DISS. ETH NO. 22870

Automated Registration of Terrestrial Laser Scanner Point Clouds

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH

presented by

Pascal Willy Theiler

Master of Science (M.Sc.), ETH Zurich

born on 25.11.1986

citizen of Switzerland

accepted on the recommendation of

Prof. Dr. Konrad Schindler, examiner
ETH Zurich, Switzerland

Prof. Dr. Jon Mills, co-examiner
Newcastle University, England

2015

IGP Mitteilungen Nr. 117
Automated Registration of Terrestrial Laser Scanner Point Clouds
Pascal W. Theiler

Published by:
Institute of Geodesy and Photogrammetry
ETH Zurich
CH - 8093 Zürich

ISBN 978-3-03837-004-8

Abstract

Recording and modelling our environment is of general interest in various application fields such as geo-sciences or architecture. Terrestrial Laser Scanning (TLS) has thereby evolved as a standard tool to acquire accurate 3D data of objects or areas of interest. Over the last decade, TLS sensors have experienced rapid development, for example regarding measurement speed and accuracy, while at the same time the demands from end-users has increased . To keep face with this evolution, the processing of TLS data needs to be highly efficient.

This thesis addresses the automated registration of 3D point clouds from static terrestrial laser scanners. Due to the line-of-sight measurement principle of TLS, multiple scans from varying viewpoints are necessary e.g. to record complete 3D objects. Registration (or alignment) thus becomes necessary to orient the various, locally defined laser scans within a common coordinate system. The state-of-practice approach to register TLS point clouds is nowadays either to manually place artificial markers within the acquired scene, or to use a semi-automated processing scheme including manual point picking. The overall aim of this thesis is to present new methods for the automated registration of static TLS point clouds. In particular, only the natural scene content and no markers should be exploited, and no assumptions are made about initial orientation of the scanners. Aligning static TLS point clouds is especially demanding because of *(i)* the size of the point clouds, typically with millions of points, *(ii)* the uneven point distribution over the whole scene due to the regular angular measurement principle of TLS sensors, and *(iii)* the typically rather low overlap between scan pairs.

At first, a method to achieve pairwise coarse alignment has been developed. The approach is based on extracting naturally existing planar surfaces. 3D keypoints, generated via plane triplet intersection, are then matched based on a newly developed descriptor as well as geometrical constraints. The method is able to correctly align scans of indoor environments in $\approx 90\%$ of all tests. In a second approach, we have aimed for a more generic coarse registration method. The basic idea is to reduce the large point clouds to derive sparse sets of 3D keypoints. Matching of these keypoints is based only on geometrical information by adapting an efficient four-point matching algorithm. The resulting registration scheme achieves pairwise alignments with success rates of nearly 100% in practical runtimes (seconds to minutes) for a wide range of TLS applications. Remaining failures are typically caused by unambiguous scene content (e.g. symmetric rooms). Finally, the registration scheme was extended to an entire pipeline. The resulting framework includes the previously developed keypoint-based pairwise matching approach and complements it with a generic, global point cloud alignment scheme. The pipeline is completed by state-of-the-art fine registration methods. The framework achieves correct alignment of all scans for a variety of real-world datasets in practical runtimes (e.g. below seven minutes for a fully connected network of eight scans).

Zusammenfassung

Die Erfassung und Modellierung unserer Umwelt ist von allgemeinem Interesse in verschiedensten Anwendungsgebieten wie Geowissenschaften oder Architektur. Terrestrisches Laserscanning (TLS) hat sich dabei als wertvolle Methode zur genauen dreidimensionalen Erfassung von Objekten und Gebieten etabliert. Im Laufe des letzten Jahrzehnts haben TLS Sensoren eine rasante Entwicklung erlebt, insbesondere im Bereich der Datenerfassung, während gleichzeitig die Anforderungen seitens der Endnutzer gestiegen sind. Um mit dieser Entwicklung Schritt zu halten bedarf es effizienter Methoden zur Verarbeitung der TLS-Daten.

Diese Dissertation befasst sich mit der automatischen Registrierung von 3D-Punktwolken aus statischem terrestrischem Laserscanning. Da Laserscanner zur Kategorie der optischen Messinstrumente gehören, braucht es mehrere Scanaufnahmen aus unterschiedlichen Blickwinkeln, um ein Objekt vollständig und lückenlos zu erfassen. Dadurch wird eine Registrierung der einzelnen Scans notwendig, was der Orientierung der Punktwolken in einem gemeinsamen Koordinatensystem entspricht. Dabei werden in der Praxis meist Zielmarken eingesetzt, welche im aufzunehmenden Gebiet platziert werden. Das übergeordnete Ziel dieser Arbeit ist die Entwicklung von neuen Methoden zur automatischen Registrierung von TLS-Daten unter Verwendung der natürlich vorhandenen Umgebung ohne zusätzliche Zielmarken. Die automatische Registrierung von statischen TLS-Punktwolken wird durch *(i)* die Scangrösse mit Millionen von Punkten, *(ii)* die unregelmässige Punktverteilung über das Aufnahmegebiet hinweg und *(iii)* die häufig gering gewählte Überlappung zwischen Scanpaaren zusätzlich erschwert.

In dieser Arbeit wird zuerst eine neu entwickelte Methode zur groben Ausrichtung von Laserscans beschrieben. Diese basiert auf dem Matching von 3D Keypoints, welche durch die Verschneidung von jeweils drei aus den Scans extrahierten Ebenen definiert sind. Die Methode ermöglicht in 90% der Fälle eine korrekte Registrierung von Scans eines Innenraumes mit klar definierten Ebenen. In einem zweiten Ansatz wird ein ausschliesslich geometrisches Matching von Keypoints mit Hilfe eines Vier-Punkt-Algorithmus angestrebt. Die Methode ist generisch konzipiert und kann auf beliebige Keypoints angewendet werden. Experimente mit Erfolgsraten von annähernd 100% haben gezeigt, dass die entwickelte Methode für eine breite Palette von TLS-Anwendungen einsetzbar ist. Zum Schluss wurde ein kompletter Arbeitsablauf zur automatischen Registrierung von Laserscans entwickelt, welcher sowohl die Grob- als auch die Feinausrichtung von mehreren Punktwolken beinhaltet. Die zuvor entwickelte Matchingstrategie zur paarweisen Grobausrichtung wird mit einem globalen Verifizierungsschritt ergänzt. Die Feinregistrierung basiert auf Standardmethoden. Mithilfe dieses Registrierungsverfahren konnten Scans verschiedenster, realer TLS-Datensätzen automatisch und korrekt zueinander orentiert werden.

Acknowledgements

First and foremost I would like to thank **Prof. Konrad Schindler** for giving me the opportunity to conduct a PhD under his supervision. His deep knowledge in geometrical processing and his ability to describe even complex problems in simple terms were highly impressive, inspiring and instructive.

I also want to thank **Prof. Hilmar Ingensand** who motivated me to start a PhD. Although he was not able to supervise me due to his up-coming retirement, he offered me to work as an assistant in his group. He organized the supervision through Prof. Konrad Schindler and thus made it possible to work in two different groups simultaneously.

I am very thankful to his successor **Prof. Andreas Wieser** for not only keeping me as an assistant in his group but also for his inputs and readiness to additionally support me in my PhD.

In retrospect I can honestly say that this "hybrid" situation of assisting in one group and researching in the other was very valuable and fruitful, especially because the cooperation between the two groups is excellent.

My gratitude also goes to **Prof. Jon Mills** from Newcastle University, who agreed to be co-examiner, and as such was willing to spend some demanding hours (or days) reviewing my thesis.

A very special thanks goes to **Dr. Jan Dirk Wegner**, who not only played a very helpful role as motivation coach, but who also became a really good friend. Apart from sharing his knowledge, he steadily prompted me to pursue the important issues which perfectly compensated my tendency of loosing myself in details.

The privilege of participating in two groups also put me in the nice situation of being surrounded by several "good fairies" (officially called administrative staff). They made my working time at ETH very pleasant, especially with all the organized social events. A great thanks to **Susanna, Monique, Alessandra and Helena**.

I very much enjoyed these years at ETH, and many people have contributed to that great experience. But I still like to specially thank **Dr. Sebastian Tilch** and **Dr. Piotr Tokarczyk**. I have spent quite some time with Sebastian, first as fellow students in the master studies, followed by our common years as assistants. Over four years, we have shared not only our office, but also many thoughts and ideas, often not only concerning work. In Piotr I have found a companion and friend. Together we have spent a very entertaining time, be it during short office breaks, on "Hönggerberg walks", after work, or during our legendary trip through Australia.

I would also like to thank all **working colleagues**, present and past, for the great time we had. From the *GSEG* group (former *geomETH*); Claude, Claudia, Daniel, David, Ephraim, Eugenio, Geo, Henri, Jemil, Ladina, Madeleine, Mariusz, Melanie, Nicole, Rainer, Robert, Stefan and Tobias. And from the *Photi* group; Audrey, Charis, Christoph, David, Javier, Kathrin, Laura, Manos, Maros, Michal, Nusret, Silvano, Stefan, Sultan, Timo, Wilfried, Zeeshan.

Furthermore, I would like to mention the many **students** with whom I have worked over these years, be that during lectures, labs or as a supervisor of student projects. It was always a pleasure to pass on know-how and to teach interested students.

A great thanks goes to my **family** and **friends** for their support and for the nice time I have spent with them outside ETH which provided for me the perfect balance between work and life. An extra thanks goes to my uncle **Hansruedi** for proof-reading this thesis. Finally I want to say a very special “thank you” to **Carmen** for her patience, for her interest in my work, and for her always being here for me and finding the right way to motivate and support me.

Contents

List of Figures

List of Tables

List of Abbreviations

4PCS	4-Points Congruent Sets
ALS	Airborne Laser Scanning
CPU	Central Processing Unit
DoF	Degree of Freedom
DoG	Difference-of-Gaussians
FPFH	Fast Point Feature Histogram
GCA	Global Consistent Alignment
GNSS	Global Navigation Satellite System
Go-ICP	Globally Optimal Iterative Closest Point
GPU	Graphics Processing Unit
GS	Global Scan Registration Success
GSR	Global Scan Registration Success Rate
ICP	Iterative Closest Point
i.i.d.	Independent and Identically Distributed
INS	Inertial Navigation System
ISS	Intrinsic Shape Signatures
K-4PCS	Keypoint-based 4-Points Congruent Sets
LS3D	Least Squares 3D Surface Matching
LS	Laser Scanning
LUM	Global Registration Method by Lu and Milios
MAE	Mean Angular Error
MLESAC	Maximum Likelihood Estimation Sample Consensus
MSAC	M-Estimator Sample Consensus
MST	Minimum Spanning Tree
MTE	Mean Translation Error
NP-hard	Non-deterministic Polynomial-time hard
PCL	Point Cloud Library
PE	Pairwise Alignment Error

PER	Pairwise Alignment Error Rate
PFH	Point Feature Histogram
PS	Pairwise Alignment Success
PSR	Pairwise Alignment Success Rate
LiDAR	Light Detection and Ranging
RADAR	Radio Detection and Ranging
RANSAC	Random Sample Consensus
RMSE	Root Mean Square Error
SAC-IA	Sample Consensus Initial Alignment
SIFT	Scale Invariant Feature Transform
SURF	Speeded up Robust Features
SVD	Singular Value Decomposition
THRIFT	3D version of SIFT descriptor
TLS	Terrestrial Laser Scanning
ToF	Time of Flight
TSR	Total Project Alignment Success Rate
VER	Virtual Alignment Error Rate
VSR	Virtual Alignment Success Rate

Chapter 1

Introduction

1.1 Motivation

Laser Scanning (LS) has nowadays become a standard method to acquire high resolution 3D data. Given an object of interest, LS records its geometry using non-contact distance measurements and angular readings. The results of LS measurements are dense 3D point clouds which generally serve as a basis for further tasks like 3D surface generation, object detection, or deformation analysis. The importance of laser scanners for the global market has recently been analysed and published by MarketsandMarkets (2015). The market value of 3D scanners was thereby estimated to reach 3.8 billion US dollars by the year of 2020, with an average annual growth rate of $\approx 9\%$ between 2015 and 2020. According to this report, the main reason for this development is the increasing competition in the laser scanner market. Moreover, because LS ensures high product quality in short acquisition time, scanners are increasingly used in various application fields such as geo-sciences, automotive, health-care, archaeology, architecture and many more.

The focus of this thesis is on static *Terrestrial Laser Scanning*, which is a special case of LS where a ground-based instrument is used in a stationary mode. The development of TLS over the last ten years was rapid, especially with respect to accuracy and data acquisition speed. Today it is common to have sensors which measure with frequencies of up to one million points per second and simultaneously achieve distance accuracies in the order of a few millimetres. Arguably, the challenge of working with TLS is not the data acquisition but the time-consuming and cumbersome post-processing of the large point clouds. Depending on the application, one typically reaches ratios between acquisition and processing time of one to ten, i.e., a single day of scanning yields ten days of post-processing.

The need for more efficient processing of terrestrial laser scans have triggered a large number of publications, in particular considering the fact that up-to-now, data analysis often includes manual work. The goal of the research in TLS point cloud processing is thus to exploit the

full potential of scanners and achieve economic efficiency for new and existing applications. The general course is thereby to develop automated methods for solving time-consuming and often manually accomplished steps such as object detection, object modelling, filtering, or registration.

This thesis addresses the task of 3D point cloud registration which can be considered as one of the basic steps of scan data processing. To cover larger areas and acquire complete 3D objects with static TLS, the fixed instrument setup and the line-of-sight measurement principle necessitates for multiple acquisitions from different viewpoints. This in turn makes it necessary to register the different scans relative to each other, because every view has its own local coordinate system centred at the scanner position with typically random orientation.

Registration (also called *alignment* or *orientation*) is defined as the task of finding rigid-body transformation parameters which align a scan within a predefined reference frame, or a local system defined by an arbitrary reference scan, respectively. In the case of TLS with absolute, metric distance measurements, this amounts to the estimation of transformation parameters with six *Degrees of Freedom* (DoF); three rotation and three translation.

Registration is generally accomplished as follows: find corresponding points between the to-be-registered point clouds, then derive the parameters of the rigid-body transformation in a least squares sense based on these correspondences. The registration of laser scans is a six DoF problem and hence requires a minimum of three corresponding points per scan pair. It is important to note that assuming a vertical z-axis of the scanner (i.e. the instrument is levelled) registration amounts to estimating four parameters which is solvable using only two correspondences. In practice, TLS sensors are not always levelled, e.g., if mounted on a crane, or flipped to record more details on the ground, etc. So as not to restrict our methods to the special case of levelled instruments, and to remain as generic as possible, registration is considered to be a six DoF problem throughout this thesis.

Until now, the state-of-practice approach to register TLS point clouds is based on manually placed artificial markers within the acquired scene. Typical markers are spheres, checker-boards, or retro-reflective targets. The parameters of the rigid-body transformation are calculated based on the position of the markers (typically reduced to a single point such as the centre of a sphere). Often a subsequent *Iterative Closest Point* (ICP) method refines the alignment. The detection is either done manually or automatically. Examples of automated approaches are described in Section 2.2.1. Also professional software has kept up with the trend. Nowadays we have several tools for (semi-)automated registration of laser scans based on typical markers.

While the marker-based registration is very robust and reliable (which is probably the reason it is still mostly applied in industry) it also suffers from some drawbacks.

- Placing markers in the scene requires additional time and is not a trivial task. One must ensure the visibility of the markers across various scans (at least three per scan pair). Additionally, degrading constellations (e.g. markers along a single line) should be avoided.
- Although scanners work with very high measurement frequencies, a typical single acquisition still requires several minutes up to an hour (depending on the required resolution). A scan project consisting of many stations therefore quickly reaches several working hours during which one has to make sure that the markers remain stable.
- The in-scene placed markers may disturb further post-processing tasks such as object modelling or data visualization, i.e., they are considered to be artefacts. Hence they have to be (manually) removed from the scans.
- Due to the line-of-sight acquisition characteristic of TLS, objects which are occluded by the markers are not recorded. If markers are applied on the object of interest, the loss of information can hardly be corrected, otherwise (e.g. if placed on tripods) further scans from different viewpoints might solve the problem but potentially increase the acquisition time.

A logical consequence is to avoid the usage of markers, but retain the robustness and reliability of the resulting registration. Therefore a lot of research has been done over recent years regarding TLS point cloud registration based only on natural scene content. An extensive description of this development is given in Section 2.2. Alternatively to markers, an interactive, semi-automated method has emerged as a further state-of-the-art approach. It also exploits the natural scene content and is quite efficient for simple or medium complex scenes. Coarse registration is thereby accomplished by manually picking at least three corresponding points between each scan pair. An automated refinement based on this resulting alignments yields the final relative orientations of the scans.

It is important to note that the challenge of point cloud registration appears in various research fields such as computer vision, geodesy, photogrammetry, or computer graphics. Although this has caused increasing progress towards the automation of point cloud processing (including registration), most approaches are not designed to work for TLS point clouds. The challenge of registering laser scans stems from the fact that the basic point clouds feature the following special characteristics which we need to take into account during processing. A more detailed explanation about these properties are given in Section 2.1.

- The sheer number of points per scan (typically many millions) calls for very efficient processing methods.
- The regular angular measurement principle of static laser scanners causes quadratically decreasing point densities with increasing distance to the scanner origin (referred to as *near-field bias*).

- As does every observation, TLS measurements suffer from uncertainties, i.e., outliers, noise and systematic errors.

Moreover, to be efficient, scan operators tend to reduce field time by minimizing the number of scans within a project. Therefore scan stations are set farther apart which directly yields small point cloud overlaps. Hence registration becomes harder because the number of potential correspondences reduces with decreasing overlap.

1.2 Research Goals

The overall aim of our research is to develop automated, efficient, and marker-less TLS point cloud registration strategies with focus on the applicability in real-world scanning projects. Developed methods should thus be reliable, robust, produce accurate alignment results, but still have practical runtimes. Moreover, we are not interested in absolute positioning of the scans in a higher level coordinate system, but remain on a local level by orienting scans relative to each other. Normally, an arbitrary scanner coordinate system is kept fixed and serves as the local reference coordinate system.

Typically the task of registering point clouds is divided into two steps: first one searches for an initial coarse registration which roughly aligns two or more point clouds relative to each other. Second these alignments are refined by alternating between the search for nearest neighbours and the minimization of the respective point-to-point distances until convergence is reached (cf. Section 2.2.3). The latter is a highly non-convex problem which thus tends to converge to local (but wrong) optimums. To ensure the convergence to the desired (correct) optimum, a preceding coarse orientation of the point clouds is therefore unavoidable. While there are numerous generally applicable fine registration methods (cf. Section 2.2.3), coarse registration is arguably the harder and less investigated task. Therefore, the main focus of this thesis is to develop methods for the automated, coarse alignment which then serves as the input for a state-of-the-art fine registration step. We thereby pay special attention to the unique characteristics of static laser scanning point clouds (described in Section 2.1.1).

Apart from this major goal of developing applicable registration methods, we are also interested in the following additional research questions:

- In this thesis, methods are presented which aim at the automated registration of TLS point clouds. A very important question is therefore whether the automation is justified or if manual registration conducted by an expert is already practical enough? What do we gain with an automated method in direct comparison to manual registration?

- A basic condition to register two point clouds is the availability of corresponding points. As already mentioned, decreasing the overlap between adjacent scans reduces the acquisition time because fewer scans are necessary to cover the same area. But how far can an operator go in minimizing this overlap until the developed registration methods fail? Is this breakpoint acceptable for real-world scan projects?
- The framework of Chapter 6 is a complete registration pipeline including the simultaneous fine alignment of multiple point clouds. How well does this refinement perform w.r.t. registration accuracy? Is this result justified taking into consideration the theoretical measurement accuracy of the employed laser scanners?
- We also address the effect of voxel grid filtering, typically applied in order to initially reduce the density of point clouds. How does this reduction influence the registration? What do we gain when working with these simplified, more evenly distributed point clouds?

1.3 Structure and Contributions

In accordance with the ETH Zurich Doctorate Ordinance of 2013, this thesis is written as a *cumulative dissertation*. The developed methods, improvements and results have been continuously presented in various publications which are directly used as the *core chapters* of this thesis (Chapters 3, 4, 5, 6). Apart from correcting spelling errors and adjusting abbreviations, as well as mathematical symbols for consistency, these chapters contain the original papers without changes to the content. For publishers' typeset versions of the papers, we refer to the original publications. The core chapters are complemented by Chapters 2 and 7. The former provide background information about TLS and a survey about the current state-of-practice as well as ongoing research on point cloud registration, while the latter includes the final remarks and an outlook. In the following, our main contributions with regard to TLS point cloud registration are summarized.

Contributions in Chapter 3. In a first attempt naturally existing planar surfaces are extracted from a 3D scene and exploited for pairwise point cloud alignment. The developed method is suitable to register scans acquired in man-made environments which typically feature large numbers of well defined planes (e.g. façades, streets). The main contributions can be summarized as follows:

- A method for the fully automated extraction of dominant planes from a 3D point cloud is proposed, where planes do not necessarily have to be connected. The extraction is based on randomly sampled points and does not require point normals.

- We present a new keypoint type as the virtual intersection of plane triplets encoded with the properties of the basic planar surfaces (e.g. extent, smoothness) and the characteristics of the geometrical intersection itself (e.g. intersection angle).
- A matching strategy to align two mutually overlapping scans is developed. The method combines geometric and descriptor information within a *Random Sampling Consensus* scheme (RANSAC; Fischler and Bolles, 1981).
- We successfully demonstrate the applicability of the registration scheme for an indoor laser scanning dataset, even in the presence of a high degree of symmetry.

Contributions in Chapter 4. In a second approach, a more generic coarse registration method is presented. It is based on adopting an efficient four-point matching strategy for the task of TLS alignment. The large laser scans are reduced to sparse clouds of 3D keypoints and matched solely based on geometrical information. The following aspects are thereby the main contributions:

- A highly efficient point-based matching strategy called *4-Points Congruent Sets* (4PCS; Aiger et al., 2008) is adapted for the task of TLS point cloud registration.
- We propose a tailored reduction of a scanner point cloud to a sparse cloud of 3D keypoints with high discriminative power suitable for point cloud registration. The reduction is based on voxel grid sampling to even out the point density (up to some degree) followed by a 3D keypoint extraction.
- The combination of the reduction of scans to sparse keypoint clouds and the efficient geometric point matching leads to a new method called *Keypoint-based 4-Points Congruent Sets* (K-4PCS) for pairwise coarse registration of point clouds. The resulting method does not require any assumptions about scene content or scan configuration, nor point descriptors, and is thus very robust against viewpoint changes.
- The efficiency and practicability of the developed K-4PCS method is demonstrated on various, real-world datasets with different challenges such as ambiguous scene geometry or large number of outliers.

Contributions in Chapter 5. The initial results of using 3D keypoints in combination with efficient geometrical point matching still suffers from some remaining problems. For example, highly complex scenes or low scan overlap cause large runtimes, and misalignments arise in the presence of ambiguous scene geometry (e.g. repetitive scenes, symmetrical structures).

These drawbacks are addressed in Chapter 5, where extensions and improvements on the

K-4PCS pairwise coarse registration are tested and discussed. We evaluate four different aspects with the aim to speed-up the matching process and to increase the quality of the underlying evaluation function used to define the best alignment. This yields the following contributions:

- Instead of only considering the overlap (i.e. ratio between total number of points and number of points with a close neighbour) in the evaluation function, we apply *M-Estimator Sample Consensus* (MSAC; Torr and Zisserman, 2000) which takes into account the residuals between corresponding points (truncated to make it robust against outliers).

- A soft prior term in the evaluation function is proposed to favour registrations with larger translations. This counters the near-field bias and accounts for the fact that scanners are often placed far apart, e.g., to cover larger areas.

- Apart from the generated speed-up of pairwise matching using straight-forward parallelization[1], the idea of soft voting to detect re-occurring alignments is investigated. The computational burden of the time-consuming evaluation of possible alignments could be reduced if only often appearing solutions have to be tested. However, we found that the underlying assumption of this approach does not hold for 4PCS matching.

Contributions in Chapter 6. Finally, the keypoint-based pairwise matching approach is extended to a generic, global (i.e. multi-view) point cloud alignment scheme which is able to automatically register real-world scan projects robustly and accurately. The framework includes state-of-the-art fine registration methods, but the main research focus is set on the coarse registration, resulting in the following major contributions:

- Global consistency is induced exploiting loop constraints within multi-view scan projects during coarse registration. Loop closures define higher-order constraints which constitute the basis of a discrete optimization problem. Assuming a number of possible pairwise alignments, energy minimization yields the globally best fitting registration per pair.

- We use K-4PCS as described in Chapters 4 and 5 to retrieve sets of possible pairwise coarse alignments. But the global consistency verification method is generic, thus any kind of pairwise hypothesis generator can be applied.

- A complete registration pipeline including coarse as well as fine registration of multiple TLS point clouds is presented. The framework is fully automated and only needs minimal, generally a priori, known information about the scene (e.g. approximated scan overlap).

[1]Note that the contribution of parallelizing the 4PCS matching method was originally presented in Theiler et al. (2014b) (see Chapter 5), but was also integrated in Theiler et al. (2014a) (see Chapter 4) during first revision.

- We show the results of the complete framework as well as the influence of global versus simple pairwise coarse registration on six demanding datasets. The proposed method is able to robustly register entire scan projects in practical runtimes.
- Additionally, one automatically receives information about final network stability due to the global consistency verification. This could be used to semi-automatically correct misalignments, if necessary.

1.4 Relevance to Science and Economy

The application fields of TLS sensors have rapidly increased over the last decade while at the same time the sensors have further evolved regarding measurement speed and accuracy. The main effect of this development is that more and more data needs to be processed while the demands regarding actuality of the data also increases. Both requirements, i.e., the handling of large data and the fast data analysis, are calling for efficient and thus automated point cloud processing tools. The approaches presented in this thesis are directly applicable to the task of automated point cloud registration and are thus highly relevant for science as well as economy. Although designed with special focus on TLS registration, the developed methods are generic and can in principle be applied to any type of point cloud as long as the task remains a six DoF problem (i.e. known scale). The results of the different approaches described in the core chapters were made available within the following publications:

[1] P.W. Theiler, K. Schindler, Automatic registration of terrestrial laser scanner point clouds using natural planar surfaces. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences **1**(3), 22nd ISPRS Congress, Melbourne, Australia, 2012

[2] P.W. Theiler, J.D. Wegner, K. Schindler, Keypoint-based 4-points congruent sets – automated marker-less registration of laser scans. ISPRS Journal of Photogrammetry and Remote Sensing **96**, 2014

[3] P.W. Theiler, J.D. Wegner, K. Schindler, Fast registration of laser scans with 4-points congruent sets – what works and what doesn't. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences **2**(3), Photogrammetric Computer Vision, Zurich, Switzerland, 2014

[4] P.W. Theiler, J.D. Wegner, K. Schindler, Globally consistent registration of terrestrial laser scans via graph optimization. ISPRS Journal of Photogrammetry and Remote Sensing **109**, 2015

The source codes of all developed methods were released, either on the web (see research homepage; Theiler), or in the case of the *Keypoint-based 4-Points Congruent Sets* (K-4PCS) algorithm as part of the open-source *Point Cloud Library* (PCL; Rusu and Cousins, 2011). The scientific community as well as the commercial sector have thus full access to the methods which allows to either continue the research and further improve methods, or to integrate them in commercial and non-commercial software. Also, the results of all presented experiments are reproducible and can thus be verified. In addition, the generically designed methods can be partially used and integrated in other pipelines.

The final framework (cf. Chapter 6) is able to robustly align real-world TLS datasets in practical runtimes (e.g. eight scans in below seven minutes), where even experienced users would have difficulties to perform the registrations faster. Moreover, the registration pipeline was successfully tested on datasets which have such a high scene complexity that manually identifying corresponding points becomes very challenging, or near impossible. The global consistency verification additionally allows the inclusion of a semi-automated control step within the framework, making it attractive for professional software. In particular, the registration produces additional information about uncertainties within the resulting scan network which could be used by an operator to control and, if necessary, correct alignments at specific network locations.

Chapter 2

Background

The main aim of the work described in this thesis is to develop methods for the automated registration of TLS point clouds. In the following, the relevant background will be explained, divided into two parts: first an overview on terrestrial laser scanning is given (cf. Section 2.1). More precisely, the basic measurement principle is explained, insights about the challenges of working with TLS data are provided, and an overview on modern TLS sensors is given. Second, the task of point cloud registration is explained in detail, i.e., the problem definition as well as a recap of existing methods to solve the task (cf. Section 2.2). In particular, state-of-practice marker-based methods as well as strategies exploiting the natural scene content are described. The final section addresses the various iterative closest point methods for fine registration.

2.1 Terrestrial Laser Scanning

3D laser scanning is the process of capturing the geometry of a surface by means of non-contact laser-based distance measurements. The geometry is recorded by moving the laser over the object of interest, finally yielding a discrete 3D representation of the surface, generally known as a *point cloud*.

There are various possibilities to classify laser scanning. Depending on the sensor platform one has to distinguish between TLS and *Airborne Laser Scanning* (ALS), where primarily the former is addressed in this thesis. Although designed for static TLS data, the developed methods are in principle also applicable to ALS point clouds, because registering such point clouds still amounts to a six DoF problem. Note that aerial scans are typically complemented by additional sensors (INS, GNSS) to directly provide initial orientations of the resulting points clouds. Another important difference is given by the sensor operation mode, more precisely, if the sensor is used in a *static* or *kinematic* mode. Kinematic TLS combines the ideas of a moving platform as given in airborne laser scanning with the high resolution and accuracy achieved with terrestrial scanners. Its main application field is mobile mapping. Again, registration

of kinematic laser scans is normally facilitated with the use of additional sensors (odometer, GNSS). Point cloud registration is thus considerably simplified because initial orientations are directly available.

This thesis aims at the registration of static terrestrial laser scanning point clouds without the aid of additional sensors. Compared to ALS or kinematic TLS the task is more challenging, because we do not have navigation information (and hence no initial orientation) and are typically confronted with large viewpoint changes between consecutive scans. The following sections give a selective overview on important aspects of static TLS for point cloud registration. Readers interested in airborne and mobile mapping applications, or in further information on TLS are referred to e.g. Vosselman and Maas (2010).

2.1.1 Measurement Principle

Terrestrial laser scanning is an optical 3D measurement technique. In contrast to passive systems such as cameras, a scanner actively emits a laser signal which is reflected by an object, and is again received by the sensor. Scanning of the surroundings is achieved by incrementally changing the direction of the emitted laser beam. Thus the geometry of objects is recorded and represented as a densely sampled 3D point cloud. For a static TLS sensor, which remains unmoved during acquisition (e.g. by mounting it on a tripod), the variation of the laser beam direction is accomplished using *(i)* two orthogonally placed rotating mirrors (camera view scanner) or *(ii)* a rotating mirror in combination with a mechanical movement of the instrument (panorama scanners). Panorama scanners have a larger field of view of up to $360°$ in horizontal, and up to $310°$ in vertical (relative to the scanner coordinate system). The vertical limitation is only caused by the scanner hardware itself.

The distances between sensor and surface are derived using either *Time-of-Flight* (ToF), *phase difference* of a modulated laser signal, or triangulation. ToF (or pulse) scanners use the time delay ΔT between emitted and received laser pulse together with the speed of light in vacuum c to calculate the distance to the object

$$d = \frac{c}{\nu} \cdot \frac{\Delta T}{2} \quad , \tag{2.1}$$

where ν accounts for the reduced speed of light waves if travelling through air. Note that this principle of distance measurement is similar to RADAR (*Radio Detection and Ranging*) except that it is based on laser (i.e. light) which led to the term LiDAR (*Light Detection and Ranging*), often used in the field of laser scanning. The second possibility to derive the object-sensor distance is based on the phase difference of an amplitude modulated laser signal. Given the

detected phase difference $\Delta\varphi$ and the wavelength of the modulated signal λ, we can derive the time delay

$$\Delta T = \frac{\Delta\varphi}{2\pi} \cdot \frac{\lambda}{c} \ . \tag{2.2}$$

Inserting Eq. (2.2) into Eq. (2.1) gives the distance d to the object. The third laser-based distance measurement principle is *triangulation*. In contrast to the other two principles, the distance is not derived from time-of-flight measurements, but from the position of the (focussed) laser response on a position-sensitive detector (e.g. CCD-array). The underlying idea is that the position of the returned laser on the detector changes with the distance to the object. With a given relative position between laser source and laser spot detector, absolute distances to the surface can be calculated.

Generally speaking, systems based on phase difference are very fast nowadays, with up to one million points per second, very accurate (typical range noise of $1 - 3$ mm), but have a limited operating range of some hundred metres (depending on the signal modulation). Such scanners are often used in mid-range applications like building inspection. In contrast, pulse laser scanners typically have a much larger measurement range of currently up to $6{,}000$ m but operate with lower frequencies. This makes them suitable for tasks such as large-scale deformation analysis or monitoring of entire landscapes. Note, modern pulse scanners can run in various modes to increase measurement frequency while reducing the maximum operating range. The distance accuracy is slightly worse and typically ranges between 5 mm to 10 mm. Triangulation-based scanners are designed for close-range applications (< 5 m), where triangulation surpasses the accuracy of phase-based and pulse-based range measurement. Typically, triangulation-based scanners are applied in industrial applications with limited operating range, but high accuracy requirements. The limited application range of triangulation-based systems stems from the fact that the distance accuracy depends on the triangulation angle, i.e., the ratio between base length and distance to the object.

In addition to the distance, the vertical and horizontal angle of an emitted laser beam is recorded via encoder discs (similar to tachymeters) or incremental encoder of the motor movement. For each emitted and received signal one finally obtains 3D polar coordinates defined by the distance and the two angles (horizontal and vertical in the scanner coordinate system). By transforming the polar coordinates into 3D Cartesian coordinates centred in the scanner origin, this finally yields a 3D point cloud, i.e., a discrete representation of the surrounding area. Most TLS sensors additionally record the intensity of the returned signal[1], resulting in 4D information per point. Instead of single laser intensities, recent developments attempt to gather multi-spectral or even hyper-spectral information, using dual- or multi-waveform

[1] in this context sometimes also called reflectance. We avoid this term because it is incorrectly used w.r.t. its meaning in physics.

laser scanners (e.g. Wei et al., 2012; Hakala et al., 2012). As a consequence, one receives for each measured 3D point additional information from the different spectral dimensions. Thus further knowledge about the the points' reflectance properties is gained, which opens up new possibilities in application fields such as agriculture (e.g. Gaulton et al., 2013). Additional information per point can also be gathered using integrated cameras or external cameras with known relative position to the scanner. Nowadays, there are numerous commercial laser scanners with integrated RGB-cameras. Other research projects are also combining scanners with multi- or hyper-spectral cameras as in Hemmleb et al. (2006).

A further development within the scope of pulse-based terrestrial laser scanning is the usage of the entire waveform of a returned pulse. Since around 2004, the *full-waveform* technology has been successfully used in airborne laser scanning (for a literature survey the reader is referred to Mallet and Bretar, 2009). The technology has recently also found its way into terrestrial applications (e.g. Doneus et al., 2009). The method works as follows: Instead of simply detecting the most dominant peak within the returned pulse to derive a single 3D point, the full, returned waveform is densely sampled and can be used (i) to detect further peaks caused by multiple reflections along the path of a single laser pulse or (ii) to derive further information about the physical backscattering characteristics of the measured surface. In the case of (i) one refers to such systems as multi-pulse scanners. Scanning results in a varying number of echoes along any given measurement direction, which are typically classified into "first", "intermediate", and "last" pulse points. Regarding (ii), not only the pulse maxima are of interest, but the shape of the returned echo itself (or of the various peaks, respectively). By analysing the waveform one can derive additional information such as surface orientation or roughness, which is especially useful to classify the points, e.g., to detect vegetation within a point cloud. Multi-pulse and full-waveform scanners are often applied in forestry (e.g. Blair et al., 1999; Reutebuch et al., 2005) or more recently also in urban areas (e.g. Jutzi and Stilla, 2005; Mallet et al., 2008).

2.1.2 TLS Point Cloud Characteristics

The result of a single static laser scan is a regular point cloud such as shown in Figure 2.1. *Regular* in this context means that the point cloud was generated using constant angular increments during scanning. As a consequence, a static TLS point cloud from a single station can be represented as a single panoramic range image without information loss.[2] The image axes of this panoramic representation are given by the measurement angles recorded during

[2] Assuming multi-pulse laser scanners, a loss-less representation of the 3D point cloud is still possible (i.e. with a multi-channel image), but the advantage of working in 2D is significantly reduced.

scanning, and pixel values represent the distances between objects and instrument (Figure 2.2a). Given the often recorded intensities of the laser response a further image can be derived (see Figure 2.2b). In the case of multi-spectral or hyper-spectral laser scanners, the representation becomes a multi-channel intensity image.

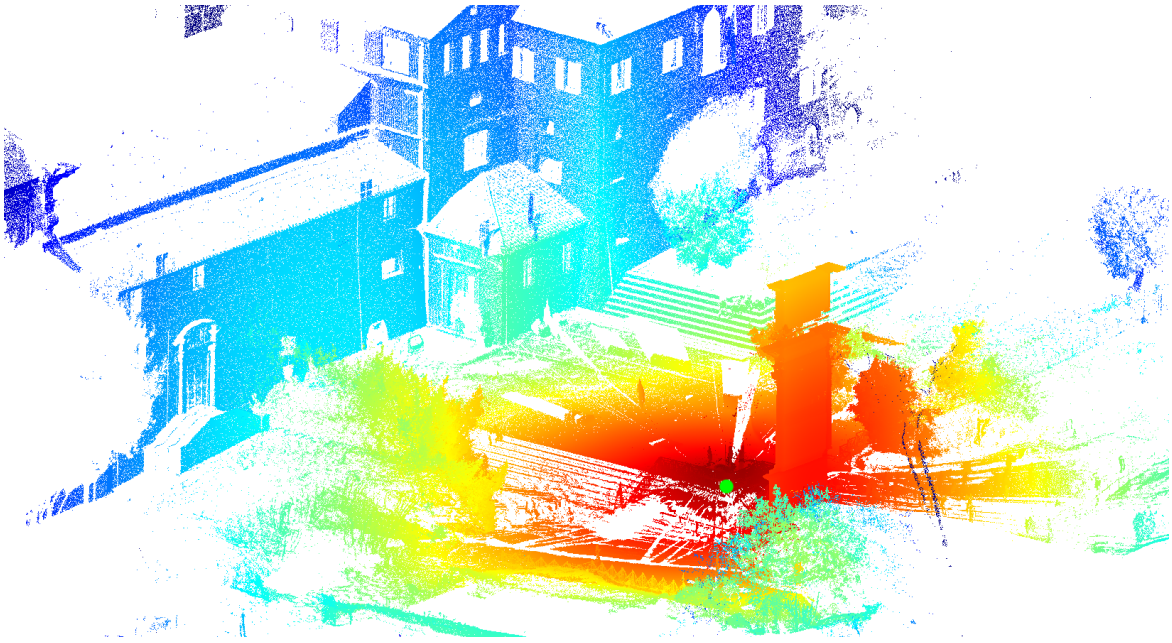
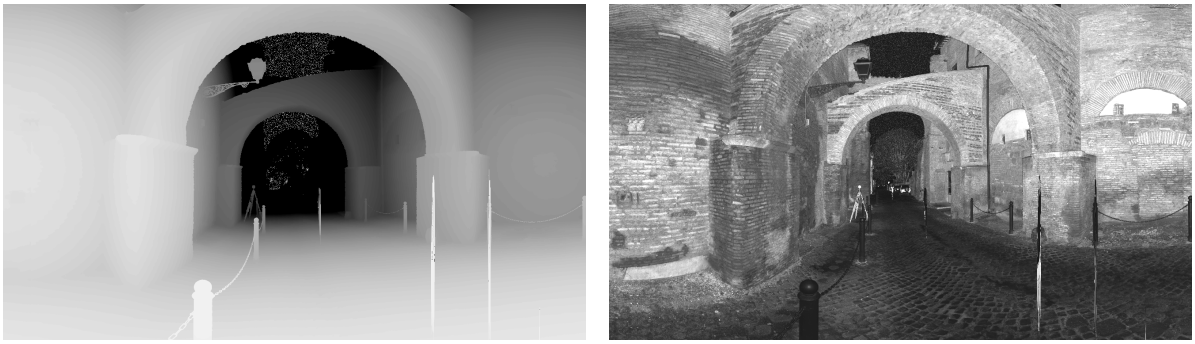


Figure 2.1: An example of a single scan point cloud. Colour values represent the range w.r.t. the scanning device; the green circle marks the instrument position.



(a) Range image (b) Intensity image

Figure 2.2: Assuming a single-pulse laser scanner, data acquisition with constant angle increments yields a regular point cloud which can be represented by grey-scale, raster images without information loss.

The decreasing point density with increasing distance to the scanning device becomes evident in Figure 2.1, e.g., on the floor. This effect is caused by the polar measurement principle of static laser scanning (i.e. regular angular increment steps). For example, an angular resolution of $0.02°$ leads to a nearest point distance of 3.5 mm at 10 m distance, while at 5 m we have a

nearest point distance of $1.7\,\text{mm}$. Note that the distance to the closest neighbour is linearly increasing with the distance, hence the point density is quadratically decreasing. In conclusion, the density and therefore the degree of detail is very high when close to the scanner, whereas objects further away are only sparsely recorded. We refer to this effect as *near-field bias*.

An important consequence of the polar measurement principle of TLS is the varying point resolutions on the same object recorded from different viewpoints. This makes it difficult for automated processes to identify corresponding objects in different point clouds. Furthermore with increasing distance (d) to the recorded point, the intensity of the returned signal decreases ($\sim\frac{1}{d^2}$) until the signal becomes undetectable. This results in even sparser point clouds at large distances. The maximum measurement distance is therefore not only dependent on the emitted signal, but also on the object's reflectance characteristics (e.g. material, surface orientation).

Additionally, one can also recognize data holes (shadows) caused by the line-of-sight measurement principle of static TLS, because only surfaces directly visible from the station are recorded. Thus multiple scans are generally necessary to record an object or scene of interest which in turn requires a posteriori registration of the resulting point clouds. Although it is not a main topic in this thesis, we give a brief summary on potential error sources which can occur within a TLS point cloud, because they potentially influence the registration result. As usual with measurement devices, error sources are divided into random noise, systematic errors, and gross errors (i.e. outliers).

Due to the fact that static TLS sensors are rather high-end instruments (compared to other active range sensors) they produce low random noise relative to the measurement range and resulting point density. Sadly, no standard has yet emerged to declare distance and angular accuracies of TLS sensors which makes comparisons difficult or even impossible. Moreover, angular accuracies are often not specified by the manufacturer. According to Vosselman and Maas (2010), typical angular accuracies are around 0.002° which leads to a lateral point accuracy of $1\,\text{mm}$ at $25\,\text{m}$. Distance accuracies are generally separated into a constant error term and a distance dependent value. Typical values are shown in Table 2.1. Systematic errors are perturbations with direct impact on the observations. Examples for such errors are instrument wobble or violation of the orthogonal axis constraints. There are several ways to model such effects and to remove them from the observations by calibrating the sensor (see e.g. Lichti and Licht, 2006). Also, there are systematic influences which can not be included within a general correction model and are thus not removed via calibration (e.g. refraction). Outliers are, for example, caused by non-static objects (e.g. vegetation, cars, people), or multiple reflections of the laser beam (also called *multi-path effect*). They may cause errors during registration, i.e., complete misalignments if the registration only depends on a small number of points including one or more outliers, or alignment inaccuracies during fine registration.

2.1.3 Employed Sensors

The number of available static laser scanners has increased significantly over the last decade. In Figure 2.3 an example selection of static TLS sensors is shown. The various scanners were developed with different focuses, for example w.r.t. weight and size (Faro Focus3D X130), measurement range (RIEGL VZ-4000), accuracy (Zoller+Fröhlich Imager 5006i), or in-field flexibility (Leica ScanStation P20).

(a) Z+F Imager 5006i (b) Faro Focus3D X130 (c) RIEGL VZ-4000 (d) Leica ScanStation P20

Figure 2.3: Examples of static terrestrial laser scanners.

	Z+F Imager 5006i	Faro Focus3D X130
Observation range	$0.4 - 79\,\text{m}$	$0.6 - 120\,\text{m}$
Measurement frequency	$508'000$ pts/s	$976'000$ pts/s
Field of view	$360°$ H and $310°$ V	$360°$ H and $300°$ V
Distance error (constant)	$\leq 1\,\text{mm}$ (up to $50\,\text{m}$)	$2\,\text{mm}$ (up to $25\,\text{m}$)
Distance error (scale)*	$1\,\text{mm}$ at $25\,\text{m}$	$1\,\text{mm}$ at $25\,\text{m}$
Angular resolution	$0.0018°$	$0.009°$
Instrument Dimensions	$30 \times 19 \times 37\,\text{cm}^3$	$24 \times 20 \times 10\,\text{cm}^3$
Instrument Weight	$14\,\text{kg}$	$5\,\text{kg}$
Release Date	2008	2011

Table 2.1: Properties of the employed TLS sensors. (*) the scale term of the distance accuracy decreases with smaller material reflectance and is not necessarily linear; the values presented here are for objects with 90% reflectivity.

All test datasets employed in this thesis are generated using either the Faro Focus3D X130 (Figure 2.3b) or the Zoller+Fröhlich (Z+F) Imager 5006i (Figure 2.3a). In Table 2.1 we show the important specifications of those two sensors. Both are panorama scanners and distance measurements are based on phase difference with amplitude modulation. As a result, the sensors have high measurement accuracies, ensure fast data acquisition, but have a low

observation range. Note that a direct comparison between the sensors is biased due to the difference of three years between the release dates.

To conclude, the two sensors have similar specifications. The Faro Focus3D X130 is smaller, lighter, has a larger observation range and a higher measurement frequency. In contrast, the Z+F Imager 5006i achieves better distance measurement accuracies. Both sensors additionally record the intensity of the returned signal. The internally available camera of the Faro Focus3D X130 was not used during data acquisition.

2.2 Point Cloud Registration

Terrestrial laser scanners are line-of-sight instruments and hence recording an object of interest from a single station is often not sufficient. To model complex objects or cover large areas, one needs to measure from different viewpoints. Each individual acquisition (i.e. scan) produces a point cloud defined in the scanner's local coordinate system. To combine the different point clouds, the transformation of the individual scans into a common coordinate system becomes necessary. The task of estimating the parameters of these transformations is called *registration*; also referred to as *orientation* or *alignment*.

This thesis addresses the task of registering TLS point clouds relative to each other, i.e., to align the scans within a local system normally defined by the coordinate system of one arbitrary scan. A special case is the relative registration of only two scans, often referred to as *pairwise registration*. Note, if the target coordinate system would be externally defined (e.g. a national coordinate system) registration is also called *georeferencing*. The alignments of scans within a higher level coordinate system requires external position information and thus additional sensors such as *Global Navigation Satellite Systems* (GNSS).

Because laser scanners measure absolute, metric distances the registration of a 3D point cloud is defined by a rigid-body transformation with six DoF, i.e., three translation and three rotation parameters. Throughout this thesis we exploit the idea of homogeneous coordinates to describe a 3D rigid-body transformation

$$\mathbf{T}(\mathbf{t},\mathbf{R}) = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\intercal & 1 \end{pmatrix} \quad , \tag{2.3}$$

where $\mathbf{R}$ is a 3×3 matrix representing the rotation and $\mathbf{t}$ the 3D translation vector. The length of the translation is denoted $t = \|\mathbf{t}\|_2$, and equals the distance between the scanner and the origin of our reference coordinate system, respectively.

The task of point cloud registration is typically carried out in two consecutive steps. The parameters of the transformation are first estimated and second applied to the source point cloud $\mathcal{S}$ to achieve alignment with the target scan $\mathcal{T}$. The latter is a straight-forward task and does not need further explanation. Generally speaking, the estimation of the transformation parameters $\mathbf{T}$ is accomplished by minimizing a distance measure δ between corresponding points $C = \{c_s, c_t\}$ of the source and the target scan. Note, to calculate the parameters of a 3D rigid-body transformation, at least three pairs of corresponding points are required.

$$\sum_{i,j} \delta(\mathbf{T} \cdot \mathbf{p}_i, \mathbf{q}_j) \rightarrow \min, \;\; \mathbf{p} \in \mathcal{S}, \;\; \mathbf{q} \in \mathcal{T}, \;\; i \in c_s, \;\; j \in c_t \tag{2.4}$$

Most often the Euclidean distance is used for δ, truncated to increase robustness against outliers. A standard approach to solve Eq. (2.4) is to apply unweighted total least squares for example using *Singular Value Decomposition* (SVD). Rotation is thereby derived from the Eigen vectors, while the difference between the centroids of the sets of corresponding points yields the translation vector (Arun et al., 1987). More diverse approaches are based on weighted least squares adjustment as in Gruen and Akca (2005), or quaternions to simplify the mathematical handling of rotations (Horn, 1987; Walker et al., 1991).

Given the various existing solutions to derive the transformation parameters, we can see that the critical step is finding the basic correspondences, i.e., feature detection followed by feature matching. Note that the term *feature* is used very generally in this thesis, e.g., it can mean points, parametric primitives, or even surface normals. A primary focus of current research is to automate the detection and matching of such features. We point out, that one can also accomplish correspondence estimation manually via point picking. With low or medium complex scenes this can be quite efficient, especially if implemented as a semi-automated process where the manual pre-alignment is automatically refined. Such tools are common in professional software.

In the next sections, an overview on different point cloud registration approaches is given. This includes a summary on important methods separated into the following parts:

- The state-of-the-art strategy to detect and match features is to use well-defined markers as described in Section 2.2.1.
- In Section 2.2.2 the possibility of exploiting the naturally available scene content without placing artificial markers is explained, which is a major topic in various research fields.
- Finally *Iterative Closest Point* approaches for fine registration are summarized as a special case of marker-less alignment, where matching is based on an iterative correspondence search (cf. Section 2.2.3).

Due to the cumulative structure of this thesis, more details on relevant approaches can be found in the respective core chapters.

2.2.1 Marker-based Registration

The registration of TLS point clouds based on markers is still the most commonly adopted method in scan projects realized by professional companies, especially in the fields of geodesy. The reason behind this is the robustness and accuracy one can achieve if suitable markers are employed. Examples of state-of-the-art markers are shown in Figure 2.4. An overview on the pros and cons, and a study about the achievable accuracy of various markers, is provided by Becerik-Gerber et al. (2011).

(a) Spheres (b) Retro-reflective targets (c) Checker-boards

Figure 2.4: Common markers used in TLS projects.

At least three markers placed within the overlapping area of two scans are necessary to align the respective point clouds. Typical markers are large enough to be redundantly recorded (i.e. many laser beams fall onto the signalized area). These multiple observations are used to derive a single, but highly accurate point (normally the centre of the marker). Because markers are generally simple geometric primitives, this reduction is straight-forward and will not be further explained.

The detection of the primitives (i.e. markers) within the point clouds is either carried out manually, semi-automatically or fully automatically. The task is simplified by the fact that most markers *(i)* feature high intensity contrast compared to their surrounding area, making them visible in the respective intensity images, and/or *(ii)* are clearly detached objects in the 3D point cloud and thus cause depth jumps in the range image. Manual detection is carried out by e.g. picking the centre point of a checker-board, while semi-automated approaches provide fine adjustment algorithms for accurate localization of the markers, but still need an initial marker position from user inputs. Examples for the automated extraction are proposed in Bornaz et al. (2002) for retro-reflective markers, in Akca (2003) for checker-boards, or in Franaszek et al.

(2009) for spheres. Modern commercial software often provides (semi-)automated extraction tools for the most common markers like checker-boards and spheres.

Matching of the detected markers is nowadays mostly carried out automatically. Since the number of markers is normally small, a brute-force correspondence search based on the relative distance to each other becomes feasible (assuming a unique distribution). Another option would be to manually label each marker with unique codes (numbers, names). Identical markers receive the same label across different scans which directly yields the required correspondences.

2.2.2 Marker-less Registration

The basic idea of marker-less point cloud registration is to exploit features derived from the naturally available scene content. This section addresses the idea to achieve an initial, coarse alignment based on these features.

The goal of *coarse registration* is to provide transformations which align the respective point clouds roughly, but accurate enough for the non-convex refinement (using more, or all points; cf. Section 2.2.3) to converge to the desired, correct alignment. This task is challenging, especially if one assumes to have no information about the initial position and orientation of the different point clouds. Additionally, the solution must be provided in "one shot" because no (globally) convex objective function for the task of coarse registration can be formalized and thus no continuous optimization (e.g. gradient descent, Newton's method) can be carried out. One can say that the task of coarse, marker-less alignment is currently a major research topic in the field of point cloud processing.

In the following we distinguish between primitive-, image-, point- and normal-based registration schemes. Note that this classification is not in all cases optimal, because some published methods are based on combined approaches (e.g. image-based followed by point-based registration). At the end of the section a general overview on feature matching is provided.

Primitive-based. Registration strategies based on parametric primitives are to some degree comparable to marker-based approaches, but only naturally available primitives are used as features. Depending on the application, such objects are for example planes in man-made environments (e.g. cities), or cylinders to model pipes in industrial scenes. We point out, that for special tasks exploiting primitives as features can be advantageous, because one simultaneously achieves point cloud registration and CAD-level geometric reconstruction. Again, one has to find at least three corresponding primitives in the overlapping area to relatively align two point clouds. The detection of these parametric primitives can be carried out semi-automatically, by selecting the points corresponding to a feature followed by an automated modelling of a

specific primitive. But the amount of approaches presented over the last years clearly shows a trend towards automated detection of primitive features.

There are numerous proposed strategies which exploit planes for the task of point cloud registration (e.g. He et al., 2005; Dold and Brenner, 2006; Brenner et al., 2008). Our own primitive-based method (Theiler and Schindler, 2012) also uses planar surfaces, but solves the matching in conjunction with a point-based approach and with the help of a feature descriptor (cf. Chapter 3). Rabbani et al. (2007) have proposed a method which uses a combination of different primitives typically present in industrial environments. Planes, spheres, cylinders and tori are automatically detected in the scans and matched across point cloud pairs.

Image-based. The fact, that a regular TLS point cloud can be represented by range images, and in the case of recorded LiDAR intensities additionally with intensity images (cf. Section 2.1.2), constitutes the basis of image-based point cloud registration. Note that the known geometry of the scan images makes it possible to switch between 2D and 3D points without information loss. The underlying idea of image-based approaches is to extract out of the large images a sample of points/pixels (also called *keypoints*, *tiepoints*, or *feature points*) with high discriminative power and which are thus suitable for matching.

Well known keypoint extractors from image processing are *(i)* high curvature detectors such as *Difference-of-Gaussians* (DoG; Lindeberg, 1993; Lowe, 1999) or *(ii)* corner detectors based on the second-moment matrix (e.g. Förstner, 1986; Harris and Stephens, 1988; Shi and Tomasi, 1994). The typically recorded laser intensities make it possible to apply keypoint detectors on intensity images (Roth, 1999; Bendels et al., 2004; Böhm and Becker, 2007; Kang et al., 2009). Alternatively, the same interest operators can also be employed using range images (Barnea and Filin, 2008). Additionally, modern scanners have an integrated camera and thus yield coloured point clouds and a respective RGB-image representation on which keypoint detectors can again be applied (e.g. Zeisl et al., 2013). We point out, that in this thesis we do not assume to have colour information, because cameras are not generally an integral part of a TLS sensor system. And even if a camera is available, images are often not collected due to the additional acquisition time and their rather bad colour quality. It can be expected that in the future the number of commercial laser scanners with integrated cameras will increase, while the image quality will improve as well. Potential ways to adapt to this trend and to exploit colour information within the presented methods are mentioned in the outlook (see Section 7.3).

As known from image matching, the extracted 2D keypoints can be encoded by descriptors such as *Scale Invariant Feature Transform* (SIFT; Lowe, 2004) or *Speeded Up Robust Features* (SURF; Bay et al., 2006). However, matching solely based on descriptors lacks robustness against strong viewpoint changes.

Point-based. The basic idea of point-based alignment is similar to image-based registration. One samples the large point clouds by extracting a sparse but highly discriminative cloud of keypoints, which serves as the basis for matching. To increase the robustness against viewpoint changes, 3D keypoints are directly extracted from the point clouds. This is mostly accomplished by adopting ideas from 2D interest operators into 3D. An example is the 3D version of the DoG keypoint detector as presented e.g. in Flitton et al. (2010), or the 3D *Harris* corner detector as in Sipiran and Bustos (2011). Typically, grey values of a 2D image are for example replaced by laser intensities or range information, image gradients by intensity differences or local variations of the point normals within a 3D neighbourhood.

The concept of 3D keypoint descriptors has also been investigated, either by adapting known 2D descriptors into 3D such as THRIFT (3D version of SIFT; Flint et al., 2007), or developing descriptors specifically designed for the 3D space. For the latter case, probably the first approach was presented by Johnson and Hebert (1999) who introduced *Spin Images*. More recent methods are the *Point Feature Histogram* (PFH; Rusu et al., 2008), its faster version called *Fast Point Feature Histogram* (FPFH; Rusu et al., 2009), or the *Intrinsic Shape Signatures* (ISS; Zhong, 2009). Recent overviews on the different 3D keypoint detectors and descriptors and their performance for point cloud registration in real-world datasets are given in Tombari et al. (2013) and Hänsch et al. (2014).

It is important to note that there are methods where feature detection is not based on interest operators, but rather on randomly selected points (e.g. Aiger et al., 2008; Leng et al., 2014).

Normal-based. Very generic approaches are describing the complete surface e.g. based on salient directions of normals within the point cloud as in Makadia et al. (2006) or Novák and Schindler (2013). Salient directions are also exploited in Zeisl et al. (2013) to normalize the RGB image feature descriptors.

Feature Matching. The different approaches to carry out feature matching can be summarized as follows: if only a sparse amount of features is present, a geometrical constraint brute-force matching using point-point distances is feasible. Assuming a unique distribution of features, the relative metric distances can be used to find the biggest congruent subset which is considered to represent the correct matches. This general strategy can be enhanced by adopting additional information such as feature descriptors. Hence one is able to either match keypoints relying on descriptor distances, or to combine descriptor-based and geometrically constrained matching. Note, the problem of stand-alone descriptor-based matching is that appearance descriptors derived from 2D images lack robustness with strongly varying viewpoints, while descriptors based on geometrical properties within a 3D neighbourhood strongly depend on the local point density.

Alternatively, efficient RANSAC-based strategies exist, where a random sample of (at least three) features are picked, matched and used to derive the parameters of the transformation. Evaluation of the resulting alignment is then again based on the complete feature set (or even the whole point cloud). Random sampling and evaluation is repeated until a feasible solution for the registration is found. RANSAC not only increases time-efficiency due to working with a smaller number of point samples, but is also more robust even in the presence of a large number of outliers.

2.2.3 Fine Registration

The general idea of *fine registration* is to minimize the distance between corresponding points in two or more point clouds and as such follows the goal described in Section 2.2 and defined in Eq. (2.3). Let us first address pairwise refinement approaches, which are generally termed *Iterative Closest Point* (ICP) methods. Assuming roughly registered point clouds, a good estimation of correspondences is given by the nearest neighbours between the coarsely aligned source scan and the target point clouds. ICP methods proceed as follows:

(1) Apply nearest neighbour search (based on kD-tree or alternatives) within the transformed source and the target point clouds to estimate the point-to-point correspondences.

(2) Calculate an update of the transformation parameters based on minimizing the distance between corresponding points.

(3) Check for convergence, e.g., based on the *Root Mean Square Error* ($RMSE$), and if not reached, repeat process starting with (1).

Due to the fact that this optimization strategy is highly non-convex, the preceding coarse alignment of the point clouds is crucial to ensure the convergence of ICP towards the global optimal solution.

The idea of iterative alignment refinement using nearest neighbours was first presented by Besl and McKay (1992) and Chen and Medioni (1992). The main difference between these two methods is how residuals between the point clouds are calculated. In Besl and McKay (1992) they are directly derived from point-to-point distance between nearest neighbours, while in Chen and Medioni (1992) the point-to-plane distance is exploited. Since these original versions of ICP, there have been numerous variants proposed. We refrain from explaining the different variants and refer instead to the overviews and performance studies given in Rusinkiewicz and Levoy (2001) and more recently in Pomerleau et al. (2013) with respect to real-world applications.

An interesting advancement is the work done by Williams and Bennamoun (2001), who extended the ICP registration scheme for *global* (or *multi-view*) alignments, i.e., the simultaneous registration of multiple point clouds. A related method for a 2D localization of a scanning robot was presented some years earlier by Lu and Milios (1997), and was later adapted to 3D by Borrmann et al. (2008). With global registration methods, the error accumulation of consecutive pairwise alignment is eliminated (or at least reduced), assuming that we have some kind of constraints over multiple point clouds (i.e. overlap across more than two scans).

Chapter 3

Plane-based Point Cloud Registration

Automatic Registration of Terrestrial Laser Scanner Point Clouds using Natural Planar Surfaces

Pascal Willy Theiler, Konrad Schindler

ISPRS Annals, 22nd ISPRS Congress, Melbourne, Australia, 2012.

(Author version; for publishers' typeset version please refer to the original conference paper)

3.1 Abstract

Terrestrial laser scanners have become a standard piece of surveying equipment, used in diverse fields like geomatics, manufacturing and medicine. However, the processing of today's large point clouds is time-consuming, cumbersome and not automated enough. A basic step of post-processing is the registration of scans from different viewpoints. At present this is still done using artificial targets or tiepoints, mostly by manual clicking.

The aim of this registration step is a coarse alignment, which can then be improved with the existing algorithm for fine registration. The focus of this paper is to provide such a coarse registration in a fully automatic fashion, and without placing any target objects in the scene. The basic idea is to use virtual tiepoints generated by intersecting planar surfaces in the scene. Such planes are detected in the data with RANSAC and optimally fitted using least squares estimation. Due to the huge amount of recorded points, planes can be determined very accurately, resulting in well-defined tiepoints. Given two sets of potential tiepoints recovered in two different scans, registration is performed by searching for the assignment which preserves the geometric configuration of the largest possible subset of all tiepoints. Since exhaustive search over all possible assignments is intractable even for moderate numbers of points, the

search is guided by matching individual pairs of tiepoints with the help of a novel descriptor based on the properties of a point's parent planes. Experiments show that the proposed method is able to successfully coarse register TLS point clouds without the need for artificial targets.

3.2 Introduction

Terrestrial Laser Scanners (TLS) are increasingly used the last ten years, in a continuously growing number of applications ranging from cultural heritage documentation, surveying, industry and manufacturing to medicine. Many applications have the goal to generate high-quality geometric models. The processing of the resulting large datasets with billions of points is time-consuming, cumbersome and not automated enough. Due to the fact that TLS is a line-of-sight instrument with limited range of coverage, most applications require a series of scans to obtain a complete model. Hence, the scans need to be registered in a common coordinate system.

3.3 Related Work

To speed up the registration process, numerous approaches have been published over the last couple of years, which aim at the automatic registration of two or more point clouds. In this section we review the state of the art.

The registration amounts to finding the relative orientation of two or more point clouds. If the scan orientations are known from external sensors (e.g. from GNSS and INS), the problem does not arise. Such direct georeferencing is often applied in mobile laser scanner applications (e.g. Asai et al., 2005). Due to the uncertainty of the additional sensors, the resulting registration is often rather coarse. Furthermore, the costs for the sensor system increase.

The standard technique to achieve a coarse registration without additional sensors is to place artificial targets in the scene and identify them in different scans. The targets are typically objects, which are invariant to the scanner viewpoint (e.g. planes, spheres, cones). Often, they are retro-reflective to simplify their automatic recognition (e.g. Bornaz et al., 2002). Many commercial software packages support automatic registration with reflective targets. A related approach finds non-reflective spheres (Franaszek et al., 2009).

To avoid the need to place targets in the scene, a logical next step is to base the registration on natural geometric elements in the scene such as points, lines or surfaces (Goshtasby, 2005). Thus, the main challenge is to reliably extract and match corresponding features in different

scans. Roth (1999) describes a method based on point features, which are extracted from the laser intensity image with an interest operator and transferred to 3D using the range information. The matching is accomplished by an exhaustive search for congruent tiepoint triangles.

Seo et al. (2005) investigate the possibility to apply standard interest point operators to register laser point clouds. Similar approaches are presented in Bendels et al. (2004) and Böhm and Becker (2007). They apply the SIFT operator (Lowe, 1999) on the intensity image to find adequate tiepoints. Likewise, Moldovan et al. (2009) use intensity information and the SIFT operator to register multiple scans. To further improve the registration with the SIFT operator Mateo and Binefa (2009) first extract dominant planes in the scan and apply the interest point operator on them. Plane extraction is done by calculating the local normal for each point with singular value decomposition. Robust matching of the extracted point features is often achieved with RANSAC (Fischler and Bolles, 1981). Kang et al. (2009) present an approach based on pixel-to-pixel correspondence in the intensity image, followed by outlier detection and computation of the transformation parameters in the 3D space.

Instead of exploiting the intensity data, Basdogan and Oztireli (2008) apply a geometric descriptor, which is based on the distance from each point to the centre of mass of its neighbours, at every single 3D point. Thus, the method is only suitable for small point clouds. Barnea and Filin (2008) present a tiepoint extractor by applying a 3D corner detector to the range image. Johnson and Hebert (1999) use *Spin Images* as descriptor to find correspondences. This descriptor is generated by computing a local basis at an oriented point (3D point with local surface normal), and storing the resulting two-parameter description of nearby points to describe the local geometry. Shan et al. (2004) improves that approach with geometric constraints between simple configurations of multiple tiepoints.

He et al. (2005) propose an approach to register range images using *Complete Plane Patches*. The main idea is to extract planar surfaces and pick out only those, which are complete (not occluded). Correspondence is established using an interpretation tree and several constrains. Dold and Brenner (2006) have developed a similar approach based on planar patches found by region growing. Patches are matched using their area, boundary length, bounding box and mean intensity value. Additionally, an on-board image sensor is used to improve the registration. Instead of constraining the matching based on the plane properties, Brenner and Dold (2007) use the intersection angle to prune the number of possible plane triples to calculate the coarse transformation.

The approaches described so far are primarily designed for coarse registration. The results are then polished with a fine registration algorithm. The most established method is the *Iterative Closest Point* (ICP) algorithm (Besl and McKay, 1992; Chen and Medioni, 1992).

The algorithm iteratively minimizes the distances of all points in one scan to the nearest point or plane in the other. There are many variants and extensions of the initial algorithm (e.g. Masuda and Yokoya, 1995; Bergevin et al., 1996; Bae and Lichti, 2004), aimed to increase computational efficiency, robustness, convergence etc. ICP is a local optimization scheme and requires a good initial approximation of the transformation. An alternative is *Least Squares 3D Surface Matching* (LS3D; Gruen and Akca, 2005). Again, the method finds a local optimum and needs good initial values to converge to the correct solution.

To summarize, the crucial step is the coarse registration, whereas the refinement of an approximate registration can be considered solved.

3.4 Proposed Method

The goal of our method is a coarse registration of two terrestrial laser point clouds without the need for artificial objects. The principle workflow is shown in Figure 3.1.

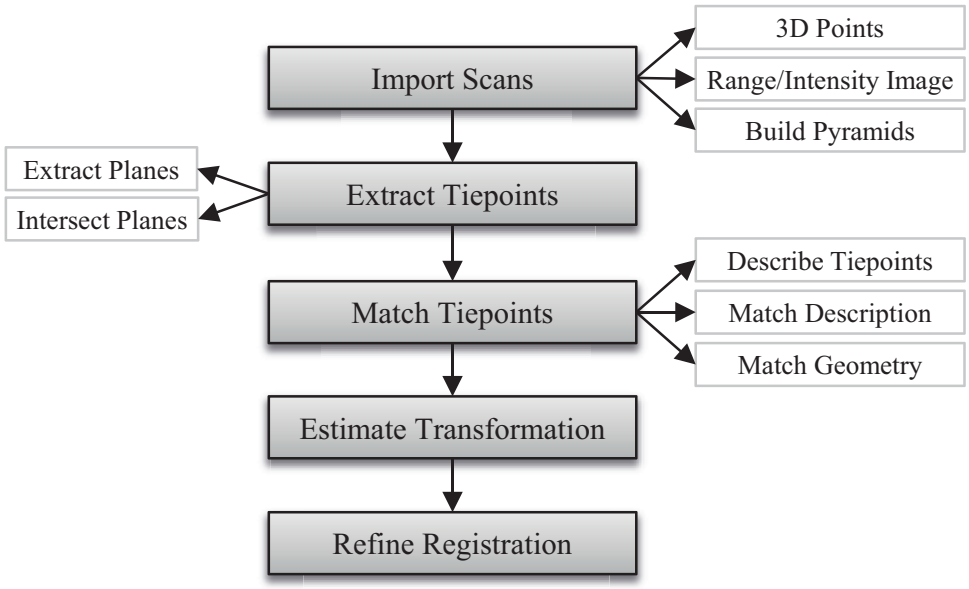


Figure 3.1: Workflow of proposed method for autonomous coarse registration of terrestrial laser scans.

The focus of the paper is on tiepoint extraction and matching. To complete the registration process, a fine registration with a standard algorithm (e.g. ICP, LS3D) is necessary. As the goal is coarse registration, the paper focuses on pairwise registration (i.e. two scans), which is sufficient to generate approximate transformations for further processing.

Our method is based on *virtual tiepoints* which are generated by intersecting triples of scene planes. The planes are detected with RANSAC, embedded in a multi-scale pyramid, which on the one hand increases the chance to find the dominant planes and on the other hand reduces the computation time. The reasoning behind the apparent detour of generating virtual tiepoints – rather than directly matching the planes – is the following: plane matching so far proved to be rather unreliable, because the geometric properties of planar segments tend to vary a lot across viewpoints. Points, corresponding to triples of intersecting planes, have additional geometric invariants, which allow for more powerful local descriptors.

Matching based on geometric constraints between tiepoints is very reliable, but of combinatorial complexity and thus intractable for useful numbers of tiepoints. On the contrary, matching points individually using local descriptors is efficient, but error-prone and often ambiguous. We thus opt to combine the two steps in a two-stage procedure to get the best of both worlds. First the combinatorial set of putative correspondences is filtered with a novel geometric descriptor, by discarding the large majority of correspondences whose descriptors are very different. Then, rather than taking final decisions based on descriptors, matching within that reduced set of candidate matches is accomplished by searching the largest subset, for which all point-to-point distances are consistent between the two scans.

3.5 Tiepoint Extraction

The tiepoints are virtual 3D points, which are generated by intersection of three non-parallel scene planes. These planes are extracted from the scan data by means of RANSAC on a multi-scale pyramid. Individual laser scans are represented as range images, which due to the polar measurement principle of the scanner does not entail any loss of information. For each scan pyramids are created by repeated down sampling of the range image by a factor of 0.5. For our purposes it is more important to avoid smoothing over range discontinuities, whereas geometric aliasing is not a concern, thus we use nearest neighbour resampling. The required number of pyramid levels depends on the point density and on the geometry of the scene – denser scans and larger number of dominant planes require more levels.

Plane extraction starts at the highest pyramid level (i.e. the one with the smallest range image, respectively the lowest 3D point density). Planes are found by iterative RANSAC. To increase the chance of finding correct planes we constrain the random sampling to points within a maximal radius. As usual, the parameters of detected planes are re-estimated with all inlier points. Laser scans have an absolute scale, such that thresholds can be specified in metric world units.

With a set of points (three or more) plane fitting is accomplished in a total least squares sense by estimating the normal vector with *Singular Value Decomposition* (SVD). Given the normal vector $\mathbf{n}_i$ the orthogonal distance to the origin $d_{i,0}^{\perp}$ is found by projecting the 3D points' centre of mass $\bar{\mathbf{x}}_i$ onto the normal vector, $d_{i,0}^{\perp} = \bar{\mathbf{x}}_i^{\mathsf{T}} \cdot \mathbf{n}_i$.

We point out that in range scans the point density in world coordinates decreases with the distance from the scanner. The threshold Λ for the minimum support of a plane is thus adapted to the range. Given the total number of points in the scan n, the current pyramid level ξ and a user-specified proportion κ, as well as the mean range of the scan $\|\overline{\mathbf{r}_0}\|$, the threshold for a plane with the mean range $\|\overline{\mathbf{r}_i}\|$ is computed as

$$\Lambda_i > \Lambda = n \cdot \frac{\kappa}{\xi} \cdot \frac{\|\overline{\mathbf{r}_0}\|}{\|\overline{\mathbf{r}_i}\|} \quad . \tag{3.1}$$

Planes which fulfil the condition are accepted. When no further planes can be detected, the extraction continues on the next lower pyramid level. The scalar κ specifies what fraction of the total scan a plane shall minimally cover. The appropriate value depends on the size of the planes in the recorded scene. The linear scaling with the pyramid level adapts the threshold to the point density of a given level while still biasing the extraction process towards dominant planes.

Figure 3.2 shows an example scan with the detected planes coded by different point colours. The same result in a 3D view is presented in Figure 3.3, with the ceiling plane removed for better visibility. The figure demonstrates the completeness and correctness of the plane extraction – almost all planes of reasonable point count have been found, with only few spurious planes.

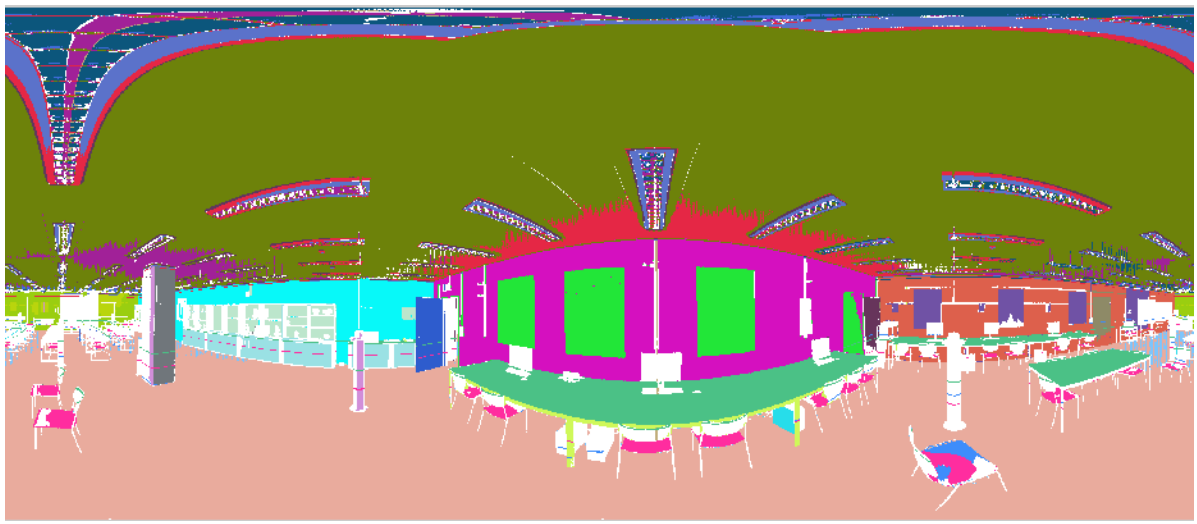


Figure 3.2: Panorama image with the extracted planes in an indoor environment.

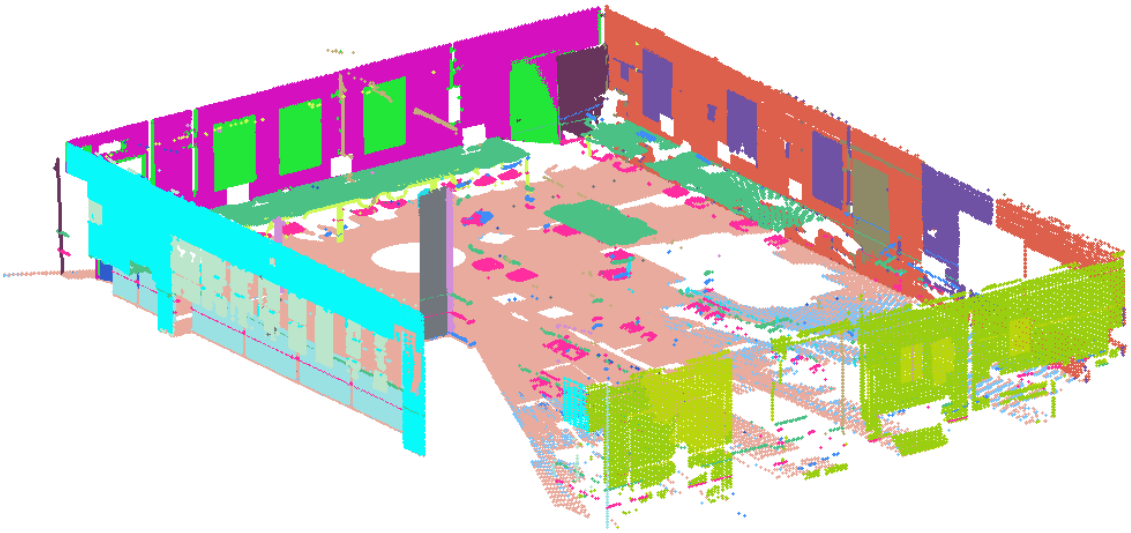


Figure 3.3: Extracted planes in the 3D space in an indoor environment (without the ceiling).

For all triples of detected planes, the intersection point $\mathbf{p}(u, v, w)$ is computed analytically to yield a virtual tiepoint.

$$\mathbf{p}(u,v,w) = \begin{pmatrix} \mathbf{n}_u^\intercal \\ \mathbf{n}_v^\intercal \\ \mathbf{n}_w^\intercal \end{pmatrix}^{-1} \cdot \begin{pmatrix} d_{u,0}^\perp \\ d_{v,0}^\perp \\ d_{w,0}^\perp \end{pmatrix} = \mathbf{A}^{-1} \cdot \mathbf{b} \tag{3.2}$$

The quality of each tiepoint is calculated using the reciprocal condition number of the matrix $\mathbf{A}$. A well-conditioned matrix indicates that the planes are not near-parallel and all intersection angles are large enough. Consequently the tiepoint is well defined. Very badly conditioned tiepoints are discarded.

The order of the parent planes is based on the z-value of the plane normal. Ambiguous order was treated by taking all solutions as possible tiepoints.

3.6 Tiepoint Matching

Laser scans have an absolute Euclidean scale and thus distances between tiepoints are directly comparable across scans. For any two pairs of corresponding tiepoints the point-to-point distances in both scans must therefore be the same. This geometric constraint is exploited in our scheme. Scan matching is formulated as finding the largest possible set of correspondences, for which all pairwise distances are the same (up to noise). This is an instance of the maximum clique problem, which is NP-hard. Even though there are faster approximations, the problem

is computationally too expensive for large point sets. To make it tractable, we therefore conservatively prune the exhaustive set of possible matches with a novel descriptor before geometric matching, which greatly reduces the number of putative correspondences and hence the search space for the maximum clique.

3.6.1 Construction of the Description Vector

For each tiepoint we construct a descriptor vector, which encodes local properties of the point, respectively the planes from which it has been constructed. The vector has the following entries:

- Reciprocal condition number (1 value)
- Intersection angles between planes (3 values)
- Extent of planar segments (6 values)
- Smoothness of planes (3 values)

We go onto explain the individual entries of the 13-dimensional descriptor in more detail.

Reciprocal Condition Number. This scalar is already calculated when intersecting the three planes. It encodes the geometric quality of the intersection.

Intersection Angles. The three parent planes of a tiepoint give rise to three pairwise intersection angles, calculated simply as the scalar product of the two unit normal vectors. By convention we use the smaller of the two intersection angles and divide angles by $\frac{\pi}{2}$, such that the values are between 0 and 1.

Extent of Segments. The tiepoints' parent planes are constructed by fitting a set of 3D scan points found with RANSAC. For each plane we calculate the bounding rectangle of those points. The width and height of the rectangle are found by principal component analysis in the 2D coordinate system of the plane. Outliers outside the three-sigma interval are ignored. To obtain values between 0 and 1, the width and height values are divided by the maximum possible range, which equates to twice the maximum measurement range of the scanner.

Plane Smoothness. Although all planes are extracted with the same absolute threshold on residuals (in our experiments 1 cm), their average residuals are quite different due to varying material properties of the planar surfaces. To preserve that information, the mean residuals of the three plane fits are added to the descriptor. The values are scaled by the inlier threshold to map them to the range $[0 \ldots 1]$.

3.6.2 Descriptor-based Pruning

So far the relative scaling of the descriptor entries is arbitrary. To obtain meaningful similarities between descriptors, the right relative scaling between the descriptor dimensions is needed. We have experimentally determined suitable weights.

The experiments showed that the most reliable values are the intersection angles and the reciprocal condition number. This can be explained by the fact that the angles and thus the condition number depend only on the quality of the normal vectors, which are very accurate due to the large number of points used to calculate them. On the other hand, the extent of the planar segments depends on the viewpoint and the associated occlusions. The smoothness is also less reliable, because it is sensitive to the range – more distant planes with the same surface properties are noisier. A possible solution would be to normalize the smoothness value with the mean range value of a plane. However this has not yet been tested, and is left for future work. In Table 3.1 the empirical weights are shown which we have used in our experiments.

Property	**Weight**
Reciprocal Condition Value	10
Intersection Angle	100
Plane Extent	1
Plane Smoothness	5

Table 3.1: Weighting of the description vector.

After applying the weights, Euclidean distances are used for descriptor matching, and all matches below a threshold are declared putative correspondences. Note that, contrary to most descriptor-based schemes, we only prune very dissimilar matches. We do not attempt to disambiguate matches below the threshold based on their descriptors, since in our experience the descriptor distances are too unreliable to do so and vary across different scanning environment. A variable threshold is used for accepting candidate matches, and adjusted such that the number of putative matches is not too high for the subsequent geometric verification.

3.6.3 Geometric Constraint Matching

The Euclidean distance between tiepoints is invariant across scans and can thus be used to reject false correspondences. Finding the largest possible subset of the reduced pool of matching candidates is a maximum clique problem, which has exponential complexity. Intuitively speaking, it can only be solved to global optimality by (nearly) exhaustive search. At present

we solve the problem with a greedy multi-start heuristic. Although it works well, we are planning to switch to a more sophisticated approximate optimization scheme in the future.

The current scheme works as follows. For all pairs of candidate matches the point-to-point distances are computed in both scans and compared to a threshold (in the experiments $10\,\text{cm}$). Pairs which do not exceed the threshold are flagged as compatible, resulting in a symmetric binary matrix of pairwise compatibilities. Each candidate match in turn (i.e. each line in the matrix) is chosen as seed point, and all candidates are removed for which the point-to-point distances to the seed are incompatible. In the remaining candidate set the one with the smallest number of compatible matches (i.e. the smallest number of '1' entries) is removed, and that step is iterated, until no more incompatibilities remain.

The cliques of compatible matches for every seed are sorted by decreasing size (number of correspondences), and starting from the biggest one, the rigid transformation is estimated. If the residuals of the transformation are above a threshold, the clique is discarded and the next smaller one is tested, until a valid transformation has been found. The last step is required since testing pairwise distances does not account for long-range error accumulation. The output of the scheme are two sets of tiepoints in the two scans, which have the same (high) cardinality and the same geometric configuration, i.e., they can be matched without geometric contradictions. Figure 3.4 shows a subset of matched tiepoints. While some points are purely virtual, others correspond to existing objects (e.g. room corners).

3.7 Experimental Evaluation

3.7.1 Test Setup

The proposed method has been tested with an indoor dataset consisting of four scans. The rectangular room ($\approx 15\,\text{m} \times 10\,\text{m}$) has desks and tables, chairs, round pillars and some whiteboards on the walls. The scans have been acquired with the *Zoller+Fröhlich Imager 5006i* and have a field of view of 360° horizontal and 150° vertical. Each scan consists of 2.7 million points (2502×1076).

The following parameters were used: five pyramid levels for plane fitting; outlier threshold $1\,\text{cm}$; minimum plane cardinality 0.1% of all scan points; minimal reciprocal condition number to accept a tiepoint 0.1; maximum number of candidate matches after descriptor matching 5000 (corresponding to descriptor distances between 0.01 and 0.001); threshold for compatible geometric distances $10\,\text{cm}$; maximum allowable mean residual of rigid transformation $10\,\text{cm}$. Except when testing the noise sensitivity, these parameters were left unchanged.

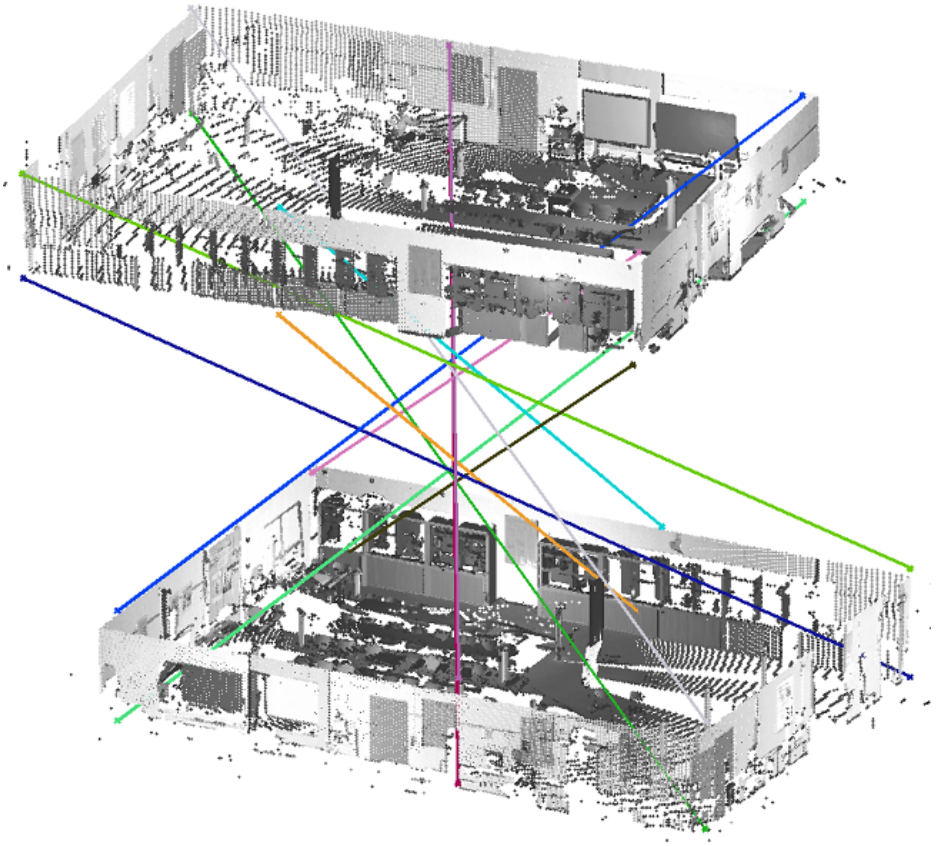


Figure 3.4: Set of matched tiepoints in two scans.

3.7.2 Success Rate

The success rate has been computed for all pairs of scans in the test dataset, see Table 3.2. All experiments were repeated 50 times with the same parameters, because of the randomness induced by RANSAC plane fitting. The estimated transformation parameters were compared to manually registered ground truth.

	s1	s2	s3	s4	**Total**
s1	—	86%	80%	88%	**87%**
s2	84%	—	100%	88%	**81%**
s3	82%	100%	—	100%	**94%**
s4	86%	88%	98%	—	**91%**
Total	**84%**	**91%**	**93%**	**92%**	**90%**

Table 3.2: Success rate of proposed method using test sets.

As shown in Table 3.2, 90% of the scan pairs were correctly matched. Most of the mistakes are due to the high degree of symmetry of the scene: regarding only the main walls, several registrations are equally correct. The remaining dominant planes (e.g. tables) mostly reduce the ambiguity to two cases, which differ by a rotation of $180°$ around the vertical axis.

Further tests were performed to assess, how the success rate depends on the number of extracted planes. In Figure 3.5, the minimum, maximum and mean number of extracted planes in two scans is visualized. As expected, with more planes, the number of successful registrations increases, at the price of increased computational cost. Note the timing is based on an unoptimized *Matlab* implementation on a current eight-core machine including the whole working process (from import until the resulting transformation).

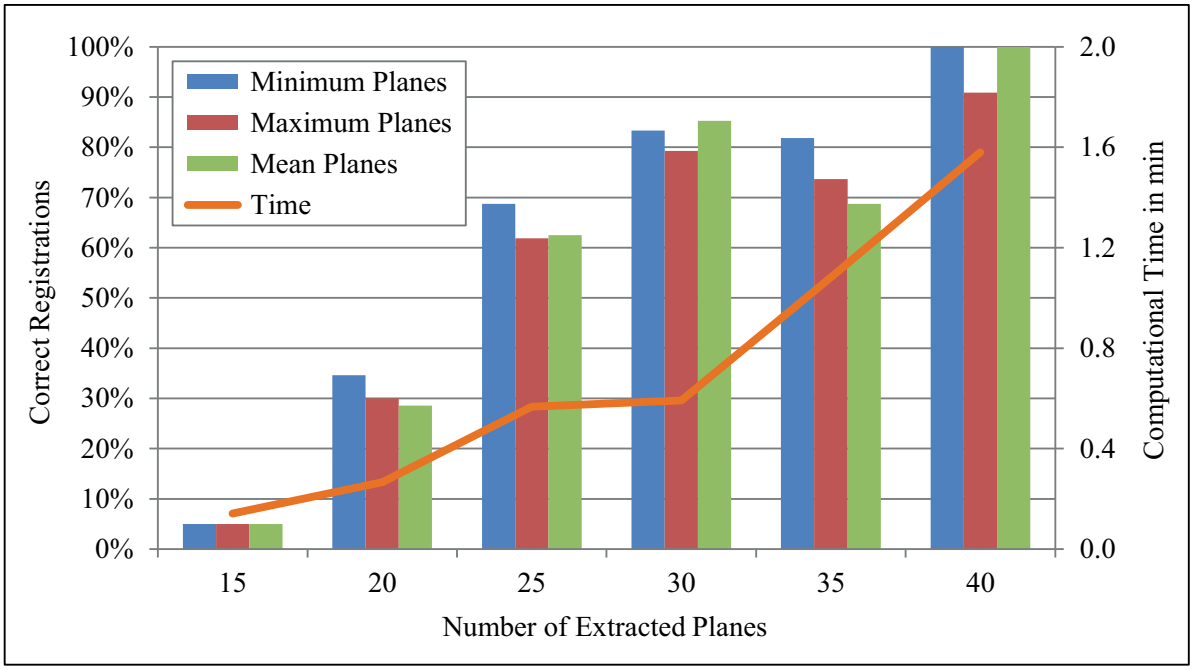


Figure 3.5: Correctness of registration and computation time for varying numbers of extracted planes.

As an extreme case to test the limits of the method, the registration of an additional scan was attempted, taken from the ground, and thus leading to massive occlusions by the furniture. The success rate decreases to 39%, and the most dissimilar scan pairs could not be matched at all.

3.7.3 Sensitivity to Noise

Noise affects the proposed method only during plane extraction, since all subsequent steps are based on the planes and not on the 3D points themselves. To test the sensitivity to noise in the

point cloud, we have repeated the test with different levels of synthetically added Gaussian i.i.d. noise (see Figure 3.6). In these experiments, the outlier threshold of RANSAC was set to three sigma. For completeness we also show the results with constants thresholds of 1 cm (low) and 10 cm (high).

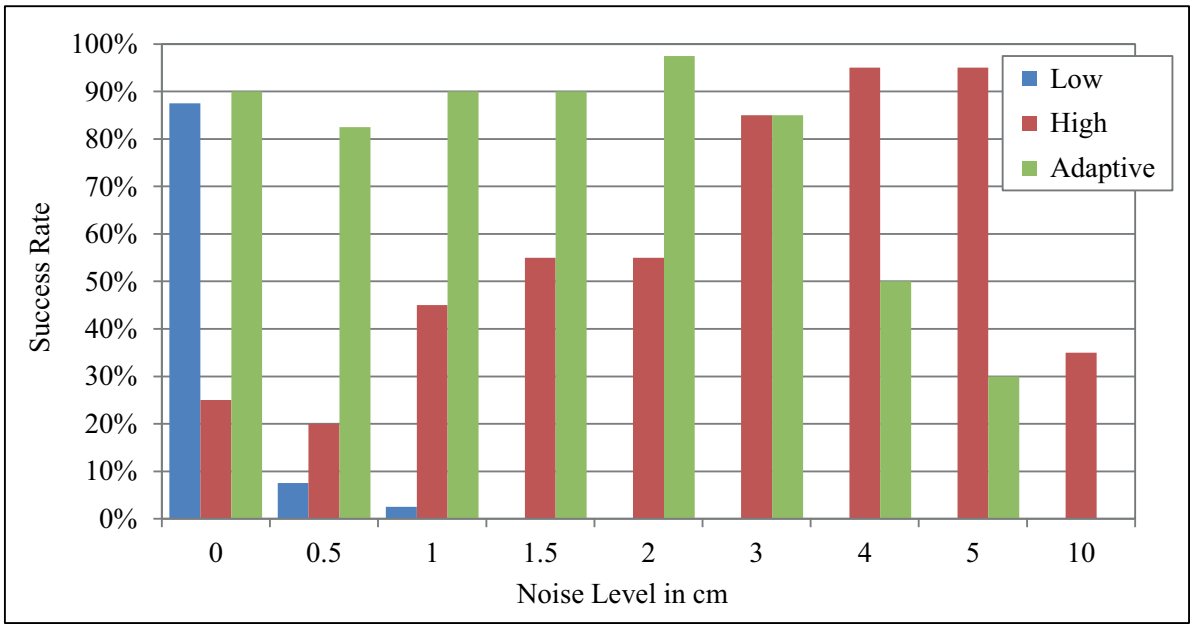


Figure 3.6: Successful registrations with different noise level regarding the outlier threshold.

As expected, a low threshold quickly causes the method to break down. A higher threshold is less efficient in case of lower noise, but surpasses the lower and even the adaptive threshold with increasing noise (>3 cm), since it has a higher chance of finding very noisy planes, even though the inlier/outlier separation might not be accurate.

Overall, when the correct threshold (respectively the measurement uncertainty) is known, the results start to deteriorate for sigma bigger than 3 cm. Given that laser scanners typically have measurement uncertainties below 1 cm, the robustness of our algorithm should be sufficient for practical applications.

3.7.4 Contribution of Descriptors and Geometric Constraints

As already mentioned in Section 3.7.1 the number of candidate matches was limited (in our tests to 5000). The mean number of candidates during a correct registration process was 4000, of which in average 45 were accepted. The geometric matching thus can successfully discover correct cliques which cover only 1% of the candidates. Reducing the maximum candidate

number results in lower success rates, by wrongly pruning correct correspondences. This confirms the claim that descriptor matching alone is not yet reliable enough.

3.8 Conclusions

Our proposed algorithm is able to perform coarse registration of two laser scans in a fully automated fashion, such that registration can be completed with a suitable algorithm such as ICP.

Our preliminary tests have shown the potential of the proposed method, although further improvements are necessary, especially regarding the efficiency of the descriptor. So far, only geometrical information is used to describe a tiepoint. We plan to also use intensity measurements of the scanner itself, and possibly also colour information of a camera with known orientation, to construct stronger descriptors. Also, splitting the matching into two sequential steps might allow one to better exploit weaker features like the plane extent, which at present have almost no influence due to their excessively low weight.

Furthermore we will also test the method in outdoor environments. We expect that extensions to other geometric elements may become necessary, since fewer planes are available outdoors.

Chapter 4

Keypoint-based Registration via 4-Points Congruent Sets

Keypoint-based 4-Points Congruent Sets – Automated marker-less Registration of Laser Scans

Pascal Willy Theiler, Jan Dirk Wegner, Konrad Schindler

ISPRS Journal of Photogrammetry and Remote Sensing, Volume 96, 2014

(Author version; for publishers' typeset version please refer to the original journal paper)

4.1 Abstract

We propose a method to automatically register two point clouds acquired with a terrestrial laser scanner without placing any markers in the scene. What makes this task challenging are the strongly varying point densities caused by the line-of-sight measurement principle, and the huge amount of data. The first property leads to low point densities in potential overlap areas with scans taken from different viewpoints while the latter calls for highly efficient methods in terms of runtime and memory requirements.

A crucial yet largely unsolved step is the initial coarse alignment of two scans without any simplifying assumptions, that is, point clouds are given in arbitrary local coordinates and no knowledge about their relative orientation is available. Once coarse alignment has been solved, scans can easily be fine-registered with standard methods like least-squares surface or iterative closest point matching. In order to drastically thin out the original point clouds while retaining characteristic features, we resort to extracting 3D keypoints. Such clouds of keypoints, which can be viewed as a sparse but nevertheless discriminative representation of the original scans,

are then used as input to a very efficient matching method originally developed in computer graphics, called *4-Points Congruent Sets* (4PCS) algorithm. We adapt the 4PCS matching approach to better suit the characteristics of laser scans.

The resulting *Keypoint-based 4-Points Congruent Sets* (K-4PCS) method is extensively evaluated on challenging indoor and outdoor scans. Beyond the evaluation on real terrestrial laser scans, we also perform experiments with simulated indoor scenes, paying particular attention to the sensitivity of the approach with respect to highly symmetric scenes.

4.2 Introduction

Airborne, mobile, and static terrestrial laser scanners are standard devices to acquire 3D data for a wide range of applications. In this work we deal with LiDAR point clouds of static *Terrestrial Laser Scanners* (TLS) that are commonly used in surveying, archaeology, manufacturing etc. Typically, multiple scans from different viewpoints are needed to fully cover large outdoor objects or complex indoor facilities in full detail. A prerequisite for any further processing of such data, like semantic classification or complete 3D reconstruction, is the relative orientation of all scans. The registration puts all scans into a common coordinate frame, thus assembling the scanned object of interest from the individual scans.

The industry standard at present is to place artificial markers in the scene during the measurement campaign. Corresponding markers (or targets) that are visible in at least two scans are then extracted either manually or automatically (e.g. Akca, 2003; Franaszek et al., 2009) to determine the relative orientation of the scans. This procedure is rather time-consuming, markers must remain stable during the measurement campaign and they should be distributed in such a way that no ill-defined geometric constellations arise. In addition, they inevitably occlude small parts of the scene and they usually have to be removed during post-processing because they are not acceptable in the final product.

To circumvent artificial markers altogether, quite some effort has been spent on finding fully automated marker-less methods for LiDAR point cloud registration. In contrast to point clouds computed via dense matching of images, the scale is inherently known in LiDAR point clouds, since the sensor directly measures distances. Therefore, registration of LiDAR point clouds can be solved with a rigid-body transformation with six degrees of freedom. Typically, such transformation parameters are estimated in a two-step procedure: An initial coarse registration roughly aligns scans with a precision that avoids the following fine registration to get stuck in a local minimum. It turns out that the first step (coarse registration) is much harder than the second (fine registration).

Various solutions for fine registration of scans exist. The most common approach today certainly is the *Iterative Closest Point* (ICP) algorithm introduced in the seminal works of Besl and McKay (1992), Chen and Medioni (1992) and Zhang (1994), and since then refined in different ways (e.g. Bergevin et al., 1996; Bae and Lichti, 2004; Minguez et al., 2006; Censi, 2008). The basic principle of ICP is to minimize the Euclidean distances between nearby points. ICP solves this problem iteratively, by first establishing point-to-point correspondences, and second estimating the transformation parameters. After applying the transformation to all points, point correspondences are sought again and a new, refined set of transformation parameters is estimated. This procedure is then repeated until convergence. A general property of all such methods is that they involve a non-convex objective function which is optimized locally. Clearly, fine registration without a sufficiently precise coarse alignment is therefore prone to get stuck in local minima. More precisely, if raw scans are not well aligned initially, a sufficiently precise transformation into a common reference frame can hardly be found because the convergence basin of the non-convex objective function is too small. Please refer to Pottmann et al. (2006) and Bae (2009) for comprehensive investigations on the convergence properties of such techniques. An idea that naturally comes into mind is thus to first coarsely align both raw scans so that this initial, rough solution is inside the ICP convergence basin. ICP then takes over and accomplishes fine registration.

Here, we follow this line of thought and propose a fully automated, marker-less, coarse registration approach, where the coarsely aligned point clouds serve as input to standard ICP. What makes this task challenging are *(i)* the huge amount of data (millions of points), which calls for computationally efficient techniques, *(ii)* the typically large baselines between adjacent scanning viewpoints (to limit time and costs in the field) and *(iii)* the quadratic decrease of the point density with distance from the scanner (due to the angular sampling of scanning LiDAR devices). The latter causes different point densities on the same surfaces in different scans. As we will see later on in this paper, this property calls for custom-designed processing steps: standard approaches from computer vision or computer graphics cannot be applied directly because they usually assume approximately even point distributions within and between different scans.

In this paper we adapt the *4-Points Congruent Sets* (4PCS) approach for coarse registration, proposed by Aiger et al. (2008), to the aforementioned challenges. To keep registration computationally tractable we downsample raw scans and represent them with a sparse cloud of 3D keypoints. We test two different kinds of keypoints, the 3D *Difference-of-Gaussians* (DoG) detector and the 3D *Harris* corner detector. The first relies on LiDAR intensities, whereas the second fires at distinct geometrical structures. In contrast to heavily downsampling point clouds at random, less aggressive downsampling followed by keypoint extraction better

preserves salient features of the scene and thus offers better repeatability across scans. We call the combined method, which utilizes 3D keypoints as input to a (slightly modified) 4PCS algorithm *Keypoint-based 4-Points Congruent Sets* (K-4PCS). We plan to release the test data and the source code of the proposed method – integrated in the open-source *Point Cloud Library* (PCL; Rusu and Cousins, 2011) – after publication.

4.3 Related Work

Coarse, marker-free registration of point clouds typically follows a two-step strategy. First raw point clouds are reduced to sparse sets of features, and second corresponding features are sought in overlapping areas to estimate transformation parameters that align the point clouds sufficiently well. The result then serves as input to standard fine registration like ICP (Besl and McKay, 1992). A common approach to feature extraction are 2D keypoints that are well known from image processing and can be applied to either intensity or range images of scans (Böhm and Becker, 2007; Kang et al., 2009). A drawback of pure 2D features is that they cannot cope well with strong viewpoint changes. 3D features are generally more robust (Allaire et al., 2008; Lo and Siebert, 2009; Flitton et al., 2010; Flint et al., 2007) and we will thus use them in our work. Another popular line of thought, particularly in man-made environments, is to derive features from planar surfaces (Brenner and Dold, 2007; Brenner et al., 2008; Theiler and Schindler, 2012). More general strategies that can handle arbitrarily shaped surfaces rely on salient directions (Novák and Schindler, 2013) or on a combination of salient directions and 2D features after an ortho-rectification process (Zeisl et al., 2013). It should be noted that approaches purely based on geometric object properties are limited to scenes of medium complexity due to self-occlusions, which occur when very complex surfaces are acquired from different viewpoints.

Sampling and matching of extracted features often follows strategies related to RANSAC (Fischler and Bolles, 1981). A popular variant called *Sample Consensus Initial Alignment* was proposed by Rusu et al. (2009). They match corresponding point triplets to align scans, which however implies runtime complexity per sample of $\mathcal{O}(n^3)$. An elegant method to do coarse and fine registration in one step was proposed recently by Yang et al. (2013). They develop a globally optimal version of ICP coined Go-ICP, which avoids mismatches due to local minima by global optimization with a clever branch-and-bound scheme. It runs efficiently on rather small point clouds of up to a few thousand points, but does not scale up to the typical size of laser scans.

We propose a completely automated method for marker-less coarse registration of TLS point

clouds of arbitrary relative orientation based on *(i)* a sparse representation via discriminative 3D keypoints that are *(ii)* matched efficiently with a variant of the 4PCS method of Aiger et al. (2008). They showed that the computational burden of point cloud matching can be reduced to $\mathcal{O}(n^2)$, by adding a fourth point in such a way that the four points are roughly coplanar. They achieve high efficiency and success rates if matching uniformly distributed point clouds. Here, we build upon our preliminary work (Theiler et al., 2013), but give a more in-depth description of the standard method and its extensions, add a second, purely geometric keypoint detector, conduct extensive experiments on synthetic data concerning robustness against repetitive structures, and present much extended experiments on real indoor and outdoor data. Experimental results show that K-4PCS reaches high success rates above 95% in moderately difficult scenarios and $\approx 65\%$ success even for extremely challenging scenes, with accuracies easily high enough to initialize standard ICP.

4.4 Conceptual Overview

The proposed method combines ideas from image processing and computational geometry. On the one hand we make use of *Difference-of-Gaussians* (DoG) keypoints comparable to the ones introduced by Lowe (1999) – also known as SIFT keypoints – but adapted to 3D. Alternatively, we use keypoints found by a 3D *Harris* corner detector. The two methods are representative of 3D keypoint extraction: DoG takes into account point intensities, whereas Harris is purely geometry-based (see details in Section 4.5). On the other hand, the original 4PCS algorithm of Aiger et al. (2008) is adapted to handle the resulting sparse clouds of keypoints. This algorithm (see Section 4.6) is an efficient matching strategy based on corresponding keypoint quadruples and random sampling. Figure 4.1 shows the overall workflow of the algorithm, which goes as follows:

(1) Given two raw laser scans we first apply a voxel grid filter to roughly even out the strongly varying point distribution across point clouds that stems from the angular sampling measurement principle.

(2) After filtering we apply the 3D version of the DoG or Harris keypoint extractor to obtain a sparse, but nonetheless discriminative representation of the original point cloud.

(3) These clouds of 3D keypoints are matched using the 4PCS method, tuned to take into account the special characteristics of the keypoint sets.

(4) As a result we get quadruples of corresponding keypoint pairs, which are used to compute a rigid-body transformation.

(5) Finally, coarsely aligned point clouds are fine-registered with ICP.

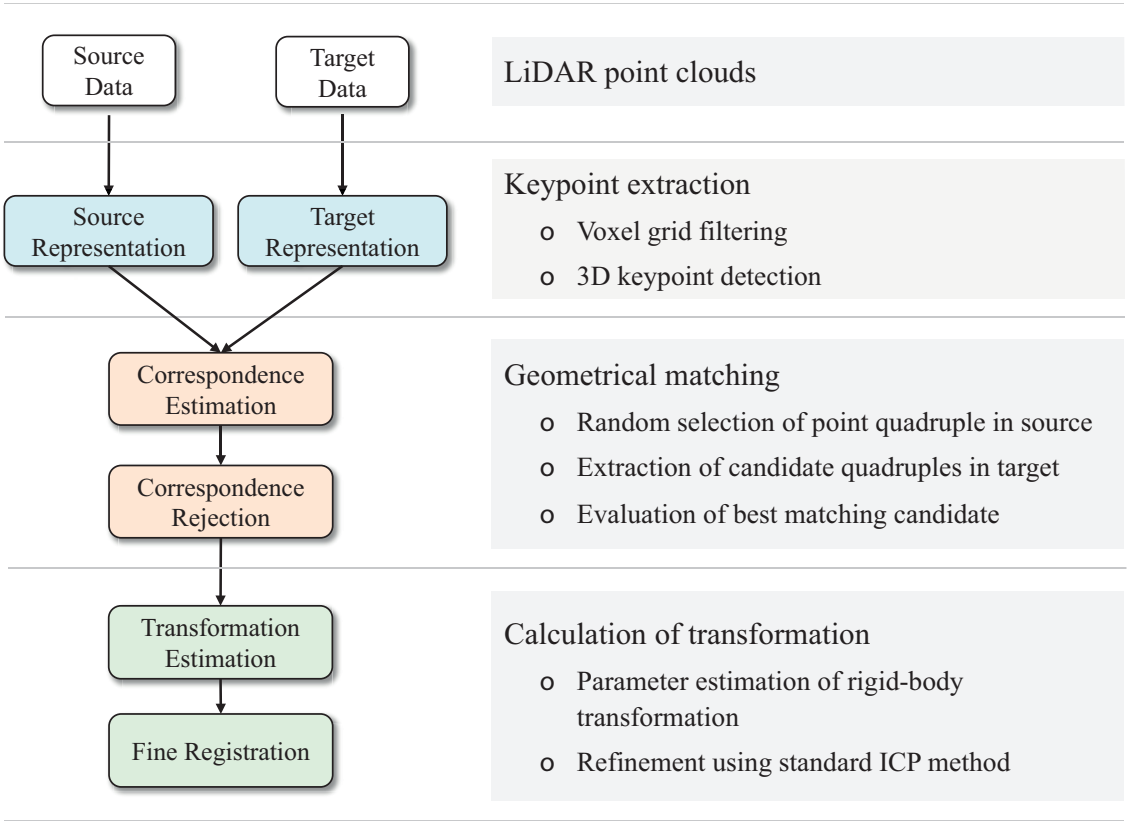


Figure 4.1: General workflow of the proposed K-4PCS method.

Note that the described workflow does not depend on the type of extracted keypoints, because the matching is only based on geometrical information. In particular, we do not match descriptor vectors of keypoints, but solely rely on their relative positions in 3D space. The entire framework is thus generic and independent from a particular kind of 3D keypoint detector. In the following sections, we describe first the extraction of 3D DoG as well as 3D Harris keypoints and second the keypoint-based matching called K-4PCS.

4.5 Keypoint Extraction

Standard TLS point clouds have tens of millions of unevenly distributed points, which makes coarse registration of the raw point clouds computationally very expensive. In order to reduce a point cloud to those points that are the most useful (and discriminative) for registration, we extract 3D keypoints. The *cloud of 3D keypoints* is a sparse representation with a high repeatability (i.e. stability). This property greatly increases the chance of finding a corresponding point in a second cloud of 3D keypoints. A visual example of the original point cloud and its sparse representation via a cloud of 3D keypoints is given in Figure 4.2.

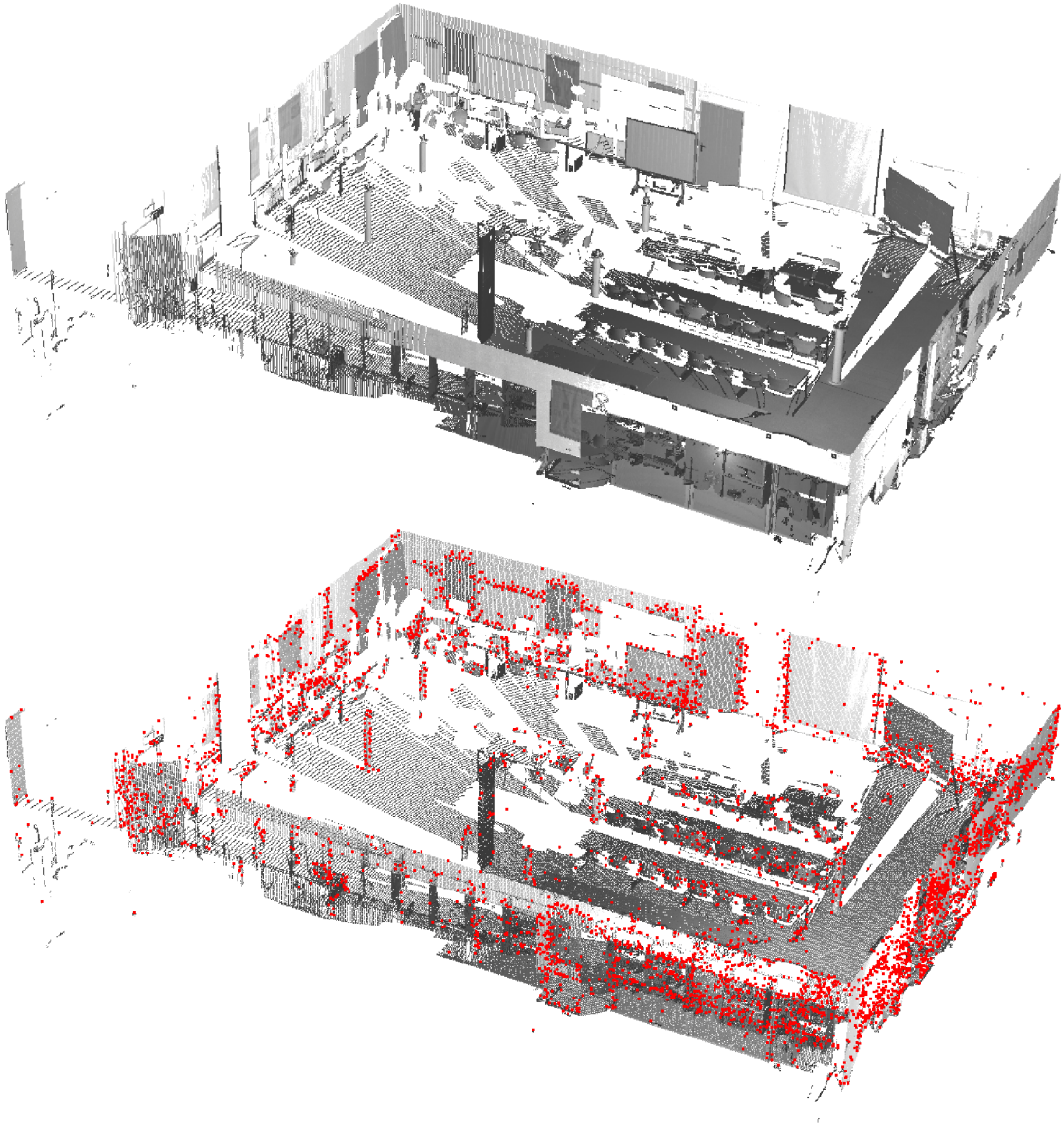


Figure 4.2: 3D keypoint extraction. The raw scan cloud (top) is sampled using a voxel grid based filter (bottom, grey), and the result serves as input for the detection of keypoints (here: DoG keypoints) which results in a sparse but highly discriminative keypoint cloud (bottom, red). The keypoints are visualized on top of the voxel gridded scan cloud. For better visibility, the room ceiling has been removed.

In order to avoid the near-field bias inherent in regular angular sampling, and to achieve an initial reduction of the point count, a voxel grid filter is first applied. It divides the three-dimensional space into a regular grid of blocks (or voxels) of size τ. All scan points within each such block are determined, and their centroid is computed to henceforth serve as representative of the respective voxel. Note that working with the centroid instead of the more commonly used voxel centre better preserves the original spatial layout of points. The filtered point cloud serves

as input to the 3D keypoint detectors. We use 3D versions of *Difference-of-Gaussians* (DoG) and *Harris* keypoint detectors. Both are available in the open-source *Point Cloud Library* (PCL; Rusu and Cousins, 2011). While conceptually the generalization to 3D is straight-forward, the technical details are not widely known, so we will briefly introduce them in the following two sections.

4.5.1 3D Difference-of-Gaussians Keypoints

Nowadays, *Difference-of-Gaussians* (DoG) keypoint extraction in 2D is a standard tool for matching tasks in image processing. Major advantages of DoG keypoints are their invariance to scaling, rotation, and translation. In the original framework of Lowe (1999) keypoints are not only detected but also encoded in a descriptor, which results in the well-known SIFT features. It should be noted that we do not make use of the descriptor and only employ DoG points for geometrical constraint matching. We thus prefer to use the term DoG (instead of SIFT).

In our framework the extraction of DoG keypoints is carried out directly in 3D. The reason is that in 2D many high-contrast edges are on object silhouettes and depth discontinuities, leading to keypoints that are unstable across viewpoints. A 3D detection scheme avoids such unstable points by selecting only points which have high contrast to their neighbours in 3D space. The DoG detector is an efficient approximation of the scale-normalized Laplacian. In 2D it is found by repeatedly blurring an image with Gaussian filters of increasing scale. The subtraction of images with adjacent blur scales results in a DoG response, in which local minima and maxima are then detected.

For the 3D version, the same principle is based on the LiDAR return intensities. Although narrow-band LiDAR responses are quite different from image intensities (i.e., objects can have the same colour but different reflectance properties and vice versa), the effectiveness of the detector does not seem to suffer significantly. Comparable to the 2D approach, the detection of 3D DoG keypoints is based on computing in each blur level τ_k (with $k = 1 \dots b$) a Gaussian response G for each point (taking into account all neighbours in a given radius $r_k = 3 \cdot \tau_k$), subtracting the responses of adjacent scales at each point to obtain DoG responses R^G (Eq. (4.1)), and finally detecting keypoints as local minima and maxima in the DoG scale space.

$$R_i^G(x, y, z, \tau_k) = G_i(x, y, z, \tau_{k+1}) - G_i(x, y, z, \tau_k) \tag{4.1}$$

A valid keypoint is found, if the DoG response of the point is bigger (maxima) or smaller (minima) than the responses of each neighbour and additionally, if the absolute value of the points response exceeds a given threshold R_{min}. The calculation is repeated q times, doubling

the base scale τ_1 of the voxel grid in each iteration (octave). The number of octaves as well as the number of scales per octave are user parameters and influence the number of extracted keypoints.

4.5.2 3D Harris Keypoints

As an alternative to the DoG detector we use the *Harris* corner detector introduced by Harris and Stephens (1988) and adapted to 3D space in Rusu and Cousins (2011). In comparison to the original idea of detecting corners and edges based on image gradients, the extraction in 3D is based on the local normals of the point cloud. It only uses geometrical properties and unlike DoG does not need intensities. First, a local normal is computed for each point from all points within a neighbourhood $\mathcal{N}$. The search radius r to locate neighbours is chosen as $r = 3 \cdot \tau$, with τ as the voxel size of the basic grid. From the normals $\mathbf{n}_i = (n_{xi}, n_{yi}, n_{zi})^\top$ in the neighbourhood a covariance matrix $\mathbf{\Sigma}$ is calculated for each point in the input point cloud,

$$\mathbf{\Sigma} = \frac{1}{|\mathcal{N}|} \cdot \sum_{i \in \mathcal{N}} \mathbf{n}_i \cdot \mathbf{n}_i^\top \quad . \tag{4.2}$$

The covariance matrix delivers a point-wise response R^H based on determinant $\det$ and trace tr,

$$R^H = \det(\mathbf{\Sigma}) - \gamma \cdot \mathrm{tr}(\mathbf{\Sigma})^2 \tag{4.3}$$

with γ an appropriately chosen constant. As for DoG, corners are extracted by searching for local maxima in the response space. The response must additionally exceed a given threshold R_{min} for a keypoint to be accepted. Note, the two thresholds are different since DoG and Harris responses do not have the same scale.

4.6 Keypoint-based 4-Points Congruent Sets Matching

The matching of extracted keypoints is based on the *4-Points Congruent Sets* (4PCS) algorithm of Aiger et al. (2008). 4PCS was originally designed for aligning partially overlapping but rather evenly distributed, sparse point clouds with arbitrary orientation. It is an efficient variation of the obvious brute-force method to randomly sample congruent point triplets. Let us first examine the brute-force solution: a point triplet is extracted randomly from the source dataset $\mathcal{S}$, and a congruent triplet is found in the target dataset $\mathcal{T}$. From the congruent triplets, the alignment (i.e. rigid-body transformation) is determined and applied to the source point cloud to obtain $\mathcal{S}'$. The alignment is verified by counting the number of point pairs from $\mathcal{S}'$

and $\mathcal{T}$ with small enough residuals. Random selection of base triplets in $\mathcal{S}$ is repeated until an alignment with large enough support is found (or a maximum number of iterations is reached).

The problem of the brute-force approach is that it is hardly tractable, having computational complexity of at least $\mathcal{O}(n^3 \log n)$ with n as the number of points in $\mathcal{T}$ (Irani and Raghavan, 1996). The main insight of 4PCS is that sampling four approximately co-planar points instead of a minimal set of three points reduces computational complexity to $\mathcal{O}(n^2)$, making it possible to align point clouds of reasonable size.

Alignment proceeds in the same way (Figure 4.3): given two input datasets $\mathcal{S}$ and $\mathcal{T}$, randomly sample a four-point base set $\mathcal{B}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) \in \mathcal{S}$ and search for a corresponding set $\mathcal{M}(\mathbf{p_1}, \mathbf{p_2}, \mathbf{q_1}, \mathbf{q_2}) \in \mathcal{T}$ with high support. However, as will be seen shortly, the corresponding sets can be found much more efficiently. To increase the geometrical stability and reliability of the transformation, the algorithm is biased towards selecting base sets with large point-to-point distances. In K-4PCS the maximum base length s_{max} is either defined by the user, or derived from the point cloud diameter and an estimate of the scan overlap.

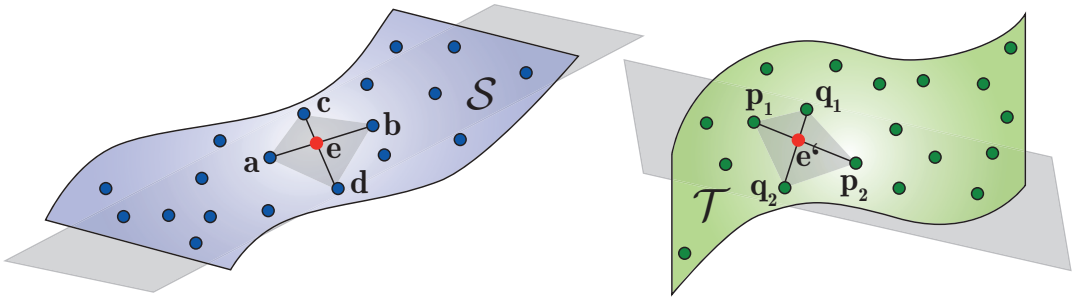


Figure 4.3: Basic principle of 4PCS with base set $\mathcal{B}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) \in \mathcal{S}$ and a corresponding congruent point set $\mathcal{M}(\mathbf{p_1}, \mathbf{p_2}, \mathbf{q_1}, \mathbf{q_2}) \in \mathcal{T}$.

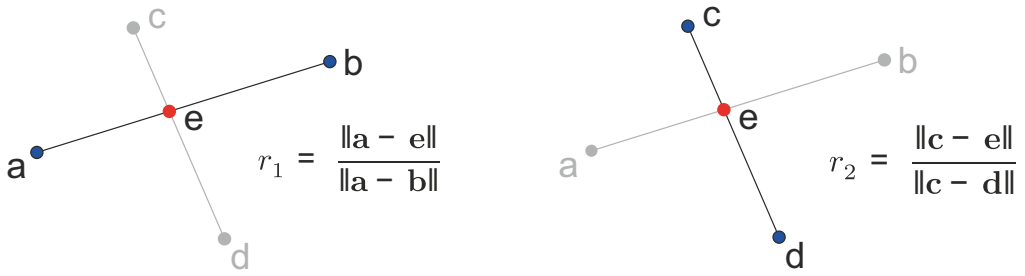


Figure 4.4: Illustration of the intersection point $\mathbf{e}$ using the base set $\mathcal{B}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) \in \mathcal{S}$.

4PCS exploits the rule that intersection ratios of the diagonals in an arbitrary planar quadrangle are invariant under affine transformation (Huttenlocher, 1991). Having picked a base set of four approximately co-planar points, the intersection point $\mathbf{e}$ of the diagonals as well as the corresponding intersection ratios r_1 and r_2 can be computed (Figure 4.4). Finding a congruent

set $\mathcal{M}$ then amounts to calculating two intersection points $\mathbf{e}(r_1), \mathbf{e}(r_2)$ per point pair $\mathbf{p_1}, \mathbf{p_2} \in \mathcal{T}$, using r_1 and r_2 of the current base $\mathcal{B} \in \mathcal{S}$ (Eq. (4.4)). A valid match $\mathcal{M}$ consists of two point pairs $\mathcal{P}\{\mathbf{p_1}, \mathbf{p_2}\}$ and $\mathcal{Q}\{\mathbf{q_1}, \mathbf{q_2}\}$ whose respective intersection points coincide. In practice we need to verify $\|\mathbf{e_p}(r_1) - \mathbf{e_q}(r_2)\| < \delta_1$ or $\|\mathbf{e_p}(r_2) - \mathbf{e_q}(r_1)\| < \delta_1$. This check is done with an efficient approximated nearest neighbour search with tolerance level δ_1.

$$\begin{aligned} \mathbf{e_p}(r_1) &= \mathbf{p_1} + r_1 \cdot (\mathbf{p_2} - \mathbf{p_1}) \\ \mathbf{e_p}(r_2) &= \mathbf{p_1} + r_2 \cdot (\mathbf{p_2} - \mathbf{p_1}) \end{aligned} \tag{4.4}$$

Up to this point, the method is able to cope with full affine transformation. However, in case of laser scan registration the task is to find a rigid-body transformation. In this case one can add additional constraints, as suggested by Aiger et al. (2008). First, instead of calculating intersection points for each point pair of $\mathcal{T}$, possible pairs are filtered, based on the length of the diagonals in the base set $\mathcal{B}$ (Eq. (4.5)), and intersection points are computed only for valid point pairs.

$$\begin{aligned} \|\mathbf{p_1} - \mathbf{p_2}\| - \|\mathbf{b} - \mathbf{a}\| < \delta_2 \\ \|\mathbf{q_1} - \mathbf{q_2}\| - \|\mathbf{d} - \mathbf{c}\| < \delta_2 \end{aligned} \tag{4.5}$$

Second, given a matched set $\mathcal{M}(\mathbf{p_1}, \mathbf{p_2}, \mathbf{q_1}, \mathbf{q_2})$, the side length of the base set $\mathcal{B}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})$ can be used to verify the congruency of the irregular rectangles, that is we check

$$\begin{aligned} \|\mathbf{a} - \mathbf{c}\| - \|\mathbf{p_1} - \mathbf{q_1}\| < \delta_3 \ , \quad \|\mathbf{a} - \mathbf{d}\| - \|\mathbf{p_1} - \mathbf{q_2}\| < \delta_3 \ , \\ \|\mathbf{b} - \mathbf{c}\| - \|\mathbf{p_2} - \mathbf{q_1}\| < \delta_3 \ , \quad \|\mathbf{b} - \mathbf{d}\| - \|\mathbf{p_2} - \mathbf{q_2}\| < \delta_3 \ . \end{aligned} \tag{4.6}$$

The best-matching $\mathcal{M}$ and $\mathcal{B}$ are defined as those which result in the largest number of matching points (the largest overlap) between the transformed source cloud $\mathcal{S}'$ and the target cloud $\mathcal{T}$. In practice the overlap is evaluated only for a fixed random subsample $\mathcal{S}'_{sub} \in \mathcal{S}'$ of 1000 points. Transformation parameters for each valid candidate match are computed using singular value decomposition and applied to $\mathcal{S}'_{sub}$. Then, a nearest neighbour in $\mathcal{T}$ is searched for each point. The overlap is found as the fraction of nearest neighbours that lie within a threshold δ_4.

Like in standard RANSAC, base set sampling, matching and evaluation is repeated w times. The details about the calculation of w can be found in (Aiger et al., 2008, Eq. (1)). The method returns the first solution which has a support larger than a given threshold Γ, or in case this threshold is not reached the solution with the highest support after all w repetitions. Γ can be directly set by the user, or it is automatically set equal to the estimated scan overlap.

As described above, 4PCS is based on a number of tolerances δ_i. However they are all

strongly correlated and can be related to the mean density of the input point clouds.[1] But these considerations do not directly apply to clouds of keypoints, which are not distributed uniformly. Recall that keypoints occur at characteristic and discriminative object parts but not on homogeneous surfaces. Consequently, we can hardly derive tolerances from the mean point density. Moreover, the probability of finding corresponding keypoints is much higher than that of finding correspondences in a random set of the same size, since keypoints are chosen to be repeatable across scans.

In our framework the stability, repeatability and number of extracted keypoints depend mainly on the voxel size τ of the initial voxel grid filter, which at the same time serves as base scale for DoG keypoint extraction, respectively defines the neighbourhood radius of the Harris keypoint detector. Therefore, we propose the following scheme to set the parameters in a data-driven manner: we define $\delta_1, \delta_2, \delta_3$ w.r.t. the voxel grid spacing and set

$$\begin{aligned} \delta_1 &= \delta_3 = 4 \cdot \tau \\ \delta_2 &= \tau \quad . \end{aligned} \tag{4.7}$$

The inlier threshold $\delta_4 = \varrho^2$ is still estimated from the point cloud density ϱ, because K-4PCS allows to verify a match either using the keypoint cloud itself, or using the (usually voxel grid filtered) input point cloud. However, in the experiments described in Sections 4.7 and 4.8.3, the support of a match is solely based on the resulting overlap of the keypoint cloud. We found that with these modified parameters 4PCS works much better with keypoints than with randomly decimated point clouds of the same size.

4.7 Experiments

We test efficiency, success rate, the sensitivity of parameters with respect to different scenes, and robustness of K-4PCS on four different TLS datasets. In the following two subsections, we describe and analyse results of two indoor and two outdoor datasets. The data sets address different challenges; in general one can say that the indoor datasets have rather large overlaps but a high degree of symmetry, whereas the outdoor scans have lower overlap.

To generate reference registrations we manually aligned all scans in a dataset, followed by a refinement with standard ICP. Residuals between corresponding point pairs of the transformed source point cloud $\mathcal{S}'$ and the target point cloud $\mathcal{T}$ are used to estimate the accuracy of the

[1] More details about the calculation of these tolerances of the 4PCS algorithm can be found in Aiger et al. (2008) and associated open-source code.

manual registration, further called $\sigma^\star$. All point correspondences which have been used in the final ICP iteration form the overlap and are henceforth called *true correspondences* $C^\star = \{c_s, c_t\}$.

The goal of K-4PCS is to coarsely align source and target point clouds, in such a way that the solution falls into the convergence basin of a fine registration method. To define a successful trial, the processing pipeline for experiments has thus been extended to include a refinement of the initial solution based on the standard ICP algorithm. Therefore, the success PS of a trial can directly be evaluated based on the accuracy of the converged ICP solution ($RMSE_{\text{ICP}}$) with respect to manual registration accuracy $\sigma^\star$ (Eq. (4.8))). Note that the $RMSE$ is only calculated using the true correspondences $C^\star$.

$$PS = \begin{cases} 1 & \text{if} \quad RMSE_{\text{ICP}} \leq 3 \cdot \sigma^\star \\ 0 & \text{if} \quad RMSE_{\text{ICP}} > 3 \cdot \sigma^\star \end{cases} \tag{4.8}$$

K-4PCS is a randomized matching strategy. To even out fluctuations due to the randomization, the success is evaluated on a series of $m = 50$ trials. Our evaluation criteria is the success rate, defined by

$$PSR = \frac{1}{m} \cdot \sum_i S \ , \quad m = 1 \ldots 50 \ . \tag{4.9}$$

In addition to the success rate, the metric accuracy of a coarse alignment is addressed. The more precise we estimate an initial alignment, the lower the chance that fine registration converges towards a wrong local optimum plus convergence requires fewer iterations and registration is faster. All statistical values are computed with respect to the manually aligned references (with accuracy $\sigma^\star$). On the one hand, the transformation parameters – estimated using the four correspondences received from K-4PCS without an ICP refinement step – are directly compared to the reference. Differences are split into *Mean Angular Error* (MAE) and *Mean Translation Error* (MTE). On the other hand, the metric accuracy is represented by the $RMSE$ between target $\mathcal{T}$ and transformed source $\mathcal{S}'$ – applying the transformation parameters of the coarse alignment – while considering only the true correspondences $C^\star$. Naturally, the metric accuracy is only calculated on successful trials, ignoring falsely matched pairs.

$$RMSE_{\text{M}} = \sqrt{\frac{1}{|C^\star|} \sum_{i \in c_s, j \in c_t} (\mathbf{u'}_i - \mathbf{v}_j)^2} \ , \quad \mathbf{u'} \in \mathcal{S}', \ \ \mathbf{v} \in \mathcal{T} \tag{4.10}$$

Finally, we also record runtimes of the keypoint matching method, which is implemented as a multi-threaded algorithm using *OpenMP*. That is, the w repetitions (base selection, candidate matching and validation) are distributed over different threads/cores and run in parallel. All

experiments are carried out on a standard 64 Bit Windows 7 desktop computer with twelve cores (3.2 GHz) and 16 GB RAM.

As described in Sections 4.5 and 4.6, K-4PCS has a few parameters that have to be set by the user. To find good settings and test parameter sensitivity across different types of scenes we perform extensive parameter studies. Preliminary tests have shown that some parameters did not need to be changed, but can be set constant over all tests. We thus focus on those that have to be adapted to the task. Given the expected large influence of the basic voxel grid resolution τ, all tests were done for three different voxel sizes $\tau = \{200\,\text{mm}, 100\,\text{mm}, 50\,\text{mm}\}$. Limited spatial extent and large overlap of indoor scans result in lower runtimes, allowing us to additionally evaluate a voxel size of $25\,\text{mm}$. For DoG keypoint extraction the number of octaves as well as the number of scales per octave was set to five. This proved to be a good trade-off between computation time and keypoint quality. A more crucial parameter, which strongly influences the number of extracted DoG and Harris keypoints, is the minimum response threshold R_{min}. The impact of this threshold on the success rate was evaluated on one of the datasets (see Section 4.7.1).

Regarding K-4CPS parameters, the minimum support threshold for a registration to be deemed successful is always set to $\Gamma = 1$. This ensures that the matching method always runs for the full w iterations and runtime remains comparable across different setting, which otherwise would strongly depend on the estimated overlap. Note however, this is the most conservative setting possible, and runtimes could often be significantly reduced with a more realistic and informed threshold. We have conducted additional tests with one indoor and one outdoor data set to evaluate varying estimations of the overlap. The maximum distance s_{max} of the base samples during matching is in general automatically calculated using the estimated overlap.

4.7.1 Indoor Applications

We have tested K-4PCS on two different indoor datasets with different scene structures posing various challenges. The first dataset ("Office") consists of five scans acquired in a laboratory room with dimensions $\approx 15\,\text{m} \times 10\,\text{m} \times 3\,\text{m}$. Five scans have been acquired with a *Zoller+Fröhlich TLS Imager 5006i* that has a field of view of $360°$ horizontally and $150°$ vertically, and a maximum range of $\approx 79\,\text{m}$. This room mainly contains standard office furniture like tables, chairs, shelves and blackboards and additionally features several cylindrical pillars (Figure 4.5). Although the room has a rather simple geometrical layout, automated alignment is demanding due to multiple rotational symmetries. On the other hand, its limited spatial extent yields large scan overlaps in spite of the limited scan range, which facilitates point cloud registration.

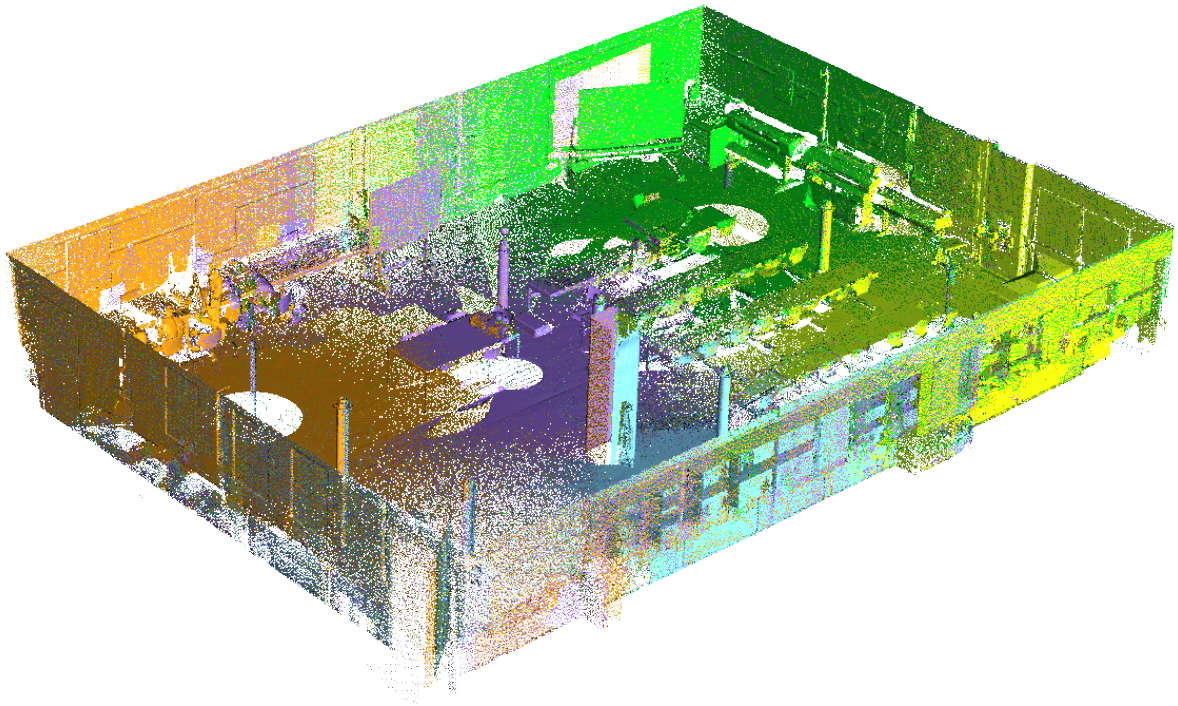


Figure 4.5: Dataset “Office” acquired in a laboratory room. The ceiling has been removed and only 33% of the data are displayed, for a better visual impression.

A first test on dataset “Office” has been carried out with an estimated overlap of 80%. Recall that the maximum base length is automatically calculated from this setting. Both keypoint detectors (DoG and Harris) were applied to point clouds of four different voxel sizes. Minimum response thresholds of valid keypoints are set to $R^G_{min} = 0.01$ for DoG, respectively $R^H_{min} = 0.0001$ for the Harris keypoint detector. Table 4.1 shows results regarding success rate PSR, as well as runtime T of keypoint matching. All values are averaged over all trials of all different scan pairs. Furthermore, the mean number of extracted keypoints k is given. Due to the large standard deviation of k of up to 180, the values are rounded to the nearest multiple of hundred.

	DoG				**Harris**			
τ in mm	200	100	50	25	200	100	50	25
PSR in %	63.8	96.2	97.6	96.8	84.4	87.8	90.2	82.2
T in s	1	8	12	21	$\ll 1$	$\ll 1$	$\ll 1$	1
k	400	1200	3200	8200	100	300	800	2300

Table 4.1: Results of dataset “Office”: mean success rate PSR, matching time T and number of extracted keypoints k over all scan pairs and trials. All tests were run with four different voxel sizes τ and are based on either DoG or Harris keypoints.

The registration success rate is generally high and reaches $\approx 98\%$ with optimal parameter settings. A close inspection of failure cases revealed that these are predominantly caused by the large degree of symmetry, leading to false registrations that are $180°$ rotated. One property of matching that biases results towards rotated solutions (in case of high degrees of scene symmetry) is its tendency to favour solutions with minimal translations, because even after voxel grid filtering more keypoints are detected in the scanner's near field. In case of scan pairs in opposite room corners, this causes the matching to fail more often (e.g. s1-s2; see Table 4.2).

	PSR in %	MAE in °	MTE in mm	$RMSE_{\mathrm{M}}$ in mm	$RMSE_{\mathrm{ICP}}$ in mm
s1-s2	86.0	0.8 ± 0.7	169 ± 94	168 ± 65	16 ± 0
s1-s3	100.0	0.5 ± 0.4	128 ± 58	125 ± 49	11 ± 0
s1-s4	98.0	0.7 ± 0.6	141 ± 99	148 ± 72	4 ± 0
s1-s5	98.0	0.6 ± 0.6	133 ± 99	138 ± 84	11 ± 0
s2-s3	100.0	0.3 ± 0.2	72 ± 32	73 ± 23	11 ± 0
s2-s4	100.0	0.6 ± 0.5	112 ± 88	114 ± 63	9 ± 0
s2-s5	100.0	0.5 ± 0.4	132 ± 71	124 ± 42	9 ± 0
s3-s4	98.0	0.6 ± 0.6	99 ± 39	132 ± 54	6 ± 0
s3-s5	100.0	0.5 ± 0.4	100 ± 78	122 ± 64	10 ± 0
s4-s5	96.0	0.8 ± 0.6	146 ± 64	173 ± 65	10 ± 0
Mean	**97.6**	$\mathbf{0.6 \pm 0.5}$	$\mathbf{122 \pm 72}$	$\mathbf{131 \pm 58}$	$\mathbf{10 \pm 0}$

Table 4.2: Metric accuracy represented by the *Mean Angular Error* (MAE), *Mean Translation Error* (MTE) and *Root Mean Square Error* between true correspondences before ($RMSE_{\mathrm{M}}$) and after ICP refinement ($RMSE_{\mathrm{ICP}}$). Results are shown for dataset "Office", using $\tau = 50$ mm and DoG keypoints.

We can also observe the expected correlation between voxel size τ of the input point cloud and the number of extracted keypoints k in Table 4.1. Decreasing τ better preserves details and naturally results in more detected keypoints. This in turn leads to a higher repeatability and thus to higher success rates PSR. We always detect less Harris than DoG keypoints because the geometrical structure of dataset "Office" is rather simple (while it does exhibit significant texture on the walls). As a consequence, the runtime with Harris keypoints is much lower compared to DoG.

The slight drop of the success rate at the smallest τ-level (25 mm) for both keypoint detectors is most probably caused by the following effect: if τ is chosen smaller than the mean point density of the input point cloud in the overlap area, keypoint detection is biased towards extracting

more keypoints in dense, non-overlapping areas. This increases the chance of selecting bases that are located outside the scan overlap, and consequently reduces the probability of finding a correct match. In conclusion, highest success rates are achieved by setting τ in the range of the estimated point density in the overlapping area (i.e. here ≈ 50 mm). Runtime is again a function of the voxel size: smaller τ generally yields higher success rates, at the cost of longer computation time.

The metric accuracy was tested using a voxel size $\tau = 50$ mm and DoG keypoints. The results are shown in Table 4.2. Note that the standard deviation of the mean value is referring to the average standard deviation over the single scan pairs. It can be seen, that the angular error is $\approx 0.6^\circ$ and does not change significantly between scan pairs. The translation error as well as the $RMSE_{\mathrm{M}}$ are < 15 cm, whereas slightly worse solutions are reached in case of less successful scan pairs. After ICP refinement, the $RMSE$ considering the true correspondences is reduced to ≈ 1 cm, which is on the order of the scanner's measurement accuracy. We noted that all standard deviations (1σ) are in the same order as the mean values. In conclusion, the reachable accuracy of the coarse alignment is quite variable, but in most cases (here $\approx 98\%$) is still good enough to put the scans into the convergence basin of ICP.

In a further test we have assessed the influence of minimum response threshold R_{min} of both keypoint detectors. Like the voxel size τ, this threshold has a direct impact on the number of detected keypoints. Table 4.3 compares the number of detected DoG keypoints and the resulting success rates at varying $R^G_{min} \in \{0.005, 0.01, 0.02, 0.04\}$.

	$R_{min} = 0.04$		$R_{min} = 0.02$		$R_{min} = 0.01$		$R_{min} = 0.005$	
τ in mm	k	PSR in %	k	PSR in %	k	PSR in %	k	PSR in %
200	100	26.6	300	58.4	400	63.8	500	62.4
100	200	20.2	700	75.6	1200	96.2	1500	98.0
50	300	15.0	1700	82.6	3200	97.6	4200	98.4
25	800	18.0	4300	81.2	8200	96.8	11000	98.6

Table 4.3: Success rate PSR and number of detected DoG keypoints k with different minimum response thresholds R_{min} applied on dataset "Office".

It comes as no surprise that the number of keypoints increases with a decreasing minimum response R_{min} and with a decreasing voxel size τ. Naturally, PSR highly depends on the number of detected keypoints. Recall that runtime quadratically increases with the number of keypoints and thus larger keypoint numbers than necessary are in general not desirable.

The matching part of K-4PCS depends on three parameters (cf. Section 4.6). While the minimum scale τ has been varied for each previous test, the maximum sampling distance for the base selection s_{max} as well as the estimated overlap were so far set to constant values derived from preliminary tests. Although s_{max} is usually automatically derived from the estimated overlap, additional tests were performed to assess success rates while varying both matching parameters. Note, to be independent of the absolute point cloud size, the maximum baseline is again given in percent of the point cloud diameter.

It turns out that the success rate is not significantly altered if choosing different parameters in a range of $0.6 \ldots 1.0$, which is probably caused by the large overlap of scans in dataset "Office". Note, the number of sampling iterations w is derived from the estimated overlap (higher overlap $\rightarrow$ fewer iterations). Hence, tests with high overlap have shorter runtime, but can result in lower success rates.

The second indoor dataset ("Hall") has been acquired in a large workshop that houses construction experiments of the civil engineering department at ETH. It consists of large, regular structures and contains some big machinery. To cover the large scene with dimensions of $\approx 100\,\text{m} \times 30\,\text{m} \times 12\,\text{m}$ eight scans were required, in spite of rather small overlaps ($\approx 40\% - 60\%$). The scans have been captured with a *Faro Focus3D* TLS scanner with a field of view of $360° \times 152.5°$ and a maximum measurement distance of $150\,\text{m}$. Constrained by the tube-like geometry of the room, only immediately adjacent scans have a reasonable overlap, leading to seven scan pairs in total (Figure 4.6).

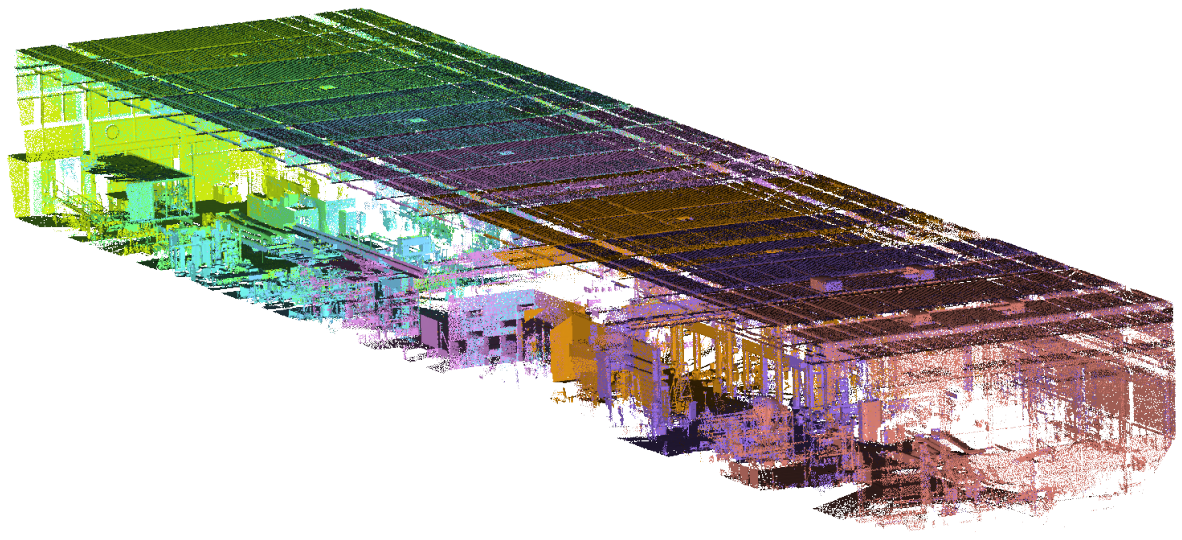


Figure 4.6: Dataset "Hall" of a big workshop: all eight scans have been acquired in a consecutive way and are displayed with a point count of 25%. The wall facing the viewer has been removed for better visualization.

Again, all tests were conducted with four different voxel sizes τ. Due to the very large extent of the room and to keep keypoint numbers at a manageable level, the minimum response threshold was increase to $R^G_{min} = 0.02$ for Dog keypoints and $R^H_{min} = 0.002$ for Harris keypoints, respectively. Estimated overlaps are set to 0.6 for the first three scan pairs, which were acquired with higher overlaps to increase the level of detail in the first part of the hall. The four other scan pairs have been tested with a value of 0.5.

	DoG				**Harris**			
τ in mm	200	100	50	25	200	100	50	25
PSR in %	60.6	60.9	53.1	42.9	45.4	47.4	47.1	15.1
T in s	8	17	38	124	2	6	11	80
k	1100	2500	5600	13200	600	1500	2800	10200

Table 4.4: Results of dataset "Hall": mean success rate PSR, matching time T and number of extracted keypoints k over all scan pairs and trials. All tests were run with four different voxel sizes τ and are based on either DoG or Harris keypoints.

Compared to dataset "Office" the success rate PSR is much lower, reaching a maximum of $\approx 61\%$ based on DoG keypoints and a voxel size of $\tau = 100$ mm. The explanation of this lower PSR lies in the challenging tube-like geometry. As already mentioned previously, the verification by the number of matched points tends to return solutions where the translation between scan pairs is small. In case of scenes with repetitive structures like those in the elongated workshop, it tends to place scans at the same location, with no displacement.

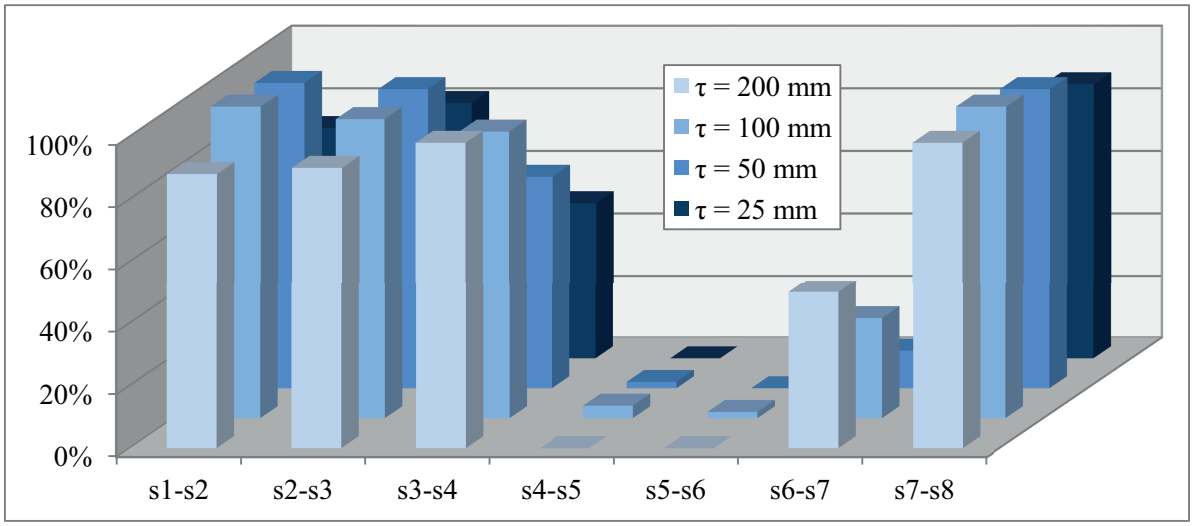


Figure 4.7: Success rate PSR of the different scan pairs of data set "Hall" based on four different voxel sizes τ and DoG keypoints.

A close inspection of Figure 4.7 supports this finding: the vast majority of false registrations occurs in the middle of the hall (s5, s6). Here, scans cover only two walls on the long sides of the workshop, where the structure is repetitive (Figure 4.6). The low success rate PSR of the three scan pairs containing s5 or s6 ($< 50\%$) lowers the overall success rate as shown in Table 4.4, although the remaining pairs are mostly successfully matched (on average in 96% with $\tau = 100\,\text{mm}$ and based on DoG keypoints). The diagnosis is reinforced by comparing results of the DoG and Harris detectors. Overall, matching based on DoG keypoints is more successful than with Harris keypoints, although total keypoint numbers are comparable: the purely geometric Harris keypoints are ambiguous, whereas contrast due to wall texture, floor markings etc. generates some useful DoG keypoints.

	PSR in %	MAE in °	MTE in mm	$RMSE_{\text{M}}$ in mm	$RMSE_{\text{ICP}}$ in mm
s1-s2	98.0	0.8 ± 0.8	331 ± 334	389 ± 316	7 ± 0
s2-s3	96.0	0.8 ± 1.0	377 ± 283	418 ± 282	11 ± 0
s3-s4	68.0	1.9 ± 2.1	926 ± 662	1122 ± 689	8 ± 0
s4-s5	2.0	2.1 ± 1.6	673 ± 0	647 ± 0	5 ± 0
s5-s6	0.0	—	—	—	—
s6-s7	12.0	3.1 ± 3.5	866 ± 456	1340 ± 687	11 ± 0
s7-s8	96.0	1.2 ± 1.4	587 ± 533	616 ± 408	13 ± 0
Mean	**53.1**	$\mathbf{1.2 \pm 1.3}$	$\mathbf{537 \pm 435}$	$\mathbf{621 \pm 410}$	$\mathbf{10 \pm 0}$

Table 4.5: Metric accuracy represented by the *Mean Angular Error* (MAE), *Mean Translation Error* (MTE) and *Root Mean Square Error* between true correspondences before ($RMSE_{\text{M}}$) and after ICP refinement ($RMSE_{\text{ICP}}$). Results for dataset "Hall" using $\tau = 50\,\text{mm}$ and DoG keypoints.

Considering the metric accuracy shown in Table 4.5, MAE, MTE as well as $RMSE_{\text{M}}$ are worse than in dataset "Office". Probable reason for this effect, is the lower overlap between neighbouring scans, leading to less stable base samples and consequently to lower accuracies. Although the average accuracy is three to four times worse than in dataset "Office", the alignment is still good enough to be correctly refined by ICP. The accuracy of $\approx 1\,\text{cm}$ after refinement is again close to the expected measurement accuracy of the scanning device.

4.7.2 Outdoor Applications

The applicability of K-4CPS to outdoor TLS projects is evaluated on two challenging datasets. Dataset "Arch" (Figure 4.8) was acquired in an urban area, dataset "Wood" in a forest (Fig-

ure 4.9). Similar to the experiments on indoor scans, suitable ranges for all parameters were defined according to preliminary tests.

Dataset "Arch" consists of four high-resolution scans with more than twenty million points per acquisition. The scans cover a Roman arch and its surroundings including paths, vegetation, small parks, and a building (Figure 4.8) and were acquired using a *Zoller+Fröhlich TLS Imager 5006i* (cf. dataset "Office"). What makes this dataset challenging are *(i)* low overlap of adjacent scan pairs ($\approx 40\%$), *(ii)* vegetation and *(iii)* numerous artefacts caused by moving people who where visiting the arch during scan acquisition. Only four scan pairs have reasonable overlap for matching.

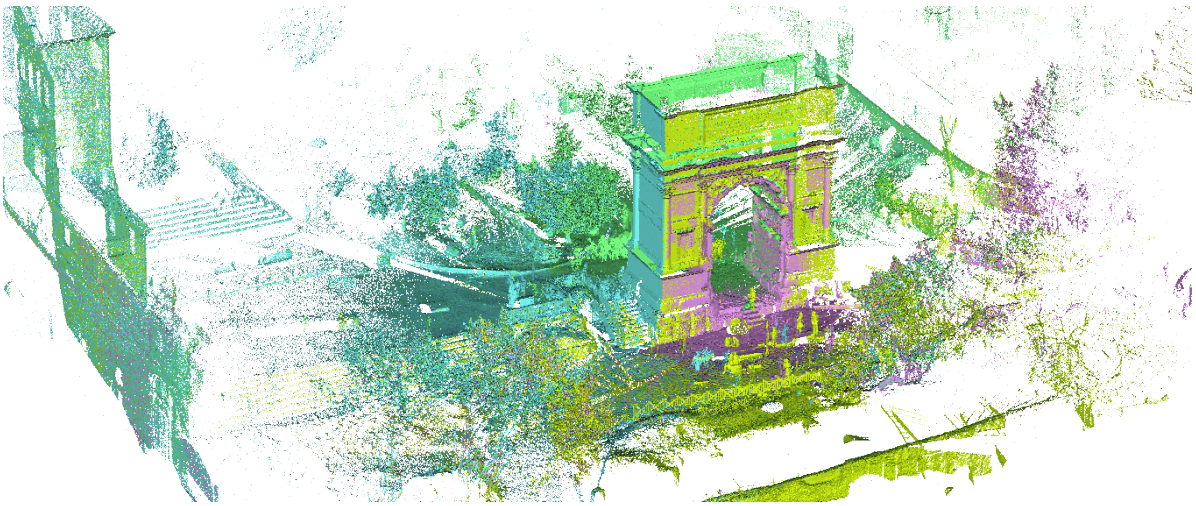


Figure 4.8: Dataset "Arch" with four scans distributed around a Roman arch. Note that walking people cause artefacts and adjacent scan pairs have low overlap. For reasons of visualization data are downsampled to 25% of the original scan resolution.

In a first experiment, the average success rate, matching runtime and number of extracted keypoints averaged over all four scan pairs are analysed. Parameter settings are: for DoG $R^G_{min} = 0.01$ and for Harris $R^H_{min} = 0.001$, representing a good trade-off between quality and number of keypoints; estimated overlap 0.4. Tests are run with three different voxel sizes $\tau = \{0.2\,\text{m}, 0.1\,\text{m}, 0.05\,\text{m}\}$; results are shown in Table 4.6. Recall that indoor experiments achieve best results if the voxel size approximately corresponds to the mean point density in overlap areas. Dataset "Arch" has a rather low mean point density due to the small overlap areas and we thus did not test voxel sizes below $0.05\,\text{m}$.

With $\tau = 50\,\text{mm}$ for DoG keypoints and $\tau = \{50\,\text{mm}, 100\,\text{mm}\}$ for Harris keypoints we achieve high success rates of $PSR > 90\%$. Harris keypoints clearly work better on this dataset, because a lack of distinctive texture in the scan intensities leads to fewer and less reliable DoG keypoints. Runtimes are higher compared to both indoor datasets due to the low overlap

between scan pairs, which requires a higher iteration number w, but still remain feasible for many practical applications.

	DoG			**Harris**		
τ in mm	200	100	50	200	100	50
PSR in %	23.0	56.0	92.0	85.0	100.0	96.5
T in s	12	64	387	40	192	887
k	1300	3600	11000	1900	6500	17700

Table 4.6: Results of dataset "Arch": mean success rate PSR, matching time T and number of extracted keypoints k over all scan pairs and trials. All tests were run with three different voxel sizes τ and are based on either DoG or Harris keypoints.

	PSR in %	MAE in °	MTE in mm	$RMSE_{\mathrm{M}}$ in mm	$RMSE_{\mathrm{ICP}}$ in mm
s1-s2	74.0	2.4 ± 2.3	1275 ± 767	1498 ± 804	22 ± 5
s1-s3	96.0	1.2 ± 1.4	743 ± 481	904 ± 529	15 ± 0
s2-s3	98.0	0.9 ± 0.7	478 ± 418	619 ± 404	18 ± 2
s4-s3	100.0	0.5 ± 0.5	303 ± 169	347 ± 156	12 ± 3
Mean	**92.0**	$\mathbf{1.2 \pm 1.2}$	$\mathbf{660 \pm 437}$	$\mathbf{796 \pm 450}$	$\mathbf{16 \pm 2}$

Table 4.7: Metric accuracy represented by the *Mean Angular Error* (MAE), *Mean Translation Error* (MTE) and the *Root Mean Square Error* between true correspondences before ($RMSE_{\mathrm{M}}$) and after ICP refinement ($RMSE_{\mathrm{ICP}}$). Results are for dataset "Arch", $\tau = 50$ mm and DoG keypoints.

Generally Table 4.7 shows that the metric accuracy is again significantly lower compared to dataset "Hall" in the indoor applications, due to even smaller overlap between adjacent scans. The mean angular accuracy is $\approx 1°$ while the translation error is around 0.7 m. The refinement step reduces the $RMSE$ by a factor of 60 to ≈ 1.6 cm and again reaches the expected scanner accuracy. Note that scan pair s1-s2 has significantly worse accuracies, probably caused by the even lower overlap between this two scans, and the associated smaller base length between the correspondences. The smaller overlap also explains the lower success rate of $PSR = 74\%$ for this scan pair.

An additional experiment with dataset "Arch" addresses the sensitivity to the estimated overlap (with varying values of $\{0.3, 0.4, 0.5\}$), using DoG keypoints and setting $\tau = 50$ mm. The tests uncover that PSR is more sensitive to the estimated overlap compared to indoor dataset

"Office". An estimated overlap of 0.3 gives clearly better results with success rates of 100%, but at the cost of much larger runtimes ($> 1'000$ s).

To evaluate K-4PCS under extreme conditions we perform experiments on dataset "Wood" of a forested area (Figure 4.9). It consists of two groups of three scans. All scans in a group overlap with $\approx 50\%$ and mainly contain bushes and trees. The data has been acquired with the a *Faro Focus3D* laser scanner (cf. dataset "Hall").

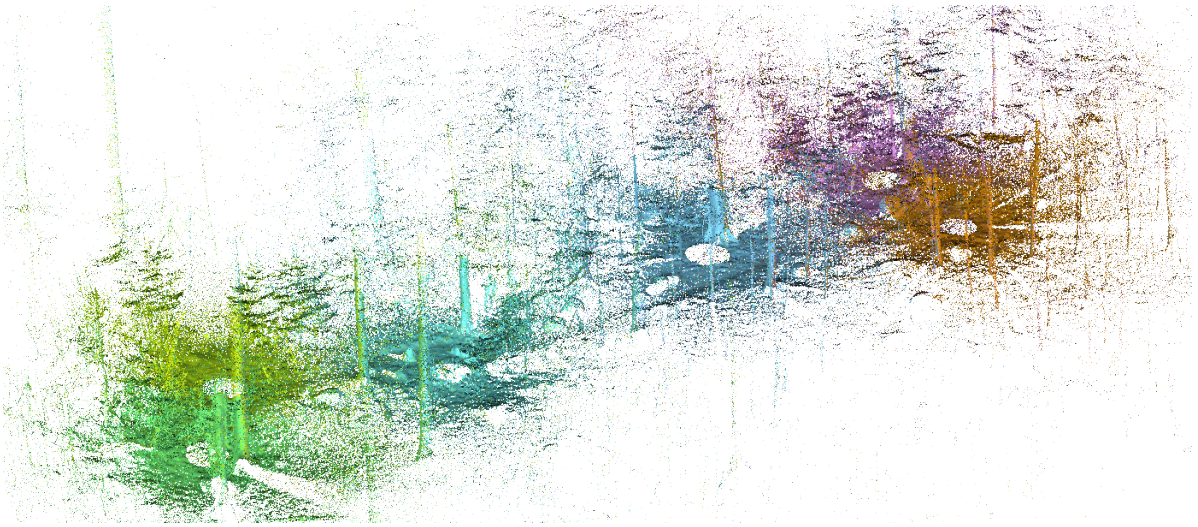


Figure 4.9: Dataset "Wood" of a forested area. Unstructured trees and bushes make this dataset challenging. The illustration only displays 1% of the points of each scan.

	DoG			**Harris**		
τ in mm	200	100	50	200	100	50
PSR in %	32.7	45.3	64.7	66.0	66.3	68.0
T in s	8	34	116	25	44	100
k	1100	3500	9200	1600	4000	8700

Table 4.8: Results of dataset "Wood": mean success rate PSR, matching time T and number of extracted keypoints k over all scan pairs and trials. All tests were run with three different voxel sizes τ and are based on either DoG or Harris keypoints.

Again, we run the same tests. Estimated overlap is fixed to 0.5 for all scan pairs. The large amount of vegetation with strongly varying reflectance and geometry heavily increases the number of keypoints. To keep the point count at a reasonable level the minimum response is set to $R^G_{min} = 0.02$ and $R^H_{min} = 0.002$. Table 4.8 shows that results on this challenging dataset are $\approx 30\%$ points below those of dataset "Arch". Figure 4.10 reveals that automated

coarse alignment for two scan pairs (out of six in total) almost completely fails. If we put those two aside and only look at average success rates over all trials of the four remaining pairs, we achieve $PSR = 96\%$ for DoG and $PSR = 99\%$ for Harris keypoints, respectively. Similar to indoor dataset "Hall" of the tube-like workshop, failure cases are caused by repetitive structures, which, somewhat surprisingly, also appear in the forest setting.

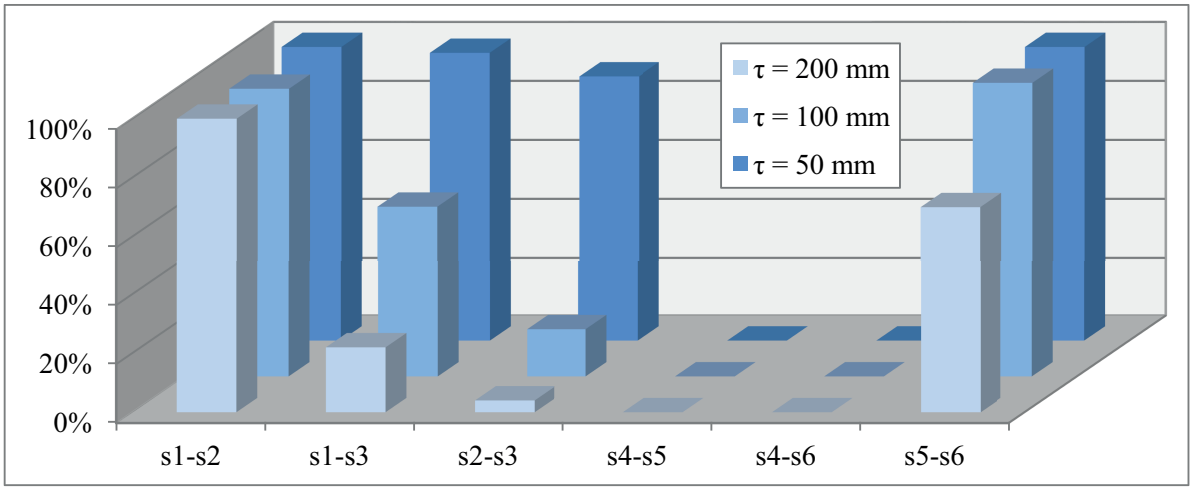


Figure 4.10: Success rate PSR of the different scan pairs of data set "Wood" based on three different voxel sizes τ and DoG keypoints.

	PSR in %	MAE in °	MTE in mm	$RMSE_{\mathrm{M}}$ in mm	$RMSE_{\mathrm{ICP}}$ in mm
s1-s2	100.0	2.5 ± 2.2	797 ± 367	924 ± 367	4 ± 0
s1-s3	98.0	3.8 ± 3.0	1102 ± 408	1463 ± 569	5 ± 0
s2-s3	90.0	1.8 ± 1.5	582 ± 268	718 ± 321	6 ± 0
s4-s5	0.0	—	—	—	—
s4-s6	0.0	—	—	—	—
s5-s6	100.0	1.6 ± 1.4	469 ± 198	523 ± 207	2 ± 0
Mean	**64.7**	**2.5 ± 2.1**	**740 ± 311**	**909 ± 366**	**4 ± 0**

Table 4.9: Metric accuracy represented by the *Mean Angular Error* (MAE), *Mean Translation Error* (MTE) and the *Root Mean Square Error* between true correspondences before ($RMSE_{\mathrm{M}}$) and after ICP refinement ($RMSE_{\mathrm{ICP}}$). Here, the results of the tests based on dataset "Wood", $\tau = 50$ mm and DoG keypoints are shown.

The achieved metric accuracy is visualized in Table 4.9. The values are again based on $\tau = 50$ mm and DoG keypoints. On the one hand, the angular accuracy is worse compared to previous tests with other datasets. The reason for this effect is probably the limited extent of the point clouds in lateral direction (w.r.t. the main displacement of the scanner, see Figure 4.9).

This in combination with the low overlap of $\approx 50\%$ limits the base length of candidate keypoints and consequently reduces angular accuracy. Standard ICP brings the accuracy down to < 1 cm.

After comprehensive experiments with two indoor ("Office", "Hall") and two outdoor datasets ("Arch", "Wood"), the major bottleneck that remains apparently is how to deal with symmetries and repetitive structures. To get a deeper understanding of this potential failure scenario we perform additional experiments on simulated scans.

4.8 Evaluation on Synthetic Data

Experimental results on both indoor and outdoor scans reveal that the large majority of failure cases is caused by translationally or rotationally symmetric scene content. To investigate this issue in detail under controlled conditions, we turn to synthetic datasets. To that end we have designed a LiDAR point cloud simulator for simple indoor scenes (Section 4.8.1) and have performed multiple tests (Section 4.8.3).

4.8.1 Simulator

This algorithm simulates TLS point clouds taken indoors. A simple symmetric room is set up with six major planes Pl_j, $j = 1 \dots 6$, representing ceiling, floor, and walls. The dependency of K-4PCS on objects that break symmetry is analysed by introducing an additional seventh plane Pl_7 which cuts off a part of the symmetric room (Figure 4.11). To address different levels of symmetry, the normal $\mathbf{n_7}$ and the orthogonal distance $d_7^{\perp}$ (with respect to the room origin) of the seventh plane are varied. The geometrical construction of a virtual scan is carried out as follows. The virtual scanner is placed inside the room, with its orientation set randomly and its inclination is limited to vary only up to a few degrees around zero to simulate an approximately levelled instrument. After setting the angular resolution of the simulated scanner, a direction vector $\mathbf{r}_i$ for each laser beam i is calculated. Then starting from the instrument position $\mathbf{x_0}$, all beams are intersected with each plane Pl. The length of the beam is obviously $\|\mathbf{p} - \mathbf{x_0}\|$ and can directly be calculated from

$$\|\mathbf{r}_i\|_j = \frac{d_j^{\perp} - \mathbf{n}_j \cdot \mathbf{x_0}}{\mathbf{r}_i} \cdot \mathbf{n}_j \quad . \tag{4.11}$$

Since LiDAR is a line-of-sight instruments, the intersection point with the shortest distance $\min_j (\|\mathbf{r}_i\|_j)$ is kept for every beam i. To simulate the range measurement accuracy of a LiDAR instrument, Gaussian noise with $\sigma = 5$ mm is added to the distance.

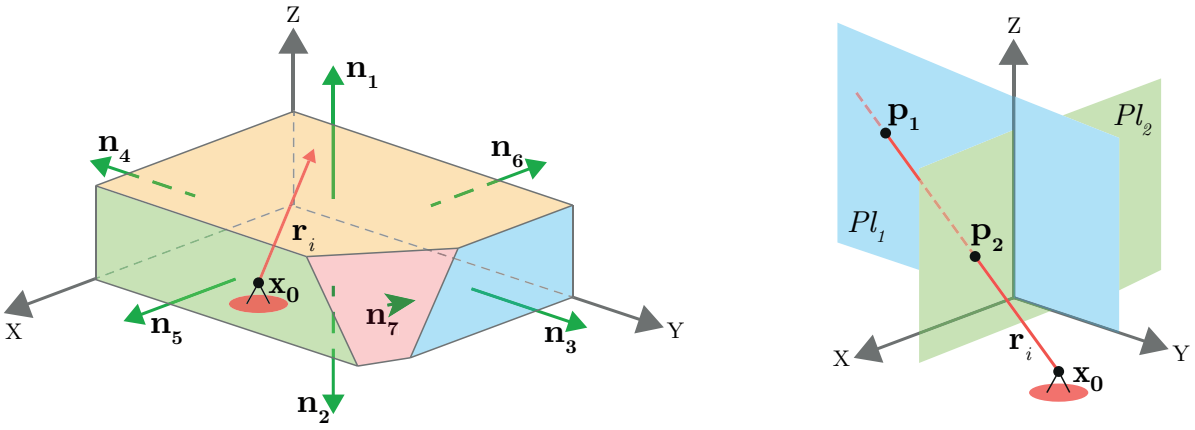


Figure 4.11: Principle of the TLS simulator. Left: geometrical setup of the synthetic indoor room with six major planes and symmetry-breaking seventh plane (red). Right: point cloud generation by searching for closest intersection points $\mathbf{p}$ between the seven room planes Pl and the measurement beams $\mathbf{r}_i$ sent from a scan station $\mathbf{x_0}$.

Working with DoG keypoints additionally requires a grey value. For real scans this is the intensity of the received measurement signal recorded by the LiDAR instrument. We assign intensities to the synthetic point cloud via small patches from real scan intensity images of indoor walls, floors, and ceilings. A texture image per plane is generated by randomly assembling small patches until the required image size (given by the room dimension) is reached. Image resolution is set to 5 mm to ensure that neighbouring LiDAR points get intensities information from different pixels. Examples of texture images are shown in Figure 4.12.

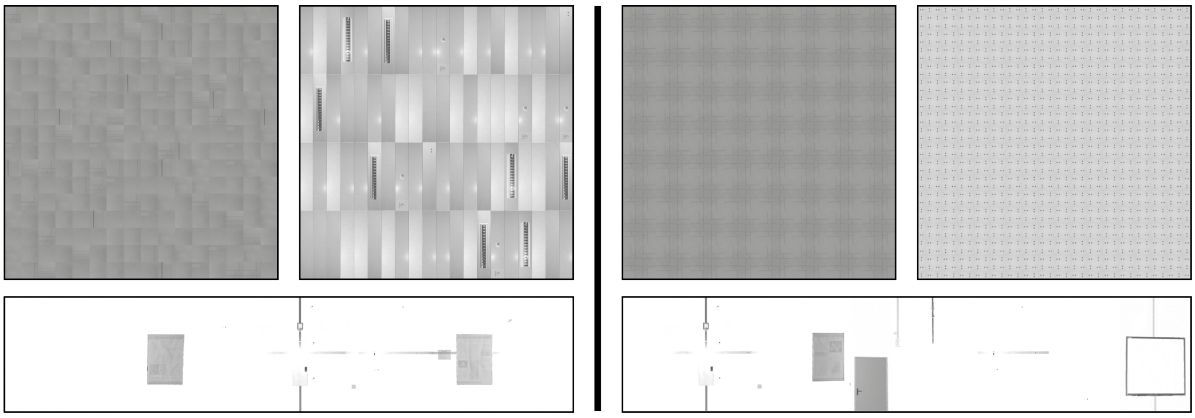


Figure 4.12: Generated texture images (floor, ceiling and a wall example) to map intensity values onto the point cloud. Left: Texture images of setup 1 to simulate a geometrically symmetrical room with asymmetric textured planes. Right: Texture images of setup 2, which simulates a geometrically and radiometrically symmetrical room. Note that texture images of walls do not need to be symmetric, because rotation- symmetry is achieved by mapping the same texture onto all walls.

When looking up the synthetic intensities, original texture values are modified based on the incidence angle of the beam on the wall, to mimic the behaviour of a real scanner. We apply an exponential damping function to decrease intensity with smaller incidence angles.

4.8.2 Synthetic Datasets

We generate two different collections of synthetic datasets (i.e. setups) to evaluate the influence of symmetric scene content on registration performance with respect to geometry and to radiometry. Both setups are based on a quadratic room with a side length of $20\,\mathrm{m}$ and a height of $4\,\mathrm{m}$.

We first address geometric symmetry in setup 1. Radiometric (texture) variation is created by randomly assembling different small patches to texture images. This is repeated for each room plane so that all planes are differently textured. The room's geometric layout is thus $90°$ rotationally symmetric with respect to its vertical axis, as opposed to its radiometric layout, which is not symmetric. To evaluate the influence of concurrent radiometric and geometric symmetry, setup 2 has rotationally symmetric texture images for ceiling and floor. All walls are textured with the same image (which in itself does not have to be symmetric). The room of setup 2 is $90°$ rotationally symmetric (with respect to the vertical axis of the room) in terms of both geometry and radiometry.

Examples of the applied texture images are shown in Figure 4.12. In Figure 4.13 an example of a room of setup 1 is shown left whereas a room of setup 2 is shown right. Note that both examples already contain the seventh plane that cuts off the corner (pointing towards the reader) to break symmetry.

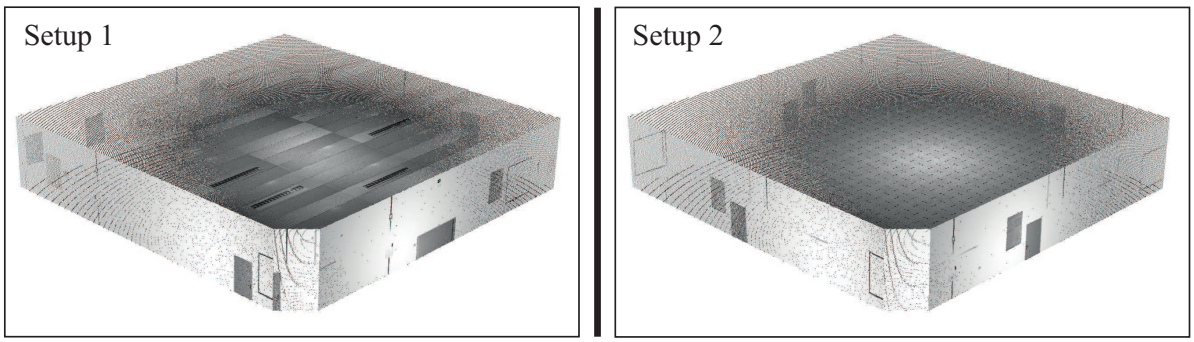


Figure 4.13: Examples of the simulated point clouds with a symmetry-breaking plane at the room corner in front. Left: setup 1 with geometrical symmetric room but asymmetric radiometry (texture). Right: example of setup 2 with repetitive texture patterns on the walls leading to a $90°$ rotational symmetry in geometry and radiometry.

Six datasets are simulated for both setups 1 and 2. More precisely, we test six different settings of the symmetry-breaking plane that can be divided into two test scenarios *A* and *B*, each including a zero case (Figure 4.14) without the seventh plane.

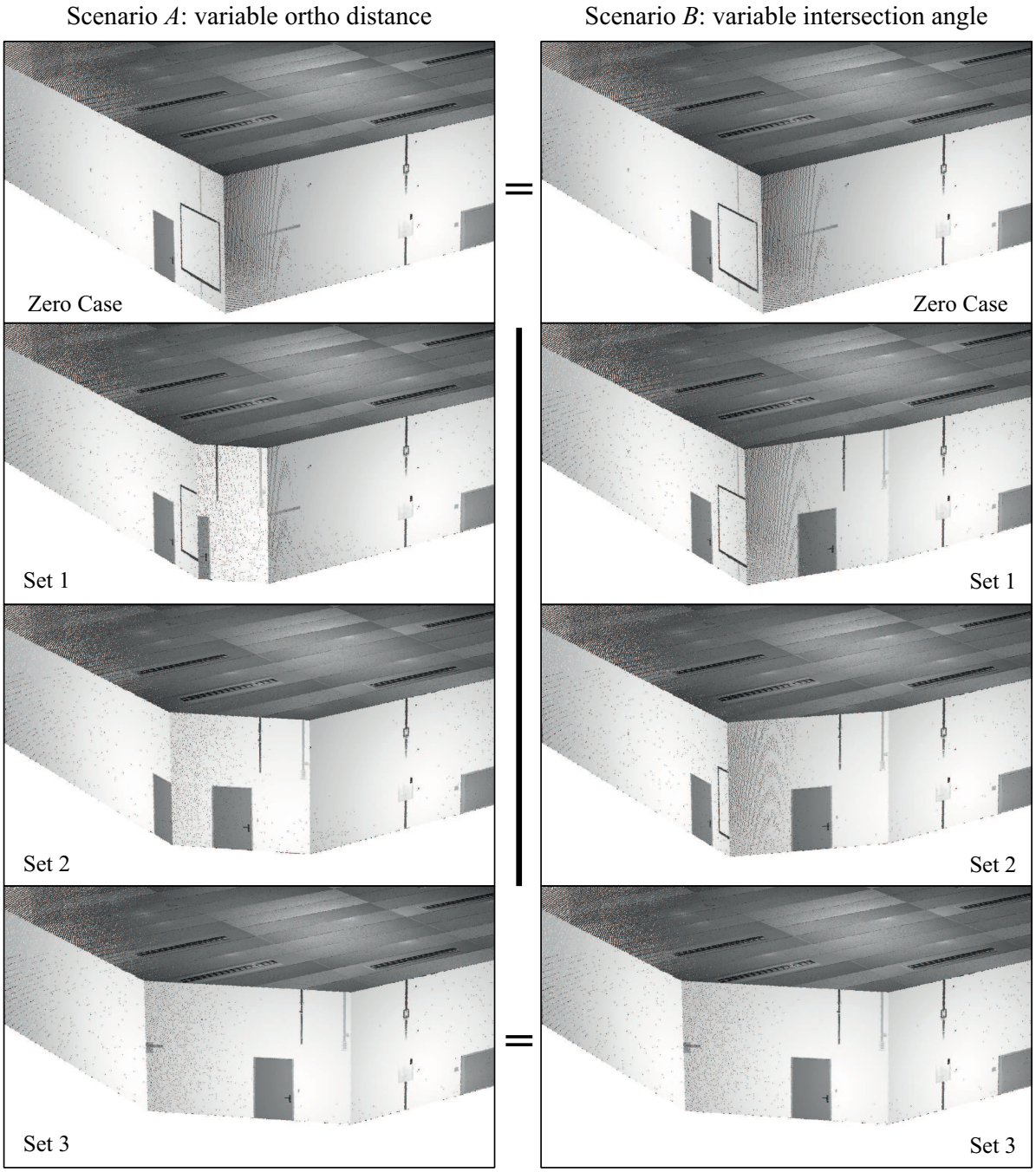


Figure 4.14: Each of the two setups consists of six datasets with variable properties of the symmetry-breaking additional plane. In test scenario *A* (left) the orthogonal distance is increased by steps of 1 m whereas in test scenario *B* (right) the intersection angle with a room wall is varied. The images show setup 1, without radiometric symmetry.

Scenario *A* simply moves the cutting plane into the room in three steps (orthogonal distances to the room corner of 1 m, 2 m, and 3 m) whereas the angles to the walls on both sides are $45°$ (left in Figure 4.14). In *B* a vertical plane cuts off the same room corner as in *A* but with three different angles of $11.25°$, $22.5°$, and $45°$ (right in Figure 4.14), while the intersection line between the cutting plane and the reference wall is kept.

All rooms (six for setup 1 and six for setup 2) are virtually scanned from five positions, such that the room is well covered. The five scan positions are the same over all experiments, so as to make the different runs directly comparable.

4.8.3 Results

As the simulated environment lacks geometric clutter like chairs or tables, the use of the 3D Harris detector is not reasonable. Hence, all tests with simulated data are performed with 3D DoG keypoints. In the following, we will first show results of setup 1. It addresses the challenge of geometrical symmetry, while the radiometry (LiDAR intensity texture) is randomly assigned. Thereafter, we describe setup 2 with radiometric and geometrical symmetries.

To evaluate the influence of the symmetry-breaking plane, we first analyse success rate (PSR) of the zero case without any additional plane.

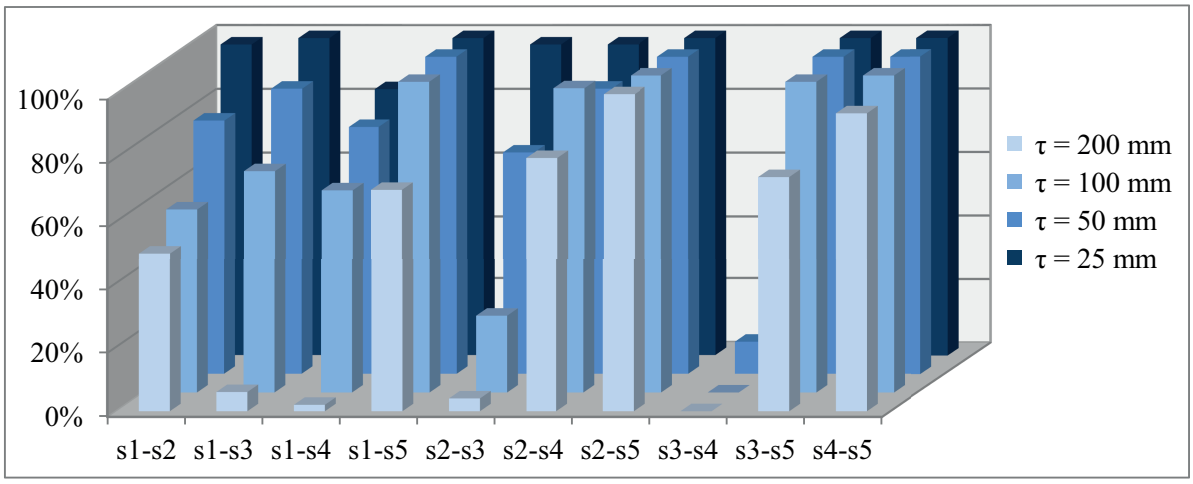


Figure 4.15: Success rates PSR (vertical axis) of the zero case of setup 1 per scan pair (horizontal axis), as a function of voxel size τ (depth/colour).

Figure 4.15 clearly shows an increasing PSR with decreasing voxel grid size τ, for all scan pairs. This dependency has also been observed in real TLS datasets (see Section 4.7). Obviously, τ strongly influences the amount and repeatability of keypoints and consequently the success rate

PSR. Scans s3-s4, placed on opposite room corners and close (<5 m) to the walls, stand out with very low PSR, especially at higher τ. Recall that K-4PCS is somewhat biased towards solutions with small translation. In case of opposite room corners, where the large majority of keypoints is detected in the scanner's near field, the method tends to return $180°$ rotated solutions.

The influence of the degree of symmetry for setup 1 on the success rate is shown in Figure 4.16 for both scenarios *A* and *B*.

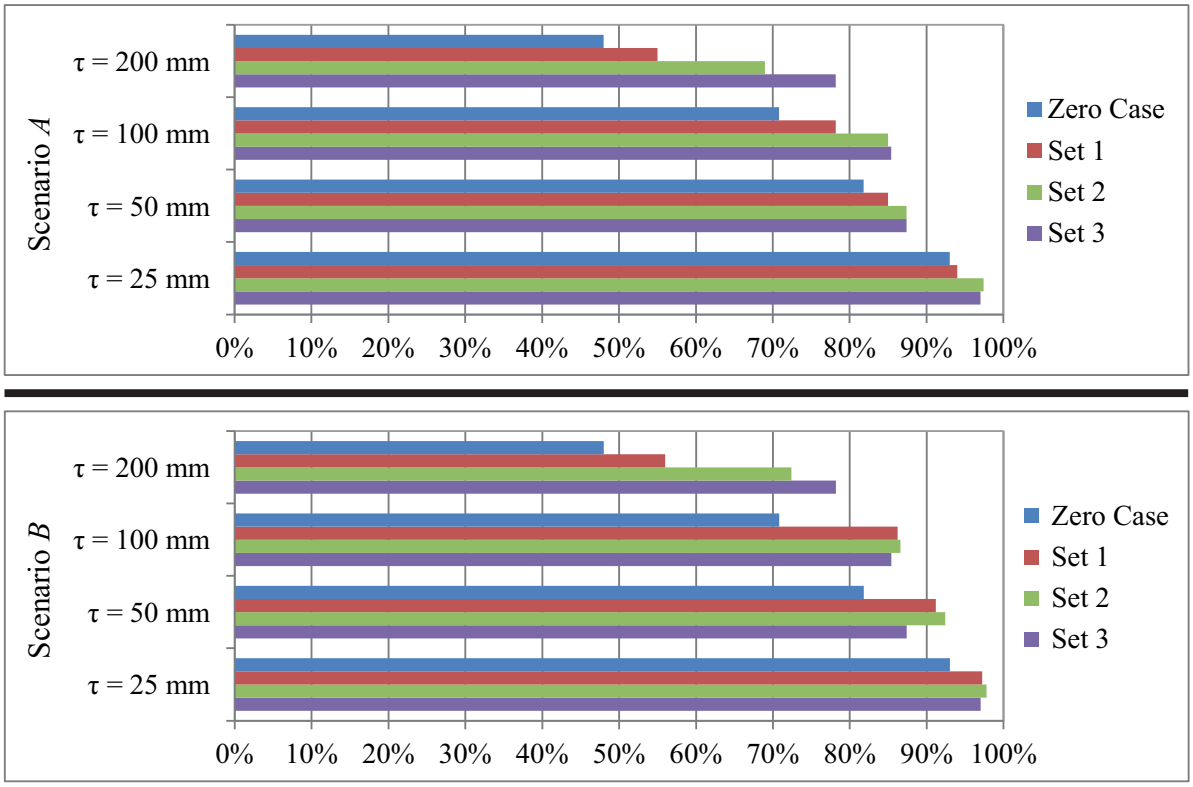


Figure 4.16: Mean success rates PSR for scenarios *A* (top) and *B* (bottom) of setup 1, grouped with respect to voxel grid size τ. Blue bars indicate the zero case without any additional plane.

The zero case already delivers relatively high success rates for setup 1 (with only geometrical symmetry and random texture). Breaking geometrical symmetry by moving the additional plane into the room further improves results significantly. It is evident that the use of DoG keypoints, which rely on texture, stabilizes the algorithm.

However, as soon as texture becomes repetitive and symmetric, the picture drastically changes, as can be seen in Figures 4.17 and 4.18. In setup 2 with symmetric geometry and texture, the results severely degrade. Registration for the zero case without any additional plane that breaks symmetry almost completely fails for most scan pairs (Figure 4.17).

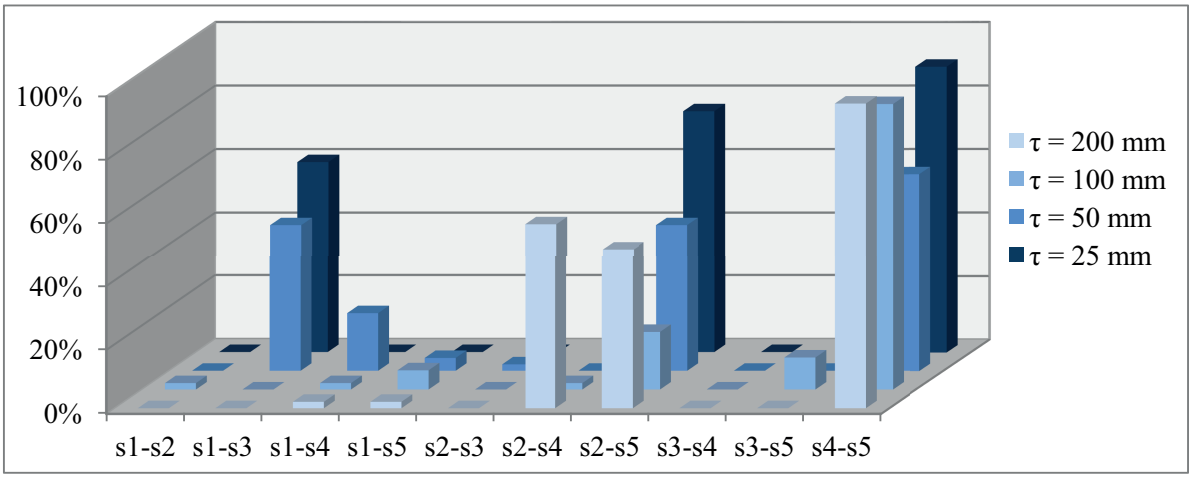


Figure 4.17: Success rates PSR (vertical axis) of the zero case of setup 2 per scan pair (horizontal axis), as a function of voxel size τ (depth/colour).

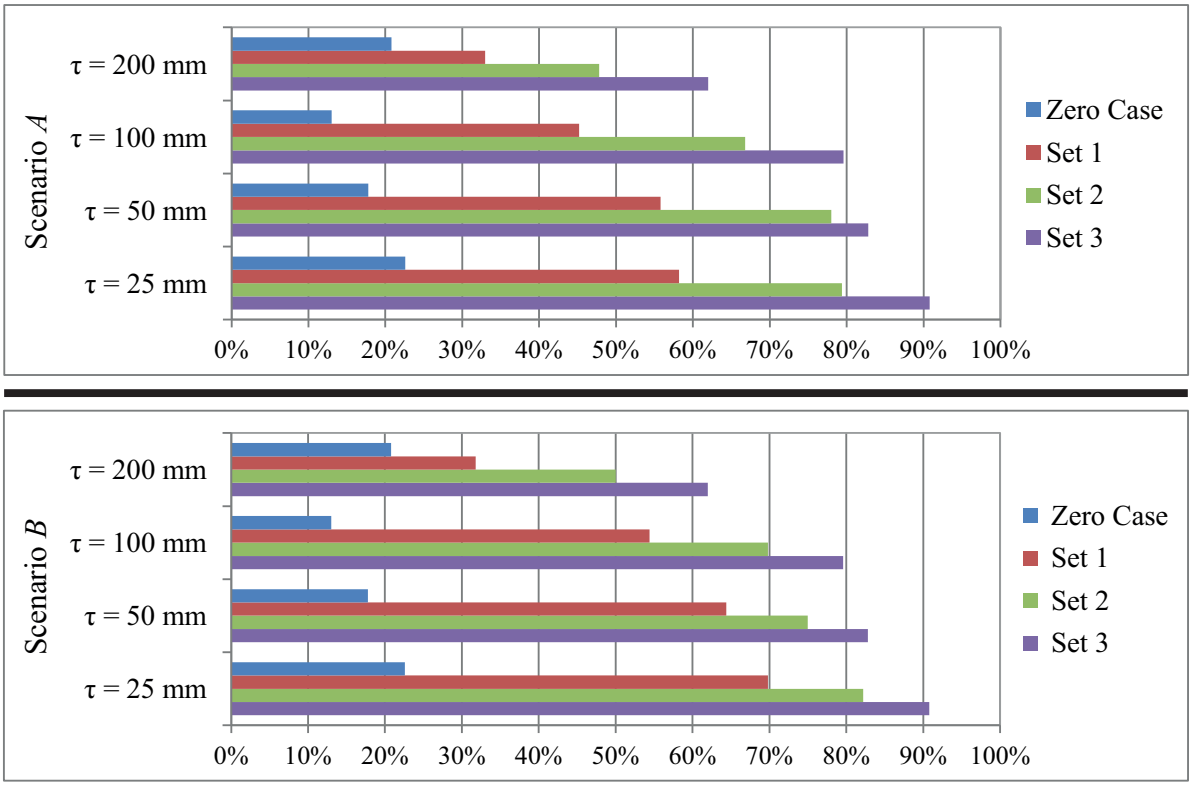


Figure 4.18: Mean success rates PSR for scenarios A (top) and B (bottom) of setup 2, grouped with respect to voxel grid size τ. Blue bars indicate the zero case without any additional plane.

However, the more the plane is moved into the room and breaks the symmetry, the higher the success rate (Figure 4.18). As expected complete or near-complete symmetry causes matching failures. As asymmetry is introduced matching success gradually increases, but from

the experiment we conclude that a significant amount of asymmetry is needed to reach high success rates.

4.9 Conclusions and Outlook

We have presented and analysed the *Keypoint-based 4-Points Congruent Sets* (K-4PCS) method for coarse registration of TLS point clouds. The strategy, to represent original raw scans as sparse clouds of 3D keypoints (DoG, Harris) and register these with an adapted version of Aiger's 4PCS approach yields high success rates at reasonable computation times, as long as the scanned scene is not too repetitive or symmetric. Experimental results on multiple indoor and outdoor test datasets have shown sufficient geometrical registration accuracy for a subsequent ICP refinement.

Two main challenges remain for future work: *(i)* to further bring down processing times, and *(ii)* to improve robustness against scenes with translational or rotational symmetries.

The straight-forward parallelization over independent random samples already yields realistic runtimes in most cases. Still, low overlap combined with the necessity to use a large number of keypoints lead to processing time > 10 min. One possibility to further reduce computation times is to adopt a multi-scale approach, in order to solve the bulk of the work with fewer points and iteratively refine the registration with more data. A further idea to be explored is to cluster candidate solutions based on the transformation parameters, with the hope that a small number of promising candidate transformations will emerge quicker and fewer samples are wasted.

Experiments with real as well as synthetic scans demonstrate that robustness to repetitive and symmetric structures must still be improved. While extreme cases do not have a unique solution, and are difficult even for humans, we still lose cases that do contain the necessary structure. One possible direction is to augment the keypoints with descriptors, so as to overcome geometric ambiguity with local appearance information. A further idea is to include prior information, such as expected or approximate distances between scans.

Finally, on a more conceptual note, it would be interesting to find better measures for the goodness of the solution than the number of matching points. Our experiments show that there are cases where a larger overlap can be achieved with a wrong solution, i.e., the (widely employed) objective function does not yet adequately model all aspects of the registration problem.

Chapter 5

Extensions and Improvements on Keypoint-based TLS Registration

Fast Registration of Laser Scans with 4-points Congruent Sets – What Works and What Doesn't

Pascal Willy Theiler, Jan Dirk Wegner, Konrad Schindler

ISPRS Annals, Photogrammetric Computer Vision (PCV), ISPRS Technical Commission III Midterm Symposium, Zurich, Switzerland, 2014.

(Author version; for publishers' typeset version please refer to the original conference paper)

5.1 Abstract

Sampling-based algorithms in the mould of RANSAC have emerged as one of the most successful methods for the fully automated registration of point clouds acquired by terrestrial laser scanning. Sampling methods in conjunction with 3D keypoint extraction, have shown promising results, e.g. the recent K-4PCS (Theiler et al., 2013). However, they still exhibit certain improbable failures, and are computationally expensive and slow if the overlap between scans is low.

Here, we examine several variations of the basic K-4PCS framework that have the potential to improve its runtime and robustness. Since the method is inherently parallelizable, straightforward multi-threading already brings down runtimes to a practically acceptable level (seconds to minutes). At a conceptual level, replacing the RANSAC error function with the more principled MSAC function (Torr and Zisserman, 2000) and introducing a minimum-distance prior to counter the near-field bias reduce failure rates by a factor of up to 4. On the other hand,

replacing the repeated evaluation of the RANSAC error function with a voting scheme over the transformation parameters proved not to be generally applicable for the scan registration problem.

All these possible extensions are tested experimentally on multiple challenging outdoor and indoor scenarios.

5.2 Introduction

Static *Terrestrial Laser Scanners* (TLS) have become standard devices to acquire 3D data for a wide range of applications like as-built mapping of large industrial facilities, documentation of heritage sites, or manufacturing. Multiple scans from different viewpoints usually have to be acquired to fully cover complex objects. To combine all these scans into a single point cloud, the relative orientation between them (rigid-body transformation with six degrees of freedom) has to be found. This process of aligning all scans in a common reference system is usually called registration and is a prerequisite for any further analysis like 3D reconstruction or semantic object segmentation. In practice, scan registration is either done completely manually or automatically based on artificial markers placed in the scene during acquisition (e.g. Akca, 2003; Franaszek et al., 2009). Although this procedure solves the registration, it is time-consuming in the field, markers occlude small scene parts and often have to be removed from the result, they must remain static until completion of all scans, and they should never be occluded by moving objects like cars etc. Thus, much effort has been spent to develop fully automated, marker-less methods for scan registration and to avoid artificial markers completely.

Various solutions for fine registration of scans have been proposed, most notably the *Iterative Closest Point* (ICP) algorithm (Besl and McKay, 1992; Chen and Medioni, 1992) and its variants (e.g. Bergevin et al., 1996; Bae and Lichti, 2004; Minguez et al., 2006; Censi, 2008). On the contrary, no practical, generally applicable and efficient solution to coarse register big TLS point clouds is available yet. ICP-like approaches minimize a sum of Euclidean distances between potentially corresponding points, i.e., they aim at locally optimizing a highly non-convex objective function. Consequently, applying ICP directly to scans with arbitrary relative orientation will in most cases fail (see Pottmann et al., 2006, for details on convergence properties). A common strategy is to first roughly register scans with a robust, but less accurate method, and let ICP take over from that initial solution.

Here, we deal with the coarse registration to obtain an initial solution, which must only be good enough for standard ICP to accomplish fine registration. What makes coarse registration of arbitrarily oriented TLS point clouds difficult are *(i)* unevenly distributed scan points due

to the polar measurement principle, *(ii)* the sheer amount of data (millions of points), *(iii)* the often limited overlap and strong change of viewpoint between neighbouring scans, to save acquisition time in the field.

To achieve coarse TLS point cloud alignment we build upon work of Theiler et al. (2013) that adapts the *4-Points Congruent Sets* (4PCS; Aiger et al., 2008) and applies it to clouds of keypoints extracted from raw scans. Although that method is already a lot faster than brute-force random sampling, it is still too slow for routine use in practice; and it suffers from the near-field bias, i.e., it is biased towards wrong solutions with small translation (nearly identical scanner positions) because more keypoints are found close to the scanner, where more detail is observed.

As a simple measure to drastically speed up the computation we exploit the fact that the method is parallelizable by design and can be distributed across multiple processors now routinely available in standard desktop computers. To improve its robustness we replace the $0-1$ error function of RANSAC with a truncated least-squares error (that combination is known as MSAC), and introduce a simple prior that favours scan positions above a certain minimum distance to counter the near-field bias. These two measures both do not increase the runtime.

Finally, it has been suggested that one can avoid the repeated evaluation of the full cost function for each random sample, and instead find re-occurring solutions via Hough-style voting (e.g. Torii et al., 2011). The intuition is that correct solutions have higher support and will therefore be found more often, whereas there is no systematic structure in the incorrect samples. It turns out that this strategy does not work well for the scan registration problem, and seriously degrades the result.

Source code within the framework of the open-source *Point Cloud Library* (PCL; Rusu and Cousins, 2011) as well as test data will be made available after publication.

5.3 Related Work

A variety of approaches exist for fully automated, coarse registration of point clouds without artificial markers. Most of them have a common structure: first, extract a set of features or keypoints; second, sample subsets and match overlapping areas; and third, align the point clouds with the transformation estimated from the best match. Finally, coarsely aligned point clouds are fine-registered with standard ICP (Besl and McKay, 1992) or some equivalent. While this general workflow is usually the same, different techniques have been proposed for each step.

One strategy for feature extraction is to describe the scene geometry via planar surfaces (e.g. Dold and Brenner, 2006) or to search for salient directions (e.g. Novák and Schindler, 2013), which can be augmented with 2D features (Zeisl et al., 2013).

Another natural strategy is to compute 2D keypoints in range or intensity images of scans (Böhm and Becker, 2007; Kang et al., 2009). Generally, methods that strongly rely on surface geometry or on 2D features are sensitive to strong viewpoint changes and become unreliable in scenes with large depth range and/or frequent self-occlusions. Representing point clouds with a sparse set of 3D features appears to be more robust to viewpoint changes (Flint et al., 2007; Allaire et al., 2008; Lo and Siebert, 2009; Flitton et al., 2010).

Variants of RANSAC (Fischler and Bolles, 1981) are often used to find potentially corresponding points. For example, Rusu et al. (2009) proposed a method they coined *Sample Consensus Initial Alignment* (SAC-IA) to match corresponding point triplets between clouds of n points. However, its complexity is $\mathcal{O}(n^3)$, which quickly becomes infeasible if dealing with laser scans of several million points. Aiger et al. (2008) reduced runtime complexity to $\mathcal{O}(n^2)$ by adding a fourth point to the triplets such that all four points are roughly coplanar. This method is efficient for moderately large point clouds and provides high success rates if points are uniformly distributed.

Yang et al. (2013) recently proposed a globally optimal solution of ICP named Go-ICP. They show that coarse and fine registration can be done in one step by global optimization with a branch -and-bound scheme that avoids local minima. The method has been demonstrated on point clouds with a few thousand points, but does not scale up to larger data sets.

In the present paper, we evaluate several possible extensions of the K-4PCS method (Theiler et al., 2013), a variant of the 4PCS method that matches sparser clouds of 3D keypoints extracted from the original data. The following extensions are tested:

- a second, purely geometric keypoint detector that captures discriminative, local point distributions,
- parallelization and nested clustering to speed up the computation,
- MSAC (Torr and Zisserman, 2000) instead of plain RANSAC, that uses a truncated least-squares penalty rather than a binary inlier/outlier threshold,
- a refined cost function that penalizes unlikely solutions where the scanner positions are too similar.

Overall, we achieve a significant speed-up, while at the same time decreasing the failure rate of the registration procedure.

5.4 K-4PCS: Coarse Point Cloud Registration

In an earlier work we proposed the *Keypoint-based 4-Points Congruent Sets* method (K-4PCS; Theiler et al., 2013). In a nutshell, it uses sparse clouds of 3D keypoints, which are matched with the *4-Points Congruent Sets* (4PCS) algorithm of Aiger et al. (2008). In this paper, we present modifications and extensions to the original K-4PCS, which focus on its main bottlenecks, namely limited robustness against uneven keypoint distribution as well as repeated structures, and long computation times in case of low scan overlap. We start with a brief overview of the original K-4PCS (i.e., 3D keypoint extraction, keypoint matching), more details can be found in Theiler et al. (2013).

5.4.1 3D Keypoint Extraction

Although the matching algorithm of Aiger et al. (2008) is computationally more efficient than simple three-point sampling, handling standard TLS point clouds with millions of points is still infeasible. Therefore, the point cloud size is significantly reduced by extracting a (much smaller) set of discriminative keypoints, while discarding all other points.

First a voxel grid filter is applied, which divides the 3D space into a regular grid of blocks (i.e. voxels) of size τ. Each block is represented by the centroid of all inlying scan points. On the one hand, this ensures a drastic reduction of the point count, and on the other hand it mitigates the strongly unequal point density inherent in polar acquisition methods.

Then, 3D keypoints are extracted, which are a lot sparser than the full voxel grid, but repeatable and thus useful for matching. Note, K-4PCS matching does not involve local descriptors of keypoints, rather it relies solely on the relative keypoint positions. Registration is thus independent of a particular kind of keypoint detector. We test (see Section 5.6) with 3D *Difference-of-Gaussians* (DoG) and 3D *Harris* keypoint detectors, which are described in the following.

3D DoG Keypoints. DoG keypoints were introduced by Lowe (1999) as part of an image matching framework. Being invariant to scaling, rotation as well as translation, DoG keypoints are standard in 2D image processing, especially in combination with the SIFT descriptor. Here, we extract DoG keypoints directly in 3D to avoid points that are unstable across different view points (e.g. object silhouettes, depth discontinuities). The 3D DoG keypoint extractor (Rusu and Cousins, 2011) uses LiDAR return intensities I to select points with high contrast to their direct neighbours in 3D space. At different blur levels τ_k (with $k = 1 \dots m$), a Gaussian

response G for each point is calculated, taking into account all points in the neighbourhood $\mathcal{N}$ with Euclidean distances $d < 3 \cdot \tau_k$ (Eq. (5.1)).

$$\begin{aligned} G_i(\tau_k) &= \frac{1}{\sum\limits_{j \in \mathcal{N}} g_j(\tau_k)} \cdot \sum_{j \in \mathcal{N}} g_j(\tau_k) \cdot I_j \\ g_j(\tau_k) &= e^{-0.5 \cdot d_j^2 / \tau_k^2} \end{aligned} \tag{5.1}$$

To obtain DoG responses R^G at each point, adjacent scales are subtracted (Eq. (5.2)),

$$R_i^G(\tau_k) = G_i(\tau_{k+1}) - G_i(\tau_k) \tag{5.2}$$

A valid keypoint is found, if the DoG response of the point is a local maximum or minimum in a $3 \times 3 \times 3$ neighbourhood, and exceeds a given threshold R_{min}^G.

3D Harris Keypoints. In contrast to the original Harris corner detector (Harris and Stephens, 1988) based on image gradients, the 3D version (Rusu and Cousins, 2011) starts from local point normals. Thus, only geometrical properties are used and, unlike DoG, the detector is independent of laser intensities. For a point i, the local normal $\mathbf{n}_i = (n_{xi}, n_{yi}, n_{zi})^\top$ is calculated using the set $\mathcal{N}$ of neighbours within $r = 3 \cdot \tau$. The covariance matrix at each point is determined using the normals of all points inside neighbourhood $\mathcal{N}$. Then for each point, the Harris response

$$R^H = \frac{1}{|\mathcal{N}|} \cdot \det(\sum_{i \in \mathcal{N}} \mathbf{n}_i \cdot \mathbf{n}_i^\top) - \gamma \cdot \frac{1}{|\mathcal{N}|} \cdot \mathrm{tr}(\sum_{i \in \mathcal{N}} \mathbf{n}_i \cdot \mathbf{n}_i^\top)^2 \tag{5.3}$$

is determined, here γ is a constant. Similar to the DoG detector, keypoints are local maxima in response space that must exceed a threshold R_{min}^H.

5.4.2 Geometrically Constrained Keypoint Matching

K-4PCS registration (Theiler et al., 2013) builds upon the *4-Points Congruent Sets* (4PCS) algorithm, which is an efficient matching algorithm on the basis of geometric constraints. The straight-forward solution to register two 3D point clouds is to find two sets of congruent point triplets and to use them for deriving the parameters of a rigid-body transformation. However, this approach has computational complexity of at least $\mathcal{O}(n^3 \log n)$ with n the number of points in the point cloud (Irani and Raghavan, 1996). Aiger et al. (2008) showed that adding a fourth point (which is approximately coplanar to the base triplet) and matching quadruples reduces runtime to $\mathcal{O}(n^2)$.

Matching of point quadruples is in general similar to the standard approach based on three corresponding points (Figure 5.1). From a source point cloud $\mathcal{S}$, a four-point base $\mathcal{B}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) \in \mathcal{S}$ is randomly sampled. The points of $\mathcal{B}$ are thereby constrained to approximately lay on a plane.

Matches $\mathcal{M}(\mathbf{p_1}, \mathbf{p_2}, \mathbf{q_1}, \mathbf{q_2}) \in \mathcal{T}$ are then detected and evaluated using a support score (i.e. fraction of inliers between $\mathcal{T}$ and the transformed $\mathcal{S}$). The random base selection, matching, and evaluation is repeated w times. The returned solution is the match which score either exceeds a given threshold Γ or the match with the largest support after all trials.

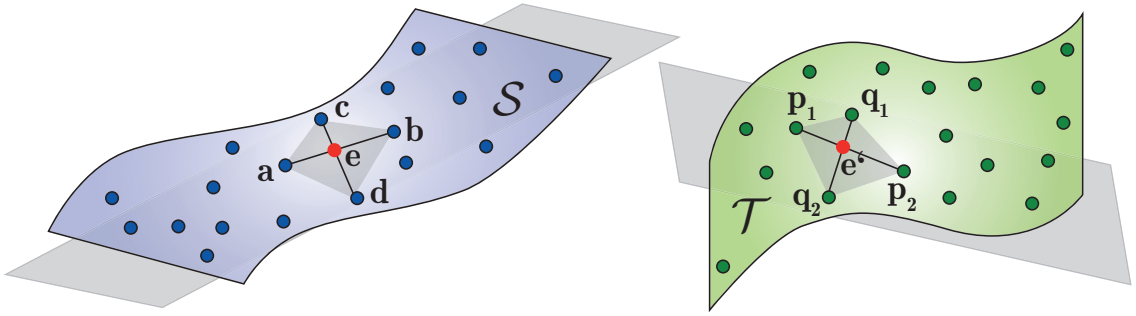


Figure 5.1: Principle of 4PCS with base set $\mathcal{B}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) \in \mathcal{S}$ and a corresponding congruent point set $\mathcal{M}(\mathbf{p_1}, \mathbf{p_2}, \mathbf{q_1}, \mathbf{q_2}) \in \mathcal{T}$ making use of the diagonal intersection point $\mathbf{e}$.

The computational advantage of matching quadruples rather than triplets comes from the fact that intersection ratios (r_1, r_2, Eq. (5.4)) of quadrangle diagonals are invariant under affine and therefore also under rigid-body transformations (Huttenlocher, 1991).

$$r_1 = \frac{\|\mathbf{a} - \mathbf{e}\|}{\|\mathbf{a} - \mathbf{b}\|} \quad , \quad r_2 = \frac{\|\mathbf{c} - \mathbf{e}\|}{\|\mathbf{c} - \mathbf{d}\|} \tag{5.4}$$

Given diagonal ratios (r_1, r_2) from $\mathcal{B}$, matches can efficiently be found by calculating intersection points,

$$\begin{aligned} \mathbf{e}'_{\mathbf{p}}(r_1) &= \mathbf{p_1} + r_1 \cdot (\mathbf{p_2} - \mathbf{p_1}) \\ \mathbf{e}'_{\mathbf{p}}(r_2) &= \mathbf{p_1} + r_2 \cdot (\mathbf{p_2} - \mathbf{p_1}) \quad . \end{aligned} \tag{5.5}$$

$\mathbf{e}'_{\mathbf{q}}(r_1)$ and $\mathbf{e}'_{\mathbf{q}}(r_2)$ are calculated similarly using $\mathbf{q_1}$ and $\mathbf{q_2}$, then quadrangles are checked for affine congruency,

$$\begin{aligned} &\|\mathbf{e}'_{\mathbf{p}}(r_1) - \mathbf{e}'_{\mathbf{q}}(r_2)\| < \delta \ \text{ or} \\ &\|\mathbf{e}'_{\mathbf{p}}(r_2) - \mathbf{e}'_{\mathbf{q}}(r_1)\| < \delta \quad . \end{aligned} \tag{5.6}$$

Because the absolute scale of TLS point clouds is known computational cost can be further reduced, by restricting possible matches to those where the diagonals in $\mathcal{M} \in \mathcal{T}$ and in $\mathcal{B} \in \mathcal{S}$ have similar lengths.

5.5 Extensions and Modifications

Although K-4PCS is generally well-suited to register big, unevenly distributed TLS point clouds, runtime becomes impractical in case of small overlaps, and failures still occur in presence of weak or repeated structures. Here, we introduce several possible ways to accelerate computation and to make K-4PCS more robust.

5.5.1 Conceptual Improvements

In standard 4PCS matching, the best match is evaluated based on a score value, namely the fraction of points in source cloud for which a match is found in the target cloud. Experiments reveal that this evaluation criterion does not always produce a correct registration (i.e. one from which ICP converges to the desired solution). One obvious reason is the inlier threshold δ that must be adapted to the estimated input point cloud density. Evaluation based on the sparse cloud of keypoints thus leads to a rather large δ which in turn yields comparatively high support also for wrong (e.g. symmetrical) solutions. Therefore, we use a modified matching cost.

MSAC. In K-4PCS the cost ρ_R of a putative solution is calculated like in RANSAC. Given the (squared) Euclidean residuals ε^2 between points i in the transformed source cloud and their closest neighbours in the target cloud,

$$\rho_R = \frac{1}{n}\sum_i \rho(\varepsilon_i^2) \quad with \quad \rho(\varepsilon_i^2) = \begin{cases} 0 & \varepsilon_i^2 \leq \delta^2 \\ 1 & \varepsilon_i^2 > \delta^2 \end{cases} . \tag{5.7}$$

A drawback of this binary decision is the strong dependency of the resulting score on the inlier threshold δ. Since the inlier decision is already based on the squared Euclidean distance ε^2, the cost ρ_i for a given point can, without additional computation, be changed to

$$\rho_A(\varepsilon_i^2) = \begin{cases} \varepsilon_i^2/\delta^2 & \varepsilon_i^2 < \delta^2 \\ 1 & \varepsilon_i^2 \geq \delta^2 \end{cases} . \tag{5.8}$$

This modified cost function $\rho_A(\varepsilon^2)$ is called *M-Estimator Sample Consensus* (MSAC; Torr and Zisserman, 2000). It decreases the influence of the threshold δ by considering the residuals of inliers, while outliers still receive a fixed penalty. Note, although further improvement could possibly be achieved by introducing an estimator of the threshold δ (MLESAC; Torr and Zisserman, 2000), we did not pursue this avenue because it increases the processing time.

Translation Costs. The second term of our proposed cost function tries to counter the effect that K-4PCS matching is biased towards solutions with small translation t, which is a direct result of the higher keypoint density in the near-field of the scanner station. The total cost is thus $\rho_A + \lambda\rho_B$. We introduce a sigmoid function, limited by a task-defined minimum and a maximum translation (t_{low},t_{up}).

$$\rho_B(t) = \begin{cases} 1 & t < t_{low} \\ \frac{1}{2} + \frac{1}{2}\cos\left(-\frac{t - t_{low}}{t_{up} - t_{low}}\pi\right) & t \in [t_{low}, t_{up}] \\ 0 & t > t_{up} \end{cases} \tag{5.9}$$

This cost function favours solutions with larger translation, whereas overly close scanner positions become less likely. Such an assumption is reasonable for most TLS applications, where stations are placed apart to cover complementary scene parts. Note that the cost function does not impose any hard constraints, but can be viewed as soft prior that favours scanner locations that are far enough apart. Special cases where two scans are acquired from almost the same viewpoint (e.g. one above the other due to obstacles) can still be successfully registered, but need substantially higher support from the data.

Note that a similar prior could also be constructed for the rotation between scans, e.g., favouring parallel z-axis if instrument is levelled. We have so far not experienced failures with grossly wrong rotation, most likely because in our applications there always was a large number of points on the ground.

5.5.2 Speeding up Computation

A main shortcoming of K-4PCS are its long computation times in presence of low overlap between adjacent scans (with $< 40\%$ overlap up to tens of minutes per pair). We resort to parallelization and also test nested clustering to reduce runtime.

Multi-Threading. The 4PCS method is by construction parallel: each of the w trials (i.e., random base selection, match detection, match evaluation) is independent of all others. A straight-forward speed-up is therefore possible by running the trials, which are the main bottleneck, in parallel.

Our implementation uses the *OpenMP* application programming interface (OpenMP ARB) to distribute separate trials of K-4PCS to different threads. The best solution per thread is recorded and after all w trials, the overall winner is determined (in a single thread). As the

computationally dominating part of the matching runs in parallel, we can expect a speed-up by a factor close to the number of cores available for multi-threading. In practice this factor is reduced by the non-parallelizable parts before and after sample evaluation and by the multi-threading overhead.

Clustering. In K-4PCS, like generally in random sampling methods, the evaluation of candidate matches is a main bottleneck, and it can be expected that in fact evaluation is carried out multiple times for similar possible solutions (i.e. with approximately the same transformation parameters, but resulting from different candidate matches). It has been observed that if the correct solution is the one with the largest support in the data, then – at least in principle – it should be found more often than any other solution during the w trials (Torii et al., 2011). That is, the correct solution should be detectable as a local maximum in a histogram over the w sets of transformation parameters.

Based on this intuition, we try to speed up the search for the correct solution. In practice, more then one solution occurs multiple times, so one has to detect a small, fixed number of maxima that correspond to clusters of possible alignments. Representatives for each of these candidate solutions are scored as described previously, whereas all other solutions are discarded.

Building a histogram in a 6-dimensional solution space is rather impractical. The usual strategy is thus to sacrifice some discriminative power and reduce the voting space. We represent the transformation corresponding to a sample (dubbed a "possible solution") by the length of its translation vector and the angle of its rotation in axis-angle representation. As already explained in Section 5.5.1, K-4PCS is biased towards solutions with very small translation. To compensate this effect we weight histogram entries with a sigmoid function between $[0.5 \ldots 1]$, truncated by t_{low} and t_{up}. A weighted 2D occurrence histogram of translation vector and rotation angle is thus generated, smoothed with a Gaussian filter (1σ), and ten local maxima are detected (see Figure 5.2).

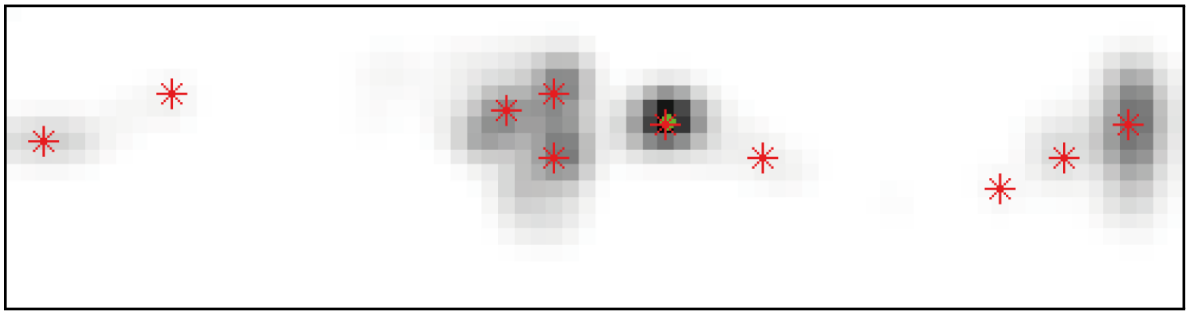


Figure 5.2: Example of a smoothed 2D histogram of all possible solutions with the ten detected local maxima (red) and the true solution from the ground truth (green). Abscissa: 3D rotation angle; ordinate: 3D translation.

Because the two-parameter representation of a 3D alignment is not unique, possible solutions in each local maximum are further accumulated into a 1D histogram based on the angle between their translation vector and a reference direction (chosen to be the translation vector of the first solution). Again, the histogram is smoothed (1σ) and five local maxima are detected. Finally, match evaluation as described in Section 5.5.1 is done on the average of the transformation candidates at each detected maximum.

At most 50 (5×10) possible solutions have to be evaluated, much fewer than the initial amount of possible solutions (in our tests between 2000 and 100000, see Section 5.6). On the downside, positions of local histogram maxima are somewhat inaccurate compared to the optimal solution and potentially too coarse to receive a high fitness score. To counter this effect a few iterations of standard ICP with decreasing inlier threshold are run on the cloud of keypoints (with the possible solution as initial guess).

5.6 Evaluation

We experimentally evaluate the previously described modifications of K-4PCS registration with respect to geometric registration accuracy, error rate, and computational efficiency. Geometric registration accuracy is measured with the *Root Mean Square Error* ($RMSE$) between true correspondences of the transformed source cloud $\mathcal{S}'$ and the target cloud $\mathcal{T}$ after coarse alignment. True correspondences are derived from ground truth that was generated through manual alignment followed by fine registration with standard ICP. The accuracy $\sigma^\star$ of the ground truth is ≈ 5 mm.

Recall that our matching method is based on random sampling of keypoint quadruples and thus results may vary. We thus repeat each registration $m = 50$ times (with constant parameters) to compute the error rate $PER = \frac{1}{m} \cdot \sum_m PE$. The variation of the error rate is determined via cross validation. Since the goal of K-4PCS is to provide a coarse alignment that serves as input to fine-registration (i.e. ICP), a test is considered successful if the solution falls into the ICP convergence basin. Therefore we execute standard ICP after coarse alignment. The criteria for success is henceforth the $RMSE$ after refinement ($RMSE_{\text{ICP}}$):

$$PE = \begin{cases} 0 & RMSE_{\text{ICP}} \leq 3 \cdot \sigma^\star \\ 1 & RMSE_{\text{ICP}} > 3 \cdot \sigma^\star \end{cases} \tag{5.10}$$

Computational efficiency is represented by the total runtime of K-4PCS, i.e., the time to extract keypoints T_{KP} and the matching time T_M. Because T_{KP} (< 15 s for all tests) can be

considered as part of the pre-processing and has not been optimized (e.g. parallelized), only T_M is evaluated in detail. All tests were carried out on a 64 Bit desktop computer with eight cores (3.4 GHz) and 16 GB RAM.

We have run tests on four datasets (including the two used in Theiler et al., 2013). The different datasets address different challenges of registering TLS point clouds. On the one hand, we use an indoor dataset (dubbed "Office") with rather large scan overlaps of $\approx 80\%$ and simple geometry, which, however, gives rise to rotationally symmetric solutions. On the other hand, we test on three outdoor datasets (dubbed "House", "Arch", "Wood"). While datasets "House" and "Arch" represent standard TLS projects with the scanner positioned around an object of interest, dataset "Wood" is an extreme case, where scans are taken in the middle of a forest.

The evaluation is based on tests with fixed basic K-4PCS parameters. Suitable values for the cell size of the voxel grid (τ), the keypoint type (*KP-type*), minimum keypoint response R_{min} and the estimated overlap (*Overlap*) were set empirically according to the scanner setup and the environment (see Table 5.1).

	τ	*KP-type*	R_{min}	*Overlap*
"Office"	5 cm	DoG	0.01	80%
"House"	10 cm	Harris	0.01	50%
"Arch"	10 cm	Harris	0.01	40%
"Wood"	10 cm	Harris	0.001	50%

Table 5.1: Fixed K-4PCS parameters for the different test datasets.

The discussed modifications of the K-4PCS framework are evaluated incrementally:

(a) the original K-4PCS framework,
(b) with multi-threading (see Section 5.5.2),
(c) like (b), with the score function based on MSAC (see Section 5.5.1),
(d) like (c), with additional translation cost (see Section 5.5.1),
(e) using nested clustering followed by (d) (see Section 5.5.2).

5.6.1 "Office" Dataset

The indoor dataset "Office" was acquired in a standard office room of size $10 \times 15\,\mathrm{m}^2$ and consists of five scans of medium resolution ($\approx 10^7$ points). Apart from standard office furniture (tables, chairs, shelves), the room features several cylindrical pillars (Figure 5.3).

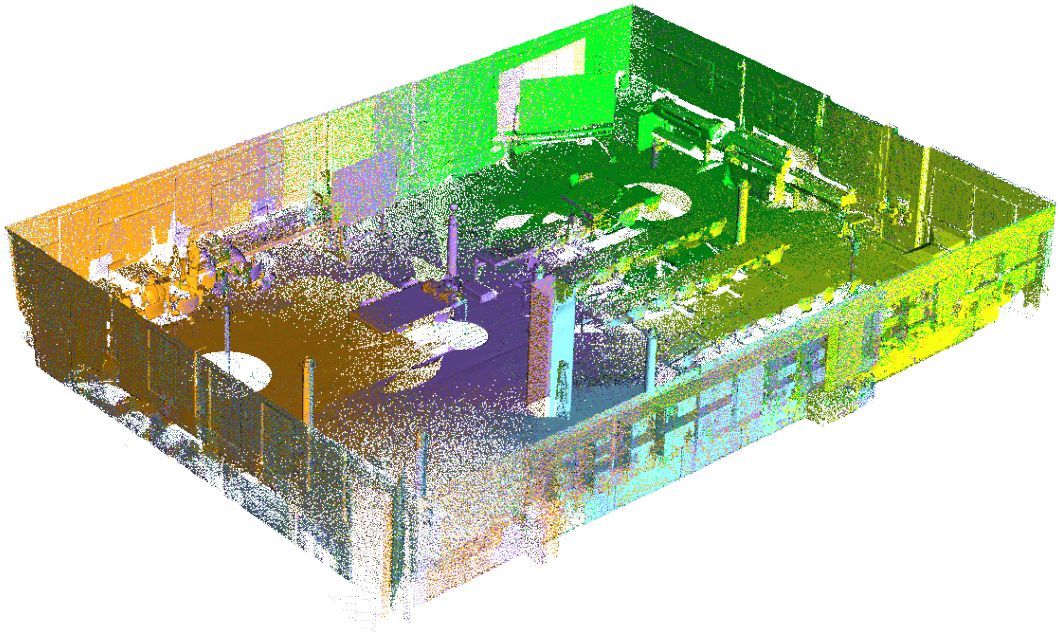


Figure 5.3: "Office" dataset with five scans acquired in a standard office. For better visibility, the ceiling has been removed and only 33% of the points are displayed.

	PER in %	T_M in s	$RMSE$ in cm
(a)	2.8 ± 0.8	89 ± 19	14.2 ± 5.3
(b)	3.4 ± 0.9	18 ± 5	14.6 ± 5.9
(c)	2.6 ± 0.9	18 ± 5	12.7 ± 5.8
(d)	0.8 ± 0.4	18 ± 5	13.2 ± 6.3
(e)	7.6 ± 1.3	4 ± 0	7.8 ± 11.5

Table 5.2: Failure rate PER, matching time T_M, and geometrical accuracy of coarse alignment ($RMSE$) of indoor "Office" dataset.

Table 5.2 shows results of the five test setups. The success rate of the original K-4PCS ($\approx 97\%$) is already quite high. Nevertheless, the proposed modifications (test (c) and (d)) improve the robustness of the matching method. While the replacement of standard RANSAC with MSAC decreases the failure rate only by $<1\%$, the extended score function including penalties for low translation (here $R_{min} = 1\,\text{m}$, $R_{max} = 4\,\text{m}$) is able to prune almost all wrong solutions. Note that 0.8% failures means four trials out of 500. For "Office" the introduction of translation costs significantly improves results because incorrect solutions of standard K-4PCS are caused by the rotation symmetry of the dataset (especially in case of scans in opposite room corners). More precisely, wrong solutions stem from $180°$ rotated solutions that place one scan almost directly on top of the other, which is discouraged by the translation cost function.

Regarding computational efficiency, test (b) shows that distributing the main part of the matching method onto eight cores reduces runtime by a factor of ≈ 5. As expected, the modifications in tests (c) and (d) do not increase runtime. Evaluation based on nested clustering further reduces runtime by a factor of ≈ 5 leading to a total speed-up of ≈ 23. On the negative side, a significant drop of the success rate can be observed in test (e). This indicates that the assumption is not always true that the correct solution is drawn more often than any single wrong solution.

Because of the large speed-up resulting from clustering, another test is done with twice the number of trials w. Naturally, this increases runtime but still gives a two-fold speed-up over test (b). In that test the success rate reaches almost 100% again. However, as we will see in further experiments, this modification is not generally applicable and was therefore not pursued further.

The alignment accuracy does not change significantly. Note the rather large standard deviation of the $RMSE$ in test (e), which indicates that some solutions are even more accurate than 8 cm while others are significantly worse. The reason for the accuracy improvement in (e) is the additional ICP refinement step applied directly to the clouds of keypoints.

5.6.2 "House" Dataset

The "House" dataset consists of six consecutive outdoor scans with $> 2 \cdot 10^7$ points yield four scan pairs with reasonable overlap ($\approx 50\%$). The surroundings of the house are dominated by grassland, vegetation, small paths, and a street (Figure 5.4). Thus, scan overlaps mainly comprise flat ground and the house itself, which again gives rise to rotationally symmetric solutions. In addition, the wooden façade construction includes repetitive structures that make correct coarse registration challenging. Failure rates, runtimes, and accuracies for "House" are summarized in Table 5.3.

	PER in %	T_M in s	$RMSE$ in cm
(a)	16.5 ± 2.5	947 ± 58	72.1 ± 26.1
(b)	15.5 ± 2.9	170 ± 10	80.9 ± 31.1
(c)	15.0 ± 2.8	169 ± 10	78.5 ± 37.5
(d)	12.5 ± 2.6	168 ± 9	79.9 ± 35.2
(e)	60.0 ± 3.8	141 ± 5	95.5 ± 52.1

Table 5.3: Test results of "House" dataset: failure rate PER, matching time T_M, and accuracy ($RMSE$) of coarse alignment.

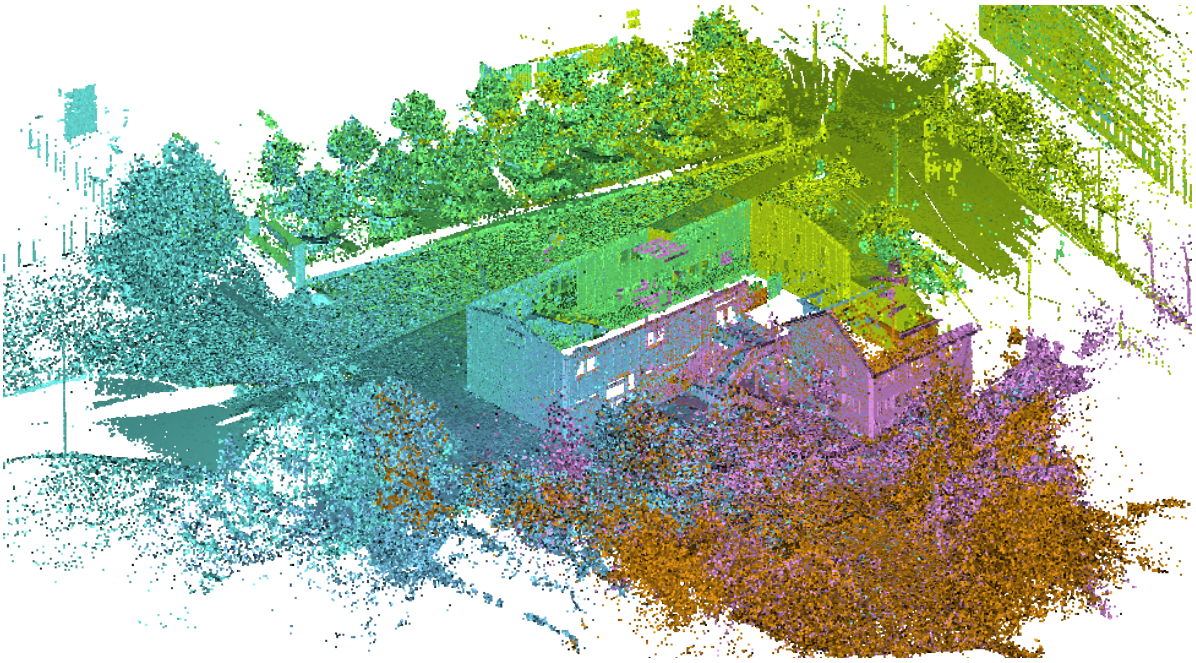


Figure 5.4: "House" dataset with six scans acquired around a house. Only 25% of the points are displayed.

Success rates are generally lower compared to the "Office" dataset, but still $\approx 88\%$ of the runs are registered correctly using the full score function. The positive effect of the proposed modifications can still be observed, but their relative impact is lower (failure rate is reduced by a factor of ≈ 1.2).

The effect of MSAC is insignificant, while the translation cost has some impact. The translation cost function is adapted to the larger distances between scanner positions with $t_{low} = 5\,\text{m}$ and $t_{up} = 10\,\text{m}$. Test (b) shows that multi-threading again reduces the computational cost of matching by a factor > 5 and remains constant over tests (c) and (d). The $RMSE$ of the true correspondences after coarse alignment remains the same, and sufficient for ICP to converge to a correct solution.

In contrast, nested clustering does not work for this dataset, resulting in PER of $\approx 60\%$. The reason is that the histogram often does not contain a clear peak at the position of the correct solution. Thus the correct solution appears not often enough to be detected as one of the most dominant local maxima (Figure 5.5). Moreover, clustering is only $\approx 20\%$ faster than the multi-threaded solution, which is due to the following: first, the number of possible solutions to evaluate for the "House" dataset (≈ 10000) is lower than for "Office" (≈ 50000); second, the larger amount of keypoints (≈ 5200) leads to an increased effort to generate potential solutions, which decreases the influence of the faster evaluation step. In conclusion, clustering essentially fails and does not make sense here.

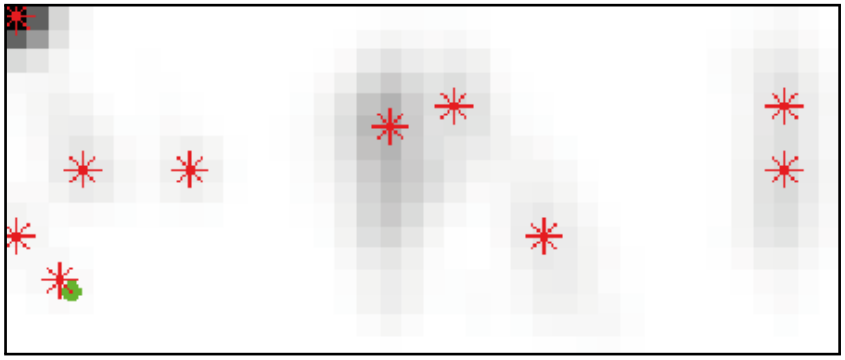


Figure 5.5: An example of the occurrence histogram of tests with the “House” dataset. The correct solution (green) does not appear as a clear peak among the ten maxima (red) causing the method to fail in $\approx 60\%$ of all tests based on nested clustering. Abscissa: 3D rotation angle; ordinate: 3D translation.

5.6.3 “Arch” Dataset

The “Arch” dataset (Figure 5.6) consists of four scans with high resolution ($> 2 \cdot 10^7$ points per scan). Scans cover a Roman arch in Rome and its surrounding paths, buildings, and vegetation. In addition to the low overlap between adjacent scans ($\approx 40\%$), vegetation and artefacts caused by moving people make registration of this dataset challenging. The setup results in only four sufficiently overlapping, matchable scan pairs.

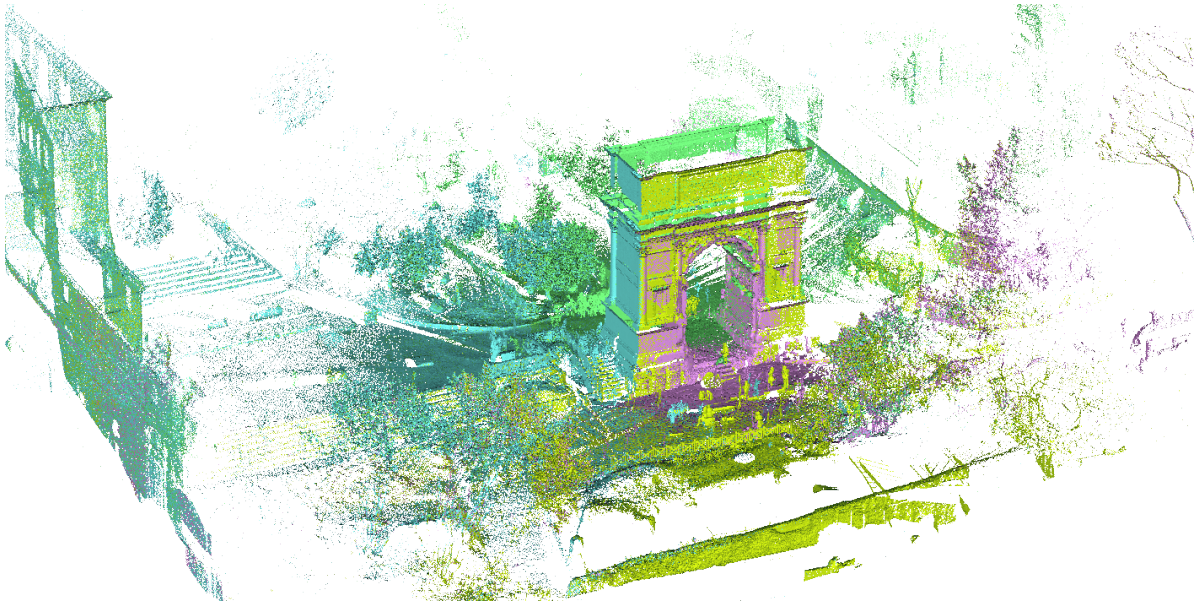


Figure 5.6: “Arch” dataset with four scans positioned around a Roman arch. Notice the artefacts caused by moving people in front of the arch. For better visibility, only 25% of the points are displayed.

Test results in Table 5.4 reflect the findings described in Section 5.6.2. The computational efficiency using multi-threading is significantly increased by a factor of >5 for tests (b), (c) and (d). The error rate with the original K-4PCS framework is already quite low ($\approx 5\%$). MSAC did not significantly improve this, while the translation cost ($t_{low} = 5\,\text{m}$ and $t_{up} = 10\,\text{m}$) leads to noticeably higher success rates ($+2\%$).

	PER in %	T_M in s	$RMSE$ in cm
(a)	4.5 ± 1.6	682 ± 21	94.6 ± 42.3
(b)	5.5 ± 1.8	105 ± 2	96.8 ± 43.0
(c)	5.0 ± 1.7	105 ± 2	93.7 ± 41.3
(d)	3.0 ± 1.4	105 ± 2	87.1 ± 43.4
(e)	35.5 ± 3.8	106 ± 2	102.0 ± 44.6

Table 5.4: Failure rate PER, matching time T_M, and accuracy of the coarse alignment ($RMSE$) of for the "Arch" dataset.

Test (e) shows a marked increase of the error rate by $\approx 30\%$, while no speed-up was achieved. Reasons for this failure are similar to the ones described in Section 5.6.2. Overall, clustering as a replacement for full scoring of all samples appears not to be suitable for scan registration, at least for realistic outdoor settings.

5.6.4 "Wood" Dataset

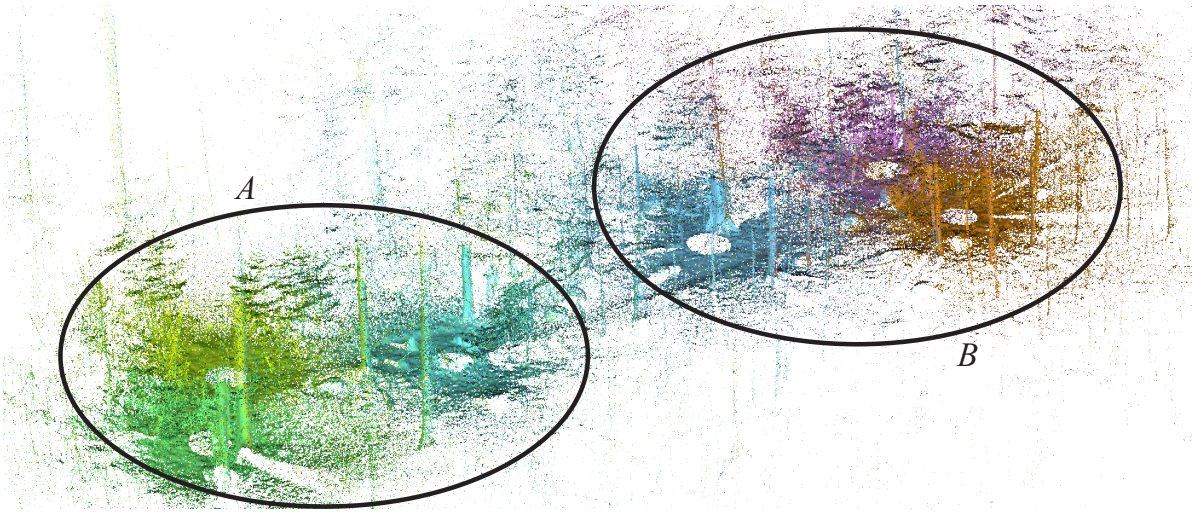


Figure 5.7: "Wood" dataset with six scans divided into two triples *A* and *B*. Only 1% of the data is displayed.

The "Wood" dataset is used to test the proposed framework under extreme conditions. Six high resolution scans – divided into two groups *A*, *B* that only marginally overlap – were acquired in a forested area dominated by bushes and trees (Figure 5.7). The three scans per group all have overlaps of $\approx 50\%$.

In Table 5.5 the test results (error rate, geometrical accuracy, and matching time) are shown for both scan groups. We treat the two groups separately because *(i)* the characteristics of the forested areas are different (*A* is in a less dense part of the forest) and *(ii)* the scans of *B* are placed further apart from each other. Thus, we set different thresholds t_{low}, t_{up} for tests (d) (A: $t_{low} = 2\,\text{m}$, $t_{up} = 5\,\text{m}$; B: $t_{low} = 2\,\text{m}$, $t_{up} = 8\,\text{m}$).

		PER in %	T_M in s	*RMSE* in cm
A	(a)	1.3 ± 1.1	239 ± 6	70.7 ± 34.8
	(b)	1.3 ± 1.1	57 ± 2	73.8 ± 36.7
	(c)	0.0 ± 0.0	57 ± 2	65.0 ± 33.1
	(d)	0.7 ± 0.7	57 ± 2	67.0 ± 35.1
	(e)	85.3 ± 3.2	48 ± 1	130.9 ± 82.9
B	(a)	66.7 ± 4.3	235 ± 12	53.4 ± 28.7
	(b)	68.0 ± 4.3	57 ± 3	57.4 ± 31.2
	(c)	66.0 ± 4.3	57 ± 2	32.9 ± 14.7
	(d)	53.3 ± 4.6	58 ± 2	42.8 ± 19.0
	(e)	99.3 ± 0.7	—	—

Table 5.5: Failure rate PER, matching time T_M, and accuracy of the coarse alignment ($RMSE$) with the "Wood" dataset.

Considering the tests with *A*, the small PER of $\approx 2\%$ in test (a) and (b) proves, that K-4PCS already works very well even in this unstructured environment. Integrating MSAC and translation costs additionally reduces the error rate (0.7% equals one failure case). Multi-threading again strongly reduces runtime (factor ≈ 4) bringing it down to just below 1 minute. Tests with scan triplet *B* show generally lower success rates ($33\% - 47\%$), seemingly caused by the already very large distances between scanner stations. This is confirmed by a serious reduction of PER if translation costs are considered (-14%), although many failure cases remain.

A major problem in the presence of dense vegetation is that the theoretical overlap between scan pairs is not reflected in a similar amount of corresponding keypoints, because their repeatability suffers – in scans from different viewing directions the detector fires rather randomly on different bits of vegetation.

Nested clustering completely fails for the "Wood" dataset. The problem lies in the unordered distribution of keypoints extracted from dense vegetation, leading to a very diffuse distribution of the solutions. The histogram becomes smeared and no clear peaks are detectable. Note that for *B* only one correct solution was found, so no time and accuracy were calculated.

5.7 Conclusions and Outlook

We have evaluated modifications to the K-4PCS matching method for fully automated, markerless point cloud registration. The extensions aim to improve computational efficiency and robustness in the presence of symmetry and repeated structures.

Including MSAC (Torr and Zisserman, 2000) and a translation cost term in the score function proved beneficial, and improved the success rates. The gains are mostly achieved by avoiding relatively rare, but disturbing failures in moderately difficult scenarios, whereas more fundamental methodological upgrades are still required to address difficult cases with frequent failures. Straight-forward multi-threading speeds up matching by a factor of ≈ 5 (or more, with last-generation machines with $12 - 24$ cores). For all evaluated datasets runtime is < 3 minutes even with only small overlap, such that the algorithm becomes practical for productive applications. Compared to manual (coarse) registration, a reasonably fast automatic method is obviously easier to scale up to large projects with many scans (by simply using more computers). Beyond this advantage, we believe that also for a single scan pair further tuning of the implementation could potentially make K-4PCS faster than manual registration, even by an experienced operator.

With the proposed modifications the geometrical accuracy of the coarse alignment remains unchanged, and is generally accurate enough for subsequent ICP refinement. Overall, we have demonstrated that K-4PCS allows for fully automated scan registration in scan projects with not too difficult recording setups and, if appropriately implemented, is fast enough for practical use. Switching from standard sample evaluation to clustering of consistent solutions turned out not to be a viable alternative. Under special conditions it produced a significant speed-up, at the price of a somewhat higher failure rate. In general, the assumption that the correct solution appears more often does not seem to hold as well as the more standard assumption that it has more inliers.

In spite of the proposed improvements the large majority of the remaining failures can still be attributed to symmetries or repeated structures. Although in some cases MSAC and the translation cost term alleviate the problem, they are not able to fully address it, and new solutions are needed. In future work we hope to score individual keypoints by their degree of

uniqueness and saliency (e.g. Shtrom et al., 2013), such that one can prefer those points in a scene which are least likely to cause confusions. Another topic for future work is how to exploit the redundancy in larger scan projects. Typically, the pairwise registration will be successful for many more scan pairs than needed, and it seems natural to utilize the resulting constraints to detect and remedy incorrect scan pairs, either iteratively or in one global process.

Chapter 6

Globally Consistent Registration of Multiple Point Clouds

Globally Consistent Registration of Terrestrial Laser Scans via Graph Optimization

Pascal Willy Theiler, Jan Dirk Wegner, Konrad Schindler

ISPRS Journal of Photogrammetry and Remote Sensing, Volume 109, 2015

(Author version; for publishers' typeset version please refer to the original journal paper)

6.1 Abstract

In this paper we present a framework for the automatic registration of multiple terrestrial laser scans. The proposed method can handle arbitrary point clouds with reasonable pairwise overlap, without knowledge about their initial orientation and without the need for artificial markers or other specific objects. The framework is divided into a coarse and a fine registration part, which each start with pairwise registration and then enforce consistent global alignment across all scans. While we put forward a complete, functional registration system, the novel contribution of the paper lies in the coarse global alignment step. Merging multiple scans into a consistent network creates loops along which the relative transformations must add up. We pose the task of finding a global alignment as picking the best candidates from a set of putative pairwise registrations, such that they satisfy the loop constraints. This yields a discrete optimization problem that can be solved efficiently with modern combinatorial methods. Having found a coarse global alignment in this way, the framework proceeds by pairwise refinement with standard ICP, followed by global refinement to evenly spread the residual errors.

The framework was tested on six challenging, real-world datasets. The discrete global alignment step effectively detects, removes and corrects failures of the pairwise registration procedure, finally producing a globally consistent coarse scan network which can be used as initial guess for the highly non-convex refinement. Our overall system reaches success rates close to 100% at acceptable runtimes < 1 hour, even in challenging conditions such as scanning in the forest.

6.2 Introduction

Static *Terrestrial Laser Scanners* (TLS) are widely used to acquire dense 3D point clouds for various applications in geo-sciences, robotics, entertainment, archaeology, and many more. As laser scanning is a line-of-sight technology, multiple scans from different viewpoints are usually necessary to cover the geometry of a 3D scene. A prerequisite for further processing is thus to align all individual scans in a common coordinate system, to obtain one large point cloud of the complete scene.

In practice, scan registration is at present mostly based on artificial markers (e.g. retro-reflective cylinders, spheres, checker-board targets) that are placed in the scene during scan acquisition. These markers are extracted from different scans either manually or automatically (e.g. Akca, 2003; Franaszek et al., 2009), and used to determine the desired six DoF rigid body transformations.[1] Marker-based scan registration is very reliable, but has several disadvantages. Positioning the markers is time consuming and requires careful planning, to ensure visibility of the markers in different scans, and to avoid degenerate constellations. Additionally, one must ensure that all markers remain stable for the length of the measurement campaign. Finally, markers occlude (small) parts of the scene and often have to be removed from the data for further analysis or visualization.

In order to circumvent markers altogether, this paper presents a method for marker-less TLS point cloud registration based on natural 3D keypoints. The method can handle multiple scans acquired from arbitrary positions, without any prior knowledge about their relative orientations. Compared to the generic problem of aligning point clouds, specific challenges of TLS are *(i)* near-field bias: the polar measurement principle of static TLS causes a quadratic decrease of the point density with increasing distance from the sensor. This results in a very uneven point density (respectively resolution) within a scan. *(ii)* Outliers: real point clouds are contaminated by gross errors, e.g. due to moving objects, reflections or complicated surfaces

[1] TLS directly measures absolute distances, thus only 3D translations and rotations must be estimated.

such as vegetation, etc.[2] *(iii)* Sheer point cloud size: current TLS operate at frequencies up to one million points per second. Scan projects featuring several hundred million points are common and have to be handled in reasonable time.

In most cases TLS registration is divided into an initial coarse alignment and a subsequent fine registration. Coarse alignment uses only a sparse set of corresponding points whereas fine registration, initialized with the coarse solution, typically uses a much larger subset or even all points. Fine registration amounts to minimizing the point-to-point or point-to-surface distances between the point clouds, usually with the *Iterative Closest Point* algorithm (ICP; Besl and McKay, 1992; Chen and Medioni, 1992) or some variant of it (e.g. Bergevin et al., 1996; Bae and Lichti, 2004; Minguez et al., 2006; Bouaziz et al., 2013). These methods locally minimize the sum of per-point residuals, which is conceptually straight-forward, but highly non-convex, and notoriously prone to converge to weak local minima. Arguably the more critical part of the pipeline is coarse registration, which aims to provide a rough initial transformation that is good enough as starting value for ICP, so that the latter reaches a useful minimum. An interesting one-step approach without separate coarse registration has been introduced by Yang et al. (2013). They propose to globally solve the non-convex objective function with a branch-and-bound scheme. The resulting *Globally Optimal ICP* (Go-ICP) works very well with small point clouds, but is intractable for large point clouds like those produced by TLS.

In this paper we describe a complete framework for fully automated, marker-less registration of multiple TLS point clouds acquired from arbitrary positions. The framework follows the common procedure to split the registration into coarse initialization and fine registration, and uses existing methods for the latter – in particular a combination of standard pairwise ICP and the global refinement algorithm of Lu and Milios (1997), from now on called LUM. The main methodological novelty of the proposed framework is an efficient way to find a coarse alignment that is consistent across all overlapping scans in a project. Note that this step, while often overlooked, is crucial for the entire registration pipeline: for a successful refinement with local methods the initial alignment must be roughly correct for all scans. In other words a possibly quite inaccurate, but reasonably consistent network must be established before fine registration. Statistically speaking, coarse registration does not need to have high accuracy, but it must have high reliability.

Consequently, we design a scheme that exploits the available redundancy already at this stage, but without the computational cost associated with high accuracy. Our global alignment method assumes that some hypothesis generator is available, which returns putative pairwise

[2]Note, the fact that TLS is a rather high-end sensor does not mean that it is less prone to outliers. Low-cost alternatives like robotic scanners or even the KINECT will usually have higher random noise and systematic errors, but not necessarily more gross errors.

alignments between two scans. I.e., for any two scans with reasonable overlap, the hypothesis generator can find a set of relative transformations that are plausible in the light of the observed points. We start from our previous work on pairwise alignment with *Keypoint-based 4-Points Congruent Sets* (K-4PCS; Theiler et al., 2014a), but the framework is generic and can use any other hypothesis generator instead. Given a discrete set of putative alignments between scan pairs, we develop a graph-based energy minimization scheme which selects one of the putative transformations for each pair in such a way that the total loop closure error across all scan pairs in the project is as low as possible. The formulation allows for an efficient solution with combinatorial optimization algorithms. The global coarse registration, in conjunction with standard fine registration algorithms, yields correct scan alignment ($< 3\%$ failures) without any manual intervention across a range of applications scenarios (urban, forest, archaeological sites, indoors). An open-source implementation of the complete registration pipeline will be made available in conjunction with the paper.

6.3 Related Work

6.3.1 Pairwise Coarse Registration

Coarse, pairwise point cloud registration typically has two consecutive parts. First, raw point clouds are reduced to sets of sparse features (a.k.a. keypoints) and, second, correspondences are sought. Transformation parameters based on groups of three or more corresponding features are estimated, which establish the relative orientation[3] of both point clouds in a common reference frame. Typically, salient geometric entities are used to establish correspondences (although there are methods which randomly select a small set of features, e.g., Masuda and Yokoya, 1995; Barnea and Filin, 2008; Leng et al., 2014). The most popular features are either 3D keypoints, or planar surfaces, respectively surface normals.

A straight-forward approach is to leverage the power of interest point extraction in conventional images. To that end, 2D interest points are extracted from intensity or range images and lifted to 3D with the known range measurement. E.g., Böhm and Becker (2007) detect SIFT features (Lowe, 2004) in intensity images to derive pixel correspondences and estimate the rigid-body transformation from the associated 3D points. Kang et al. (2009) propose a similar approach, but add an outlier detection step based on 3D distances between putative feature point pairs.

[3]We follow the terminology of photogrammetry: "relative orientation" refers to the full transformation between two instrument coordinate systems, including both the rotation *and* the translation.

3D keypoint detectors are often direct extensions of standard 2D methods like *Difference-of-Gaussians* (DoG) in 3D space (e.g. Allaire et al., 2008; Flitton et al., 2010). Another 3D keypoint extractor is adapted from the *Harris* corner detector (Harris and Stephens, 1988) by replacing image gradients with point normals (e.g. Sipiran and Bustos, 2011). Feature points are often encoded with descriptors which serve to measure similarity when searching for correspondences. For example, Flint et al. (2007) introduce a 3D version of the SURF descriptor (Bay et al., 2006) to compare keypoints in range images. Descriptors specifically for 3D point cloud data typically describe a point's neighbourhood by histograms of the point distribution and/or the variability of the normals. Perhaps the first instance of this idea are *Spin Images* (Johnson and Hebert, 1999), more recent versions include *Point Feature Histograms* (PFH; Rusu et al., 2008) and their accelerated version (FPFH; Rusu et al., 2009). Overviews on the state-of-the-art keypoint detection and matching in 3D space, as well as performance evaluations, can be found in Tombari et al. (2013) and Hänsch et al. (2014).

An alternative strategy, which is however limited to man-made environments, relies on surfaces rather than keypoints, such as planes (Dold and Brenner, 2006; Brenner et al., 2008; Theiler and Schindler, 2012) or other geometrical primitives (e.g. spheres, cones) as in Rabbani et al. (2007). A more robust variant does not detect explicit planes, but uses salient directions of the (point-based) surface normals for alignment (Makadia et al., 2006; Zeisl et al., 2013; Novák and Schindler, 2013).

In our work, we also reduce the points to a relatively small set of (a few thousand) salient 3D keypoints for coarse registration. However, we prefer not to use descriptors, because they lack robustness to strong viewpoint changes. Such viewpoint changes are commonplace in TLS, as operators try to minimize the number of scans in order to cut down field time. Instead of matching descriptors, unambiguous geometric keypoint configurations form the basis of our pairwise registration method (K-4PCS; cf. Section 6.4.1).

6.3.2 Global Fine Registration

Global fine registration of multiple point clouds aims to evenly redistribute the accumulated loop closing errors that result from chaining pairwise registrations. Most approaches build upon precise pairwise orientations to reduce the computational effort for global refinement. The goal is to polish the overall network by minimizing the distance between corresponding points, either incrementally (e.g. Chen and Medioni, 1992; Pulli, 1999) or in one shot (e.g. Bergevin et al., 1996; Neugebauer, 1997; Benjemaa and Schmitt, 1998). Often, this task is formalized with the help of a *scan graph*, in which each scan forms a node, and each known pairwise transformation forms an edge. Lu and Milios (1997) have presented the LUM approach, which

aims at the construction of a globally consistent model from scans acquired by a moving robot. This SLAM-like method, originally designed for 2D (planimetric) transformations, has been adapted to 3D by Borrmann et al. (2008), and successfully applied to different data types, including point clouds captured with the KINECT (Weber et al., 2015).

A complete method somewhat related to ours was proposed in Huber and Hebert (2003). Pairwise alignments, resulting from (mesh-based) pairwise matching of point clouds, are checked for local plausibility by inspecting the resulting overlaps and surface visibilities. Locally implausible pairs are simply pruned from the network. Given the remaining valid alignments, the method incrementally builds a *Minimum Spanning Tree* (MST), while at each iteration global refinement is carried out with the algorithm of Neugebauer (1997). If the result after refinement is globally plausible (i.e., all overlapping pairs in the partial MST are locally plausible), the edge is added to the tree until all scans are included.

The multi-scan registration framework presented here follows a similar philosophy, but considers multiple putative transformations per pair (rather than commit early) and checks global plausibility already in the coarse registration stage (rather than repeatedly call the time consuming fine registration). Experience shows that the alignment error between two TLS scans is often multi-modal (e.g. due to repetitive structure, rotational symmetry, etc.), therefore it is not advisable to commit to a single solution at this stage. We therefore sample multiple solutions with reasonably low fitting error. The message of this paper is that, with the help of consistency constraints between pairs that form loop in the scan network, one can disambiguate the pairwise transformations. Given a globally consistent coarse alignment of the entire network, we follow the standard approach and refine first with pairwise ICP, then with LUM.

6.4 Proposed Framework

Our task is fully automated registration of multiple overlapping TLS point clouds, without further constraints on scan locations and without artificial markers. Like most related works (cf. Section 6.3) we first perform a coarse registration to find approximate scanner orientations, and then polish them in a separate fine registration step (see Figure 6.1) Coarse registration starts with finding approximate pairwise orientations between overlapping scans. We do this with our previously developed K-4PCS method (Theiler et al., 2014a), but other registration schemes could be used as well.

We note in this context that in many TLS projects a part of the orientation is known a-priori from additional sensors, in particular scanners are often levelled, such that the rotations around the x- and y-axes are small. In such cases one can use simpler procedures for pairwise alignment.

However, situations exist were levelling the scanner is either not possible (e.g. when the scanner is positioned through manholes or inspection ducts) or not practical (e.g. on archaeological sites the instrument is sometimes left on the tripod and set up just so that it does not collapse). To remain as generic as possible, we do not assume levelled scanners.

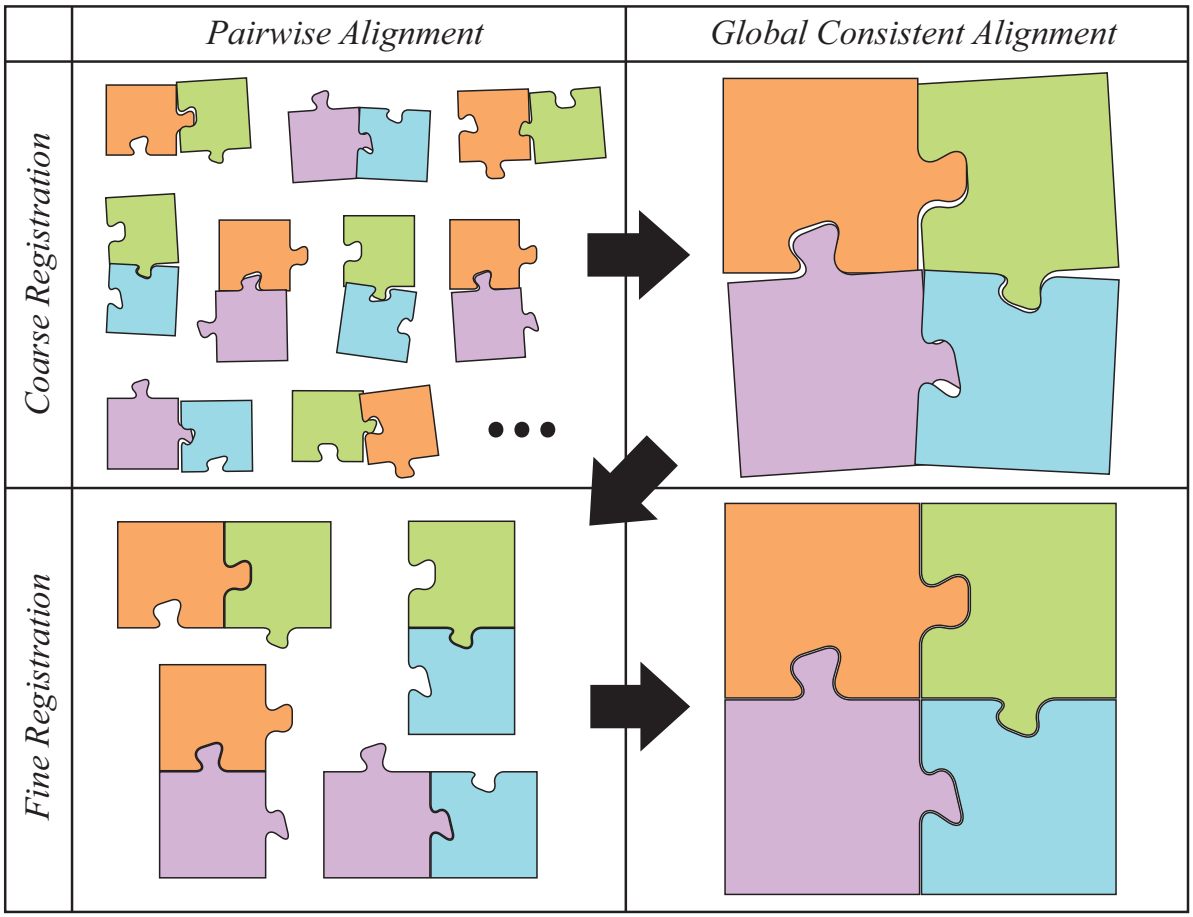


Figure 6.1: Workflow: Coarse, pairwise matching is followed by coarse alignment into a globally consistent network. This network then serves as initialization for pairwise fine registration, which in turn provides starting values for global refinement. Each puzzle piece represents one scan; equal colour depicts the same scan.

The focus of this paper is the subsequent step from pairwise registration to a coarse, but consistent alignment of all scans in a common coordinate frame (in image metrology and robotics also referred to as “multi-view orientation”, “pose estimation” or “loop closure”). For pairwise scan orientation, several useful methods exist, and the problem is rather well-understood. Still, even the best methods sometimes fail. Common failure are caused by ambiguous scene content such as are repetitive structure (e.g. façade patterns) or rotational symmetries (e.g. rooms in indoor scans). Due to the measurement principle of TLS both the point count and the resolution of geometric details can vary dramatically between scans of the

same surface. An important observation in this context is that in all these cases, the registration error of the correct solution will be reasonably low, as desired. The difficulty stems from the fact that the error function is multi-modal, i.e., there are other, wrong solutions which have low registration errors, too.

Consequently, the pairwise registration procedure will in most cases find – among others – the correct solution, if it has the capability to return multiple candidate solutions rather than commit to a single one. Our task then becomes to filter the pairwise candidate solutions in such a way that they form a globally consistent network: rigid body transformations are transitive, so any closed loop in the scan network should (approximately) add up to an identity transformation. The core of this paper is a discrete graphical model (Bishop, 2006) of the scan network. Inference in that model amounts to picking one candidate for each pairwise transformation, such that the loop constraints are enforced simultaneously for all loops up to a given length.

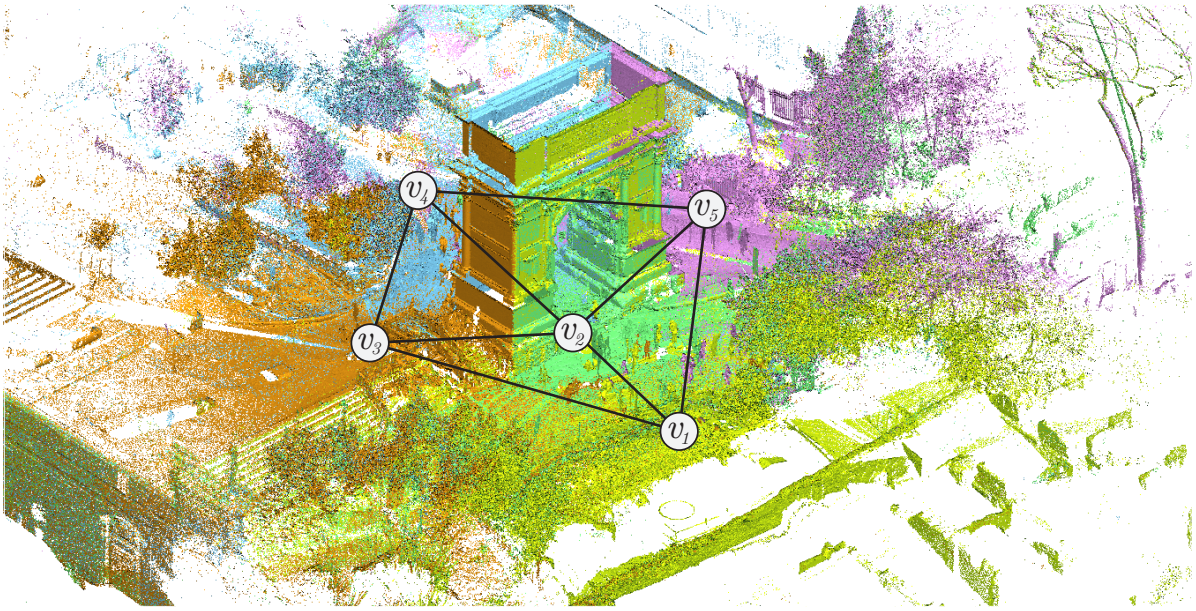
To make the description of the complete system self-contained, we start with a brief description of the K-4PCS method for pairwise, coarse registration, and then move on to the coarse alignment of all scans in a globally consistent network. We also briefly describe the subsequent fine registration which completes the end-to-end registration pipeline.

6.4.1 Pairwise Coarse Registration with K-4PCS

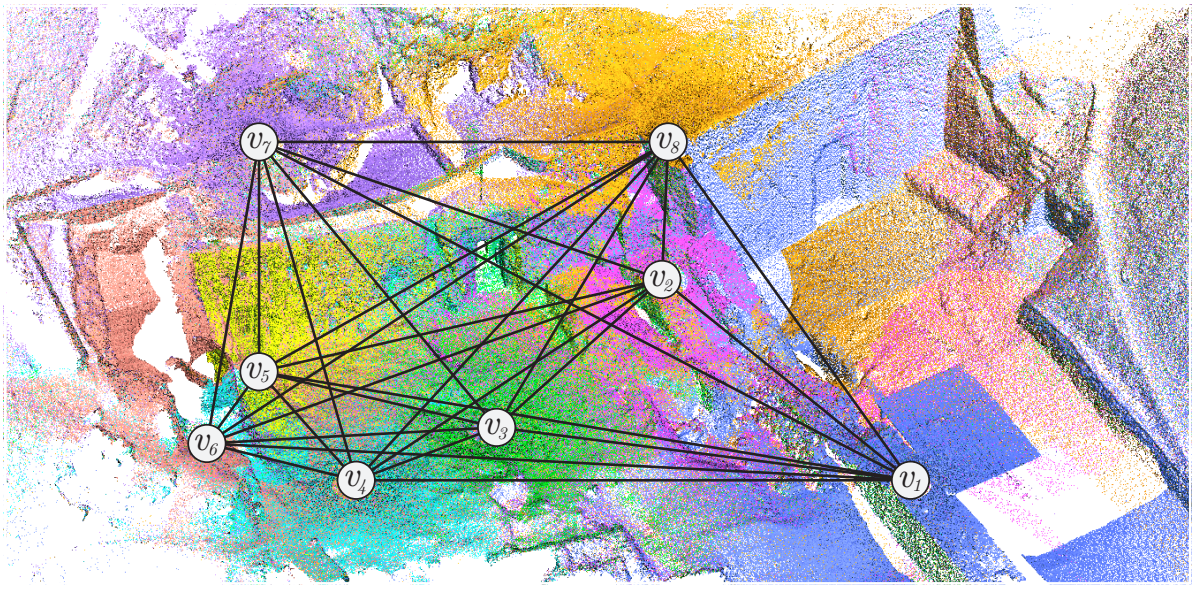
As input for globally consistent registration, we first find pairwise alignments with K-4PCS on scan pairs (i.e. edges U) within a scan graph $\mathcal{G}_S(V, U)$. If it is known in advance which scan pairs are likely to have significant overlap (e.g. from the acquisition plan) then one aligns only those pairs (Figure 6.2a). Otherwise, one conservatively starts from a fully connected scan graph (Figure 6.2b) and leaves it to the global alignment step to eliminate bogus alignments between non-overlapping scans.

We first briefly recap K-4PCS (Theiler et al., 2014a,b), for more details the interested reader is referred to the original publications. Our implementation of the algorithm, as described in this section, has been included as part of the open-source *Point Cloud Library* (PCL; Rusu and Cousins, 2011).

K-4PCS has two steps. First, reduce the raw scans to smaller sets of points that have high repeatability. This is done by voxel grid filtering to even out variations in point density, followed by 3D keypoint detection with standard methods like 3D Harris and/or 3D DoG. Then, candidate transformations between the two keypoint clouds are found with the *4-Points Congruent Sets* (4PCS) algorithm (Aiger et al., 2008). This method is a RANSAC-type robust



(a) Adjacency graph



(b) Fully connected graph

Figure 6.2: Examples of basic scan graphs $\mathcal{G}_S(V, U)$: (a) Manually defined edges with sufficient mutual overlap and (b) fully connected graph without prior information.

fitting method, which repeatedly samples a small base set, finds putative point correspondences, estimates the transformation between the two point clouds from them, and scores the different random samples by how well they are supported by the two (key)point clouds. 4PCS exploits the fact that a base set of four roughly co-planar points is actually more efficient than the minimum of three points, because only computations with subsets of two points are needed to test the base sets for affine, respectively metric congruency (Huttenlocher, 1991).

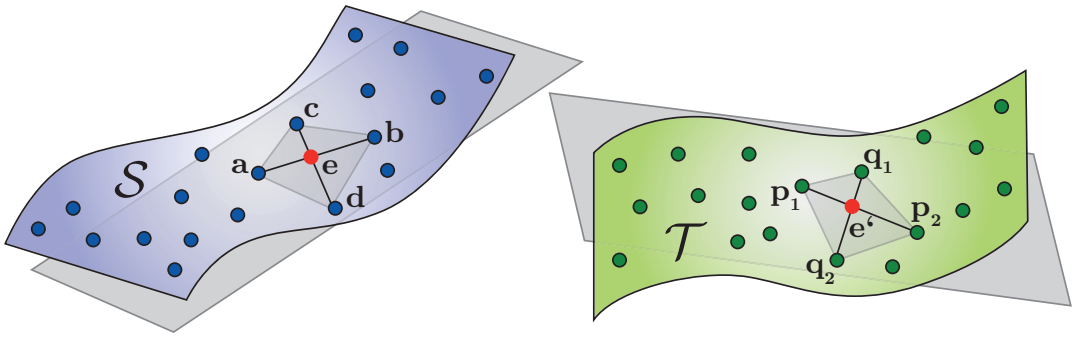


Figure 6.3: 4PCS matching: Given a point quadruple $\mathcal{B}(\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d}) \in \mathcal{S}$ we seek congruent four-point sets $\mathcal{M}(\mathbf{p_1},\mathbf{p_2},\mathbf{q_1},\mathbf{q_2}) \in \mathcal{T}$. Congruency is verified via intersection ratios r_1, r_2 of the diagonals in $\mathcal{B}$.

Given a base set $\mathcal{B}(\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d})$ in the source point cloud $\mathcal{S}$, one searches for a matching base set $\mathcal{M}(\mathbf{p_1},\mathbf{p_2},\mathbf{q_1},\mathbf{q_2})$ in the target point cloud $\mathcal{T}$ as follows (see Figure 6.3):

- Calculate the diagonal intersection point $\mathbf{e} = \text{intersect}(\mathbf{ab},\mathbf{cd})$
- Calculate the diagonal intersection ratios $r_1 = \|\mathbf{a}-\mathbf{e}\|/\|\mathbf{a}-\mathbf{b}\|$ and $r_2 = \|\mathbf{c}-\mathbf{e}\|/\|\mathbf{c}-\mathbf{d}\|$
- Extract point pairs $\mathcal{P}(\mathbf{p}_1,\mathbf{p}_2)$ and $\mathcal{Q}(\mathbf{q}_1,\mathbf{q}_2)$ from $\mathcal{T}$ whose pairwise distances roughly match the diagonals of $\mathcal{B}$, i.e., $\|\mathbf{p_2}-\mathbf{p_1}\| \approx \|\mathbf{b}-\mathbf{a}\|$ and $\|\mathbf{q_2}-\mathbf{q_1}\| \approx \|\mathbf{d}-\mathbf{c}\|$
- For each resulting quadruple $\mathcal{M}(\mathbf{p_1},\mathbf{p_2},\mathbf{q_1},\mathbf{q_2})$, check whether the diagonal intersection points (using r_1, r_2) coincide,

$$\begin{aligned} &\mathbf{p_1} + r_1 \cdot (\mathbf{p_2}-\mathbf{p_1}) \approx \mathbf{q_1} + r_2 \cdot (\mathbf{q_2}-\mathbf{q_1}) \quad \text{or} \\ &\mathbf{p_1} + r_2 \cdot (\mathbf{p_2}-\mathbf{p_1}) \approx \mathbf{q_1} + r_1 \cdot (\mathbf{q_2}-\mathbf{q_1}) \end{aligned} \tag{6.1}$$

- For base sets which pass the test, check also for similar side lengths of the quadrangles $\mathcal{B}$ and $\mathcal{M}$
- Base sets $\mathcal{M}$ and $\mathcal{B}$ that survive both tests are congruent; fit a rigid-body transformation and find its support in the keypoint clouds

To compute the support that a sample receives from the n points of $\mathcal{T}$, K-4PCS uses *M-estimator Sample Consensus* (MSAC; Torr and Zisserman, 2000), with a truncated quadratic cost

$$\rho_A = \frac{1}{n}\sum_{j=1}^{n} \min\left(\frac{\varepsilon_j^2}{\delta^2},\ 1\right), \tag{6.2}$$

with ϵ_j the residual errors and δ a truncation threshold. Additionally, a mild prior on the translation t favours configurations where the two scanners are not too close to each other.

$$\rho_B(t) = \begin{cases} 0 & t > t_{up} \\ \frac{1}{2} + \frac{1}{2}\cos\left(-\frac{t - t_{low}}{t_{up} - t_{low}}\pi\right) & t \in [t_{low}, t_{up}] \\ 1 & t < t_{low} \end{cases} \tag{6.3}$$

The boundaries t_{low} and t_{up} define a smooth transition from the maximal translation cost 1 to 0. Taken together, the final cost function, scaled to the range $[0 \dots 1]$, is

$$\rho = \frac{1}{1+\lambda} \cdot (\rho_A + \lambda \cdot \rho_B) \quad . \tag{6.4}$$

The above matching scheme, repeated for multiple randomly sampled base sets, delivers a list of alignment candidates, ranked by their matching cost.

6.4.2 Global Graph-based Coarse Registration

By the nature of TLS, pairwise registration based on point/normal geometry is sometimes ambiguous (cf. Section 6.3). The key to robust alignment is to exploit the redundancy afforded by loop constraints in the scan graph. We point out that the role of redundancy is quite different for fine and coarse registration. Global fine registration shall minimize the influence of random noise on the transformation parameters, by distributing the residual errors evenly across the project. Global coarse registration shall weed out grossly wrong transformation parameters, whereas its accuracy must only be good enough to initialize fine registration.

By starting from a set of putative pairwise alignments, our two-step pipeline discretizes the problem. Given the candidate alignments, we seek to select one of the candidates for each pair, respectively for each edge of the scan graph (Figure 6.2), such that the overall discrepancies are minimized. That formulation makes it possible to leverage efficient discrete optimization algorithms for inference.[4] It will of course sometimes happen that no correct candidate is found for a pair. This case can be addressed with a virtual solution, i.e. a "joker" transformation, which, independent of the two scans' relative orientation, has a cost (fitting error) slightly higher than the maximum expected cost of a correct registration.

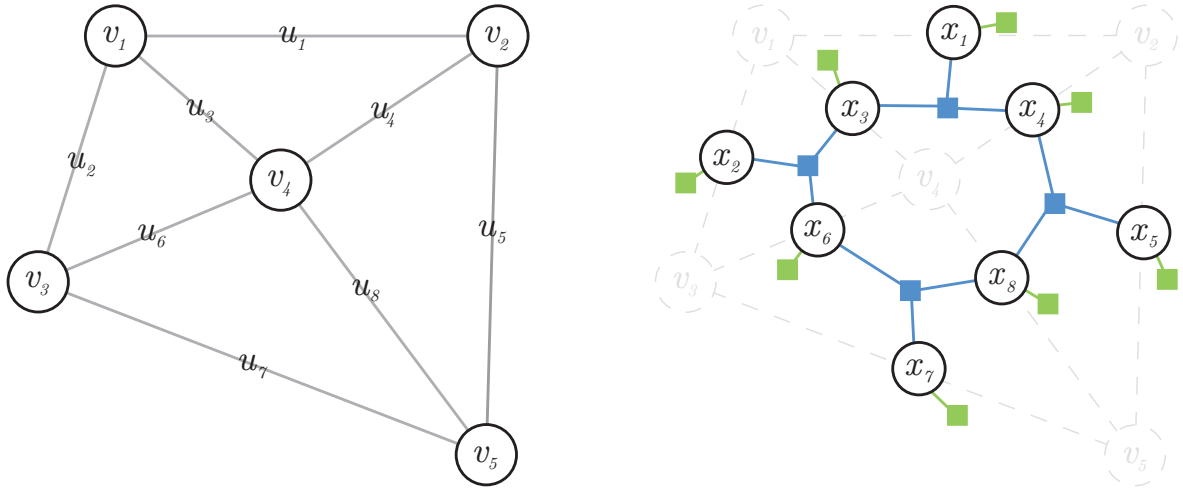
Inevitably, it will sometimes happen that for one or more pairs all alignment hypotheses are wrong. For the discrete (forced choice) optimization we thus introduce an additional *virtual solution* per pair. These "joker" transformations have the same cost for any choice of pairwise

[4]Note, discretization of multi-dimensional problems with the help of data-driven proposals has been successfully used in image analysis tasks, including for example optical flow (Lempitsky et al., 2008) and multi-target tracking (Milan et al., 2013).

orientation parameters, and that cost is set higher than the maximum expected cost of a correct registration. In this way virtual solutions will only be selected if all available candidates are inconsistent with the overall network. Note that a virtual solution does not hold valid transformation parameters. It only serves to identify scan pairs where none of the candidate transformations can be correct. After global coarse registration, such pairs are eliminated from the scan graph and not used in later steps.

To include the loop constraints, we construct a graphical model $\mathcal{G}_R(X, L)$ from the scan graph $\mathcal{G}_S(V, U)$. The variables $x_i \in X$ of that model are scan pairs, and the domain of a variable x_i is its set of l labels (i.e. candidate transformations) $\mathbf{T}_i(\mathbf{t}_i, \mathbf{R}_i)$ indexed through a superscript, $x_i \in \{\mathbf{T}_i^1, \mathbf{T}_i^2, \ldots, \mathbf{T}_i^l\}$. The unary potential of a specific transformation is its pairwise alignment cost $\rho(x_i)$ from Eq. (6.4). Higher-order potentials of order ≥ 3 connect variables that form closed loops in the scan graph.

Loosely speaking, the graphical model is a dual representation of the scan graph, where vertices x_i correspond to edges $u_i \in U$ (see Figure 6.4).



(a) Basic scan graph $\mathcal{G}_S$

(b) Corresponding graphical model $\mathcal{G}_R$

Figure 6.4: Example of scan graph $\mathcal{G}_S(V, U)$ and corresponding graphical model $\mathcal{G}_R(X, L)$. Scan pairs are variables in the graphical model (black circles in (b)), pairwise registration costs ρ are the unary potentials (green) and loops form higher-order cliques (blue) with cost ψ. For readability, only third-order cliques are shown.

The higher-order potentials $\psi_k(x_u, x_v, x_w, \ldots)$ encode the consistency constraints, i.e., they become higher as the loop-closing error grows. The energy of the graphical model thus reads

$$E = \frac{1}{\omega} \cdot \sum_i \rho(x_i) + \sum_k \psi_k(x_u, x_v, x_w, \ldots) \; . \tag{6.5}$$

The weight ω balances the importance of the unary and higher-order terms. The higher-order potentials ψ_k (in graphical model terminology "cliques") measure the discrepancy of the transformations around a loop. In principle one could take into account cliques of arbitrary order, but lower orders (shorter loops) are sufficient to discard grossly wrong transformations, and are computationally more efficient. We thus extract only cliques of order three and four exhaustively, and add cliques of order ≥ 5 only where necessary to ensure all scan pairs are constrained. E.g., if a scan is only registered to its two immediate neighbours in a loop of five scans around a building, this will give rise to a clique of order five. Loops are extracted from the scan graph with conventional *Depth First Search*.

The consistency potential ψ_k penalizes loop closing errors. To that end we chain all transformations around a loop, $\mathbf{T}_\varepsilon(x_u, x_v, x_w, \ldots) = \mathbf{T}_u \cdot \mathbf{T}_v \cdot \mathbf{T}_w \cdot \ldots$, which in the error-free case would yield an identity transformation. As clique potential we use a combination of the deviations from 0 translation and 0 rotation. The translation error is defined as the magnitude $t_\varepsilon = \|\mathbf{t}_\varepsilon\|$ of the residual 3D translation vector, truncated at a maximum t_{max} for robustness. Similarly, the rotation $\mathbf{R}_\varepsilon$ is converted to axis-angle representation and the angle θ_ε is defined as 3D rotation error, and truncated at a maximum value θ_{max}.

To account for error propagation, the truncation values are adapted to the clique size: given an estimate for the maximum deviation t_{pair} of a single transformation, the value for a loop of length h becomes $t_{max} = \sqrt{h} \cdot t_{pair}$, and similar for the angle. Empirically, assigning the translation and rotation equal weights and using the squared error as cost worked best. The cost ψ_k is again scaled to $[0 \ldots 1]$ to arrive at a similar value range as for the unary terms $\rho(x_i)$.

$$\psi_k = \frac{1}{2}\left(\min\left(\frac{t_\varepsilon}{t_{max}}\,,\,1\right) + \min\left(\frac{\theta_\varepsilon}{\theta_{max}}\,,\,1\right)\right) \tag{6.6}$$

The most consistent global alignment corresponds to the choice of indices j, respectively candidate transformations $\mathbf{T}_i^j$ that minimizes the energy E (see Eq. (6.5)). The energy minimization is NP-hard, but in practice relatively well-behaved, due to the limited clique size. We use the Lazy-Flipper algorithm (Andres et al., 2012a), as implemented in the open-source *OpenGM* library (Andres et al., 2012b). Generally, one could apply various discrete optimization methods, such as message passing algorithms (e.g. Loopy Belief Propagation; Kschischang et al., 2001) or move-making algorithms (e.g. α-Expansion; Boykov et al., 2001). The Lazy-Flipper is particularly suitable for our application: it uses a clever data structure to efficiently perform a depth-limited exhaustive search around a given initial solution. As our initial labels (i.e. the best pairwise alignments) can be expected to often be good guesses, the algorithm converges quickly. Moreover, the Lazy-Flipper by construction guarantees that changing any subset of transformations up to the depth limit η will not lead to a better solution.

Since our graphical models are relatively small, a fixed computational budget will correspond to a relatively high η, and thus to a strong minimum.

For each loop in the graph we have to compute transformation residuals $\mathbf{T}_\varepsilon$ for all combinations of labels (possible alignments) of the involved scan pairs. This means that the computational cost grows quickly with the number of labels l (polynomially with order l). Luckily, the ordering defined by the unary cost is relatively reliable, meaning that the best pairwise solution is in most cases among the candidates with lowest costs.

We exploit this and devise an iterative minimization scheme. In the first iteration, we assign each vertex the pairwise transformation with the lowest cost ρ and compute the energy. We then identify the vertices that are not part of any consistent loop, double the size of the label set (i.e., we add the second-best pairwise transformation) and minimize the energy. This last step is repeated, doubling the set of candidate labels in every iteration, until the energy no longer decreases. As already explained above, we always include an extra label for the virtual solution, which serves as a fall-back when the correct pairwise transformation is completely missing. That label has a cost chosen such that it is higher than that of any correct solution, but lower than most grossly wrong ones. It will therefore be selected only if none of the available labels can be correct. Accordingly, loops that include a virtual solution never count as consistent.

The result of the presented global coarse alignment is a complete set of pairwise scan orientations, which is as consistent as possible throughout the global scan network.

6.4.3 Fine Registration

Coarse registration, if successful, delivers the necessary input for precise fine registration, namely relative transformation parameters to place all scans into a common coordinate frame, which are accurate enough to serve as initial values for maximum-likelihood estimation (network adjustment). Technically, fine registration must at the same time establish point-to-point correspondence between all scans and minimize the sum of residuals over all such corresponding points. This is a brittle, highly non-convex problem, hence care has to be taken to ensure convergence to a good minimum of the error function. Moreover, there are typically millions of points, so an additional challenge is to come up with a computationally tractable scheme.

Like several others, we prefer to approach the task in two steps. First, an independent fine registration of scan pairs, with standard robust ICP; followed by global fine registration with LUM. Both ICP and LUM are standard tools available off-the-shelf in open-source as well as commercial software, hence we only give a brief run-down with an emphasis on the

particularities of our framework. In our implementation we exploit the versions available in PCL.

As input to fine registration we use the voxel grid resampled point clouds of the original, dense input scans. Compared to the raw data, the resampled point clouds have two advantages: *(i)* a more even point density, which ensures that point correspondences are representative of surface registration errors; and *(ii)* a lower point count, which greatly speeds up the computation.

To further reduce the computational burden we prune the scan graph $\mathcal{G}_S(V, U)$ to a minimal set of reliable edges, by computing its *Minimum Spanning Tree* (MST; Prim, 1957) with edges U weighted according to the pairwise matching cost ρ (Eq. (6.4)). The MST is a sub-graph $\mathcal{G}'_S(V, U')$ in which each scan (vertex V) is only connected to the best-matching adjacent scan. We point out that in the absence of a global coarse registration, the MST can potentially also prune incorrect pairwise misalignments. However, this procedure has no robustness and will only work if the pairwise costs ρ are reliable, for two reasons: first, one commits to the edges with lowest ρ at the point of constructing the scan graph, without checking loop closures; and second, one trusts that any two scans are connected by such edges, otherwise pruning will fragment the network.

Having found the MST, we refine the transformations (edges U') of $\mathcal{G}'_S$ with conventional ICP, i.e., alternation between point-to-point assignment with the nearest neighbour rule and minimization of the (appropriately truncated) sum of Euclidean distances over all corresponding points. Note in this context that corresponding scan points may still exhibit significant residuals after coarse registration with K-4PCS (and hence after global coarse alignment, which does not change the pairwise transformation parameters), because the transformations at this point have been estimated from only four keypoint correspondences. Also recall that the ICP error is highly non-convex and prone to getting stuck in weak local minima. We thus employ a multi-scale strategy and gradually home in on a better solution with smaller convergence basin, by iteratively decreasing the maximum allowable point-to-point distance in the correspondence rejection step. ICP iterations continue until the point-to-point distances reach the resolution of the point clouds, i.e., the size of the voxel grid cells.

The minimum spanning tree is cycle-free, therefore applying ICP only along its edges will result in error build-up from the root of the tree to the leaves. Adjacent scans of the original scan graph $\mathcal{G}_S$ will have high relative accuracy if they are linked by an edge of the MST, whereas adjacent scans that belong to different branches of the MST will exhibit larger misalignments, since their relative transformations have not been refined directly. To globally redistribute the registration errors across the scan network we run LUM, in the six DoF version introduced by Borrmann et al. (2008). For each scan pair within the network, the method calculates the

alignment errors and covariance matrix based on corresponding points (i.e. nearest neighbours up to a maximum distance to prune false correspondences).

The final scan orientations within the network are then calculated using maximum-likelihood estimation by minimizing the pairwise Mahalanobis distances between corresponding points. Note that in each iteration the method recalculates the alignment errors and covariance matrices, but not the nearest neighbour assignment. Therefore, the initial residuals should be at most as large as the point distance (voxel grid size) within a scan, which explains the need for the preceding pairwise fine registration.

6.5 Experimental Evaluation

We evaluate our framework on six different TLS datasets, each featuring different challenges. To generate ground truth, all datasets were manually aligned using in-scene targets (if available) or natural point correspondences. The manual coarse registration was refined in a commercial software (Geomagic) using multi-view ICP. The registered reference datasets are shown in Figure 6.5, and described in more detail in Section 6.5.1. Apart from the resulting reference positions of all scans in a common coordinate frame, we also retrieve pairwise alignment information, i.e., the pairwise transformation parameters $\mathbf{T}^\star(\mathbf{t}^\star, \mathbf{R}^\star)$ and the point correspondences after the last ICP iteration, which we use as true correspondences $C^\star = \{c_\mathcal{S}, c_\mathcal{T}\}$ for error estimation. The accuracy of the ground truth $\sigma^\star$ is given by the *Root Mean Square Error* ($RMSE$) between points in $C^\star$.

We experimentally evaluate first the complete framework and second the global coarse registration step into a consistent network, which has been the focus of the present paper. For that part we also perform a comprehensive analysis and evaluate the influence of the most important parameters. For detailed parameter studies of K-4PCS we refer the reader to our previous work (Theiler et al., 2014a,b). Since the 4PCS algorithm is a variant of RANSAC, each run will lead to a different result. We therefore repeat each test $m = 100$ times, and record success/error rates as well as average runtimes over all m trials. The Final alignment after fine registration is additionally evaluated in terms of mean metric accuracy. All experiments were run on 64 Bit Windows 7 desktop computers with 16 GB of RAM and twelve 3.2 GHz CPU cores.

6.5.1 Test Datasets

We use four outdoor and two indoor scan projects for our experiments. A summary of all test datasets is provided in Table 6.1 including details like the number of scans, average number of

points per scan, number of scan pairs, and the application of the underlying project. Datasets were acquired with a *Zoller+Fröhlich Imager 5006i* and a *Faro Focus 3D*. Both sensors have very similar acquisition characteristics w.r.t. the quality and appearance of the acquired data, thus we handle the resulting data equally.

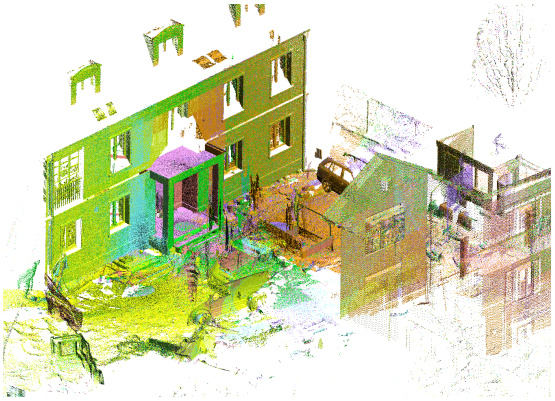
	Dataset	#Scans	#Pts/Scan	#Pairs	Application
Outdoor	"Façade"	7	25M	21*	Urban Surveying
	"Arch"	5	26M	8	Heritage Documentation
	"Courtyard"	8	13M	28*	Archaeology
	"Trees"	6	20M	10	Forestry
Ind.	"Office"	5	10M	10*	Indoor Navigation
	"Facility"	23	11M	61	Facility Management

Table 6.1: Description of the test datasets. (*) stands for fully connected scan graph without adjacency assumptions.

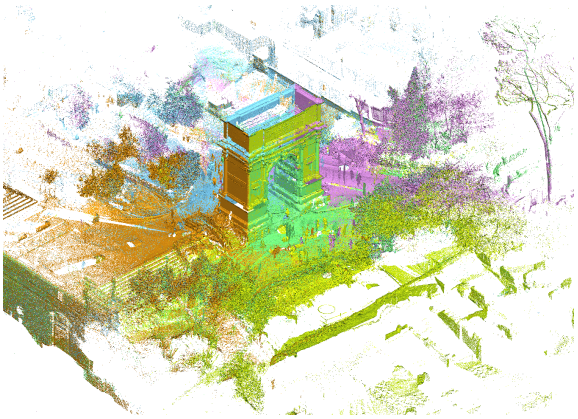
Generally speaking, outdoor scan projects often cover wide areas and, compared to indoor projects, feature larger translations between scan positions. This, in turn, usually yields smaller overlaps between adjacent scans. A typical outdoor dataset in this sense is "Arch"[5] with up to $25\,\mathrm{m}$ distance between adjacent scans, and associated overlaps $< 40\%$ of the points. Furthermore, outdoor environments are less controlled, and moving objects like people or cars can cause artefacts (present e.g. in "Arch" and "Façade"). Scan projects that contain mainly vegetation are particularly challenging to register, because they lack salient, repeatable and unambiguous structures. We use "Trees" to test a rather extreme case, with scans acquired in a forest, and with low overlap of $\approx 40\%$. Another challenge for registration are predominantly flat scenes with a limited amount of vertical objects. This situation is covered to some degree by the "Courtyard" dataset, showing the entrance to an ancient Egyptian tomb and its surroundings.

In indoor applications, in our case especially the "Office" dataset, rotational symmetry can lead to ambiguous, almost equally good pairwise registrations. The limited extent of indoor spaces typically yields larger overlaps (here $\approx 80\%$). "Facility" is a limiting case. It shows a complicated industrial environment, where pipes and other obstacles limit the scan overlap and the number of available point-to-point correspondences. The dataset shall illustrate the breaking point of the proposed system – we note that for several scan pairs we were even unable to do a manual registration without relying on the artificial markers. The dataset is also quite large, consisting of 23 scans, respectively 61 pairs with reasonable overlap.

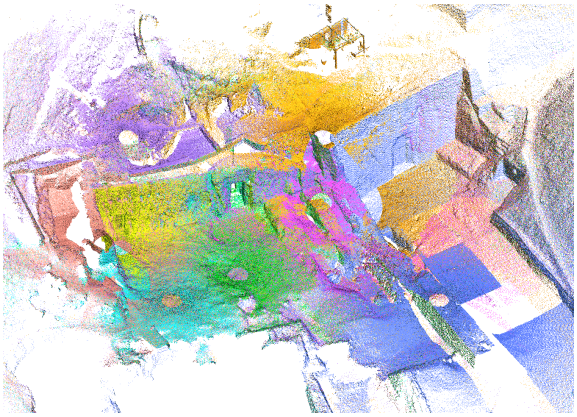
[5]Compared to Theiler et al. (2014a) we use an additional fifth scan of the "Arch" which overlaps even less with the rest of the data.



(a) “Façade”

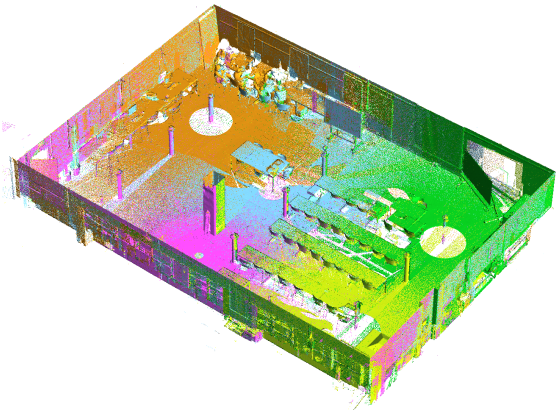


(b) “Arch”

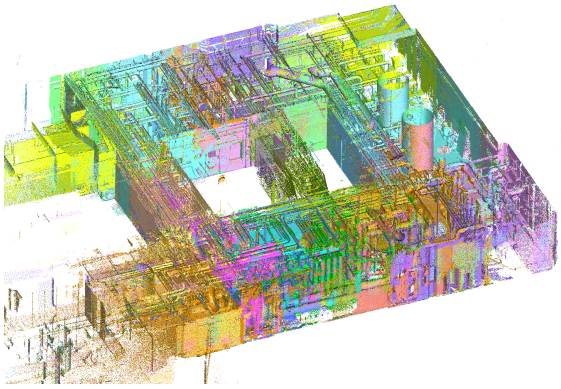


(c) “Courtyard”

(d) “Trees”



(e) “Office”



(f) “Facility”

Figure 6.5: TLS test datasets used in the experiments. For a better visual impression only a small portion of the points is displayed. Moreover, the ceilings of the indoor datasets (e) and (f) have been removed (only for visualization).

For all datasets, we apply a voxel grid filter with a voxel size of 0.1 m, keeping the centroid of all scan points in each voxel. From the filtered scans, we extract 3D DoG keypoints for “Office”,

"Facility" and "Façade", and 3D Harris keypoints for the remaining datasets.[6] Parameters are adjusted such that $1000 - 5000$ keypoints are found per scan.

6.5.2 End-to-end Evaluation

The overall goal of our framework is to register all scans in a common reference frame. We thus first evaluate the complete system, meaning that we start from raw scans and assess the quality of the final scan network. Scanner positions and orientations, given by a transformation matrix $\mathbf{T}(\mathbf{t}, \mathbf{R})$ relative to the network origin, are directly compared to the ground truth. The origin is anchored at one of the local scan coordinate systems, see below. We first check for inliers $\mathcal{I}$ (i.e. correctly registered scans) based on their position $\mathbf{t}$ in the network. Given the ground truth with position $\mathbf{t}^\star$ and accuracy $\sigma^\star$ we define a successful registration as in Eq. (6.7).[7]

$$GS = \begin{cases} 1 & \|\mathbf{t} - \mathbf{t}^\star\|_2 \leq 3 \cdot \sigma^\star \\ 0 & \|\mathbf{t} - \mathbf{t}^\star\|_2 > 3 \cdot \sigma^\star \end{cases} \tag{6.7}$$

We then calculate an overall scan success rate GSR, by simply counting the number of correctly registered scans over all pairs and all test runs. Second, we also quote the project success rate TSR, where a run counts as correctly registered only if all scans are successfully aligned. This rather strict definition corresponds to the goal of fully automatic registration (arguably, a system that registers e.g. nine out of ten scans automatically and needs three manual clicks for the last scan would still be useful to the end-user). To verify the metric accuracy of the alignment, we compute the inliers' mean position error $\varnothing\varepsilon_t$ and the mean orientation error $\varnothing\varepsilon_\theta$ in 3D axis angle representation.

$$\begin{aligned} \varnothing\varepsilon_t &= \tfrac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \|\mathbf{t}_i - \mathbf{t}_i^\star\|_2 \\ \varnothing\varepsilon_\theta &= \tfrac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \measuredangle(\mathbf{R}_i \cdot \mathbf{R}_i^\star) \end{aligned} \tag{6.8}$$

For completeness, we note that to compare scan positions and orientations we need to fix the coordinate system. The datum definition will fail whenever an unsuccessful scan is involved. We thus let each scan position in turn define the coordinate system and count the inliers for

[6] As a general rule, 3D Harris keypoints work slightly better for outdoor scenes, whereas 3D DoG keypoints are well-suited indoors, and more generally for smooth surfaces with high-contrast texture in the intensity.

[7] Although this metric does not directly include the rotations, we found that scans with low translation error also have low rotation error, which is somewhat intuitive given the 360° field of view and the vanishingly small chance of a rotation-invariant scene.

each of them. The datum with the highest inlier count is kept, such that the success rates and accuracies refer to the largest correctly aligned sub-network.

	GSR	TSR	$\varnothing\varepsilon_t$ in mm	$\varnothing\varepsilon_\theta$ in mdeg
"Façade"	100%	100%	8.8 ± 1.0	27.2 ± 6.4
"Arch"	98%	90%	11.4 ± 10.5	18.9 ± 11.3
"Courtyard"	100%	100%	27.8 ± 1.0	18.2 ± 1.1
"Trees"	97%	92%	3.6 ± 0.7	16.2 ± 2.3
"Office"	100%	100%	5.4 ± 0.5	36.3 ± 8.2
"Facility"	53%	0%	6.7 ± 6.7	29.1 ± 19.4

Table 6.2: System-level results. Overall scan success rate GSR, project success rate TSR, and average registration errors $\varnothing\varepsilon_t$ and $\varnothing\varepsilon_\theta$.

Table 6.2 shows the final results of our proposed framework given the six test datasets; i.e., the scan success rate GSR, the project success rate TSR as well as the registration accuracies represented by $\varnothing\varepsilon_t$ and $\varnothing\varepsilon_\theta$. For three of the six datasets we achieve $GSR = 100\%$, which means that in 100 trials not a single scan was misregistered. Consequently, also at the project level we reach a perfect $(TSR = 100\%)$. Even for cases with low scan overlap ("Arch") and difficult conditions ("Trees") we still achieve scan registration success rates of $\geq 97\%$, corresponding to a $\geq 90\%$ chance that the whole project is registered correctly. To correct the unsuccessful runs, it would in most cases suffice to manually register one scan.

The "Facility" dataset shows the limits of our method. At first glance it seems that the proposed framework fails completely, considering the TSR of 0%, meaning that no test run was able to reproduce the whole network correctly. Still, on average $>50\%$ of all scans could be registered correctly. A closer inspection reveals that the framework consistently produces two large sub-networks which are, each by themselves, correctly aligned. However, the connection between the two parts is weak and the system does not manage to put them together (see Figure 6.6), so only the scans in the larger part count towards the GSR. Hence, even this seemingly catastrophic failure could in most cases be remedied by manually identifying three correspondences.

As a general comment along those lines, we point out that the global verification of consistency delivers explicit information about network stability, respectively local weaknesses already during coarse registration. That is, for each edge U in the scan graph we get not only the globally most consistent pairwise alignment, but also whether it is included in a consistent loop, and thus controlled by redundancy. In a practical end-user software, one can thus automatically

diagnose local weaknesses in the network, and give feedback which alignments to manually check and correct, if necessary.

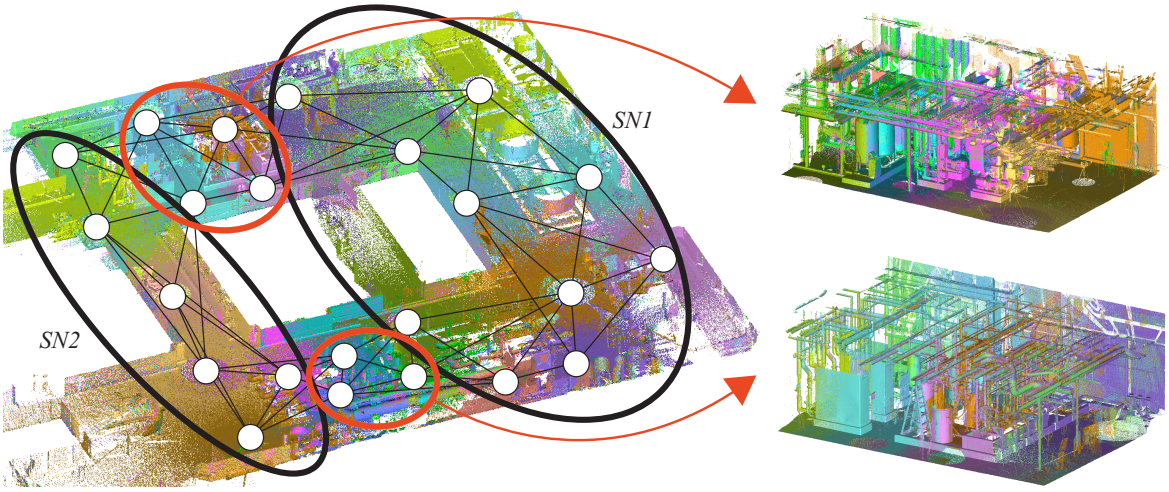


Figure 6.6: Registration of dataset "Facility". The framework produces two correct sub-networks *SN1, SN2* but these cannot be registered w.r.t. each other. Red ellipses mark the areas – visualized on the right with removed ceiling and front wall – where too many pairwise misalignments due to very high scene complexity occur.

The registration accuracy of the correctly registered scans matches the expectations: we generally achieve translation errors on the order of 10 mm. Relative to the $\tau = 100$ mm resolution (voxel grid spacing) of the input point clouds, this amounts to $\varnothing\varepsilon_t < 0.1 \cdot \tau$. Notice how the lack of vertical structures affects the location uncertainty for the "Courtyard". One can also observe a trend that indoor scans are registered a bit more accurately in terms of position, which appears to be caused by the larger overlap and the presence of numerous orthogonal planes near the scanner. The inverse seems to be true for rotations. The limited baselines in indoor spaces lead to slightly lower rotation accuracy.

In Table 6.3 we quote the runtimes of the framework. The total time T_{Total} is further divided into the different registration steps; pairwise coarse registration with K-4PCS ($T_{\text{K-4PCS}}$), global consistent coarse registration (T_{GCA}), pairwise refinement with ICP (T_{ICP}) and global optimization with LUM (T_{LUM}). The current system is able to register scan projects in practically relevant time. Starting from raw scans with up to 50M points, processing times range from less than two minutes for the five scans of "Office" to just under one hour for the 23 scans of "Facility". Our prototype is implemented in *C/C++*, but not yet optimized for maximum speed.

The main computational burden is the pairwise coarse registration with K-4PCS which, in the absence of additional information, has to be carried out for all potentially overlapping scan pairs. The runtimes also clearly show how the computational cost of pairwise coarse registration

	$T_{\text{K-4PCS}}$ in s	T_{GCA} in s	T_{ICP} in s	T_{LUM} in s	T_{Total} in s
"Façade"	190 ± 6	<1	8 ± 1	2 ± 0	200 ± 6
"Arch"	1030 ± 13	<1	103 ± 30	8 ± 0	1141 ± 32
"Courtyard"	342 ± 4	<1	39 ± 1	17 ± 0	398 ± 4
"Trees"	1840 ± 27	<1	40 ± 3	7 ± 0	1887 ± 28
"Office"	97 ± 7	<1	3 ± 0	1 ± 0	101 ± 7
"Facility"	3493 ± 74	2 ± 0	36 ± 11	5 ± 1	3536 ± 74

Table 6.3: Runtime of the current implementation in seconds. We show the total runtime T_{Total} as well as the time needed for individual steps of the pipeline.

grows with decreasing overlap. The runtime for the proposed global coarse registration is negligible, with values ≤ 2 s. Hence verifying global consistency does not significantly increase the runtime, while it ensures that the coarse alignment ends, as far as possible, with a globally consistent network.

6.5.3 Coarse Registration Results

The main innovation of this paper is to exploit loop constraints already at an early stage, to achieve a coarse alignment that is globally consistent across the scan network. We therefore separately evaluate the gain through this additional step (in the following abbreviated as GCA), compared to the baseline of directly using the best pairwise K-4PCS registrations. The primary goal of the global consistency check is to weed out erroneous pairwise registrations. As error metric we therefore use the number of misregistered scan pairs. Coarse registration is not a goal in itself, but only a precursor for fine registration, thus it is not trivial to decide when the coarse orientation of two scans has been "successful". We propose the following, rather conservative definition: for each coarse alignment, run an independent fine registration with ICP; if that step converges to a low residual error, the coarse registration was certainly good enough as initialization and is counted as correct. The error after ICP is measured by the $RMSE$ of point-wise residuals, evaluated at the true correspondences $C^\star$ from the ground truth.

$$PS = \begin{cases} 1 & RMSE_{\text{ICP}}(C^\star) \leq 3 \cdot \sigma^\star \\ 0 & RMSE_{\text{ICP}}(C^\star) > 3 \cdot \sigma^\star \end{cases} \tag{6.9}$$

Given the definition of a successful pairwise, coarse alignment PS in Eq. (6.9) we derive success PSR and error rates PER over all m trials and all scan pairs per dataset. Throughout,

we regard the pairwise registration step as a black box and use the same default parameters. A detailed study of K-4PCS and its tuning parameters can be found in (Theiler et al., 2014a,b).

Parameters for the graphical model are also kept constant for all datasets. Inference is done with 1000 rounds of the Lazy-Flipper algorithm. As a sanity check, we have compared the Lazy-Flipper to brute-force exhaustive minimization (only for the smaller "Office", "Arch" and "Forest" datasets). Both methods returned the same result, while the Lazy-Flipper naturally is a lot faster. The unary weight is set to $\frac{1}{\omega} = \frac{1}{2}$. The unary cost for a virtual solution is $\rho = 1$; moreover, higher order cliques containing one or more virtual solutions are always assigned a cost $\psi = 0.6$. As truncation bounds for the loop closing error we set $t_{pair} = 0.5\,\mathrm{m}$ and $\theta_{pair} = 5°$, corresponding to the empirical rotation and translation accuracies of unrefined pairwise K-4PCS registration. Ordering K-4PCS candidates by their matching cost ρ is in many cases fairly reliable. Since this is exploited by the iterative inference scheme (cf. Section 6.4.2) we nevertheless allow for a maximum of 500 labels per vertex x_i, leading to at most ten iterations. That number is rarely reached, because we terminate early if all vertices are controlled by a valid loop, or if the total energy did not decrease during four iterations.

Figure 6.7 summarizes the results of stand-alone K-4PCS, and those after consistency verification (GCA). We distinguish between correct pairwise alignments (green) and misalignments (red), according to Eq. (6.9). Interpreting the results is complicated by the fact that after the global consistency check virtual solutions may exist. We make the two possible cases explicit: *(a)* picking the virtual solution is the optimal choice (VSR), because no correct candidate is present; or *(b)* the optimization chooses a virtual solution, although a correct candidate would have been available (VER). The latter case does not harm the final registration result (pairs that have been assigned a virtual solution are simply ignored in the subsequent MST calculation), nevertheless it is not the desired outcome, in the sense that a better solution of the consistency check would have been possible. As shown in Figure 6.7 stand-alone K-4PCS already achieves $PSR > 80\%$ even for challenging datasets such as "Arch" or "Trees" with low overlap and complex scenery. In the case of "Office" it reaches 99% (i.e. ten misaligned scan pairs out of 1000), in spite of a high degree of symmetry. The limits of pure pairwise registration are again illustrated by "Facility" with only $PSR \approx 58\%$, due to complicated tube structures and obstacles that impair the repeatability of the keypoints.

We go on to compare these results to those after GCA. Most importantly, the loop constraints consistently reduce the number of misaligned pairs: PER decreases significantly, in most cases to less than half of the value before GCA. The biggest part of the improvement ($1 < VSR < 11$ percentage points) comes from identifying pairs without any suitable candidates, and correctly assigning the virtual solution. Moreover, PSR improves up to three percent points, which indicates that the optimization manages to uncover "hidden" candidates which are present, but

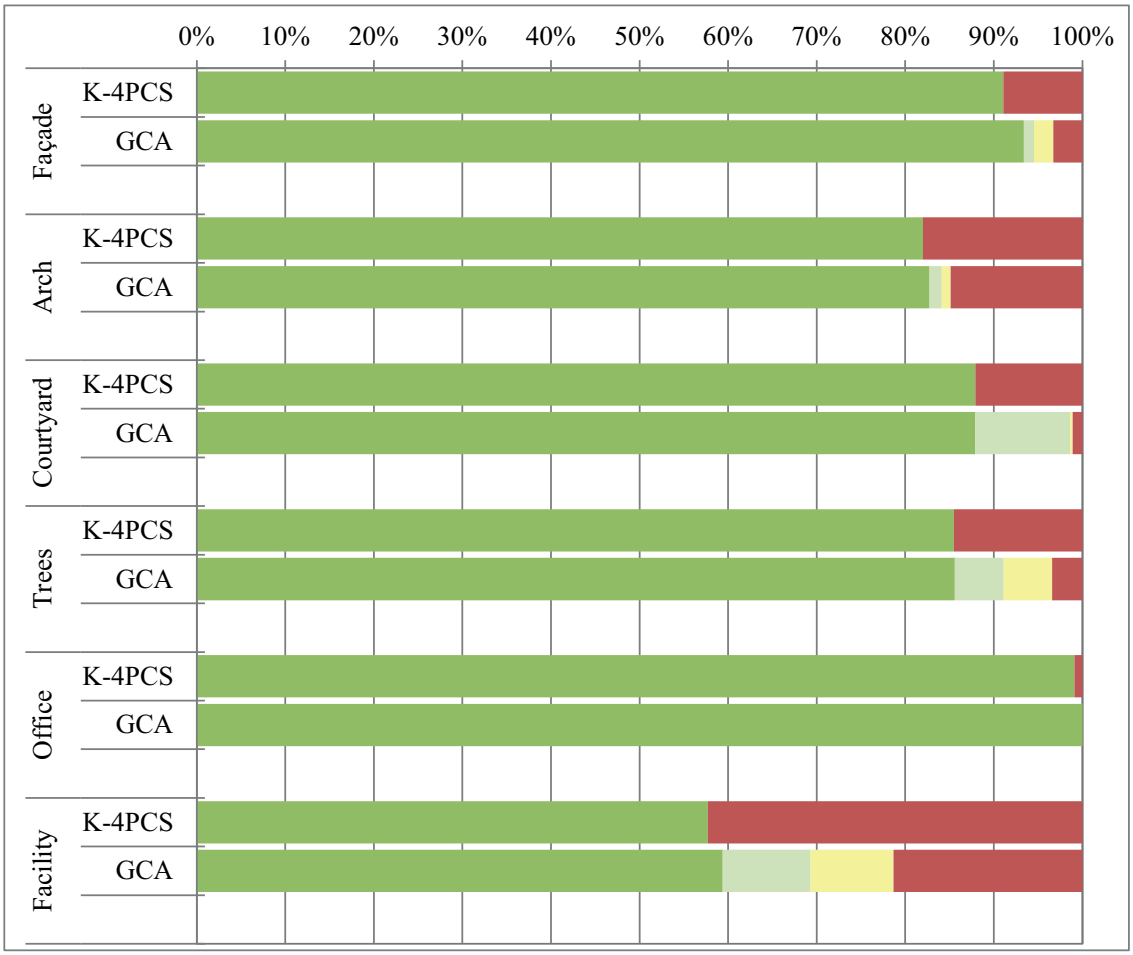


Figure 6.7: Comparison of pairwise coarse registration (stand-alone K-4PCS) and globally consistent coarse alignment (GCA). The most important quality measure in the overall context is the error rate (PER, red). We furthermore show success rates (PSR, green) as well as the fraction of virtual solutions that were returned correctly (VSR, light green) or unnecessarily (VER, yellow).

not ranked first in terms of pairwise cost ρ. On the negative side, the global coarse registration is not able to eliminate all pairwise errors, as evidenced by the remaining PER. For four out of the six datasets only a small amount of errors remain ($<4\%$), whereas a much larger fraction of wrong pairs survive for "Arch" (15%) and "Facility" (21%). We found two different sources for these errors:

(A) Pairwise K-4PCS generates too few correct alignments. As more and more pairs have no correct candidates (and need a virtual solution), we also lose loop constraints, until eventually the network fragments into parts that are not connected by valid loops. In such cases (e.g. present in "Facility") the optimization can base its decisions only on the unary ρ, and the potential gain (locally) vanishes.

(B) The optimization yields pairwise alignments which are globally consistent, but not accurate enough to let ICP converge to the ground truth. That case is the reason for most of the remaining errors. It is particularly prominent for “Arch”, where the convergence radius of ICP appears to be particularly small, perhaps due to low overlap in combination with repetitive structures and scanning artefacts.

We point out that in the overall framework the MST can compensate quite a few type (B) errors – see results for “Arch” in Section 6.5.2. Note, the inverse problem to (B) can occur, too. The behaviour of ICP is data-dependent, and in some cases candidate transformations exceed the consistency tolerance and are dropped, although they would have sufficed as initialization. Such cases inflate VER at the cost of PSR.

To summarize, the graph-based global coarse alignment significantly reduces pairwise misalignments that occur with unchecked pairwise registration. A small amount of errors remain undetected, mainly because of roughly consistent alignment that are too far off for ICP refinement.

Empirically, the crucial parameters of the graphical model are the estimated translation and rotation uncertainties t_{pair}, θ_{pair} of a (coarse) pairwise transformation, which are needed to compute the higher-order terms (cf. Section 6.4.2). We therefore conduct additional tests for varying accuracies $t_{pair} \in \{1\,\text{m}; 0.5\,\text{m}; 0.25\,\text{m}; 0.125\,\text{m}\}$ and $\theta_{pair} \in \{10°; 5°; 2.5°; 1.25°\}$.The corresponding success and error rates on the “Façade” and “Trees” data are shown in Figure 6.8. Experiments on further datasets yield similar results and conclusions.

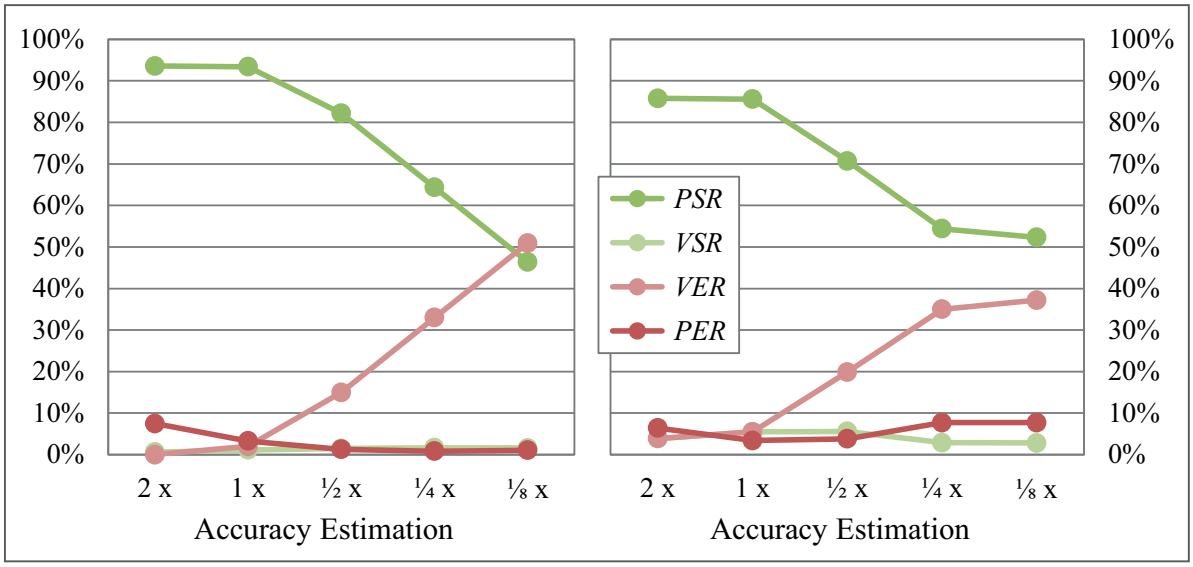


Figure 6.8: Influence of the a-priori transformation accuracy on the success and error rates of “Façade“ (left) and “Trees“ (right). Transformation accuracies are given relative to the standard parameter $t_{pair} = 0.5\,\text{m}$ and $\theta_{pair} = 5°$.

Figure 6.8 reveals a clear influence of the transformation accuracy on the different evaluation criteria. In case of high uncertainty $[1\,\mathrm{m}; 10^\circ]$ (equals "2x") we achieve a similar number of correct pairwise alignments as with our default values ("1x") but at the same time more misalignments slip through. Fewer virtual solutions are chosen, which increases the chance of error type (B). With the default values $[0.5\,\mathrm{m}; 5^\circ]$ the PSR can be retained, while reducing the PER to $\approx 3\%$. Further decreasing the thresholds continues to improve VSR and hence reduces PER. Simultaneously, the rate of unnecessary rejections VER strongly increases, at the cost of PSR. In the case of "Trees" the trend flattens with $[0.125\,\mathrm{m}; 1.25^\circ]$ ($\frac{1}{4}$ of the defaults), since there are hardly any consistent loops left, and the graphical model degenerates to an expensive way of picking the pairwise registration candidates with the lowest costs.

In short, stricter t_{pair} and θ_{pair} will, as expected, reject more misalignments and increase VSR. But as an unwanted side effect they also reject many correct alignments and strongly increase VER at the cost of PSR. The proposed default values appear to be – at least for our scan data – a reasonable trade-off to filter most misalignments while avoiding unwarranted rejections that weaken the network.

6.6 Conclusions and Outlook

We have described a complete framework for automated, marker-less registration of multiple TLS point clouds into a common coordinate system. A unique feature of the proposed framework is the addition of a coarse global registration step. Like most multi-scan registration methods we start with pairwise coarse alignment, in our case via K-4PCS. However, rather than commit to a single best solution for each pair, we retain a list of candidate solutions. Then, still at the level of coarse registration, we set up a graphical model to make explicit use of the redundancy afforded by loops of three or more scans in the network. By discrete inference in the graphical model, we pick one of the candidate transformations for each scan pair, such that the alignment is globally consistent throughout the scan network. The global consistency check ensures that the starting values for the subsequent refinement are reliable ("robustness through redundancy"), at negligible computational cost. Refinement with pairwise ICP and global LUM completes the registration pipeline.

Experiments on six different datasets demonstrate that the framework is applicable to challenging, real-world TLS projects, in most cases reaching success rates of $90\% - 100\%$. Final registration residuals are around $1\,\mathrm{cm}$, corresponding to the expected uncertainty of the raw scan data, respectively 10% of the point cloud resolution after voxel grid sampling. The computation time depends mainly on the amount of overlap between individual scans. With

the current prototype implementation it ranges from less than two minutes for five scans of an indoor project to just below an hour for the most difficult dataset with 23 unordered scans of medium overlap.

The main source of registration failures is the initial pairwise coarse registration. In complicated cases that step still misses too many correct transformations to build up consistent loops. Since the proposed global coarse registration is based on loop constraints, it cannot improve the results in their absence. Thinking about a practical deployment in an user software, the global consistency check has the convenient property that, even if it fails, it still provides information about the location of the problem. It is thus in any case a suitable basis for semi-automatic procedures with self-diagnosis.

A smaller number of failures remain due to the brittleness of ICP. I.e., coarse alignment in principle succeeds, but the subsequent refinement fails, in spite of a reasonable initialization. We see two main strategies to address these irksome failures; either use a more accurate coarse registration or apply a less delicate fine registration. The first option appears less attractive, because it shifts computations to the coarse stage that need high point density. This is against the philosophy of a coarse initialization, and would almost certainly slow it down a lot. The second option goes in the direction of robust ICP variants (e.g. sparse ICP; Bouaziz et al., 2013). Although we do take standard measures like truncating the point-to-point distances and multi-scale computation, an even more sophisticated fine registration scheme is certainly desirable. Unfortunately, no clear standard has emerged yet.

On a general note, our framework is designed for unfiltered TLS point clouds with outliers (e.g. moving objects, multipath effects). Such gross errors are sometimes removed prior to registration which potentially increases the chance of ICP converging to the desired minimum, but we purposely avoid such preprocessing. Arguably, cleaning individual scans requires a greater amount of user input than manual registration, or strict assumptions about the observed scene.

Chapter 7

Conclusions and Outlook

7.1 Synopsis of Results

The methods and results presented in this thesis are the outcome of a four-and-half-year study on automated registration of laser scanner point clouds from static terrestrial laser scanners. The major milestones of the research undertaken within this study are described in the core chapters (Chapters 3, 4, 5, 6). All proposed methods aim at the automation of TLS point cloud alignment without the need for markers and without prior knowledge about the initial positions of the scanners.

First, a 3D keypoint extractor and descriptor derived from intersecting plane triplets have been developed. Matching of the keypoints is performed by combining geometrical and descriptor information within a RANSAC scheme. The method yields coarse pairwise alignments which are already quite accurate due to the high redundancy of plane fitting, but can nevertheless be refined with ICP, if desired. The resulting approach is well-suited to register TLS point clouds in man-made environments, but depends on the presence of planar surfaces within the scenes, and thus has a limited application range.

To be more general, second, a geometrical point-matching method (4PCS) was adopted for TLS point cloud alignment. The resulting registration scheme, called K-4PCS, combines the advantages of 3D keypoints with the efficient, geometrical constraint matching algorithm. The proposed method is very generic and can handle any keypoint type, because it does not require descriptors, but is solely based on the geometric layout. We could show that the method successfully registers pairs of laser scans from real-world and challenging datasets in various environments ranging from indoor to extreme outdoor cases such as forested areas.

The K-4PCS pairwise alignment scheme was, in a third step, engineered to even better cope with the special challenges of TLS point cloud registration. On the one hand, we considerably reduced the runtime of pairwise alignment by parallelizing the matching across several cores (as

many as available). A further idea was to integrate a soft voting scheme to detect re-occurring alignments during RANSAC iterations, to remove seldom solutions, and hence to reduce the amount of possible alignments prior to the time-intensive evaluation. The approach proves to be not applicable within our method of four-point matching and was discarded. On the other hand the evaluation function to define the optimal coarse alignment was modified and extended to support *(i)* more accurate solutions using a residual-based cost function, and *(ii)* to favour scanner positions at larger distances by integrating a cost term for too-small alignment translations.

In a final step, we used the pairwise alignment scheme as a building block for a complete registration pipeline which is able to automatically align multiple point clouds. The framework is again generic and allows the use of different approaches for coarse as well as fine registration. In the presented experiments we used K-4PCS for initial, pairwise alignment of scans, but one can replace it by any other method which is able to produce multiple pairwise alignment hypothesis. A global verification step based on loop constraints over three or more scans has been developed which ensures global consistent pairwise alignments already on a coarse registration level. For fine registration standard methods are exploited, i.e., pairwise ICP followed by a global refinement to remove the accumulated error from consecutive pairwise alignment. The achieved high success rates for several real-world datasets proved the general applicability and effectiveness of the proposed methods for automated TLS point cloud registration, while neither markers nor initial orientations of the scans are required.

7.2 Discussion of Contributions

As stated in Section 1.2, the overall aim of this thesis was to develop methods for the automated, efficient, marker-less registration of TLS point clouds. In the following, a systematic overview of the main strengths and weaknesses of the different methods developed for this task is given.

7.2.1 Plane-based Registration

The method described in Chapter 3 aims at the pairwise registration of TLS point clouds using 3D keypoints. Instead of using a local interest point operator, keypoints are generated by detecting and intersecting plane triplets. A novel 3D point descriptor to encode the keypoints holds characteristics of the underlying planar surfaces and geometrical properties of their intersections. Note that many keypoints found by plane intersection could not be extracted with a conventional keypoint detector, because they are purely virtual and do not correspond to

a physical corner of the scanned surfaces. Because we only need to see part of the underlying planes to extract the respective keypoint, the proposed detector is especially valuable if a large number of obstacles are present in the scanned scene.

Test results using stand-alone descriptor-based matching demonstrated that the point description is not reliable enough. Hence the descriptor is used to initially prune wrong matches, followed by geometrically constrained matching based on the remaining point correspondences. This behaviour is in line with the well-known observation, that keypoint descriptors become unreliable with strongly changing viewpoints. In conclusion, it is always recommended to either use descriptor-based matching in conjunction with geometrical constraints (e.g. relative distances between keypoints), or else to exploit the descriptor distances between keypoints to prune implausible matches prior to geometrical constraint correspondence search.

Due to the fact that the keypoint detector is based on naturally available planar surfaces, we have concurrently developed a RANSAC based plane segmentation scheme for TLS point clouds based on their range image representation. The extraction does not require point normals and is thus very robust against noise. To increase efficiency, the segmentation uses multi-scale (range) image pyramids. Because the underlying plane fitting is highly redundant, the resulting initial alignments are quite accurate, but can still be refined using standard ICP. The practicability of the method was demonstrated on an indoor dataset with well-defined planes such as the four room walls. Although most outdoor datasets may also contain many approximately planar regions, preliminary tests showed that the method cannot reliably match scenes with fewer and less precise planes.

In conclusion, although the approach works well for man-made scenes, we did not further pursue the method. Since registration based on e.g. planes inevitably depends on the presence of the specific primitives, this line of thought was abandoned and the focus was set on developing more generally applicable approaches.

7.2.2 Keypoint-based Registration

In Chapter 4, a second, purely geometric registration approach was presented, which focuses on efficiently finding congruent keypoint configurations. The proposed method (K-4PCS) does not require any point descriptors and is thus generic with regard to the applied keypoint extractor. We so far conducted tests with 3D DoG and 3D Harris keypoints.

The underlying idea is (again) to reduce the large scan to a sparse number of highly discriminative keypoints. We found that extracting keypoints within image representations yields an unfavourable, uneven distribution, i.e., due to the near-field bias produced by laser scanners

more keypoints are detected close to the scanner. We therefore propose to directly extract keypoints from the 3D point cloud. To further reduce the near-field bias, a preceding voxel grid filter is applied to even out the point density. Choosing a suitable voxel grid size proved to be quite important. The optimal grid size is thereby a compromise between keeping as much detail as possible, but at the same time evening out the point density in the overlapping area. Assuming a user knows about the regular sampling size (i.e. the angular measurement interval) of the acquired scans, the resulting point sampling distance in the estimated overlapping area can be calculated and thus this parameter can be set in a data-driven manner.

Correspondences are detected using only geometrical information within an efficient four-point sampling strategy. Compared to sampling the minimum of three points, base sets of four coplanar points reduces the computational complexity to $\mathcal{O}(n^2)$. Matching sets of 3D keypoints solely based on their relative geometric position thus becomes tractable. Note, with increased scene complexity (yielding typically large number of keypoints) and decreasing scan pair overlap, runtimes again reach critical dimensions relative to manual registration.

Experiments on four real-world datasets demonstrated the effectiveness of the matching approach for the task of pairwise coarse registration. The method produce alignments with high success rates which can be finalized by standard ICP refinement. Experiments with synthetic and real scenes show that errors mainly occur in scenes with many repetitive or symmetric structures. In particular, we found that the often applied objective function, which calculates the to-be-maximized overlap as the fraction of inliers (i.e. close neighbours between the aligned point clouds), does not always produce the globally correct solution in case of TLS point cloud registration.

To summarize, the initial version of K-4PCS already produced promising results and is generic enough to perform coarse registration in various environments ranging from simple to very complex scenes. But two major drawbacks remain, i.e., the rather large runtime in case of complex scenes and/or low scan overlap, and the sensitivity w.r.t. rotational-symmetric and repetitive structures. These challenges were addressed by the approaches explained in Chapter 5. A straight-forward solution is to distribute RANSAC iterations onto multiple cores and run the K-4PCS matching scheme in parallel.[1] We could thus speed-up the matching by a factor close to the number of available cores. Given the fact that over the last ten years the amount of cores in modern computers have increased from one to currently twelve (or even more), we can expect that the gain will further increase. Also, moving computational expensive operations from CPU to GPU could further reduce the runtime significantly.

[1] Recall that the parallel version has after revision already been implemented in Theiler et al. (2014a), but the idea was originally presented in Theiler et al. (2014b)

Another line of thought was to reduce the runtime required for the computationally expensive evaluation of valid four-point matches. We proposed a soft voting scheme to find a small set of possible alignments prior to the actual evaluation, and to only evaluate re-occurring solutions. This reduces on the one hand the time spent on re-evaluating similar alignments, and on the other hand, one-off solutions are not evaluated at all. The basic principle behind this is, that assuming the correct solution produces the largest amount of inliers (i.e. close neighbours after alignment), the chance of picking a valid base sample that leads to that solution is also large and should thus re-occur more often. Although this idea has successfully been used in other research areas (e.g. in image matching; Torii et al., 2011), the underlying principle does not hold in the case of four-point matching. It appears that although the correct configuration does bring a large number of keypoints into correspondence, there are not necessarily many co-planar quadruples in the sparse keypoint clouds. Hence a correct alignment is not found more often than other, wrong solutions.

The robustness of K-4PCS improves with the extended objective function. Although MSAC does not seem to influence the result, it is still theoretically more sound to base the decision on residuals and not on a hard inlier/outlier threshold. The soft prior towards larger alignment distances proves to be valuable and well-suited for the task of scan registration. It is able to additionally reduce the near-field bias, and thus increases robustness of registering real-world datasets. For special scenarios where smaller translations are present, this prior could simply be switched off without restrictions. In some applications it would also be interesting to additionally use soft prior terms for rotations to favour solutions with e.g. small rotations around x- and y-axis in case of levelled sensor setup, or small z-rotations if rough orientations are known e.g. from a compass. Rotation priors are not yet implemented and left for future work.

Although the experiments showed an increased robustness, pairwise misalignments nevertheless appear. The main source of error is still ambiguous scene geometry (e.g. rotational symmetries, repetitive structures) yielding multiple similarly good alignment solutions. As a final conclusion, finding the correct pairwise alignment solely based on geometrical information derived from the overlap of two scans is sometimes not possible, because the underlying evaluation function is multi-modal. Consequently, one should move this decision onto a global level, taking into account multi-view constraints, as done in Chapter 6.

7.2.3 Multi-view Registration

Finally, a complete registration framework was developed which solves the alignment task for multiple TLS point clouds acquired from arbitrary positions. The main novelty behind the

presented pipeline is to ensure global consistent orientations already on a coarse registration level. The framework starts with K-4PCS pairwise coarse registration to generate multiple, putative alignments. The globally best fitting among those hypotheses are then found in a global consistency verification. Only globally consistent alignments are preserved and refined by the subsequent pairwise ICP. A multi-view refinement removes errors accumulated during successive pairwise alignments.

The results of the experiments with our final registration framework confirmed very high reliability for several tested real-world datasets. Also runtimes are feasible for practical applications (e.g. a fully connected network of eight scans in below seven minutes), although there is still room for improvement (cf. Section 7.3). The framework is thus competitive in comparison to manually achieved registrations even when carried out by an expert.

A general problem of automated approaches is that the returned alignments are not guaranteed to be correct, especially considering the non-convex problem of ICP-based fine registration. Typically, one thus has to visually inspect the alignment result, which reduces the gain of automation compared to manually achieved registration, where the inspection is part of the process. Unlike other automated approaches, our framework ensures to return globally consistent alignments. Moreover, we additionally obtain knowledge about the quality of the resulting scan network, i.e., with the proposed global consistency verification a user receives insights about the local network stability, or how well the network is locally controlled, respectively. This could be exploited in a semi-automated registration pipeline, where the alignment is automatically derived based on our proposed methods, and only if necessary, manually inspected and corrected at the (automatically) detected unstable network locations.

Regarding alignment accuracy, we achieve final position errors of around $1\,\text{cm}$ after global refinement, i.e., a per-point measurement accuracy of typical TLS sensors, respectively the definition accuracy of many standard objects. It is important to note that, apart from the global fine registration method implemented in our framework (i.e. LUM; Lu and Milios, 1997), there are numerous other approaches such as multi-view ICP (Williams and Bennamoun, 2001). Recall that the focus in this thesis was on providing coarse registrations and thus refinement is based on available methods/tools. Hence we did not try to further optimize the fine alignment after reaching acceptable accuracies in practical runtime with the current approach.

Our experiments also provided rather strong evidence that working with a voxel grid sampled point cloud is not only a valid approach for coarse registration, but is also sufficient for fine alignment. In the presented experiments, we have worked with a voxel grid of $10\,\text{cm}$ and have finally achieved registration accuracies of $\approx 1\,\text{cm}$, i.e., $\frac{1}{10}$ of the grid resolution. Empirical tests with lower point sampling distances (or with the original scans) yielded only a minor increase

of the final position accuracy, which indicates that we have (almost) reached the maximum achievable registration accuracy even with coarsely sampled point clouds. A probable reason is that the voxel grids still feature massive redundancy with regard to the six DoF registration task. Simultaneously, processing time strongly increases for coarse as well as fine registration if we work with larger, less densely sampled point clouds.

A further question where this thesis may provide useful insights concerns the robustness of automated registration methods with decreasing point cloud overlap. We have performed tests on various real-world datasets; the extreme case regarding scan pair overlap is the "Arch" project with overlaps of $< 40\%$. The results on this dataset confirmed the practicality of the proposed registration pipeline, and in particular of K-4PCS for pairwise alignment even in the presence of small overlap. However, to ensure the selection of basic random samples in the overlapping area, more RANSAC repetitions have to be carried out with decreasing overlap, which naturally increases the runtime for keypoint matching. An interesting conclusion is thereby, that in some cases it might make sense for a scan operator to acquire more scans in the field and thus producing higher pair overlaps (while he could potentially reduce resolution, too). This is because the resulting "time waste" during acquisition is to some degree already compensated by a much faster automated registration, while reliability increases, due to the higher redundancy and the larger amount of network constraints.

7.3 Outlook

There is a great potential for further applications and improvements of the presented registration algorithms. First of all, up-to-now the different methods have only been tested on TLS point clouds (for which they were designed). But the developed methods are on conceptual level independent of the point cloud type. For example, the planar-based alignment method of Chapter 3 is potentially well-suited for the task of ALS point cloud registration, due to the typically large amount of available planes especially in urban areas. In addition, it will be interesting to test the K-4PCS pairwise registration as well as the whole alignment framework for point clouds derived from other sensors such as KINECT. Moreover, the basic principle of K-4PCS (invariant intersection ratios of the diagonals of coplanar four-point sets) is also valid under affine transformation. Therefore the method is in principle also applicable to point clouds with unknown scale (e.g. from photogrammetry), but would require further work to improve reliability and computational efficiency.

The generic design of our final proposed registration pipeline can be exploited by replacing different parts with useful alternatives. For example, the pairwise coarse alignment is so far

based on K-4PCS, but any method which is able to produce pairwise alignment hypothesis can be applied instead. Especially because the focus in this thesis was not set on fine registration methods, there are many promising alternatives (such as multi-view ICP) to replace the proposed two-step refinement approach.

On a more conceptual note, the general performance regarding efficiency and success rate of the proposed methods could probably be enhanced if we would reduce the DoF of our registration problem. Recall that so far the presented methods do not presume any kind of constraint regarding scanner setup (e.g. verticality). Assuming we can remove two DoFs because the scanner is levelled, the proposed registration pipeline could nevertheless be applied, although K-4PCS would be replaced by a simple point-to-point distance check. I.e., instead of sampling four points it would be enough to search for two points with equal distance to each other (up to some degree). However, global consistency verification and refinement would directly be applicable. Another option to integrate the scanner levelling within our framework is to introduce another soft prior during K-4PCS pairwise matching (similar to the translation prior in Chapter 5) which would favour solutions with low rotations around the x- and y-axis.

Moreover, the robustness of the registration pipeline could be further improved. Remaining errors are either caused by globally consistent alignments which are too inaccurate for the successive ICP to converge to the correct solution, or else because locally too many incorrect pairwise hypothesis occur which yields uncontrolled alignments after the global consistency verification. ICP variants with larger convergence basin or increased robustness against outliers (e.g. sparse ICP; Bouaziz et al., 2013) could potentially reduce the amount of the former errors. By integrating promising ICP variants into a multi-view ICP scheme, one would simultaneously avoid the current two-step approach for refinement. The latter error source has to be addressed in the coarse pairwise registration stage, i.e., either further improve K-4PCS or use an alternative hypothesis generator for pairwise alignment.

On the one hand, especially because K-4PCS does not depend on the employed keypoint type, further research effort could be directed into experimenting with other 3D keypoint detectors. On the other hand, a possible extension of K-4PCS would be to combine the geometrical constraint matching with keypoint descriptors based on local surface geometry, and, if available, colour information, and further spectral or full-waveform characteristics. In addition to the point-to-point distances the four-point matching scheme could then exploit descriptor (dis)similarity to reject unlikely correspondences. Although one would lose to some degree the general applicability of the method, this could significantly speed up the matching and/or avoid misalignments in certain situations.

Another line of thought was recently presented in Mohamad et al. (2014) by introducing a

generalized 4PCS matching approach. The main idea is to remove the planarity constraint on the four-point sets, which results in a less restrictive base point sampling. In case of low overlap (i.e. small amount of corresponding points) this generalization could prove to be very valuable.

Regarding the critical runtime of K-4PCS in complex scenes and/or in case of low scan overlap, another recently proposed method by Mellado et al. (2014) promises to generate a further speed-up of the four-point matching approach. On a final note, as usual, a professional implementation could also further decrease runtime, especially if one would exploit the GPU for time-critical operations.

Bibliography

D. Aiger, N.J. Mitra, D. Cohen-Or, 4-points congruent sets for robust pairwise surface registration. ACM Transactions on Graphics **27**(3), 1–10 (2008)

D. Akca, Full automatic registration of laser scanner point clouds, in *Optical 3D Measurement Techniques VI*, 2003, pp. 330–337

S. Allaire, J.J. Kim, S.L. Breen, D.A. Jaffray, V. Pekar, Full orientation invariance and improved feature selectivity of 3D SIFT with application to medical image analysis, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2008, pp. 1–8

B. Andres, J. Kappes, T. Beier, U. Köthe, F. Hamprecht, The Lazy Flipper: efficient depth-limited exhaustive search in discrete graphical models, in *European Conference on Computer Vision*, 2012a, pp. 154–166

B. Andres, T. Beier, J.H. Kappes, OpenGM: A C++ library for discrete graphical models. arXiv: **1206.0111** (2012b)

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

T. Asai, M. Kanbara, N. Yokoya, 3D modeling of outdoor environments by integrating omni-directional range and color images, in *5th International Conference on 3-D Digital Imaging and Modeling*, 2005, pp. 447–454

K.-H. Bae, Evaluation of the convergence region of an automated registration method for 3D laser scanner point clouds. Sensors **9**(1), 355–375 (2009)

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

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

C. Basdogan, A.C. Oztireli, A new feature-based method for robust and efficient rigid-body registration of overlapping point clouds. The Visual Computer **24**(7-9), 679–688 (2008)

H. Bay, T. Tuytelaars, L. Van Gool, SURF: Speeded Up Robust Features, in *European Conference on Computer Vision*, 2006, pp. 404–417

B. Becerik-Gerber, F. Jazizadeh, G. Kavulya, G. Calis, Assessment of target types and layouts in 3d laser scanning for registration accuracy. Automation in Construction **20**(5), 649–658 (2011)

G.H. Bendels, P. Degener, R. Wahl, M. Körtgen, R. Klein, Image-based registration of 3D-range data using feature surface elements, in *5th International Conference on Virtual Reality, Archaeology and Intelligent Cultural Heritage*, 2004, pp. 115–124

R. Benjemaa, F. Schmitt, A solution for the registration of multiple 3D point sets using unit quaternions, in *European Conference on Computer Vision* 1998, pp. 34–50

R. Bergevin, M. Soucy, H. Gagnon, D. Laurendeau, Towards a general multi-view registration technique. IEEE Transactions on Pattern Analysis and Machine Intelligence **18**(5), 540–547 (1996)

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

C.M. Bishop, *Pattern recognition and machine learning* (Springer, New York, USA, 2006)

J.B. Blair, D.L. Rabine, M.A. Hofton, The laser vegetation imaging sensor: a medium-altitude, digitisation-only, airborne laser altimeter for mapping vegetation and topography. ISPRS Journal of Photogrammetry and Remote Sensing **54**(2–3), 115–122 (1999)

J. Böhm, S. Becker, Automatic marker-free registration of terrestrial laser scans using reflectance features, in *Optical 3D Measurement Techniques VIII*, 2007, pp. 338–344

L. Bornaz, A. Lingua, F. Rinaudo, A new software for the automatic registration of 3d digital models acquired using laser scanner devices, in *International Workshop on Scanning for Cultural Heritage Recording*, 2002, pp. 52–57

D. Borrmann, J. Elseberg, K. Lingemann, A. Nüchter, J. Hertzberg, Globally consistent 3D mapping with scan matching. Robotics and Autonomous Systems **56**(2), 130–142 (2008)

S. Bouaziz, A. Tagliasacchi, M. Pauly, Sparse iterative closest point. Computer Graphics Forum **32**(5), 113–123 (2013)

Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence **23**(11), 1222–1239 (2001)

C. Brenner, C. Dold, N. Ripperda, Coarse orientation of terrestrial laser scans in urban environments. ISPRS Journal of Photogrammetry and Remote Sensing **63**(1), 4–18 (2008)

C. Brenner, C. Dold, Automatic relative orientation of terrestrial laser scans using planar structures and angle constraints. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **36**(Part 3/W52), 84–89 (2007)

A. Censi, An ICP variant using a point-to-line metric, in *IEEE International Conference on Robotics and Automation*, 2008, pp. 19–25

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

C. Dold, C. Brenner, Registration of terrestrial laser scanning data using planar patches and image data. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **36**(Part 5), 25–27 (2006)

M. Doneus, M. Pfennigbauer, N. Studnicka, A. Ullrich, Terrestrial waveform laser scanning for documentation of cultural heritage, in *XXIIth CIPA Symposium Kyoto Japan*, 2009

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

A. Flint, A. Dick, A. van den Hangel, THRIFT: local 3D structure recognition, in *9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications*, 2007, pp. 182–188

G. Flitton, T. Breckon, N. Megherbi Bouallagu, Object recognition using 3D SIFT in complex CT volumes, in *British Machine Vision Conference*, 2010, pp. 1–12

W. Förstner, A feature based correspondence algorithm for image matching. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **26**(3), 150–166 (1986)

M. Franaszek, G.S. Cheok, C. Witzgall, Fast automatic registration of range images from 3d imaging systems using sphere targets. Automation in Construction **18**(3), 265–274 (2009)

R. Gaulton, F.M. Danson, F.A. Ramirez, O. Gunawan, The potential of dual-wavelength laser scanning for estimating vegetation moisture content. Remote Sensing of Environment **132**, 32–39 (2013)

Geomagic, *Product Website*, `http://www.geomagic.com/`. Accessed: 21.09.2015

A.A. Goshtasby, 2-D and 3-D image registration for medical. Remote Sensing, and Industrial Applications, 34–39 (2005)

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

T. Hakala, J. Suomalainen, S. Kaasalainen, Y. Chen, Full waveform hyperspectral lidar for terrestrial laser scanning. Optics Express **20**(7), 7119–7127 (2012)

R. Hänsch, T. Webera, O. Hellwicha, Comparison of 3D interest point detectors and descriptors for point cloud fusion. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences **1**, 57–64 (2014)

C. Harris, M. Stephens, A combined corner and edge detector, in *4th Alvey Vision Conference*, 1988, pp. 147–151

W. He, W. Ma, H. Zha, Automatic registration of range images based on correspondence of complete plane patches, in *5th International Conference on 3-D Digital Imaging and Modeling*, 2005, pp. 470–475

M. Hemmleb, F. Weritz, A. Schiemenz, A. Grote, C. Maierhofer, Multi-spectral data acquisition and processing techniques for damage detection on building surfaces. Image engineering and vision metrology **36**(B5) (2006)

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

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

D.P. Huttenlocher, Fast affine point matching: an output-sensitive method, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1991, pp. 263–268

S. Irani, P. Raghavan, Combinatorial and experimental results for randomized point matching algorithms, in *12th Annual Symposium on Computational Geometry*, 1996, pp. 68–77

A.E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence **21**(5), 433–449 (1999)

B. Jutzi, U. Stilla, Waveform processing of laser pulses for reconstruction of surfaces in urban areas. Measurement Techniques **2**(3.1), 2 (2005)

Z. Kang, J. Li, L. Zhang, Q. Thao, S. Zlatanova, Automatic registration of terrestrial laser scanning point clouds using panoramic reflectance images. Sensors **9**(4), 2621–2646 (2009)

F.R. Kschischang, B.J. Frey, H.-A. Loeliger, Factor graphs and the sum-product algorithm. IEEE Transactions on Information Theory **47**(2), 498–519 (2001)

V. Lempitsky, S. Roth, C. Rother, FusionFlow: discrete-continuous optimization for optical flow estimation, in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8

X.X. Leng, J. Xiao, D.Y. Li, An initial registration method of point clouds based on random sampling. Applied Mechanics and Materials **513–517**, 3680–6383 (2014)

D.D. Lichti, M.G. Licht, Experiences with terrestrial laser scanner modelling and accuracy assessment. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **36**(5), 155–160 (2006)

T. Lindeberg, Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. International Journal of Computer Vision **11**(3), 283–318 (1993)

T.-W.R. Lo, J.P. Siebert, Local feature extraction and matching on range images: 2.5D SIFT. Computer Vision and Image Understanding **113**(12), 1235–1250 (2009)

D. Lowe, Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)

D.G. Lowe, Object recognition from local scale-invariant features, in *7th IEEE International Conference on Computer Vision*, vol. 2, 1999, pp. 1150–1157

F. Lu, E. Milios, Globally consistent range scan alignment for environment mapping. Autonomous Robots **4**(4), 333–349 (1997)

A. Makadia, A. Patterson, K. Daniilidis, Fully automatic registration of 3D point clouds, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 1297–1304

C. Mallet, F. Bretar, Full-waveform topographic lidar: State-of-the-art. ISPRS Journal of Photogrammetry and Remote Sensing **64**(1), 1–16 (2009)

C. Mallet, F. Bretar, U. Soergel, Analysis of full-waveform lidar data for classification of urban areas. Photogrammetrie Fernerkundung GeoInformation (PFG) **5**, 337–349 (2008)

MarketsandMarkets, *3D scanner Market worth $3,705.9 Million by 2020*, `http://www.marketsandmarkets.com/PressReleases/3d-scanner.asp`, 2015. Accessed: 02.10.2015

T. Masuda, N. Yokoya, A robust method for registration and segmentation of multiple range images. Computer Vision and Image Understanding **61**(3), 295–307 (1995)

X. Mateo, X. Binefa, Plane filtering for the registration of urban range laser imagery. Pattern Recognition and Image Analysis **5524**, 136–143 (2009)

N. Mellado, D. Aiger, N.J. Mitra, Super 4PCS fast global pointcloud registration via smart indexing. Computer Graphics Forum **33**(5), 205–215 (2014)

A. Milan, K. Schindler, S. Roth, Detection- and trajectory-level exclusion in multiple object tracking, in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3682–3689

J. Minguez, L. Montesano, F. Lamiraux, Metric-based iterative closest point scan matching for sensor displacement estimation. IEEE Transactions on Robotics **22**(5), 1047–1054 (2006)

M. Mohamad, D. Rappaport, M. Greenspan, Generalized 4-points congruent sets for 3D registration, in *2nd International Conference on 3D Vision*, vol. 1, 2014, pp. 83–90

D. Moldovan, S. Yano, N. Inoue, Automatic registration of multiple range views based on feature matching and feature-depth correspondences, in *IEEE 5th International Conference on Intelligent Computer Communication and Processing*, 2009, pp. 169–172

P.J. Neugebauer, Reconstruction of real-world objects via simultaneous registration and robust combination of multiple range images. International Journal of Shape Modeling **3**(1-2), 71–90 (1997)

D. Novák, K. Schindler, Approximate registration of point clouds with large scale differences. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **2**(5/W2), 211–216 (2013)

F. Pomerleau, F. Colas, R. Siegwart, S. Magnenat, Comparing ICP variants on real-world data sets. Autonomous Robots **34**(3), 133–148 (2013)

H. Pottmann, Q.-X. Huang, Y.-L. Yang, S.-M. Hu, Geometry and convergence analysis of algorithms for registration of 3D shapes. International Journal of Computer Vision **67**(3), 277–296 (2006)

R.C. Prim, Shortest connection networks and some generalizations. Bell System Technical Journal **36**(6), 1389–1401 (1957)

K. Pulli, Multiview registration for large data sets, in *2nd International Conference on 3-D Digital Imaging and Modeling*, 1999, pp. 160–168

T. Rabbani, S. Dijkman, F. van den Heuvel, G. Vosselman, An integrated approach for modelling and global registration of point clouds. ISPRS Journal of Photogrammetry and Remote Sensing **61**(6), 355–370 (2007)

S.E. Reutebuch, H.-E. Andersen, R.J. McGaughey, Light detection and ranging (lidar): An emerging tool for multiple resource inventory. Journal of Forestry **103**(6), 286–292 (2005)

G. Roth, Registering two overlapping range images, in *2nd International Conference on 3D Digital Imaging and Modeling*, 1999, pp. 230–239

S. Rusinkiewicz, M. Levoy, Efficient variants of the ICP algorithm, in *3rd International Conference on 3-D Digital Imaging and Modeling*, 2001, pp. 145–152

R.B. Rusu, S. Cousins, 3D is here: Point Cloud Library (PCL), in *IEEE International Conference on Robotics and Automation*, 2011, pp. 1–4

R.B. Rusu, N. Blodow, M. Beetz, Fast Point Feature Histograms (FPFH) for 3D registration, in *IEEE International Conference on Robotics and Automation*, 2009, pp. 3212–3217

R.B. Rusu, N. Blodow, Z.C. Marton, M. Beetz, Aligning point cloud views using persistent feature histograms, in *International Conference on Intelligent Robots and Systems*, 2008, pp. 3384–3391

J.K. Seo, G.C. Sharp, S.W. Lee, Range data registration using photometric features, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2005, pp. 1140–1145

Y. Shan, B. Matei, H.S. Sawhney, R. Kumar, D. Huber, M. Hebert, Linear model hashing and batch ransac for rapid and accurate object recognition, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. 121–128

J. Shi, C. Tomasi, Good features to track, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600

E. Shtrom, G. Leifman, A. Tal, Saliency detection in large point sets, in *IEEE International Conference on Computer Vision*, 2013, pp. 3591–3598

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

P.W. Theiler, K. Schindler, Automatic registration of terrestrial laser scanner point clouds using natural planar surfaces. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **1**(3), 173–178 (2012)

P.W. Theiler, J.D. Wegner, K. Schindler, Markerless point cloud registration with keypoint-based 4-points congruent sets. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **2**(5/W2), 283–288 (2013)

P.W. Theiler, *PhD Project Website*, `http://www.igp.ethz.ch/photogrammetry/research/tls_registration/`. Accessed: 21.09.2015

P.W. Theiler, J.D. Wegner, K. Schindler, Keypoint-based 4-points congruent sets – automated marker-less registration of laser scans. ISPRS Journal of Photogrammetry and Remote Sensing **96**, 149–163 (2014a)

P.W. Theiler, J.D. Wegner, K. Schindler, Fast registration of laser scans with 4-points congruent sets – what works and what doesn't. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences **2**(3), 149–156 (2014b)

F. Tombari, S. Salti, L. Di Stefano, Performance evaluation of 3D keypoint detectors. International Journal of Computer Vision **102**(1-3), 198–220 (2013)

A. Torii, M. Havlena, T. Pajdla, Omnidirectional image stabilization for visual object recognition. International Journal of Computer Vision **91**(2), 157–174 (2011)

P.H.S. Torr, A. Zisserman, MLESAC: a new robust estimator with application to estimating image geometry. Computer Vision and Image Understanding **78**(1), 138–156 (2000)

G.V. Vosselman, H.-G. Maas, *Airborne and terrestrial laser scanning* (Whittles Publishing, Caithness, Scotland, 2010)

M.W. Walker, L. Shao, R.A. Volz, Estimating 3-d location parameters using dual number quaternions. CVGIP: Image Understanding **54**(3), 358–367 (1991)

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

G. Wei, S. Shalei, Z. Bo, S. Shuo, L. Faquan, C. Xuewu, Multi-wavelength canopy LiDAR for remote sensing of vegetation: Design and system performance. ISPRS Journal of Photogrammetry and Remote Sensing **69**, 1–9 (2012)

J. Williams, M. Bennamoun, Simultaneous registration of multiple corresponding point sets. Computer Vision and Image Understanding **81**(1), 117–142 (2001)

J. Yang, H. Li, Y. Jia, Go-ICP: solving 3D registration efficiently and globally optimally, in *IEEE International Conference on Computer Vision*, 2013, pp. 1457–1464

B. Zeisl, K. Köser, M. Pollefeys, Automatic registration of RGB-D scans via salient directions, in *IEEE International Conference on Computer Vision*, 2013, pp. 2808–2815

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

Y. Zhong, Intrinsic shape signatures: a shape descriptor for 3D object recognition, in *IEEE 12th International Conference on Computer Vision Workshops*, 2009, pp. 689–696