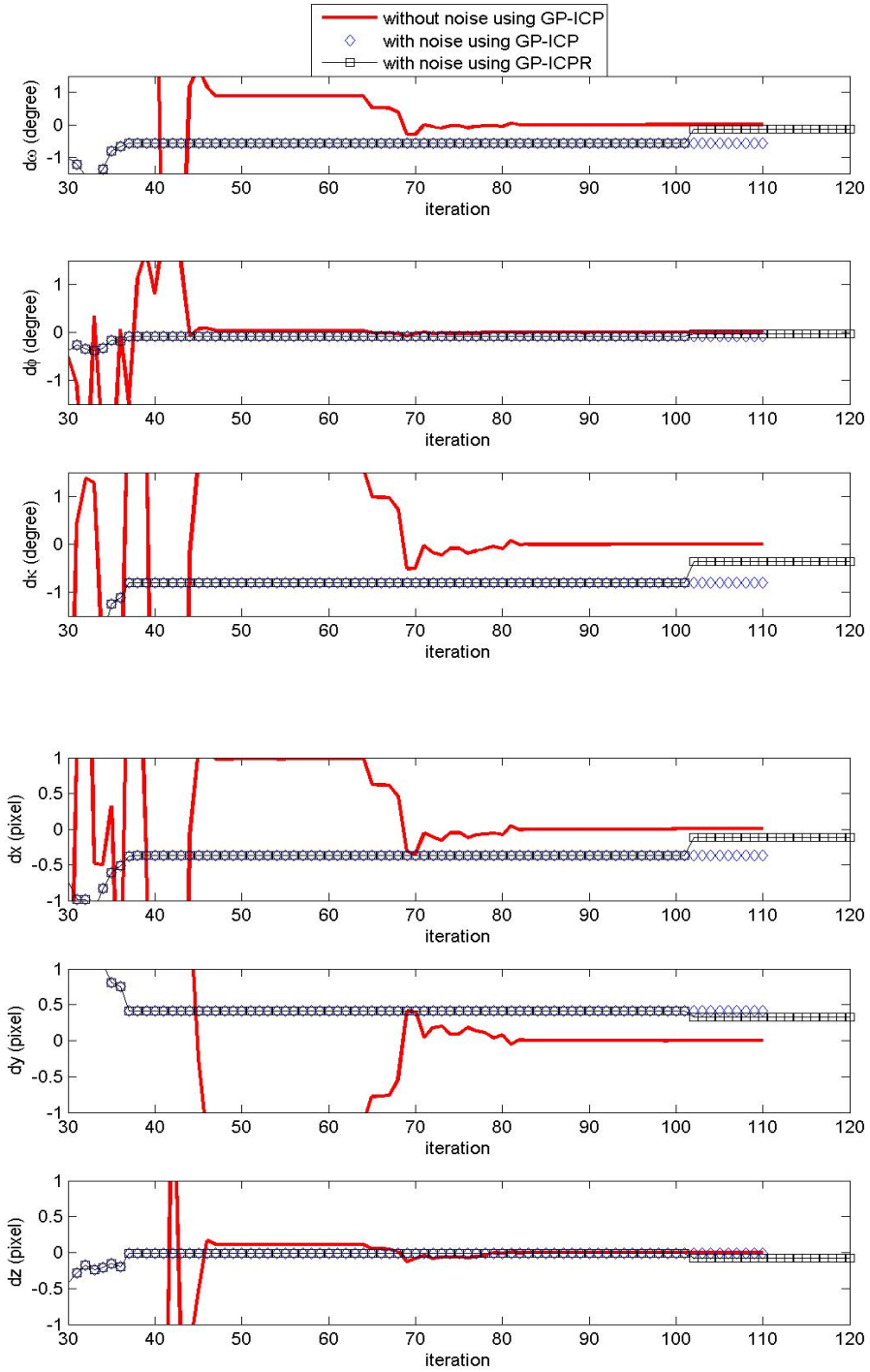


FIGURE 5.6: Accuracy test of the registration of two point clouds from the cactus with a realistic noise ($\sigma_r = 0.5 \times 10^{-1}$ [pixel] and $\sigma_a = 0.6 \times 10^{-4}$ [rad]) with three methods: without noise using GP-ICP, with noise using GP-ICP, and with noise using GP-ICPR.

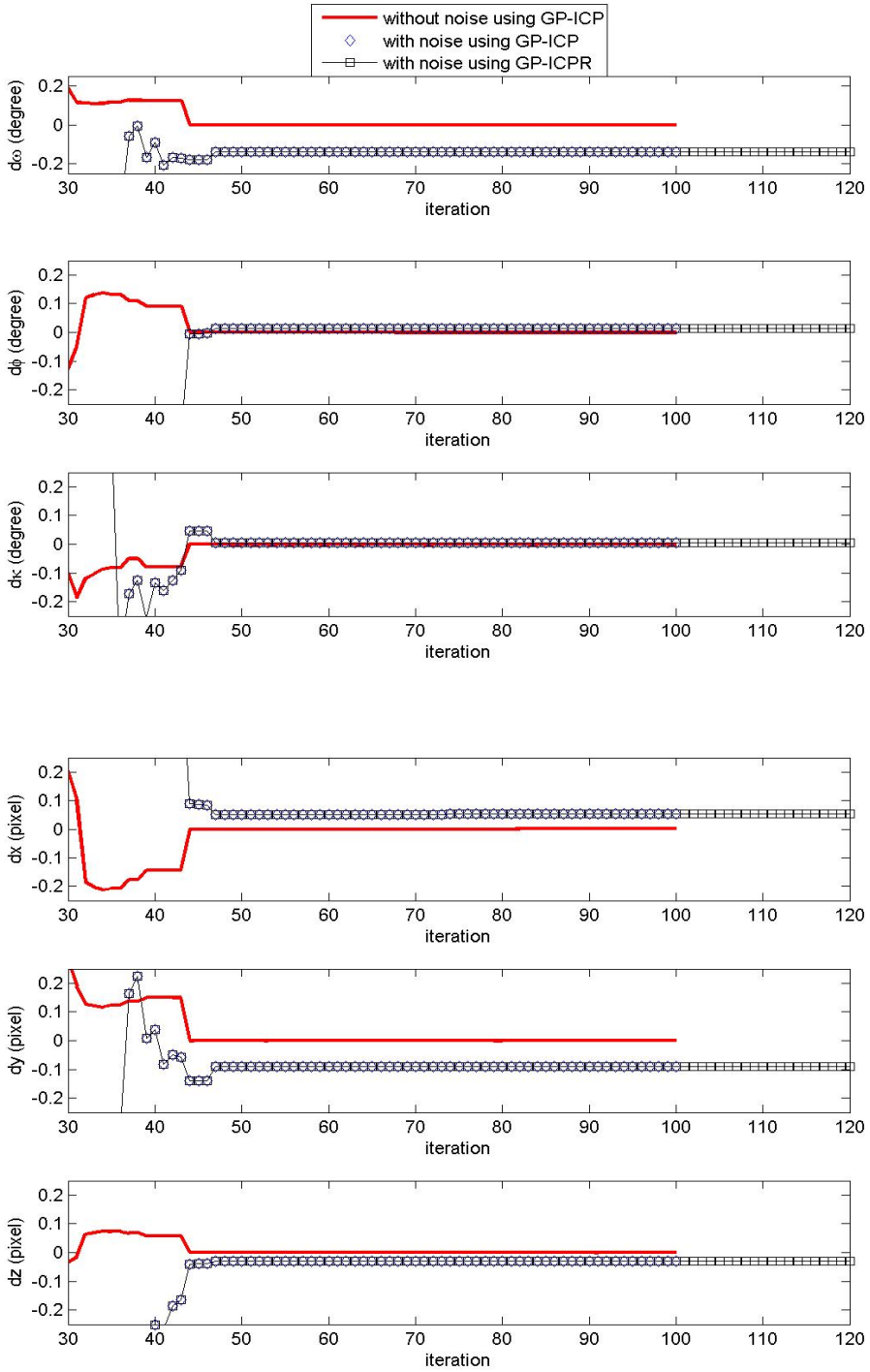


FIGURE 5.7: Accuracy test of the registration of two point clouds from the golf club with a realistic noise ($\sigma_r = 0.4$ [pixel] and $\sigma_a = 0.6 \times 10^{-4}$ [rad]) using three methods: without noise using GP-ICP, with noise using GP-ICP, and with noise using GP-ICPR.

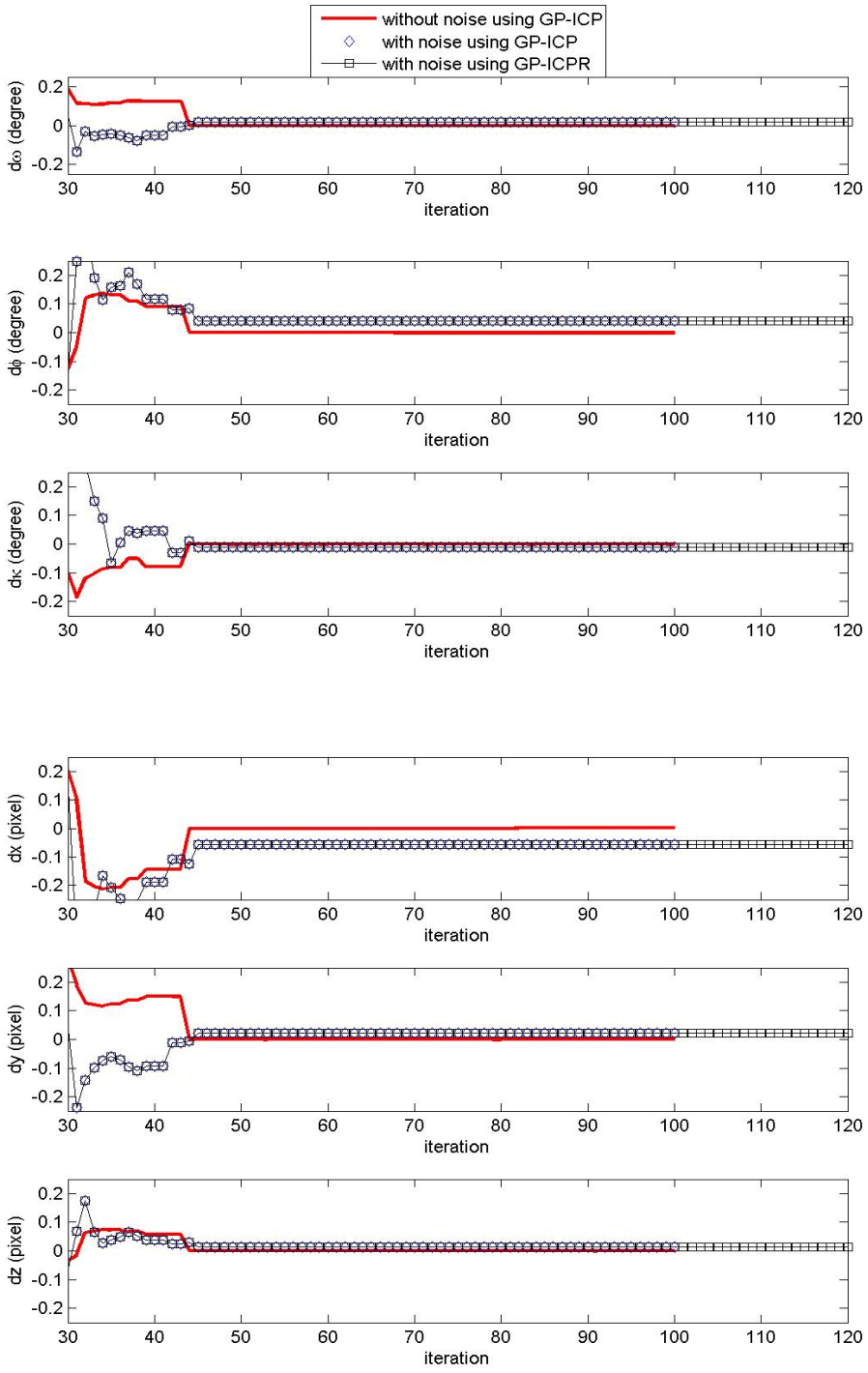


FIGURE 5.8: Accuracy test of the registration of two point clouds from the golf club with a realistic noise ($\sigma_r = 0.5 \times 10^{-1}$ [pixel] and $\sigma_a = 0.6 \times 10^{-4}$ [rad]) using three methods: without noise using GP-ICP, with noise using GP-ICP, and with noise using GP-ICPR.

5.3 Convergence region

In this section, the convergence region of GP-ICPR will be evaluated in different situations and will be compared with that of the original ICP with random sampling proposed by Besl and McKay (1992) and Masuda and Yokoya (1995). As stated earlier in the chapter, it is difficult to determine the convergence of a registration algorithm since it depends on both initial rotation and translation parameters even in the case of ignoring the scale difference between point clouds. In other words, to obtain $\pm 50^\circ$ rotational convergence region around an axis in one situation does not guarantee that we will have the same or a larger rotational convergence region in other situations. Therefore, the scope of the test for the convergence region of GP-ICPR in this dissertation with the simulated data must be stated as follows:

- These tests for the convergence region were conducted with GP-ICP since the convergence region does not heavily depend on whether or not the proposed RANSAC procedure is used, from the fact that the same method for finding correspondence is used in both methods.

- The cactus and the golf club were used for the tests. In all the tests, these simulated point clouds were rotated by a fixed amount, $-50^\circ < R^{initial} < 50^\circ$, where $R^{initial}$ is a rotational angle around an axis. For translations, three kinds of the tests were performed coinciding with the $R^{initial}$: no translation, a translation of (H/4, L/4, W/2) named translation 1, and a translation of (-H/4, L/4, W/2) named translation 2, where H, L, and W are the height, length, and the width of either the cactus or the golf club, respectively.

- $T_{normal}^{iter=0}$ was set to be $|R^{initial}| + 10^\circ$ and it was changed from a maximum value of a test point cloud to zero with increment of 10 pixel. For example, in the test either by GP-ICP or the original ICP with random sampling, a new $T_{distance}^{iter=0}$ is tried unless the solution of a registration algorithm converges, until $T_{distance}^{iter=0}$ reaches to zero. Therefore, if the final $T_{distance}^{iter=0}$ is zero, it means that the registration algorithm did fail to find a solution within a maximum $T_{distance}^{iter=0}$.

The results of the convergence tests of both GP-ICPR and the original ICP with random sampling are presented in Figures 5.9 - 5.14. In all the tests, GP-ICPR provides $\pm 50^\circ$ rotational convergence region, which is the maximum possible rotational convergence angle in these tests. On the other hand, the original ICP with random sampling's rotational convergence region in these tests is at best $\pm 5^\circ$. For example, in the case of Figures 5.13 and 5.14, the original ICP with random sampling does not converge into a solution in any test region, i.e. the rotational convergence region is zero. Although GP-ICPR requires a different $T_{distance}^{iter=0}$ in a different situation, the convergence region of GP-ICPR is reasonably large for practical applications.

The success rate of the original ICP with random samples is poor since it does not escape from local minima in the ways that it finds a corresponding point, i.e. using the nearest neighbour point as the corresponding point. GP-ICPR provides a way of avoiding these kinds of local minima, although it still has limitations. In the case of the direct georeferencing method, the translation parameters usually converge more easily. In other words, finding possible corresponding points is a part of the problem for GP-ICPR, the ICP, or its variants. In these cases, finding the translational parameters is more difficult than the rotational parameters since the estimated translational parameters are simply the translational differences in the centroids of the selected corresponding points, unless a weighted least square method is employed. In the case of direct georeferencing methods, to find the translational parameters is easier, only because a good set of corresponding points is already given.

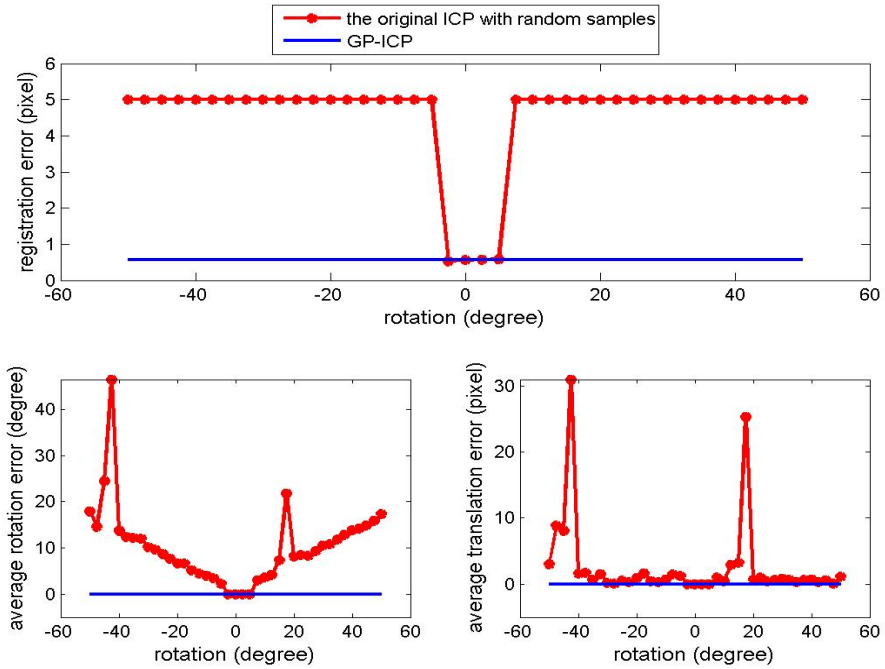


FIGURE 5.9: Convergence region test of the point clouds from the cactus without translation.

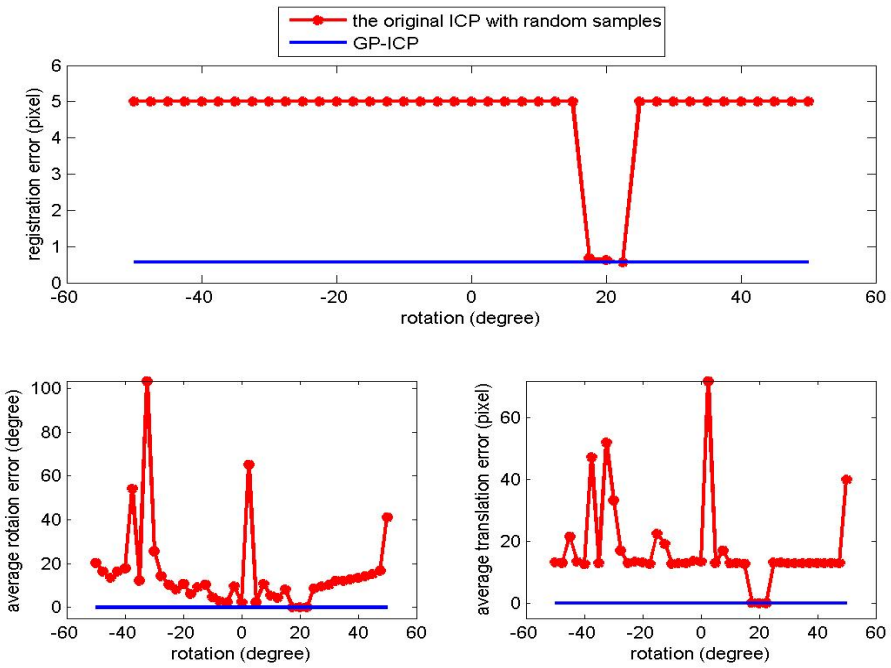


FIGURE 5.10: Convergence region test of the point clouds from the cactus with a translation, the translation 1, i.e. (H/4, L/4, W/2).

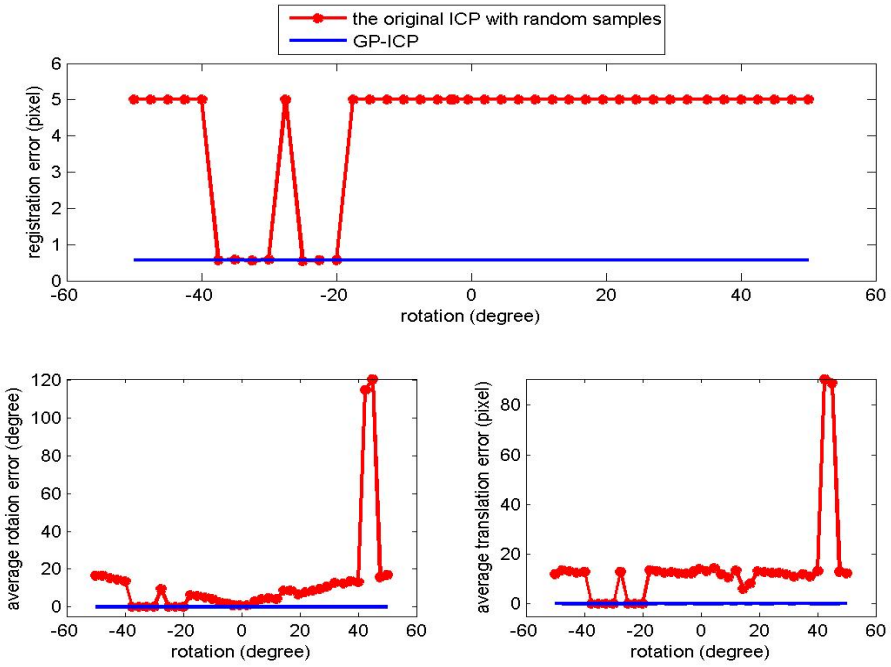


FIGURE 5.11: Convergence region test of the point clouds from the cactus with a translation, the translation 2, i.e. (-H/4, L/4, W/2).

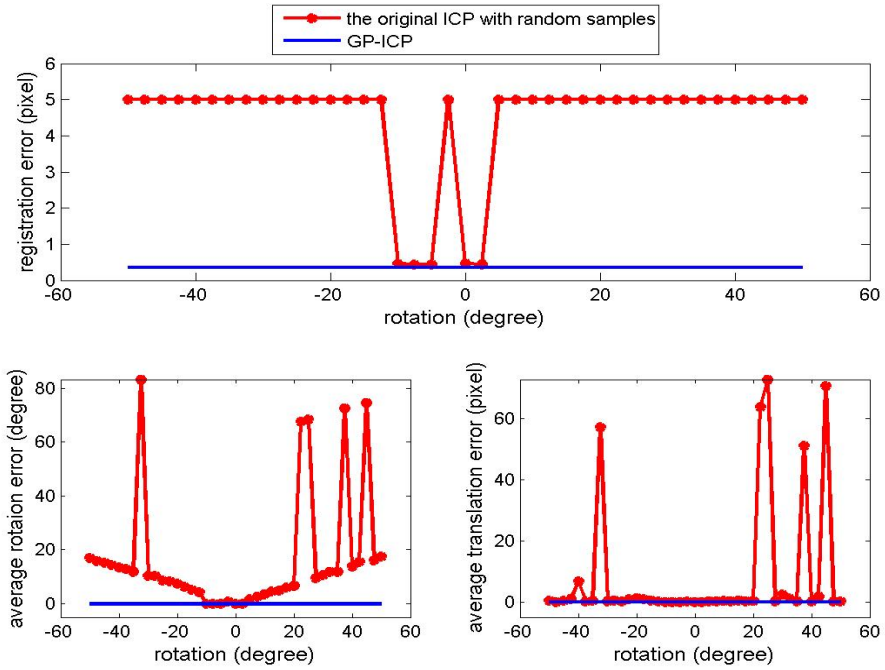


FIGURE 5.12: Convergence region test of the point clouds from the golf club without translation.

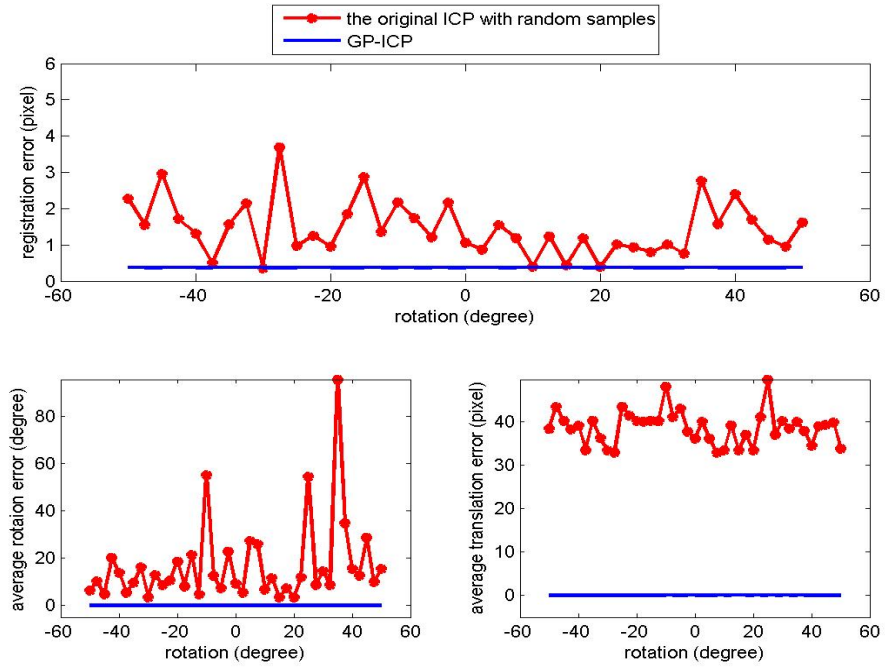


FIGURE 5.13: Convergence region test of the point clouds from the golf club with a translation, the translation 1, i.e. (H/4, L/4, W/2).

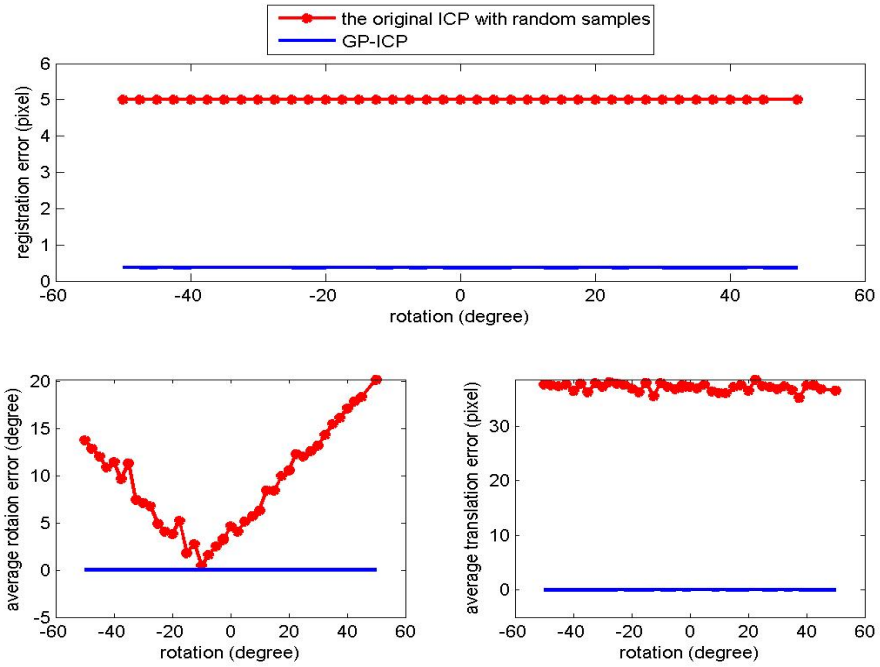


FIGURE 5.14: Convergence region test of the point clouds from the golf club with a translation, the translation 2, i.e. (-H/4, L/4, W/2).

5.4 Effect of the initial threshold for distance

Figure 5.15 shows the initial threshold for distance, $T_{distance}^{iter=0}$, in the convergence tests performed in the previous section. As explained earlier, $T_{distance}^{iter=0}$ was decreased from a maximum value to zero in these tests. If a solution close to the truth is found, then the iteration was stopped and the current $T_{distance}^{iter=0}$ was recorded. For example, in the case of the cactus without translation, i.e. Figure 5.15-(a), the maximum $T_{distance}^{iter=0}$ is set to 100 pixels. Basically, we would like to have a constant $T_{distance}^{iter=0}$ over the entire convergence region of a registration method. However, in the cases of the simulated data, the required $T_{distance}^{iter=0}$ with which a good estimation of the true relative transformation of the point clouds is obtained, changes in an unpredictable manner mainly because its point density is much lower than that of either close-range or terrestrial laser scanner data. The probability of finding a good set of corresponding points is decreased with a larger $T_{distance}^{iter=0}$. Fortunately, it will be shown in later chapters that a similar level of the rotational convergence region with the simulated datasets is maintained with the tests with point clouds on both close-range and terrestrial laser scanners with a smaller deviation in $T_{distance}^{iter=0}$.

5.5 Summary

In this chapter, GP-ICPR improved the precision of GP-ICP by as much as a factor of 5 in the presence of the noise described in Section 5.2.4 as shown in Table 5.10. It also indicates that GP-ICPR improves the precision of the ICP by approximately the same level since the precision of GP-ICP is similar to or smaller than that of the ICP or its variants unless an outlier procedure is included in a registration algorithm. In addition, in the cases of the simulated datasets, GP-ICP and GP-ICPR achieved much larger rotational convergence regions, at least $\pm 50^{\circ}$, than the original ICP with random sample developed by Masuda and Yokoya (1995) as shown in Figures 5.9 - 5.14.

From this section's results, three statements on the performance of GP-ICPR and GP-ICP with the simulated data are made as follows:

- Using GP-ICPR, a set of relative transformation parameters that provide a lower registration error is obtained and its D_{std}^{ps} is much closer to the estimated CRLB of the registration error than other methods, e.g. GP-ICP.
- GP-ICPR provides a more accurate estimation of the transformation parameters for the simulated datasets.
- The tests of GP-ICPR with simulated datasets infer that GP-ICPR provides a set of the estimated relative transformation close to the truth with the registration of two point clouds from laser scanners as well.

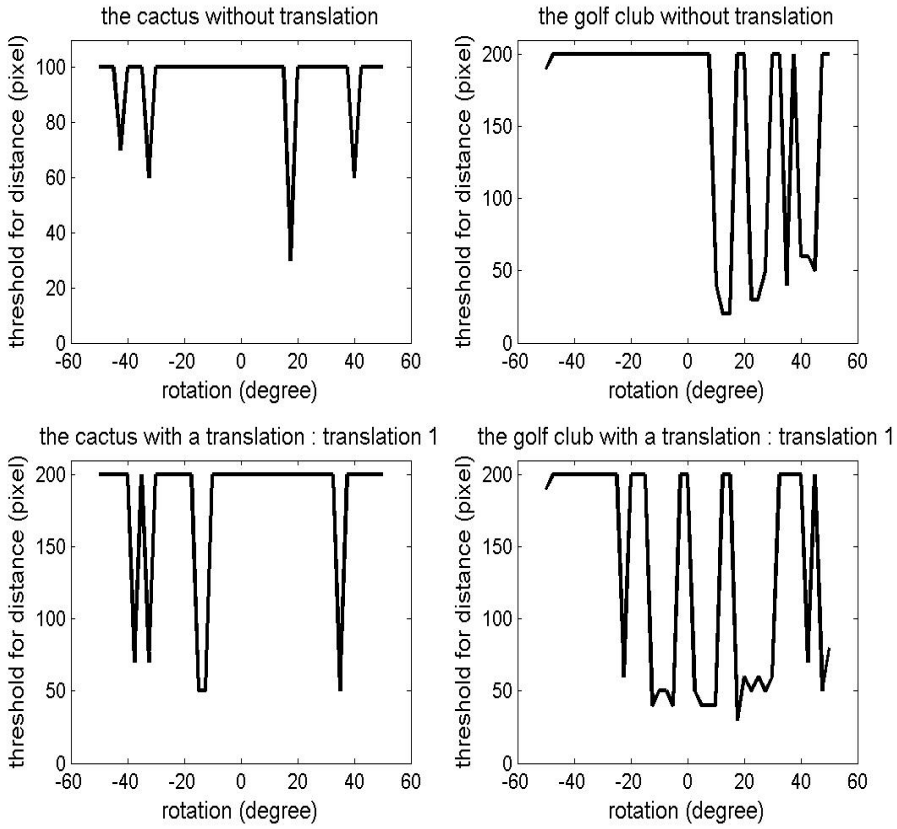


FIGURE 5.15: $T^{iter=0}_{distance}$ for the convergence region tests in Section 5.3.

Chapter 6

Experiment II : Close-range laser scanner data

In this chapter, the precision of the estimated transformation parameters and the convergence region in the registration of close-range scanner data with GP-ICP will be evaluated. The close-range scanner data were obtained from the Stanford 3D scanning repository (Stanford Computer Graphic Laboratory, 2006) which were scanned with a Cyberware 3030MS (Cyberware, 2006). These datasets were registered by a modified ICP algorithm (Turk and Levoy, 1994) and the relative transformation parameters estimated from their modified ICP algorithm are also available through the Stanford 3D scanning repository. Unfortunately, the manufacturer of this laser scanner does not provide enough information to evaluate the positional and angular uncertainties, i.e. σ_r and σ_a, of the laser scanner. Therefore, the accuracy of either GP-ICP or their modified ICP (Turk and Levoy, 1994) can not be discussed. In addition, GP-ICPR could not be applied to these close-range scanner data for the same reason. However, by observing the change of curvature of the registered point clouds, the method that provides a better solution was determined analytically.

For the registration of the Stanford datasets, Turk and Levoy (1994) applied their modified ICP method after constructing meshes, i.e. triangulation, of the point clouds. On the other hand, GP-ICP and GP-ICPR directly use a point cloud without a procedure for constructing the meshes. As GP-ICPR tries to minimise the distances between a set of possible corresponding points and surfaces, this method is more concerned with the smoothness between two registered point clouds. Therefore, some differences between the estimated transformation parameters are expected between GP-ICPR and Turk and Levoy's method. Note that Turk and Levoy's (1994) method will be referred to as the Stanford method from now until the end of the dissertation.

6.1 Registration error analysis

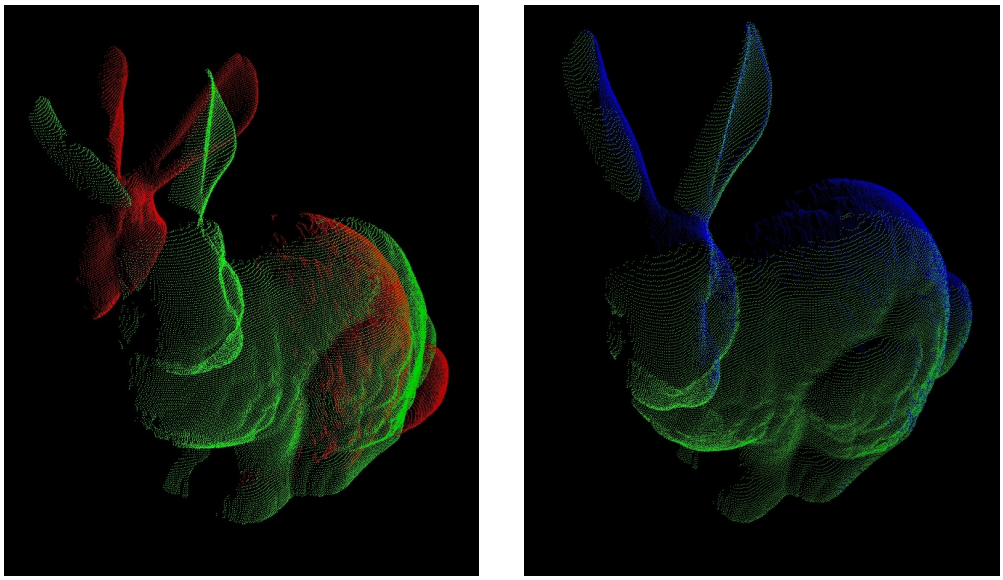
For tests of the precision of GP-ICP, two datasets from the Stanford 3D scanning repository will be used: the "Stanford bunny" and the "happy Buddha" as these are referred to by computer graphics researchers. These datasets have approximately ten point clouds and two sets of them from both the Stanford bunny and the happy Buddha will be used in this section. One point cloud of the Stanford bunny was named "bunny000" following its original file name, bun000.ply. The other was named "bunny090", again following its original file name, bun090.ply. One point cloud of the happy Buddha was named "happybuddha_StandRight_0" after its original file name, happyStandRight_0.ply. The other was named "happybuddha_StandRight_48" again after its original file name, happyStandRight_48.ply. It must be noted that the pre-processed data from one of the Stanford graphics research group's smoothing procedures, e.g. Curless and Levoy (1996), were not used in this thesis. Instead, a set of raw point clouds from the close-range scanner was utilised to evaluate the performance of GP-ICP. In addition, the pixel sizes of the Stanford bunny and the happy Buddha are about 2.30mm and 1.14mm, respectively.

6.1.1 The Stanford bunny

Diagrams before registration and after registration of the Stanford bunny are presented in Figure 6.1 and the details of the Stanford bunny are listed in Table 6.1. The bunny000 and bunny090 share about 20% of the overlapping region. Note that the number of points in the overlapping region of the point clouds in Table 6.1 was counted by GP-ICPR. Let us discuss the registration of the Stanford bunny using GP-ICP as shown in Table 6.3. Since Turk and Levoy (1994) only provided the estimated transformation parameters and did not provide any kind of registration errors, the registration errors of GP-ICP could not be directly compared with their method. The mean and standard deviation of the distance between a point and its corresponding surface, D^{ps}_{mean} and D^{ps}_{std}, are on the order of 0.20 pixel. Once again, the mean distance between selected corresponding points, D^{pp}_{mean}, is three times greater than D^{ps}_{mean}. Since D^{ps}_{mean} and D^{ps}_{std} are much smaller than a size of a pixel, it could be stated that GP-ICP provides a precise relative transformation between the bunny000 and the bunny090.

	bunny000	bunny090
the Stanford bunny	8859/40256 = 22.0%	8859/30379 = 29.1%

TABLE 6.1: Sizes of the overlapping regions of parts of the Stanford bunny are presented as percentage of the whole dataset, i.e. the denominator is the total number of points in a point cloud and the numerator is the number of points in the overlapping region.



(a) bunny000 (green) + bunny090 (red) (b) bunny000 (green) + bunny090 (blue)

FIGURE 6.1: Before and after the registration of the parts of the Stanford bunny, (H, W, L) $\simeq$ (15cm, 15cm, 12cm), using GP-ICP. The bunny000 is green in the above. The bunny090 is red in (a) and is blue in (b).

	ω [degree]	ϕ [degree]	κ [degree]	x [mm]	y [mm]	z [mm]
the Stanford method	173.03	89.82	-173.11	0.02	-0.03	-0.07
GP-ICP	173.55	89.77	-173.53	0.18	-0.06	-0.24
difference	-0.51	0.05	0.42	-0.16	0.02	0.17

TABLE 6.2: Estimated relative transformation parameters between the bunny000 and the bunny090 by the Stanford method and GP-ICP.

	registration error [mm]	registration error [pixel]
D^{pp}_{mean}	1.70	0.74
D^{ps}_{mean}	0.45	0.20
D^{ps}_{std}	0.33	0.14

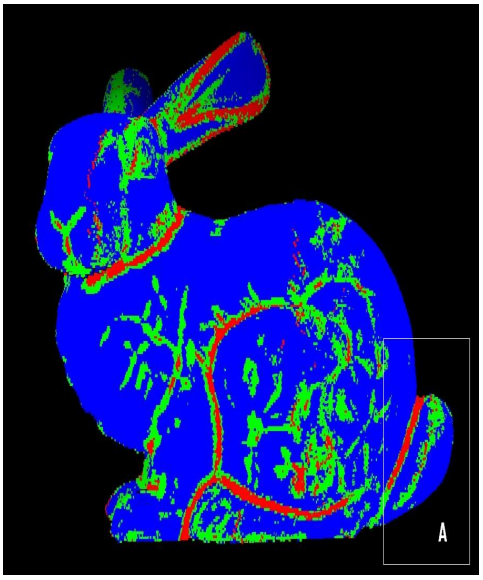
TABLE 6.3: Registration errors of GP-ICP with the bunny000 and the bunny090.

To test the estimated relative transformation parameters between the bunny000 and the bunny090, which are presented in Table 6.2, two different methods were used: Turk and Levoy's method (1994) named the Stanford method and GP-ICP. The rotational and translational maximum differences are approximately 0.5° and 0.2mm $\simeq$ 0.1 pixel, respectively, although the absolute difference in the estimated parameters does not provide much insight on the question of the accuracy of the estimation procedure.

Let us observe the change of curvature of the registered point clouds of the Stanford bunny in Figure 6.2. As mentioned earlier, this is an analytical resource to identify which method is better than the other. The superiority of one method over an other can then be decided, but only if the difference in the changes of the curvature of two registered point clouds by two different methods is sufficiently large to be observed by an user. In other words, if the change of curvature in a flat region of a registered point cloud by one method is relatively larger than that of the other method, it can be declared that the first method does not provide a globally smooth set of the registered point cloud. Figures 6.2 and 6.3 show the colour maps of the change of the curvature and the distance between a point and its corresponding surface of the registered point clouds of the Stanford bunny using the Stanford method and GP-ICP. The colour maps of M_{cc}, i.e. Eq. 4.22, are very similar except for the tail region of the Stanford bunny, i.e. region A in Figure 6.2. A significant difference in the change of the curvature exists in this region but it is not large enough to claim the superiority of GP-ICP.

(a) A picture of the Stanford bunny

(b) The Stanford method



(c) GP-ICP

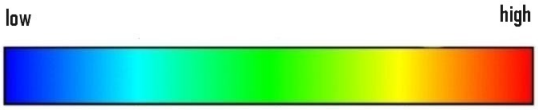
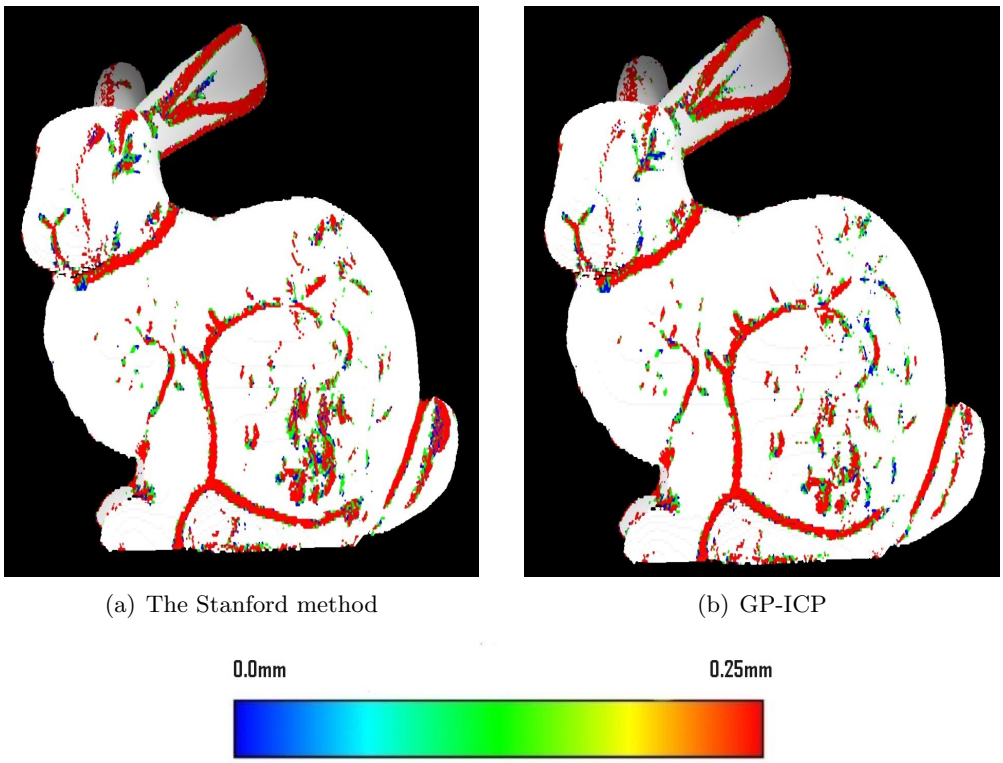


FIGURE 6.2: Colour maps of the change of curvature of the registered point clouds, bunny000 and bunny090, using the Stanford method and GP-ICP. Note that the change of curvature is unitless.



(a) The Stanford method (b) GP-ICP

FIGURE 6.3: Colour maps of the distance between a point and its local surfaces are presented in (a) and (b) in the cases of the Stanford method and GP-ICP. A lower curvature region was assigned as white and the same threshold for M_{cc} was used for two data.

6.1.2 The happy Buddha

Diagrams before registration and after registration of the happy Buddha dataset by the Stanford method and GP-ICP are presented in Figure 6.4 and the details of the point clouds are listed in Table 6.4. The happybuddha_StandRight_0 and the happybuddha_StandRight_48 share about 8% of the overlapping region. Note that the number of points in the overlapping region of the point clouds in Table 6.4 was again counted by GP-ICPR. Table 6.5 presents the registration errors from GP-ICP. D^{ps}_{mean} and D^{ps}_{std} are approximately to the order of 0.2 pixel. Once again sub-pixel precision is achieved and D^{pp}_{mean} is slightly greater than D^{ps}_{std}.

Let us discuss the estimated relative transformation parameters presented in Table 6.6. The rotational and translational maximum differences are approximately 0.4° and 1.1mm $\simeq$ 0.8 pixel, respectively. A larger difference in the estimated transformation parameters is observed, especially translational parameters, than in the case of the registration of the Stanford bunny. Since the difference in the estimated translational parameters is in the order of one pixel, a significant difference in the change of curvature is expected to be observed.

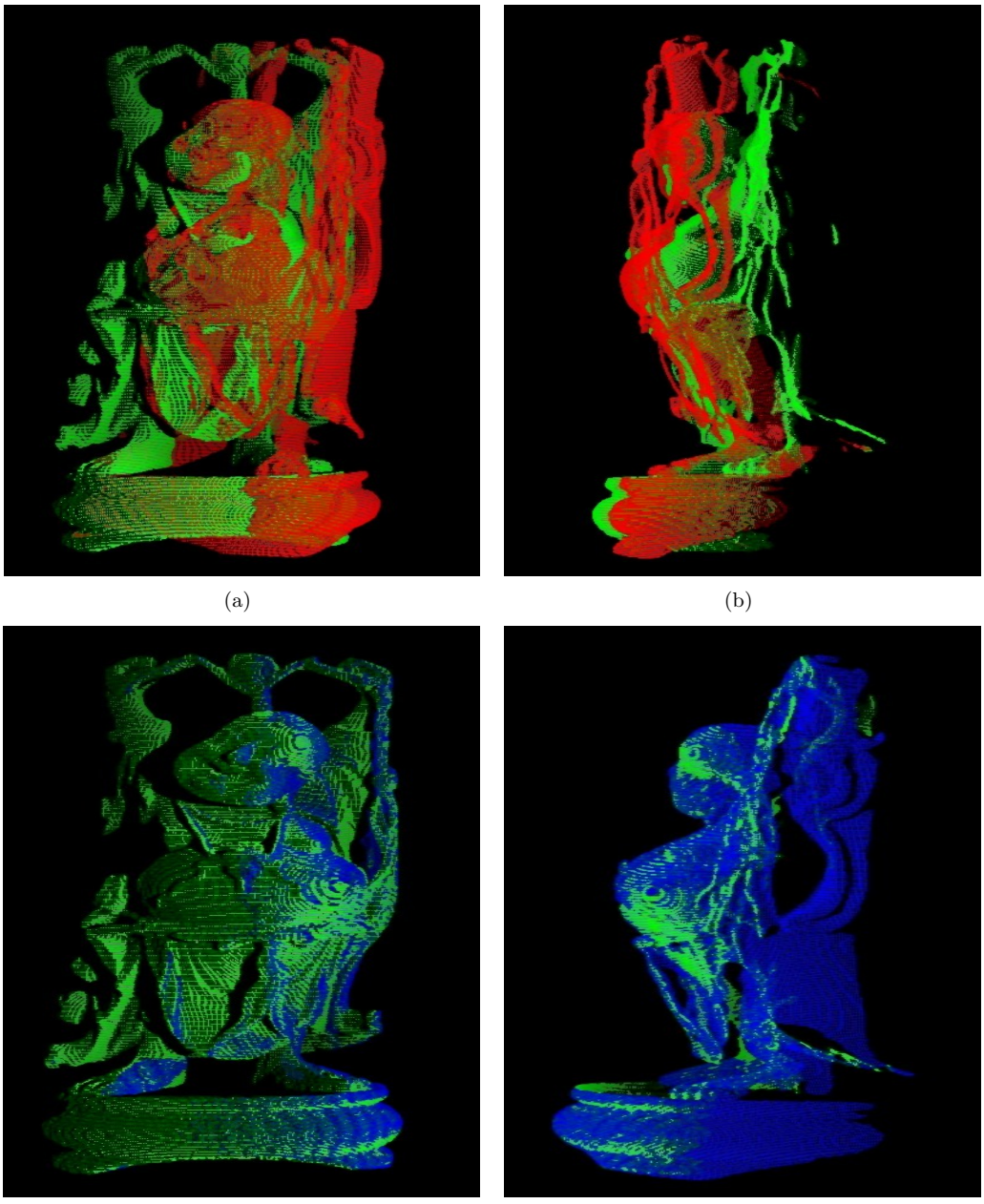
	happybuddha_StandRight_0	happybuddha_StandRight_48
the happy Buddha	6352/78056 = 8.1%	6352/69158 = 9.2%

TABLE 6.4: Sizes of the overlapping regions of parts of the happy Buddha represented as percentage of the whole dataset, i.e. the denominator is the total number of points in a point cloud and the numerator is the number of points in the overlapping region.

Now visually investigate the change of curvature of the registered point cloud of the happy Buddha. Figures 6.5 and 6.6 show the change of curvature and the distance of a point from its local surface in the registered point clouds by the Stanford method and GP-ICP, respectively. It can be clearly observed that regions A and B of Figure 6.5 have large differences in M_{cc}. Therefore, it is assured that GP-ICP provides a better solution, which maintains the geometric smoothness of the point clouds after registration, since the regions A and B in Figure 6.5 must be low curvature regions. Furthermore, from the colour maps of the distance of a point from its local surface, Figure 6.6, it is also observed that the distance between a point to its local surface in the regions A and B registered by the Stanford method is to the order of 0.25mm $\simeq$ 0.18 pixel.

	registration error [mm]	registration error [pixel]
D^{pp}_{mean}	0.34	0.30
D^{ps}_{mean}	0.23	0.20
D^{ps}_{std}	0.20	0.18

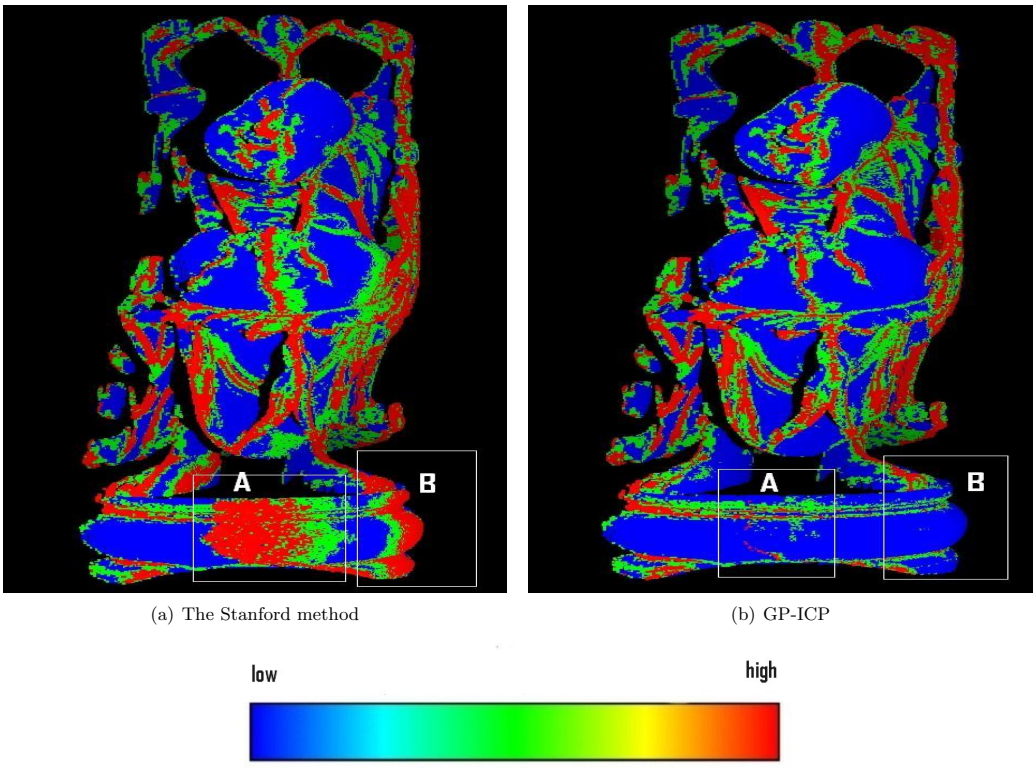
TABLE 6.5: Registration errors of the registered happy Buddha dataset by GP-ICP using the happybuddha_StandRight_0 and the happybuddha_StandRight_48.



(a) (b)

(c) (d)

FIGURE 6.4: Before and after registration of the parts of the happy Buddha using the Stanford method and GP-ICP, (H, W, L) $\simeq$ (20cm, 8cm, 7cm). Happybuddha_StandRight_0 is green in the above. Happybuddha_StandRight_48 is red in (a) and (b) and is blue in both (c) and (d).



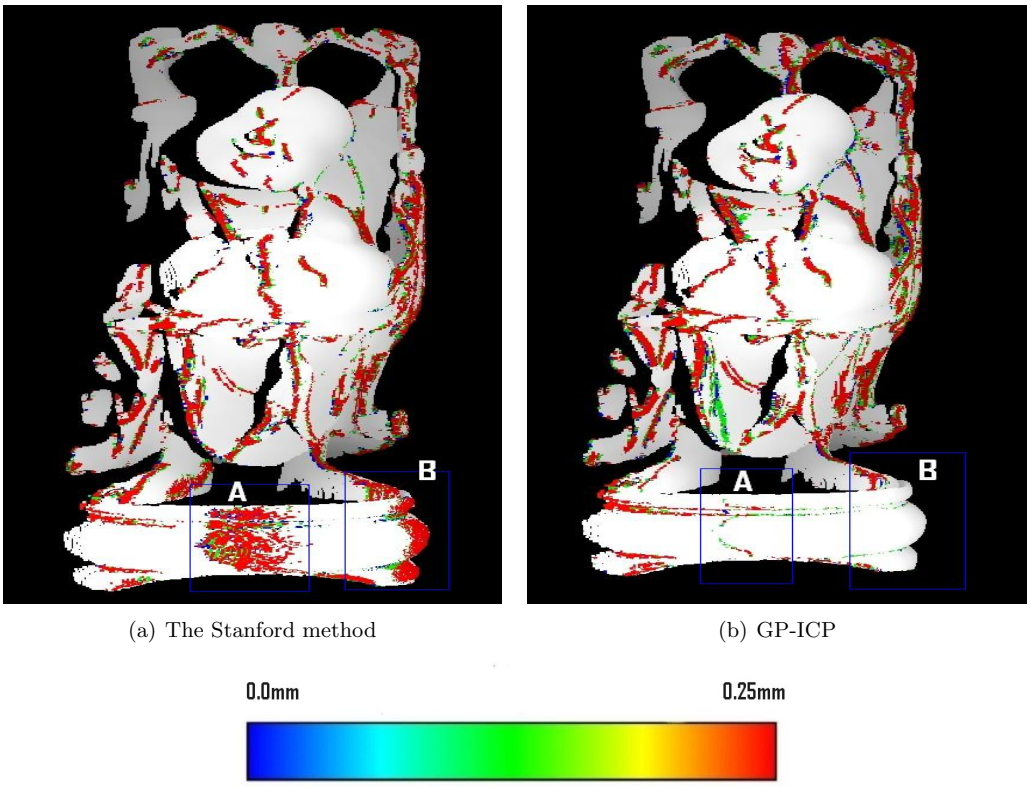
(a) The Stanford method

(b) GP-ICP

FIGURE 6.5: The top is a picture of the happy Buddha. The colour maps of the change of curvature of the registered point clouds, happybuddha_StandRight_0 and happybuddha_StandRight_48, are presented in (a) and (b) registered by the Stanford method and GP-ICP. Note that the change of curvature is unitless.

	ω [degree]	ϕ [degree]	κ [degree]	x [mm]	y [mm]	z [mm]
the Stanford method	0.064	48.32	-0.33	-0.55	0.0090	0.92
GP-ICP	0.028	47.95	0.098	0.55	-0.054	-0.020
difference	0.035	0.37	-0.42	-1.1	0.063	0.94

TABLE 6.6: Estimated relative transformation parameters of happybuddha_StandRight_0 and happybuddha_StandRight_48 using the Stanford method and GP-ICP.



(a) The Stanford method (b) GP-ICP

FIGURE 6.6: Colour maps of the distance between a point and its local surfaces are presented in (a) and (b) in the cases of the Stanford method and GP-ICP. A lower curvature region is assigned as white and the same threshold for M_{cc} is used for two data.

6.2 Convergence region

In this section, the convergence region of GP-ICP will be evaluated using close-range laser scanner datasets, i.e. the Stanford bunny and the happy Buddha. The scope of the tests can be stated as follows:

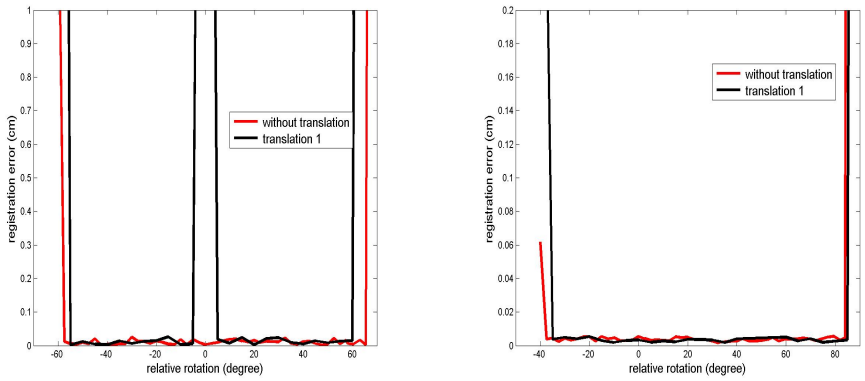
- As mentioned in Section 5.3, the convergence region of GP-ICP is approximately similar to that of GP-ICPR. In other words, convergence region evaluated with the close-range scanner datasets in this section is valid for either GP-ICP or GP-ICPR.

- Unlike the convergence region tests with the simulated data, the Stanford bunny and the happy Buddha were, from the registered state, rotated around an axis in both clockwise and counter-clockwise until GP-ICP fails to obtain a solution. Therefore, a point cloud's rotational convergence region can be asymmetric, e.g. $-40^\circ < R^{convergence} < 20^\circ$, where $R^{convergence}$ is the rotational convergence region of a point cloud.

- Since the true transformation was known and this test was designed to evaluate the convergence region of GP-ICP, the algorithm was stopped if the difference between the true and the estimated transformation parameters was sufficiently small, regardless of the magnitude of the registration error in the last iteration. In other words, in this test, GP-ICP did not try to find the smallest possible registration error. Therefore, in the plot of D_{std}^{ps} in Figure 6.7, a little fluctuation is observed in the registration errors. In addition, a similar fluctuation is observed in the errors of the estimated transformation parameters as shown in Figure 6.8.

- $T_{normal}^{iter=0}$ was again set to be $|R^{initial}| + 10^\circ$. A threshold for distance, $T_{distance}^{iter=0}$, was changed from a maximum value to zero with increment of 5cm. As stated in Section 5.3, if the final $T_{distance}^{iter=0}$ is zero, it means that a registration algorithm failed to find a good solution.

The results of the convergence test of GP-ICP with the two Stanford datasets are presented in Figure 6.7. In the case of no translation, the rotational convergence region of two datasets from the happy Buddha is about $-50^\circ < R^{convergence} < 60^\circ$ and that of two datasets from the Stanford bunny is about $-40^\circ < R^{convergence} < 80^\circ$. As the happy Buddha was translated by (H/4, L/4, W/2) which was named "translation 1" of the object in Section 5.3, we have almost the same rotational convergence region as the cases without translation of the happy Buddha. However, in the case of the Stanford bunny translated by its translation 1, there is a region of discontinuity in the rotational convergence region, about $\pm 5^\circ$ from its translation 1 as shown in Figures 6.7(a) and 6.8(c).

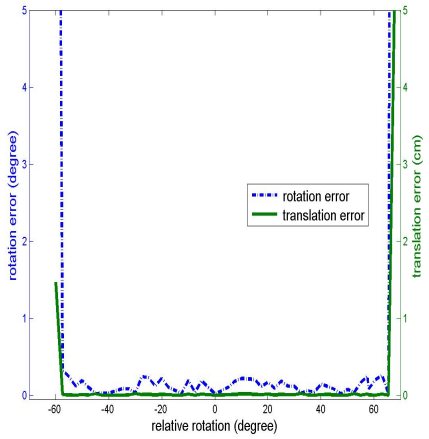


(a) D_{std}^{ps} of the Stanford bunny (b) D_{std}^{ps} of the happy Buddha

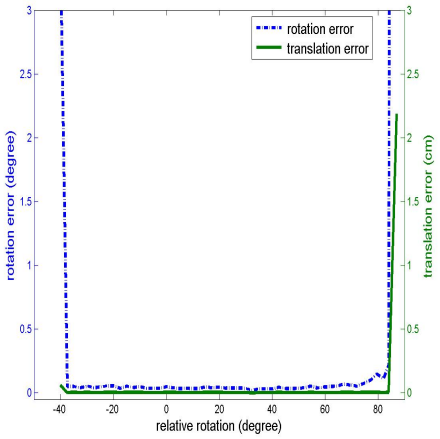
FIGURE 6.7: Convergence region tests of datasets from the Stanford 3D repository for GP-ICP. Red and black curves represent the cases without and with translation, respectively. Therefore, the zero rotations of the curves represent different relative transformations of the data. (a) and (b) are the D_{std}^{ps} of the registered point clouds: the Stanford bunny and the happy Buddha, respectively.

This is a kind of slide effect mentioned by Rusinkiewicz and Levoy (2001). In their cases, the slide effect refers to the case in which it is difficult to find a set of corresponding points when there are only a small number of geometrically distinguishable features, e.g. a point cloud of an engraved plate. This can also be explained in terms of the method of collecting samples for finding a set of possible corresponding points. In GP-ICP, we first select a set of the nearest neighbours in the point cloud of a query point from the other point cloud. Then the best possible corresponding point for the query point is selected as explained in Section 4.3. For example, the nearest red neighbours of a green query point in the region A of Figure 6.9 are not the corresponding points of the green query point. Its true corresponding point is far from region A. In many cases, this problem is avoided using GP-ICP for finding corresponding points as observed in the convergence test of the happy Buddha in the cases of either with or without translation. This problem exists in the Stanford bunny but not in the happy Buddha since the happy Buddha has more geometrically distinct features, i.e. higher curvature points, than the Stanford bunny.

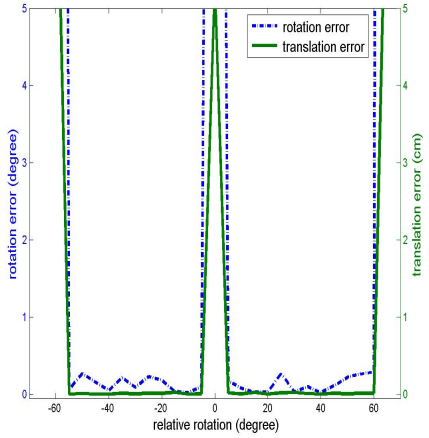
The asymmetry in the rotational convergence region is mainly caused by the ratio of overlapping regions and the geometric shape of an object. It is also observed that the asymmetry in the Stanford bunny is relatively smaller than that of the happy Buddha. Figure 6.10 is utilised to explain this asymmetry in the rotational convergence region. The Stanford bunny is chosen for the explanation of asymmetry in the rotational convergence region since it is much more visually clearer and easier to explain than it is for the happy Buddha. The positive rotation is indicated by the direction of rotation



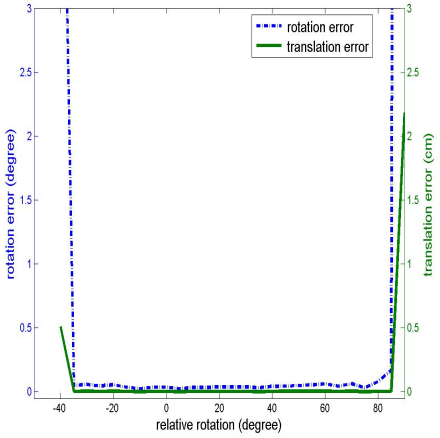
(a) No translation, Stanford bunny



(b) No translation, happy Buddha



(c) Translation 1, Stanford bunny



(d) Translation 1, happy Buddha

FIGURE 6.8: Registration errors of data from the Stanford 3D repository using GP-ICP. (a) and (b) are the errors in the estimated transformation parameters by GP-ICP for the Stanford bunny and the happy Buddha, respectively, without a relative translation. (c) and (d) are the same values when a point cloud is translated to translation 1 of the datasets.

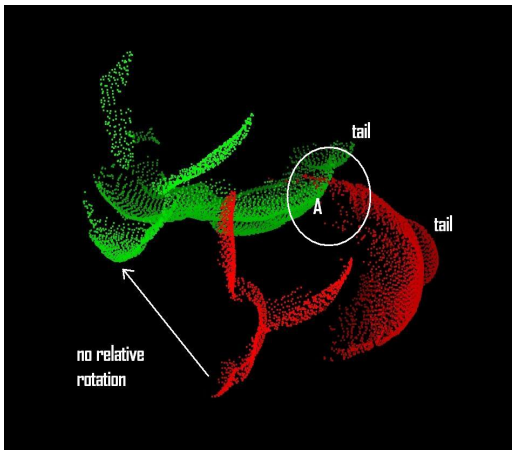


FIGURE 6.9: Bunny000 and bunny090 translated to translation 1 with zero relative rotation. There is no relative rotation and the relative translation is (H/4, L/4, W/2) where H, L, and W are the height, length, and width of the object, i.e. the Stanford bunny.

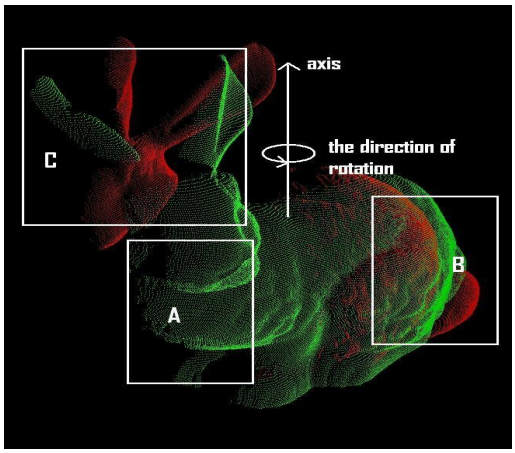
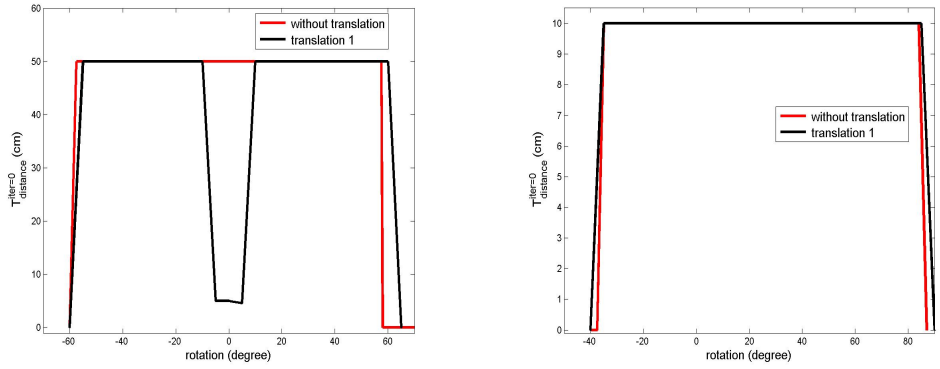


FIGURE 6.10: Schematic outline of the convergence tests with the Stanford bunny. The axis and the direction of the initial rotations for the convergence tests are presented. The red point cloud, the bunny090, is rotated around the axis either to or against the direction of rotation in the picture.

in Figure 6.10.

First, it is clearly seen that there are almost no corresponding points in region A between bunny000 and bunny090. Furthermore, region B has a larger set of corresponding points between the point clouds than region C. In addition, in region B we find many more distinctive regions than in region C, in terms of the geometric shape of the regions. Although the parts of the Stanford bunny's ears are in region C, the change of curvature around region C is much lower than that of region B. The presence of a higher curvature region around the ears is not much help for finding a set of corresponding points. Simply speaking, with either GP-ICP or a modified ICP algorithm based on geometric roughness, the algorithm has a much larger probability of finding a set of possible corresponding points in region B than in region C. That is why there is approximately 60° rotational convergence region in the positive direction of rotation. In the negative direction of rotation, there is a 50° rotational convergence region. In general, for this test of the convergence region of GP-ICP, we have at least ±40° rotation convergence region with two point clouds from either the Stanford bunny or the happy Buddha. As mentioned in Section 1.2, this rotational convergence region is large enough for practical applications using either terrestrial or close-range laser scanners.

Like the results of Section 5.4, the required $T_{distance}^{iter=0}$ with which a good estimation of the true relative transformation of the point clouds is obtained, changed in an unpredictable manner in the convergence region test with the simulated datasets. In the case of close-range laser scanner data, GP-ICP provides a precise estimation of the relative transformation of the point clouds with a steadier and smaller deviation in $T_{distance}^{iter=0}$ as presented in Figure 6.11.



(a) Stanford bunny (b) Happy Buddha

FIGURE 6.11: Initial thresholds for distance, i.e. $T_{distance}^{iter=0}$, for the registration tests of the Stanford bunny and the happy Buddha using GP-ICP without translation.

6.3 Summary

The precision of the registration error of a set of close-range scanner data from the Stanford 3D scanning repository (Stanford Computer Graphic Laboratory, 2006) was evaluated and the estimated parameters of GP-ICP were directly compared with those of a registration method by Turk and Levoy (1994) named the Stanford method. In addition, the accuracy of the registrations by GP-ICP and the Stanford method was analytically evaluated by the colour maps of the change of curvature. Furthermore, the rotational convergence region of GP-ICP was on the order of 10° in the cases of these close-range scanner data as shown in Figure 6.8. It was found that the estimated transformation parameters by GP-ICP with close-range laser scanner datasets is better than or equal to those of the Stanford method.

Chapter 7

Experiment III : Terrestrial laser scanner data

In this chapter, the precision, accuracy, and convergence region in the registration of terrestrial laser scanner data using GP-ICPR will be evaluated. As mentioned earlier in Chapter 5, it is difficult to evaluate the accuracy of the estimation in real applications since any knowledge of the truth is not available. Therefore, the scope or argument of the accuracy tests in this chapter must be stated as follows:

- The bias in the estimated transformation parameters by a registration method with terrestrial laser scanner data can not be directly measured.
- The magnitude of the bias in the estimated transformation parameters can be observed indirectly using a colour map of the geometric properties, e.g. the change of curvature. A colour map of radiometric properties from a laser scanner, e.g. intensity of the returning laser, can be also used, although these values from two or more point clouds must be normalised and this procedure is not trivial.
- Note that two proposed statistical inferences, Eqs. 4.36 and 4.37, are independent of each other and are also based on the CRLB of the registration error.

7.1 Registration error analysis and accuracy tests

Two datasets from the Ayuthaya Buddha in Thailand and the Agia Sanmarina church in Greece were scanned by a Riegl LMS-Z210 (Riegl, 2006) and a Leica HDS2500 (Leica, 2006), respectively. These datasets were used in Chapter 3 and were registered using the direct georeferencing method, GP-ICP, GP-ICPR, and a commercial software package, Cyclone (Leica, 2006, Version 5.0). Note that only the datasets from the Agia Sanmarina church were tested for registration by Cyclone since the number

of targets in the case of the Ayuthaya dataset is not large enough for registration by this software. In addition, it must be noted that the accuracy of either GP-ICP or GP-ICPR has already been tested with simulated data in Chapter 5. GP-ICPR was shown to provide a more accurate estimation in the registration of the simulated datasets.

7.1.1 Ayuthaya Buddha

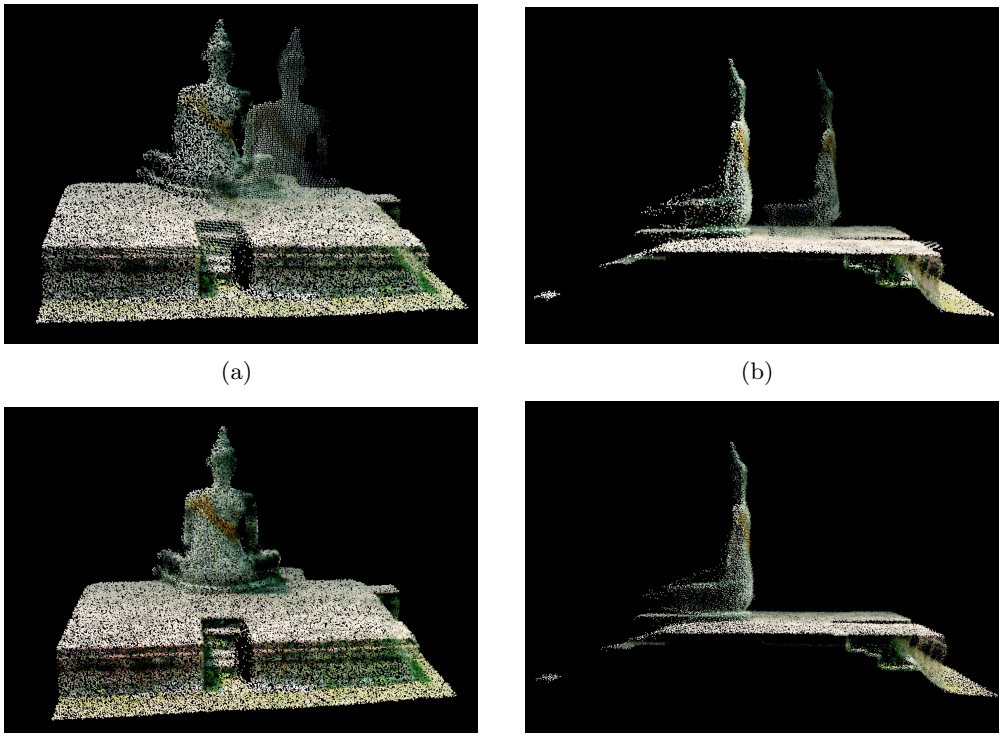
Diagrams before registration and after registration of the Ayuthaya Buddha datasets by GP-ICPR are given in Figure 7.1. The total numbers of points in the point clouds and the number of points in the overlapping regions is found in Table 7.1. Note that there is a large difference in the size of the overlapping regions as a proportion of total point cloud size. The colour maps of the distance between a point and its local surface are presented in Figure 7.2. We clearly see a higher curvature area which is supposed to be a lower curvature region in Figure 7.2(b). Without any doubt, it is concluded that the direct georeferencing method provides the largest biased estimation among the three methods. It is also observed that GP-ICP and GP-ICPR provides a geometrically correct registration result as shown in Figure 7.2(c) and Figure 7.2(d), respectively.

The plots of the estimated transformation parameters versus iteration number are presented in Figure 7.3. It is observed that the set of the estimated parameters of GP-ICP is shifted by GP-ICPR. However, different from the cases of the simulated data shown in Figures 5.5 and 5.8, it can not be confirmed that GP-ICPR guides the results of GP-ICP to the truth.

	point cloud 1	point cloud 2
Ayuthaya Buddha	6676/149986 = 4%	3805/5639 = 67%

TABLE 7.1: Sizes of the overlapping regions of the point clouds from the Ayuthaya Buddha represented in percentage.

From Table 7.2, the registration error of GP-ICPR, i.e. D^{ps}_{mean} or D^{ps}_{std}, is smaller than that of GP-ICP and is larger than the Cramer-Rao lower bound of the registration error of the point clouds. In addition, there is about 30% improvement in the closeness to the CRLB of the registration error. Furthermore, GP-ICPR discovers a set of inliers, which provides a smaller registration error than that from the direct georeferencing method and also satisfies the proposed statistical inferences within GP-ICPR with 95% confidence level.



(a) (b)

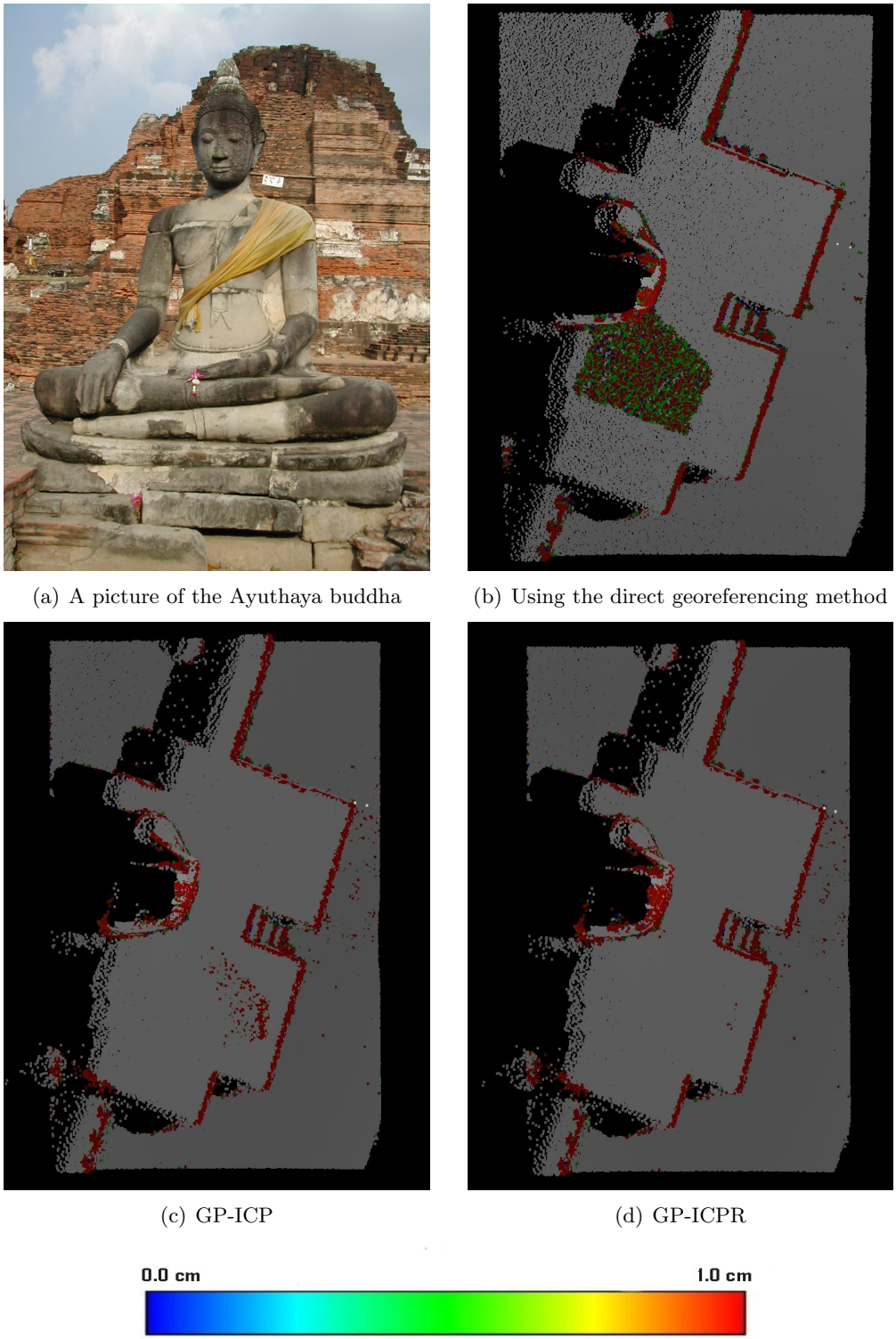
(c) (d)

FIGURE 7.1: Ayuthaya Buddha statue in Thailand scanned by a Riegl LMS-Z210. The details of these point clouds can be found in the caption of Figure 3.2. (a) and (b) are before registration. (c) and (d) are after the registration by GP-ICPR.

	ratio of inliers inliers/total	D^{ps}_{mean} [cm]	D^{ps}_{std} [cm]	CRLB [cm]
GP-ICP	·	2.2441	2.5501	·
GP-ICPR	932/1394 = 67%	0.1216	1.4992	1.3094

Registration errors compared to the CRLB using two different methods	
50.0 %	GP-ICP (Ayuthaya)
87.3 %	GP-ICPR (Ayuthaya)

TABLE 7.2: Top table shows the mean and standard deviation of the distance between points and their corresponding surfaces using two methods, i.e GP-ICPR and GP-ICPR. It also presents the ratio of inliers and the Cramer-Rao lower bound (CRLB) of the registration errors estimated from Eq. 4.25. Note that neither the ratio of inliers nor the CRLB is available in the case of GP-ICP. The bottom bar graph represents how much the values of D^{ps}_{std} in two different proposed methods are close to the Cramer-Rao lower bound of the registration error of these point clouds. For example, if it is 100%, then the resultant registration error is as small as it can be, i.e. the CRLB.



(a) A picture of the Ayuthaya buddha

(b) Using the direct georeferencing method

(c) GP-ICP

(d) GP-ICPR

FIGURE 7.2: Picture of the Ayuthaya Buddha statue and colour maps of the distance between a point and its corresponding surface of the registered point cloud using three methods: the direct georeferencing method, GP-ICP, and GP-ICPR.

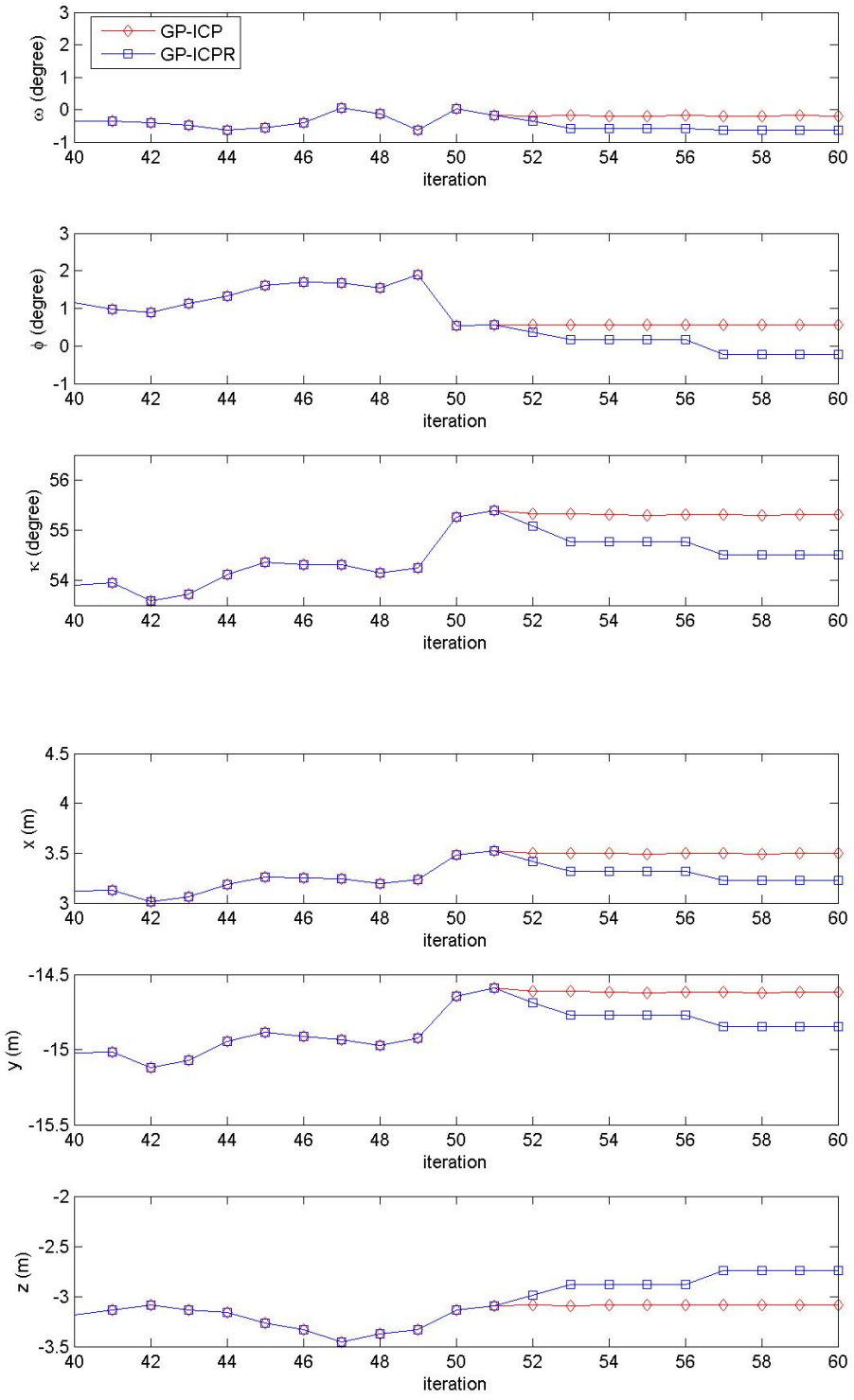


FIGURE 7.3: Estimated transformation parameters in the cases of GP-ICP and GP-ICPR versus iteration.

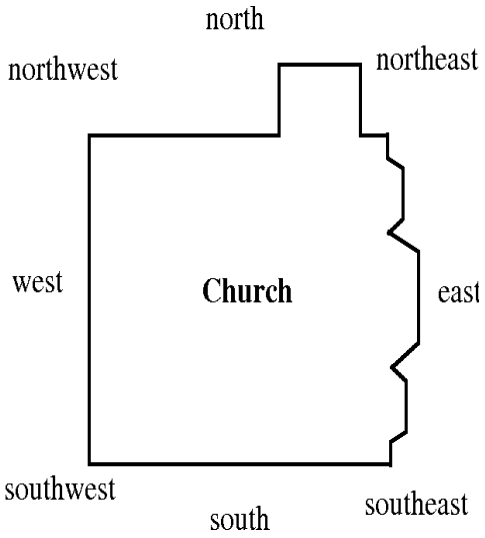
7.1.2 Agia Sanmarina Church

The Agia Sanmarina Church in Greece was scanned from nine different locations around the church as shown in Figure 7.4. Seven of the point clouds were named after the locations of the laser scanner, e.g. east or southeast. Two of the point clouds were named crane1 and crane2 since they were scanned from a crane in order to capture the top of the church. A schematic drawing (top view), a picture and the point clouds before and after registration by GP-ICPR are presented in Figure 7.4. In addition, the estimated transformation parameters and the absolute difference in the parameters in the case of the four different methods: the direct georeferencing method, the Cyclone, GP-ICP, and GP-ICPR, are presented in Appendix D.

Before proceeding further, it must be noted that, as stated in Section 1.2, this dissertation aims to develop an automated registration algorithm for two point clouds, not for multiple point clouds. Therefore, the nine point clouds from the Agia Sanmarina church were registered incrementally, i.e. one by one, although, in principle, the registrations of a set of multiple point clouds need to be performed simultaneously (Bergevin et al., 1996; Rodrigues et al., 2002). Since the point clouds are incrementally registered, the size of individual point cloud pairs is not important so long as every point cloud pair shares at least 5% overlapping region.

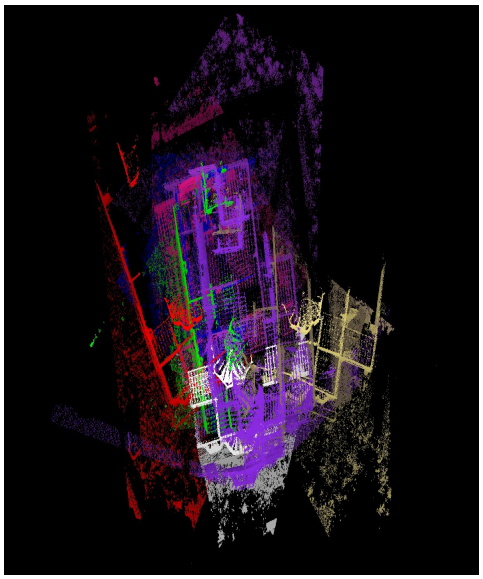
As done with the Ayuthaya Buddha datasets, the colour map of the distance between a point and its local surface shown in Figures 7.5 and 7.6 will be first investigated. No visible difference between three methods: the direct georeferencing method, Cyclone software, and GP-ICP, is observed in the northwest side of the church, i.e. Figure 7.6. However, on the south side of the church, a significant difference in D^{ps} is observed as shown in Figure 7.5. A significant bias toward or away from the south of the church is clearly observed in the case of the direct georeferencing method. In addition, a smaller bias than that obtained by the direct georeferencing method can be observed in the case of Cyclone, although the distance from a point to its local surface in the problem area is still in the order of centimetres. Furthermore, the estimated transformation parameter errors in the Cyclone case introduce an extra bias toward or away from the south of the church. However, in the case of GP-ICP, no visibly large bias is observed as shown in Figures 7.6(e) and 7.6(f). Since the colour maps of D^{ps} in the case of GP-ICPR are not much different from those of GP-ICP, the colour maps of D^{ps} in the case of GP-ICPR are not plotted since only small differences were observed in the colour maps of D^{ps}.

Let us evaluate how much the estimated parameters of the four methods differ. Although the absolute differences in the estimated parameters from different methods are presented in Appendix D, as we are aware, this does not tell us anything important. Therefore, using the Hotelling T^2 method as described in Appendix E, the differences

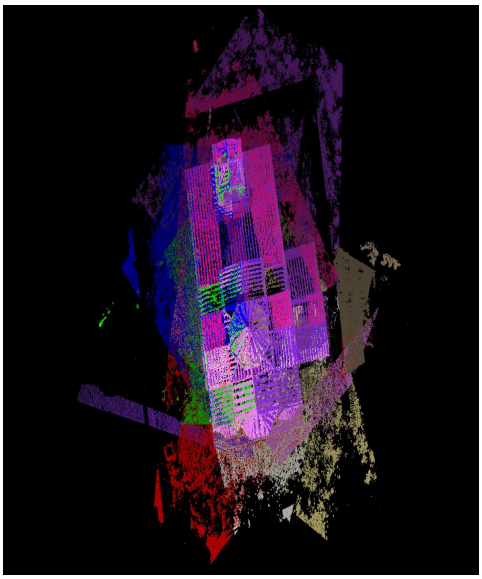


(a) Schematic drawing (top view)

(b) Southeast



(c) Before registration using GP-ICPR



(d) After registration using GP-ICPR

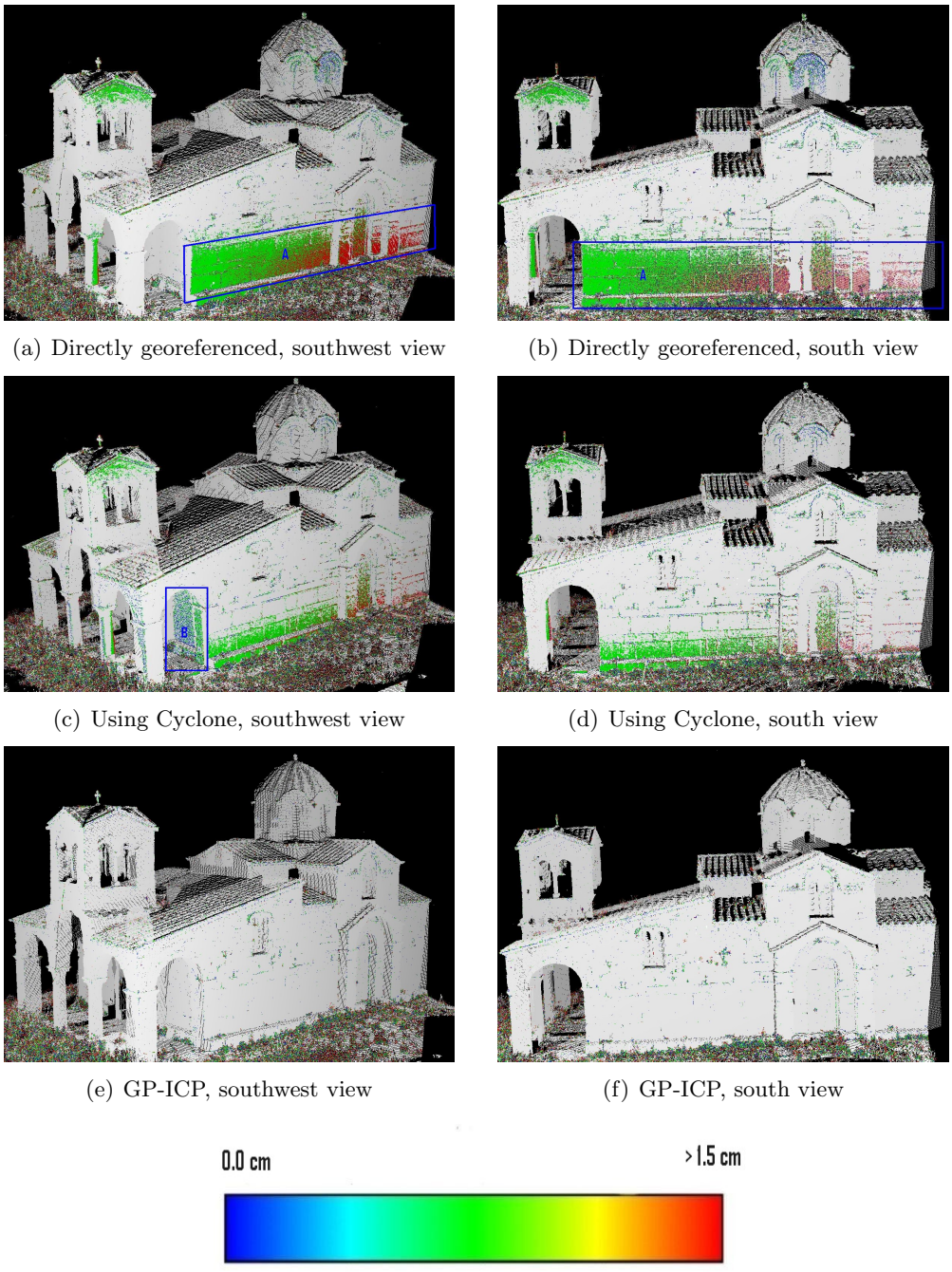
FIGURE 7.4: Pictures of the Agia Sanmarina Church in Greece scanned by a Leica HDS2500. The details of these point clouds can be found in Figure 3.1. (a) is a schematic drawing of the church and (b) is the southeast side of the church. Before and after the registration of the nine point clouds from the church are presented in (c) and (d), respectively.

in the estimated parameters for the four different methods: the direct georeferencing method, Cyclone, GP-ICP, and GP-ICPR, are evaluated with the assumption that the difference in the estimated parameters follows a Gaussian distribution and they are listed in Table 7.3. Using this, these facts are found: the difference between the direct georeferencing and GP-ICP is the largest and that between GP-ICPR and GP-ICP is the smallest. That is why a visual difference in the colour maps of D^{ps} is not observed in the cases of GP-ICPR and GP-ICP.

Now let us investigate the registration errors of the registered point clouds by different methods: the direct georeferencing method, Cyclone, GP-ICP, and GP-ICPR. For the registration of point clouds, Cyclone first roughly registers the point clouds using direct georeferencing information and then refines the registration using its own kind of the ICP. Cyclone only provides its own definition of the registration error, called the mean absolute error, over the entire registered point clouds. We can only guess that the mean absolute error from Cyclone is the average absolute difference between the observed, i.e. the selected points in a point cloud, and the corresponding points adjusted by their algorithm for the registration of the point clouds. Cyclone's mean absolute error is about 1.5cm for the whole point clouds and, on the other hand, GP-ICPR's mean absolute error is about 0.3cm.

In the directly georeferenced case, D^{ps}_{mean} and D^{ps}_{std} are in the order of centimetres, which is why the same order of D^{ps}, i.e. registration error, was observed in Figure 7.5. From Table 7.4, it is also observed that Umeyama's definition of the registration error, i.e. Eq. 4.2, is very optimistic. $D^{Umeyama}$ in the case of the registration of the Greek church is in the order of millimetres, i.e. smaller than half of a pixel. However, it is clearly observed in the highlighted boxes in Figures 7.5(a) and 7.5(b) that the direct georeferencing method provides neither precision nor accuracy as small as half a pixel.

From Table 7.5, the registration error of GP-ICPR, i.e D^{ps}_{mean} or D^{ps}_{std}, is smaller than that of GP-ICP and is larger than the Cramer-Rao lower bound of the registration error of the point clouds. There is about 50% improvement in the closeness to the CRLB of the registration error.



(a) Directly georeferenced, southwest view

(b) Directly georeferenced, south view

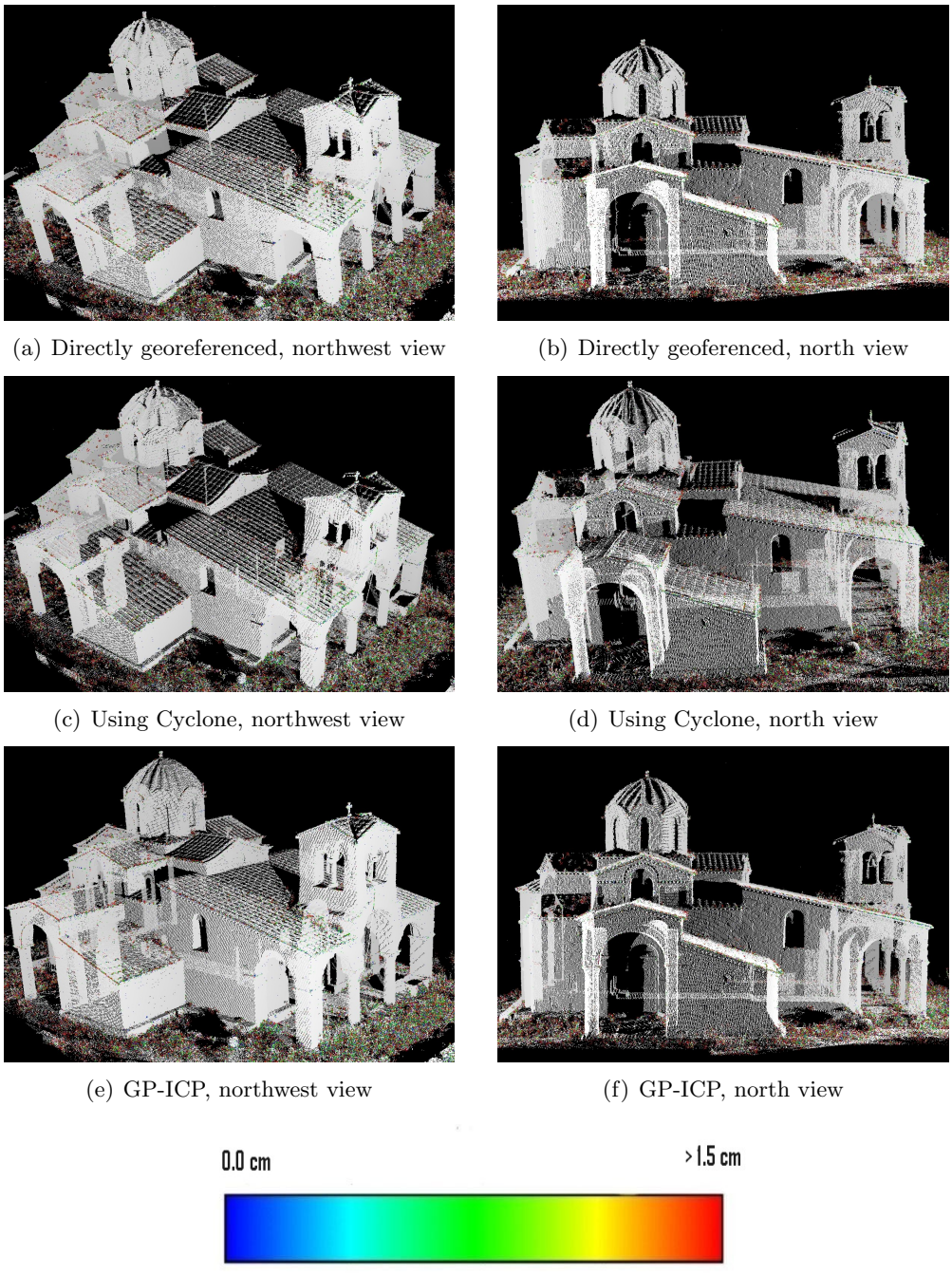
(c) Using Cyclone, southwest view

(d) Using Cyclone, south view

(e) GP-ICP, southwest view

(f) GP-ICP, south view

FIGURE 7.5: Distance from a point to its local surface, D^{ps}, of the registered Agia Sanmarina church data using three methods: the direct georeferencing method, Cyclone, and GP-ICP. These are the southwest and south views of the church. A low change of curvature region has been assigned white and the same threshold value for all three different methods is used.



(a) Directly georeferenced, northwest view

(b) Directly geoferenced, north view

(c) Using Cyclone, northwest view

(d) Using Cyclone, north view

(e) GP-ICP, northwest view

(f) GP-ICP, north view

FIGURE 7.6: Distance from a point to its local surface, D^{ps}, of the registered Agia Sanmarina church data using three methods: the direct georeferencing method, Cyclone, and GP-ICP. These are the northwest and north views of the church. A low change of curvature region has been assigned white and the same threshold value for all three different methods is used.

	$d\omega$ [degree]	$d\phi$ [degree]	$d\kappa$ [degree]
$\delta_{Cyclone\_p}$	$0.27 \times 10^{-3} \pm 0.40$	$-0.98 \times 10^{-2} \pm 0.11$	$0.78 \times 10^{-1} \pm 0.63$
δ_{direct_p}	-0.44 ± 5.45	$-0.84 \times 10^{-2} \pm 0.27$	0.51 ± 5.29
δ_{RANSAC_p}	$-0.27 \times 10^{-2} \pm 0.19$	$0.22 \times 10^{-2} \pm 0.042$	$-0.44 \times 10^{-1} \pm 0.13$

	dx [cm]	dy [cm]	dz [cm]
$\delta_{Cyclone\_p}$	-0.71 ± 3.85	-1.46 ± 9.39	-0.22 ± 3.92
δ_{direct_p}	-1.00 ± 8.31	-1.88 ± 18.24	0.06 ± 3.78
δ_{RANSAC_p}	-0.06 ± 1.30	-0.11 ± 2.6	0.57 ± 1.10

TABLE 7.3: Difference in the estimated parameters by three other methods, i.e. the direct georeferencing method, Cyclone, and GP-ICPR, from GP-ICP using Hotelling T^2 method with 95% confidence level as presented in Appendix E. $\delta_{Cyclone\_p}$ and δ_{direct_p} represent the differences in the estimated parameters by Cyclone and the direct georeferencing information from that of GP-ICP, respectively. In addition, δ_{RANSAC_p} is the difference between the estimated parameters from GP-ICP and GP-ICPR.

	Direct Georeferencing		
registration error	$D^{Umeyama}$ [cm]	D^{pp}_{mean} [cm]	D^{pp}_{std} [cm]
northeast	0.11	1.69	2.92
southeast	0.03	1.38	2.38
south	0.01	0.76	1.34
southwest	0.31	3.42	5.66
west	0.12	1.88	3.23
northwest	0.07	1.86	3.17
north	0.01	0.71	1.25
crane1	0.02	1.32	2.30
crane2	0.03	1.68	2.89
mean	0.08	1.63	2.79

TABLE 7.4: Registration errors of the point clouds from the Agia Sanmarina church using the direct georeferencing method. Umeyama's definition of the minimum registration error, $D^{Umeyama}$, was described in Eq. 4.2.

	GP-ICP		GP-ICPR			
registration error	D^{ps}_{mean} [cm]	D^{ps}_{std} [cm]	ratio of inliers inliers/total	D^{ps}_{mean} [cm]	D^{ps}_{std} [cm]	CRLB [cm]
northeast	0.40	0.51	1059/2000 = 53%	0.010	0.28	0.22
southeast	0.37	0.47	1114/2000 = 56%	0.013	0.27	0.23
south	0.58	1.00	986/1803 = 55%	0.0064	0.25	0.21
southwest	0.42	0.71	1355/2000 = 68%	0.0059	0.26	0.23
west	0.79	1.3	321/672 = 48%	0.024	0.19	0.17
northwest	0.52	0.92	561/849 = 66%	0.0013	0.29	0.23
north	0.58	1.00	517/896 = 58%	0.0035	0.23	0.19
crane1	0.32	0.38	1045/2000 = 52%	0.0078	0.26	0.21
crane2	0.59	0.86	875/1324 = 66%	0.0046	0.26	0.21
mean	0.51	0.79	58%	0.0085	0.25	0.21

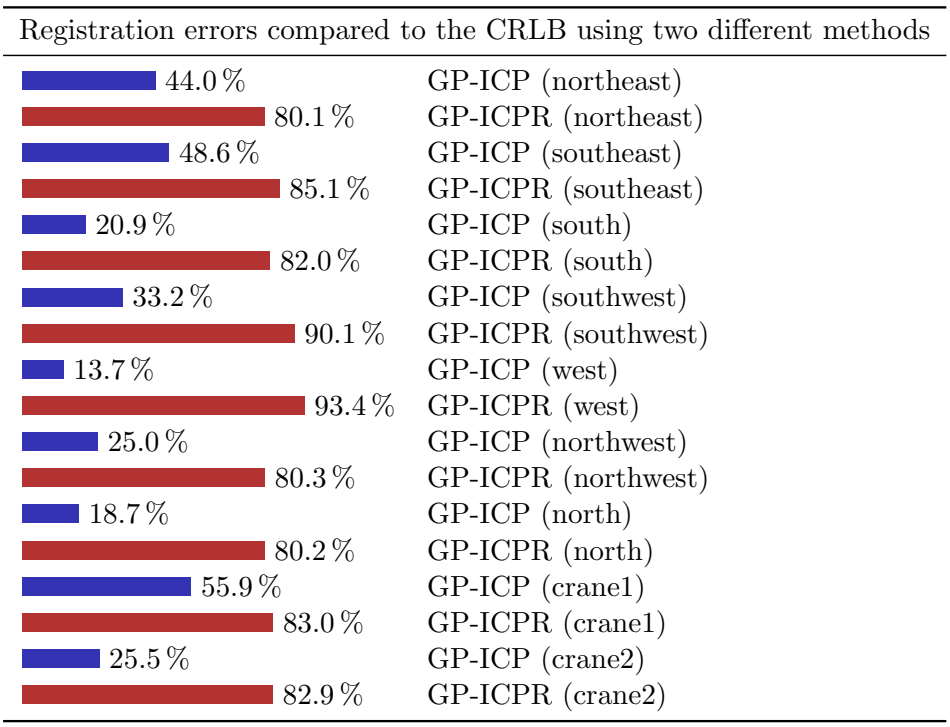


TABLE 7.5: Top table shows the mean and standard deviation of the distance between points and their corresponding surfaces using two methods: GP-ICP and GP-ICPR. It also presents the ratio of inliers and the Cramer-Rao lower bound (CRLB) of the registration errors estimated from Eq. 4.25. Note that neither the ratio of inliers nor the CRLB is available in the case of GP-ICP. The bottom bar graph represents how much the values of D^{ps}_{std} in two different proposed methods, are close to the Cramer-Rao lower bound of the registration error of these point clouds. For example, if it is 100%, then the resultant registration error is as small as it can be, i.e. the CRLB.

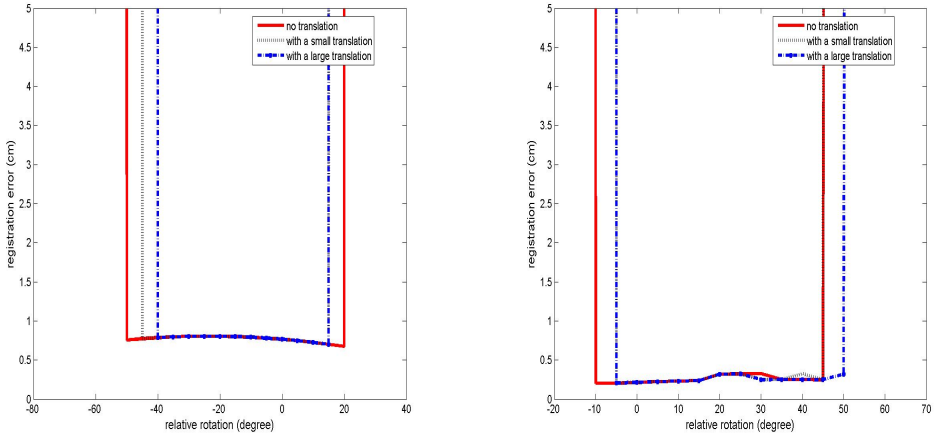
7.2 Convergence region

In this section, the convergence region of GP-ICPR with terrestrial laser scanner data will be evaluated. The scope of the test is the same as that for the close-range laser scanner data described in Section 6.2. A rotational convergence region of a set of point clouds with a registration algorithm is dependent on both its relative translation and $T_{distance}^{iter=0}$. The convergence region of terrestrial laser scanner data is especially dependent on $T_{distance}^{iter=0}$ since a terrestrial laser scanner measures physically larger objects than a close-range laser scanner. In other words, the point clouds from terrestrial laser scanners usually require a much larger $T_{distance}^{iter=0}$ for the same relative rotation angle than close-range scanner datasets. Therefore, it is more difficult to properly find the convergence region of a registration method. Note that $T_{distance}^{iter=0}$ is again changed from a maximum value of a test point cloud to zero with an increment of 5 cm. If a required $T_{distance}^{iter=0}$ is changed greater than the current test $T_{distance}^{iter=0}$, then the required $T_{distance}^{iter=0}$ can never be found unless manually changed, i.e. to set a new and larger maximum value. Therefore, the results in this section must be regarded as the rotational convergence region for terrestrial laser scanners with a fixed maximum $T_{distance}^{iter=0}$. In fact, this has been true for all the convergence region tests in this dissertation.

The results of the convergence test are presented in Figures 7.7 and 7.8. Note that the positive direction of the rotation around the y axis is counter-clockwise around the axis of the test rotation as shown in Figure 7.9. For the case of the registration of the east and the northeast clouds, the rotation convergence region is about $-50^\circ < R^{convergence} < 20^\circ$ with no relative translation. In addition, in presence of the relative translation, it is observed that the rotational convergence region is reduced and shifted to the positive direction of the rotation by an amount in the order of 5°. In the case of the registration of the east and the southeast clouds, about $-10^\circ < R^{convergence} < 45^\circ$ to be the rotational convergence region is observed, with no relative translation. In the presence of relative translations, the shift of the rotational convergence region is also observed, although the absolute size of the rotational convergence region is about the same as in the case of no relative translation.

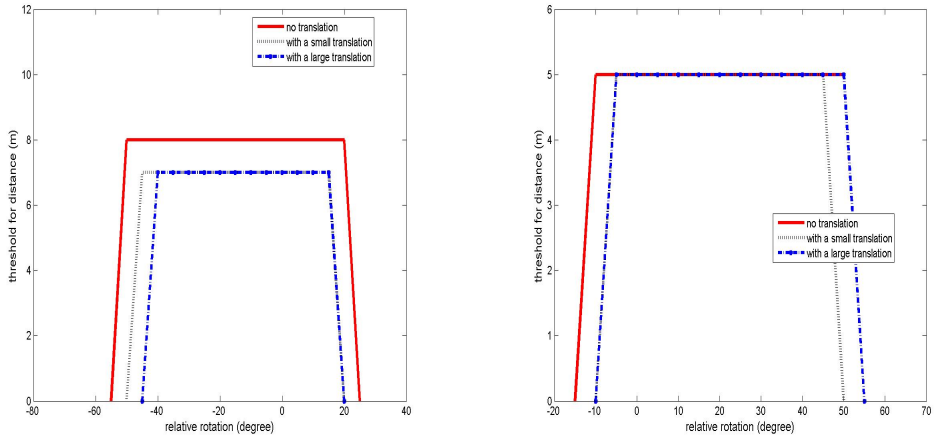
Figure 7.8 shows that a very rough estimation of the required $T_{distance}^{iter=0}$ is only required for the proper registration of a set of point clouds. For example, as shown in Figure 7.8(a), 8m and 7m were used for $T_{distance}^{iter=0}$ over the entire convergence region in cases both with and without relative translation for the registration of the east and the northeast point clouds. Even if the absolute size of the rotation convergence region of the registration of terrestrial laser scanner data is large enough for practical applications, an asymmetry in the rotational convergence region is still observed as seen in Section 6.2 for close-range laser scanner data.

Let us try to explain the cause of this asymmetry in the rotational convergence



(a) East and northeast (b) East and southeast

FIGURE 7.7: Registration errors, D^{ps}, for the convergence region test of GP-ICPR with the Agia Sanmarina church data. Large and small translations are [dx, dy, dz] = [0.5m, 0.5m, 1.0m] and [0.1m, 0.1m, 0.1m], respectively. The northeast of the Agia Sanmarina church was rotated around the y axis which is about the surface normal of the ground.



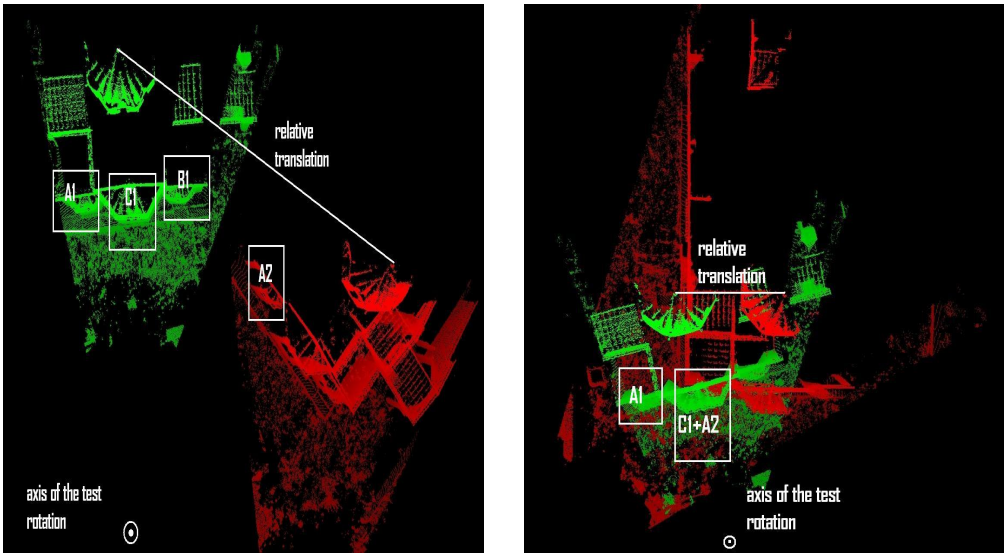
(a) Initial threshold for distance (b) Initial threshold for distance

FIGURE 7.8: Initial threshold for distance, $T^{iter=0}_{distance}$, for the convergence region test of GP-ICPR with the Agia Sanmarina church data. A large and small translations are [dx, dy, dz] = [0.5m, 0.5m, 1.0m] and [0.1m, 0.1m, 0.1m], respectively. The northeast of the Agia Sanmarina church was rotated around the y axis which is approximately the surface normal of the ground.

region. The top and side views of the parts of the Agia Sanmarina church data are presented in Figure 7.9. Note that region A1 in the green is the corresponding region of A2 in the red. In Figures 7.9(a) and 7.9(c), the northeast of the church is rotated by -50° around the y axis, i.e. the axis of the test rotation, and also translated [dx, dy, dz] = [0.5m, 0.5m, 1.0m], i.e. the large translation in Figure 7.7. This transformation is one of the limits of the convergence region as shown in Figure 7.7(a). It is observed that region B1 of the church is the closer region to region A2 than its true corresponding region, i.e. A1. In addition, the geometric shape of region B1 is very similar with region A1 and furthermore to reach region A1, we need to go through region C1, which has a similar shape but a different scale. In the case of the east and the southeast clouds presented in Figures 7.9(b) and 7.9(d), the latter is rotated around y axis by -15° and again translated by [0.5m, 0.5m, 1.0m]. Note that this transformation is also a limit of the convergence region between the east and the southeast of the church as shown in Figure 7.7(b). In this case, a smaller rotation convergence limit is achieved with the same translation because region A2 is much closer to region C1 than A1. Therefore GP-ICPR is likely to find a possible corresponding set of region A2 in region C1 rather than A1.

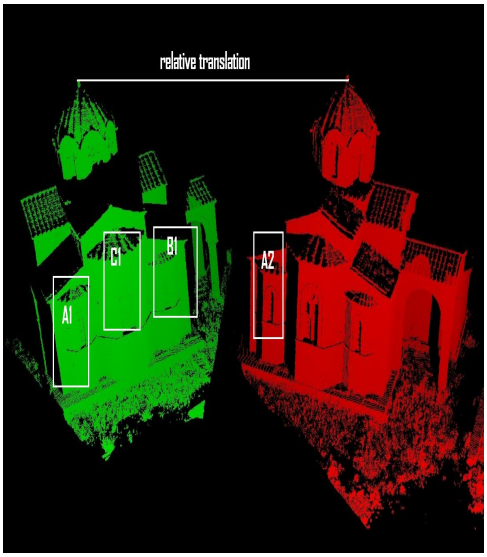
7.3 Summary

The precision of GP-ICPR was found to be better than that of GP-ICP and Cyclone as shown in Tables 7.4 and 7.5. Registration accuracy was analytically evaluated using the colour maps of the change of curvature from the registered point clouds by four different methods, i.e. the direct georeferencing method, a commercial software: Cyclone, GP-ICP and GP-ICPR. It was found that the results of GP-ICP and GP-ICPR were much closer to the truth than those of the other methods. Furthermore, the differences in the estimated relative transformation parameters from the four registration methods were evaluated using the Hotelling T^2 method. It was also shown that the results of GP-ICP and GP-ICPR are much closer to each other than the results from the other methods. In addition, the rotational convergence region of GP-ICPR was also in the order of 10° as shown in Figure 7.7. Therefore, it is concluded that relative transformation parameters estimated by GP-ICPR and GP-ICP are more precise and accurate than those by both the direct georeferencing method and Cyclone software.

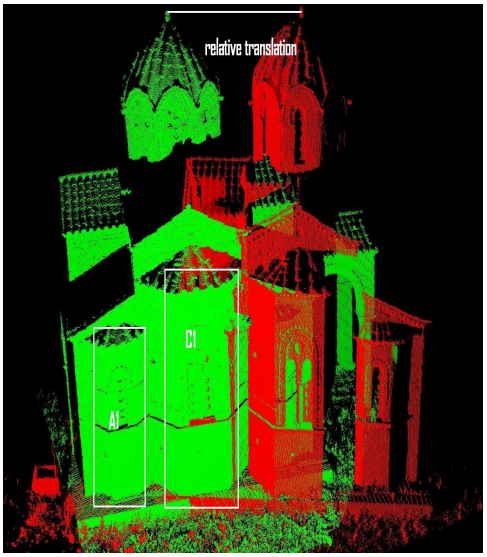


(a) East and northeast, top view

(b) East and southeast, top view



(c) East and northeast, side view



(d) East and southeast, side view

FIGURE 7.9: Top and side views of the Agia Sanmarina church data for the convergence region test in the case of the large translation, i.e. [dx, dy, dz] = [0.5m, 0.5m, 1.0m]. Green is the east of the church. The red point clouds in (a) and (b) are the northeast and the southeast of the church, respectively.

Chapter 8

Conclusions and future directions

This dissertation has presented two automated methods for the registration of two partially overlapping point clouds, named the Geometric Primitive Iterative Closest Point (GP-ICP) and the Geometric Primitive ICP with the RANSAC (GP-ICPR), respectively. These methods use geometric primitives such as surface normal vectors and the change of geometric curvature, neighbourhood search, the position uncertainty of a laser scanner, and the RANSAC (Random Sample Consensus) procedure.

8.1 Summary of the thesis

As summarised by Chen et al. (1999) and stated in Section 1.2, the limitations of the existing automated registration methods, i.e. the ICP and its variants, can be listed as follows: inability to produce an evaluation of the accuracy of the estimated transformation, necessity of a good initial alignment, and requirement of sufficient local features, i.e. geometric primitives. This thesis produced solutions to overcome these limitations. First, a method for evaluating the accuracy of the estimated relative transformation was presented in Chapter 3 and Sections 4.4 and 4.6. Second, the GP-ICP and the GP-ICPR minimised the necessity of a good initial alignment as shown in Sections 5.3, 6.2 and 7.2. Third, the GP-ICP and GP-ICPR were found to provide a solution for the requirement of sufficient local features of the ICP and its variants by testing with several datasets which had different amounts of geometrically distinctive features (Chapters 5 - 7).

In addition, the GP-ICP and the GP-ICPR provided a larger rotational convergence region, in the order of 10°, for real data using the geometric primitives estimated by a point and its neighbourhood. As stated in Section 1.2, this rotational convergence region is large enough for practical applications. Both the position uncertainty and a lower bound of variance in the measurement by a laser scanner were explicitly expressed in Chapter 3 and Section 4.4, respectively. Furthermore, this position uncertainty of

measurement by laser scanners was utilised in order to improve the precision of the registration error and also to provide an evaluation of the accuracy of the estimated transformation parameters.

As shown with simulated datasets in Table 5.10, the GP-ICPR was found to improve the precision of the GP-ICP by as much as a factor of 5 in the presence of the noise described in Chapter 3. Since the precision of GP-ICP is similar to that of the ICP or its variants, it can be claimed that the precision of the GP-ICPR is better than that of either the ICP or its variants.

In Chapter 6, the precision of the registration error of a set of close-range scanner data from the Stanford 3D scanning repository (Stanford Computer Graphic Laboratory, 2006) was evaluated and parameters estimated by the GP-ICP were directly compared with those from the Stanford method, i.e. Turk and Levoy's (1994) method. The colour maps of the change of curvature defined in Eq. 4.22 were utilised as a method for evaluating of the accuracy of registrations by the GP-ICP and the Stanford method. In addition, the rotational convergence region of the GP-ICP was in the order of 10° with close-range scanner data as shown in Figure 6.8.

In cases of terrestrial laser scanner data, the precision of registration error improved by the GP-ICPR by up to a factor of 5 compared with the GP-ICP, the ICP, and its variants as shown in Table 7.5. It was also found that the results of the GP-ICP and the GP-ICPR were much closer to the truth than those of the other methods such as direct georeferencing and a commercial software Cyclone (Version 5.0), using the colour maps of the change of curvature from the registered point clouds and the proposed statistical test presented in Section 4.6. Furthermore, the differences in the transformation parameters estimated from the four different registration methods were evaluated using the Hotelling T^2 method described in Appendix E with the assumption that the difference in the estimated parameters follows a Gaussian distribution. It was shown that the results of the GP-ICP and GP-ICPR were much closer to each other than those of the other methods. With the terrestrial scanner datasets, the rotational convergence region of the GP-ICPR was in the order of 10° as shown in Figure 7.7.

8.2 Conclusion

The GP-ICP and the GP-ICPR methods provide both a large rotational convergence region and an acceptable size of translational convergence region as shown from the extensive tests with simulated data and the datasets from close-range and terrestrial laser scanners. In addition, an explicit expression of the position uncertainty in the measurement by laser scanners provides a lower bound for the registration error between point clouds and produces a criterion to remove outliers for the RANSAC procedure. Furthermore, the lower bound of the registration error estimated by the position un-

certainty of measurement by laser scanners provides an evaluation of the accuracy of the estimated parameters, which is an important criterion to terminate the GP-ICPR with the assurance that the solution is close to the truth.

8.3 Future directions

Although the GP-ICP and the GP-ICPR provided a much larger rotation convergence region than the existing registration methods, they did not produce a large translational convergence region, e.g. in the order of a metre. A method of finding correspondence in a hyper-space such as the extended Gaussian image method (Horn, 1984; Dold, 2005) rather than the spatial domain can increase the size of the convergence region by providing a good initial alignment for point clouds.

Although, as stated in Section 1.2, an evaluation of the efficiency of the GP-ICP and the GP-ICPR is not one of the objectives in this dissertation, a fast nearest neighbourhood search algorithm could improve the execution time of the GP-ICP, the GP-ICPR, or any other ICP variant. Therefore, development of such an algorithm is an important further research area, especially for terrestrial laser scanner data. The execution time of a registration algorithm could also be improved by introducing a fast method for resampling as well. Although a method for resampling was developed in this dissertation using the position uncertainty of Chapter 3 in conjunction with a clustering method such as Pauly et al. (2002), there is room for improvement, such as developing a method for selecting a threshold value for the size or number of points of allowed clusters.

An explicit expression of the position uncertainty, which was utilised both to improve precision and to provide an evaluation of accuracy, can be used for other applications such as error analysis for laser scanners for the measurement of spherical or cylindrical objects. For example, this position uncertainty can produce a validation method for an estimation process such as determining the centre of spherical targets or the radius of cylindrical objects.

References

Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Croz, J. D., Greenbaum, A., Hammarling, S., McKenney, A. and Sorensen, D., 1999. LAPACK (Linear Algebra PACKage) Users' Guide, third edn, Society for Industrial and Applied Mathematics, ISBN 0898714478, 407 pages.

Arfken, G. B., 1995. Mathematical methods for physicists, fourth edn, Academic Press Inc., ISBN 0120598264, 1029 pages.

Arun, K. S., Huang, T. S. and Blostein, S. D., 1987. Least-squares fitting of two 3-D point sets. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 9(5), pp. 698–700.

Arya, S., Mount, D., Netanyahu, N. S., Silverman, R. and Wu, A. Y., 1998. An optimal algorithm for approximate nearest neighbour searching. Journal of the ACM (Association for Computing Machinery) 45, pp. 891–923.

Atkinson, K. B. (ed.), 1996. Close Range Photogrammetry and Machine Vision, Engineering and Science, Whittles Publishing, Scotland, UK, ISBN 1870325737, 371 pages.

Bae, K.-H. and Lichti, D. D., 2004. Automated registration of unorganised point clouds from terrestrial laser scanners. International Archives of Photogrammetry and Remote Sensing (IAPRS) 35(part B5), pp. 222–227.

Bae, K.-H., Belton, D. and Lichti, D. D., 2005. A framework for position uncertainty of unorganised three-dimensional point clouds from near-monostatic laser scanners using covariance analysis. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences (IAPRS) 36(part 3/W19), pp. 7–12.

Batalama, S. N. and Kazakos, D., 1997. On the generalized Cramer-Rao bound for the estimation of the location. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Signal Processing 45(2), pp. 487–492.

Bergevin, R., Soucy, M., Gagnon, H. and Laurendeau, D., 1996. Towards a general multi-view registration technique. Institute of Electrical and Electronics Engineers (IEEE) Transaction on Pattern Analysis and Machine Intelligence 18(5), pp. 540–547.

Berkmann, J. and Caelli, T., 1994. Computation of surface geometry and segmentation using covariance techniques. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 16(11), pp. 1114–1116.

Besl, P. J. and McKay, N. D., 1992. A method for registration of 3-D shapes. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 14(2), pp. 239–256.

Blais, F., Beraldin, J. A. and El-Hakim, S., 2000. Range error analysis of an integrated time-of-flight, triangulation, and photogrammetric 3D laser scanning system. Proceedings of The International Society for Optical Engineering (SPIE)'s Aerosense, Orlando, Florida, USA 4035, pp. 236–247.

Blais, G. and Levine, M. D., 1995. Registering multiview range data to create 3D computer objects. Institute of Electrical and Electronics Engineers (IEEE) Transaction on Pattern Analysis and Machine Intelligence 17(8), pp. 820–824.

Bretar, F. and Roux, M., 1999. Hybrid image segmentation using LiDAR 3D planar primitives. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences (IAPRS) 36(part 3/W19), pp. 72–78.

Campbell, R. J. and Flynn, P. J., 2001. A survey of free-form object representation and recognition techniques. Computer Vision and Image Understanding 81, pp. 166–210.

Chen, C.-S., Hung, Y.-P. and Cheng, J.-B., 1999. Ransac-based DARCES: a new approach to fast automatic registration of partially overlapping range images. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 21(11), pp. 1229–1234.

Chen, Y. and Medioni, G., 1992. Object modelling by registration of multiple range. Image and Vision Computing 10(3), pp. 145–155.

Crosilla, F. and Beinat, A., 2002. Use of generalised Procrustes analysis for the photogrammetric block adjustment by independent models. The International Society for Photogrammetry and Remote Sensing (ISPRS) Journal of Photogrammetry and Remote Sensing 56, pp. 195–209.

Curless, B. and Levoy, M., 1996. A volumetric method for building complex models from range images. Proceedings of Association for Computing Machinery (ACM) Special

Interest Group on Computer Graphics and Interactive Techniques (SIGGRAPH) '96, pp. 303–312.

Cyberware, 2006. http://www.cyberware.com/, accessed on 16th April 2006.

Devara, P. C. S., 1998. Remote sensing of atmospheric aerosols from active and passive optical techniques. International Journal of Remote Sensing 19(17), pp. 3271–3288.

Dold, C., 2005. Extended gaussian images for the registration of terrestrial scan data. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences (IAPRS) 36(part 3/W19), pp. 180–185.

Dorai, C., Wang, G., Jain, A. K. and Mercer, C., 1998. Registration and integration of multiple object views for 3D model construction. Institute of Electrical and Electronics Engineers (IEEE) Transaction on Pattern Analysis and Machine Intelligence 20(1), pp. 83–89.

Dorai, C., Weng, J. and Jain, A. K., 1997. Optimal registration of object views using range data. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 19(10), pp. 1131–1138.

Dyn, N., Hormann, K., Kim, S.-J. and Levin, D., 2001. Optimizing 3D triangulations using discrete curvature analysis. Proceedings of Mathematical methods for curves and surfaces (Oslo '2000), pp. 135–146.

Eggert, D. W., Fitzgibbon, A. W. and Fisher, R. B., 1998. Simultaneous registration of multiple range views for use in reverse engineering of CAD models. Computer vision and image understanding 69(3), pp. 253–272.

Eggert, D. W., Lorusso, A. and Fisher, R. B., 1997. Estimating 3-D rigid body transformations: a comparison of four major algorithms. Machine Vision and Applications 9, pp. 272–290.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the Association for Computing Machinery (ACM) 24(6), pp. 381–395.

Fitzgibbon, A., 2003. Rubust registration of 2D and 3D point sets. Image and Vision Computing 21(13-14), pp. 1145–1153.

Förstner, W., 2004. Projective Geometry for Photogrammetric Orientation Procedures, Tutorial in International Society for Photogrammetry and Remote Sensing (ISPRS) Congress, Istanbul, 99 pages.

Garland, M. and Heckbert, P. S., 1997. Surface simplification using quadric error metrics. Proceedings of Association for Computing Machinery (ACM) Special Interest Group on Computer Graphics and Interactive Techniques (SIGGRAPH), pp. 209–216.

Golub, G. H. and Loan, C. F. V., 1989. Matrix computations, second edn, Johns Hopkins University Press, ISBN 0801854148, 664 pages.

Gordon, S. J., 2005. Structural Deformation Measurement using Terrestrial laser scanners. PhD thesis, Department of Spatial Sciences, Curtin University of Technology, 229 pages.

Gruen, A. and Akca, D., 2004. Least squares 3D surface matching. International Society for Photogrammetry and Remote Sensing (ISPRS) Panoramic Photogrammetry Workshop, Dresden, Germany (on CD-ROM).

Gruen, A. and Akca, D., 2005. Least squares 3D surface and curve matching. International Society for Photogrammetry and Remote Sensing (ISPRS) Journal of Photogrammetry and Remote Sensing 59(3), pp. 151–174.

Haralick, R. M. and Shapiro, L. G., 1993. Compute and robot vision: vol. 2, Addison-Wesley Publishing Company, ISBN 0201569434, 630 pages.

Haralick, R. M., Joo, H., Lee, C. H., Zhang, X., Vaidya, V. G. and Kim, M. B., 1989. Pose estimation from corresponding point data. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Systems, Man, and Cybernetics 19(6), pp. 1426–1446.

Hartley, R. and Zisserman, A., 2001. Multiple view geometry in computer vision, Cambridge University Press, ISBN 0521540518, 672 pages.

Higuchi, K., Herbert, M. and Ikeuchi, K., 1995. Building 3-D models from unregistered range images. Graphical models and image processing 57(4), pp. 315–333.

Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W., 1992. Surface reconstruction from unorganized points. Computer Graphics 26, pp. 71–78.

Horn, B. K. P., 1984. Extended gaussian images. Proceeding of the Institute of Electrical and Electronics Engineers (IEEE) 72(12), pp. 1671–1684.

Horn, B. K. P., 1987. Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America 4(4), pp. 629–642.

Huber, D. F. and Hebert, M., 2003. Fully automatic registration of multiple 3D data. Image and Vision Computing 21(7), pp. 637–650.

Johnson, A. and Kang, S. B., 1997. Registration and integration of textured 3-D data. International Conference on Recent Advances in 3-D Digital Imaging and Modeling (3DIM), pp. 234 – 241.

Johnson, R. A. and Wichern, D. W., 1998. Applied multivariate statistical analysis, fourth edn, Prentice-Hall Inc., ISBN 0130800848, 816 pages.

Jost, T. and Hügli, H., 2002. Fast ICP algorithms for shape registration. In: L. J. V. Gool (ed.), The German Association for Pattern Recognition (der Deutschen Arbeitsgemeinschaft für Mustererkennung, DAGM) Symposium, Lecture Notes in Computer Science, Vol. 2449, Springer, pp. 91–99.

Kanatani, K., 1996. Statistical optimization for geometric computation: theory and practice, first edn, Elsevier Science, ISBN 0486443086, 509 pages.

Kanazawa, Y. and Kanatani, K., 1996. Optimal line fitting and reliability evaluation. The Institute of Electronics, Information and Communication Engineers (IEICE) Transactions on Information and Systems E79-D(9), pp. 1317–1322.

Kay, S. M., 1993. Fundamentals of Statistical Signal Processing : Estimation Theory, first edn, Prentice Hall, ISBN 0133457117, 625 pages.

Koch, K.-R., 1999. Parameter estimation and hypothesis testing in linear models, second edn, Springer, ISBN 3540652574, 332 pages.

Leica, 2006. http://hds.leica-geosystems.com/, accessed on 16th April 2006.

Lichti, D. D., Gordon, S. and Tipdecho, T., 2005. Error models and propagation in directly georeferenced terrestrial laser scanner networks. American Society of Civil Engineering (ASCE) Journal of Survey Engineering 31(4), pp. 135–142.

Linsen, L., 2001. Point cloud representation. Technical Report, Faculty of Computer Science, University of Karlsruhe.

Maas, H. G., 2000. Least-sqaures matching with airborne laser scanning data in a TIN structure. The International Society for Photogrammetry and Remote Sensing (ISPRS) International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 33(3A), pp. 548–555.

Maas, H.-G. and Vosselman, G., 1999. Two algorithms for extracting building models from raw laser altimetry data. International Society for Photogrammetry and Remote Sensing (ISPRS) Journal of Photogrammetry and Remote Sensing 54, pp. 153–163.

Mardia, K., Kent, J. and Bibby, J., 1979. Multivariate Analysis, first edn, Academic Press, ISBN 0124712525, 518 pages.

Masuda, T. and Yokoya, N., 1995. A robust method for registration and segmentation of multiple range images. Computer vision and image understanding 61(3), pp. 295–307.

Masuda, T., Sakaue, K. and Yokoya, N., 1996. Registration and integration of multiple range image for 3-D model construction. Proceedings of the 13th International Conference on Pattern Recognition 1, pp. 879–883.

McGlone, C., Mikhail, E. and Bethel, J., 2004. Manual of Photogrammetry, fifth edn, ASPRS (American Society of Photogrammetry and Remote Sensing), ISBN 1570830711, 1151 pages.

Mensi, 2006. http://www.trimble.com/3dlaserscanners.shtml, accessed on 16th April 2006.

Mitra, M. J., Nguyen, A. and Guibas, L., 2004. Estimating surface normals in noisy point cloud data. International Journal of Computational Geometry and Applications 14(4,5), pp. 261–276.

Neugebauer, P. J., 1997. Geometrical cloning of 3D objects via simultaneous registration of multiple range images. Proceedings of the 1997 International Conference on Shape Modeling and Applications (SMA '97), pp. 130–139.

Pauly, M., Gross, M. and Kobbelt, L., 2002. Efficient simplification of point-sampled surfaces. Proceedings of Institute of Electrical and Electronics Engineers (IEEE) Visualization 2002, pp. 163 – 170.

Pauly, M., Mitra, N. J. and Guibas, L. J., 2004. Uncertainty and variability in point cloud surface data. Proceedings of Eurographics Symposium on Point-based Graphics, http://citeseer.ist.psu.edu/pauly04uncertainty.html, accessed on 16th April 2006.

Pennec, X. and Thirion, J., 1997. A framework for uncertainty and validation of 3D registration methods based on points and frames. International Journal of Computer Vision 25(3), pp. 203–229.

Pfeifer, N., 2005. A subdivision algorithm for smooth 3D terrain models. International Society for Photogrammetry and Remote Sensing (ISPRS) Journal of Photogrammetry and Remote Sensing 59(3), pp. 115–127.

Pulli, K., 1999. Multiview registration for large data sets. Proceedings of 3-D Digital Imaging and Modelling (3DIM), pp. 160–168.

Rao, C. R., 1973. Linear statistical inference and its applications, second edn, John Wiley and Sons, ISBN 0471218758, 625 pages.

Riegl, 2006. http://www.riegl.com/, accessed on 16th April 2006.

Rodrigues, M., Fisher, R. and Liu, Y., 2002. Special issue on registration and fusion of range images. Computer Vision and Image Understanding 87, pp. 1–7.

Rusinkiewicz, S. and Levoy, M., 2001. Efficient variant of the ICP algorithm. Proceedings of 3-D Digital Imaging and Modelling (3DIM), pp. 145–152.

Samet, H., 1989. The design and analysis of spatial data structures, Addison-Wesley Publishing Company, ISBN 0201502550, 493 pages.

Samet, H., 1990. Application of spatial data structures, computer graphics, image processing, and GIS, Addison-Wesley Publishing Company, ISBN 020150300X, 507 pages.

Sedgewick, R., 1988. Algorithms, second edn, Addison-Wesley Publishing Company, ISBN 0201066734, 657 pages.

Sharkarji, C. M., 1998. Least-squares fitting algorithms of the NIST algorithm testing system. Journal of research of the National Institute of Standard and Technology 103(6), pp. 633–641.

Sharp, G. C., Lee, S. W. and Wehe, D. K., 2002. ICP registration using invariant features. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 24(1), pp. 90–102.

Simon, D., 1996. Fast and Accurate Shape-Based Registration. PhD thesis, Robotics Institute, Carnegie Mellon University, 196 pages.

Stamos, I. and Leordeanu, M., 2003. Automated feature-based range registration of urban scenes of large scale. Institute of Electrical and Electronics Engineers (IEEE) Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) II, pp. 555–561.

Stanford Computer Graphic Laboratory, 2006. The Stanford 3D Scanning Repository. http://graphics.stanford.edu/data/3Dscanrep/, accessed on 16th April 2006.

Stein, F. and Medioni, G., 1992. Structural indexing: efficient 3-D object recognition. Institute of Electrical and Electronics Engineers (IEEE) Transaction on Pattern Analysis and Machine Intelligence 14(21), pp. 125–145.

Stoddart, A. J., Lemke, S., Hilton, A. and Renn, T., 1996. Estimating pose uncertainty for surface registration. Proceedings of British Machine Vision Association (BMVC), http://www.bmva.ac.uk/bmvc/1996/, accessed on 16th April 2006.

Tasdizen, T. and Whitaker, R., 2003. Cramer-Rao bounds for nonparametric surface reconstruction from range data. Proceedings of 3-D Digital Imaging and Modelling (3DIM), pp. 70–77.

Taubin, G., 1991. Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 13(11), pp. 1115–1138.

Turk, G. and Levoy, M., 1994. Zippered polygon meshes from range images. Proceedings of Association for Computing Machinery (ACM) Special Interest Group on Computer Graphics and Interactive Techniques (SIGGRAPH '94), pp. 311–318.

Umeyama, S., 1991. Least squares estimation of transformation parameters between two patterns. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 13(4), pp. 376–380.

Weik, S., 1997. Registration of 3-D partial surface models using luminance and depth information. Proceedings of 3-D Digital Imaging and Modelling (3DIM), pp. 93–100.

Whaite, P. and Ferrie, F. P., 1991. From uncertainty to visual exploration. Institute of Electrical and Electronics Engineers (IEEE) Transactions on Pattern Analysis and Machine Intelligence 13(10), pp. 1038 – 1049.

Wolf, P. R. and Dewitt, B., 1999. Elements of Photogrammetry - with Applications in GIS, third edn, McGraw-Hill, ISBN 0072924543, 624 pages.

Zhang, Z., 1994. Iterative point matching for registration of free-form curves and surfaces. International Journal of Computer Vision 13(2), pp. 119–152.

Appendix A

Representation of rotation

In this section, a matrix representation of rotation is introduced for reference. One can find further discussion in Haralick and Shapiro (1993, p. 127), Atkinson (1996) or Wolf and Dewitt (1999). The Euler representation of rotation is used throughout the thesis, although there are many alternative representations such as quaternions or Rodrigues notation. Let ω, ϕ, and κ be the rotation angles around x, y, and z axes, respectively. Then the rotation matrices around the axes, $\mathbf{R}(\cdot)$, are defined as follows:

$$\mathbf{R}(\omega) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{pmatrix}$$

$$\mathbf{R}(\phi) = \begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix}$$

and

$$\mathbf{R}(\kappa) = \begin{pmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

In a similar manner, a general rotation matrix in three dimensional space, $\mathbf{R}(\omega, \phi, \kappa)$, is defined

$$\mathbf{R}(\omega, \phi, \kappa) = \mathbf{R}(\kappa)\mathbf{R}(\phi)\mathbf{R}(\omega) =$$
$$\begin{pmatrix} \cos\phi\cos\kappa & \sin\omega\sin\phi\cos\kappa + \cos\omega\sin\kappa & -\cos\omega\sin\phi\cos\kappa + \sin\omega\sin\kappa \\ -\cos\phi\sin\kappa & -\sin\omega\sin\phi\sin\kappa + \cos\omega\cos\kappa & \cos\omega\sin\phi\cos\kappa + \sin\omega\cos\kappa \\ \sin\phi & -\sin\omega\cos\phi & \cos\omega\cos\phi \end{pmatrix}.$$

Appendix B

Cramer-Rao lower bound for a scalar parameter

In this section, the Cramer-Rao lower bound (CRLB) for an unbiased scalar estimator is revisited. The Cramer-Rao lower bound is a well established fact in statistics or signal processing. However, it not well informed in the research related with point clouds from range sensors. One may refer to Rao (1973, Ch. 5), Mardia et al. (1979, Ch. 4), Kay (1993, Ch. 3), Kanatani (1996, Ch. 3) and Batalama and Kazakos (1997) for further discussion. In addition, as more discussion on some technical terms, e.g. the Fisher information, can be obtained from the references listed above, no further explanation of these technical terms related with the Cramer-Rao lower bound is included here. Note that the sole reason of proving the Cramer-Rao lower bound herein is to throughly understand the meaning of Eq. B.13 which is the CRLB for a scalar parameter and the key equation of the evaluation of the CRLB of measurement by laser scanners.

Suppose there is a set of data, $X = x_1 \dots x_n$ parameterised by θ:

$$x_i = g(\theta) + \epsilon_i, \tag{B.1}$$

where $\epsilon_i \sim P(x_{i=1\dots n}; \theta)$, $g(\theta)$ is known as an estimator function which depends on the problem in question and let $P(x_{i=1\dots n}; \theta)$ be the probability density function of X. Simply speaking, ϵ_i is noise of datum x_i and $g(\theta)$ is a kind of function to fit the data, e.g. a first order polynomial or sine function. We would like to find the estimator function of X which provides an unbiased estimate of the parameter, $\hat{\theta}$, as

$$\hat{\theta} = g(X = \{x_{i=1} \dots x_n\}).$$

If the independence of the probability density functions is assumed, then the likelihood

function of the data is

$$L(X;\theta) = \prod_{i=1}^{n} P(x_i;\theta)$$

and the log-likelihood function is

$$l(X;\theta) = \ln L(X;\theta) = \sum_{i=1}^{n} \ln P(x_i;\theta).$$

From these likelihood functions, the score function is defined as

$$s(X;\theta) = \frac{\partial}{\partial\theta} l(X;\theta) = \frac{\partial}{\partial\theta} \sum_{i=1}^{n} \ln P(x_i;\theta) = \sum_{i=1}^{n} \frac{\partial}{\partial\theta} \ln P(x_i;\theta). \tag{B.2}$$

In addition, the Fisher information, $\mathbb{F}(\theta)$, is defined as

$$\mathbb{F}(\theta) = V(s) = E(s^2) = \sum_{i=1}^{n} E\left[\left(\frac{\partial}{\partial\theta} \ln P(x_i;\theta)\right)^2\right] \tag{B.3}$$

$$= \sum_{i=1}^{n} \int \left(\frac{\partial}{\partial\theta} \ln P(x_i;\theta)\right)^2 P(x_i;\theta)\, dx_i, \tag{B.4}$$

where $V(\cdot)$ and $E(\cdot)$ are the variance and expectation operators with respect to $P(x_i;\theta)$, respectively.

Theorem B.1. *For the case of Eq. B.1, it is assumed that $P(x_i;\,\theta)$ satisfies the regularity condition as follow*

$$E\left[s(X;\theta)\right] = \sum_{i=1}^{n} E\left[\frac{\partial}{\partial\theta} \ln P(x_i;\theta)\right] = 0 \tag{B.5}$$

for all θ. Then the variance of any unbiased estimator, $\hat{\theta}$, must satisfy

$$V(\hat{\theta}) \geq \frac{\left(\frac{\partial}{\partial} g(\theta)\right)^2}{-\sum_{i=1}^{n} E\left[\frac{\partial^2}{\partial\theta^2} \ln P(x_i;\theta)\right]}. \tag{B.6}$$

Furthermore, an unbiased estimator, which has the lower bound for all θ, can be found if and only if

$$\sum_{i=1}^{n} \frac{\partial}{\partial\theta} \ln P(x_i;\theta) = \left(\frac{\partial g(\theta)}{\partial\theta}\right)^{-1} \mathbb{F}(\theta)\left(\hat{\theta} - g(\theta)\right). \tag{B.7}$$

Proof. First we examine the regularity condition

$$E\left[s(X;\theta)\right] = \sum_{i=1}^{n} E\left[\frac{\partial}{\partial\theta}\ln P(x_i;\theta)\right] = \sum_{i=1}^{n}\int\left[\frac{\partial}{\partial\theta}\ln P(x_i;\theta)\right]P(x_i;\theta)dx_i$$
$$= \sum_{i=1}^{n}\int\left[\frac{\partial}{\partial\theta}P(x_i;\theta)\right]dx_i$$

If the order of differentiation and integration may be interchangeable, i.e. the derivative of the p.d.f. is well defined, then the above equation satisfies

$$\sum_{i=1}^{n}\int\left[\frac{\partial}{\partial\theta}P(x_i;\theta)\right]dx_i = \sum_{i=1}^{n}\frac{\partial}{\partial\theta}\int P(x_i;\theta)dx_i = 0$$

since the integral of $P(x_i;\theta)$ is one, i.e. a constant. This condition is well satisfied with an exponential type p.d.f., e.g. a Gaussian (Kay, 1993). This is not true if the p.d.f. is a uniform, e.g. a boxcar, function.

In subsequent equation the summation symbol will be dropped. For example, $P(x_i;\theta)$ is $\sum_{min(i)}^{max(i)} P(x_i;\theta)$. An unbiased estimator, $\hat{\theta}$, must satisfy that

$$E(\hat{\theta}) = \int\hat{\theta}P(x_i;\theta)dx_i = g(\theta) \tag{B.8}$$

Its partial differentiation with respect to θ is given as

$$\int\hat{\theta}\frac{\partial}{\partial\theta}P(x_i;\theta)dx_i = \frac{\partial g(\theta)}{\partial\theta},$$

and, using the regularity condition, we can obtain

$$\int(\hat{\theta}-g(\theta))\left[\frac{\partial}{\partial\theta}\ln P(x_i;\theta)\right]P(x_i;\theta)dx_i = \frac{\partial g(\theta)}{\partial\theta}.$$

Applying the Cauchy-Schwartz inequality (Arfken, 1995, p. 569), this yields

$$\left[\frac{\partial g(\theta)}{\partial\theta}\right]^2 \leq \int(\hat{\theta}-g(\theta))^2P(x_i;\theta)dx_i\int\left[\frac{\partial}{\partial\theta}\ln P(x_i;\theta)\right]^2P(x_i;\theta)dx_i$$

or

$$V(\hat{\theta}) = \int(\hat{\theta}-g(\theta))^2P(x_i;\theta)dx_i \geq \frac{\left[\frac{\partial g(\theta)}{\partial\theta}\right]^2}{\int\left[\frac{\partial}{\partial\theta}\ln P(x_i;\theta)\right]^2P(x_i;\theta)dx_i}. \tag{B.9}$$

The Cramer-Rao lower bound, Eq. B.9, can be rewritten as

$$\therefore V(\hat{\theta}) \geq \frac{\left[\frac{\partial g(\theta)}{\partial \theta}\right]^2}{-E\left[\frac{\partial^2}{\partial \theta^2}\ln P(x_i;\theta)\right]} \tag{B.10}$$

since

$$E\left[\left(\frac{\partial}{\partial \theta}\ln P(x_i;\theta)\right)^2\right] = -E\left[\frac{\partial^2}{\partial \theta^2}\ln P(x_i;\theta)\right]. \tag{B.11}$$

We have proved the Cramer-Rao lower bound for a scalar parameter. Now the condition for the equality of CRLB can be given as from the Cauchy-Schwartz inequality,

$$\frac{\partial}{\partial \theta}\ln P(x_i;\theta) = c(\theta)(\hat{\theta} - g(\theta))$$

where $c(\theta)$ is a function of θ. Differentiate this and we can obtain

$$\frac{\partial^2}{\partial \theta^2}\ln P(x_i;\theta) = \frac{\partial c(\theta)}{\partial \theta}(\hat{\theta} - g(\theta)) - \frac{\partial g(\theta)}{\partial \theta}c(\theta)$$
$$E\left[\frac{\partial^2}{\partial \theta^2}\ln P(x_i;\theta)\right] = -\frac{\partial g(\theta)}{\partial \theta}c(\theta)$$

since $E\left(\hat{\theta} - g\left(\theta\right)\right) = 0$. Then using Eq. B.11, we can prove Eq B.7 by

$$c(\theta) = (\frac{\partial g(\theta)}{\partial \theta})^{-1}E\left[\frac{\partial}{\partial \theta}\ln P(x_i;\theta)\right] = (\frac{\partial g(\theta)}{\partial \theta})^{-1}\mathbb{F}(\theta)$$
$$\therefore \frac{\partial}{\partial \theta}\ln P(x_i;\theta) = (\frac{\partial g(\theta)}{\partial \theta})^{-1}\mathbb{F}(\theta)(\hat{\theta} - g(\theta))$$

□

If $g(\theta) = \theta$ and $P(x_{i=1\ldots n};\theta)$ is a Gaussian with variance σ_i^2, then Theorem B.1 can be expressed as follows.

Theorem B.2. *Suppose we have a set of data, $X = x_1 \ldots x_n$, and it is a function of an unknown scalar parameter θ as*

$$x_i = \theta + \epsilon_i,$$

where

$$\epsilon_i \sim P(x_{i=1\ldots n};\theta) = \frac{1}{\sqrt{2\pi\sigma_i^2}}\exp\left[-\frac{1}{2\sigma_i^2}(x_i - \theta)^2\right].$$

In addition, it is assumed that $P(x_i;\,\theta)$ satisfied the regularity condition

$$E\left[s(X;\theta)\right] = \sum_{i=1}^{n} E\left[\frac{\partial}{\partial\theta}\ln P(x_i;\theta)\right] = 0, \tag{B.12}$$

for all θ. Then the variance of any unbiased estimator, $\hat{\theta}$, must satisfy

$$V(\hat{\theta}) \geq \frac{1}{-\sum_{i=1}^{n} E\left[\frac{\partial^2}{\partial\theta^2}\ln P(x_i;\theta)\right]} = \frac{1}{\sum_{i=1}^{n}\frac{1}{\sigma_i^2}}. \tag{B.13}$$

An unbiased estimator, which have the lower bound for all θ, can be found if and only if

$$\sum_{i=1}^{n}\frac{\partial}{\partial\theta}\ln P(x_i;\theta) = \mathbb{F}(\theta)(\hat{\theta}-\theta). \tag{B.14}$$

Theorem B.2 is a simple extension of Theorem B.1 so it is presented without proof. It has several properties as follows:

- Unbiased estimator, $\hat{\theta}$, satisfies

$$\sum_{i=1}^{n}\frac{1}{\sigma_i^2}(x_i-\hat{\theta}) = 0 \tag{B.15}$$

  and therefore the unbiased estimator in this case always satisfies the regularity condition. If σ_i^2 is a constant within the data, i.e. x_i, then

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} x_i. \tag{B.16}$$

- The Fisher information is given as

$$\mathbb{F}(\theta) = \sum_{i=1}^{n}\frac{1}{\sigma_i^2}. \tag{B.17}$$

  Again if σ_i^2 is a constant, σ^2, within the data, then

$$\mathbb{F}(\theta) = \frac{N}{\sigma^2}.$$

  so the CRLB in this case is given as

$$V(\hat{\theta}) = \frac{1}{\mathbb{F}(\theta)} = \frac{\sigma^2}{N}. \tag{B.18}$$

Appendix C

Preprocessing procedures for raw point clouds from laser scanners

In this section, several preprocessing procedures for raw point clouds from laser scanner will be proposed. These methods are not necessary procedures for the proposed registration method. However, they may help to reduce either iterations or execution time. It must be noted that the effects of only some threshold values of these preprocessing procedures are quantified or analysed in this section.

Using the change of curvature defined in Section 4.3.2, a group of high curvature points can be detected, i.e. edge points. These edge points provide initial information for many applications using point clouds from laser scanners, e.g. automated feature extraction. Of course, edge points are also useful for the registration of point clouds since there is a better probability to obtain reliable correspondence from a pool of the edge points than that of low curvature points. As an application of the edge detection method, an algorithm for tree detection will be introduced in this section. This is useful for terrestrial laser scanners, which are often utilised for outdoor applications. For the registration or feature extraction of these data, a routine for removing unwanted data is important in terms of robustness and correctness.

However, it must be noted that the proposed algorithm for tree detection does not utilise colour or intensity information from either a laser scanner or a photogrammetric instrument. Therefore, this algorithm must be regarded as a preprocessing technique for the registration or feature extraction of laser scanner data.

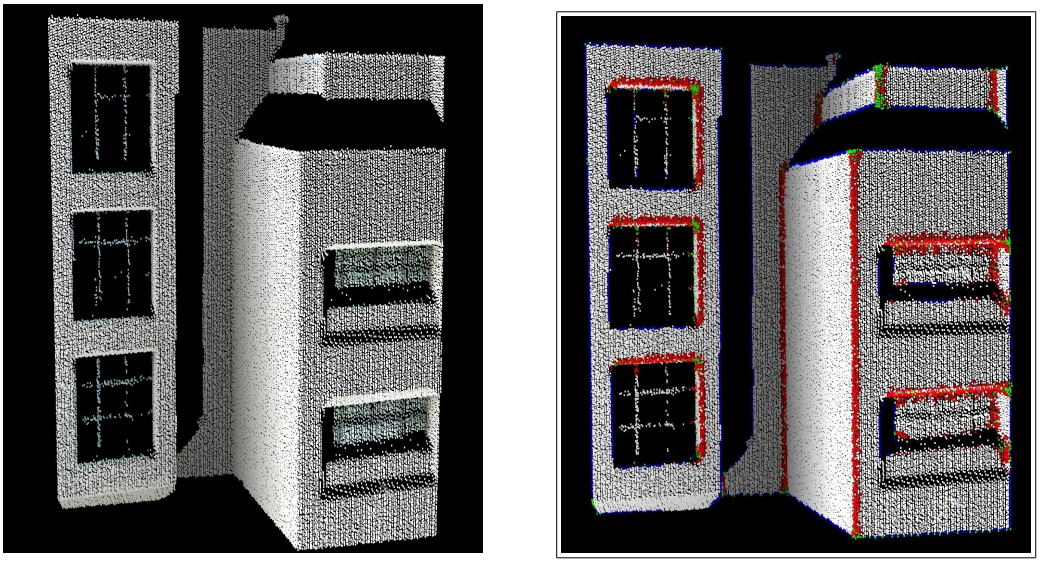
C.1 Edge detection

A simple fact that edge and corner points have higher curvature than other points is used as a basis for the proposed edge detection algorithm. Let $Edge\_or\_not(p_i)$ be a

boolean flag which indicates whether or not p_i is on edge. It can be simply defined as

$$Edge\_or\_not(p_i) = \begin{cases} \text{TRUE} & \text{if } M_{cc}(p_i) > T_{edge} \\ \text{FALSE} & \text{if } M_{cc}(p_i) \leq T_{edge} \end{cases}$$

where T_{edge} is a threshold set by the users. $\langle M_{cc}(p_i) \rangle + \langle M_{cc}(p_i) \rangle_{std}$ has been used as T_{edge} where $\langle M_{cc}(p_i) \rangle$ and $\langle M_{cc}(p_i) \rangle_{std}$ are the mean and standard deviation of changes of curvature in the point cloud, respectively.



(a) Point cloud of a building (b) Edge and boundary points

FIGURE C.1: (a) Point cloud of a building scanned with a Riegl LMS-Z210. The size of the point cloud is $(L, W, H) \simeq$ (17.0m, 10.0m, 16.0m). The density of the point cloud defined by the average distance of a point from its neighbour points is about 11cm which is larger than expected, since there are many empty spaces, e.g. windows in the building. (b) Detected edge and boundary points are respectively denoted by red and blue.

C.2 Boundary point detection

Boundary points also provide useful information as shown by the blue points in Figure C.3. A boundary point detection method based on the angle between a point and its neighbour points will be introduced. In three-dimensional space, the sum of the angles between a point and its neighbour points is not 2π. Consequently, the angles between points in three-dimensional space is not an useful indicator of boundary points. To overcome this problem, every point in a local neighbourhood is projected onto a plane formed by the point and its neighbour points. Note that this is a local, not global, projection. Suppose $\overline{p_i}$ and $\overline{p_{neighbour\{j,p_i\}}}$ are the projected points of p_i and $p_{neighbour\{j,p_i\}}$ and the dimension of $\overline{p_i}$ is two in the case of three-dimensional point clouds as shown in Figure C.2. In other words, $\overline{p_i}$ is the projected, local, and two-dimensional coordinate that is represented by $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$ of $\mathbf{COV}(p_i)$. Let $\varphi(\overline{p_{neighbour\{1,p_i\}}}, \overline{p_{neighbour\{j,p_i\}}})$ be the angle between $\overline{p_{neighbour\{1,p_i\}}}$, $\overline{p_i}$, and $\overline{p_{neighbour\{j,p_i\}}}$ and let $Boundary\_or\_not(p_i)$ be a boolean flag which indicates whether p_i is boundary or not. A brief description of the proposed algorithm is given in Algorithm 4.

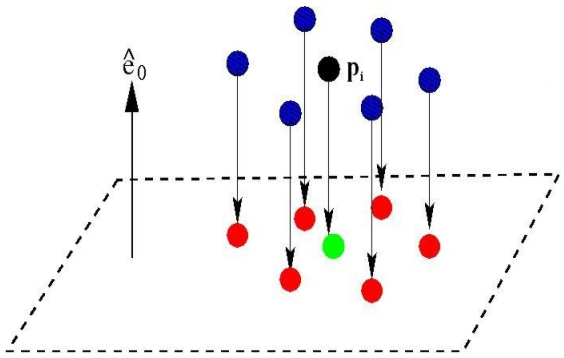
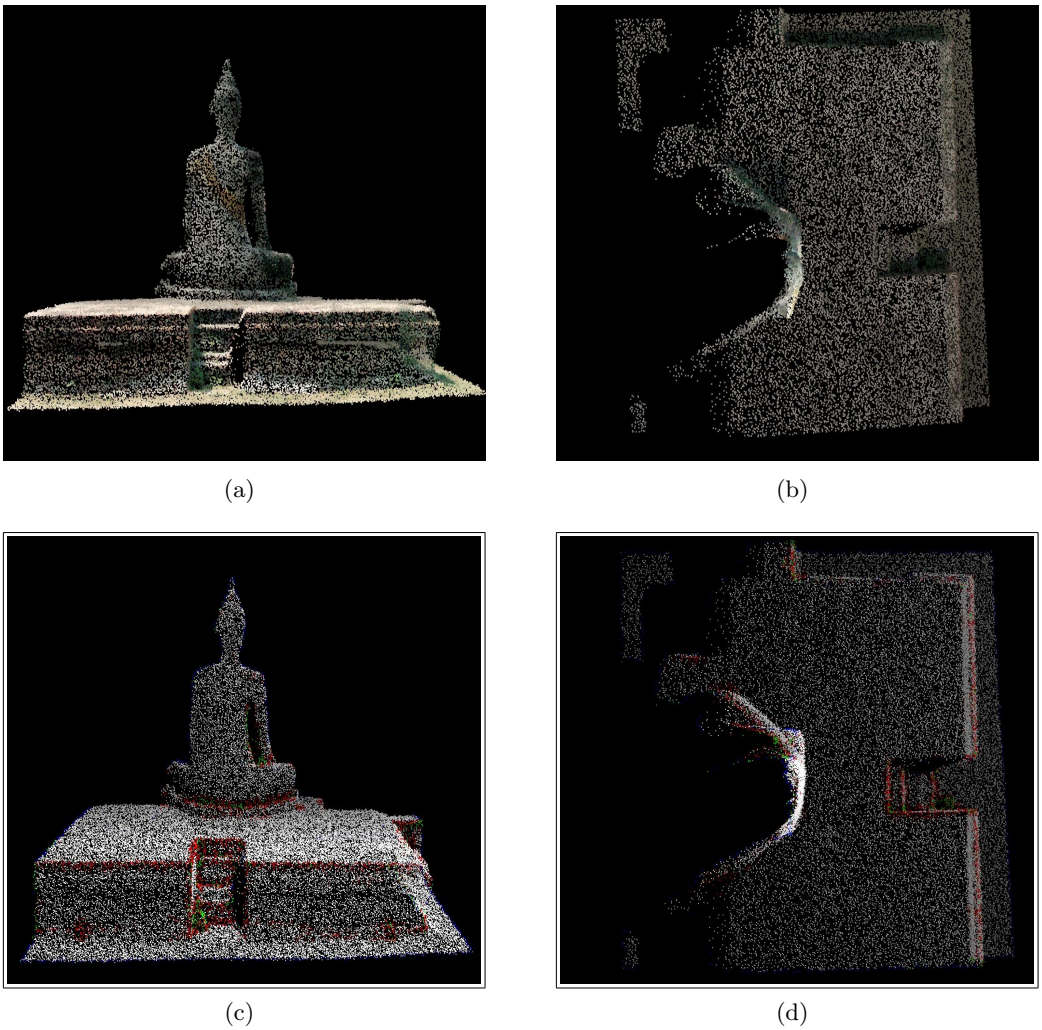


FIGURE C.2: p_i and $\overline{p_i}$ are black and green, respectively. Note that p_i and $\overline{p_i}$ are a query point and its projected point in the plane whose normal vector is $\hat{\mathbf{e}}_0$, respectively. $p_{neighbour\{j,p_i\}}$ and $\overline{p_{neighbour\{j,p_i\}}}$ are represented blue and red, respectively. Also note that $p_{neighbour\{j,p_i\}}$ and $\overline{p_{neighbour\{j,p_i\}}}$ the jth neighbour point of the query point and its projected point, respectively.



(a) (b)

(c) (d)

FIGURE C.3: (a) Point clouds of the Ayuthaya Buddha statue in Thailand (back) (b) Point cloud of the Buddha statue (top) (c) edge and boundary of the point cloud (back) (d) edge and boundary of the point cloud (top) The size of the point cloud is $(L, W, H) \simeq$ (6.0m, 8.0m, 4.0m). The density of the point cloud that is defined by the average distance of a point from its neighbour points is about 4cm. Red and blue represent edge and boundary points, respectively.

Algorithm 4 Boundary points detection algorithm

1: **for** i = 1 to i = Numberofpointcloud **do**
2: Assign the initial index on the neighbour points of p_i based on the distances from it.
3: Select a starting point, e.g. point 1 in Figure 4.2.
4: Calculate $\varphi(p_{neighbour\{1,p_i\}}, p_{neighbour\{j,p_i\}})$ where $j = 2 \cdots k$.
5: Assign the new indices to $p_{neighbour\{j,p_i\}}$ by $\varphi(p_{neighbour\{1,p_i\}}, p_{neighbour\{j,p_i\}})$ as shown in Figure 4.2-(a).
6: Calculate the angle between the closest neighbour points based on the newly assigned indices, i.e. $\varphi(p_{neighbour\{1,p_i\}}, p_{neighbour\{2,p_i\}})$, $\varphi(p_{neighbour\{2,p_i\}}, p_{neighbour\{3,p_i\}})$, and so on.
7:

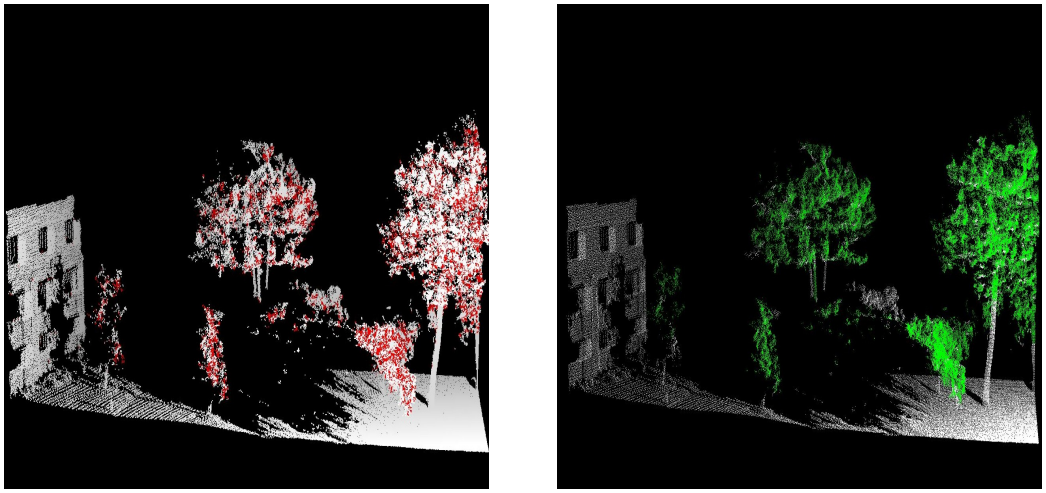
$$\mathit{Boundary\_or\_not}(p_i) = \begin{cases} \text{TRUE} & \text{if } MAX(\varphi(p_{neighbour\{j,p_i\}}, p_{neighbour\{j+1,p_i\}})) > T_{boundary} \\ \text{FALSE} & \text{if } MAX(\varphi(p_{neighbour\{j,p_i\}}, p_{neighbour\{j+1,p_i\}})) \leq T_{boundary} \end{cases}$$

where $j = 1 \cdots k$ and if $j = k$, then $j + 1 = 1$. $MAX(b)$ is the maximum of the array of scalar b.

8: **end for**

C.3 Tree detection using the change of curvature

In this section, an algorithm for tree detection in a point cloud from laser scanners will be introduced as an application of edge detection. This can be used as a preprocessing method for the registration of point clouds since points in trees may have a larger position error than that in other objects, e.g. buildings. The proposed algorithm is a region-growing method and utilises the fact that trees have a higher curvature than man-made objects. It must be noted that the proposed method for tree detection is a preprocessing procedure for the registration of point clouds in order to avoid the unnecessary execution time for searching corresponding points. Therefore, I do not quantify the success rate of the proposed method for tree detection but empirically and visually show its effectiveness. In addition, the ground is assumed to be already detected. As shown in Figure C.4-(a), a large portion of higher curvature points are on the trees. However, structures other than trees also have higher curvature points so they need to be detected. To filter out the points on man-made objects, the percentage of edges in the neighbourhood, $T_{initial\_filter}$, is used. For the sake of a more strict filtering condition, other kinds of thresholds may be used such as the percentage of high curvature points in the neighbourhood and the direction of a surface normal vector compared with the surface normal vector of the ground.



(a) Side view (b) Side view

FIGURE C.4: Tree detection within a point cloud. The size of the point cloud is $(L, W, H) \simeq$ (45.0m, 30.0m, 18.0m) and the density of the point cloud is about 13cm. (a) Red points have higher curvature than white ones. (b) Green points are detected trees. This was scanned with a Riegl LMS-Z210.

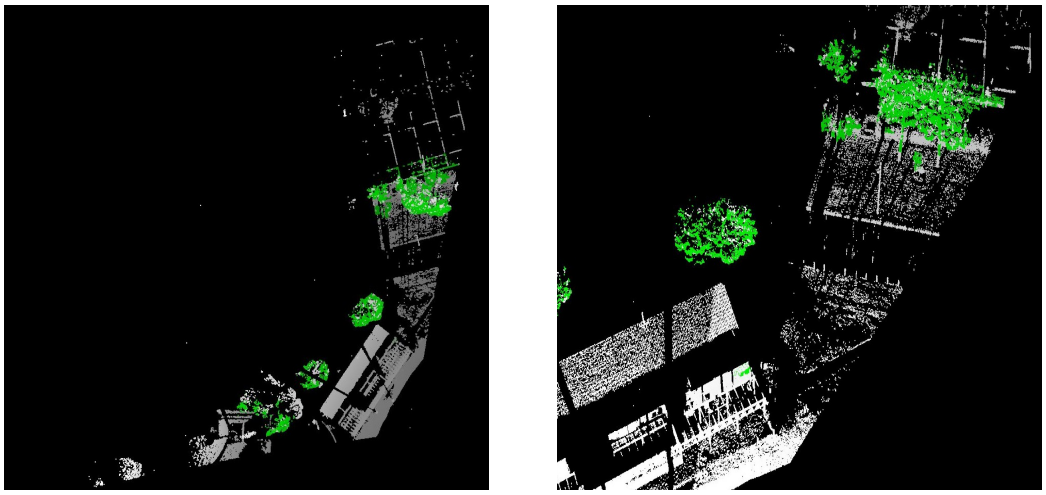
First points on the trees need to be detected and used as seed points for region growing. A realistic and reasonable termination criterion for region growing is critical and the percentage of edges, $\tau(p_i)$, in the neighbourhood, which is used as the thresh-

old, $T_{region\_growing}$. Let $SEED$ be a seed point and also let $p_{region\_growing\{j,SEED\}}$ and $n_{region\_growing}$ be the jth and the number of neighbourhood of $SEED$ for region-growing, respectively. In addition, let *Tree_or_not*(p_i) and *Ground_or_not*(p_i) be boolean flags for tree and ground, respectively. A brief description of the proposed algorithm is given in Algorithm 5. Results of tree detection are displayed in Figures C.4 and C.5. $T_{initial\_filter}$ and $T_{region\_growing}$ were set as 0.10 and 0.95, respectively.

It is important to choose an optimal $T_{initial\_filter}$. It depends mainly on the geometric shape of man-made objects and the density of the point cloud. In other words, it is difficult to express $T_{initial\_filter}$ in an explicit form, although 0.10 has been used as $T_{initial\_filter}$ without problem as shown in Figures C.4 and C.5. Radiometric information, e.g. the NDVI (Normalized Difference Vegetation Index), may be used in order to reinforce this method by filtering out the higher curvature points on man-made objects. Incorrect tree detection is found in the top-right side of Figure C.5 which is caused by low point density in the region. The spatial density of a point cloud is inversely proportional to the nominal distance from the scanner. If the spatial density of a point cloud is too low, i.e. an insufficient number of points represent the surface of objects, then good estimates of the normal vector and change of curvature are difficult to obtain and thus incorrect tree detection may result.

Algorithm 5 Tree detection algorithm

1: Detect edge points as explained in Section C.1 and let their *Tree_or_not*() be TRUE.
2: **for** $i = 1$ to $i = Numberofpointcloud$ **do**
3: **if** $\tau(p_i) \leq T_{initial\_filter}$ **then**
4: *Tree_or_not*(p_i) = FALSE.
5: **end if**
6: **end for**
7: **for** $i = 1$ to $i = Numberofpointcloud$ **do**
8: $SEED = p_i$
9: **repeat**
10: *Tree_or_not*($p_{region\_growing\{j,SEED\}}$) = TRUE, $j = 1 \cdots n_{region\_growing}$
11: $SEED \leftarrow$ the closest point to the ground among $p_{region\_growing\{j,SEED\}}$
12: **until** *Ground_or_not*($SEED$) $\neq$ FALSE **AND** $\tau(SEED) \leq T_{region\_growing}$
13: **end for**



(a) Top view (b) Zoomed top view

FIGURE C.5: Tree detection. The size of the point cloud is $(L, W, H) \simeq$ (110.0m, 120.0m, 23.0m) and the density of the point cloud is about 0.10m. (a) Red points have higher curvature than white ones. Green points are detected trees. (b) The part of (a) is magnified. These point clouds were scanned with a Mensi GS200 (Mensi, 2006).

Appendix D

Estimated transformation parameters of the Agia Sanmarina church datasets using four different methods

	ω [degree]	ϕ [degree]	κ [degree]	x [m]	y [m]	z [m]
northeast	5.2348	42.5217	-9.2648	15.6092	-0.3940	-2.1943
southeast	1.7733	-28.5439	4.9421	-12.1713	-0.0750	0.7681
south	16.0144	-66.0504	21.2193	-21.8109	-1.8719	-13.1771
southwest	166.5090	-65.8329	176.2946	-23.1611	-2.9780	-31.9504
west	174.8801	-13.5084	-179.6002	-6.5053	-3.9809	-43.7694
northwest	169.9493	23.5739	-177.7773	9.5302	-5.0149	-49.6184
north	163.3609	88.6154	183.5745	18.4227	-1.9446	-20.6847
crane1	-30.1376	55.3583	20.2144	14.1157	6.3386	-9.8567
crane2	-20.1909	-34.0563	-5.7680	-12.8165	7.3489	-3.2605

TABLE D.1: Using the direct georeferencing method

	ω [degree]	ϕ [degree]	κ [degree]	x [m]	y [m]	z [m]
northeast	5.2084	42.5260	-9.1781	15.6110	-0.3800	-2.1950
southeast	1.8632	-28.5569	5.0269	-12.1740	-0.1020	0.7630
south	16.1080	-66.0712	21.2994	-21.8090	-1.8790	-13.1860
southwest	166.2199	-65.7700	176.0294	-23.1430	-3.0180	-31.9570
west	174.9543	-13.4983	-179.7258	-6.4930	-3.9570	-43.7770
northwest	170.0068	23.6248	-177.8590	9.5510	-4.9780	-49.6150
north	167.3256	88.6397	179.6328	18.4250	-1.9070	-20.6870
crane1	-30.3202	55.2826	20.4302	14.0990	6.3760	-9.8560
crane2	-20.0202	-34.1123	-5.6648	-12.8290	7.3100	-3.2580

TABLE D.2: Using Cyclone

	ω [degree]	ϕ [degree]	κ [degree]	x [m]	y [m]	z [m]
northeast	5.2066	42.5293	-9.1077	15.6144	-0.3845	-2.1981
southeast	1.8421	-28.5486	5.0304	-12.1673	-0.0963	0.7672
south	16.1661	-66.0764	21.2756	-21.8038	-1.8800	-13.1864
southwest	166.0686	-65.7628	175.5593	-23.1379	-3.0263	-31.9498
west	175.0059	-13.4713	-179.8025	-6.4807	-3.9242	-43.7650
northwest	170.1063	23.6703	-177.9640	9.5790	-4.9151	-49.6002
north	167.4104	88.6878	179.6167	18.4339	-1.8966	-20.6826
crane1	-30.2648	55.2775	20.4575	14.1022	6.3764	-9.8568
crane2	-20.1978	-34.1528	-5.7772	-12.8375	7.3431	-3.2768

TABLE D.3: Using GP-ICP

	ω [degree]	ϕ [degree]	κ [degree]	x [m]	y [m]	z [m]
northeast	5.2353	42.5466	-9.1497	15.6140	-0.4004	-2.1971
southeast	1.8513	-28.5850	5.0745	-12.1795	-0.0905	0.7592
south	16.1592	-66.1008	21.2677	-21.8043	-1.8812	-13.1922
southwest	166.2541	-65.7560	175.6679	-23.1342	-3.0050	-31.9689
west	175.0918	-13.4585	-179.7856	-6.4674	-3.8991	-43.7818
northwest	170.1388	23.7165	-177.9080	9.5946	-4.9015	-49.6031
north	167.1116	88.6460	179.8689	18.4255	-1.9212	-20.6836
crane1	-30.2730	55.2927	20.4547	14.0981	6.3579	-9.8534
crane2	-20.2012	-34.1688	-5.8058	-12.8392	7.3471	-3.2786

TABLE D.4: Using GP-ICPR

	$d\omega$ [degree]	$d\phi$ [degree]	$d\kappa$ [degree]	dx [cm]	dy [cm]	dz [cm]
northeast	-0.028	0.0076	0.16	0.52	0.95	-0.38
southeast	0.069	-0.0047	0.088	0.40	-2.13	-0.090
south	0.15	-0.026	0.056	0.71	-0.81	-0.93
southwest	-0.44	0.070	-0.74	2.32	-4.83	0.060
west	0.13	0.037	-0.20	2.46	5.67	0.44
northwest	0.16	0.096	-0.19	4.88	9.98	1.82
north	4.050	0.072	-3.96	1.12	4.80	0.21
crane1	-0.13	-0.081	0.24	-1.35	3.78	-0.010
crane2	-0.0069	-0.096	-0.0092	-2.10	-0.58	-1.63
mean	0.44	0.0084	-0.51	1.00	1.87	-0.057

TABLE D.5: Using the direct georeferencing method vs GP-ICP

	$d\omega$ [degree]	$d\phi$ [degree]	$d\kappa$ [degree]	dx [cm]	dy [cm]	dz [cm]
northeast	-0.0018	0.0033	0.070	0.34	-0.45	-0.31
southeast	-0.021	0.0083	0.0035	0.67	0.57	0.42
south	0.058	-0.0052	-0.024	0.52	-0.10	-0.040
southwest	-0.15	0.0072	-0.47	0.51	-0.83	0.72
west	0.052	0.027	-0.077	1.23	3.28	1.20
northwest	0.10	0.050	-0.11	2.80	6.29	1.48
north	0.085	0.048	-0.016	0.89	1.040	0.44
crane1	0.055	-0.0051	0.027	0.32	0.040	-0.080
crane2	-0.18	-0.041	-0.11	-0.85	3.31	-1.88
mean	-0.00027	0.0098	-0.078	0.71	1.46	0.22

TABLE D.6: Using Cyclone vs GP-ICP

	$d\omega$ [degree]	$d\phi$ [degree]	$d\kappa$ [degree]	dx [cm]	dy [cm]	dz [cm]
northeast	-0.029	-0.017	0.042	0.044	1.6	-0.096
southeast	-0.0092	0.036	-0.044	1.2	-0.58	0.80
south	0.0069	0.024	0.0079	0.051	0.12	0.58
southwest	-0.19	-0.0068	-0.11	-0.37	-2.1	1.9
west	-0.086	-0.013	-0.017	-1.3	-2.5	1.7
northwest	-0.032	-0.046	-0.056	-1.6	-1.4	0.29
north	0.30	0.042	-0.25	0.84	2.5	0.098
crane1	0.0082	-0.015	0.0028	0.41	1.8	-0.34
crane2	0.0034	0.016	0.029	0.17	-0.40	0.18
mean	-0.0027	0.0022	-0.044	-0.059	-0.11	0.57

TABLE D.7: GP-ICPR vs GP-ICP

Appendix E

Statistical multivariate analysis : Hotelling T^2 test

In this section, a multivariate analysis of variance, so called MANOVA, is briefly introduced without proof. One can find many practically useful examples in Chapter six of Johnson and Wichern (1998) and some useful properties in Mardia et al. (1979, pp. 73-79).

When we have more than one sample and they have different statistics, a method for multivariate analysis is required. Since we would like to compare the estimated transformation parameters from different kinds of methods such as the direct georeferencing method, commercial software, GP-ICP, and GP-ICPR, Hotelling's T^2 distribution is chosen as the statistical inference for the validity of the estimated parameters.

Lemma E.1. *Definition of Hotelling's T^2 test statistic and its relationship with the Fisher distributions*

Suppose that $\bar{\mathbf{X}} = \{X(1,1), X(2,1), \cdots, X(n,1)\}$ and $\bar{\mathbf{S}}$ are the mean vector and unbiased covariance matrix of a sample with n from $N(\boldsymbol{\delta}, \boldsymbol{\Sigma})$ which is a multivariate Gaussian function with $\boldsymbol{\delta}$ and $\boldsymbol{\Sigma}$ as the mean and variance matrix, respectively. Note that $\mathbf{X} \in \Re^{n\times 1}$ and $\mathbf{S} \in \Re^{n\times n}$. In addition, let us assume that each element of the sample is independent of each other, i.e. $\boldsymbol{\Sigma}$ is a diagonal matrix. The Hotelling's T^2 test statistic, T^2, is defined as

$$T^2 = n\left(\bar{\mathbf{x}} - \boldsymbol{\delta}\right)^T \mathbf{S}^{-1}\left(\bar{\mathbf{x}} - \boldsymbol{\delta}\right).$$

In addition, the relationship between the Hotelling's T^2 and the Fisher distribution is given as

$$T^2 \sim \frac{(n-1)\,p}{(n-p)} F_{n,n-p}(\alpha)$$

where $F_{n,n-p}(\alpha)$ is a Fisher distribution with n and $n-p$ degrees of freedom and α is the upper significant level of this Fisher distribution. In addition, we can estimate the (simultaneous) confidence interval of $X(i,1)$, $X(\hat{i},1)$, as follows,

$$\hat{X}(i,1) = \bar{X}(i,1) \pm \left[\frac{(n-1)\,p}{(n-p)} F_{n,n-p}(\alpha)\right]^{\frac{1}{2}} \left[\frac{S(i,i)}{n}\right]^{\frac{1}{2}}$$

where $S(i,i)$ is the ith diagonal component of $\mathbf{S}$ and p is the rank of $\mathbf{S}$.