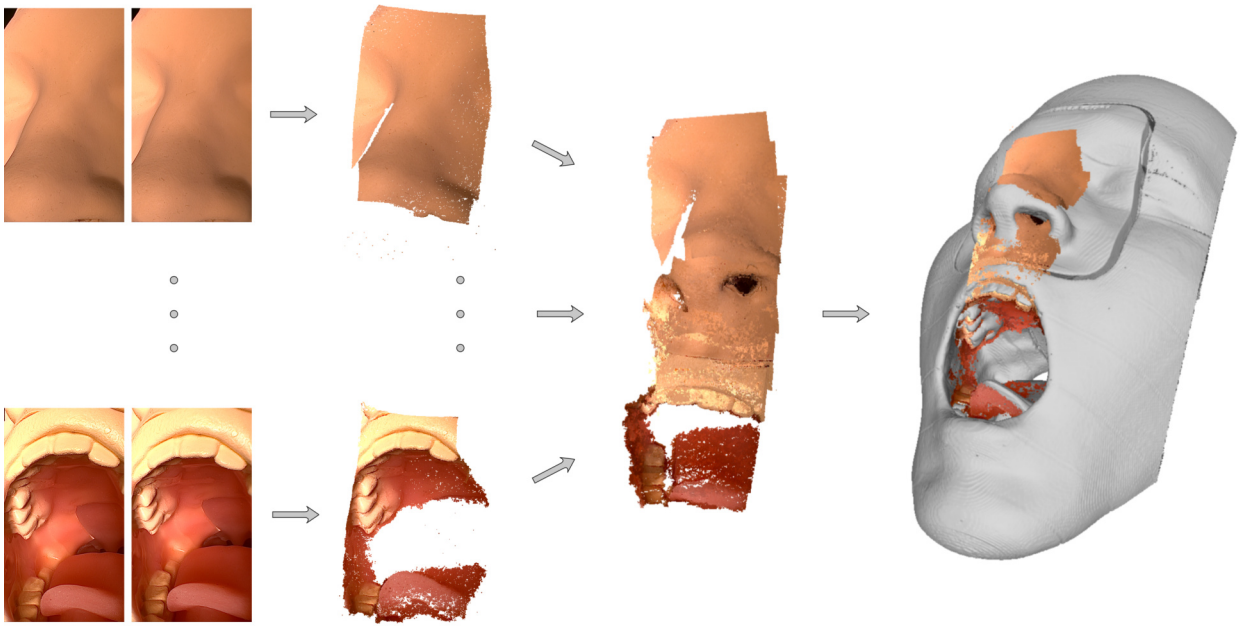
CHALMERS
UNIVERSITY OF TECHNOLOGY



Automatic registration of anatomical structures between stereo-endoscopic images and CT images

Master's thesis in Biomedical Engineering

Sophie Beckmann

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022
www.chalmers.se

Master's thesis 2022

Automatic registration of anatomical structures between stereo-endoscopic images and CT images

SOPHIE BECKMANN

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Signal Processing and Biomedical Engineering
Computer Vision and Medical Image Analysis Research Group
Chalmers University of Technology
Gothenburg, Sweden 2022

Automatic registration of anatomical structures between stereo-endoscopic images and CT images
SOPHIE BECKMANN

Supervisor: Jean-Claude Rosenthal, Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute, HHI
Examiner: Fredrik Kahl, Department of Electrical Engineering

Master's Thesis 2022
Department of Electrical Engineering
Division of Signal Processing and Biomedical Engineering
Computer Vision and Medical Image Analysis Research Group
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Visualization of the registration process utilizing stereo endoscopic-images to obtain point clouds which are registered among each other and additionally with a 3D CT model

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2022

Automatic registration of anatomical structures between stereo-endoscopic images and CT images
SOPHIE BECKMANN
Department of Electrical Engineering
Chalmers University of Technology

Abstract

In today's field of medical technology, image-guided surgery has become an essential component making image processing and analysis indispensable. Imaging systems such as (stereo-)endoscopy allow intra-operative 3D reconstructions and linkage with pre-operative data, e.g. CT images. To achieve this, image and point cloud registration algorithms are utilized where two or more images/point clouds are transformed into a common coordinate system. For this master's thesis, the aim is to develop and implement a modular framework for the registration of anatomical structures between stereo-endoscopic images in form of point clouds among each other as well as with 3D models from CT images.

The implementation is primarily based on Open3D and includes an analysis and registration pipeline. The analysis pipeline includes downsampling, normal estimation, feature extraction based on Fast Point Feature Histograms and correspondence estimation. In the registration pipeline the user can select between two global registration methods, namely RANSAC and TEASER, and three variants of the ICP algorithm, namely point-to-point ICP, point-to-plane ICP and colored ICP. In addition, a semi-automatic approach was implemented for the global registration between endoscopic point clouds and 3D CT models where the user selects correspondences.

The framework is evaluated on three datasets of which two are acquired utilizing stereo-endoscopes for ENT surgeries and laparoscopy. Objects are a medical head phantom and a specimen. An additional dataset are stereo frames from a porcine cadaver where ground truth data in form of a pose graph is available.

The results show that generally, TEASER performed more accurately than RANSAC, but differences between the ICP variants are neglectable. The automatic registration with CT data failed indicating that the estimated correspondences have a tremendous outlier ratio. However, the semi-automatic approach is an acceptable solution.

In conclusion, this thesis demonstrates that point cloud registration in the medical field is possible specifically among an imaging modality but remains challenging when utilizing differing modalities.

Keywords: image processing, image analysis, point cloud registration, medical image analysis, computer tomography, stereo endoscopy, iterative closest point, teaser, ransac

Acknowledgements

First and foremost, I would like to thank my supervisor Jean-Claude Rosenthal at the Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute, HHI in Berlin for his guidance throughout this project and whose invaluable expertise made this thesis possible.

I am also grateful to Eric Wisotzky for his constant input and to Jens Guether who was always helpful with C++ and Linux related problems. Additionally, special thanks to my group leader Christian Weissig and department heads Dr. Anna Hilsmann and Prof. Dr. Peter Eisert for their approval and support.

I would like to extend my sincere thanks to Prof. Dr. Fredrik Kahl, head of the Computer Vision Group at Chalmers University of Technology, for his supervision and endorsement.

Lastly, I would be remiss in not mentioning my friends and family, especially my parents and my husband Tyl, for their constant support and encouragement both for this thesis and throughout my studies.

Sophie Beckmann, Berlin, July 2022

List of Acronyms

CT	Computer Tomography.
DoF	Degrees of Freedom.
ENT	Ear, Nose and Throat.
FPFH	Fast Point Feature Histrogram.
GNC-TLS	Graduated Non-Convexity Truncated Least Squares.
ICP	Iterative Closest Point.
PCA	Principal Component Analysis.
PFH	Point Feature Histrogram.
RANSAC	RAndom SAmple Consensus.
RMSE	Root Mean Square Error.
SIFT	Scale-Invariant Feature Transform.
SKB	Semantic Kernels Binarized.
SPFH	Simple Point Feature Histrogram.
STAN	STereoscopic ANalyzer.
SURF	Speeded Up Robust Features.
TEASER	Truncated least squares Estimation And SEmidefinite Relaxation.
TIM	Translation Invariant Measurement.
TLS	Truncated Least Squares.
TRIM	Translation and Rotation Invariant Measurement.

Nomenclature

Below is the nomenclature of sets and variables that have been used throughout this thesis.

Sets

$\mathcal{P}$	Target point cloud
$\mathcal{Q}$	Source point cloud
$\mathcal{K}$	Correspondence set

Variables

R	Rotation
t	Translation
T	Transformation

Contents

List of Figures

List of Tables

1
Introduction

Image processing and image analysis are two essential components in today's medical field and its use increases continuously [1]. This rise also includes point cloud registration algorithms where two or more clouds of points in a 3D space are aligned [2, 3]. This chapter gives an introduction into the topic of this thesis and contains the background, aim, limitations and ethical considerations. Furthermore, an overview of the following chapters is given.

1.1 Background

In the medical field, imaging techniques are used to visualize the human anatomy and physiology [1]. Therewith, diseases and abnormalities of the patient can be examined and evaluated. Imaging technologies include non-invasive techniques e.g. ultrasound, x-ray and Magnetic Resonance Imaging (MRI) as well as invasive techniques for instance endoscopy. The different imaging modalities yield varying signals and thus, are used in different diagnostic or therapeutic situations of the patient's treatment. In addition, the practitioner may decide to combine several techniques to take advantage of their differing imaging modalities during a procedure e.g. by combining pre-operative date with intra-operative data.

One non-invasive technology that uses ionizing radiation in the form of x-rays is Computer Tomography (CT) [1]. Here, the anatomical structures of the body are displayed as slices which can then be reconstructed into 3D models [4]. These models may then be used to measure anatomical structures e.g. to manufacture a patient specific implant [5]. Figure 1.1 displays such a measurement.

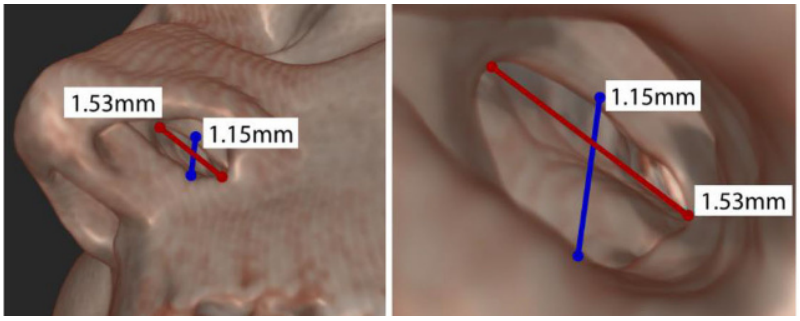


Figure 1.1: Reconstructed model from Computer Tomography (CT) slices by [4]

Endoscopy, on the other hand, is a (minimally) invasive technique [1]. It is an optical imaging technology allowing the practitioner to see internal body parts and is used to examine a variety of physiological systems. For instance, endoscopy is used in a wide field of surgical scenarios: Ear, Nose and Throat (ENT) and abdominal surgeries as well as for arthroscopy. Today, most endoscopes incorporate one or two digital cameras as well as light sources as visualized in Figure 1.2.

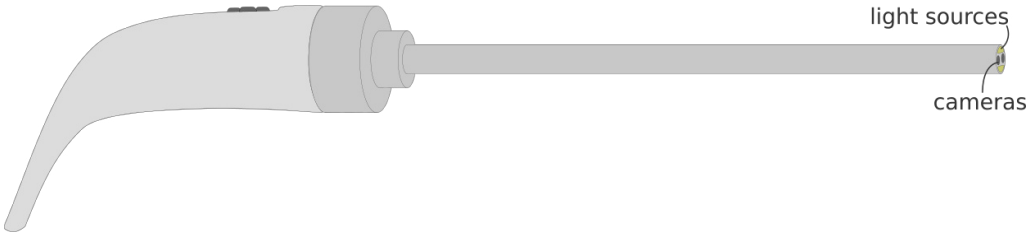
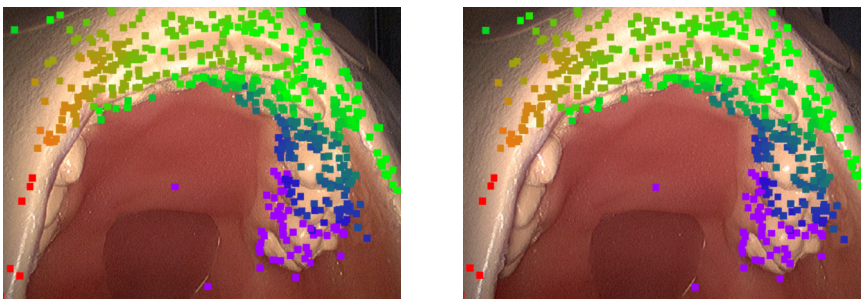


Figure 1.2: Stereo endoscope comprising two cameras and two light sources.

By incorporating two cameras, these systems can take advantage of stereo vision comparable to how most species observe the world using two eyes while focusing with both on the same object [6, 7]. This results in an inward movement of the eyes which is referred to as convergence. This convergence causes perceived binocular disparity and enables depth perception despite our eyes mapping a 3D world into a 2D space. As a result, objects appear in 3D and hence, this allows us to see with a precise position and distance estimation [8].

In image capturing systems, this effect is primarily achieved by two cameras focused on the same object equivalent to how our eyes are positioned. In the medical field, applications using stereo vision are surgical microscopy, endoscopy and robotic-assisted surgery [9]. These applications can provide a more realistic true-to-scale representation of the human anatomy than ordinary 2D images. Thereby, the perceived depth impression of the surgical scene allows a more rapid and effective performance and improves the training of medical staff [9–11]. Figure 1.3 shows the left and right stereo view of a mouth phantom captured with a stereo endoscope.



(a) Left stereo view **(b)** Right stereo view

Figure 1.3: Left **(a)** and right **(b)** stereo views of an ENT head phantom depicting corresponding feature points near the mouth / oral cavity

From a stereo pair, the depth can be calculated for each image pixel allowing a

3D reconstruction in form of a point cloud (see Figure 1.4). A point cloud is a representation of an object or environment as data points usually in a 3D space. Hence, a point cloud does not only store the x and y coordinates of an image, but also the depth z. In medical imaging, point clouds enable contactless and radiation-free measurements e.g. to manufacture implants designed specifically for each patient [4]. However, due to the medical topography and the endoscope's limited field-of-view, some areas are out of sight and not included in the point cloud with a single-shot [12]. Hence, for an authentic representation, several point clouds have to be acquired from different viewpoints and transformed into one common coordinate system which is referred to as point cloud registration.

Figure 1.4: Reconstructed point cloud of a mouth phantom

In point cloud registration, similar to image registration, two or more point clouds of the same 3D scene are aligned by using feature extraction and determining correspondences between the clouds [12, 13]. A registration algorithm estimates the transformation between each pair of point clouds by employing said correspondences and applies the transformation to locate each point cloud within a common global coordinate system. Then, the data of the point clouds is fused yielding one large-scale point cloud. In addition to overcome the line-of-sight problem, point cloud registration is used in surgical navigation [14, 15]. However, registration problems in the medical field are especially challenging since a static world assumption is not applicable most of the time. For instance, soft tissue, patient movement and blood flow alter images, and hence also point clouds, considerably [1]. This lowers the accuracy of the registration result or may even hinder a proper registration.

This thesis is conducted in cooperation with the Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute (HHI) which is a research institute in Berlin, Germany [16]. Specifically, it is carried out within the *Capture and Display Systems* group of the *Vision and Imaging Technologies* department at HHI. The research group focuses on ultra-high definition 3D scene analysis and display solutions which are used in media, industrial and medical applications [17].

1.2 Aim

In this thesis, the imaging devices are stereo-endoscopes for ENT surgery and laparoscopy. Therefore, the aim of this master's thesis is to develop a modular framework for point cloud registration obtained from stereoscopic images in the context of endoscopic surgery. This includes feature extraction, correspondence estimation and a selection of algorithms which register point clouds (semi-)automatically. In addition, the registration with 3D CT data is evaluated.

For data acquisition, three 3D endoscopes are used in combination with a selection of objects: a specimen where CAD model functions as a reference and a medical head phantom for which a CT model is available. Additionally, reconstructed point clouds from the SCARED Challenge based on the method of Rosenthal et al. is utilized (see [18]). For this dataset, pose information between the stereo pairs is available since the dataset was obtained with a da Vinci Xi surgical robot by [18].

The evaluation compares the different registration methods and their application on different datasets and test scenarios.

1.3 Limitations

Since this thesis project is conducted in a restricted time period, several limitations have to be set. Only a selected number of registration methods will be implemented and evaluated. Furthermore, only the quality of the registrations is evaluated; the time for the estimation is not taken into account.

1.4 Ethical Considerations

Ethical considerations include the utilized data and how this thesis is of importance to society. No patient data was used and was therefore no concern within the scope of this thesis. One dataset was acquired within the scope of a challenge using porcine cadavers [18]. Here, the ethical decisions rest with the challenge organizer.

This thesis is important to society to improve the training and performance of surgeries as well as the creation of implants. The registration framework can be used as a basis to observe surgeries more precisely and to train medical staff with 3D reconstructions of the human body. In addition, more accurate measurements for implants and prostheses can be made without exposing the patient to radiation.

1.5 Outline of this thesis

This thesis comprises six chapters. This chapter, Chapter 1, is an introduction and overview of the topic containing the aim, the limitations and the ethical considera-

tions.

Chapter 2 explains the fundamental theory. This includes the point cloud acquisition process based on the stereo principle and the analysis of point clouds. For the point cloud registration, two global registration methods and three variants of a local registration method are presented.

Chapter 3 describes the methods. In particular, the objects and reference data which are aimed to be registered as well as the endoscopes and software used for the acquisition process are mentioned. In addition, the three test scenarios are outlined.

Chapter 4 presents the results obtained with the previously explained methods.

Chapter 5 discusses the results and gives remarks about possible improvements and future research.

Chapter 6 provides a conclusion.

2

Theory

This chapter provides a detailed description of the theory on which this thesis and its methods are based on. First, the camera models for image and point cloud acquisition, their parameters and crucial aspects are outlined. Subsequently, feature extraction and correspondence estimation is presented. Then, an overview of point cloud registration including its mathematical fundamentals is given in addition to the presentation of a selection of registration methods.

2.1 Image and Point Cloud Acquisition

In computer vision, 3D images and point clouds can be obtained by active techniques which emit a signal e.g. laser light and observe the back-scattered signal [12]. Passive techniques, on the other hand, make use of matching points in two or more images of the same scene either taken from one camera in motion or by several cameras. In this thesis, point clouds are obtained by two identical cameras facing the same direction and hence, only this acquisition process is outlined [12, 19].

2.1.1 Interior Orientation of a Camera

The interior orientation of a camera is based on the pinhole camera model, a widely used and also the most basic representation of image capturing devices [7, 19]. The model is illustrated in Figure 2.1.

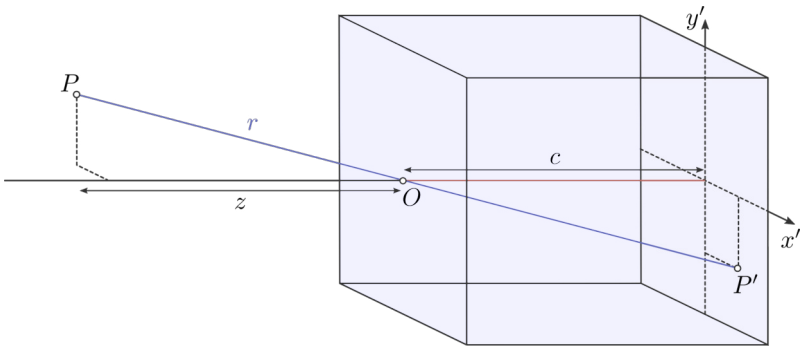


Figure 2.1: The principal of a pinhole camera model based on [19]. The image point P' is defined by the principal distance c and the depth z from the perspective centre O, through which all light rays r pass, to the object point P.

The cube represents the interior of the camera, while the outer part is referred to as the exterior [7, 19]. The perspective centre O is the pinhole through which all light rays r pass. Inside the camera, the distance between the image plane and the perspective centre is the principal distance c. Outside the camera, the distance between the object and the perspective centre is indicated as the object distance or depth z. The light rays travel from the object point P in the exterior through the pinhole onto the image plane at the back of the camera creating the image point P'. In digital imaging, colour images are often generated utilizing a CMOS-based sensor and hence, each point P' is represented by a pixel consisting of the three color channels red, green and blue within a 2D grid.

Beyond spatial position of the perspective centre and the principal distance within the camera coordinate system, physical cameras have optical distortions and alignment offsets w.r.t. to the sensor and cannot be formulated with the pure pinhole model. For instance, a camera often incorporates a lens and hence, an external perspective O and an internal perspective centre O' can be defined. For these cameras, the principal distance c is approximately equal to the focal length f, if the focus is set to infinity. Furthermore, the perspective centre may have an offset and consequently, the principal point H' does not lie in the center of the image coordinate system (see Figure 2.2). As a result, the position of the point P' is shifted.

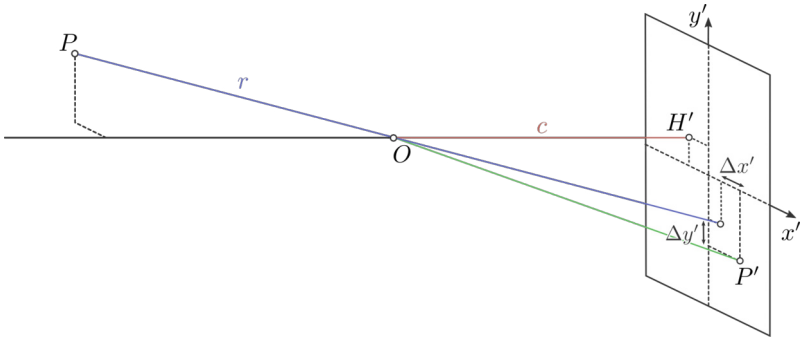


Figure 2.2: The interior orientation of a camera based on the pinhole camera model including an offset of the perspective centre O [19].

The offset of the perspective centre can be taken into account using the measured coordinates of the image point P' (x'_p, y'_p), the coordinates of the principal point H' (x'_h, y'_h) and the correction values error in the image plane $(\Delta x', \Delta y')$ [19]. Then, the vector r can be defined as follows:

$$r = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} x'_p - x'_h - \Delta x' \\ y'_p - y'_h - \Delta y' \\ -c \end{bmatrix} \tag{2.1}$$

The interior orientation parameters are determined by single camera calibration (see subsection 2.1.4).

2.1.2 Exterior Orientation of a Camera

The exterior orientation of the camera defines its position and orientation within a global coordinate system (see Figure 2.3) and is described by six parameters [19].

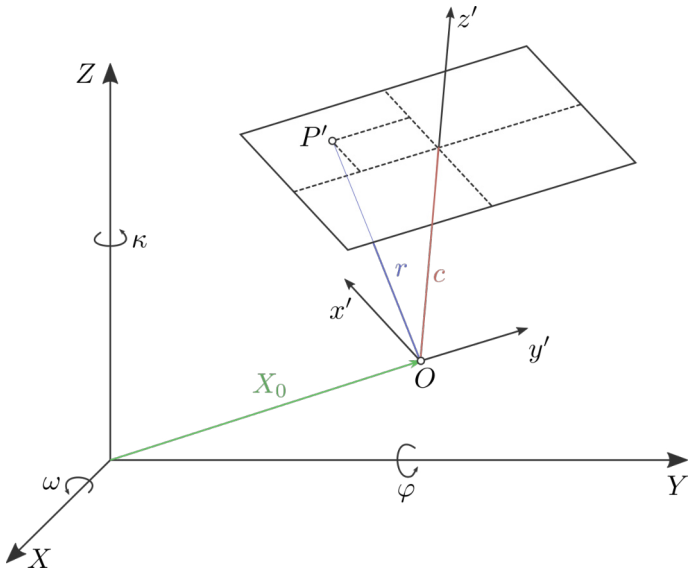


Figure 2.3: The exterior orientation of the camera and image coordinate systems according to [19]. The vector X_0 in conjunction with the angles ϕ, ω and κ define the position and orientation of the camera coordinate system.

The vector X_0 specifies the position of the perspective centre O along the X, Y and Z axes while the rotation matrix R incorporates the angles ω, ϕ and κ:

$$X_0 = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} \tag{2.2}$$

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \text{ with } R = R_\omega \cdot R_\phi \cdot R_\kappa \tag{2.3}$$

The rotation matrix can be calculated using Euler angles. Alternatively, rotations and spatial orientations of objects in 3D space may be described by quarternions [19–21]. The quaternions model is a four dimensional numerical system introduced by William Rowan Hamilton in 1843 which avoids ambiguity resulting in singularities compared to when using Euler angles to describe rotations. Furthermore, it simplifies the calculations and decreases computational time. For the model, Hamilton extended the real numbers with three imaginary units: i, j and k. The skew field $\mathbb{H}$ of Hamilton's quaternions is defined as follows:

$$\mathbb{H} := \{q = q_0 + q_1 \cdot i + q_2 \cdot j + q_3 \cdot k \mid q_0, q_1, q_2, q_3 \in \mathbb{R}\} \tag{2.4}$$

A unit quaternion is defined as $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$ with $q_0 \geq 0$. The rotation matrix of the unit quaternion q around a vector p is then described by:

$$R = q \cdot p \cdot q* \tag{2.5}$$

$$= \begin{pmatrix} q_0 \\ q_1 i \\ q_2 j \\ q_3 k \end{pmatrix} \begin{pmatrix} 0 \\ xi \\ yj \\ zk \end{pmatrix} \begin{pmatrix} q_0 \\ -q_1 i \\ -q_2 j \\ -q_3 k \end{pmatrix} \tag{2.6}$$

$$= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_o^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{2.7}$$

A more detailed derivation of the rotation matrix can be found in section A.1.

2.1.3 Lens Distortion

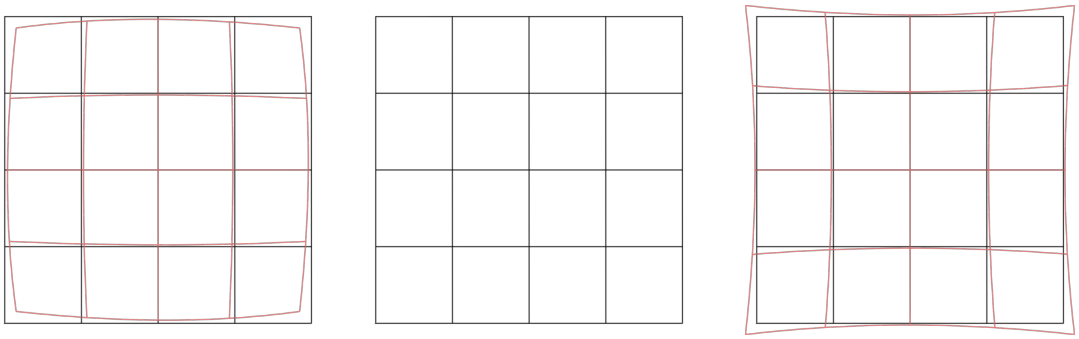
The image forming process often comes with image distortions due to the non-linear characteristics of lenses, so that straight lines in the real world are no longer straight lines in the image [19, 22]. The ideal situation without distortion effects is depicted in Figure 2.1 where the angle of incidence of the light rays is equal to the angle of emergence. However, camera lenses cause aberrations resulting in radial and tangential distortions. Radial distortions occur when light rays have a greater angle difference between incidence and emergence at the edges than at the centre. Tangential distortions arise when a lens is tilted and de-centred. In practice, only radial distortion (see Figure 2.4) are compensated while tangential distortions may be neglected due to increased assembling/manufacturing quality.

The ideal case with no distortions is referred to as orthoscopic. A (radial) barrel distortion is caused when the aperture is closer to the object resulting in a wider field of view of the lens than the size of the image sensor. Consequently, straight lines appear curved inwards. Pincushion distortion, on the other hand, occurs when the aperture is moved towards the image and therefore, the field of view is smaller than the size of the image sensor. As a result, straight lines are curved outwards.

The radial distortions can be described by an infinite series [22]:

$$\begin{aligned} x_u &= x_d(1 + k_1 r_d^2 + k_2 r_d^4 + ...) \\ y_u &= y_d(1 + k_1 r_d^2 + k_2 r_d^4 + ...) \end{aligned} \tag{2.8}$$

where $m_u = (x_u, y_u)$ and $m_d = (x_d, y_d)$ describe the undistorted and distorted image



(a) Barrel distortion **(b)** Orthoscopic **(c)** Pincushion distortion

Figure 2.4: Principles of radial lens distortion based on [19]. Radial distortions cause straight lines to curve **(a)** inward and **(c)** outward compared to the undistorted ideal case in **(b)**.

point respectively, the distorted radius is denoted by $r_d = \sqrt{x_d^2 + y_d^2}$ and where k_1 and k_2 are the radial distortion parameters. The order of the polynomial indicates the accuracy with which the distortion is eliminated.

2.1.4 Single Camera Calibration

Single camera calibration includes the internal and external orientation as well as the elimination of lens distortion. Usually, the calibration process is carried out by taking several pictures with different viewpoints of a calibration pattern e.g. a chequerboard pattern with a known dimension and structure. Based on detected corner points geometric characteristics can be derived e.g. different angles and distances [23]. This allows to estimate the camera and lens parameters based on the made observations compared to the underlying chequerboard reference data with the help of 2D-3D point correspondences.

For the determination of the interior orientation of each camera, the intrinsic camera parameters are stored in the 3x3 matrix K, referred to as the camera calibration matrix [7, 19]:

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.9}$$

The focal lengths in x and y dimensions are denoted by f_x and f_y respectively while (c_x, c_y) indicate the optical center in pixel coordinates also referred to as the principal point [7, 19]. The skew s describes the inclination between the sensor axes and the optical axes e.g. due to the rare case of non-squared pixels. In practice, the skew may be set to zero and the optical center to half the width in both x and y direction. As a result, the focal lengths are the only unknown intrinsic parameters.

Together with the extrinsic parameters, namely the rotation R and the translation t, the (total) projection matrix $P = K[R|t]$ can be specified [7, 19]. Then, the

projection between the known image point (x, y, z) and the corresponding known homogeneous object point $(X, Y, Z, 1)$ has the following form:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.10}$$

Both, the intrinsic and the extrinsic parameters, are then estimated using a linear equation system consisting of several projections.

Radial lens distortions are eliminated by assuring that straight lines in the 3D space also appear straight in the image plane [22]. With the usage of a calibration pattern the radial distortion parameters in Equation 2.8 are estimated to the desired degree of accuracy.

2.1.5 Stereo Camera Calibration

Besides the process of single camera calibration, it needs to be extended to stereo camera calibration, which describes the pose of the cameras w.r.t each other using the pinhole model. The ideal stereo model consists of two axis-parallel, non-convergent camera as shown in Figure 2.5 [19].

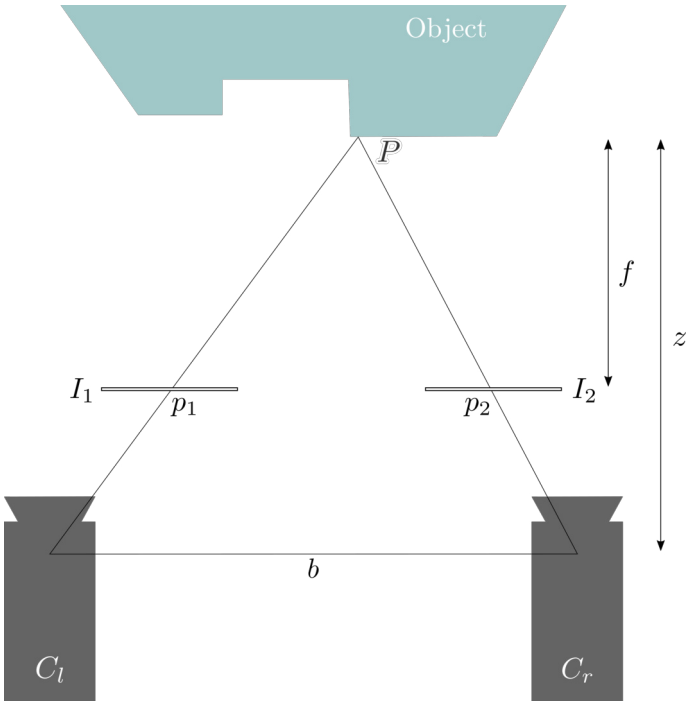


Figure 2.5: Special case: Stereo model consisting of two axis-parallel, non-convergent cameras C_l and C_r according to [19]. The focal length f and the stereo base b are crucial parameters to calculate the depth z.

Two identical cameras, C_l and C_r, face an object with parallel axes [19]. The distance between their perspective centres is the stereo base b. The depth z is given by the perpendicular distance from the stereo base to the object point P. The ratio of the depth and the stereo base specifies the accuracy with which the object point is represented in the image planes I_1 and I_2 respectively. The points p_1 and p_2 are corresponding image points in their respective image planes.

Assuming parallel axes, the disparity d is obtained by the distance or pixel difference in horizontal direction between two corresponding image points p_i and p_j from their respective image [19, 24]:

$$d_{ij} = p_i - p_j \tag{2.11}$$

The acquisition of the relationship between the point p_1 and the corresponding point p_2 is referred to as image matching and is based on finding feature points in each image [12, 19]. Local features focus on edge elements and striking regions within an image. These are efficient, relatively stable across a moderately wide range of perspectives and enable a unique identification as well as an accurate localization. The location of a local feature is obtained by a feature detector while a feature descriptor represents the characteristics of that feature. Since the location is distinctive, it is often referred to as a keypoint while its characteristic representation is a keypoint descriptor.

A variety of different keypoint detectors and descriptors exist to find matching image points [12, 19]. For 2D imaging, popular and robust feature detectors and descriptors include Scale-Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF). Both methods make use of the Gaussian blurr, however, the approach of approximation differs, and the keypoint descriptor is stored in form of a histogram [25, 26]. Another feature descriptor is Semantic Kernels Binarized (SKB) proposed by Zilly et al. where a located keypoint is described by filtered semantic kernels i.e. edges, ridges, corners, blobs, and saddles [27]. This method reduces the runtime and descriptor size compared to the former methods considerably and is specifically designed for real-time applications.

The relationship between corresponding points in stereo imaging is described by the fundamental matrix F [7, 19, 28]. This 3×3 matrix depends on the extrinsic and intrinsic camera parameters and must satisfy the following condition for all i:

$$x_i'^T F x_i = 0. \tag{2.12}$$

Here, x_i represents a set of points in one (e.g. left) image and x_i' are the corresponding points in another (e.g. right) image [7, 28]. The respective initial projection matrices are defined as

$$\begin{aligned} P &= K[I|0] \\ P' &= K'[R|t] \end{aligned} \tag{2.13}$$

where I is the identity matrix, R the rotation, t the translation and where K and K' represent the corresponding calibration matrices. Then, the fundamental matrix is defined as follows:

$$F = K'^{-T}[t]_{\times}RK^{-1} \quad \text{with} \quad t = -RC \tag{2.14}$$

Here, C describes the camera centre of the right camera.

In stereo image matching, it is often assumed that the corresponding points lie along a horizontal line as presented in Figure 2.6 [19]. In this ideal case, the axes of the cameras are parallel and hence, the obtained images lie within the same plane. The figure shows that the image acquisition of the object point P takes place along the projection lines r_1 and r_2 passing through the image points p_1 and p_2 and the perspective centre of the cameras O_1 and O_2. The project lines in conjunction with the stereo base form the epipolar plane. The intersection of the epipolar plane with the image planes is referred to as the epipolar lines k_1 and k_2. In this ideal case the epipolar lines are parallel to the horizontal direction of the images and hence, the image point q_2 of an additional object point Q only results in a horizontal shift along the epipolar line. Consequently, by assuming parallel axes of the cameras, the correspondence-problem is simplified to a horizontal line and hence, for the calculation of the disparity only the horizontal direction has to be taken into account.

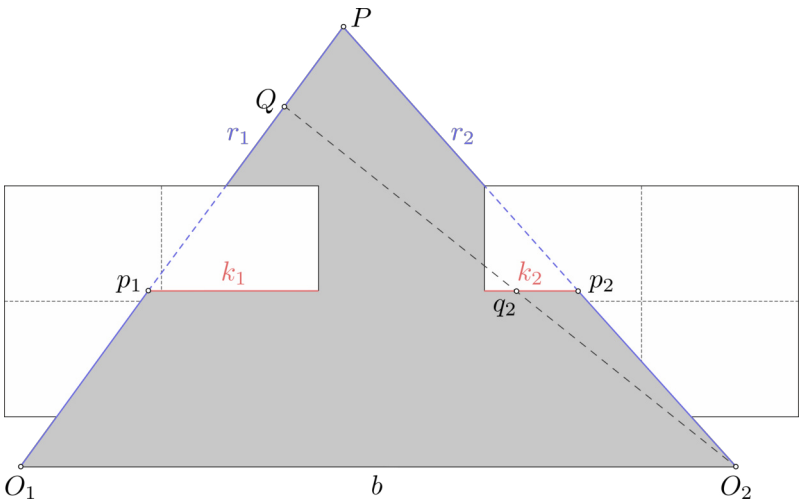


Figure 2.6: Epipolar geometry for parallel image axes based on [19]. The object point P is captured along the projection lines r_1 and r_2 resulting in the image points p_1 and p_2.

Assuming parallel-axes, the following applies to the parameters in Equation 2.14: $K' = K$, $R = I$ and $t = (1, 0, 0)^T$. As a result, the fundamental matrix has the form:

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \tag{2.15}$$

2.1.6 Stereo Rectification

Although it is generally assumed that the two cameras have parallel axes as described in the previous section, this is usually not the case in real-life scenarios [19, 28]. Instead, the cameras converge resulting in convergent image planes as visualized in Figure 2.7. For the user, this leads to exhaustion and eye strain. Furthermore, the epipolar lines are slanted and as a consequence, image matching and the calculation of the disparity between corresponding points is hindered as the correspondence search is no longer one-dimensional search problem.

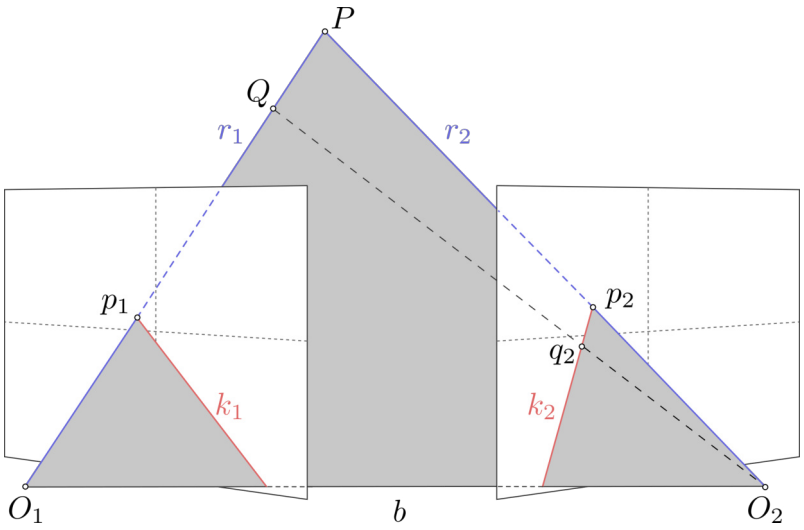


Figure 2.7: Epipolar geometry for convergent image axes based on [19]

These slight geometrical distortions can be corrected using image rectification [28, 29]. This method transforms the image planes parallel to the baseline by incorporating roll, tilt, y-shift and zoom and facilitates the challenge of finding corresponding feature points between images. It may be implemented in various ways of which some approaches apply a calibration process. However, calibration might not be sufficient or the data is not available and hence, correspondence-based rectification processes are utilized. These approaches use robust feature detectors such as SIFT, SURF or SKB to determine correspondences between the images (see subsection 2.1.5). Although these descriptors are robust, outliers are still present and hence, a technique such as RANSAC is applied (see subsection 2.3.2).

When roll, tilt, y-shift and zoom are estimated, they are stored in two homography matrices H and H' which are then applied onto their corresponding projection matrix in Equation 2.13 [28]. The rectifying homographies have the following form:

$$H = KR^T K^{-1} \tag{2.16}$$

2.1.7 Point Cloud Generation

From the rectified images, a point cloud can be reconstructed in 3D based on the estimation of the disparity map and the calibration data. The disparity map visualizing the horizontal shifts in pixel coordinates is generated according to Equation 2.11. To improve the disparity map, an iterative sweep method as presented by Waizenegger et al. may be utilized [30].

Then, with the usage of the determined stereo camera calibration data, the depth values can be calculated for each pixel pair:

$$z_{ij} = \frac{b \cdot f}{d_{ij}} \tag{2.17}$$

However, in real-life scenarios, the projection lines r_1 and r_2 as visualized in Figure 2.6 do not meet. Hence, the depth is estimated by finding the shortest distance between the projection lines. Together with the x and y coordinates, the data is stored in form of a point cloud.

2.2 Point Cloud Analysis

Point cloud registration is the process of fusing multiple point cloud of the same object or environment in to one common point cloud. In order to register point clouds, they must first be analyzed. This includes normal estimation, the calculation of geometric features and correspondence estimation [12, 19].

2.2.1 Normal Estimation

A normal vector $n = (n_x, n_y, n_z)^T$ describes the orientation of the surface in a point p [8]. A variety of methods exist to compute normal vectors in point clouds of which most perform eigenvalue decomposition. Furthermore, normals are determined by taking the neighboring points into account by applying a k nearest neighbors (k-NN) algorithm e.g. by using a k-d tree and/or radius search.

A k-dimensional tree (or k-d tree) is a binary tree often used in range and nearest neighbor search [31]. Each level of a tree represents one dimension. For point cloud registration, usually a 3D tree is used and hence, the tree consists of three levels. For each dimension, the median of values is determined splitting the dataset (here: point cloud) in two. Hence, for three dimensions, the dataset is split into eight cells referred to as leaf cells. These leaf cells may then be split again.

For the normal estimation, the centroid (or mean) μ of the neighborhood of a point p and the 3×3 covariance matrix Σ are determined [8, 32]:

$$\mu = \frac{1}{k} \sum_{i=1}^{k} p_i \tag{2.18}$$

$$\Sigma = \frac{1}{k} \sum_{i=1}^{k} (p_i - \mu)^T (p_i - \mu) \tag{2.19}$$

Then, the eigen vectors $\vec{v_m}$ and eigen values λ_m of the covariance matrix are obtained which fullfill the following condition:

$$\Sigma \cdot \vec{v_m} = \lambda_m \cdot \vec{v_m} \quad \text{with} \quad m \in \{0, 1, 2\} \tag{2.20}$$

Both, the eigen values and vectors, can be acquired by Principal Component Analysis (PCA), a method which extracts the principal components of a dataset by the size of the variance [8, 32]. For an m-dimensional dataset, m uncorrelated principal components and hence m eigen vectors and values are obtained. The eigen vector with the smallest corresponding eigen value constitutes the normal vector.

2.2.2 Fast Point Feature Histograms

Most image and point cloud registration methods depend on the calculation of geometrical features [19]. A variety of different approaches exist with which features

can be extracted; however, the current state-of-the-art approach for point cloud registration is called Fast Point Feature Histrogram (FPFH) and was presented by Rusu et al. in 2009 [33, 34]. An FPFH is based on Point Feature Histrogram (PFH) which are pose-invariant local features that describe the geometrical properties at a point p [34]. Their computation makes use of the neighborhood and the normal vector of a point.

A **Point Feature Histrogram** (**PFH**) describes the mean curvature at a query point p_q to encode the geometrical properties of the immediate neighborhood. Figure 2.8 illustrates the region of influence for a query point p_q (blue). The radius of the sphere whose centre lies at p_q defines the k neighbors which are then interconnected to a mesh. For each point pair, four features are computed and stored in a 16-bin histogram.

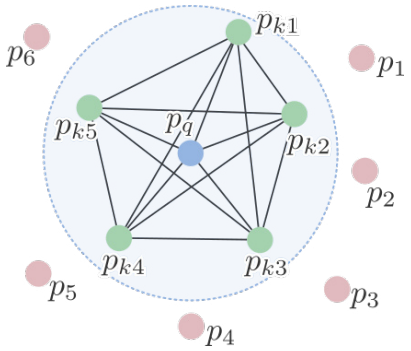


Figure 2.8: Influence region for a Point Feature Histogram based on [34]. The query point (blue) is connected with its nearest neighbors (green) within a defined radius

For two points p_i and p_j, their interconnection is computed using their normal vectors n_i and n_j where the point p_i has the smaller angle φ [32]. Their relationship is then calculated using a *Darboux frame* which is visualized in Figure 2.9.

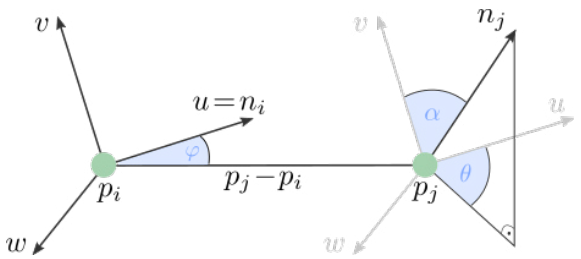


Figure 2.9: Darboux frame based on [32]. The normal vectors of the two points p_i and p_j are denoted by n_i and n_j.

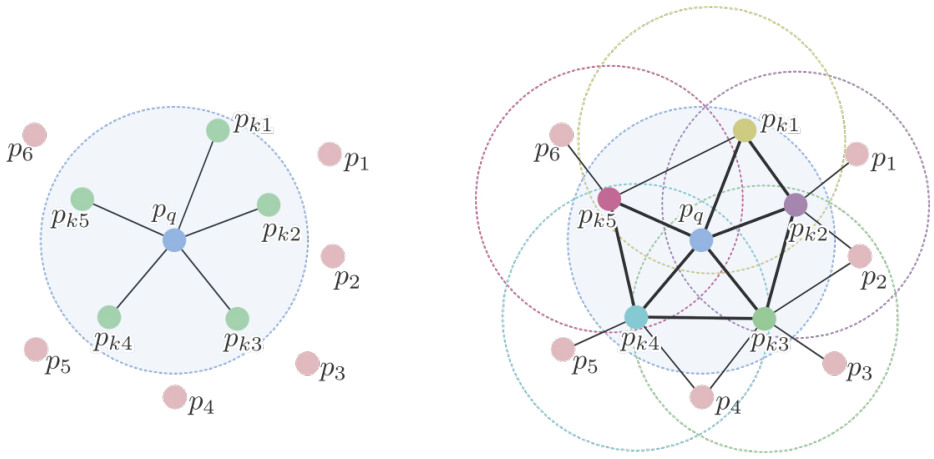
In the frame, the vectors u, v and w constitute the Cartesian coordinate system at the point p_i:

$$
\begin{aligned}
u &= n_i \\
v &= u \times \frac{(p_j - p_i)}{\|p_j - p_i\|} \\
w &= u \times v
\end{aligned}
\tag{2.21}
$$

Then, four features can be calculated to describe the difference between the normal vectors n_i and n_j. This includes the angles α, ϕ and θ as well as the Euclidean distance d:

$$
\begin{aligned}
\alpha &= v \cdot n_j \\
\varphi &= \frac{u \cdot (p_j - p_i)}{d} \\
\theta &= \arctan(\omega \cdot n_j, u \cdot n_j) \\
d &= \|p_j - p_i\|
\end{aligned}
\tag{2.22}
$$

A **Simple Point Feature Histrogram** (**SPFH**) takes only the interconnections to the nearest neighbors of the query point p_q into account as visualized in Figure 2.10a [34]. Hence, the neighbors are not fully interconnected as with the PFH.



(a) SPFH **(b)** FPFH

Figure 2.10: Influence region and relationships for **(a)** Simple Point Feature Histogram and **(b)** Fast Point Feature Histogram according to [34]. The query point (blue) is connected with its nearest neighbors within a defined radius. In **(b)**, the second neighbors are incorporated as well.

A **Fast Point Feature Histrogram** (**FPFH**) is computed on the basis of the SPFH (see 2.10b). Here, first the SPFH for a point p_q with its k neighbors is calculated. Then, in a subsequent step, the SPFH calculation is repeated for each neighboring point p_k. The final FPFH is calculated by incorporating the weight w_k which describes the distance between a query point p_q and its neighboring point p_k:

$$FPFH(p_q) = SPFH(p_q) + \frac{1}{k}\sum_{i=1}^{k}\frac{1}{\omega_k}\cdot SPFH(p_k) \tag{2.23}$$

Each feature is stored in a 33-bin histogram. Notice how some interconnections in the illustration are counted twice since the connection lies within the radius of two SPFH.

An advantage of the FPFH compared to the PFH is the complexity [32, 34]. For the PFH the following applies: for each neighborhood with k neighbors, $\frac{k-1}{2}$ of these relationships are calculated resulting in $O(n \cdot k^2)$ computations where n denotes the number of points within a point cloud. By using the FPFH, the amount of computations can be reduced to $O(n \cdot k)$.

2.2.3 Correspondence Estimation

For some registration methods, correspondences between the to be registered point clouds have to be estimated in beforehand. The correspondence problem is ill-posed since a robust and unique solution may not exist: there might not be a corresponding point, several points may be eligible as a corresponding point because of ambiguous structures in the object or the presence of noise [19]. In general, the solutions assume several conditions e.g. constant intensities, steady lightning and ambient effects as well as stable object texture for the duration of the image acquisition process. However, for real point clouds it is not uncommon to have more than 95% of outlier correspondences [35].

The estimation of correspondences based on FPFH descriptors can be conducted as follows: let $\mathcal{P}$ be the target point cloud and $\mathcal{Q}$ the source point cloud. Correspondences are estimated by comparing the feature histogram of a point p_i in $\mathcal{P}$ with histograms of points in $\mathcal{Q}$ [33, 34]. The histogram which is most similar to the target point is selected and the correspondence is saved in the correspondence set $\mathcal{K} = \{p_i,\ q_i\}$.

2.3 Point Cloud Registration

In point cloud registration, similar to image registration, two or more point clouds are aligned to map an object or a scene usually in a 3D space [2, 3]. Mathematically, the registration of two point clouds can be outlined as follows: let $\{\mathcal{P}, \mathcal{Q}\}$ be two point clouds of finite but possibly of different size where $\mathcal{P}$ denotes the target point cloud and $\mathcal{Q}$ indicates the source point cloud [36]. In general, both point clouds are expected to be within a real vector space of the same dimension $\mathbb{R}^d$. In case of point clouds taken from stereo images, a 3D vector space i.e. $\mathbb{R}^3$ is assumed. A registration algorithm yields to find the optimal transformation of the source point cloud to map onto the target point cloud given a correspondence set and hence, the best alignment between these point clouds.

A distinction is made between *correspondence-based registration* algorithms and *simultaneous pose and correspondence registration* methods [3]. Both approaches assume that there is a finite number of correspondences between the two point clouds. These are computed by finding a matching point in the target point cloud with respect to a point in the source point cloud e.g. by feature matching or by finding the smallest distance between points. The latter, also estimates the pose of the source point cloud concurrently.

Additionally, a difference is drawn between *global registration* and *local registration* methods. Global registration methods yield a coarse alignment between point clouds and do not require an initial alignment [37]. Furthermore, these methods usually operate on down-sampled point clouds for more efficiency. Local registration methods, on the other hand, refine the result of a global registration method and strive for the most accurate registration achievable.

2.3.1 Types of Transformations

For the registration, the type of transformation and its number of DoF have to be selected in accordance with the problem definition. Here, a distinction is made between linear transformations and non-rigid ones [38]. Linear transformations keep straight lines parallel and include translation, rotation and scaling as visualized in Figure 2.11. Non-rigid transformations, on the other hand, do not preserve angles and might not maintain parallelism.

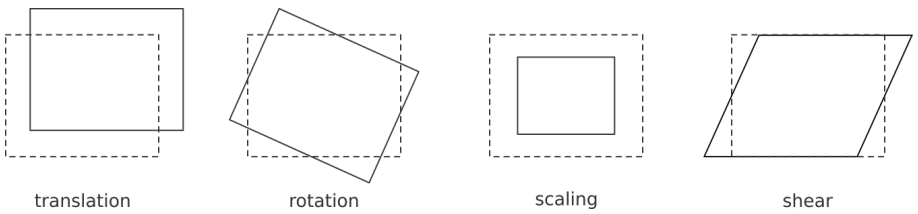


Figure 2.11: Types of transformations in 2D space according to [19, 39]

The **rigid transformation** (also referred to as Euclidean transformation) comprises six DoF in 3D space including three translations and three rotations and hence, a minimum of three correspondences is needed to find an adequate solution [2, 40]. Mathematically, the relationship between a point in the target point cloud $\mathcal{P}$ and a point in the source point cloud $\mathcal{Q}$ can be expressed as follows:

$$\mathcal{P} = tR\mathcal{Q} \tag{2.24}$$

$$\begin{bmatrix} x_\mathcal{P} \\ y_\mathcal{P} \\ z_\mathcal{P} \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_\mathcal{Q} \\ y_\mathcal{Q} \\ z_\mathcal{Q} \\ 1 \end{bmatrix} \tag{2.25}$$

The **similarity transformation** incorporates seven DoF involving three translations, three rotations and a scaling factor [2, 19, 40]:

$$\mathcal{P} = tRs\mathcal{Q} \tag{2.26}$$

$$\begin{bmatrix} x_\mathcal{P} \\ y_\mathcal{P} \\ z_\mathcal{P} \\ 1 \end{bmatrix} = \begin{bmatrix} s \cdot r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & s \cdot r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & s \cdot r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_\mathcal{Q} \\ y_\mathcal{Q} \\ z_\mathcal{Q} \\ 1 \end{bmatrix} \tag{2.27}$$

The **affine transformation** keeps straight lines parallel but does not preserve angles since in addition to including three translations, three rotations and three separate scaling factors, three angles for the shear h are considered as well [2, 19, 40]. Hence, in 3D space, this transformation has twelve DoF and is denoted as follows:

$$\mathcal{P} = tRs\mathcal{Q} \tag{2.28}$$

$$\begin{bmatrix} x_\mathcal{P} \\ y_\mathcal{P} \\ z_\mathcal{P} \\ 1 \end{bmatrix} = \begin{bmatrix} s_x \cdot r_{11} & h_x^y \cdot r_{12} & h_x^z \cdot r_{13} & t_x \\ h_y^x \cdot r_{21} & s_y \cdot r_{22} & h_y^z \cdot r_{23} & t_y \\ h_z^x \cdot r_{31} & h_z^y \cdot r_{32} & s_z \cdot r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_\mathcal{Q} \\ y_\mathcal{Q} \\ z_\mathcal{Q} \\ 1 \end{bmatrix} \tag{2.29}$$

The **projective transformation** (also known as homography) maps one plane to another via a central projection and comprises 15 DoF [19, 40]. In contrast to the affine transformation, this transformation does not preserve parallelism, but straight lines are still kept straight. In addition to the parameters incorporated in the affine transformation, the projective transformation includes the projection vector c:

$$\mathcal{P} = tRsc\mathcal{Q} \tag{2.30}$$

$$\begin{bmatrix} x_\mathcal{P} \\ y_\mathcal{P} \\ z_\mathcal{P} \\ 1 \end{bmatrix} = \begin{bmatrix} s_x \cdot r_{11} & h_x^y \cdot r_{12} & h_x^z \cdot r_{13} & t_x \\ h_y^x \cdot r_{21} & s_y \cdot r_{22} & h_y^z \cdot r_{23} & t_y \\ h_z^x \cdot r_{31} & h_z^y \cdot r_{32} & s_z \cdot r_{33} & t_z \\ c_x & c_y & c_z & 1 \end{bmatrix} \begin{bmatrix} x_\mathcal{Q} \\ y_\mathcal{Q} \\ z_\mathcal{Q} \\ 1 \end{bmatrix} \tag{2.31}$$

2.3.2 Random Sample and Consensus (RANSAC)

RAndom SAmple Consensus (RANSAC) is an iterative algorithm to find a suitable model for a set of data and was published by Fischler and Bolles in 1981 [41]. In point cloud registration, this method is classified as a global and correspondence-based registration algorithm [3]. The algorithm uses random sub-sampling to find a model with the greatest amount of inlier correspondences which is also referred as consensus set [7, 33]. It is especially known for being robust in the presence of outliers. Algorithm 1 describes its procedure for point cloud registration.

Algorithm 1 RANSAC

Input: Point clouds $\mathcal{Q} = \{q_i\}$ and $\mathcal{P} = \{p_j\}$, correspondence set $\mathcal{K} = \{p_i,\ q_i\}$, number of correspondences $N_{\mathcal{K}}$, inlier threshold D, distance threshold d
Output:

function RANSAC($\mathcal{Q}$, $\mathcal{P}$, D, d)
  $N = \infty$ ▷ Number of iterations
  $N_{done} = 0$ ▷ Number of iterations
  while $N > N_{done}$ **do**
    Select $N_{\mathcal{K}}$ random correspondences
    Compute transformation T based on that subset and apply to $\mathcal{Q}$
    Determine inlier correspondences within the distance threshold d
    if number of inliers $\geq$ inlier threshold D **then**
      Re-calculate the transformation T with all inliers and terminate
    else
      Select a new subset
    end if
    Update N based on Equation 2.32
  end while
  Use the best result
end function

Figure 2.12 illustrates this algorithm. The green points are points within the target point cloud $\mathcal{P}$ while the red ones represent the source point cloud $\mathcal{Q}$. The distance threshold d specifies if a data point is an inlier or not and therefore, if it is classified as part of the consensus set [7].

For the algorithm, several parameters have to be set: the distance threshold, the number of iterations and the acceptance rate for the largest consensus set [7]. In general, the distance threshold is chosen empirically i.e. the data point is an inlier with a probability of α. In practice, the probability for a data point to be an inlier is set to 95% and therefore, α is specified as 0.95. The amount of subsets and therefore the number of iterations N is chosen adequately high to make sure with a probability p that there exists at least one subset without outliers. Generally, this probability p is set to 0.99. The relationship between this probability, the number of iterations, the probability w that any point is an inlier and the size of the subset s is the following:

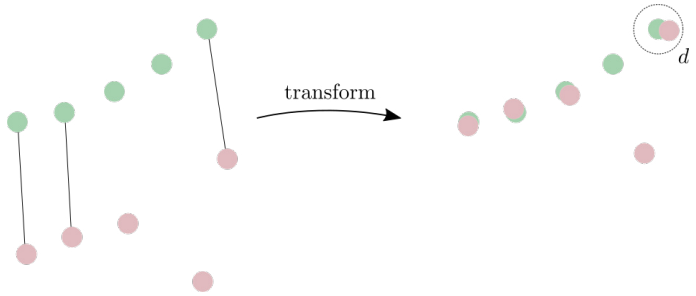


Figure 2.12: Illustration of the RANSAC algorithm for point cloud registration based on [7]. Given correspondences between the target point cloud (green) and the source point cloud (red), the transformation is estimated. The distance threshold d defines which correspondences are inliers.

$$N = \frac{\log(1-p)}{\log(1-\omega^s)} \tag{2.32}$$

Hence, if the dataset contains a large amount of outliers, the number of iterations has to be set greater than when less outliers are present. The third parameter which has to be chosen is the acceptance rate for the size of a consensus set [7]. A general rule is to set this rate similar to the proportion of inliers in the entire dataset.

Since the proportion of outliers is often unknown and therefore, the number of iterations cannot be estimated appropriately, and adaptive procedure may be chosen [7]. Here, the algorithm is initialized using an estimated worst-case scenario for the proportion of outliers $\epsilon = 1 - \omega$. Then, the number of iterations may be updated in case larger consensus sets are found. In this way, the number of iterations N can be initialised with infinite iterations and be decreased adaptively with each subset which has a larger consensus set than any subset beforehand.

2.3.3 Truncated least squares Estimation And SEmidefinite Relaxation (TEASER)

Truncated least squares Estimation And SEmidefinite Relaxation (TEASER) is a fast and certifiable point cloud registration method and was proposed by Yang et al. in 2020 [3]. This method is a global correspondence-based approach and its distinctive characteristic compared to other registration algorithms is that the registration result has to be certified. This means that the algorithm must present a certificate for the solution's quality or alternatively proclaim failure. Furthermore, the scale, the rotation and the translation are determined consecutively and in addition, it yields a higher accuracy and robustness than current state-of-the-art methods.

Assume $(\boldsymbol{p}_i, \boldsymbol{q}_i)$ is the i-th correspondence in a correspondence set $\mathcal{K}$ based on the two point clouds $\mathcal{P}$ and $\mathcal{Q}$ [3]. The correspondences comply to the following function where ϵ_i signifies noise:

$$p_i = s \cdot Rq_i + t + o_i + \epsilon_i \tag{2.33}$$

The vector o_i is a vector of zeros if the correspondence is an inlier or a vector of arbitrary numbers in case of an outlier correspondence. The correspondence is an inlier if the target point p_i is equivalent to a 3D transformation of the source point q_i plus noise. It is an outlier, if p_i is an arbitrary vector.

Most solvers use convex functions as a basis [42]. Those are functions where a line segment between any two points on the graph of that function lies above the graph [43]. An example of a convex function is least squares. Although convex functions are widely used, they are sensitive to outliers since large residuals dominate the cost. The TEASER algorithm relies on Truncated Least Squares (TLS) estimation, a non-linear and non-convex least squares method visualized in Figure 2.13 [3, 44]. As demonstrated, for large residuals the cost is constant while for small residuals, the cost is equal to the least squares function.

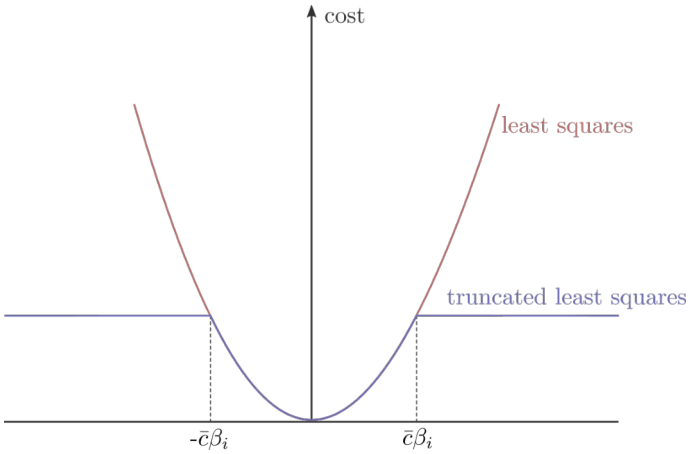


Figure 2.13: Truncated least squares according to [44].

Mathematically, the TLS registration has the following form:

$$\min_{s>0,\, t\in\mathbb{R}^3,\, R\in SO(3)} \sum_{i=1}^{N} \min\left(\frac{1}{\beta_i}\|p_i - (sRq_i + t)\|^2, \bar{c}^2\right) \tag{2.34}$$

Hence, a least squares solution is computed for small residuals i.e. if $(\frac{1}{\beta_i}\|p_i - (sRq_i + t)\|^2 \leq \bar{c}^2)$. In case of large residuals, the measurements are discarded. Furthermore, it is assumed that the inlier noise ϵ_i is smaller than the bound β_i: $\|\epsilon_i\| \leq \beta_i$. Additionally, it can be assumed that $\bar{c} = 1$.

A continuation of the above is Graduated Non-Convexity Truncated Least Squares (GNC-TLS) [3, 42]. This method is initiated with a convex function (here: least squares) and gradually changes to a non-convex function until a robust estimation is achieved. As visualized in Figure 2.14, this is accomplished by steadily increasing the factor μ in a surrogate function in an iterative optimization process.

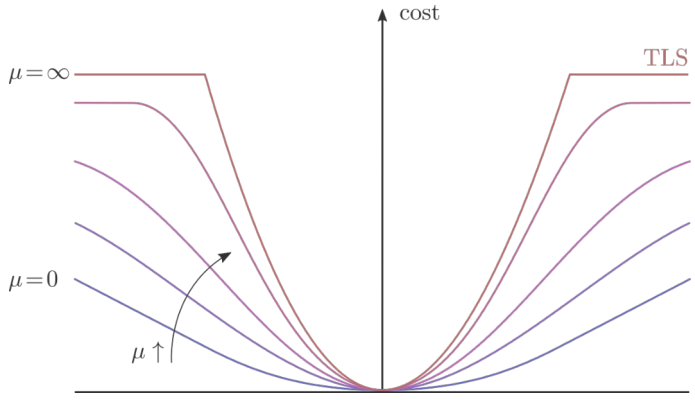


Figure 2.14: Graduated non-convexity truncated least squares as presented by [42].

As stated, in TEASER, the scale, rotation and translation are decoupled and estimated consecutively [3]. This is achieved by applying measurements that are invariant to translation and/or rotation. Translation Invariant Measurements (TIMs) are based on the idea that the absolute locations of the points in $\mathcal{P}$ are affected by the translation t, but the relative positions are not. For two correspondence pairs $\mathcal{k}_i = (p_i, q_i)$ and $\mathcal{k}_j = (p_j, q_j)$, the distance between the points p_i and p_j has the following form:

$$\underbrace{p_j - p_i}_{\bar{p}_{ij}} = sR\underbrace{(q_j - q_i)}_{\bar{q}_{ij}} + \cancel{(t - t)} + \underbrace{(o_j - o_i)}_{\bar{o}_{ij}} + \underbrace{(\epsilon_j - \epsilon_i)}_{\bar{\epsilon}_{ij}} \tag{2.35}$$

Notice that the translation is eliminated by subtraction and the relationship only depends on the scale and rotation. Hence, a TIM can be obtained by calculating $\bar{p}_{ij}$ and $\bar{q}_{ij}$ and satisfies the following model:

$$\bar{p}_{ij} = sR\bar{q}_{ij} + \bar{o}_{ij} + \bar{\epsilon}_{ij} \tag{2.36}$$

Here, $\bar{o}_{ij}$ is zero if both correspondences $\mathcal{k}_i$ and $\mathcal{k}_j$ are inliers or otherwise an arbitrary vector.

Translation and Rotation Invariant Measurements (TRIMs) are built on the concept that although relative locations of TIMs are still affected by the rotation R, their distances are not. Hence, the norm of every TIM is calculated to construct a rotation invariant form:

$$\|\bar{p}_{ij}\| = \|sR\bar{q}_{ij} + \bar{o}_{ij} + \bar{\epsilon}_{ij}\| \tag{2.37}$$

$$= \|sR\bar{q}_{ij}\| + \tilde{o}_{ij} + \tilde{\epsilon}_{ij} \tag{2.38}$$

Then, for an inlier measurement and by taking the rotation invariance of the norm into account as well as the scale being $s > 0$, the following TIM can be formulated by dividing the above term by $\|\bar{q}_{ij}\|$:

$$s_{ij} = s + o^s_{ij} + \epsilon^s_{ij} \quad \text{with} \quad s_{ij} = \frac{\|\bar{p}_{ij}\|}{\|\bar{q}_{ij}\|},\ o^s_{ij} = \frac{\tilde{o}_{ij}}{\|\bar{q}_{ij}\|} \text{ and } \epsilon^s_{ij} = \frac{\tilde{\epsilon}_{ij}}{\|\bar{q}_{ij}\|} \tag{2.39}$$

In addition, the term $\alpha_{ij} = \frac{\delta_{ij}}{\|\bar{q}_{ij}\|}$ is phrased where $|\tilde{\epsilon}_{ij}| \leq \delta_{ij}$.

Based on these measurements, the scale, the rotation and the translation can be estimated based on adjusted TLS equations. At first the scale is computed using TRIMs and the following adaption of Equation 2.34:

$$\hat{s} = \arg\min_{s} \sum_{n=1}^{N} \min(\frac{(s - s_n)^2}{\alpha_n},\ \bar{c}^2) \tag{2.40}$$

Then the rotation is determined using TIMs and the previously estimated scale:

$$\hat{R} = \arg\min_{R \in SO(3)} \sum_{n=1}^{N} \min(\frac{\|\bar{p}_n - \hat{s}R\bar{q}_n\|^2}{\delta_n^2},\ \bar{c}^2) \tag{2.41}$$

Finally, the translation is estimated:

$$\hat{t}_j = \arg\min_{t_j} \sum_{i=1}^{N} \min(\frac{(t_j - [p_i - \hat{s}\hat{R}q_i]_j)^2}{\beta_i^2},\ \bar{c}^2) \quad \text{with } j = 1, 2, 3 \tag{2.42}$$

Here, j corresponds to the j-th entry of the translation vector i.e. each component for the vector is estimated independently.

TEASER also makes use of graph theory to discard outliers [3]. In general, a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ consists of a set of vertices or points $\mathcal{V}$ and a set of edges or lines $\mathcal{E}$ where the connection between two vertices forms an edge. In the case of TEASER, the vertices constitute the correspondences and the edges are established by the TIMs and TRIMs. In a first stage, gross outliers of s_{ij} are rejected by the TLS function resulting in a pruned graph $\mathcal{G}'(\mathcal{V}, \mathcal{E}')$ where $\mathcal{E}'$ is a subset of $\mathcal{E}$. To discard even more outliers, a maximum clique inlier selection is applied in a second stage. A clique is a subset of the vertices $\mathcal{V}$ where all vertices within that subgraph are pairwise adjacent. The maximum clique inlier selection aims to find the subgraph with the greatest clique number i.e. greatest amount of vertices.

The procedure of TEASER is summarized in Algorithm 2.

Algorithm 2 TEASER

Input: correspondence set $\mathcal{K} = \{p_i,\ q_i\}$ and bounds β_i for $i = 1, ..., N$, threshold $\bar{c}^2$, graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$
Output: Estimation of the scale, rotation and translation: $\hat{s}$, $\hat{R}$, $\hat{t}$

function TEASER($\mathcal{K}$, β_i, $\bar{c}^2$, $\mathcal{G}(\mathcal{V}, \mathcal{E})$)
% Compute TIMs $\forall i, j \in \mathcal{E}$
$\bar{p}_{ij} = p_j - p_i$
$\bar{q}_{ij} = q_j - q_i$
$\bar{\delta}_{ij} = \beta_i + \beta_j$

% Compute TRIMs $\forall i, j \in \mathcal{E}$
$s_{ij} = \frac{\|\bar{p}_{ij}\|}{\|\bar{q}_{ij}\|}$
$\alpha_{ij} = \frac{\bar{\delta}_{ij}}{\|\bar{q}_{ij}\|}$

% Estimation of s
$\hat{s} =$ `estimation_of_s`$(s_{ij},\ \alpha_{ij}\colon\ \forall i, j \in \mathcal{E},\ \bar{c}^2)$

% Update graph $\mathcal{G}$
$\mathcal{G}'(\mathcal{V}, \mathcal{E}') =$ `grossOutlierRemoval`$(\mathcal{G}(\mathcal{V}, \mathcal{E}))$
$\mathcal{G}(\mathcal{V}', \mathcal{E}'') =$ `maxClique`$(\mathcal{G}'(\mathcal{V}, \mathcal{E}'))$

% Estimation of R and t
$\hat{R} =$ `estimation_of_R`$(\{\bar{p}_{ij},\ \bar{q}_{ij},\ \delta_{i,j}\colon\ \forall i, j \in \xi''\},\ \bar{c}^2,\ \hat{s})$
$\hat{t} =$ `estimation_of_t`$(\{p_i,\ q_i,\ \beta_i\colon\ i \in \mathcal{V}'\},\ \bar{c}^2,\ \hat{s},\ \hat{R})$
end function

2.3.4 Iterative Closest Point (ICP)

The Iterative Closest Point (ICP) algorithm is a popular local registration method for overlapping 3D shapes and was introduced by Besl and McKay in 1992 [45]. This method estimates correspondences between two point clouds and aligns these by computing a rigid transformation. Hence, this algorithm depends both on correspondence search and pose estimation. By now several variants of the algorithm exist, but its fundamental procedure is outlined in Algorithm 3.

Apart from the target point cloud $\mathcal{P}$ and the source point cloud $\mathcal{Q}$, the algorithm needs an initial transformation T_0 computed by a global registration method as an input [45]. In addition, two convergence criteria can be defined: the maximum error E and/or the maximum number of iterations N_{max}.

First, the correspondence set $\mathcal{K} = \{k_i\}$ with N_k correspondences is computed: for each point q in the source point cloud $\mathcal{Q}$, the closest point in the target point cloud $\mathcal{P}$ is estimated by finding the point p yielding the smallest distance:

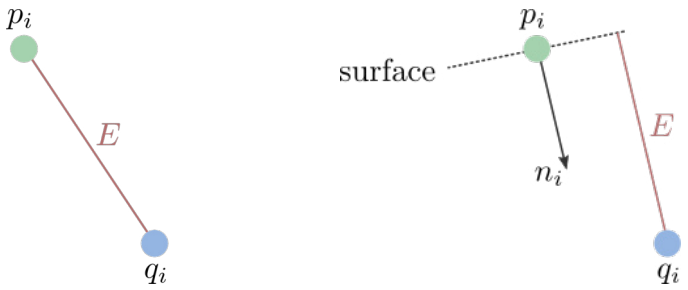
Algorithm 3 ICP

Input: Point clouds $\mathcal{Q} = \{q_i\}$ and $\mathcal{P} = \{p_j\}$, initial transformation T_0
Output: Transformed source point cloud $\mathcal{Q}$

function ICP($\mathcal{P}$, $\mathcal{Q}$, T_0)
  $T = T_0$ ▷ Transformation
  while $E > d$ || $N < N_{max}$ **do**
    for i $= 1$ **to** N_q **do**
      $k_i =$ `findClosestPoint`$(q_i, \mathcal{P})$
    end for
    $T =$ `minimizeError`$(\mathcal{K})$
    $\mathcal{Q} =$ `applyTransformation`$(\mathcal{Q}, T)$
    $N = N + 1$
  end while
end function

$$e(q, \mathcal{P}) = \min_{p \in \mathcal{P}} \|p - q\| \tag{2.43}$$

Then, by minimizing the error $E(R, t)$, the transformation is estimated. Figure 2.15 visualizes the error function for three common ICP variants.



(a) Point-to-point **(b)** Point-to-plane

Figure 2.15: Variants of the ICP method according to [45, 46]. p_i denotes the target point, n_i indicates its normal vector and q_i signifies the source point.

For the standard version of the ICP algorithm, also referred to as *point-to-point ICP*, the error function has the following form:

$$E(R, t) = \sum_{i=1}^{N_k} \|(p_i - (Rq_i + t)\|^2 \tag{2.44}$$

Another version of the ICP algorithm is the *point-to-plane ICP* which enhances performance by employing surface normal information [46, 47]. Here, the error function includes the surface normal n_i for every target point p_i:

$$E(R, t) = \sum_{i=1}^{N_k} \|(n_i(p_i - (Rq_i + t)\|^2 \tag{2.45}$$

An additional variation is the *colored-ICP* where in addition to the geometric properties, the color is incorporated [48]. Here, the error function has the following form:

$$E(R,t) = \sum_{i=1}^{N_k} \|(C_p(f((Rq_i + t) - C(q))\|^2 \tag{2.46}$$

C_p is the continuous color function defined on the tangent plane of p and the function $f(Rq_i + t) - C(q))$ projects the point q onto that tangent plane.

After estimating the transformation with one of the error functions, the result is applied to the source point cloud. Another iteration begins if the error is greater than the defined threshold or if the number of iterations N is below the previously defined maximum.

2.3.5 Multiway Registration

So far, only pairwise registration was discussed. However, in many cases more than two point clouds have to be aligned. A pose graph, as visualized in Figure 2.16, displays a registration process for multiple point clouds.

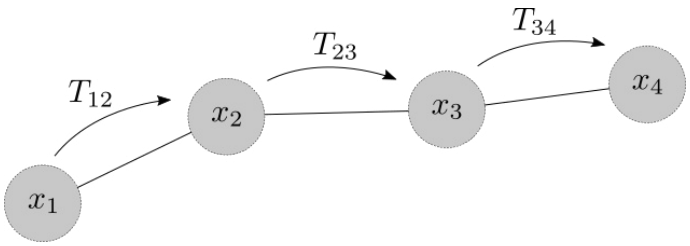


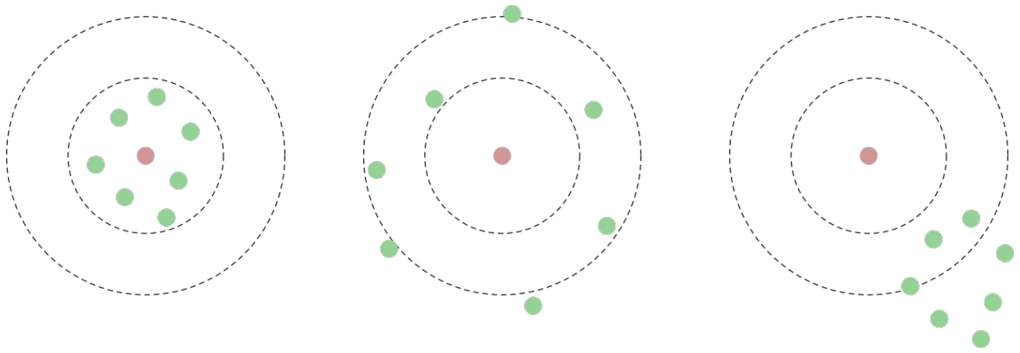
Figure 2.16: Pose graph consisting of nodes $\{x_i\}$ and edges $\{T_{ij}\}$ based on [49]

For each point cloud at node $\{x_i\}$, the edge or relative transformation $\{T_{ij}\}$ is estimated with reference to an adjacent point cloud $\{x_j\}$ [49]. Then, the position of each point cloud within the common global coordinate system is calculated by multiplying the position of the first point cloud with the subsequent relative transformations.

2.3.6 Evaluation

The evaluation of point cloud registration is predominantly based on quality measures. Two important parameters are the accuracy and precision whose difference is illustrated in Figure 2.17 [19]. Accuracy describes how close a result is in comparison to a ground truth or other measurement standard. Hence, a higher accuracy implies a higher level of agreement. Precision, on the other hand, describes the spread or variance from repeated measurements.

The evaluation of point cloud registration is preferably done in comparison to a ground truth. However, in many cases ground truth data is not given.



(a) Accurate + precise **(b)** Accurate **(c)** Precise

Figure 2.17: Assessment of measurement tasks: accuracy vs. precision according to [19]

The Root Mean Square Error (RMSE) is a widespread positional relative accuracy measure to calculate actual-target comparison e.g. the difference between measured values and the ground truth [19]:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n}} \tag{2.47}$$

where X_i describes the measured value and n is the number of data points. The smaller the value of the RMSE, the better the registration. The RMSE lends a comparatively high weight to large errors since the errors are squared before being averaged.

In the open-source library Open3D, the primary result metrices for pairwise registration are the fitness F, the amount of inlier correspondences with respect to the size of the target point cloud, and the inlier $RMSE$, the RMSE of all inlier correspondences:

$$F = \frac{N_{\text{fi}}}{N_p} \tag{2.48}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N_{\text{fi}}}(\boldsymbol{p}_i - \boldsymbol{q}_i)^2}{N_{\text{fi}}}} \tag{2.49}$$

Here, N_{fi} is the size of the inlier correspondence set and N_p the number of points in the target point cloud. For both measures applies, that in case of no inlier correspondences, the measure is set to 0. Within this thesis, the RMSE is the inlier RMSE as presented in Open3D.

In case of ground truth available as a pose graph, the euclidean distance between each estimated pose $\{x_i,\ y_i,\ z_i\}$ and the ground truth pose $\{\bar{x}_i,\ \bar{y}_i,\ \bar{z}_i\}$ is calculated:

$$d_i = \sqrt{(x_i - \bar{x}_i)^2 + (y_i - \bar{y}_i)^2 + (z_i - \bar{z}_i)^2} \tag{2.50}$$

In addition, median, mean and standard deviation are calculated to evaluate multi-way registration.

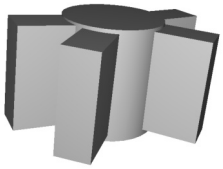
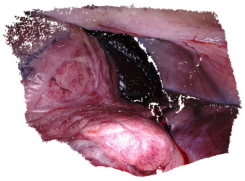
3
Methods

In this chapter, the methods of the project are outlined. This includes an overview of the hardware, the software as well as objects and reference data. In addition, the specifications for the registration algorithms are defined and the testing configurations and pipelines are specified.

3.1 Point Cloud Acquisition and Reference Data

A number of different objects and reference data are used to evaluate the registration pipeline, as listed in Table 3.1.

Table 3.1: Objects and reference data

Company/Project	Type	Figure	Ground truth
Fraunhofer HHI	Specimen		CAD model
SCARED Challenge	Point clouds		Pose graph
PHACON	Throat assistent		CT model

The data from the Stereo Correspondence and Reconstruction of Endoscopic Data (SCARED) challenge was acquired by Allan et al. by employing a da Vinci Xi surgical robot where each stereo pair was captured using an endoscope [18]. Ground truth

data is available as pose graphs and in the context of the challenge, Rosenthal et al. computed point clouds from stereo pairs which are aimed to be registered (see [18]).

The hardware of this project includes three digital laparoscopes/endoscopes with their respective camera control units which are specified in Table 3.2.

Table 3.2: Endoscope systems

Company	Endoscope	Light source	Application
AESCULAP	EinsteinVision 2.0	EinsteinVision 2.0	Laparoscopy
AESCULAP	EinsteinVision 3.0	-	Laparoscopy
Xion	EndoSURGERY 3D Spectar	MATRIX LED Duo	Endoscopy, microsopy

Note that the available Einstein Vision 3.0 does not possess a light source. Instead, only the external surgical light marLED® by KLSmartin is utilized. Additional hardware includes the RODEON TurnTable manufactured by Dr. Clauß Bild- und Datentechnik GmbH which is controlled using the RODEONmdodular software.

The calibration of the endoscopes was previously conducted. Hence, the internal and external camera parameters are known. For the lens distortion, an accuracy of three radial distortion parameters was selected. For the EndoSURGERY 3D Spectar by Xion, an additional calibration has to be performed before every application. This is conducted using the XION 3D ALIGN, an external tool consisting of a plane with a dot matrix and an attachment mount for the endoscope. Figure 3.1 displays the calibration setup for this system.

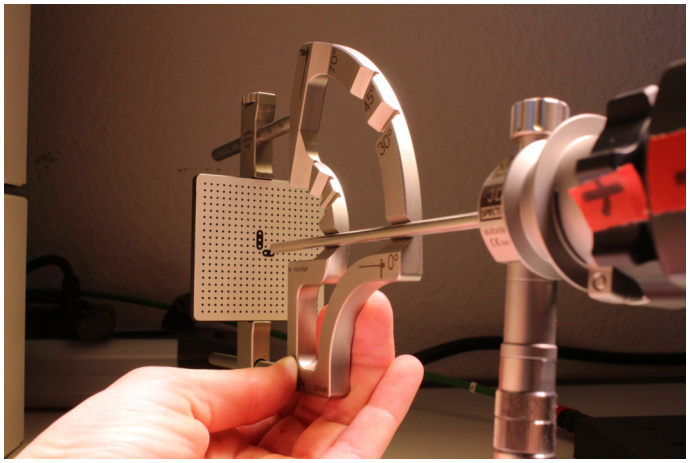


Figure 3.1: Calibration of the EndoSURGERY 3D Spectar endoscope by Xion.

For the acquisition of point clouds, the STereoscopic ANalyzer (STAN) is utilized. The app is an image analysis software displaying stereo images in real-time developed

by Fraunhofer HHI [29, 50]. Within a stereo pair, the software determines SKB features in each image and displays them color-coded corresponding to the object distance. A rectification process is applied transforming the right image to be in plane with the left. In addition, the disparity map is displayed and if applicable smoothing options in form of median and/or bilateral filters are available. Point clouds can be stored along with the respective stereo image pair.

To remove background that should not be stored as part of the point cloud, a chroma keying feature is implemented in STAN. The implementation converts the RGB color of each pixel into the HSV color space. The HSV color space represents colors by hue, saturation and value and is therefore more suitable for this application [51]. When saving a point cloud, pixels which are in the defined HSV range, i.e. with a hue between 40 and 81, a saturation between 51 and 255 and a value between 51 and 255, are not stored, mainly rejecting green color information e.g. green screen.

3.2 Point Cloud Analysis and Registration

The point cloud analysis and registration pipeline is implemented using C++14. The implementation is predominantly based on Open3D version 14, an open-source library for 3D data processing and visualization [33]. Open3D is installed from source with CUDA 11.3 support for parallel computing on the GPU. Additionally, TEASER++ is installed from source for the employment of the global registration method TEASER [3]. The implementation is executed on a Dell Precision T3600 work station with an NVIDIA GeForce GTX 980 graphics card and 32GB RAM.

An overview of the analysis and registration process including the available options is given in Figure 3.2.

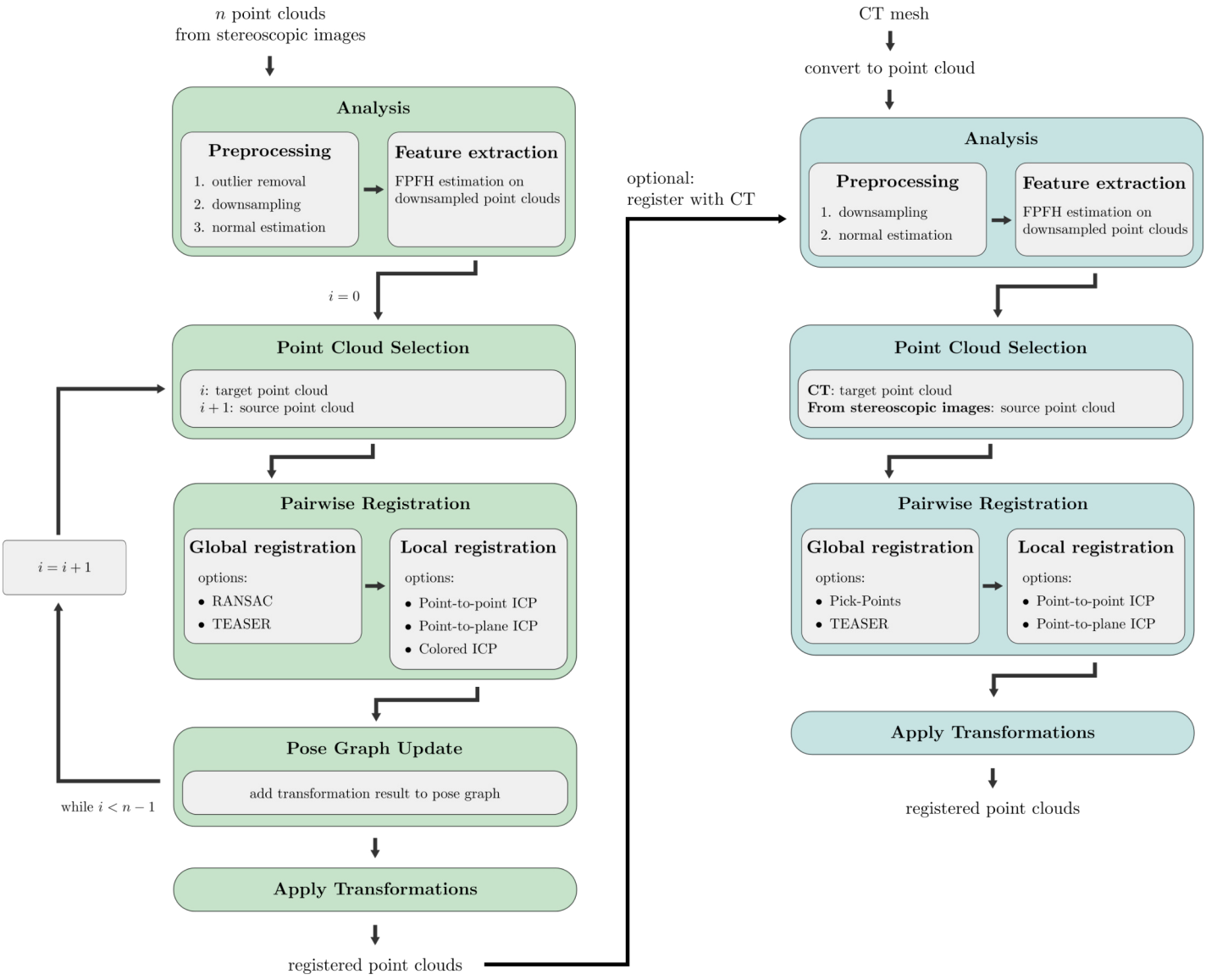


Figure 3.2: Overview of the point cloud registration pipeline

In a first step, the analysis of the point clouds acquired by stereoscopic endoscpic/microscopic images is carried out. This includes the preprocessing of point clouds, namely outlier removal and downsampling as well as normal estimation conducted both on the input point clouds and the downsampled clouds. Additionally, feature extraction based on FPFHs is performed on the downsampled point clouds.

In a subsequent step, multiway registration is conducted: each neighboring two point clouds are registered by applying pairwise registration. Here, several options for global and local registration methods are available of which all estimate a similarity transformation. Each registration result is saved in a pose graph from which the transformations are applied after iterating through all point clouds.

In an additional optional step, the registered stereoscopic endoscopic/microscopic are registered with a reconstructed 3D CT model. Here, point clouds are first down sampled and normal as well as FPFH estimation is conducted. Then, pairwise global and local registrations are performed with one of the listed options and finally, the resulting transformation is applied. The following sections specify the parameters within each step.

3.2.1 Analysis

As stated, the analysis includes the removal of outliers, downsampling and normal estimation for all point clouds and feature extraction for downsampled clouds. Table 3.3 presents the parameters for the preprocessing step with the chosen values.

Table 3.3: Parameters for Preprocessing

Preprocessing Step	**Parameter**	**Value**
Outlier removal	`nb_neighbors`	30
	`std_ratio`	2.0
Downsampling	`voxel_size`	0.1 ... 1.0
Normal estimation	`max_nn`	30
Feature extraction	`max_nn`	30

Outlier removal is conducted by applying a statistical based approach implemented by Open3D. The statistical outlier removal deletes points that are on average farther distant from their neighbors. Here, the number of neighbors and the standard deviation ratio have to be set.

For downsampling the voxel size, i.e. the size of each 3D point, has to be chosen in accordance with the sensor specifications (here: in mm). A small voxel size corresponds to a high resolution point cloud while a large voxel size results in a low resolution point cloud. Defining the voxel size is a trade-off between computation time and the accuracy of the registration. Additionally, the RAM may restrict the

process to a larger voxel size due to memory shortage.

Normal estimation and feature extraction are conducted by taking a maximum number of nearest neighbors into account.

3.2.2 Global Registration

The global registration includes an implementation for the RANSAC and TEASER algorithms. Both methods make use of downsampled point clouds, the estimation of normals and features. For TEASER, the correspondences are computed as well.

For RANSAC, Open3D provides the function `RegistrationRANSACBasedOnFeature-Matching` which first estimates correspondences based on FPFHs and then applies the RANSAC algorithm [33]. The function takes the source and target point clouds as an input as well as the corresponding FPFHs and the parameters listed in Table 3.4 are set.

Table 3.4: Parameters for RANSAC

	Parameter	**Value**
	`mutual_filter`	true
	`distance_threshold`	1 ... 10
	`TransformationEstimation`	`PointToPoint`
`RANSACConvergenceCriteria`	`max_iterations`	100
	`max_validation`	0.999

The mutual filter applies a reverse check for correspondence estimation to make sure that correspondences from source to target are also valid when estimated from target to source. The distance threshold is set to in accordance with the sensor specifications (here: in mm). For the convergence criteria, the maximum number of iterations and the maximum validation have to be set.

The TEASER registration takes the calculated correspondences as an input (see Algorithm 2) and the parameters specified in Table 3.5 are set. The value for the noise bound is defined and as suggested by the authors (see Yang et al. [3]), the value for $\bar{c}^2$ is set to 1. Since the point clouds vary in scale, scale estimation is conducted. For the rotation estimation, the number of iterations, the GNC factor, the type of estimation algorithm and the cost threshold are specified. Since this methods estimates scaling, rotation and translation subsequently, the transformation is computed according to Equation 2.26.

Table 3.5: Parameters for TEASER

	Parameter	Value
General	`noise_bound`	0.5 ... 2
	`cbar2`	1
	`estimate_scaling`	true
Rotation	`rotation_max_iterations`	1000
	`rotation_gnc_factor`	1.4
	`rotation_estimation_algorithm`	`GNC_TLS`
	`rotation_cost_threshold`	1000

For the registration with the CT model, a semi-automatic approach was implemented where the user selects correspondences between the point clouds. Figure 3.3 displays the interactive selection of five features in the CT mesh. To register a point cloud with the CT, the user has to select points within the acquired point cloud close to the location within the mesh.

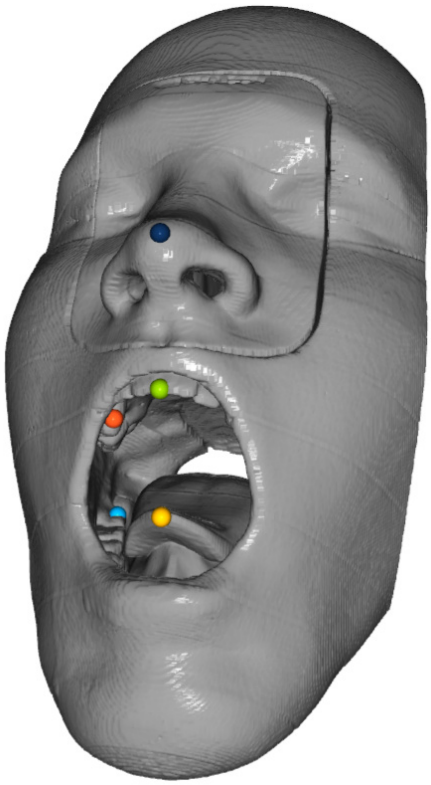


Figure 3.3: Selection of correspondences by the user

3.2.3 Local Registration

For local registration, one of the three ICP variants may be selected to refine the global registration result: point-to-point, point-to-plane or colored ICP. All functions take the source and target point clouds as an input. However, point-to-plane and colored ICP additionally require the estimation of normal vectors in advance. Table 3.6 summarizes the parameters for this registration. Although the error functions differ between the ICP variants, the same threshold and convergence criteria can be applied. Here, the distance threshold is specified as the value of the voxel size.

Table 3.6: Parameters for ICP

	Parameter	Value
	`distance_threshold`	0.25
`ICPConvergenceCriteria`	`max_iteration`	100
	`relative_fitness`	1e-6
	`relative_rmse`	1e-6

3.3 Test Setup

Three tests are conducted to evaluate the registration process and to compare its varying methods. These tests are classified based on the available objects and reference data listed in Table 3.1.

3.3.1 Test 1: Specimen

The specimen is placed approximately at the centre of the RODEON TurnTable. Point clouds are acquired using the EinsteinVision 2.0 endoscope system mounted at a fixed position 55mm far away from the centre of the turning table. A green cloth placed underneath the specimen, the external light source and the application of the chroma keying feature ensure that under- and background are not captured. In addition, the point clouds are clipped to a distance between 15 and 70mm when saved to preserve as little outliers as possible (see Figure 3.4). The turning table is rotated with an angle difference of 5° for 90° resulting in 18 captured point clouds.

As for the registration, both global registration methods are applied for a comparison. Local registration and the registration with the CAD model is not applied. For downsampling a voxel size of 0.5mm is utilized and the distance threshold and noise bound are set to 5mm and 1mm respectively.

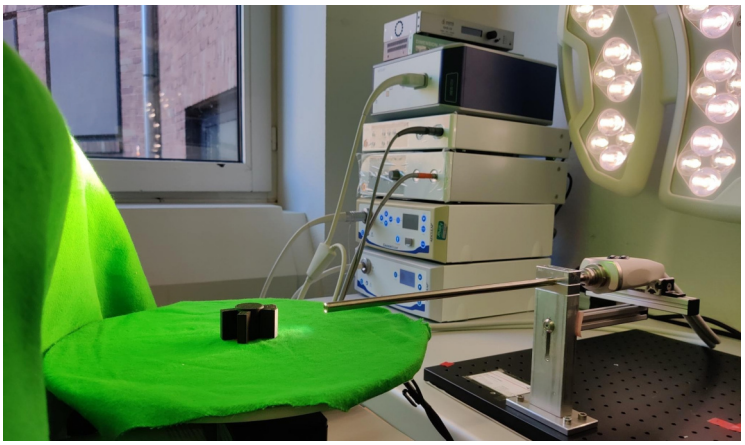


Figure 3.4: Setup for acquiring point clouds of the specimen utilizing the RODEON TurnTable and the EinsteinVision 2.0 endoscope

3.3.2 Test 2: SCARED Challenge Data

From the SCARED data challenge the selected datasets listed in Table 3.7 are registered.

Table 3.7: Selected SCARED Challenge Test Data

Dataset	Number of Frames	Point clouds (first and last frame)
1	10	
2	9	
3	15	

The selected three datasets correspond to a selection of frames from the following datasets of the SCARED challenge:

- **Dataset 1**: Training, Dataset 1, Keyframe 1, every 20th frame
- **Dataset 2**: Training, Dataset 3, Keyframe 3, every 50th frame
- **Dataset 3**: Challenge, Dataset 9, Keyframe 1, every 50th frame + 3 frames

For dataset 3, three additional frames are used apart from selecting every 50th frame to minimize great changes in movement.

For these datasets a voxel size of 1mm is used to downsample the point clouds. For dataset 1, a distance threshold of 1mm to 10mm is utilized for RANSAC while the noise bound for TEASER is set to 0.25mm to 1mm. For the other datasets, the distance threshold and the noise bound have values of 10mm and 1mm respectively. All three local registration methods are applied and compared. The registration with a CT is, however, not conducted since CT data is unavailable for this dataset.

3.3.3 Test 3: Throat Assistant

In this setup, the throat assistant is placed on the TurnTable while point clouds are acquired with the EinsteinVision 3.0 or the EndoSURGERY 3D Spectar system. To remove any background with the usage of the chroma keying feature, the throat assistant is positioned on a green cloth in addition to a green background.

For the capturing using the EinsteinVision 3.0, the endoscope is placed 35mm far away from the tip of the nose of the throat assistant and the point clouds clipping parameter is specified as 15mm for near plane and 70mm for far plane. The turn table is rotated for a total of 30°.

The EndoSURGERY 3D Spectar system is placed 10mm far away from the throat assistant and points that are not within range of 5 and 60mm are discarded. Here, the turn table is rotated for 20°.

The captured point clouds are inserted into the analysis and registration pipeline. The voxel size is set to 0.5mm. To register the endoscopic point clouds, TEASER with a noise bound of 1mm is used a global registration method while local registration is done with point-to-plane ICP. The registration result is then automatically registered with the CT model utilizing a voxel size of 1mm and with a noise bound of 2mm for TEASER. Additionally, the semi-automatic approach where the user has to select correspondences is utilized. Point-to-plane ICP is performed on the both results.

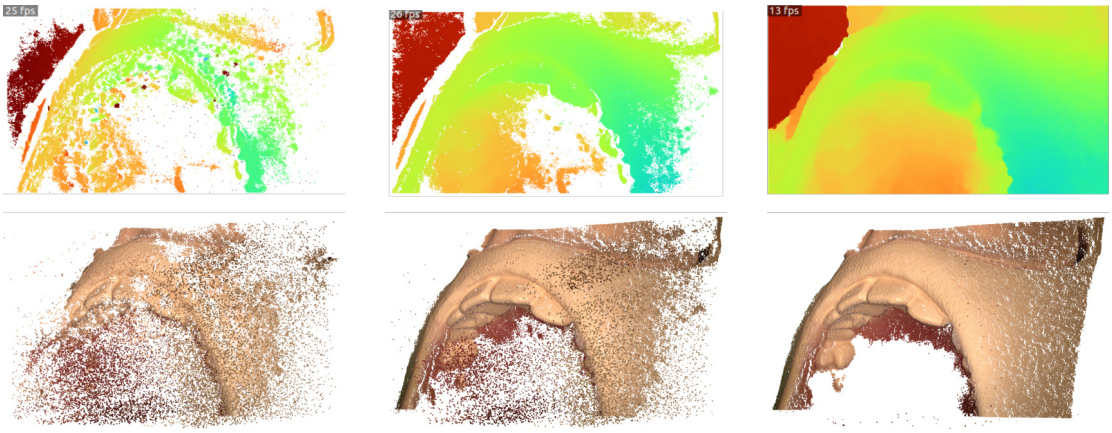
4

Results

Within this section, the results are outlined based on the previously described methods. For point cloud acquisition and point cloud analysis, exemplary results for a selection of point clouds is given. Then, the point cloud registration algorithms are evaluated based on the test setups.

4.1 Point Cloud Acquisition

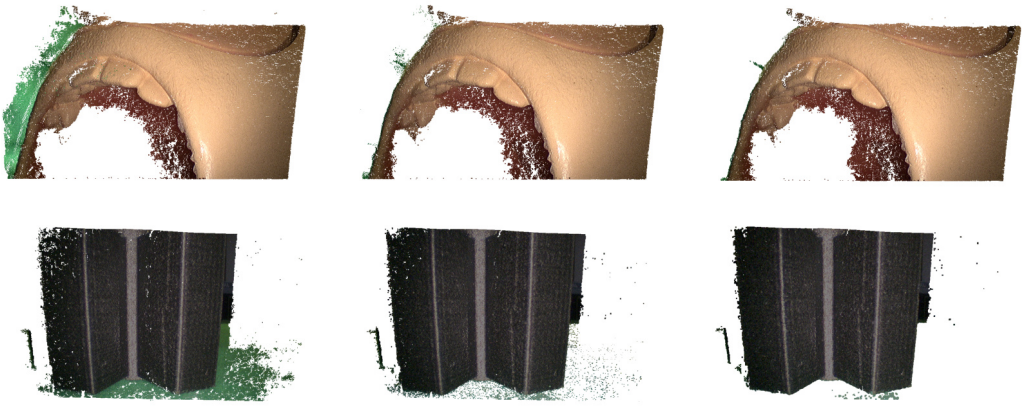
Within point cloud acquisition, the rectification of the stereo images and the application of the chroma keying feature are essential for an acceptable registration. Figure 4.1 displays the disparity map before and after rectification as well as after additionally applying median and bilateral filters. In the disparity map, blue signifies that an area or object is close to the camera while red indicates that it is further away. Before applying rectification, the disparity is not filled and appears noisy resulting in a distorted point cloud. The disparity map and the point cloud appear less distorted after applying rectification and the result improves after additionally applying median and bilateral filtering.



(a) Before **(b)** Rectified **(c)** Rectified + filtering

Figure 4.1: Example of disparity maps and point clouds before and after applying rectification and filtering. Each pair is displayed **(a)** before applying any processing steps, **(b)** after rectification and **(c)** after the additional application of median and bilateral filtering.

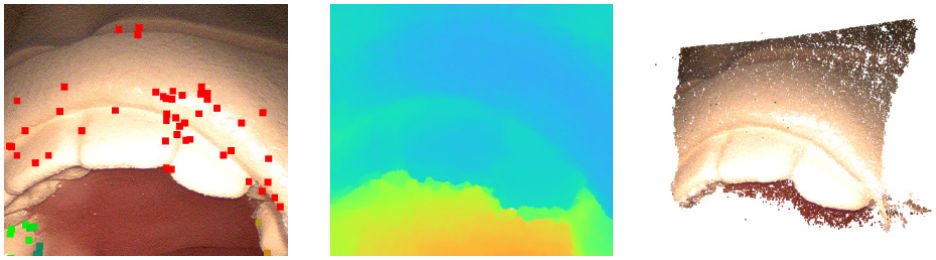
Figure 4.2 displays the removal of the background using the chroma keying feature both for the specimen and the throat assistant. Note that the point clouds where not captured at exactly the same time resulting in slightly differing clouds apart from the background removal.



(a) Original point cloud **(b)** Chroma keying **(c)** Chroma keying + median filter

Figure 4.2: Example of point clouds before and after chroma keying. Point clouds are shown **(a)** before chroma keying, **(b)** after chroma keying and **(c)** after the additional application of a median filter.

In Figure 4.3, the acquisition process for a stereo pair with few SKB features (specifically at the front teeth) is outlined. In the disparity map, the depth of the front teeth is represented by a blue hue, the edges are signified with a green hue and the palate ranges from yellow to orange. Although the palate is farther away as the tooth at the bottom right of the image, regions near the front teeth are represented by a green hue within the disparity map. As a result, red points are in plane with the front teeth in the point cloud although in reality, the palate is further to the back.

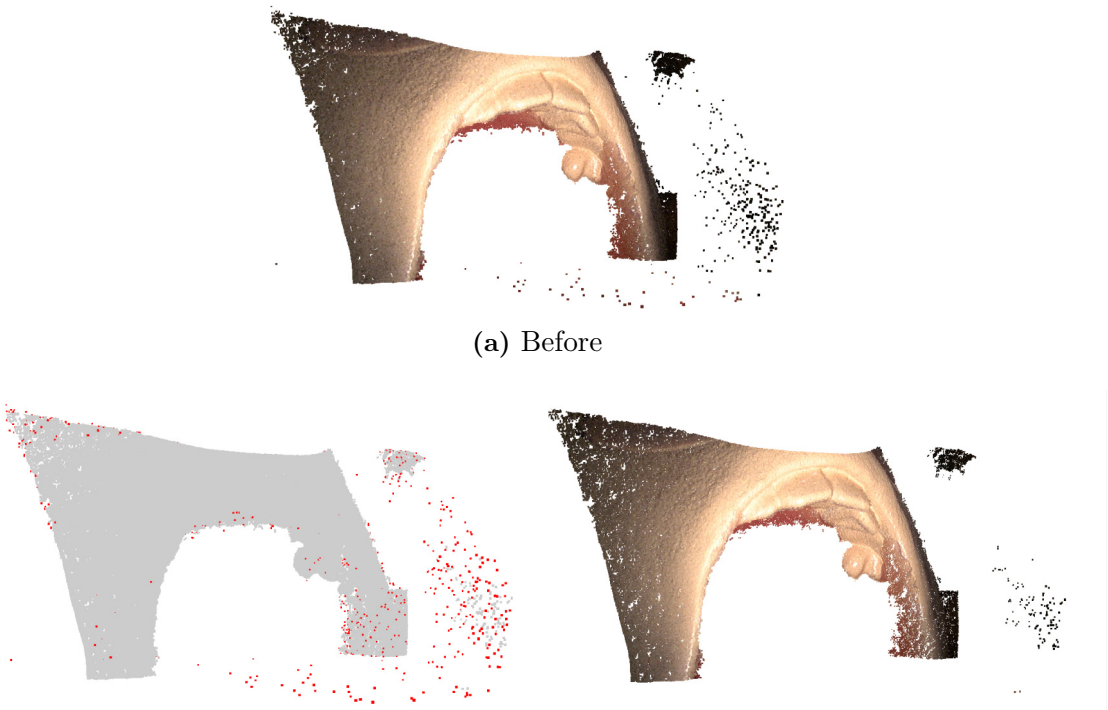


(a) Feature points **(b)** Disparity map **(c)** Point cloud

Figure 4.3: Example of a point cloud with few SKB features at sharp edges. Comparison between the **(a)** left image of the stereo pair with feature points, **(b)** the corresponding disparity map and **(c)** the resulting point cloud. Note that in **(c)**, background points where removed for visualization.

4.2 Point Cloud Analysis

Within point cloud analysis, removal of outliers, downsampling and normal estimation are presented. Outlier removal on a point cloud of the throat assistant is displayed in Figure 4.4. In 4.4b outliers are displayed in red while inliers remain grey.



(a) Before

(b) Removal

(c) After

Figure 4.4: Example of outlier removal on a selected point cloud. The point cloud **(a)** before and **(c)** after outlier removal. In **(b)** outliers are highlighted in red while inliers are gray.

Downsampling for different voxel sizes is presented in Figure 4.5. On the left hand side, a point cloud from the throat assistant captured by the EinsteinVision 2.0 endoscope is visualized while the point cloud on the right hand side corresponds to the first frame of the SCARED Dataset 1. Note that the point size for **(c)** and **(d)** was increased for better visualization.

The normal estimation for each point incorporating 30 nearest neighbors on a downsampled point cloud with a voxel size of 0.5 is given in Figure 4.6.

(a) Original point clouds

(b) voxel size: 0.1

(c) voxel size: 0.5

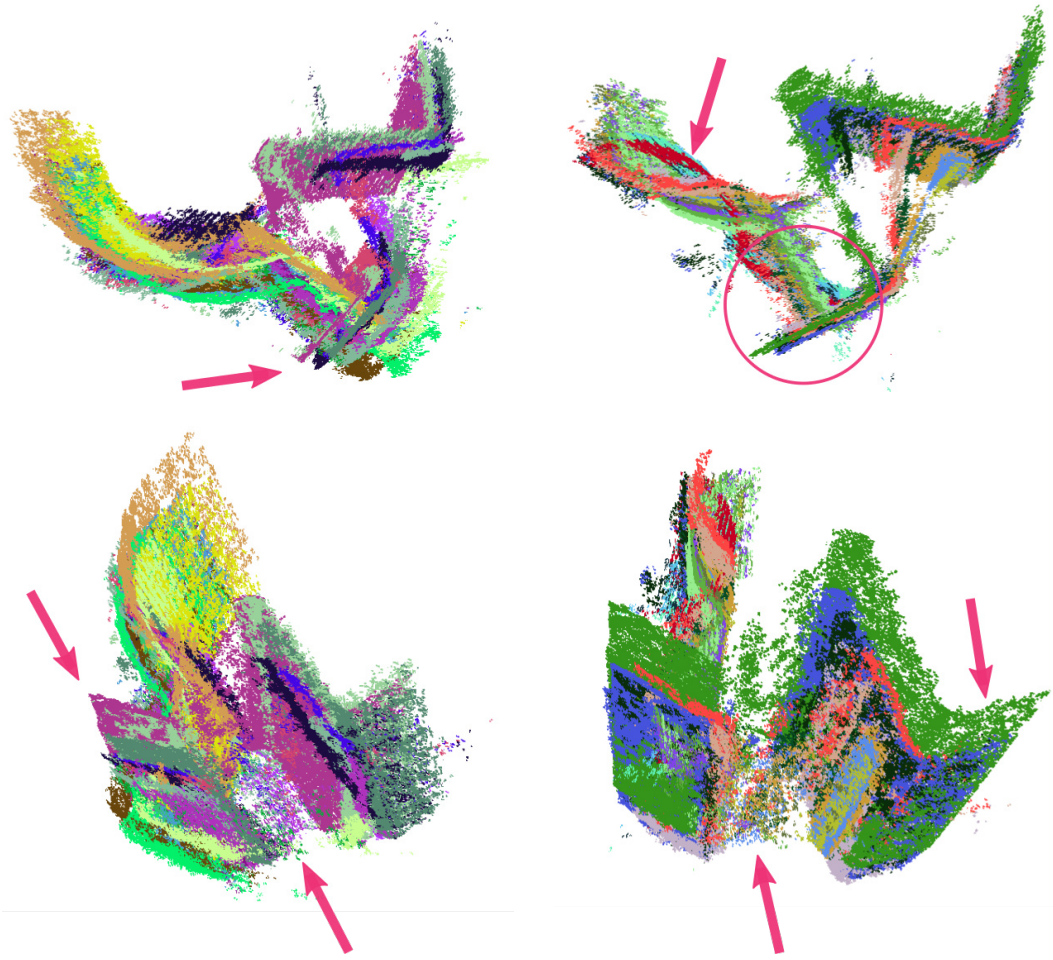
(d) voxel size: 1.0

Figure 4.5: Example of downsampled point clouds with different voxel sizes. The original point clouds in **(a)** were downsampled with voxel sizes between 0.1 and 1.0mm as visualized in **(b)** - **(d)**.

Figure 4.6: Normal estimation using 30 nearest neighbors on a downsampled point cloud with a voxel size of 0.5

4.3 Test 1: Specimen

The result for the specimen using both global registration methods are visualized in Figure 4.7 from two different viewpoints. The point clouds are colored uniformly for better visualization and the pink arrows and circles specify visible inaccuracies. For both methods, the point clouds capturing the curved side of the object face the same direction. However, the red point cloud in TEASER is flipped by approximately 180°. In addition, both methods show a visible offset at the edges and planes are shifted lengthwise. In the lower images, the bottom arrows signify areas where the density of points is less compared to other parts of the registered point cloud.



(a) RANSAC **(b)** TEASER

Figure 4.7: Point clouds after global registration for the specimen using **(a)** RANSAC and **(b)** TEASER. The pink arrows indicate discrepancies to the original object.

Figure 4.8 visualizes the corresponding pose graphs. It is apparent that the registration algorithms do not yield the same transformations. While the poses with RANSAC proceed primarily in one direction, the poses with TEASER vary greater in direction with abrupt changes.

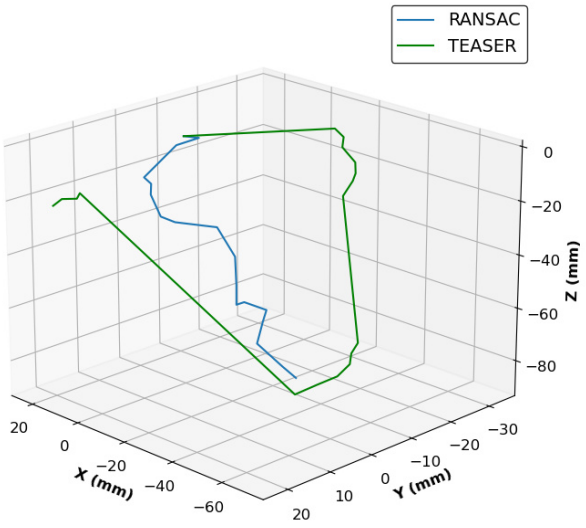


Figure 4.8: Pose graph after global registration using RANSAC (blue) and TEASER (green) for the specimen.

4.4 Test 2: SCARED Challenge

Within this section, the results for the selected datasets of the SCARED challenge are presented. For each of the three datasets, the utilized frames are displayed and results after the application of selected registration methods are shown. For dataset 1, the results for different distance thresholds are outlined additionally.

4.4.1 Dataset 1

Dataset 1 consists of ten point clouds of which the corresponding left frames are visualized in Figure 4.9.

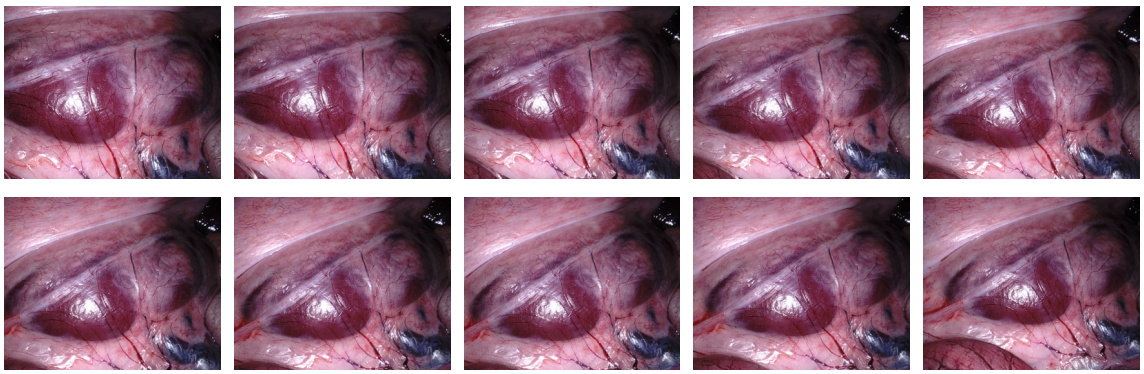
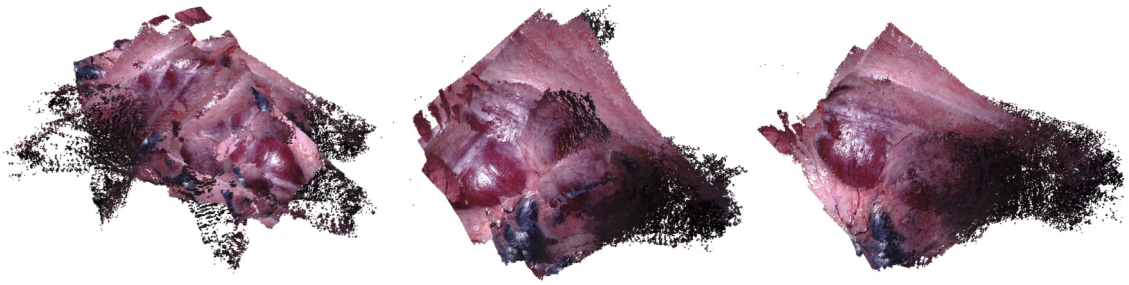


Figure 4.9: Left images of SCARED Dataset 1. The first frame is at the upper left while the last frame is at the lower right.

RANSAC

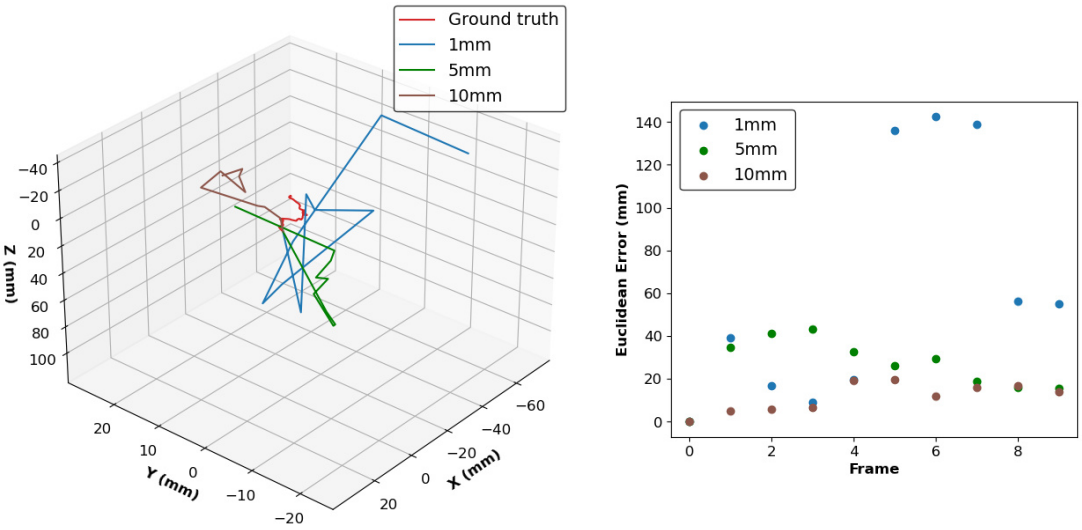
The registered point clouds following the application of RANSAC for selected distance thresholds are displayed in Figure 4.10. While the registration for a threshold of 1mm clearly does not find an acceptable transformation, the alignment is more accurate with greater thresholds.



(a) 1mm **(b)** 5mm **(c)** 10mm

Figure 4.10: Point clouds after RANSAC registration with varying distance thresholds for SCARED Dataset 1. The registration was conducted using a threshold of **(a)** 1mm, **(b)** 5mm and **(c)** 10mm.

Figure 4.11 displays the pose graphs and euclidean distances for the selected distance thresholds in comparison with the ground truth. None of the pose graphs lead into the same direction as the ground truth and pose changes occur abruptly. A distance threshold of 1mm results in the greatest euclidean distance for most poses (5 of 9 poses), while a threshold of 10mm results in the lowest values (7 of 9 poses).



(a) Pose graph **(b)** Euclidean distance

Figure 4.11: Pose graphs and euclidean distances in comparison with the ground truth after RANSAC registration with varying distance thresholds for SCARED Dataset 1. Registration was performed with a threshold of 1mm, 5mm and 10mm.

Table 4.1 outlines the statistical analysis for the RMSE, fitness and euclidean distance depending on the distance thresholds. Note, that the RMSE and fitness depend only on the estimated inliers and not the entire point clouds.

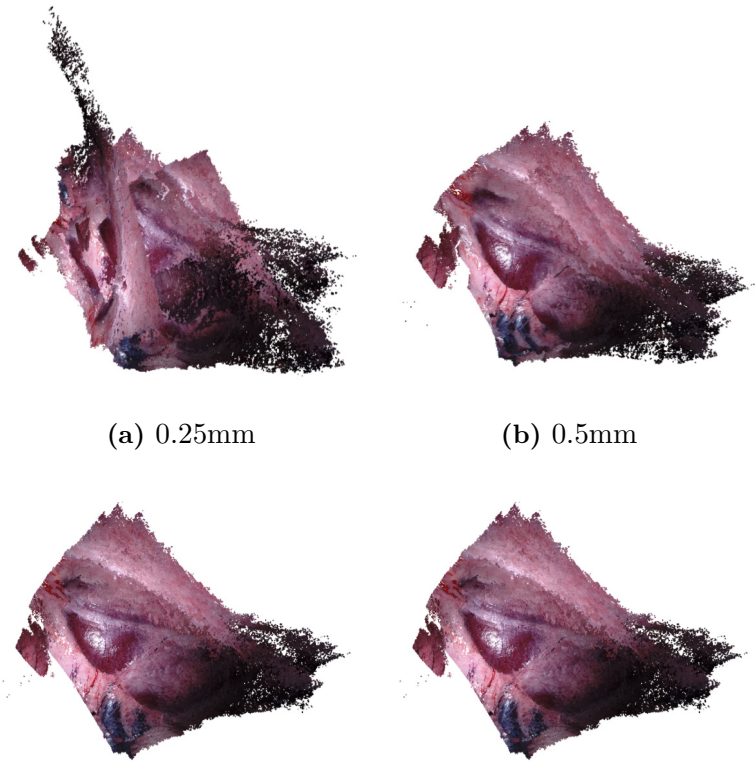
Table 4.1: Statistical analysis of the RMSE, fitness and euclidean distance after RANSAC registration with varying thresholds for SCARED Dataset 1. The euclidean distance is given in mm.

		1mm	**5mm**	**10mm**
RMSE	**median**	0.650	2.021	2.306
	average	0.648 ± 0.012	1.947 ± 0.435	2.510 ± 0.587
Fitness	**median**	0.133	0.962	0.999
	average	0.157 ± 0.095	0.898 ± 0.164	0.998 ± 0.004
Euclidean distance	**median**	46.995	28.101	12.799
	average	61.407 ± 53.45	26.178 ± 12.229	11.721 ± 5.946

While the RMSE has its lowest values at a distance threshold of 1mm, the fitness has its highest and therefore more optimal value with a threshold of 10mm. The euclidean distance has its lowest median and average values at 10mm.

TEASER

The registered point clouds following the application of TEASER for selected noise bounds are displayed in Figure 4.12. While at least one of the point clouds from the dataset is misaligned when applying a noise bound of 0.25mm, the registrations for the other selected noise bounds do not show striking aberrations although the registration results differ.

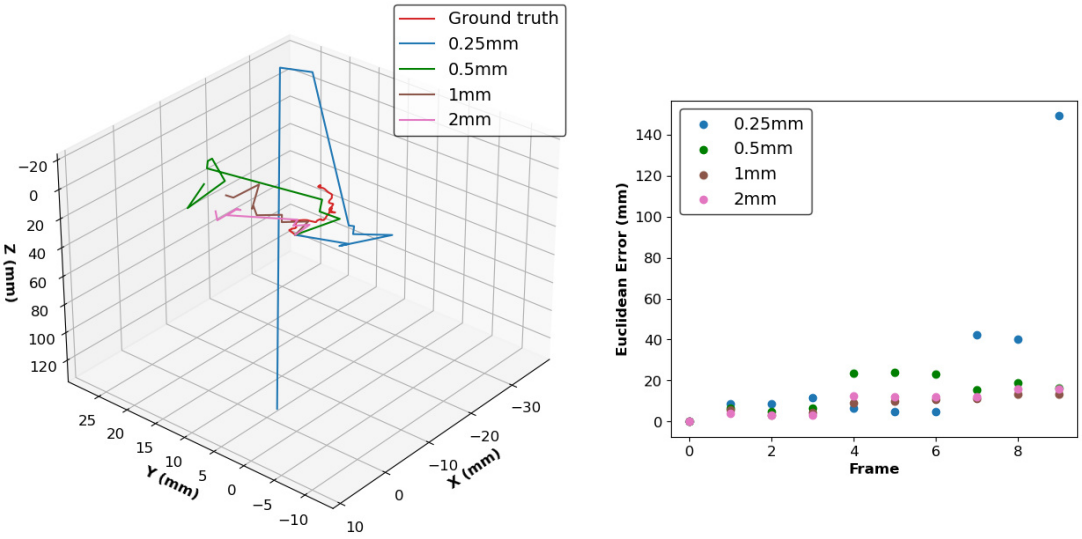


(a) 0.25mm **(b)** 0.5mm

(c) 1mm **(d)** 2mm

Figure 4.12: Point clouds after TEASER registration with varying noise bounds for SCARED Dataset 1. The results for noise bounds of **(a)** 0.25mm, **(b)** 0.5mm, **(c)** 1mm and **(d)** 2mm are displayed.

The corresponding pose graphs along with the euclidean distances are visualized in Figure 4.13. While with smaller noise bounds, the resulting pose graphs lead away from the ground truth and show drastic changes in direction, the pose graphs for noise bounds of 1mm and 2mm demonstrate smaller changes in direction. However, these noise bounds still visibly differ from the ground truth. For a noise bound of 0.25mm, the euclidean distance of the last registration is considerably larger than for the other poses exceeding an euclidean distance of 140mm.



(a) Pose graph (b) Euclidean distance

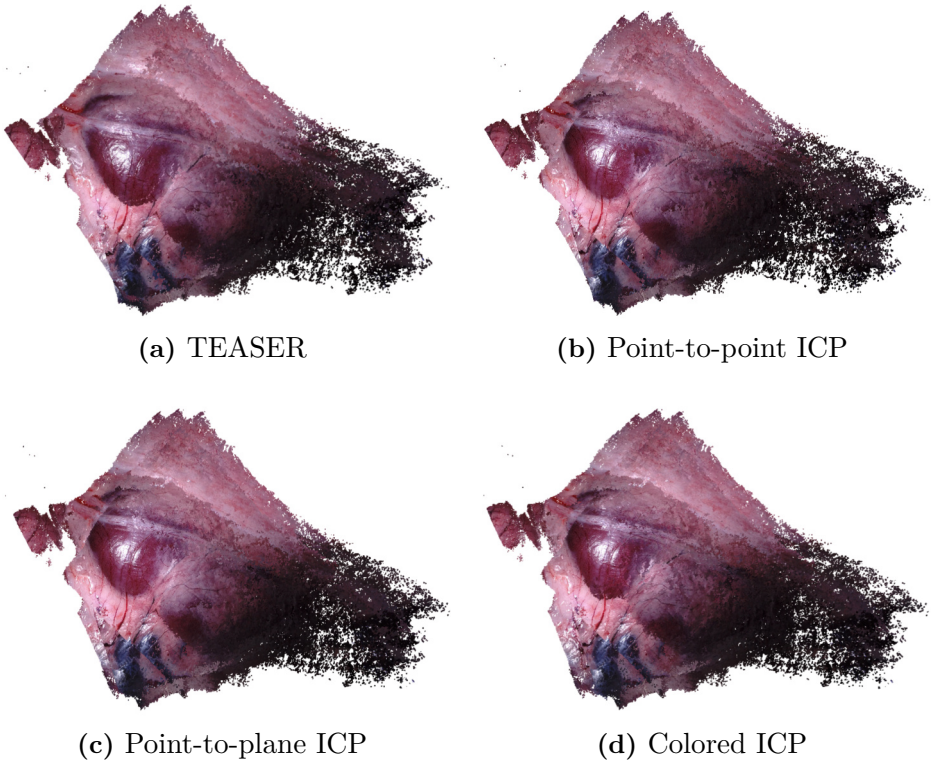
Figure 4.13: Pose graphs and euclidean distances in comparison with the ground truth after TEASER registration with varying noise bounds for SCARED Dataset 1. The noise bounds include distances of 0.25mm, 0.5mm, 1mm and 2mm.

The statistical analysis for the maximum clique and the euclidean distance are given in Table 4.2. The maximum clique increases with a greater distance of the noise bound. The euclidean distance, shows a minimal median at 0.25mm while the average is the smallest when using a bound of 1mm.

Table 4.2: Statistical analysis of the maximum clique and euclidean distance after TEASER registration with varying noise bounds for SCARED Dataset 1. The euclidean distance is given in mm.

		0.25mm	**0.5mm**	**1mm**	**2mm**
Maximum clique	**median**	8	14	29	71
	mean	7.889 ± 0.928	14.333 ± 2.646	29.444 ± 5.028	72 ± 12.207
Euclidean distance	**median**	8.617	15.786	9.555	12.111
	mean	27.668 ± 42.913	13.994 ± 8.393	8.070 ± 4.290	9.085 ± 5.613

The point clouds after the registration process employing TEASER with a noise bound of 1mm and the differing ICP variants are displayed in Figure 4.14. Visually, no striking differences between the methods can be detected.



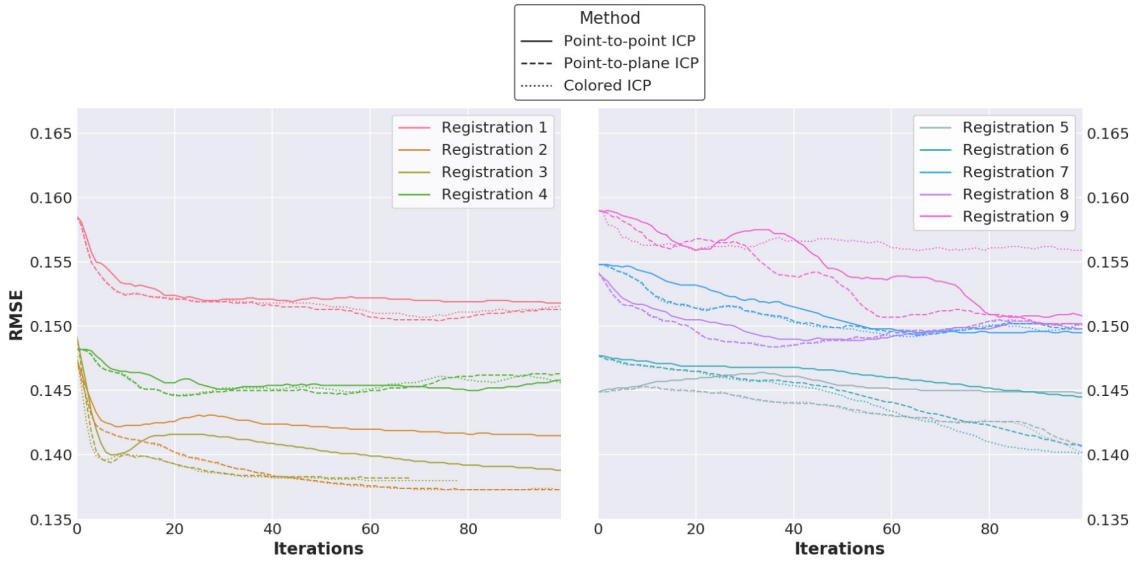
(a) TEASER **(b)** Point-to-point ICP

(c) Point-to-plane ICP **(d)** Colored ICP

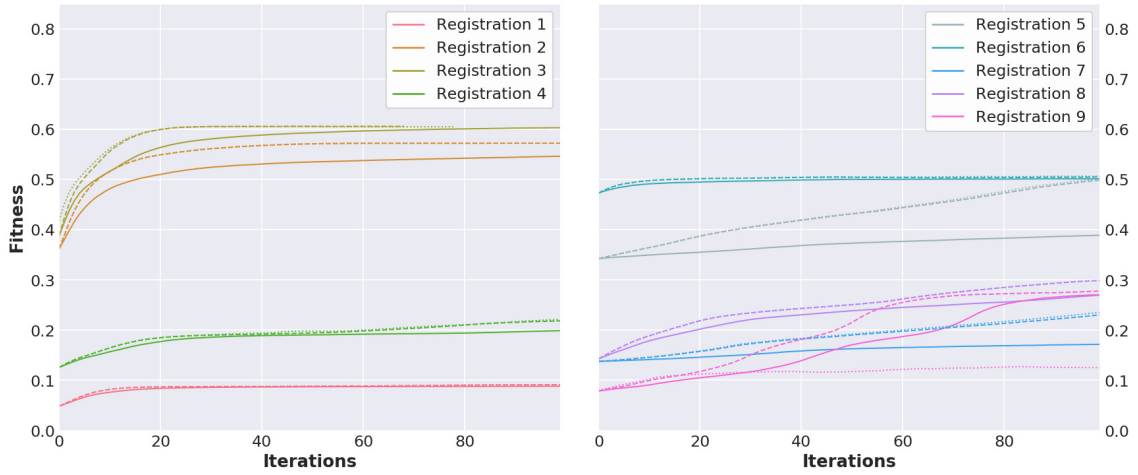
Figure 4.14: Point clouds after TEASER registration with additionally applying each of the ICP variants for SCARED Dataset 1. The point clouds after global registration with **(a)** TEASER and the additional application of **(b)** point-to-point ICP, **(c)** point-to-plane ICP and **(d)** colored ICP are displayed.

Figure 4.15 presents the RMSE and fitness for the three ICP variants for each of the nine registrations. For registration 3, point-to-plane ICP terminates after 69 iterations and colored ICP after 79 iterations. For the RMSE of all registrations, point-to-plane ICP and colored ICP converge faster within the first 20 iterations than point-to-point ICP. However, in some cases and for instance for registration 4, the RMSE increases again after 65 iterations. The fitness increases for all registrations and local registration methods.

The final values for the RMSE and fitness for the ICP variants are outlined in Table 4.3.



(a) RMSE



(b) Fitness

Figure 4.15: RMSE and Fitness for the ICP variants depending on the iterations for the SCARED Dataset 1. In beforehand, TEASER was applied as the global registration method. The differing line types indicate the local registration method while each color signifies the registration.

Table 4.3: Statistical analysis of the RMSE and fitness for the local registration for SCARED Dataset 1. Measurements are given in mm.

		point-to-point ICP	**point-to-plane ICP**	**colored ICP**
RMSE	**median**	0.146	0.146	0.146
	mean	0.146 ± 0.005	0.145 ± 0.006	0.145 ± 0.007
Fitness	**median**	0.271	0.299	0.30
	mean	0.338 ± 0.181	0.367 ± 0.182	0.351 ± 0.197

4.4.2 Dataset 2

Dataset 2 consists of nine point clouds of which the corresponding left frames are visualized in Figure 4.16.

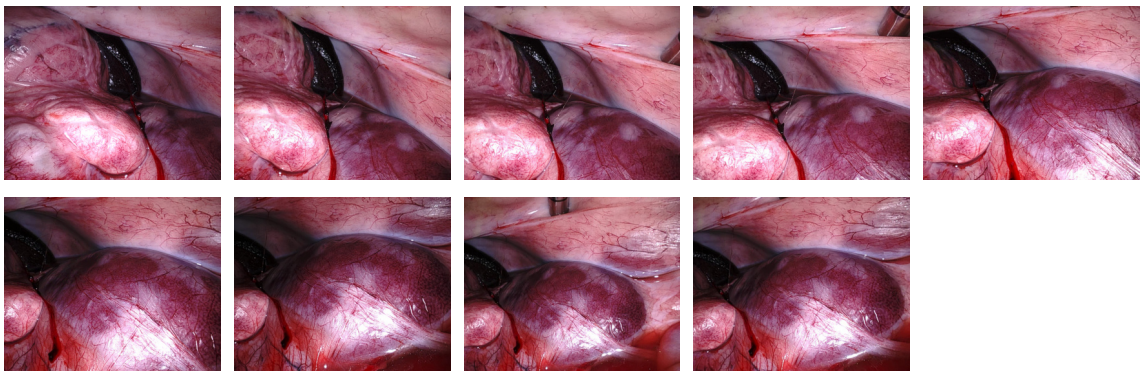
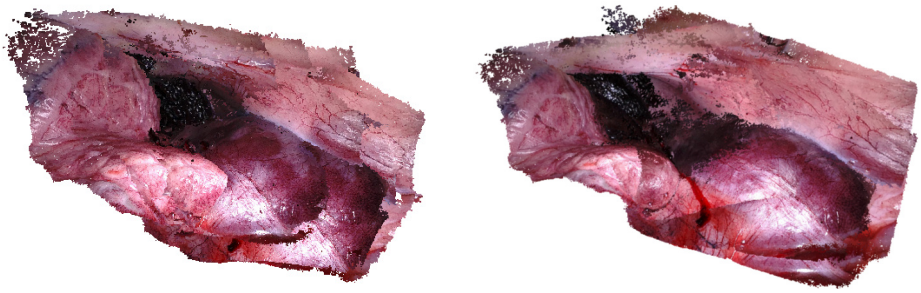


Figure 4.16: Left images of SCARED Dataset 2. The upper left image corresponds to the first frame while the lower right is the last frame.

The point clouds after employing each of the two global registration methods are shown in Figure 4.17. With RANSAC, the result is distorted while with TEASER, the result appears cleaner.



(a) RANSAC **(b)** TEASER

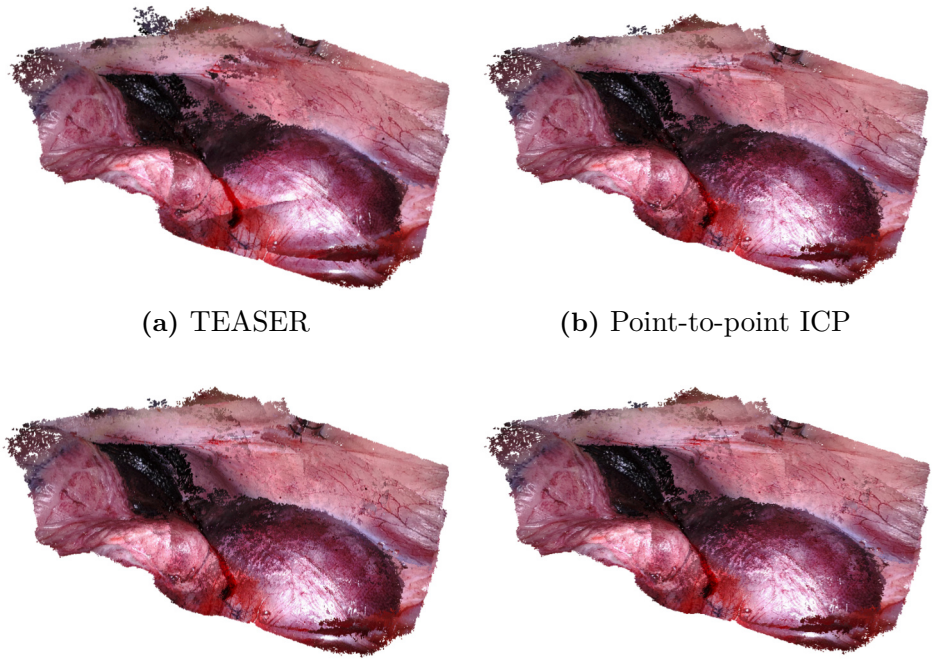
Figure 4.17: Point cloud after applying each global registration method for SCARED Dataset 2. The results for **(a)** RANSAC and **(b)** TEASER are displayed.

The median and average values for the registration with RANSAC are outlined in Table 4.4.

Table 4.4: RMSE and fitness after RANSAC registration for SCARED Dataset 2.

RMSE	**median**	3.623
	average	3.480 ± 0.900
Fitness	**median**	0.991
	average	0.973 ± 0.036

Figure 4.18 displays the point clouds after the registration process employing TEASER and the differing ICP variants. After applying any of the ICP methods, several sharp edges appear to be smoothed but no differences can be observed between the methods.



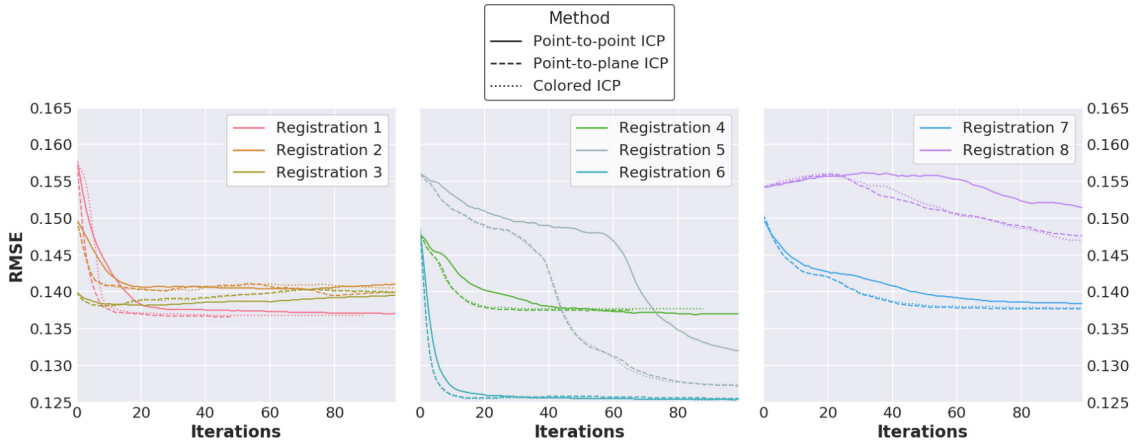
(a) TEASER **(b)** Point-to-point ICP

(c) Point-to-plane ICP **(d)** Colored ICP

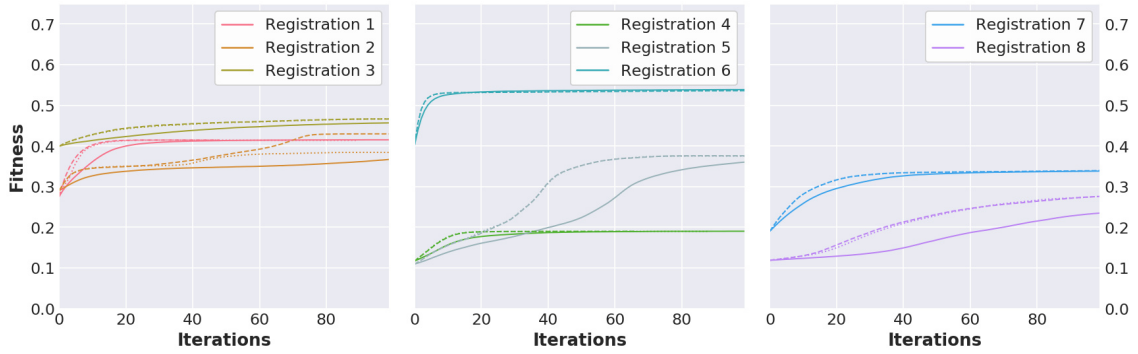
Figure 4.18: Point clouds after TEASER registration with additionally applying each of the ICP variants for SCARED Dataset 2. The point clouds after global registration with **(a)** TEASER and the additional application of **(b)** point-to-point ICP, **(c)** point-to-plane ICP and **(d)** colored ICP are displayed.

The RMSE and the fitness depending on the number of iterations for each ICP variant and for each registration are presented in Figure 4.19. Note that for Registration 1, point-to-plane ICP terminates after 49 iterations while colored ICP runs for 90 iterations. For Registration 4, point-to-plane ICP terminates after 67 iterations and colored ICP after 89 iterations.

While the RMSE converges within the first 40 iterations for most registrations regardless of the local method, for Registration 5, the RMSE decreases significantly between 40 and 60 iterations for point-to-plane as well as for colored ICP and between 60 and 80 iterations for point-to-point ICP. For Registration 8, the RMSE increases within the first 20 iterations and although the error decreases afterwards for all variants, the curve does not appear exponential as for the other registrations.



(a) RMSE

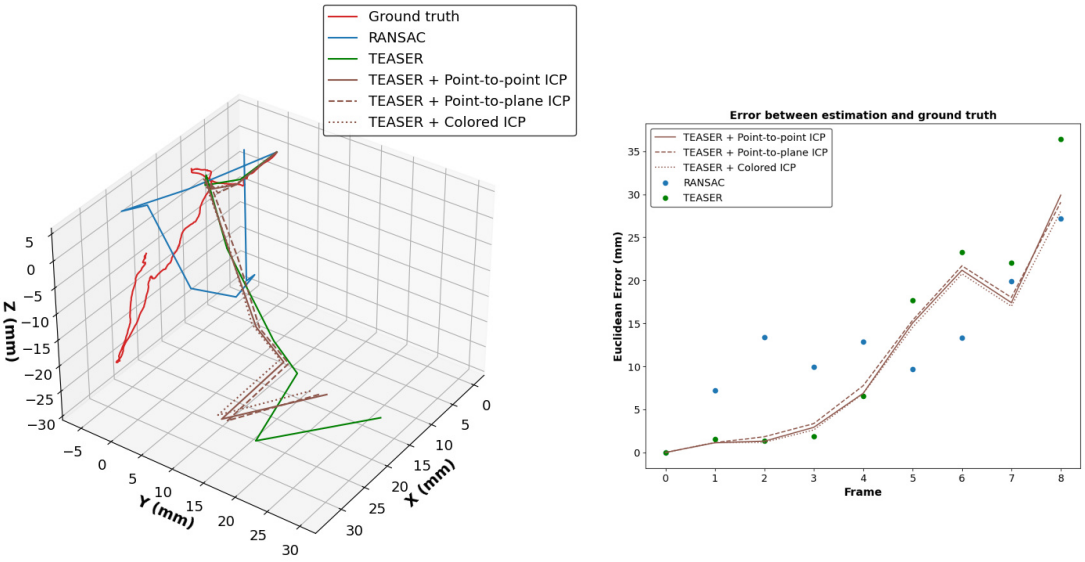


(b) Fitness

Figure 4.19: RMSE and Fitness for the ICP variants depending on the iterations for SCARED Dataset 2. In beforehand, TEASER was applied as the global registration method. The differing line types indicate the local registration method while each color signifies the registration.

The comparison to the ground truth in form of the pose graph is given in Figure 4.20. While the registration with RANSAC results in a pose graph with abrupt pose changes, the registration with TEASER appears to be closer to the ground truth. The application of any of the ICP variants after the registration utilizing TEASER alters the pose graph, but the direction remains.

The analysis of the euclidean distances for the global registration methods is given in Table 4.5 while the analysis of the RMSE, fitness and euclidean distances for the local registration methods is presented in Table 4.6.



(a) Pose graph **(b)** Euclidean distance

Figure 4.20: Pose graphs and euclidean distances in comparison with the ground truth after global and local registration for SCARED Dataset 2. Global registration includes RANSAC (blue), TEASER (green) and TEASER with any of the ICP variants (brown).

Table 4.5: Statistical analysis of the euclidean distance for the global registration for SCARED Dataset 2. Measurements are given in mm.

	RANSAC	**TEASER**
median	12.838	6.550
mean	12.601 ± 7.222	12.302 ± 12.275
maximum	27.151	36.437

Table 4.6: Statistical analysis of the RMSE, fitness and euclidean distance for the local registration for SCARED Dataset 2. The euclidean distance is given in mm.

		point-to-point ICP	**point-to-plane ICP**	**colored ICP**
RMSE	**median**	0.138	0.138	0.138
	mean	0.138 ± 0.007	0.136 ± 0.007	0.137 ± 0.007
Fitness	**median**	0.364	0.395	0.380
	mean	0.363 ± 0.113	0.378 ± 0.109	0.372 ± 0.108
Euclidean distance	**median**	6.874	7.776	6.831
	mean	10.693 ± 10.038	10.969 ± 9.865	10.271 ± 9.583

4.4.3 Dataset 3

The left images of dataset 3 consisting of 15 frames are displayed in Figure 4.21.

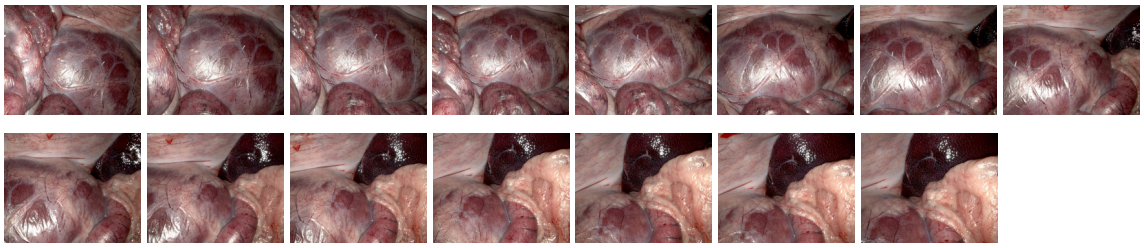
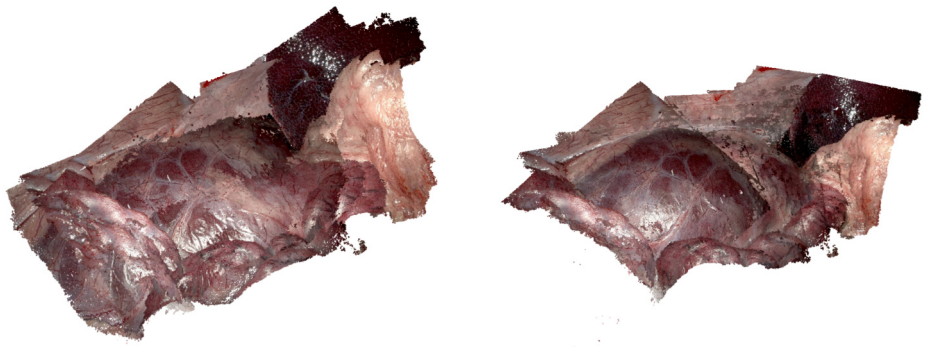


Figure 4.21: Left images of SCARED Dataset 3. The upper left image corresponds to the first frame while the lower right is the last frame.

The point clouds after the registration utilizing RANSAC and TEASER are displayed in Figure 4.22. Both registration results are orientated to the first point cloud. Although both methods generally align the rounded surface at the centre, the registrations differ visibly at the sides. With RANSAC, the sides show an offset while with TEASER, they appear to be aligned. Other differences are that with TEASER, one change of brightness is visible creating an edge at the centre and the point clouds of the later frames are tilted backwards.



(a) RANSAC **(b)** TEASER

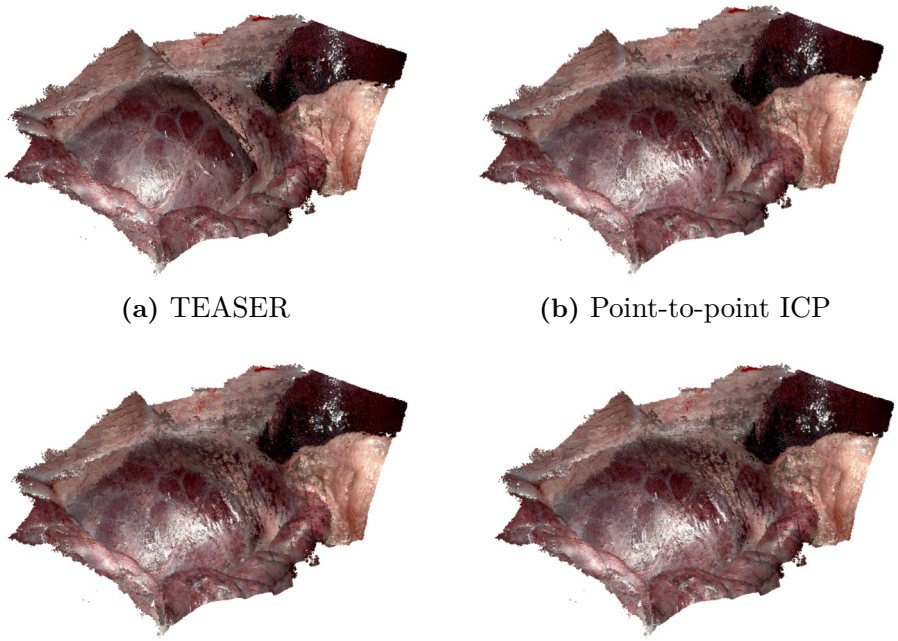
Figure 4.22: Point clouds after applying each global registration method for SCARED Dataset 3. The results for **(a)** RANSAC and **(b)** TEASER are displayed.

The median and average values for the registration with RANSAC are outlined in Table 4.7.

Table 4.7: RMSE and fitness after RANSAC registration for SCARED Dataset 3.

RMSE	**median**	1.787
	average	1.831 ± 0.369
Fitness	**median**	0.951
	average	0.931 ± 0.069

The application of each of the three ICP variants onto the registration result of TEASER is presented in Figure 4.23. While the application of any of the three ICP variants mitigates the edge at the centre, no variances can be observed between them.



(a) TEASER **(b)** Point-to-point ICP

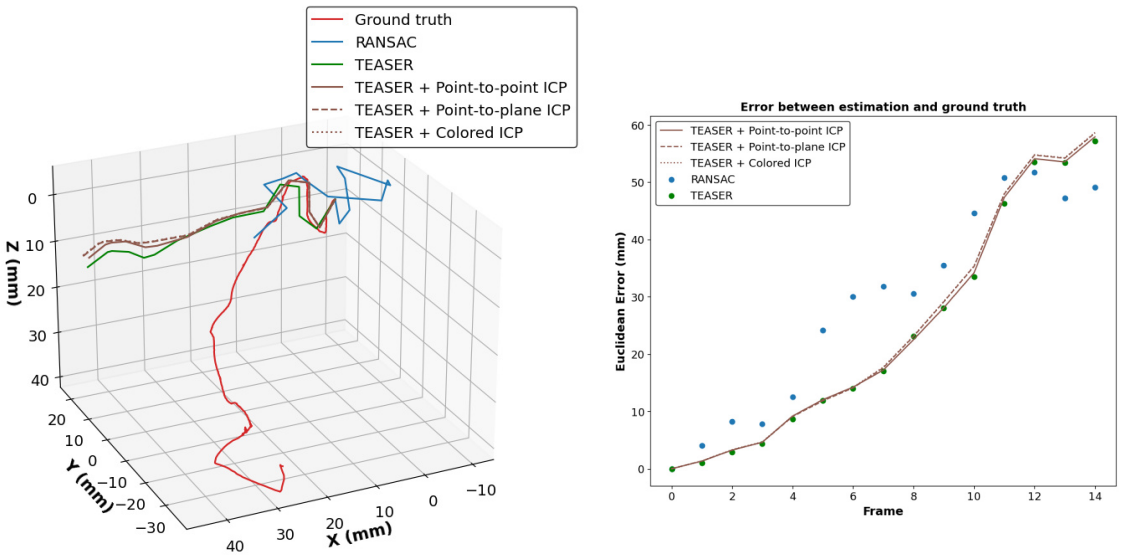
(c) Point-to-plane ICP **(d)** Colored ICP

Figure 4.23: Point clouds after TEASER registration with additionally applying each of the ICP variants for SCARED Dataset 3. The point clouds after global registration with **(a)** TEASER and the additional application of **(b)** point-to-point ICP, **(c)** point-to-plane ICP and **(d)** colored ICP are displayed.

The pose graphs and the corresponding euclidean distances for both global registration methods and the additional application of the ICP variants with TEASER are visualized in Figure 4.24. While the registration with RANSAC results in a deviated pose graph compared to the ground truth, the first poses with TEASER follow the direction of the ground truth, but drift away after five frames. The application of any of the ICP variants results in a slight change of the pose graph.

The euclidean distances reveal that they increase steadily for most frames and methods exceeding a distance of 40mm for the last four poses for all methods. The application of the local registration methods influences the euclidean distance minimally.

The statistical analysis of the euclidean distances for both global registrations is outlined in Table 4.8 while the analysis of the RMSE, fitness and euclidean distances for the local registration methods is presented in Table 4.9.



(a) Pose graph **(b)** Euclidean distance

Figure 4.24: Pose graphs and euclidean distances in comparison with the ground truth after global and local registration for SCARED Dataset 3. Registration includes RANSAC (blue), TEASER (green) and TEASER with any of the ICP variants (brown).

Table 4.8: Statistical analysis of the euclidean distance for the global registration of the SCARED Dataset 3. Measurements are given in mm.

	RANSAC	**TEASER**
median	30.526	17.055
mean	28.542 ± 17.624	23.654 ± 19.833
maximum	51.721	57.149

Table 4.9: Statistical analysis of the euclidean distance for the local registration of the SCARED Dataset 3. Measurements are given in mm.

		point-to-point ICP	**point-to-plane ICP**	**colored ICP**
RMSE	**median**	0.131	0.130	0.130
	mean	0.132 ± 0.011	0.131 ± 0.011	0.131 ± 0.011
Fitness	**median**	0.623	0.633	0.633
	mean	0.607 ± 0.142	0.616 ± 0.130	0.616 ± 0.130
Euclidean distance	**median**	17.209	17.618	17.738
	mean	23.969 ± 19.982	24.320 ± 20.289	24.391 ± 20.308

4.5 Test 3: Throat Assistant

Within this section, the registration results for the different endoscope system in combination with the throat assistant are outlined.

4.5.1 EinsteinVision 3.0

With the EinsteinVision 3.0 endoscope, seven point clouds of the throat assistant were acquired. Figure 4.25 displays the left image for each stereo pair. Chroma keying was utilized for the last three frames to exclude unwanted background at the top left.

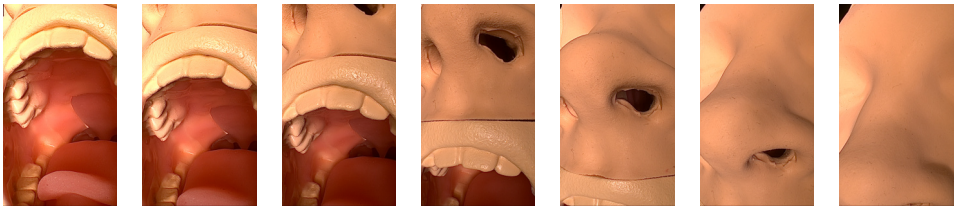
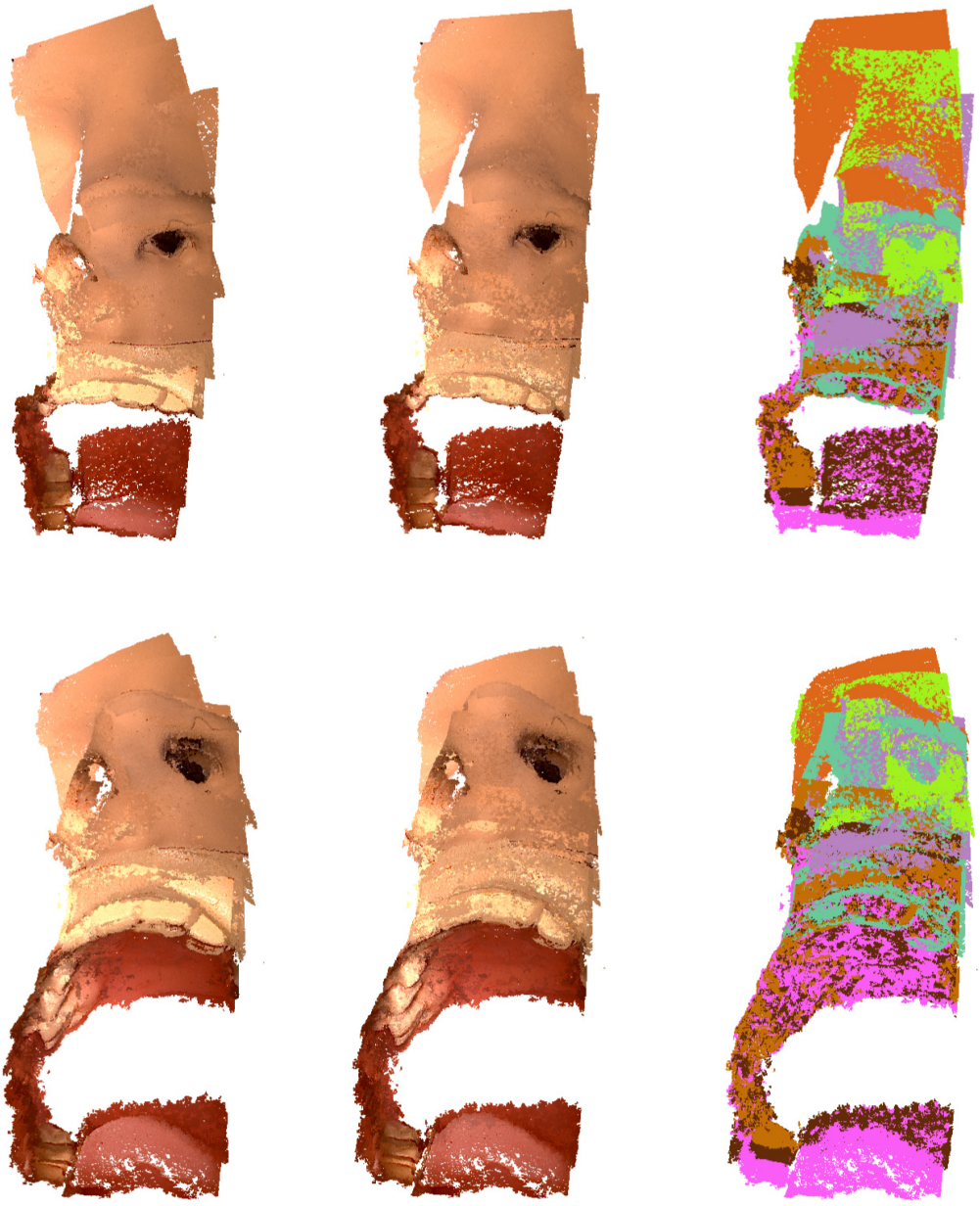


Figure 4.25: Left image of each stereo pair rotated by 90° of the throat assistant captured with the EinsteinVision 3.0 endoscope

The resulting registered point clouds after performing global registration with TEASER and additional local registration with point-to-plane ICP are displayed in Figure 4.26. The global registration results in a recognizable reconstruction of the throat assistant, but the edges of the teeth are noisy and the nose is slightly tilted to the left. After applying local registration, the teeth show less distortions and the nose appears to be straight. The uniformly colored result demonstrates the output after applying both global and local registration. In some areas, e.g. tongue and teeth, two or more point clouds are interlaced while in others, e.g. at the tip of the nose, one cloud dominates the visualization.



(a) TEASER

(b) TEASER + point-to-plane ICP

(c) TEASER + point-to-plane ICP (uniform colors)

Figure 4.26: Point clouds after employing TEASER and point-to-plane ICP for the throat assistant captured with the EinsteinVision 3.0. The point clouds after applying **(a)** TEASER and additionally **(b - c)** point-to-plane ICP are visualized. In **(a)** and **(b)** point clouds are shown in original colors while in **(c)** each point cloud is colored uniformly.

The registration with the CT model after a applying TEASER and point-to-plane ICP is visualized in Figure 4.27. Clearly, the algorithm does not find an acceptable solution. Point-to-plane ICP terminates after the first iteration with an RMSE and fitness of 0.

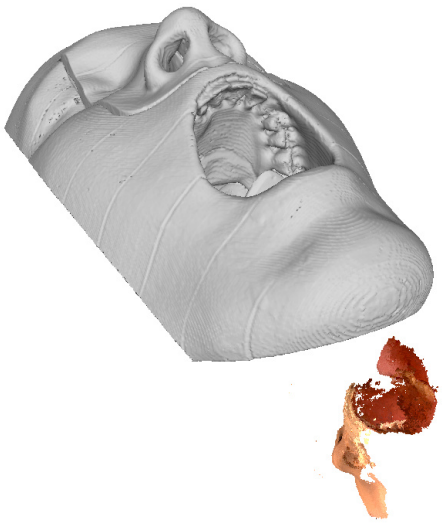
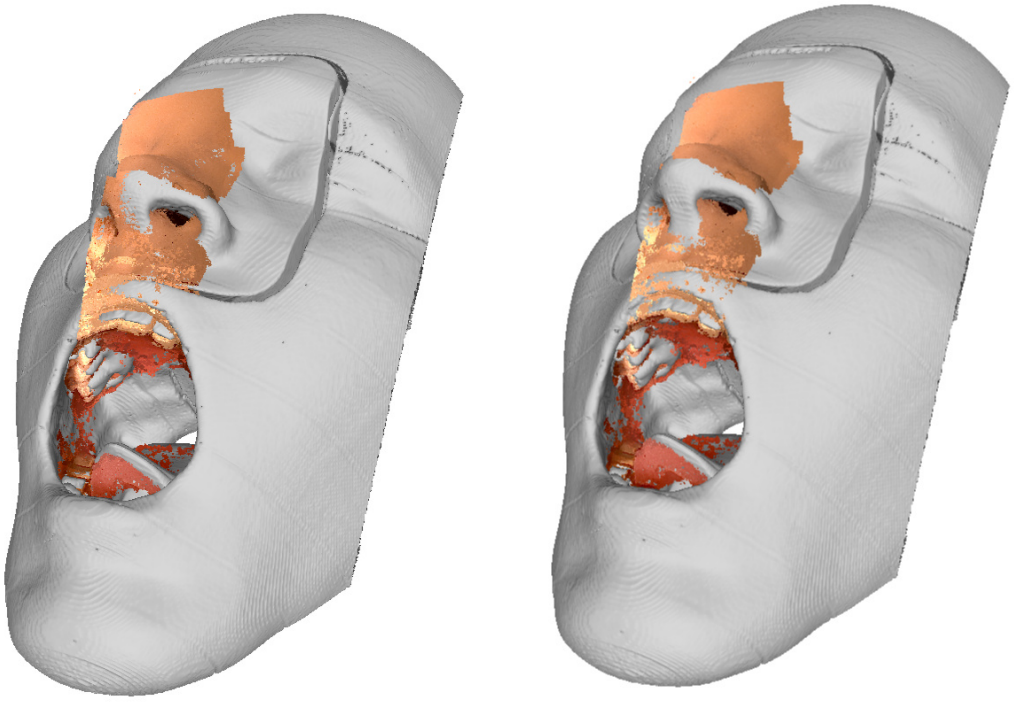


Figure 4.27: Point clouds and CT mesh after employing TEASER and point-to-plane ICP for the throat assistant captured with the EinsteinVision 3.0

The registration with the CT model after selecting five correspondences and applying point-to-plane ICP is shown in Figure 4.28. No significant changes are visible between the two results, but minor variations are observable at the columella (nose bridge between the nostrils), the front teeth as well as the molar teeth in the upper jaw. With the application of point-to-plane ICP, the inlier RMSE reduces from 0.1609 in the first iteration to 0.1584 after 100 iterations while the fitness increases from 0.1422 to 0.2173.

A detailed view of the registration result is given in Figure 4.29. It can be observed that although the tip of the nose of the endoscopic point cloud is closely aligned with the CT model, a gap is present at the nasal root. In addition, the depictions of the nostrils, the tongue and the upper molar teeth show an offset.



(a) Pick Points **(b)** Pick Points + point-to-plane ICP

Figure 4.28: Point clouds and CT mesh after employing Pick Points and point-to-plane ICP for the throat assistant captured with the EinsteinVision 3.0. The point clouds after applying **(a)** Pick-Points and additionally **(b)** point-to-plane ICP are visualized.

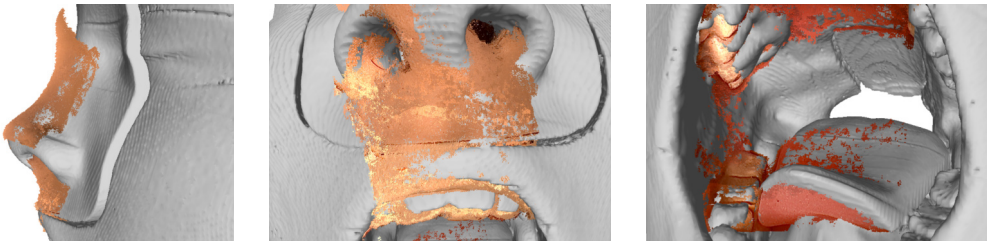


Figure 4.29: Detailed view of the point clouds after employing Pick Points and point-to-plane ICP for the throat assistant captured with the EinsteinVision 3.0

4.5.2 EndoSURGERY 3D Spectar

With the EndoSURGERY 3D Spectar endoscope, twelve point clouds were acquired of the throat assistant. The corresponding left images of each stereo pair are displayed in Figure 4.30.

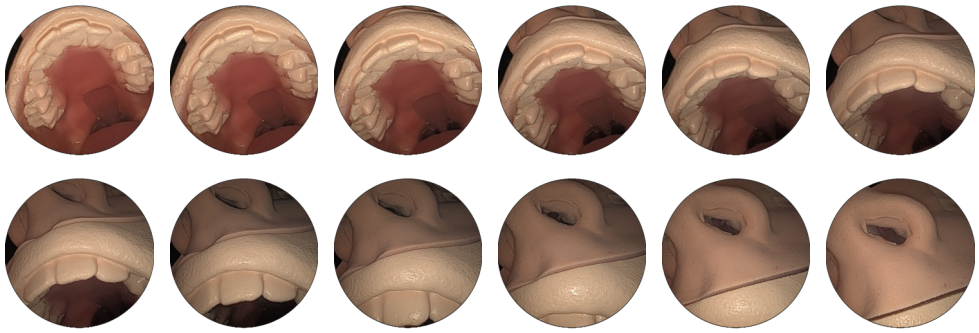


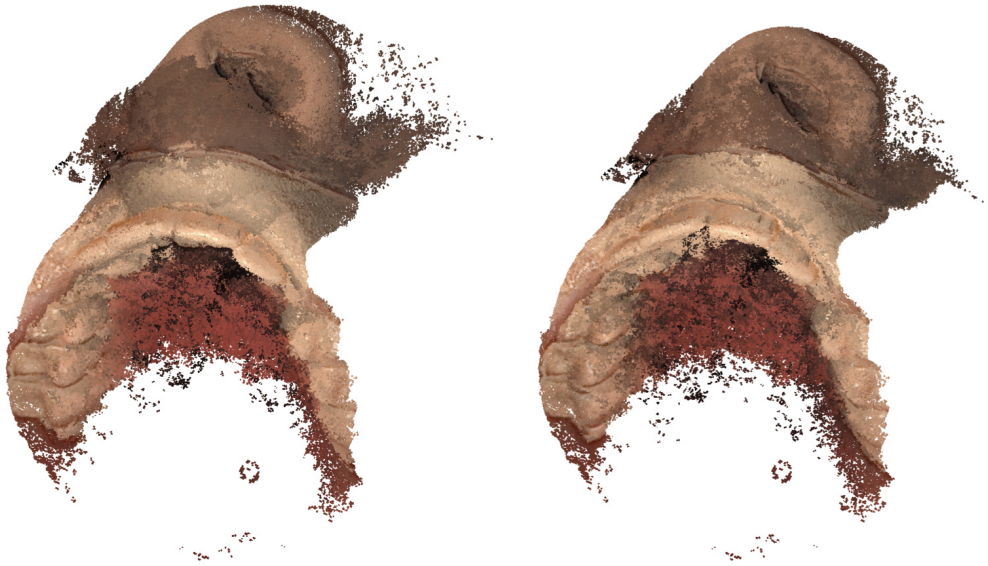
Figure 4.30: Left images of the acquisition of the throat assistant utilizing the EndoSURGERY 3D Spectar endoscope. The first frame is at the upper left while the last frame is at the lower right.

The registration results after applying TEASER and additionally point-to-plane ICP are displayed in Figure 4.31. The application of TEASER results in a recognizable partial reconstruction of the object. However, discrepancies are visible particularly at the nose. After the additional application of point-to-plane ICP, the inner side of the nose is aligned more precisely, the hard plate within the mouth consists of increased different shades of red and the nose is tilted slightly backwards.

The registration with the CT utilizing TEASER and point-to-plane ICP is displayed in Figure 4.32. After 100 iterations of point-to-plane ICP, the fitness is at 0.048 while the RMSE is 0.1899.

The registration with the CT model after selecting five correspondences and applying point-to-plane ICP is shown in Figure 4.34. In the upper view, differences are observable below and on the nose while the lower view also reveals variations at the molar teeth.

Detailed views of the final registration are displayed in Figure 4.34. While the upper lip and the front teeth appear to be aligned, the front of the nostrils and the left side of the upper molar teeth disappear in the CT model.



(a) TEASER **(b)** TEASER + point-to-plane ICP

Figure 4.31: Point clouds after employing TEASER and point-to-plane ICP for the throat assistant captured with the EndoSURGERY 3D Spectar. The point clouds after applying **(a)** TEASER and additionally **(b)** point-to-plane ICP are visualized.

Figure 4.32: Point clouds and CT mesh after employing TEASER and point-to-plane ICP for the throat assistant captured with the EndoSURGERY 3D Spectar

(a) Pick Points **(b)** Pick Points + point-to-plane ICP

Figure 4.33: Point clouds after employing Pick Points and point-to-plane ICP for the throat assistant captured with the EndoSURGERY 3D Spectar and the CT model. The point clouds after applying **(a)** Pick-Points and additionally **(b)** point-to-plane ICP are visualized

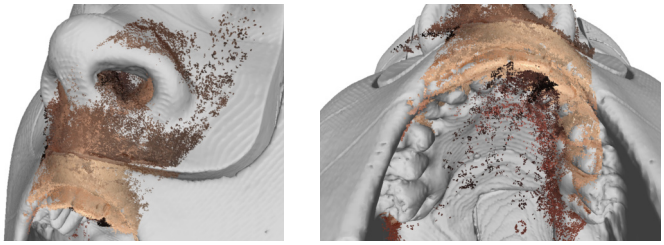


Figure 4.34: Detailed view of point clouds and CT mesh after employing Pick Points and point-to-plane ICP for the throat assistant (EndoSURGERY 3D Spectar)

5
Discussion

All test-setups show that point cloud registration on medical datasets is possible. The accuracy of the results is influenced by the acquisition, type of object or data, point cloud analysis and the registration method(s) with the utilized parameters. This chapter discusses the previously outlined results classified by each step within the pipeline.

5.1 Point Cloud Acquisition

Acquisition

The acquired data greatly influences how well a registration result represents a real-world scenario. An accurate calibration of the stereo systems is crucial to obtain an accurate depth estimation for point cloud acquisition. Additionally, rectification as well as bilateral and median filtering are necessary to acquire clean and noise-free point clouds.

The type of endoscope also influences the quality of the acquired point clouds. For instance, an incorporated light source facilitates the acquisition process and the user does not have to adjust an external light source. Another aspect is the stereo base. A small stereo base, as with the EndoSURGERY 3D Spectar, results in a less accurate depth estimation. Hence, the endoscope needs to be closer to the object and more point clouds have to be captured for the same surface compared to the EinsteinVision systems. However, endoscopes with a small stereo base are specifically designed for microscopy requiring a small tube diameter.

With chroma keying, most of the unwanted background can be removed and although some green pixels remain, they may be eliminated in outlier removal and/or do not influence the registration process. In practice, chroma keying is not necessary for acquiring point clouds of internal body organs. However, it simplifies test-setups as presented in this thesis.

Objects and Data

Point cloud registration is specifically challenging on symmetrical objects, objects with a significant amount of blind spots from differing viewpoints, on environments that are primarily based in one plane as well as point clouds from differing modalities. This is shown for the registration using the specimen, for dataset 1 of the SCARED

challenge and the registration with the CT. The registration of the specimen is particularly challenging since the object is symmetrical and steep edges result in blind spots. Hence, when the object is rotated, one side of the ledge disappears while another emerges resulting in a small overlap between the acquired clouds. This and the additional symmetric property of the object lead to falsely aligned point clouds which is signified by the abrupt changes of the pose graph.

The results for the SCARED challenge Dataset 1 show that although there are only subtle differences between neighboring captured frames and hence point clouds, both global registration methods do not find optimal transformations even for greater distance thresholds or noise bounds. This can be explained by the predominantly planar geometry of the environment. For predominantly planar surfaces, surface normals point primarily in one direction resulting in similar geometric features for all points. This leads to rotation offsets between registered clouds.

The result for the CT registration with TEASER suggests that correspondence estimation is specifically challenging on data with differing modalities. The results also indicate that the head phantom, specifically the nose, was not static and was instead titled forward compared to the CT model. To improve the registration result, prior segmentation of the CT could decrease the outlier rate. The interactive approach where the user selects several correspondences for the global registration does result in an acceptable transformation and may even be faster than an automatic correspondence estimation. However, this approach does not necessarily yield the same result after repeating the registration depending on how many correspondences are chosen and how accurate the user selects correspondences.

Another aspect is optical characteristics of the material or tissue, specifically reflectivity. For the specimen, after the registration of either global registration method, one side of the object appears with less density. However, these points do not actually represent this side of the object but are shadow points that result from the reflective behavior of the objects material (see Figure 4.7). A removal of these shadow points could improve the registration result and lead to more clarity. However, within this thesis shadow points only emerge for this specimen and such a filter is unnecessary for the other datasets.

In general, medical point cloud registration rarely incorporates objects which are as symmetrical as the specimen. However, certain tissues may be reflecting specifically if (body) fluids are present.

5.2 Point Cloud Analysis

Outlier Removal

Outlier removal is necessary for point clouds that contain a significant amount of noise that could potentially influence the registration result. In addition, outlier

points decrease the quality and visual clarity of a point cloud distracting the user from the essentials.

The chosen values for the outlier removal, namely the number of points within the neighborhood of a point and the standard deviation ratio, remove scattered outliers sufficiently. However, since outlier removal is a statistical method, clusters of outliers are not identified as such. These clusters can influence a registration result and may be irritating to the user. This particularly occurred for the acquisition using the EinsteinVision 3.0 endoscope system for the throat assistant where dark background areas are in plane with areas at the front in the point cloud. Hence, these outliers have to be eliminated by hand or an implementation of a more complex algorithm is needed.

Downsampling

Downsampling is crucial to decrease the computation time but also to ensure that the program does not terminate unexpectedly due to memory shortage. The latter is specifically critical for the global registration using TEASER. For downsampling, the voxel size has to be set in accordance with the sensor resolution and the specifications of the utilized PC.

Normal Estimation

Normal estimation influences feature extraction and correspondence estimation and hence, both global registration methods substantially. Within this thesis, only the PCA based estimation of surface normals was utilized. As visible in Figure 4.6, this method results in slanted surface normals at sharp edges specifically at the border of point clouds. Other methods including deep learning approaches may result in more accurate estimations.

Feature Extraction and Correspondence Estimation

The quality of the feature extraction directly influences the correspondence estimation and the resulting inlier ratio. All results for RANSAC suggest that the the amount of outlier correspondences is greater than 90% since this method generally performs well even with greater outlier ratios [3].

Although FPFHs are widely used for feature extraction in point cloud registration, adaptions and other approaches could improve the correspondence estimation, specifically the registration with the CT. For example, instead of computing a feature descriptor for each point, a 3D keypoint detector such as Normal Aligned Radial Feature (NARF) in conjunction with the FPFH descriptor could accelerate the correspondence estimation [52, 53]. In addition, a variety of deep learning methods have emerged for geometrical feature extraction. Methods like 3D-SmoothNet [54] or MS-SVConv [55] may yield more inlier correspondences albeit these methods were trained on indoor or outdoor scenes and not on medical datasets.

5.3 Global and Local Registration

As stated, all test-setups show that point cloud registration on medical datasets is possible. Despite that, the registration methods yield differing registration results with varying quality. Compared to datasets of indoor or outdoor scenes, areas with inaccuracies are visually harder to detect since medical point clouds do not show distinctive objects. However, the global registration results show that TEASER generally results in more accurate transformations than when utilizing RANSAC. Solely for the registration with the specimen, the result with RANSAC is better. For local registration, the visual differences between the ICP variants are neglectable.

Transformation

Within this thesis, solely the similarity transformation was utilized for all registration methods. However, medical images often include non-rigid deformations. On the other hand, non-rigid transformations have more DoF and hence, are more difficult to estimate.

Maximum Clique

For TEASER, the maximum clique may indicate the quality of the registration result before terminating the transformation estimation. As presented for SCARED Dataset 1 (see subsection 4.4.1), a larger voxel size for the downsampling step and a smaller noise bound lead to a smaller maximum clique and hence, less inlier correspondences to align the point clouds. However, a small noise bound is preferred to align the point clouds closely to each other.

RMSE and Fitness

The RMSE and fitness differ greatly between the registration methods. With RANSAC, the fitness is above 0.9 for all datasets where a distance threshold of greater than 5mm was utilized and the RMSE is above 1.5mm. For TEASER and the additional application of any ICP variant, the fitness is on average above 0.25 while the RMSE lies below 0.15mm. Hence, although a fitness close to one signifies a good registration, the RMSE is significantly larger with RANSAC indicating that the registered point clouds are not as closely aligned as with TEASER.

If several registrations are conducted with the same registration method, a higher RMSE signifies a poorly aligned point cloud compared to an acceptable registration results in the same dataset. If a point cloud is aligned poorly, all further registrations rely on that result. Hence, even if the later point clouds are aligned accurately, the overall registration result is poorly. The RMSE indicates a poor registration result compared to the other results and hence, an extension of the algorithm may be helpful to adjust the registration parameters to adaptively improve the result.

For the ICP variants, the fitness generally correlates with the inverse of the RMSE. Both decrease significantly within the first 40 iterations, but small changes occur afterwards. This indicates that 40 iterations are sufficient to improve the global

registration result while reducing the computation time. If a point cloud was poorly aligned with the global registration method, a fluctuation of the RMSE indicates that ICP does not find an accurate solution.

Pose Graph

The comparison between the estimated poses and the ground truth data for the SCARED datasets show that although a registration may visually appear to be accurate, a significant difference can be seen between the estimated poses and the ground truth. For all datasets and methods, the median and average of the euclidean distances are greater than 9mm. For the registration with TEASER, the estimated pose graphs approximately follows the ground truth for the first poses of SCARED Dataset 2 and 3 but drift further apart after several poses. In this case, Simultaneous Localization And Mapping (SLAM) [49] may be useful to optimize the pose graph. However, this may be challenging since the point clouds acquired within this thesis do not follow a closed-loop pose graph. Another approach to optimize the registration result is bundle adjustment [56].

Even if no ground truth is available, the visualization of the pose graph may be beneficial for the user to detect falsely aligned point clouds. If point clouds are captured continuously, abrupt changes in the pose graph indicate poor registration results.

5.4 Other Aspects

Color

Since the input point clouds were acquired from different viewpoints, they vary in brightness. Hence, after the registration is conducted, points representing the same spatial position do not necessarily have the same color. This is specifically impacting small details, but may also be irritating for larger regions. A filtering algorithm e.g. based on the HSV color model could merge color components based on the brightness. Another option may be to remove points from the source point cloud after the registration that have a drastic color change compared to the target point cloud at the same spatial position.

Resolution

The resolution of the final registered point cloud depends on the number of registered point clouds, the resolution of each point cloud and the components of the employed PC. If the result comprises a small number of point clouds, the entire final point cloud can be saved. However, if a large number of point clouds are registered and hence a large number of points, they must be reduced in size, otherwise they may not be storable or cannot be visualized.

6
Conclusion

The aim of this thesis was to develop a modular framework for point cloud registration obtained from stereoscopic images in the context of endoscopic surgery. It can be concluded, that point clouds from stereo-endoscopic images can be registered automatically although the results indicate a substantial amount of outlier correspondences. However, in some instances the registration results in form of a pose graph differ greatly from the ground truth specifically for subsequent results. As a result, although the registration result appears to be accurate, distances and angles may not represent the real-life scenario correctly. The global (coarse) registration with TEASER generally performed more accurately than RANSAC. For the local (fine) registration, any of the ICP variants improve the registration result and differences are neglectable.

For the user, a pose graph can be an indication for poor registration results even if ground truth data is unavailable. In addition, if several registrations are conducted, outlier RMSE and fitness within the dataset signify poorly aligned point clouds. Overall, point clouds with substantial symmetry, small overlapping areas, planar geometry and from differing modalities are difficult to align.

The automatic registration with the CT utilizing TEASER failed. This indicates that the estimated correspondences have a tremendous outlier ratio. By segmentation or adding a keypoint detector, automatic registration with a CT may be possible. However, an interactive approach where the user selects correspondences between the endoscopic point clouds and the point cloud obtained from the CT data is an acceptable solution. A substantial disadvantage of this approach is that this does not yield the same solution for multiple repetitions.

Generally, both the registration for registering endoscopic point clouds among each other and the registration with the CT may be improved by adapting the analysis pipeline, namely downsampling, normal estimation, feature extraction and correspondence estimation. Other approaches e.g. for feature description including deep learning methods may yield more inlier correspondences and hence, a better registration result. However, current deep learning feature descriptors for point clouds are generally trained on indoor or outdoor scenes; the utilization with medical datasets has to be explored.

Other future advancements of the framework may include the usage of a non-rigid transformation to transform the source point cloud. In addition, SLAM based

optimization after applying global registration as well as deep learning approaches for the global registration may yield more accurate registrations. The adaptions of color, specifically the brightness, for overlapping regions between point clouds could improve the visual clarity of the result.

Bibliography

[1] J. L. Prince and J. M. Links, *Medical imaging signals and systems*, 2nd ed. Boston: Pearson, 2015.

[2] A. A. Goshtasby, *Image Registration*. London: Springer London, 2012.

[3] H. Yang, J. Shi, and L. Carlone, "TEASER: Fast and certifiable point cloud registration," *IEEE Transactions on Robotics*, vol. 37, no. 2, pp. 314–333, Apr. 2021. DOI: 10.1109/TRO.2020.3033695.

[4] J.-C. Rosenthal *et al.*, "Endoscopic measurement of nasal septum perforations," *HNO*, vol. 70, no. S1, pp. 1–7, Feb. 2022. DOI: 10.1007/s00106-021-01102-4.

[5] E. L. Wisotzky *et al.*, "Interactive and multimodal-based augmented reality for remote assistance using a digital surgical microscope," in *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, Mar. 2019, pp. 1477–1484. DOI: 10.1109/VR.2019.8797682.

[6] G. J. Tortora and B. Derrickson, *Introduction to the Human Body: The Essentials of Anatomy and Physiology*, 10th ed. Hoboken, NJ: John Wiley & Sons, Inc., 2015.

[7] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge: Cambridge University Press, 2004.

[8] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics*. London: Springer London, 2009.

[9] K. Nomura *et al.*, "Comparison of 3d endoscopy and conventional 2d endoscopy in gastric endoscopic submucosal dissection: An ex vivo animal study," *Surgical Endoscopy*, vol. 33, no. 12, pp. 4164–4170, Dec. 2019. DOI: 10.1007/s00464-019-06726-w.

[10] J. Ilgner and M. Westhofen, "Practical aspects on the use of stereoscopic applications in operative theatres," in *2011 International Conference on 3D Imaging (IC3D)*, Dec. 2011, pp. 1–5. DOI: 10.1109/IC3D.2011.6584395.

[11] F. Bernard, P. Richard, A. Kahn, and H.-D. Fournier, "Does 3d stereoscopy support anatomical education?" *Surgical and Radiologic Anatomy*, vol. 42, no. 7, pp. 843–852, Jul. 2020. DOI: 10.1007/s00276-020-02465-z.

[12] M. Weinmann, *Reconstruction and Analysis of 3D Scenes*. Cham: Springer International Publishing, 2016.

[13] A. A. Goshtasby, *2-D and 3-d Image Registration: For Medical, Remote Sensing, and Industrial Applications*. Hoboken (N.J.): John Wiley & Sons, Inc., 2005.

[14] M. Han, Y. Dai, and J. Zhang, "Endoscopic navigation based on three-dimensional structure registration," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 24, 2020, pp. 2900–2905. DOI: 10.1109/IROS45743.2020.9340692.

[15] E. L. Wisotzky, J.-C. Rosenthal, U. Wege, A. Hilsmann, P. Eisert, and F. C. Uecker, "Surgical guidance for removal of cholesteatoma using a multispectral 3d-endoscope," *Sensors*, vol. 20, no. 18, p. 5334, Sep. 2020. DOI: 10.3390/s20185334.

[16] Fraunhofer Heinrich Hertz Institute. (2021). "About us," [Online]. Available: https://www.hhi.fraunhofer.de/en/fraunhofer-hhi/about-us.html (visited on 09/20/2021).

[17] ——, (2021). "Capture & display systems," [Online]. Available: https://www.hhi.fraunhofer.de/en/departments/vit/research-groups/capture-display-systems.html (visited on 09/20/2021).

[18] M. Allan *et al.*, "Stereo correspondence and reconstruction of endoscopic data challenge," *arXiv:2101.01133 [cs]*, Jan. 28, 2021. arXiv: 2101.01133.

[19] T. Luhmann, S. Robson, S. Kyle, and J. Boehm, *Close-Range Photogrammetry and 3D Imaging*. Berlin: De Gruyter, Inc., 2013.

[20] J. Kramer and A.-M. von Pippich, *From Natural Numbers to Quaternions*. Cham: Springer International Publishing, 2017.

[21] J. Voight, *Quaternion Algebras*. Cham: Springer International Publishing, 2021.

[22] F. Devernay and O. Faugeras, "Straight lines have to be straight," *Machine Vision and Applications*, vol. 13, no. 1, pp. 14–24, Aug. 1, 2001. DOI: 10.1007/PL00013269.

[23] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000. DOI: 10.1109/34.888718.

[24] R. Klette, *Concise Computer Vision*. London: Springer London, 2014.

[25] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, Kerkyra, Greece: IEEE, 1999, pp. 1150–1157. DOI: 10.1109/ICCV.1999.790410.

[26] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Computer Vision – ECCV 2006*, 2006, pp. 404–417. DOI: 10.1007/11744023_32.

[27] F. Zilly, C. Riechert, P. Eisert, and P. Kauff, "Semantic kernels binarized - a feature descriptor for fast and robust matching," in *2011 Conference for Visual Media Production*, Nov. 2011, pp. 39–48. DOI: 10.1109/CVMP.2011.11.

[28] F. Zilly, M. Muller, P. Eisert, and P. Kauff, "Joint estimation of epipolar geometry and rectification parameters using point correspondences for stereoscopic TV sequences," *Proceedings of 3DPVT*, p. 7, 2010.

[29] F. Zilly, M. Müller, P. Eisert, and P. Kauff, "The stereoscopic analyzer — an image-based assistance tool for stereo shooting and 3d production," in *2010 IEEE International Conference on Image Processing*, Sep. 2010, pp. 4029–4032. DOI: 10.1109/ICIP.2010.5649828.

[30] W. Waizenegger, I. Feldmann, O. Schreer, P. Kauff, and P. Eisert, "Real-time 3d body reconstruction for immersive TV," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sep. 2016, pp. 360–364. DOI: 10.1109/ICIP.2016.7532379.

[31] A. Nuchter, K. Lingemann, and J. Hertzberg, "Cached k-d tree search for ICP algorithms," in *Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)*, Aug. 2007, pp. 419–426. DOI: 10.1109/3DIM.2007.15.

[32] R. B. Rusu, "Semantic 3d object maps for everyday manipulation in human living environments," *KI - Künstliche Intelligenz*, vol. 24, no. 4, pp. 345–348, Nov. 2010. DOI: 10.1007/s13218-010-0059-6.

[33] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3d: A modern library for 3d data processing," *arXiv:1801.09847*, 2018.

[34] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3d registration," in *2009 IEEE International Conference on Robotics and Automation*, May 2009, pp. 3212–3217. DOI: 10.1109/ROBOT.2009.5152473.

[35] Á. P. Bustos and T.-J. Chin, "Guaranteed outlier removal for point cloud registration with correspondences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2868–2882, Dec. 2018. DOI: 10.1109/TPAMI.2017.2773482.

[36] B. Jian and B. C. Vemuri, "Robust point set registration using gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1633–1645, Aug. 2011. DOI: 10.1109/TPAMI.2010.223.

[37] Y. Díez, F. Roure, X. Lladó, and J. Salvi, "A qualitative review on 3d coarse registration methods," *ACM Computing Surveys*, vol. 47, no. 3, pp. 1–36, Apr. 2015. DOI: 10.1145/2692160.

[38] A. A. Goshtasby, *Theory and Applications of Image Registration*. Hoboken, N.J.: John Wiley & Sons, Incorporated, 2017.

[39] L. G. Brown, "A survey of image registration techniques," *ACM Computing Surveys*, vol. 24, no. 4, pp. 325–376, Dec. 1992. DOI: 10.1145/146370.146374.

[40] A. W. Fitzgibbon, "Robust registration of 2d and 3d point sets," vol. 21, no. 13-14, pp. 1145–1153, Oct. 2003. DOI: 10.1016/j.imavis.2003.09.004.

[41] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981. DOI: 10.1145/358669.358692.

[42] H. Yang, P. Antonante, V. Tzoumas, and L. Carlone, "Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1127–1134, Apr. 2020. DOI: 10.1109/LRA.2020.2965893.

[43] J.-B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of Convex Analysis*. Berlin, Heidelberg: Springer, 2001.

[44] H. Yang and L. Carlone, "A quaternion-based certifiably optimal solution to the wahba problem with outliers," *arXiv:1905.12536 [cs, math]*, Sep. 22, 2019. arXiv: 1905.12536.

[45] P. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, Feb. 1992. DOI: 10.1109/34.121791.

[46] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *1991 IEEE International Conference on Robotics and Automation*

Proceedings, vol. 3, Apr. 1991, pp. 2724–2729. DOI: 10.1109/ROBOT.1991.132043.

[47] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, May 2001, pp. 145–152. DOI: 10.1109/IM.2001.924423.

[48] J. Park, Q.-Y. Zhou, and V. Koltun, "Colored point cloud registration revisited," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 143–152. DOI: 10.1109/ICCV.2017.25.

[49] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based SLAM," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, 2010. DOI: 10.1109/MITS.2010.939925.

[50] Fraunhofer Heinrich Hertz Institute. (2021). "STAN – steroscopic analyzer," [Online]. Available: https://www.hhi.fraunhofer.de/en/departments/vit/technologies-and-solutions/capture/stan-steroscopic-analyzer.html (visited on 09/20/2021).

[51] W. Burger and M. J. Burge, *Digital Image Processing: An Algorithmic Introduction Using Java*. London: Springer London, 2016.

[52] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "NARF: 3d range image features for object recognition," in *Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, vol. 44, 2010, p. 2.

[53] P. Stancelova, E. Sikudova, and Z. Cernekova, "3d feature detector-descriptor pair evaluation on point clouds," in *2020 28th European Signal Processing Conference (EUSIPCO)*, Jan. 2021, pp. 590–594. DOI: 10.23919/Eusipco47968.2020.9287339.

[54] Z. Gojcic, C. Zhou, J. D. Wegner, and A. Wieser, "The perfect match: 3d point cloud matching with smoothed densities," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 5540–5549. DOI: 10.1109/CVPR.2019.00569.

[55] S. Horache, J.-E. Deschaud, and F. Goulette, "3d point cloud registration with multi-scale architecture and unsupervised transfer learning," in *2021 International Conference on 3D Vision (3DV)*, Dec. 2021, pp. 1351–1361. DOI: 10.1109/3DV53792.2021.00142.

[56] H. Huang, Y. Sun, J. Wu, J. Jiao, X. Hu, L. Zheng, L. Wang, and M. Liu, "On bundle adjustment for multiview point cloud registration," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8269–8276, Oct. 2021. DOI: 10.1109/LRA.2021.3105686.

[57] B. Siciliano and O. Khatib, Eds., *Springer Handbook of Robotics*, Cham: Springer International Publishing, 2016. DOI: 10.1007/978-3-319-32552-1.

A

Appendix 1

A.1 Derivation of the Rotation Matrix Using Quaternions

In this section, the rotation matrix will be derived using quaternions. The fundamentals of quaternions and a more detailed outline can be found in *From Natural to Quaternions* by Kramer and von Pippich and *Quaternion Algebra* by Voight [20, 21]. Furthermore, Siciliano and Khatib give an overview of position and orientation for different numerical systems in *Springer Handbook of Robotics* [57].

The skew field of Hamilton's quaternions is defined as follows:

$$\mathbb{H} := \{q = q_0 + q_1 \cdot i + q_2 \cdot j + q_3 \cdot k \mid q_0, q_1, q_2, q_3 \in \mathbb{R}\} \tag{A.1}$$

Furthermore, a multiplication table for the operators i, j and k is given:

$$\begin{aligned} &i^2 = j^2 = k^2 = -1 \\ &ij = k,\ ji = -k \\ &jk = i,\ kj = -i \\ &ki = j,\ ik = -j \end{aligned} \tag{A.2}$$

Note, that the multiplication is distributive and associative, but not commutative!

The inverse of the quaternion q is denoted as $q*$ and defined as:

$$q* = \begin{pmatrix} q_0 \\ -q_1 i \\ -q_2 j \\ -q_3 k \end{pmatrix} \tag{A.3}$$

The rotation from coordinate frame A to coordinate frame B is the multiplication

$$p_B = q \cdot p_A \cdot q* = \begin{pmatrix} q_0 \\ q_1 i \\ q_2 j \\ q_3 k \end{pmatrix} \begin{pmatrix} 0 \\ xi \\ yj \\ zk \end{pmatrix} \begin{pmatrix} q_0 \\ -q_1 i \\ -q_2 j \\ -q_3 k \end{pmatrix} \tag{A.4}$$

The first multiplication gives the following result:

$$\begin{aligned} q \cdot p_A = & q_0 xi + q_0 yj + q_0 zk \\ & + q_1 xi^2 + q_1 yij + q_1 zik \\ & + q_2 xji + q_2 yj^2 + q_2 zjk \\ & + q_3 xki + q_3 ykj + q_3 zk^2 \end{aligned} \tag{A.5}$$

The imaginary parts can then be substituted with the terms from the multiplication table in (A.2). Sorting by the imaginary part results in:

$$\begin{aligned} q \cdot p_A = & -q_1 x - q_2 y - q_3 z \\ & + i \cdot (q_0 x + q_2 z - q_3 y) \\ & + j \cdot (q_0 y - q_1 z + q_3 x) \\ & + k \cdot (q_0 z + q_1 y - q_2 x) \end{aligned} \tag{A.6}$$

This term is then multiplicated with the inverse of q:

$$\begin{aligned} q \cdot p_A \cdot q* = & (-q_1 x - q_2 y - q_3 z) \cdot (q_0 - q_1 i - q_2 j - q_3 k) \\ & + i \cdot (q_0 x + q_2 z - q_3 y) \cdot (q_0 - q_1 i - q_2 j - q_3 k) \\ & + j \cdot (q_0 y - q_1 z + q_3 x) \cdot (q_0 - q_1 i - q_2 j - q_3 k) \\ & + k \cdot (q_0 z - q_1 y - q_2 x) \cdot (q_0 - q_1 i - q_2 j - q_3 k) \end{aligned} \tag{A.7}$$

With the use of (A.2) and again, sorting by the imaginary parts, this results in:

$$\begin{aligned} q \cdot p_A \cdot q* = & -q_0 q_1 x - q_0 q_2 y - q_0 q_3 z + q_0 q_1 x + q_1 q_2 z - q_1 q_3 y \\ & + q_0 q_2 y - q_1 q_2 z + q_2 q_3 x + q_0 q_3 z + q_1 q_3 y - q_2 q_3 x \\ & + i \cdot (q_1^2 x + q_1 q_2 y + q_1 q_3 z + q_0^2 + q_0 q_2 z - q_0 q_3 y \\ & - q_0 q_3 y + q_1 q_3 z - q_3^2 x + q_0 q_2 z - q_1 q_2 y - q_2^2 x) \\ & + j \cdot (q_1 q_2 x + q_2^2 y + q_2 q_3 z + q_0 q_3 x + q_2 q_3 z - q_3^2 y \\ & + q_0^2 y - q_0 q_1 z + q_0 q_3 x - q_0 q_1 z + q_1^2 y + q_1 q_2 x) \\ & + k \cdot (q_1 q_3 x + q_2 q_3 y + q_3^2 z - q_0 q_2 x - q_2^2 z + q_2 q_3 y \\ & + q_0 q_1 y - q_1^2 z + q_1 q_3 x + q_0^2 z - q_0 q_1 y - q_0 q_2 x) \end{aligned} \tag{A.8}$$

The imaginary part becomes zero and sorting by x, y and z results in:

$$\begin{aligned} q \cdot p_A \cdot q* = & i \cdot (x(q_0^2 + q_1^2 - q_2^2 - q_3^2) + 2y(q_1 q_2 - q_0 q_3) + 2z(q_0 q_2 + q_1 q3)) \\ & + j \cdot (2x(q_0 q_3 + q_1 q_2) + y(q_0^2 - q_1^2 + q_2^2 - q_3^2) + 2z(q_2 q_3 - q_0 q_1)) \\ & + k \cdot (2x(q_1 q_3 - q_0 q_2) + 2y(q_2 q_3 - q_0 q_1) + z(q_0^2 - q_1^2 - q_2^2 + q_3^2)) \end{aligned} \tag{A.9}$$

In matrix form, the rotation matrix is then written as:

$$R = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_o^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{A.10}$$

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY