Comparative Evaluation of Point Cloud Registration Models

A report presented to  
the faculty of  
the Russ College of Engineering and Technology of Ohio University

In partial fulfillment  
of the requirements for the degree  
Master of Science

James Daniels

December 2023

© 2023 James Daniels. All Rights Reserved.

This report titled

Comparative Evaluation of Point Cloud Registration Models

by

James daniels

has been approved for  
the School of Electrical Engineering and Computer Science  
and the Russ College of Engineering and Technology by

Jundong Liu

Associate Professor

Patrick Fox

Dean, Russ College of Engineering and Technology

# Abstract

daniels, James, M.S., December 2023, Computer Science

Comparative Evaluation of Point Cloud Registration Models

Director of report: Jundong Liu

Point cloud registration is a fundamental task in preprocessing point clouds, essential for synthesizing cohesive data from multiple point clouds. Yet there is very limited publicly available work assessing the performance of registration methods in a scientifically robust and replicable manner. Existing public datasets, when analyzed in isolation, portray a limited set of environments, hence capturing only a narrow range of use cases.

In light of this, this study establishes a thorough framework for evaluating registration methods across a range of publicly available datasets, encompassing an array of environments and sensor modalities, to enable a fair and rigorous comparison. Our evaluation protocol enables testing of challenging registration problems over a range of initial misalignments and amount of overlap. These parameters are controlled to allow for the direct comparison between the output of registration methods, enabling impartial evaluation of each. A selection of common registration algorithms has also been chosen to analyze their performance against the framework. The operational efficacy of which is explored through a set of robust evaluation metrics, enabling a greater understanding of performance withing diverse contexts and under varying constraints.

The concluding reflections of this study are drawn from the experimental results, which elucidate the relative strengths, weaknesses, and optimal applications of each evaluated algorithm. These conclusions offer insights to guide future research and practical implementations, providing a structured approach to assessing algorithm suitability relative to specific conditions. These findings elucidate the intricate interplay between registration modality and operational requirements, emphasizing the need for extensive testing and evaluation to optimize implementations given the intended operational environment.

# Dedication

*For Caitlyn.*

# Acknowledgements

I wish to express my profound appreciation to Dr. Jundong Liu, who has been an incredible research advisor. His patience and encouragement have been the bedrock supporting this research journey. Special gratitude is extended to my esteemed committee members, Dr. Lonnie Welch and Dr. Chris Bartone. I am also deeply thankful to Dr. Vardges Melkonian, our external committee member, for bringing a refreshing external perspective.

**Table of Contents**

Page

[Abstract 3](#_Toc148360339)

[Dedication 4](#_Toc148360340)

[Acknowledgements 5](#_Toc148360341)

[List of Tables 8](#_Toc148360342)

[List of Figures 9](#_Toc148360343)

[List of Acronyms 10](#_Toc148360344)

[Chapter 1: Introduction 11](#_Toc148360345)

[1.1 Area Overview 11](#_Toc148360346)

[1.2 The Registration Problem 12](#_Toc148360347)

[1.3 Contributions and Report Overview 16](#_Toc148360348)

[Chapter 2: Background 18](#_Toc148360349)

[2.1 Applications 18](#_Toc148360350)

[2.2 Data for 3D Registration: Point Clouds 19](#_Toc148360351)

[2.3 Properties of Point Clouds 20](#_Toc148360352)

[2.4 Challenges of the Registration Problem 22](#_Toc148360353)

[Chapter 3: Methods 25](#_Toc148360354)

[3.1 ICP Variants 27](#_Toc148360355)

[3.1.1 Point-to-Point ICP: 28](#_Toc148360356)

[3.1.2 Point-to-Plane ICP: 28](#_Toc148360357)

[3.1.3 Multi-Scale ICP 29](#_Toc148360358)

[3.2 Feature Based 30](#_Toc148360359)

[3.2.1 RANSAC 31](#_Toc148360360)

[3.2.2 FGR 32](#_Toc148360361)

[3.3 Learning Based 32](#_Toc148360362)

[3.3.1 Deep Global Registration 33](#_Toc148360363)

[3.3.2 FMR 35](#_Toc148360364)

[Chapter 4: Methodology 38](#_Toc148360365)

[4.1 Main Objective 39](#_Toc148360366)

[4.2 Datasets 39](#_Toc148360367)

[4.3 Experimental Design 43](#_Toc148360368)

[4.4 Implementation Details 45](#_Toc148360369)

[Chapter 5: Experimental Results 47](#_Toc148360370)

[5.1 Evaluation Metrics 47](#_Toc148360371)

[5.2 Experimental Results and Analysis 49](#_Toc148360372)

[5.3 Observations 59](#_Toc148360373)

[5.4 Limitations 61](#_Toc148360374)

[Chapter 6: Conclusions and Future Work 64](#_Toc148360375)

[References 66](#_Toc148360376)

# List of Tables

Page

[**Table 1** *Registration Methods* 24](#_Toc147745681)

[**Table 2** *Overview of Datasets* 40](#_Toc147745682)

[**Table 3** *Average Combined Error Metric* 50](#_Toc147745683)

[**Table 4** *Average Success Rate* 51](#_Toc147745684)

[**Table 5** *Average Relative Translation Error* 52](#_Toc147745685)

[**Table 6** *Average Relative Rotation Error* 53](#_Toc147745686)

[**Table 7** *Average Runtime* 54](#_Toc147745687)

[**Table 8** *Correlation: Initial Misalignment with Combined Error Metric* 56](#_Toc147745688)

[**Table 9** *Correlation: Overlap with Combined Error Metric* 57](#_Toc147745689)

# List of Figures

Page

[**Figure 1** *DGR Architecture* 33](#_Toc147745690)

[**Figure 2** *Feature Metric Registration Framework* 35](#_Toc147745691)

[**Figure 3** *Example Registration Problem* 49](#_Toc147745692)

# List of Acronyms

3D – 3-Dimensional

CAD – Computer-Aided Design

CSRB – Cross Source Registration Benchmark

DGR – Deep Global Registration

FGR – Fast Global Registration

FMR – Feature Metric Registration

ICP – Iterative Closest Point

LiDAR – Light Detection and Ranging

PCR – Point Cloud Registration

RANSAC – Random Sample Consensus

SFM – Structure from Motion

SLAM – Simultaneous Location and Mapping

SONAR – Sound Navigation and Ranging

VR – Virtual Reality

# Chapter 1: Introduction

Point cloud registration is a crucial and challenging task that involves integrating multiple point clouds, all representing the same scene or object, into a unified coordinate system. The challenge stems from the need to ensure reliable, repeatable, and accurate alignment of the point clouds in all conditions and environments. The declining cost of point cloud-capturing devices is paving the way for an influx of affordable consumer products. These include high-end mobile phones equipped with LiDAR sensors and the increasing adoption of technologies such as VR headsets with integrated depth-sensing capabilities. Nevertheless, it's worth highlighting that this study concentrates on 3D point cloud data rather than standard imagery since the discussed algorithms are specifically designed for three-dimensional point sets instead of two-dimensional images.

## 1.1 Area Overview

A point cloud is a data structure that consists of a set of data points that represent the shape or surface of an object in three-dimensional space. Each point in the cloud is characterized by its x, y, and z coordinates in 3D space. The coordinates are derived from various sensor types, such as laser scanners, depth cameras, or photogrammetry techniques. With each method of acquisition, coordinates in a single point cloud are generated in a local coordinate system. The point cloud registration problem essentially deals with finding the transformation that harmonizes the local coordinate systems generated into a single global frame. The registration of point clouds is critical to generating a more comprehensive 3-Dimensional view of an object or scene.

Point cloud registration involves combining multiple point clouds of the same scene into a single coordinate system. This process becomes necessary due to the challenges in capturing complete geometric data of a space in a single scan, resulting from factors like the sensor's limited range, its field of view, and the quality of a single point cloud. Therefore, to form a comprehensive representation of an environment, 3D scans from multiple viewpoints are usually amalgamated, a process in which registration plays a pivotal role.

The main objective of registration is to determine the transformation mapping one cloud to another, establishing a shared frame of reference. The process generally involves identifying corresponding points in both point clouds and estimating the transformation that aligns this subset of points. The outcome of this process is a transformation that can map one point cloud to another's coordinate system.

## 1.2 The Registration Problem

Understanding the registration problem involves appreciating the broad spectrum of techniques and approaches it encompasses. The types of registration, the kinds of transformations involved, and the distinct metrics that determine successful matching are all integral facets of the multidimensional registration problem. At its core, point cloud registration is a pursuit to find the optimal transformation that aligns one point cloud (source) to another (target), thereby minimizing a predefined error metric.

Registration can be categorized in multiple ways. One such classification distinguishes between pairwise or multiwise registration. The former aligns only two point clouds while the latter computes a sequence of pairwise registrations. Multiwise registration can fall prone to registration errors that build with subsequent pairwise registration, thus there are methods to globally refine the alignment (Torsello et al., 2011). Starting from a sequence of scans, pairwise registration is often the first step to scene reconstruction. Given that it is the first step, the accuracy of this process is paramount to the accuracy of the final reconstruction. For the purposes of this report, we will define a registration problem as a pairwise alignment between two input point clouds.

Furthermore, registration can also be differentiated into coarse and fine registration. Coarse or global registration aims to find an initial approximation of the transformation between point clouds without any prior information about the initial pose. Fine or local registration is when there is an initial guess of the transformation to be improved. Fine registration refines the initial alignment generated by coarse registration to achieve the best possible match between the point clouds. In many scenarios it may be necessary to conduct a combination of these methods. In this report we will be assessing performance of registration algorithms in the global context, although the evaluation protocol could be adapted.

There is a spectrum of transformations possible that are all represented by a 4x4 transformation matrix, providing a concise and efficient representation for transforming points in 3D space. This representation conveniently allows for multiple transformations (rotation, translation, and scaling) to be concatenated into a single matrix. Whilst the inverse of this matrix can be used to efficiently calculate the inverse transformation.

Transformations in the 4x4 matrix can be divided into a hierarchy based on their preservation characteristics. Each transformation type in this hierarchy includes those above it:

* Rigid (Euclidean) transformations: Incorporate rotation and translation, while preserving the shape and size of objects.
* Similarity transformations: Consist of rotation, translation, and uniform scaling. They preserve shapes (due to angle conservation) but alter sizes.
* Affine transformations: Extend similarity transformations with non-uniform scaling (different scaling factors in different directions) and shearing. They preserve parallelism of lines but not necessarily the angles or lengths.
* Projective transformations: The most general form of transformation, preserving only the straightness of lines (collinearity). They can manipulate the perspective of a scene, but since they don’t preserve parallelism, length or angles they can introduce significant distortions.

In the context of point cloud registration, different methods estimate different types of transformation estimates. For instance, ICP estimates only rigid transformations, while feature based and learning methods can estimate more complicated transformations. The choice of algorithm is therefore contingent on the specific characteristics of the point clouds to be registered. For this study, the registration problems generated from the datasets do not require scaling.

The correspondence problem, a major challenge in point cloud registration, involves the identification of point pairs from each point cloud that correspond to the same location on the underlying object or scene. A typical registration solution involves identifying corresponding points and determining the transformation that minimizes the alignment error between corresponding points. This procedure is iterative, since the identification of corresponding points is affected by the position and orientation of the data sets. Numerical optimization methods, such as gradient descent, are then used to minimize an error metric that quantifies the misalignment between the corresponding points. When the alignment errors decrease below a predetermined threshold, the registration process is complete. The effectiveness of the registration can then be evaluated using error metrics that compare the estimated transformation to the ground truth. Understanding these components and how they relate to each other is crucial to developing effective registration algorithms and systems.

To this end, several techniques have been developed to address the challenge of point cloud registration. These methods can generally be divided based on their approach. These include iterative closest point (ICP) algorithms, feature-based methods, and learning-based methods. Each of these techniques has its pros and cons, and the selection of an appropriate method is largely dependent on the specific requirements of the application in use. Ongoing research in this domain seeks to enhance the accuracy, speed, and robustness of point cloud registration algorithms, thereby enabling their widespread utilization in real-world applications.

## 1.3 Contributions and Report Overview

This report delves into the world of point cloud registration, a crucial process for merging point clouds to obtain more complete usable data. This task, however, is not without challenges – factors such as noise, the absence of data, and the substantial size of point clouds add complexity. In our exploration, we will not only delve into these challenges but also evaluate the efficacy of existing methods, culminating our discussion with experimentation and analysis on the selected datasets.

Our study's primary objective is to assess the performance of point cloud registration algorithms on datasets characterized by varying attributes. With point cloud registration serving as a critical phase in applications involving 3D reconstruction, our findings offer valuable insights. These insights encapsulate the strengths and weaknesses of the examined algorithms in different contexts, thus serving as a guide for practitioners in selecting suitable algorithms for their specific applications. Ultimately, this study contributes to the field of point cloud registration by establishing an unbiased framework that can be used to benchmark different methods against one another, fostering progress for ongoing research.

The remainder of this report is organized as follows: we first discuss data properties and elucidate the challenges to effective registration, providing background in Chapter 2. Following this, Chapter 3 delves into the existing methods, selection decision making and provides a comprehensive literature review that sheds light on different design approaches providing perspective on existing paradigms. Chapter 4 is dedicated to the research design and experiment methodology. Chapter 5 presents our results and in-depth analysis, limitations of the study and draws thoughtful observations. Finally, Chapter 6 concludes the report, reflects on our contributions, and provides a possible roadmap for future work.

Top of Form

Bottom of Form

# Chapter 2: Background

## 2.1 Applications

In comparison to 2D images, point clouds provide a more granular and intricate representation of real-world geometry and surface information. This makes them indispensable in applications necessitating high levels of detail. As a result, point clouds are increasingly being used across a variety of fields, where precise registration is integral. These applications require registration to convert point clouds into a unified coordinate system that covers the entire scene completely. Registration lays the groundwork for these applications enabling the effective implementation of subsequent functionalities and techniques.

For instance, autonomous vehicles leverage point cloud registration for precise environmental perception, driving informed decision-making during navigation (Li & Ibanez-Guzman, 2020). Through the amalgamation of data from multiple sensors, vehicles can create a robust 3D model of their surroundings, ensuring safe route planning and execution. Concurrently, 3D reconstruction techniques utilize registration to assemble comprehensive representations of objects or environments by amalgamating data from differing perspectives (Wang et al., 2020), with applications spanning architecture, urban planning, and archaeology.

In object recognition, point cloud registration aids in comparing an unknown object's point cloud with a known database (Shi et al., 2020), an approach proving particularly beneficial in robotics for tasks such as object handling, manipulation, and sorting. Similarly, VR applications harness point cloud registration to fabricate immersive experiences by generating realistic 3D virtual environments based on real-world data (Feng et al., 2023).

Cultural heritage preservation employs point cloud registration to digitize and conserve geometric data related to historical sites and artifacts (Greenberg, 2020; Apollonio et al., 2014), enabling non-invasive studies and monitoring. Meanwhile, in the manufacturing sector, point cloud registration contributes to quality control via dimensional analysis (Mora et al., 2021; Wang and Kim, 2019), ensuring adherence to specifications and prompt defect identification.

Further applications include remote sensing and environmental monitoring, where point cloud registration facilitates the analysis of large-scale topographic data, tracks vegetation growth, and assesses the impacts of natural disasters (Stilla & Xu, 2023, Shao et al., 2020). Other uses include reverse engineering and in the games industry (Takimoto et al., 2016). In robotics applications, localization systems match existing 3D maps to locally obtained point clouds. As the availability and accuracy of point cloud data continue to improve, the applications of point cloud registration are anticipated to extend into new areas and use cases.

## 2.2 Data for 3D Registration: Point Clouds

A point cloud is a set of geometrically represented 3D points, . They can be captured in multiple ways, with each method offering unique advantages. Photogrammetric techniques, including the use of stereo depth cameras or structure from motion (SFM), offer the most cost-effective methods. These methods involve identifying corresponding feature points between images and determining the depth value based on the correspondence and geometric positioning of each camera sensor. Alternatively, LiDAR (Light Detection and Ranging) systems measure the time it takes for light to bounce back from an object to calculate its distance from the sensor. While LiDAR is highly beneficial for real-time systems due to its high frequency and accuracy, it comes at a relatively high cost. Other data sources include structured light, Sonar, and computer-aided design (CAD).

Depending on the acquisition method, additional parameters may also be obtained, providing additional information about the scanned area. For instance, LiDAR-generated points include intensity values that indicate the material properties based on the object's surface reflectivity. On the other hand, point clouds captured using photogrammetric techniques include RGB color values that represent the object's appearance. These additional properties can be integrated into the registration process, specifically when generating point feature descriptors. The choice of acquisition method and the attributes to be captured in the point cloud depend on the intended application and the desired level of accuracy and detail.

## 2.3 Properties of Point Clouds

Depending on the method of acquisition, point clouds exhibit varied characteristics such as point density, noise, statistical outliers, non-uniformity, and scaling of the coordinate system. An understanding of these properties is essential to the development of registration techniques, as it informs us of the data characteristics that registration methods need to accommodate. By considering these properties, researchers can create robust algorithms capable of handling diverse point cloud data and ensuring accurate registration across a wide range of applications.

Point density refers to the number of points per unit area and indicates the captured detail level within a point cloud. Higher-density point clouds provide more detailed and accurate object representations but simultaneously increase registration complexity due to an increased number of points (Zhang et al., 2021). Conversely, sparse point clouds simplify the registration process but may compromise accuracy.

Noise is another critical property to consider. Point clouds inherently contain noise and random variations in point positions, which is the result of measurement errors, calibration errors, or environmental factors during data acquisition (Araújo & Oliveira, 2020). Noise can negatively affect registration accuracy, as it may cause erroneous correspondences between points. Statistical outliers, points that significantly differ from the rest of the data, can have a similarly detrimental effect. Effective registration algorithms must be robust to noise and minimize its influence on the registration process.

The scale represents the relative size of objects within point clouds or the utilized coordinate system. Differences in scale can occur when capturing data from varying distances or when using different sensors. While some sensors, like LiDAR, intrinsically output data in real-world units like meters, other methods do not (Elhashash et al., 2022). Depending on the task, registration algorithms may need to handle point clouds with different scales and align them while accounting for these discrepancies.

Lastly, the data source significantly impacts the registration process as point clouds originate from different sensors, each with its unique characteristics like noise levels, density, and accuracy. Factors like noise and statistical outliers are typically more prevalent in low-cost sensors, making the challenge of registration significantly more difficult. Creating versatile registration algorithms capable of handling data from multiple sources is essential to enable the interoperability of sensors.

## 2.4 Challenges of the Registration Problem

The process of point cloud registration faces several hurdles that need to be surmounted to ensure reliable precise and efficient alignment. These hurdles include managing very large point clouds, identifying features, dealing with dynamic scenes, and aligning data from multiple sources. Addressing these challenges is vital for the development of effective point cloud registration algorithms. Ongoing research in this domain seeks to boost the accuracy, speed, and resilience of registration techniques, facilitating their broader use in practical applications. The two factors that most greatly impact the difficulty of a registration problem are the magnitude of the initial misalignment and the degree of overlap.

The magnitude of the initial misalignment gives the displacement the of the source point cloud from its ground truth pose. This contains the rotation and translation components that registration is trying to solve. Intuitively, the larger this initial offset the harder the registration problem is. An algorithm may behave differently due to the magnitude of the initial misalignment; thus testing should be conducted on a range of initial transformations.

Overlap is a measure of how much of a scene can be observed in both point clouds. When point clouds contain only a minor overlapping region, this poses a substantial challenge to registration. Since the point clouds that need to be aligned don’t cover the same area, some parts of the scene may be present in one but not the other. With only a limited area of overlap, it is more of a challenge to establish enough corresponding points or features (Zhang et al., 2021). This can arise when capturing point clouds from differing viewpoints, with not all objects or scene areas visible from each location. An effective registration algorithm should manage constrained overlapping areas, continuing to align the coordinate systems accurately. An algorithm may behave differently due to overlap; thus testing should be conducted on a range of overlaps.

Efficiently processing large point clouds is a pivotal challenge. Although high-resolution datasets are desired for their detail, they considerably slow down processing due to the irregular distribution of points and increased complexity (Dong et al., 2020). Given that point clouds can comprise millions of data points, the requirement for registration algorithms is to process these efficiently, averting computational bottlenecks and memory constraints. Developing algorithms that can align large datasets within a reasonable time frame is crucial.

Local features like corners and edges are key to point cloud registration, offering distinct characteristics to establish point correspondences in varying point clouds. Yet, identifying and matching these features can prove difficult, especially when features are sparse or point clouds are filled with noise (Yang et al., 2015).

Dynamic scenes, incorporating moving objects or environmental changes, present another hurdle for point cloud registration. As the scene changes, so too do the points and features available for alignment. Registration algorithms should demonstrate robustness to these scene dynamics and continue to align based on the static parts of the data (Zhou et al., 2017).

Lastly, cross-source registration involves the alignment of point clouds that have been acquired from different types of sensors, which may have different characteristics and produce point clouds with different properties. The integration of data from multiple sources introduces substantial complexity to the registration task, complicating the process of finding corresponding points (Huang et al., 2019). For example, in robotics applications registration is conducted between high resolution on-board data to sparse 3D maps which represent large areas (Fontana et al., 2021). Effective algorithms should manage these cross-source data discrepancies.

# Chapter 3: Methods

Over time, an array of strategies and algorithms have been introduced to achieve point cloud alignment. This literature review offers an overview of notable point cloud registration methods, encompassing direct optimization-based techniques such as Iterative Closest Point (ICP), feature-based strategies like Random Sample Consensus (RANSAC) and Fast Global Registration (FGR), and emerging learning-based approaches, including Deep Global Registration (DGR) and Feature Metric Registration (FMR). The table below provides an overview of the methods explored in this report:

**Table 1***Registration Methods*

|  |  |  |  |
| --- | --- | --- | --- |
| Method Name | Method Type | Date of Initial Implementation | Authors |
| Point-to-Point | ICP-Based | 1992 | Besl and Mckay |
| Point-to-Plane | ICP-Based | 1992 | Chen and Medioni |
| RANSAC | Feature-Based | 1981 | Fishler and Bolles |
| FGR | Feature-Based | 2016 | Zhou, Park and Koltun |
| DGR | Learning-Based | 2020 | Choy, Dong and Koltun |
| FMR | Learning-Based | 2020 | Huang, Mei and Zhang |

In this study, we delve into the fundamental workings of each method, examining their strengths and limitations. Our objective is to gain a deeper understanding of the underlying principles behind these methods and how they operate in the context of point cloud registration. By comprehensively exploring their inner workings, we aim to provide valuable insights into the current state-of-the-art techniques and identify potential areas for future research. Furthermore, we emphasize the importance of rigorously evaluating these algorithms on diverse datasets to determine their suitability for specific applications.

The algorithms chosen for comparison in this study were selected with great deliberation. They represent not only the evolution of point cloud registration techniques over the past few decades but also the diversity in approach—from optimization to feature-based, and eventually to learning-based methods. The intention behind such a comparison is twofold. Firstly, it offers an opportunity to witness how the paradigm has shifted and how newer methods might offer advantages or drawbacks relative to their predecessors. Secondly, by understanding the nuances of each method, we are better equipped to recommend a suitable algorithm for specific scenarios, thus paving the way for efficient and accurate point cloud registrations.

The overarching goal of this comparison is to establish a framework for method selection based on the nature of the data and the intended application. For instance, traditional methods like ICP might be more suitable in applications or environments where a rough initial alignment is provided, while feature-based or learning-based methods might be beneficial when the data contains recognizable patterns or when there's a need for generalized solutions across multiple datasets. Application scenarios for these models vary widely. ICP variants, given their deterministic nature, are often used in situations where consistency is crucial. On the other hand, feature-based techniques like RANSAC are versatile and can handle a significant amount of outlier data, making them apt for outdoor scenes or environments with a lot of extraneous data. The learning-based methods, such as DGR and FMR, with their ability to leverage vast amounts of training data, have shown promise in dynamic scenarios, like robotics or autonomous driving, where the environment can change rapidly and unpredictably. The selection of algorithms in this review emphasizes the importance of method suitability relative to the application at hand.

## 3.1 ICP Variants

Iterative Closest Point (ICP) algorithms and their variants have gained traction for their straightforwardness and effectiveness. In this paper, they are perceived as techniques that exclusively use the individual points of a point cloud without any computed features. Optimization-based registration methods focus on minimizing an objective function that quantifies the misalignment between point clouds, thereby achieving alignment. Both of these algorithms iterate over the following two step process: First a correspondence set of points are identified in the target point cloud and the source point cloud using the current transformation estimate. Second the transformation estimate is updated by minimizing the objective function over the correspondence set. Two primary categories can be identified within these methods: point-to-point and point-to-plane strategies, both utilizing iterative optimization techniques until a convergence criterion is achieved.

### 3.1.1 Point-to-Point ICP:

The point-to-point algorithm is a paramount classical registration method (Besl & McKay, 1992). This method progressively refines the estimated transformation by identifying correspondences between proximate points in the two point clouds and minimizing the sum of squared Euclidean distances between corresponding points. The objective function can be defined as,

Where, k is the correspondence set, p are points from the target point cloud P, q are points from the source point cloud Q, and the current transformation matrix is T.

ICP starts with an initial guess of the transformation and then converges to a local minimum, with the quality of the final alignment being sensitive to the initial guess of the transformation. Several ICP variants have been proposed to improve its convergence rate, robustness, and handling of noise and outliers (Rusinkiewicz & Levoy, 2001). One such example includes randomizing the initial transformation to be optimized, running multiple initial guesses concurrently and selecting the final result with the lowest misalignment. Two other ICP variants will be explored in this study.

### 3.1.2 Point-to-Plane ICP:

Point-to-plane methods augment the basic ICP approach by including surface normal information (Chen & Medioni, 1992). The objective function considers both the position and orientation of local surface patches minimizing the point-to-plane distance between corresponding points. The objective function can be defined as,

Where is the normal of point p.

This modification often leads to faster convergence and improved registration accuracy compared to point-to-point methods, particularly for point clouds with complex geometries and limited overlap (Rusinkiewicz & Levoy, 2001).

Although optimization-based registration methods are popular for their simplicity, implementation ease, and capability to handle partial data, they are often slow to converge due to their iterative techniques, especially when processing large point clouds. Moreover, these methods are prone to converge on local minima, especially when the initial misalignment is substantial. Multiple strategies have been proposed to overcome these limitations, such as using multiple initial guesses of the misalignment, although this in turn increases computation time.

### 3.1.3 Multi-Scale ICP

This involves multiple iterations of the ICP algorithm over various resolutions of the point cloud. The algorithm starts from a coarse resolution and progressively refines the registration as it shifts to finer scales. The scales are defined by the voxel sizes and maximum iteration parameters. The point clouds are down sampled using the specified voxel size at each scale, and ICP is applied with a maximum number of iterations. The resulting transformation is then used as the initial guess for the next scale. This method can help to avoid local minima and speed up the convergence of ICP.

## 3.2 Feature Based

Feature-based registration methods rely on identifying distinctive geometric features, such as edges, corners, and surface curvature, to establish correspondence. This process generally comprises three main stages: feature extraction, feature matching, and transformation estimation.

Feature extraction is the process of finding these features and computing descriptors for each point, this step is common to both of the methods examined in this paper. Once the features are extracted, a matching algorithm can be used to find corresponding features in the other point cloud. The transformation parameters are then estimated using the matched features.

In feature extraction, features that capture the geometric properties of the local neighborhood surrounding a point are identified and extracted from each point cloud. These features are typically invariant to scale, rotation, and translation, enabling robust matching across different point clouds. Noteworthy feature extraction methods include Fast Point Feature Histograms (FPFH) (Rusu et al., 2009), Spin Images (Johnson & Hebert, 1999), and SHOT (Tombari et al., 2010). FPFH is a method for computing a local geometric descriptor for each point in the point cloud, which characterizes the local geometric properties of the point's neighborhood.

Following feature extraction, a matching algorithm is employed to identify corresponding features in the other point cloud. This process often involves a nearest-neighbor search in the feature descriptor space, using distance metrics like cosine similarity or Euclidean distance to quantify feature descriptor similarity. With correspondences established, the transformation aligning the point clouds can then be estimated. This study explores two methods for estimating this transformation matrix based on the generated feature correspondences: RANSAC and FGR.

### 3.2.1 RANSAC

RANSAC is a renowned feature-based registration method used to align point clouds by identifying inlier correspondences from a set of potential matches (Fischler & Bolles, 1981). Originally used in image processing, the fitting method can be applied to point cloud registration. RANSAC iteratively selects a random subset of feature correspondences, computes a transformation using these correspondences, and then evaluates the transformation by counting the number of inliers, i.e. the number of corresponding features. In each iteration, a random subset of correspondences is chosen to compute a transformation matrix. This matrix is subsequently used to transform one of the point clouds, and inlier correspondences are recalculated using the transformed point cloud. This process is repeated a predefined number of times, and the transformation that maximizes the number of inlier correspondences is selected as the final transformation. This method can be highly effective at filtering out outliers or spurious correspondences, but this depends on the randomly sampled set selected being free from outliers.

### 3.2.2 FGR

Fast Global Registration (FGR), on the other hand, solves a pose graph optimization problem to align point clouds (Zhou et al., 2016). FGR introduces a correspondence selection process that selects matches with the lowest pairwise distances, intending to reduce the number of outliers and enhance registration efficiency. The selected correspondences are used to construct a pose graph, where each node represents a point cloud, and edges represent the spatial relationships between point clouds, defined by the feature matches. The transformation that aligns the point clouds is then estimated by solving a pose graph optimization problem, which minimizes the discrepancies between corresponding feature positions. This approach is intended to be faster than RANSAC, as there is no model proposal and evaluation needed in each iteration.

## 3.3 Learning Based

The advent of learning-based methods has sparked significant interest due to their ability to exploit neural networks in discerning underlying structures and relationships within point cloud data. By leveraging extensive training datasets, these methods can recognize patterns and inherent features, allowing for precise alignment of point clouds even when confronting complex or diverse structures.

However, some caveats that need to be considered. Primarily, learning-based approaches necessitate a substantial amount of training data to establish the mapping between point clouds. This process can be both resource-intensive and time-consuming, especially when grappling with large datasets. Additionally, the performance of learning-based techniques relies heavily on the salient features extracted from the training data. Furthermore, these approaches may exhibit reduced performance when encountering unseen data or objects, emphasizing the necessity for comprehensive training on a diverse range of datasets.

Despite these challenges, learning-based methods have shown great promise in recent years, achieving admirable performance when compared to traditional methods. While these methods are still in the early stages of development, they show great promise for a wide range of applications especially when they are trained to perform a particular task. We can expect to see the development of new neural network architectures that further improve the performance and generality of learning-based registration methods. While there are several types of learning-based registration methods, we will investigate two in detail: DGR and FMR.

### 3.3.1 Deep Global Registration

Deep Global Registration (DGR) is a deep learning-based point cloud registration method that uses a 6-dimensional convolutional neural network (CNN) with a U-net structure to learn and predict the probability of point correspondence (Choy, Dong, and Kolton, 2020). The network architecture can be seen in Figure 1. This method is structured around three primary components: correspondence confidence estimation, global registration, and local refinement.

**Figure 1***DGR Architecture*

A picture containing design

Description automatically generated

The registration pipeline of DGR begins with the extraction of distinctive features through the use of Fully Convolutional Geometric Features (FCGF), though it remains flexible for adaptation to other feature descriptors. To estimate the confidence in correspondences, DGR performs a nearest neighbor search in the feature space, generating a potential correspondence set. These 3D correspondences construct a geometric structure in a 6-dimensional space (Christopher Choy et. al, 2020), which is subsequently processed by the CNN to yield the likelihood of correct correspondence. Notably, inlier correspondences manifest on a lower-dimensional surface in the 6D space. Subsequently, the convolutional network predicts the likelihood of each correspondence, assigning it a weight. A Weighted Procrustes method is then used to determine the transformation that minimizes the error between corresponding points.

During the training of the model, a binary cross-entropy loss is used between the probability of a correspondence being correct and a set of "ground truth" correspondences. This set is leveraged to optimize the network, wherein the "ground truths" are defined by the correspondences that can be accurately aligned by the ground truth transformation.

In the final stage, DGR applies a novel Weighted Procrustes algorithm to minimize the weighted mean squared error between correspondences, using the likelihood of a correspondence being correct as the differentiable gradient. As this method focuses on the weights of correspondences instead of the coordinates, the complexity is linearly related to the number of correspondences. Consequently, the pipeline can accommodate a dense correspondence set, thus enhancing the overall registration accuracy. In effect, this method offers a closed-form solution to the rigid registration problem. The model yields the rotation and translation that minimizes the squared error given the weights as its output.

### 3.3.2 FMR

Feature Metric Registration (FMR) is a novel point cloud registration method that fuses mathematical aspects of classical algorithms with cutting-edge learning techniques. Rather than establishing point correspondences, FMR is designed to estimate transformations by minimizing a feature-metric projection error rather than a geometric projection error (Yang et al., 2020). The key idea behind FMR is that the feature difference of point clouds should decrease as alignment improves. The FMR framework consists of an Encoder module, an encoder-decoder branch, and a registration branch.

**Figure 2***Feature Metric Registration Framework*

A picture containing text, screenshot, diagram, design

Description automatically generated

The Encoder module is tasked with the extraction of distinctive, rotation-attentive features from point clouds to facilitate precise transformation estimation. Meanwhile, the encoder-decoder branch utilizes an unsupervised learning approach to train the Encoder module, wherein a decoder module recreates point clouds from the extracted features. This branch is specifically designed to make the Encoder module aware of rotation differences. In that two copies of the same point cloud have the same features but if a transformation is applied to one then the features will be different. The decoder block is comprised of four fully connected layers, employing the LeakyReLU activation function.

FMR establishes feature-metric correspondences between two point clouds, utilizing a learned feature-metric descriptor. The descriptor is learned in a semi-supervised manner, where a small set of correspondences between the point clouds are manually labeled as either inlier or outlier correspondences.

During the correspondence search stage, FMR establishes correspondences by computing the pairwise distance matrix between the learned feature-metric descriptors. To achieve this, FMR introduces a differentiable feature-metric loss function that evaluates the distance between corresponding points in the feature space. A bipartite graph matching algorithm is then applied to identify optimal correspondences between the point clouds. The authors employ a fast optimization algorithm, which significantly reduces the computational cost of registration compared to traditional methods.

Lastly, the registration branch iteratively estimates transformation parameters using an inverse compositional algorithm to directly minimize the projection error of these learned features. The optimization process minimizes the feature-metric loss function with respect to transformation parameters (rotation and translation) to align the source and target point clouds. The training of the registration branch uses a semi-supervised approach, enabling the network to learn distinctive features crucial for registration.

In essence, FMR provides a swift and robust semi-supervised approach to point cloud registration, employing learned feature-metric descriptors to establish correspondences between point clouds and accurately align them. By harnessing the power of deep learning for feature extraction and incorporating a differentiable loss function for correspondence, FMR ensures efficient and accurate registration results. The underlying philosophy of FMR is the unification of feature-based and metric-based registration approaches, drawing on the strengths of both techniques to mitigate their respective limitations.

# Chapter 4: Methodology

In this chapter, the methodology utilized in this study is discussed, delineating the approach used to assess the efficacy of different registration methods. It also covers the nuanced parameters and variations used within each registration technique, and how these elements coalesce to determine the methods’ overall performance. Section 4.1 explicitly defines the principal objective, emphasizing the transformation matrix's comprehensive representation of transformations used for aligning source frames with target coordinate systems and the role it plays in the success of the registration workflow. Section 4.2 introduces the datasets employed in this study, diverse in their origins and characteristics, they are integral in gauging the versatility and robustness of registration techniques across an array of applications and scenarios.

Section 4.3 details the experimental design of this research, offering insights into the steps involved in testing the performance of different registration methods and the significance of reproducibility. The choice of parameters, processing procedure, and initial perturbation in experimenting with registration methodologies are analyzed, elucidating how the amalgamation of these elements serves to comprehensively assess the registration techniques under investigation. Finally, section 4.4 provides implementation details of the utilized tools and models, clarifying the selection, adaptation, and usage of these methods in alignment with the study’s objectives.

## 4.1 Main Objective

In the context of computer science, the principal aim of registration is to compute the 4x4 transformation matrix that encapsulates the result of registration. This matrix encompasses the necessary rotation, and translation to align the source frame with the target coordinate system. This matrix representation also facilitates the concatenation of transformations via straightforward matrix multiplication. The process of estimating this matrix and contrasting it with the initial offset (ground-truth) plays a crucial role in the registration workflow. Ultimately, the success of registration depends on the accuracy and precision of the 4x4 matrix representation.

The 4x4 matrix representation is defined as:

The 3x3 upper left submatrix represents the rotation matrix, while the top rightmost 3x1 submatrix gives the translation vector (). The 4x4 transformation matrix, referred to as the homographic matrix, serves as the estimated variable. It can represent different types of transformations, including the ones we discussed before: rigid, similarity, affine, and projective transformations.

## 4.2 Datasets

In our examination, we employ three publicly accessible datasets for the performance assessment of point cloud registration techniques: the Laser Registration dataset from ETH Zurich's Autonomous Systems Lab (ASL), the Cross Source Registration Benchmark (CSRB) and Princeton's SUN3D Dataset. For a comprehensive evaluation, it's imperative to test a broad spectrum of datasets against various registration methodologies to determine the most suitable technique for specific applications. This selection covers quite a wide set of examples and use cases, including between clouds coming from different sensors.

For the purposes of this study, the datasets have also been split into training and testing sets by selecting subsets of the data for these purposes. This has been done with both ETH and SUN3D datasets only, since the cross-source dataset does not contain enough data to adequately train the model. To do this 6 out of 8 scenes from the ETH dataset are used for training and 5 out of 7 scenes from the SUN3D dataset, furthermore the training data is split into 80 percent for training and 20 percent for validation. This is critical to ensure we are not testing the models on previously seen data whilst we are still able to test all methods on a smaller number of testing sets. The testing program can easily be modified to enable testing across the entire dataset or to add new datasets in due course.

**Table 2***Overview of Datasets*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | Number of Registration Problem Pairs | Average Number of Points | Average Bounding Box Volume (m^3) | Average Point Density (points/m^3) |
| ETH-Apartment | 498 | 370,550 | 212.38 | 1756.37 |
| ETH-Gazebo | 314 | 136,648 | 22,816.15 | 6.43 |
| SUN3D-Hotel | 32 | 335,397 | 12.18 | 32,930.47 |
| SUN3D-Lab | 58 | 391,465 | 14.41 | 30,388.49 |
| Cross Source | 182 | Lidar: 3302  Kinect: 300,145  Sfm: 45,218 | Lidar: 51.15  Kinect: 54.33  Sfm: 0.70 | Lidar: 66.39  Kinect: 6403.71  Sfm: 93,618.63 |

The laser registration dataset from ETH Zurich’s’ Autonomous Systems Lab (ASL) is a collection of point clouds acquired using a Velodyne HDL-64E LIDAR sensor mounted on a mobile robotic platform. The sensor has high-resolution, capturing point clouds in a 360-degree horizontal field and a 26.8-degree vertical field of view, capturing up to 1.3 million points per second. Each point cloud contains roughly 250,000 points, spanning across a range of indoor and outdoor environments, including urban locales, forests, and industrial facilities. The ground truth for each point cloud pair is obtained via high-precision GPS/IMU data from an Applanix POS LV420 system integrated with the robot platform. The GPS/IMU data allows for accurate estimation of the robot's position and orientation, resulting in reliable ground truth transformations for benchmarking registration algorithms. The ground truth transformations have an accuracy of 1.8mm for translation and 0.34 degrees for rotation (Fontana et al., 2021). This dataset offers a range of challenges such as diverse point densities, obstructions, and intricate structures, providing an excellent test bed for gauging the performance of registration methods under diverse conditions.

The Cross Source Registration Benchmark (CSRB) dataset is a collection of 202 point cloud pairs from the indoor office environment. Obtained using three different modalities, the sensors possess distinct specifications like diverse fields of view, point densities, and accuracies, thus making this dataset a superior choice for evaluating the efficacy of point cloud registration techniques across different data sources. The ground truth data is obtained through manual human alignment that is then cross-referenced for validation.

The third dataset employed in this study is the SUN3D dataset, sourced from Princeton University’s Geometric Registration Benchmark. This dataset contains eight distinct sets of scene fragments, with each fragment constituted by a point cloud integrated from 50 depth frames using Truncated Signed Distance Function (TSDF) volumetric fusion. The main objective of this benchmark is to predict whether different scene fragments can be accurately aligned. Focusing exclusively on point cloud pairs that can be aligned in our investigation, we target our evaluation to algorithm performance.

By using these three distinct and representative datasets, we seek to gain a thorough understanding of the advantages and drawbacks of multiple point cloud registration methodologies, thereby contributing to the evolution of more reliable and precise methods for point cloud alignment.

## 4.3 Experimental Design

The experiment methodology being taken to test the performance of different registration methods involves several steps. The first of which is to extract the available pairs of point clouds to be registered from their respective datasets. This is done by calculating the overlap between point clouds and filtering them to include only those that exceed a certain threshold. For this report, an overlap threshold of 50% was chosen, to allow for testing registration problems with a wide range of overlap but while excluding those that are particularly challenging. This selection also represents the real-world use cases one could expect in robotics applications (Fontana et al, 2021). Full details on the preprocessing steps can be found in the Data Manual (provided as a supplement to this report).

After the initial processing step, point cloud pairs are loaded in from their respective datasets and are initially aligned to be in the same coordinate system given the given ground truth. Subsequently, a transformation is generated randomly given input parameters for the magnitude of both the translation and rotation and a random seed of the index of the experiment allows for reproducibility between methods. The rotation matrix is generated by first defining a random and a random 3D unit vector which serves as the rotation axis. The quaternion representation of the rotation is obtained and then converted to a 3x3 rotation matrix. The translation vector is simply a set of three random values, one for each dimension. The resulting transformation matrix is applied to the target point cloud and in practice becomes a new ground truth that we are trying to estimate. This process ensures reproducibility of the experiments and removes bias to easy or hard problems. For this report an initial displacement translation of up to 0.5 meters along each axis was tested along with a rotation of up to 60 degrees. This approach also removes cases where point clouds are closely initially aligned or misaligned, as is the case with some datasets, and allows for experimentation with different levels of initial perturbation to better assess registration algorithm performance.

With the point cloud pairs now offset by this randomly generated transformation, the point clouds are down sampled using voxel down sampling. In this process a 3D grid of voxels is generated, with a given voxel size as input, and then points within each voxel are reduced to a single point representing the centroid (average position of points). The voxel size chosen in this study was 0.02mm, this allows for testing high resolution point clouds whilst maintaining detailed features. Beneficially for runtime, this also reduces the overall size of the point cloud. However, the effectiveness of registration could be reduced since some detail may be lost in the scenes we are attempting to register.

Following down sampling, individual registration methods can be used to estimate the ground truth transformation. Each method takes a source and target point cloud as input an returns a 4x4 array of the estimated transformation. The performance of each method can then be tested and evaluated by comparing the estimate to the newly generated ground truth which we are trying to solve. By following this approach, it is possible to test and identify the most effective registration method for each dataset and under a wide range of initial conditions.

## 4.4 Implementation Details

For consistency and compatibility in the implementations, we utilized the Open3D library, a tool facilitating numerous 3D data processing tasks whilst other models are used from source. This evaluation protocol requires 588 experiments for testing each algorithm. While this could be easily extended to include more datasets, the time taken to evaluate each algorithm would be too long. The time taken to run experiments depends on the hardware used and the evaluation parameters, all experiments in this study are conducted on a 2.2GHz Intel Core i9 with 16GB DDR3 RAM. The approach for implementing the point cloud registration methods selected for this research is as follows. Each method's specific parameters were tuned based on preliminary testing and recommendations from existing literature to ensure the robustness and generalizability of the evaluation results.

Both FGR and RANSAC use the Open3D library's implementation with minimum modifications. Both utilize FPFH features computed using a radius size of 0.25. A distance threshold of 0.15 is used to limit the maximum correspondence distance between points when matching features. The Iterative Closest Point (ICP) method is implemented with two different variants: point-to-point and point-to-plane, and a third custom multi-scale variant. The primary input parameter is the threshold, set to 0.1, which defines the distance threshold for correspondence and the maximum number of iterations is set to 50. For point-to-plane ICP, in addition to the threshold, surface normals are computed with a 0.1 radius.

In the multi-scale ICP variant, the algorithm is applied at multiple scales. The scales are defined by the voxel sizes (0.1, 0.05, 0.01) and maximum iterations (50, 30, 14) parameters. The point clouds are down sampled using the specified voxel size at each scale, and ICP is applied with a maximum number of iterations. The resulting transformation is then used as the initial guess for the next scale, as it moves from a coarse resolution then refines the registration as it shifts to finer scales.

Finally, the Feature-Metric Registration method, which was directly obtained from a publicly accessible GitHub repository, was also utilized. The model was retrained using both ETH and SUN3D data to ensure we obtain an accurate representation of the performance on our data. However, we are limited in the quantity of cross source data available, so have decided not to use this data in training.

# Chapter 5: Experimental Results

This chapter lays out the experimental results, focusing on evaluating registration methods using a diverse set of well-established metrics to assure objective and quantitative evaluation. Section 5.1 highlights the metrics used for evaluation of each method’s performance and accuracy in relation to a known ground truth. Subsequently, sections 5.2 and 5.3 delve into the experimental results, offers analytical observations, and explores the derived implications of the results. The results underscore the pivotal role of dataset characteristics in influencing algorithm performance, suggesting a nuanced approach for algorithm selection based on specific tasks and dataset specifics. The chapter concludes in section 5.4 by reflecting on the inherent limitations of this study and their implications on the generalizability of the findings.

## 5.1 Evaluation Metrics

To ensure a rigorous and objective evaluation of the registration methods examined in this study, a set of well-established evaluation metrics will be utilized. These metrics make use of the known initial misalignment to assess the accuracy of the estimates generated by registration. While in real-world scenarios, this ground truth transformation is typically unknown, by using a known ground truth we can quantitatively evaluate registration quality. The evaluation metrics utilized in this study include Relative Translation Error (RTE), Relative Rotation Error (RRE), Root-Mean-Square Error (RMSE), Mean Absolute Error (MAE), Computation Time and a combined error metric.

RTE quantifies the discrepancy between the true and estimated translations, while RRE assesses the deviation between the true and estimated rotations (Censi, 2007). RRE is calculated as follows, first the transpose of the estimated rotation matrix is multiplied with the ground truth rotation matrix. The resulting matrix represents the relative rotation from the estimated rotation to the ground truth rotation. The sum of the elements along the main diagonal gives a measure of how aligned the rotations are, a perfect alignment would have a value of 3. This trace equals 2\*cos(theta) + 1, where theta is the angle of rotation. To isolate theta we subtract 1 and divide by 2, then take the inverse cosine. The angle is converted to degrees and returned. Success rate is defined as the proportion of transformations that fall within a predefined accuracy threshold for translation and rotation errors. The choice of the accuracy threshold is contingent on the specific application requirements and the expected error level (Rusinkiewicz & Levoy, 2001). The choice of error threshold chosen for this report is a 0.6 meter magnitude of translation and a 10 degree rotation.

Inspired by the work of Fontana et. al, we will be using a metric that combines rotation and translation error into a single indicative metric. This is done by taking the root mean squared distance between points both with the ground truth and transformation estimate applied. This metric is then made scale-invariant by dividing each distance between points by the distance of that point with respective to the centroid or mean of the points in the cloud. This is significantly more representative to be able to make comparisons between different pairs of point clouds as otherwise metrics would depend on the size of the point cloud.

RMSE is calculated by first transforming the source point cloud using both the estimated and ground truth transformation matrices. RMSE is then calculated by taking the square root of the mean of the square differences between the coordinates of the transformed points. This metric provides an understanding of the average translation error per point. The lower the RMSE, the closer the estimated transformation is to the ground truth.

Lastly, Computation Time measures the duration of time it takes for the registration algorithm to estimate the transformation, a critical factor for real-time practical applications (Low, 2004). Overall, the use of these evaluation metrics facilitates a comprehensive assessment of the performance of the registration methods under consideration, allowing for a systematic comparison of their strengths and limitations in registering point cloud data from diverse sources.

## 5.2 Experimental Results and Analysis

In total six methods are evaluated in our experiments on five diverse datasets. In this analysis we will highlight the results from the combined error metric, success rate and runtime. Analyzing the Average Success Rate using a 0.6 meter distance and 10-degree rotation threshold, where a higher score signifies a better performance, with 1 being a 100 percent success. In Figure 3 an example scene from the Cross Source Dataset depicting chairs and a table can be seen with different registration methods applied.

**Figure 3***Example Registration Problem*

Left Column (top to bottom): Intial Offset, ICP, multi-scale ICP, point-to-plane ICP

Right Column (top to bottom): Ground Truth, RANSAC, FGR, FMR



Examining algorithm performance using the combined error metric, we can see a diverse pattern of results emerges across the datasets. The best results across all the datasets are obtained by RANSAC on the SUN3D datasets. Surprisingly, the more naïve ICP approaches performed best on both the ETH datasets, with multi scale ICP in particular obtaining the lowest combined error values. Meanwhile, for the cross source dataset the deep learning approach FMR performed the best significantly outperforming other methods. However, FMR performed extremely poorly on the ETH gazebo winter dataset, suggesting the model was not able to adequately extract identifiable features in this outdoor dataset. This could be explained by limitations in the data used for in training the model. The dataset-specific performance of the algorithms varies significantly, underlining the crucial role of dataset specific characteristics on algorithm performance.

**Table 3***Average Combined Error Metric*

*Lower is better*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | cross-source | ETH-apartment | ETH-gazebo | SUN3D-hotel | SUN3D-lab |
| FGR | 1.1895 | 2.4721 | 1.6403 | 0.7136 | 0.5438 |
| FMR | **0.6082** | 2.5976 | 6.373 | 1.1627 | 0.684 |
| ICP | 1.7552 | 1.8958 | 0.9978 | 1.3268 | 1.443 |
| RANSAC | 1.6009 | 2.672 | 1.8528 | **0.2059** | **0.2854** |
| multi\_scale | 1.652 | **1.8742** | **0.9444** | 1.0824 | 1.2671 |
| point\_to\_plane | 2.0906 | 3.2323 | 0.9861 | 1.508 | 1.5009 |

The performance results were markedly different when consider the average success rate, providing several key insights. RANSAC once again stood out, due to its outstanding success rate on the SUN3D datasets, particularly on the 'sun3d-hotel’ dataset, where it reported over 90% success. Contrarily, the results were disappointing on the ETH and cross-source-dataset, where none of the algorithms achieved a success rate of more than a quarter of the cases. Overall, the success rates vary significantly across datasets and algorithms. RANSAC's exemplary performance on the SUN3D datasets is countered by the generally disappointing results observed on both the ETH and cross-source-dataset. These observations underscore the need for careful selection of algorithms based on the specific characteristics and demands of each dataset.

**Table 4***Average Success Rate*

*(0.6 meters and 10 degree threshold) – Higher is better*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | cross-source | ETH-apartment | ETH-gazebo | SUN3D-hotel | SUN3D-lab |
| FGR | 0.0659 | 0.1643 | 0.0698 | 0.3636 | 0.4746 |
| FMR | **0.2637** | 0.0743 | 0.0127 | 0.1515 | 0.3559 |
| ICP | 0.1813 | 0.1142 | 0.1905 | 0.1515 | 0.1525 |
| RANSAC | 0.0989 | **0.2244** | 0.1206 | **0.9091** | **0.7797** |
| multi\_scale | 0.2527 | 0.1263 | **0.2127** | 0.2727 | 0.2542 |
| point\_to\_plane | 0.2363 | 0.1523 | 0.1587 | 0.2727 | 0.2712 |

Unsurprisingly, ICP shines in terms of Relative Translational Error (RTE), where it outperforms other methods on both the ETH and cross-source datasets. It also showcases a robust performance on SUN3D datasets, where RANSAC takes the lead. Despite ICP's commendable RTE performance, it falls short in terms of Relative Rotational Error (RRE). This shortcoming isn't unique to ICP, however. Most of the algorithms examined, while adept at handling the translational component of transformation, struggle significantly with the rotational component. The high average rotational figures across the board are indicative of this pervasive challenge.

**Table 5***Average Relative Translation Error*

*(meters) – Lower is better*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | cross-source | ETH-apartment | ETH-gazebo | SUN3D-hotel | SUN3D-lab |
| FGR | 4.3521 | 2.9853 | 3.2521 | 0.7878 | 0.678 |
| FMR | 1.6669 | 2.984 | 11.2309 | 0.9373 | 0.857 |
| ICP | **0.8385** | **0.8154** | **0.4366** | 0.6646 | 0.7842 |
| RANSAC | 5.0238 | 3.8865 | 3.6304 | **0.1779** | **0.5729** |
| multi\_scale | 0.8901 | 1.1111 | 0.4447 | 0.7149 | 0.9181 |
| point\_to\_plane | 1.4901 | 2.6423 | 0.4883 | 0.913 | 0.9392 |

**Table 6***Average Relative Rotation Error*

*(degrees) – Lower is better*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | cross-source | ETH-apartment | ETH-gazebo | SUN3D-hotel | SUN3D-lab |
| FGR | 89.7197 | 47.4453 | 60.0322 | 40.874 | 22.9456 |
| FMR | **27.983** | 33.849 | 38.296 | 35.448 | 24.2708 |
| ICP | 30.8139 | **30.3163** | 28.3503 | 40.6011 | 30.9882 |
| RANSAC | 110.997 | 69.3765 | 89.994 | **10.8918** | **16.2964** |
| multi\_scale | 28.0237 | 32.7141 | **27.1829** | 39.9672 | 35.2473 |
| point\_to\_plane | 34.7042 | 32.3188 | 27.9662 | 49.9432 | 34.08 |

In the exploration of algorithm runtimes across five different datasets, some key findings emerge. Notably, FMR consistently outshines its peers demonstrating the shortest runtime on four out of the five datasets. Suggesting deep learning approaches such as FMR are preferable in real time applications. On the other hand, multi-scale ICP typically takes longer to run. Despite its extended runtime, the performance gains over standard ICP and point-to-plane ICP are only modest, raising questions about the trade-offs involved in using this algorithm. This disparity underlines the complex relationship between efficiency and effectiveness, which is further underscored by the dataset-dependent performance.

**Table 7***Average Runtime*

*(Seconds) – Lower is better*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | cross-source | ETH-apartment | ETH-gazebo | SUN3D-hotel | SUN3D-lab |
| FGR | 2.5073 | 10.0535 | 26.0714 | 2.5659 | 3.7146 |
| FMR | **0.4107** | **1.2943** | 3.5713 | **0.1937** | **0.2011** |
| ICP | 1.2608 | 2.9497 | **2.1143** | 0.7855 | 0.8511 |
| RANSAC | 1.6463 | 5.1766 | 9.9826 | 1.1663 | 1.7363 |
| multi\_scale | 13.1786 | 32.2087 | 9.9483 | 17.3012 | 19.9138 |
| point\_to\_plane | 1.3063 | 3.0775 | 2.2539 | 0.6588 | 0.8082 |

Indeed, the effect of the dataset on runtime is a recurrent theme across the board. Each algorithm's runtime seems to fluctuate significantly depending on the dataset at hand. This suggests the need for algorithm selection to be contingent not only on the task but also on the specific nature of the data that the algorithm will be working with. FGR serves as a case in point for this dataset dependency. Despite its decent performance across most datasets, FGR's runtime on the ETH gazebo winter dataset is notably high. We surmise that this is due to the relative size of the scenes in the gazebo winter dataset being larger than in others, as they are outdoor scenes. FGR appears less efficient in such scenarios, indicating a possible limitation of this algorithm for larger datasets by volume.

Looking at the calculated Spearman’s rank correlation coefficient between initial misalignment and the combined error metric, we observe a uniformly positive correlation between the initial misalignment and the error metric for the ICP variants. However, the effect on the feature based methods and the deep learning method is much less pronounced. There is a slight positive correlation but the p-values are high meaning we are not as confident in this finding. Overall, these results suggest that a higher initial misalignment leads to a higher combined error, especially in the case of ICP and its variants.

**Table 8***Correlation: Initial Misalignment with Combined Error Metric*

*Coefficient (p-value)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | cross-source | ETH-apartment | ETH-gazebo | SUN3D-hotel | SUN3D-lab |
| FGR | 0.033  (0.648) | 0.0512  (0.253) | -0.1266 (0.0246) | 0.413 (0.0169) | 0.206  (<0.001) |
| FMR | 0.121  (0.106) | 0.0102  (0.821) | 0.0086  (0.879) | 0.174  (<0.001) | 0.0613  (0.645) |
| ICP | 0.716  (<0.001) | 0.704  (<0.001) | 0.696  (0.008) | 0.677 (<0.001) | 0.853  (<0.001) |
| RANSAC | -0.116  (0.118) | 0.0749  (0.095) | 0.111  (0.05) | 0.326  (0.065) | 0.241  (0.066) |
| multi\_scale | 0.687  (<0.001) | 0.672  (<0.001) | 0.706  (<0.001) | 0.756 (<0.001) | 0.806  (<0.001) |
| point\_to\_plane | 0.647  (<0.001) | 0.6906 (<0.001) | 0.696  (<0.001) | 0.611 (<0.001) | 0.756  (<0.001) |

A different picture is seen in the results for the correlation between overlap and the combined error metric. Here we find with very low confidence for ICP and its variants, that there is a slight trend towards a negative correlation between overlap and combined error. For the feature based and deep learning methods, we find with much higher confidence that there is negative correlation between overlap and the error. This confirms our supposition that with greater overlap the error from registration decreases. However it is worth noting, that we do not see the same correlation on the cross source dataset.

**Table 9***Correlation: Overlap with Combined Error Metric*

*Coefficient (p-value)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | cross-source | ETH-apartment | ETH-gazebo | SUN3D-hotel | SUN3D-lab |
| FGR | -0.032  (0.664) | -0.142  (0.002) | -0.2222  (<0.001) | -0.407 (0.0187) | -0.535 (<0.001) |
| FMR | -0.0896  (0.230) | -0.258  (<0.001) | -0.1236 (0.0285) | -0.331  (0.06) | -0.572 (<0.001) |
| ICP | 0.113  (0.129) | -0.019  (0.665) | -0.149  (0.008) | -0.103  (0.569) | 0.044  (0.739) |
| RANSAC | -0.0315  (0.673) | 0.076  (0.089) | -0.185  (<0.001) | -0.336  (0.056) | -0.457 (<0.001) |
| multi\_scale | 0.125  (0.0928) | -0.03  (0.50) | -0.156  (0.005) | -0.114  (0.529) | 0.039  (0.77) |
| point\_to\_plane | 0.123  (0.099) | -0.0025  (0.956) | -0.144  (0.010) | -0.1203 (0.505) | 0.011  (0.934) |

## 5.3 Observations

This study brings to light the imperative of a nuanced approach in the selection of registration algorithms, underlining the fact that there is no one-size-fits-all solution that can be universally applied across all scenarios. Performance results indicate considerable variation across different algorithms in dealing with different problems and datasets. In some instances, an algorithm that triumphs in terms of error rates may falter when it comes to success rates, and vice versa. This suggests that no single method is universally successful in registering all the environments tested, nor is there a singular evaluation metric that can comprehensively account for all aspects of performance. From our examination of algorithm runtime, it becomes evident that while deep learning methods like FMR offer promising performance in real-time applications, the choice of algorithm must also consider the specifics of the dataset and the trade-offs between runtime and accuracy.

Indeed, the effectiveness of an algorithm appears to be intrinsically tied to the specific nature and characteristics inherent to the dataset in question and the objectives of the task at hand. As such, for optimal results, it is recommended that multiple methods be tested for each unique use case, and the most suited method selected accordingly. A combined error metric and success rate were key in highlighting the relative adaptability and efficacy of the methods in diverse environments, revealing no universally superior method. In light of our experimental results, we propose that algorithm selection be dependent upon the dataset and capture modality. For instance, based on these results, we would suggest the use of RANSAC for the SUN3D datasets given its exemplary performance, the variants of ICP for the ETH datasets, and FMR for the cross-source dataset. However, it is worth noting that external factors such as the data used during model training, will significantly influence the performance of FMR and other learning-based methods.

Given these observations, there are several recommendations to be made for future research endeavors. The greatest takeaway is the need for open standardized benchmarking against a common set of datasets that this work hopes to provide. This will ensure advancements in the field are quantified, comparable and visible to all practitioners. The variability in performance of the methods examined in this paper highlights the need for further research on algorithms or workflows to improve both accuracy and consistency. Meanwhile the discrepancies in performance across different evaluation metrics underscore the need for a multi-faceted assessment strategy. The correlation analysis reinforces the influence of dataset characteristics on algorithm performance. The correlation trends suggest that there is an interdependence between initial misalignment and overlap with algorithm performance.

In conclusion, this research underscores the complex dynamics in the performance of registration algorithms across varying datasets and scenarios. The interplay between algorithm efficiency, effectiveness, and runtime makes it evident that no universal solution to the task currently exists, rather algorithm suitability is dependent on the dataset and use case under investigation. It emphasizes the critical role of thoughtful and strategic algorithm selection based on optimizing error, runtime, and success rates in diverse scenarios. It is essential to pilot multiple methods before pinpointing a single technique for deployment. While the current methods provide valuable tools for alignment, their effectiveness in inextricably linked to the characteristics of the data they are applied to. Moving forward, as applications for these algorithms continue to expand and diversify in fields like robotics, autonomous driving, or virtual reality, it becomes even more crucial to refine our understanding and effective deployment of these varied registration algorithms to meet specific task demands.

## 5.4 Limitations

There are several limitations to this study which must be reported. Principally there is a lack in the quantity of data. This especially impacts our ability to train the models, we used on ETH and SUN3D data in training because of this. This lack of data also means we are limited in the environments and scenes that are being tested in the report, impacting the ability to generalize our conclusions to new data. Another critique of the data is in terms of accuracy, while we have a good understanding of the accuracy of both points and ground truth for the ETH datasets, we cannot say the same for the SUN3D or cross source datasets. Neither of these datasets report sensor accuracy nor ground truth accuracy.

Secondly, the hyperparameters used in function execution could be refined to tune for each dataset. Although this would take considerable time, we expect this would improve the results obtained. However, this emphasizes our conclusion that at present there is no single method that can be reliably used for each use case, and extensive testing is required for each use case.

Another critique would be that our routine for testing each dataset does not necessarily allow for a fair comparison between datasets. This occurs because we are extracting registration problems that exceed a certain degree of overlap, but this does not mean we see a uniform sample of overlaps across different datasets. To allow for a fairer comparison we could aim to uniformly sample different degrees of overlap from the testing data. Such a sample could be taken by selecting 10 registration problems for each 10-degree increment in overlap. This would have the added benefit of significantly reducing computation time of the evaluation program, by reducing the number of registration problems.

The methodology used to generate the random transformation whilst consistent, does not ensure all possible ranges of transformation are being tested. It may be preferable to consider a set of predetermined transformations that cover a wide spectrum of initial misplacements. Additionally, it might be valuable to consider testing of the same registration problems with different levels of initial perturbations, this way we could better understand how each method behaves with different transformations on the same problem.

Recent research has focused on using optimization-based techniques in conjunction with other registration methods. These hybrid registration methods use feature-based or learning-based approaches to provide an initial global transformation estimate, then fine-tune this estimate with an optimization-based approach. This offers improved performance and robustness, overcoming the sensitivity of optimization-based techniques to the initial offset. By incorporating this and other methods into the testing, we could produce better recommendations for which method to use for each task. Additionally, we would seek to explore the use of different feature descriptors, such as Spin Images (Johnson & Herbert, 1999) and SHOT (Tombari et al., 2010).

# Chapter 6: Conclusions and Future Work

In summation, point clouds, with their precision and accuracy, are invaluable for modeling the shape or surface of an object, and their versatile applications span across several fields. Indeed, the process of point cloud registration is a crucial step towards achieving a comprehensive and authentic representation of any given scene or object. Our study has led to the development of a comprehensive framework for evaluating the performance of registration methods, with recommendations made for the datasets under review. Since performance is highly dependent on individual use cases, testing of a variety of methods should be conducted before one is selected to use.

We would seek to enable the use of more publicly available datasets into the evaluation framework whilst also making the necessary modifications highlighted in the limitations section. Specifically, additional testing is needed on different feature descriptors, initial displacement transformations and degrees of overlap. In addition to adding more datasets, there is scope to augment the existing data to construct more challenging registration problems. This could involve inducing variations in data characteristics such as levels of gaussian noise, outliers and engineering partial overlap. The integration of these properties could be woven into the testing process, with the intention being to enhance testing on more disparate problems as seen in cross source data. Testing on point clouds with heterogeneous characteristics is critical for real world localization tasks that will involve the use of different sensors or acquisition techniques.

The contemporary methods, while somewhat proficient in handling same-source data, are often found wanting when confronted with cross-source registration problems. There is an evident need to augment the robustness and accuracy of registration in the face of variable data characteristics, as is the case with cross source data. Such enhancements will significantly contribute towards making our point cloud registration techniques more adaptable and versatile. More work is needed to develop methods that will successfully register scenes from different sensors, to enable the interoperability of different sensors for a greater variety of tasks. Such an undertaking is not only fundamental to several applications but also has the potential to make a profound impact on the successful emergence and adoption of nascent technologies.

Looking towards the future, our work delineates a roadmap of potential advancements and areas of improvement. Whilst providing a highly customizable routine for testing registration methods, that allows for benchmarking against existing methods. Research in this sphere is continually striving to augment the accuracy, speed, and robustness of point cloud registration algorithms, to facilitate their broader use in real-world applications.

# References

Apollonio, F. I., Gaiani, M., & Sun, Z. (2014). 3D reality-based artifact models for the management of archaeological sites using 3D Gis: A framework starting from the case study of the Pompeii Archaeological area. Journal of Archaeological Science, 41, 464-475.

Araújo, A.M.C. and Oliveira, M.M. (2020) ‘A robust statistics approach for plane detection in unorganized point clouds’, Pattern Recognition, 100, p. 107115. doi:10.1016/j.patcog.2019.107115.

Besl, P.J. and McKay, N.D. (1992) “A method for registration of 3-D shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), pp. 239–256. Available at: https://doi.org/10.1109/34.121791.

Brightman, N., Fan, L. and Zhao, Y. (2023) “Point cloud registration: A mini-review of current state, challenging issues and future directions,” AIMS Geosciences, 9(1), pp. 68–85. Available at: <https://doi.org/10.3934/geosci.2023005>.

Chen, Y. and Medioni, G. (1992) ‘Object modeling by registration of Multiple Range Images’, *Proceedings. 1991 IEEE International Conference on Robotics and Automation* [Preprint]. doi:10.1109/robot.1991.132043.

Choi, S., Zhou, Q.-Y. and Koltun, V. (2015) ‘Robust reconstruction of indoor scenes’, *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* [Preprint]. doi:10.1109/cvpr.2015.7299195.

Choy, C., Dong, W. and Koltun, V. (2020) “Deep global registration,” *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* [Preprint]. Available at: https://doi.org/10.1109/cvpr42600.2020.00259.

Choy, C. et al. (2020) ‘High-dimensional convolutional networks for geometric pattern recognition’, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) [Preprint]. doi:10.1109/cvpr42600.2020.01124.

Chui, H. and Rangarajan, A. (2003) “A new point matching algorithm for non-rigid registration,” Computer Vision and Image Understanding, 89(2-3), pp. 114–141. Available at: <https://doi.org/10.1016/s1077-3142(03)00009-2>.

Dong, Z. et al. (2020) ‘Registration of large-scale terrestrial Laser Scanner Point Clouds: A review and benchmark’, ISPRS Journal of Photogrammetry and Remote Sensing, 163, pp. 327–342. doi:10.1016/j.isprsjprs.2020.03.013.

Elhashash, M., Albanwan, H. and Qin, R. (2022) ‘A review of mobile mapping systems: From sensors to applications’, Sensors, 22(11), p. 4262. doi:10.3390/s22114262.

Farahani, N. *et al.* (2017) ‘Three-dimensional imaging and scanning: Current and future applications for pathology’, *Journal of Pathology Informatics*, 8(1), p. 36. doi:10.4103/jpi.jpi\_32\_17.

Fischler, M.A. and Bolles, R.C. (1981) “Random Sample Consensus: A Paradigm for Model Fitting with Apphcatlons to Image Analysis and Automated Cartography,” *Communications of the ACM*, 24(6), pp. 381–395. Available at: https://doi.org/10.1145/358669.358692.

Feng, L., Weng, N.G. and Ma, L. (2023) ‘3D Point Cloud Data Registration algorithm based on Augmented Reality Technology’, *Wireless Communications and Mobile Computing*, 2023, pp. 1–11. doi:10.1155/2023/5169781.

Fontana, S. et al. (2021) “A benchmark for point clouds registration algorithms,” Robotics and Autonomous Systems, 140, p. 103734. Available at: https://doi.org/10.1016/j.robot.2021.103734.

Fontana, S., Sorrenti, D.G. and Lauro, F.D. (2023) *Assessing the practical applicability of neural-based point clouds registration algorithms: A comparative analysis* [Preprint]. doi:10.22541/au.168908592.24833908/v1.

Greenberg, A. (2020) *3D models help preserve landmarks like Notre Dame*, *PBS*. Public Broadcasting Service. Available at: https://www.pbs.org/wgbh/nova/article/3d-model-point-cloud-notre-dame/ (Accessed: April 17, 2023).

Huang, X., Mei, G. and Zhang, J. (2020) “Feature-metric registration: A fast semi-supervised approach for Robust Point Cloud registration without correspondences,” *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* [Preprint]. Available at: <https://doi.org/10.1109/cvpr42600.2020.01138>.

Huang, X., Mei, G., Zhang, J., & Abbas, R. (2021). A comprehensive survey on point cloud registration. *ArXiv*. /abs/2103.02690

Huang, X. et al. (2019) ‘Fast registration for Cross-Source point clouds by using weak regional affinity and pixel-wise refinement’, 2019 IEEE International Conference on Multimedia and Expo (ICME) [Preprint]. doi:10.1109/icme.2019.00268.

Li, S. *et al.* (2016) “Tree point clouds registration using an improved ICP algorithm based on KD-tree,” *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* [Preprint]. Available at: https://doi.org/10.1109/igarss.2016.7730186.

Li, Y. and Ibanez-Guzman, J. (2020) ‘Lidar for autonomous driving: The principles, challenges, and trends for Automotive Lidar and Perception Systems’, IEEE Signal Processing Magazine, 37(4), pp. 50–61. doi:10.1109/msp.2020.2973615.

Mora, R. et al. (2021) ‘Automatic Point-Cloud Registration for Quality Control in building works’, Applied Sciences, 11(4), p. 1465. doi:10.3390/app11041465.

Myronenko, A. and Xubo Song (2010) “Point set registration: Coherent Point Drift,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(12), pp. 2262–2275. Available at: <https://doi.org/10.1109/tpami.2010.46>.

Pomerleau, F. *et al.* (2012) ‘Challenging data sets for Point Cloud Registration Algorithms’, *The International Journal of Robotics Research*, 31(14), pp. 1705–1711. doi:10.1177/0278364912458814.

Pu, X. et al. (2022) ‘Feature analysis of scanning point cloud of structure and research on hole repair technology considering space-ground multi-source 3D data acquisition’, Sensors, 22(24), p. 9627. doi:10.3390/s22249627.

Rusinkiewicz, S. and Levoy, M. (2001) “Efficient variants of the ICP algorithm,” Proceedings Third International Conference on 3-D Digital Imaging and Modeling [Preprint]. Available at: https://doi.org/10.1109/im.2001.924423.

Rusu, R.B., Blodow, N. and Beetz, M. (2009) “Fast point feature histograms (FPFH) for 3D registration,” 2009 IEEE International Conference on Robotics and Automation [Preprint]. Available at: <https://doi.org/10.1109/robot.2009.5152473>

Segal, A., Haehnel, D. and Thrun, S. (2009) “Generalized-ICP,” Robotics: Science and Systems V [Preprint]. Available at: <https://doi.org/10.15607/rss.2009.v.021>.

Shao, J. et al. (2020) ‘Slam-aided forest plot mapping combining terrestrial and mobile laser scanning’, ISPRS Journal of Photogrammetry and Remote Sensing, 163, pp. 214–230. doi:10.1016/j.isprsjprs.2020.03.008.

Shi, S. *et al.* (2020) ‘From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network’, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1. doi:10.1109/tpami.2020.2977026.

Stilla, U. and Xu, Y. (2023) ‘Change detection of urban objects using 3D point clouds: A Review’, *ISPRS Journal of Photogrammetry and Remote Sensing*, 197, pp. 228–255. doi:10.1016/j.isprsjprs.2023.01.010.

Takimoto, R.Y. *et al.* (2016) ‘3D reconstruction and Multiple Point Cloud registration using a low precision RGB-D sensor’, *Mechatronics*, 35, pp. 11–22. doi:10.1016/j.mechatronics.2015.10.014.

Torsello, A., Rodola, E. and Albarelli, A. (2011) ‘Multiview registration via graph diffusion of dual quaternions’, *CVPR 2011* [Preprint]. doi:10.1109/cvpr.2011.5995565.

Wang, Q. and Kim, M.-K. (2019) “Applications of 3D point cloud data in the construction industry: A fifteen-year review from 2004 to 2018,” *Advanced Engineering Informatics*, 39, pp. 306–319. Available at: https://doi.org/10.1016/j.aei.2019.02.007.

Wang, S. et al. (2020) ‘Robust 3D reconstruction of building surfaces from point clouds based on structural and closed constraints’, ISPRS Journal of Photogrammetry and Remote Sensing, 170, pp. 29–44. doi:10.1016/j.isprsjprs.2020.09.004.

Xiao, J., Owens, A. and Torralba, A. (2013) ‘SUN3D: A database of big spaces reconstructed using SFM and object labels’, *2013 IEEE International Conference on Computer Vision* [Preprint]. doi:10.1109/iccv.2013.458.

Zhang, Z. et al. (2021) ‘DDRNet: Fast Point Cloud Registration Network for large-scale scenes’, ISPRS Journal of Photogrammetry and Remote Sensing, 175, pp. 184–198. doi:10.1016/j.isprsjprs.2021.03.003.

Zhou, Q.-Y., Park, J. and Koltun, V. (2016) “Fast global registration,” *Computer Vision – ECCV 2016*, pp. 766–782. Available at: https://doi.org/10.1007/978-3-319-46475-6\_47.

Zhou, T. *et al.* (2017) “Unsupervised learning of depth and ego-motion from video,” *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* [Preprint]. Available at: https://doi.org/10.1109/cvpr.2017.700.