



UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR ROBOTIK
UND KOGNITIVE SYSTEME

Structured Light for Head Tracking in Radiation Therapy

Benjamin Wagner

Dissertation

Universität zu Lübeck
Institut für Robotik und Kognitive Systeme

From the Institute for Robotics and Cognitive Systems
of the University of Lübeck

Director: Prof. Dr.-Ing. Achim Schweikard

Structured Light for Head Tracking in Radiation Therapy

Dissertation
for Fulfillment of
Requirements
for the Doctoral Degree
of the University of Lübeck

from the Department of Computer Sciences

Submitted by

Benjamin Wagner
from Bad Oldesloe

Lübeck 2015

First referee: Prof. Dr.-Ing. Achim Schweikard

Second referee: Prof. Dr.-Ing. Erik Maehle

Date of oral examination: 08.12.2015

Approved for printing. Lübeck, 15.01.2016

Kurzzusammenfassung

Die optische Kopf-Lokalisierung in der robotergestützte Strahlentherapie bietet entscheidende Vorteile. Die Methode basiert auf harmlosen Laserlicht und ermöglicht eine hohe Lokalisierungsrate. Der kritische Nachteil von optischer Kopf-Lokalisierung ist die begrenzte Genauigkeit. Das Problem ist, dass die Registrierung von Oberflächen durch das deformierbare Weichgewebe beeinflusst wird. Um die Genauigkeit von optischer Kopf-Lokalisierung zu erhöhen, hat unsere Forschungsgruppe ein neuartiges Mess-System entwickelt. Durch die Analyse der Reflektion eines projizierten Punktgitters, werden eine punktbasierte Rekonstruktion der Stirnoberfläche und optische Merkmale erfasst. Letztere bieten eine Korrelation mit der Dicke des Weichgewebes der Stirn. In zukünftigen Entwicklungen sollen die optischen Merkmale dazu verwendet werden, um den Einfluss des Weichgewebes zu kompensieren. Auf diese Weise soll eine hochgenaue Registrierung bezüglich der rigiden Schädeloberfläche erreicht werden.

In dieser Arbeit wird die Erfassung von punktbasierten Oberflächenrekonstruktionen behandelt. Um die Korrespondenz zwischen projiziertem und gemessenem Licht herzustellen, wird eine Methode für binäres zeitliches Multiplexing vorgeschlagen. Die entwickelte Methode kann für Objekte eingesetzt werden, welche sich mit geringer Geschwindigkeit bewegen. Weiterhin bildet die Methode die Basis zur Erreichung einer hohen Rekonstruktionsrate. Neben der hergestellten Punktkorrespondenz setzt die Triangulation von Oberflächenpunkten auch die Kalibrierung der extrinsischen Parameter des Laser-Scan Systems voraus. Aus diesem Grund wird eine Prozedur für die hochgenaue Kalibrierung von Laserstrahlen vorgeschlagen.

Die Analyse der Genauigkeit der Triangulation hat einen root mean square (RMS) Fehler von 0.1037 mm für die Rekonstruktion einer Ebene und einen RMS Fehler von 0.1477 mm für die Oberflächenrekonstruktion eines Treppen-Phantoms ergeben. Im Fall der Oberflächenrekonstruktion der Stirn von drei Personen, hat die Analyse RMS Fehler zwischen 0.1526 mm und 0.1832 mm ergeben.

Abstract

Optical head tracking for robot-guided radiosurgery offers essential advantages. It is based on harmless laser light and it provides a high localization rate. The crucial disadvantage of optical head tracking is given by limited accuracy. The problem is that the registration of surfaces is influenced by the deformable soft tissue. In order to enhance the accuracy of optical head tracking, our research group developed a novel measurement system. By analyzing the measured reflection of a grid of projected laser spots, a point-based reconstruction of the forehead's surface and optical features are acquired. The latter provide a correlation with the thickness of the soft tissue of the forehead. In future developments, the optical features shall be utilized to compensate for the influence of the soft tissue. In this manner, a highly accurate registration with respect to the rigid skull surface shall be achieved.

In this work, the acquisition of point-based surface reconstructions is addressed. In order to establish correspondence between the projected and the captured light information, a method for binary time multiplexing is proposed. The developed method can be used for objects that move slowly. Furthermore, the method provides the basis for a high reconstruction rate. Besides the established point correspondence, the triangulation of surface points also requires the calibration of the extrinsic parameters of the laser scanning system. For this reason, a procedure for the highly accurate calibration of laser rays is proposed.

The analysis of the triangulation accuracy revealed a root mean square (RMS) error of 0.1037 mm and 0.1477 mm for the reconstruction of a plane and the surface reconstruction of a stair phantom, respectively. In the case of the reconstruction of the surface of three human foreheads, the analysis revealed RMS errors between 0.1526 mm and 0.1832 mm.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Problem statement	3
1.3	Structure of the thesis	6
2	Previous research	7
2.1	Spatial multiplexing	9
2.1.1	De Bruijn sequences	10
2.1.2	M-arrays	13
2.1.3	Customized encoding approaches	15
2.2	Time multiplexing	16
2.2.1	Temporal codes	17
2.2.2	Temporal hybrid and shifting codes	19
3	Establishing point correspondence	21
3.1	Preliminary considerations	21
3.2	Design of a structured light pattern	24
3.3	Detection of interest points	32
3.3.1	Template matching	33
3.3.2	Ellipse fitting	35
3.3.3	Paraboloid fitting	38
4	Projective geometry	41
4.1	Camera model	41
4.1.1	Model of central projection	41
4.1.2	Lens distortion	44
4.1.3	Rectification of image points	46
4.2	Homography matrix	47
4.3	Pose estimation	50

4.3.1	Initial solution	51
4.3.2	Refinement	54
4.4	Triangulation	55
4.4.1	Linear triangulation	55
4.4.2	Non-linear triangulation	57
4.5	Epipolar geometry	57
5	Calibration and surface reconstruction	61
5.1	Explicit laser ray calibration	63
5.2	Interpolation of laser rays	68
5.3	Surface reconstruction	72
6	Accuracy analysis	75
6.1	Sources of error	75
6.2	Triangulation accuracy for rigid objects	77
6.2.1	Triangulation accuracy for a plane	78
6.2.2	Triangulation accuracy for a stair phantom	80
6.2.3	Results and discussion	82
6.3	Triangulation accuracy for human foreheads	97
6.3.1	Definition of an error measure	97
6.3.2	Results and discussion	98
6.4	Tracking accuracy	106
6.4.1	Definition of tracking accuracy	107
6.4.2	Results and discussion	109
7	Simulation	115
7.1	A simulation model	115
7.2	Results and discussion	119
8	Conclusion and future works	123
A	Mathematical supplement	127
A.1	Pose of a rigid body	127
A.1.1	Rotation matrix	128
A.2	Linear coordinate transformation	129
A.3	Moore-Penrose pseudoinverse	130
A.4	Singular value decomposition	130
A.5	Principal component analysis	131

A.6	Iterative closest point algorithm	133
A.6.1	Point-to-point distance	133
A.6.2	Point-to-plane distance	135
A.7	KD trees	137
Bibliography		139

1. Introduction

1.1. Motivation

In the field of intracranial radiation therapy, a variety of modalities for the treatment of brain metastases is offered. *Stereotactic radiotherapy* (SRT) is a widely used technique which incorporates a significantly invasive immobilization of the patient's head. The head of a patient that is treated by the *Gamma Knife* [94] is immobilized by using a *Leksell frame*. The stereotactic frame is rigidly attached to the head by means of four pins that are drilled into the outer surface of the patient's skull. Furthermore, the frame is also rigidly attached to the treatment device of the *Gamma Knife*. Since the patient's head is immobilized during the whole treatment, the radiation is precisely delivered to the target area inside the brain. The mechanical alignment accuracy of the *Gamma Knife* was analyzed by comparing the planned and the measured central position of a single-isocenter. The analysis revealed an accuracy of approximately 0.25 mm [78, 94].

A more comfortable approach for intracranial treatment is given by *image-guided radiation therapy* (IGRT) which can also be used for a large variety of other cancerous diseases inside the body. IGRT is based on the frequent localization of the target area by using different imaging modalities. When the target area is moving, the localization data is used to realign the treatment beam. In this manner, a motion compensation can be achieved. Often the target area is given by moving tissue inside the body and consequently the use of IGRT is absolutely necessary. In the case of intracranial treatment, the localization capability of IGRT is used to increase the patient comfort. Since the patient's head can be localized, an invasive stereotactic frame for the rigid immobilization of the head can be avoided. Nevertheless, the motion of the patient's head has to be as minimal as possible during the treatment. A common approach for IGRT immobilization is the use of a patient-custom thermoplastic mask [14, 60, 61]. The latter can be supported by an additional bite block, a head mold or a head cushion. This type of immobilization is less in-

vasive than a stereotactic frame but wearing a thermoplastic mask for a whole treatment session can still be uncomfortable for the patient. According to [81], common IGRT immobilization systems tend to inaccuracies (i.e. head movement) of approximately 1 mm for the mean intrafraction translational shift and 2 mm for the mean interfraction translational shift.

In the last years, a large variety of different IGRT technologies has been established [76]. The highly accurate localization of the patient's head can be achieved by using the rigid skull structure for registration between a reference imaging dataset and imaging datasets that are acquired during treatment. Accuray Inc. (Sunnyvale, CA, USA) developed *6D Skull Tracking* for their *CyberKnife system*. This method uses stereoscopic X-ray images in which the rigid bone structure of the head is segmented. Subsequently, the segmented bone structure is registered to a planning dataset. The latter is acquired by *computed tomography* (CT). The *On-board Imager* (OBI) by Varian Medical Systems Inc. (Palo Alto, CA, USA) uses *cone-beam computed tomography* (CBCT) for head localization based on bone structures. Experiments with anthropomorphic head phantoms revealed that the *CyberKnife system* and the OBI system achieve sub-millimeter tracking accuracy [19, 39]. The high accuracy of X-ray based localization methods is achieved at the cost of tracking speed. This is due to the fact that X-ray imaging has to be carried out at low frame rates to guarantee acceptable additional dose levels.

By contrast, optical head tracking offers much faster scan rates. Here, visible or infrared laser light is used to reconstruct the surface of the skin. However, this method is inaccurate since the registration process is influenced by deformable structures which are given by the skin and the muscles. In a clinical evaluation, the tracking accuracy of the *Sentinel 3D laser scanning system* (developed by C-RAD, Uppsala, Sweden) was analyzed for six different tumor locations including head and neck [58]. The tracking accuracy was calculated by comparing the tracking results of the *Sentinel system* with those obtained from *megavoltage computed tomography* (MVCT). The evaluation revealed large tracking errors of more than 1 mm.

The treatment of an area as delicate as the head requires highest treatment accuracy and so far optical head tracking systems are not able to offer a sufficient tracking accuracy in the sub-millimeter range. Nevertheless, optical head tracking is a very promising approach. First of all, optical head tracking uses harmless laser light for localization. Thus, an additional radiation exposure for imaging can be avoided. This also guarantees that a high scan rate can be achieved. In this manner, the realignment of the treatment beam can

be carried out more often than compared to X-ray based localization methods. These are strong arguments to make an effort towards the enhancement of the accuracy of optical head tracking.

1.2. Problem statement

In order to overcome the problems of optical head tracking, our research group developed a novel measurement system (see Fig. 1.1). The measurement system projects a grid of discrete infrared laser spots onto the forehead of a person. Afterwards, the captured reflection of the grid is used to obtain a point-based reconstruction of the forehead. Furthermore, the captured reflection is also utilized to compute a set of optical features for each captured laser spot [93]. The novelty of this approach is given by the correlation of each feature set with the thickness of the soft tissue. In future developments, the reconstructed surface points will be combined with the optical features to offer highly accurate registration with respect to the rigid skull surface. Here, the correlation properties of the optical features will be used to compensate for the soft tissue. This would be the first system that solely uses harmless laser radiation to offer registration with the rigid bone structure of the head. In this manner, tracking errors that are related to the deformable skin and muscles can be avoided.

The developed measurement system is presented in Fig. 1.1. A beam shaped 850 nm laser diode (Thorlabs L850P010) generates an infrared laser beam which is splitted by a pellicle beamsplitter (Thorlabs CM1-BP145B2). Afterwards, the laser beam is redirected by an AXJ-V20 closed-loop scanner (laserwinkel.nl) which consists of two rotatable galvanometer driven mirrors. The redirection of the laser beam offers the projection of a grid of discrete laser spots onto the forehead of a person.

When a laser spot is projected onto the skin surface then a portion of the incident light is reflected and the remaining light penetrates in the soft tissue. The infrared wavelength of the laser light guarantees that the light attenuation of the chromophores inside the soft tissue is much lower than compared to wavelengths that belong to the visible light spectrum [4]. A low light attenuation by the chromophores offers a high light penetration depth. Thus, a sufficient radiant power of the laser source guarantees that the laser light reaches the skull surface of the forehead. Due to light scattering inside the soft tissue, a small portion of the light travels back to the skin surface of the forehead. This backs-

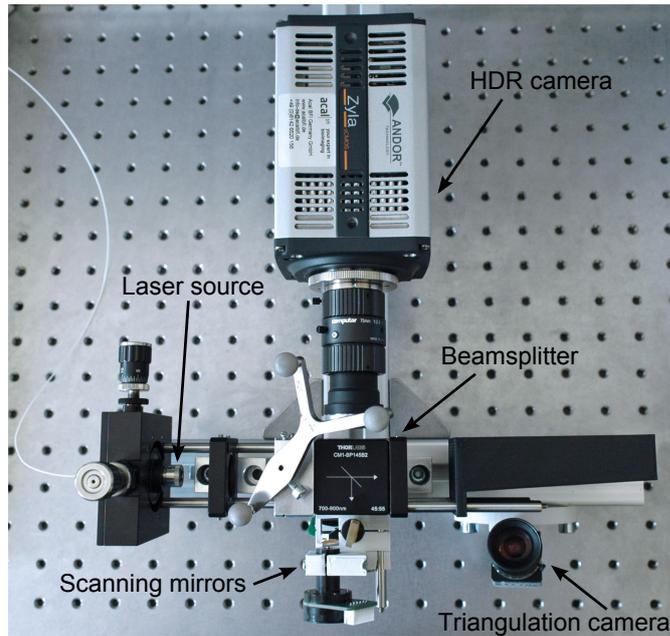


Figure 1.1.: Measurement system.

cattered light superimposes with the light that was initially reflected at the skin surface and the resulting light is captured by two cameras (see Fig. 1.1). For better clarity, the following descriptions use the simplified term reflected laser spot to refer to the described superimposed light.

A 16-bit *high dynamic range* (HDR) camera (ANDOR Zyla) captures the reflected laser spot. Utilizing the procedure described in [92], the resulting HDR image is processed to compute the explained set of optical features which offer a correlation with the thickness of the soft tissue. Besides to the HDR camera, the reflection of the laser spot is also captured by an IDS UI-3240CP-NIR-GL camera which provides 8-bit grayscale images of dimensions 1280×1024 and a maximum frame rate of 30 Hz in trigger mode. The captured reflection of the laser spot is used to reconstruct the center of the laser spot. This reconstruction is carried out by means of triangulation. For this reason, the second camera is also referred to as triangulation camera.

As explained, the two galvanometer driven mirrors are used to project a grid of discrete laser spots onto the forehead of a person. The triangulation camera operates in trigger mode and captures the reflection of the whole grid of laser spots in one single image. In this manner, a fast reconstruction of the corresponding spatial points can be achieved.

The reconstruction is carried out by means of triangulation which requires the mapping between the projected and captured light information. This is also known as the correspondence problem [6, 45].

Due to partial pattern occlusion, it can not be guaranteed that all laser spots of the projected grid are captured by the triangulation camera. Moreover, the successful detection of the captured laser spots depends on the robustness of the used point detection algorithm. As a consequence, the correspondence problem can not be solved directly. To address the correspondence problem appropriately, a structured light encoding technique has to be utilized.

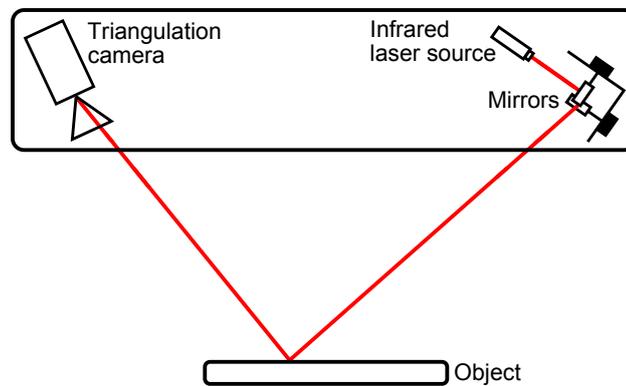


Figure 1.2.: This figure shows a diagram of the laser scanning system. The laser source generates an infrared laser beam which is redirected by two galvanometer driven mirrors. The outgoing laser beam hits the surface of an object and the reflection of the beam is captured by the triangulation camera.

This work analyzes existing structured light techniques and proposes a method to solve the correspondence problem for the described laser scanning system. The laser scanning system can be considered as a sub-system of the measurement setup presented in Fig. 1.1. The laser scanning system consists of the infrared laser source, the two galvanometer driven mirrors and the triangulation camera. A diagram of the system is shown in Fig. 1.2. Next to the established point correspondence, the triangulation of surface points also requires the calibration of the extrinsic parameters of the laser scanning system. For this reason, this work also proposes a method for the accurate calibration of laser rays. The purpose of the laser scanning system is to achieve a highly accurate point-based surface reconstruction of the forehead of a person. The resulting triangulation accuracy is analyzed and compared to the triangulation accuracy of a commercially available product. Furthermore, the developed laser scanning system was used for tracking. In this manner,

it is possible to determine how the process of tracking is influenced by the remaining triangulation errors.

1.3. Structure of the thesis

The thesis is structured as follows. In Sec. 2, the previous research in the field of structured light is reviewed. Sec. 3 describes how correspondence is established between the projected and the captured light information. In Sec. 4, models of projective geometry are described. This is the basis for the calibration and triangulation methods which are proposed in Sec. 5. The analysis of the accuracy of the triangulation is carried out in Sec. 6. In order to verify the experimental results, a model for the simulation of the laser scanning system is described in Sec. 7. In Sec. 8, the conclusion and thoughts on future works are presented.

2. Previous research

In the context of triangulation, light can be passively or actively used to reconstruct the surface of an object [72]. Passive methods commonly use two cameras which are displaced relative to each other to capture a scene which overlaps in both camera images. Triangulation is realized by using the projective geometry of both cameras which depends on the intrinsic and extrinsic camera parameters. Furthermore, corresponding image points have to be detected in the overlapping camera images. The result of triangulation is given by the corresponding spatial points located on the observed object's surface. The main disadvantage of this approach is that the detection of corresponding image points becomes complicated for textureless surfaces.

This problem can be solved by active methods which replace one of the cameras by a light projector. This approach is known as structured light and involves the establishment of the correspondence between the projected and captured light information. According to [17], the following types of structured light systems can be classified (see also Fig. 2.1).

- **Single spot:** A light spot is swept across the object's surface and the camera captures an image for each single spot. The correspondence between the projected and captured spots is directly given.
- **Single slit:** A light slit is swept across the object's surface and the camera captures an image for each single slit. The correspondence between the projected and captured slits is directly given.
- **Multiple slits:** An array of parallel light slits is projected onto the object's surface and the camera captures the whole pattern by means of one single image.
- **Grid:** A light grid is projected onto the object's surface and the camera captures the whole grid by means of one single image. The projected grid is either given by a

grid of discrete spots or a grid of connected lines.

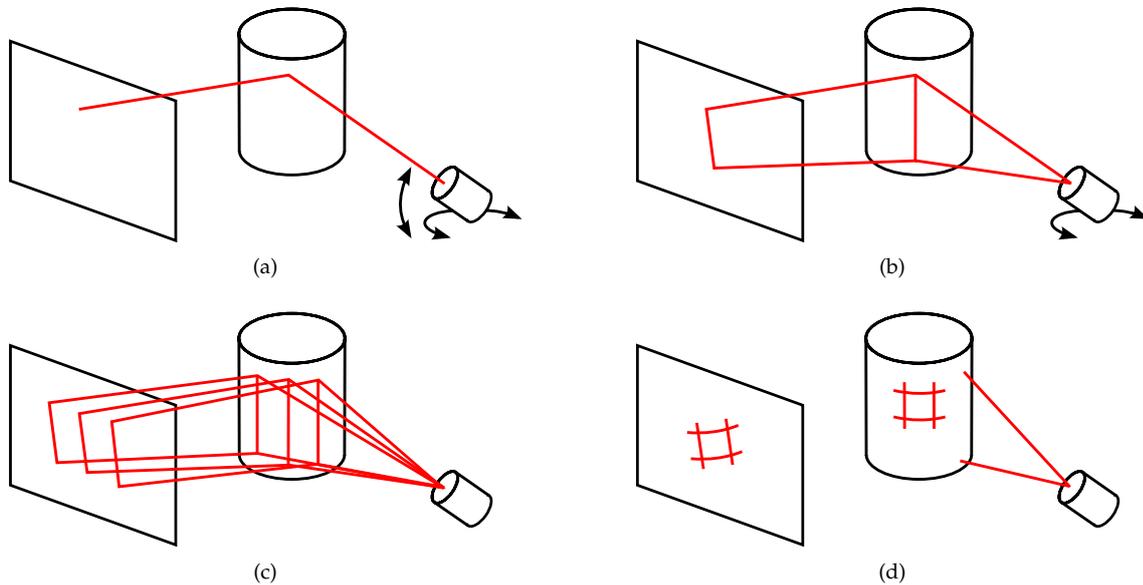


Figure 2.1.: Classification of structured light systems: (a) single spot, (b) single slit, (c) multiple slits, (d) grid. Here, image (d) only contains the projected and the captured grid without showing the path of the light rays.

Common cameras offer rates of 30 to 60 frames per second. Consequently, the reconstruction of the surface of an object can take several seconds if the scanning system follows the described *single spot* or *single slit* approach. In contrast, scanning systems that follow the *multiple slits* or *grid* approach provide a much faster reconstruction rate since the whole light pattern is captured in one single camera image. Nevertheless, partial pattern occlusion can lead to images where parts of the projected pattern are missing. Hence, such images do not offer a direct correspondence between the projected and captured light information. For this reason, *multiple slits* and *grid* approaches always involve encoding strategies to solve the correspondence problem. A broad range of encoding strategies will be introduced later in this section. Before, a brief overview of the related hardware components of common light projection systems will be given.

The most simple light projector can be realized by a line-based laser source which is tilted to sweep a laser line across the object's surface [90]. Regarding the above listed taxonomy, this approach is appropriate for the *single slit* or *multiple slits* approach. The projection of an arbitrary pattern is typically realized by an off-the-shelf *liquid-crystal display* (LCD) projector or *digital light processing* (DLP) projector [35]. A less cost-intensive alternative

is given by the use of a *diffractive optical element* (DOE) which shapes a single laser spot to project a desired pattern [3]. However, a DOE is a static component which is designed for the projection of exactly one light pattern.

As realized in this work, the projection can also be carried out by using a point-based laser source and two controllable mirrors which redirect the laser beam. This approach guarantees a high flexibility since arbitrary light patterns can be projected onto the surface of an object. Compared to off-the-shelf LCD and DLP projectors, this approach also offers the advantage that the light source can be selected individually to comply with the specifications of the scanning application.

In the following, a broad range of structured light techniques is introduced. A very detailed state of the art in structured light is also presented in [72]. As explained, the measurement system shown in Fig. 1.1 projects a grid of discrete laser spots onto the forehead of a person. For this reason, the following sections concentrate on previous research in the field of discrete encoding methods. Although the measurement system uses a single infrared wavelength for the projection, the following sections also cover color based encoding. Furthermore, also slit and stripe based projection patterns are covered. This is necessary in order to examine the diversity of general encoding topics which are incorporated by previous methods. In this manner, the conclusions of the previous methods can be used in Sec. 3 to design an appropriate light encoding for the measurement system.

2.1. Spatial multiplexing

Spatial multiplexing covers methods which incorporate the encoding of light by means of the projection of one single pattern. This type of projection is also referred to as one-shot pattern [24]. A one-shot pattern is captured in one single camera image and thus a high reconstruction rate can be achieved. Spatial multiplexing uses the local neighborhood of each projected light element to encode the position in the pattern by means of intensity, color, geometric features or distance information. The following sections introduce three different classes of spatial multiplexing given by de Bruijn sequences, M-arrays and customized encoding approaches.

2.1.1. De Bruijn sequences

A k -ary de Bruijn sequence of order n is a sequence of length k^n which contains each possible k -ary sequence of length n exactly once [84]. The latter is also known as the window property [63]. In the following, all possible k -ary sequences of length n are also referred to as sub-sequences of a de Bruijn sequence. The number m of all possible k -ary de Bruijn sequences of order n is calculated by [84]:

$$m = ((k - 1)!)^{k^{n-1}} k^{k^{n-1}-n}. \quad (2.1)$$

The construction of a de Bruijn sequence can be achieved by the computation of the Hamiltonian cycle in the respective de Bruijn graph. The Hamiltonian cycle visits each node in the graph only once. Using for example $k = n = 2$, the de Bruijn sequence is given by 0011 [13]. The latter contains the possible binary sub-sequences 00, 01, 10 and 11 exactly once. In this case, it holds that $m = 1$ and thus only one de Bruijn sequence exists. Another example is given by the configuration $k = 2$ and $n = 3$ that leads to $m = 2$. In this case, the de Bruijn sequences are given by 00010111 and 11101000. Each of them include the possible binary sub-sequences 000, 001, 010, 011, 100, 101, 110 and 111 exactly once.

For structured light systems, a de Bruijn sequence can be used to encode a projected array of colored light slits or stripes [72]. An array of colored light slits can be defined as the projection of multiple thin lines with equidistant spacing. In contrast, an array of colored light stripes can be defined as the projection of multiple thick lines without spacing. In the case of light stripes, the edges of the stripes are used for reconstruction. In general, the use of light stripes provides the acquisition of a reconstruction with a higher density. This is due to the fact that this approach does not include gaps between the projected elements. However, light stripes provide a lower detection accuracy because of the merging of colors at the stripe edges.

The parameter k defines the number of colors that are used to encode an array of light slits or stripes. Since each possible k -ary sub-sequence of length n appears only once inside a de Bruijn sequence, the correspondence between the projected and captured elements can be uniquely determined by analyzing the local neighborhood for each light element in the camera image [63]. More precisely, each element can be identified by inspecting

the $n - 1$ adjacent elements.

According to [63], de Bruijn sequences are the most used strategy for one-shot patterns. [11] proposed the projection of de Bruijn sequences based on stripes using the colors red, green, blue and white. Different values for n led to different amounts of projected stripes. In [53], a slit based de Bruijn sequence using six colors was realized. A color calibration for the camera was carried out in order to compute the transfer function of the camera. The results of a color calibration can be used to correct the scaling of the color channels which is caused during the capturing process. After pattern projection, the captured image was transformed from RGB (red, green, blue) space to HSV (hue, saturation, value) space. In this manner, the hue value was used for simple segmentation of the captured light slits. A graph based search algorithm was utilized to find the most probable correspondence between the projected and captured light slits.

[11] and [53] are both early works in the context of structured light systems. [11] mentions one of the main disadvantages of color coding. The color of the captured light in the camera image also depends on the albedo (reflection coefficient) of the object whose surface has to be reconstructed. In the case of colored objects, color components of the projected light can be subject to attenuation. This can lead to a high amount of errors when the codes of the captured light elements have to be determined. In the worst case, unnoticed code detection errors cause local errors in the surface reconstruction.

The same problem can occur in the case of partial pattern occlusion. A captured light element can be identified if it's $n - 1$ adjacent elements are available. Consequently, the appearance of partial pattern occlusion does not necessarily lead to code detection errors. Code detection errors only occur if wrong adjacent elements are utilized for the code detection. This problem can be avoided when partial pattern occlusion can be noticed. However, the detection of small holes within a pattern can not always be guaranteed. Small holes within a pattern occur when single light elements are missing. Unnoticed partial pattern occlusion is a problem for all spatial multiplexing approaches.

A disadvantage of de Bruijn sequences in general is given by the trade-off between the length of a de Bruijn sequence and the length of the included sub-sequences [53]. In order to achieve a high number of reconstructed light elements, the projected de Bruijn sequence has to provide a high length. The latter can be increased by increasing the parameters k and n . Especially the increase of n is not desirable since this will also increase the number of $n - 1$ adjacent elements that have to be identified correctly. Hence, an

increase of n increases the probability of code detection errors.

To increase the reconstruction density and the robustness of slit detection, a grid of horizontal and vertical colored slits was proposed in [71]. Six different colors were used to encode the projected slits in both axes. The colors red, green and blue were utilized to encode the horizontal slits. Furthermore, the colors magenta, cyan and yellow were used to encode the vertical slits. Captured images were transformed from RGB space to HSI (hue, saturation, intensity) space to simplify the segmentation of the captured light slits. The crossings of the horizontal and vertical slits were extracted by peak intensity detection. Subsequently, the crossings were used to find the correspondence between the projected and captured grid.

[96] proposed the projection of a stripe-based de Bruijn sequence. To increase the robustness of the code detection, an adapted de Bruijn sequence was created. The latter incorporates the constraint that two successive stripes can not have the same color. As a side effect, the length of a de Bruijn sequence and hence the number of projected stripes is reduced. A compensation is achieved by increasing the amount k of colors used for encoding. However, a high number of different colors can be a drawback. As explained, the color of the captured light also depends on the albedo of the object. Resulting code detection errors can even be intensified when the number of used colors is increased.

A similar approach compared to [96] was published in [18] and [16] which created an adapted de Bruijn sequence based on stripe and slit projection, respectively. Both methods use an adapted de Bruijn sequence that is subject to the constraint that the color of two successive stripes or slits has to differ in at least two color channels.

[63] combined the projection of a slit- and stripe-based array. In this manner, the high reconstruction density of light stripes could be combined with the high detection accuracy of light slits. The HSI space was used to construct a de Bruijn sequence of half illuminated colored light stripes. Dark and bright light slits were nested in each sub-sequence of the de Bruijn sequence.

So far, all of the introduced de Bruijn methods were based on color encoding. In [83], a different approach was realized by using a distance based encoding. A grid of horizontal and vertical light slits was projected onto the surface of an object. The color of all horizontal and vertical slits was given by blue and red, respectively. The use of red and blue slits does not provide a color encoding. Instead these colors were solely utilized to distin-

guish the horizontal and vertical slits. Both, the horizontal and vertical spacing between the light slits, followed a de Bruijn sequence. By using a distance based encoding, the described drawbacks of color encoding can be avoided.

However, distance based encoding is not robust against perspective distortion of the encoded distances in the camera image. This distortion depends on the position and orientation of the involved components. The latter are given by the light projector, the camera and the object whose surface has to be reconstructed. A perspective distortion of the distances corrupts the encoding and thus code detection errors can occur.

2.1.2. M-arrays

The equivalent of de Bruijn sequences in the 2-dimensional space is given by perfect maps which are also known as pseudorandom sequences [15, 47]. A perfect map is a matrix M of dimensions $r \times v$ in which the elements are taken from an alphabet k and in which a sub-matrix of dimensions $n \times m$ is contained exactly once (window property). The number of sub-matrices included in M is given by $rv = k^{nm}$. To avoid the occurrence of the sub-matrix which solely contains zero-elements, M can be designed to provide $rv = k^{nm} - 1$ sub-matrices. In this case, the perfect map M is referred to as M-array [3].

[57] suggested the projection of a binary 91×45 M-array using sub-matrices of dimensions 3×4 . The projected pattern contains white spots on a black background. Thus, the black spots of the M-array are not visible. To detect these black spots, another pattern has to be projected in advance. This pattern includes 91×45 white spots on a black background at the same position. By detecting all white spots in the camera image, the black and white spots of the M-array can be detected in the second camera image. In general, all spatial multiplexing approaches are one-shot methods. This also holds in the case of M-array approaches. However, the method proposed in [57] is an exception since it requires a preceding pattern to detect the encoded spots. As described in Sec. 2.1.1, unnoticed partial pattern occlusion is a problem for all spatial multiplexing approaches. This is also valid in the case of M-array approaches.

[23] projected a M-array containing 11×29 spots of three colors red, green and blue. Each sub-matrix contained five elements using the four adjacent neighbors for every position. The problems of color-based de Bruijn sequences described in Sec. 2.1.1 also apply for color-based M-arrays. The albedo of the object can cause code detection errors. To avoid

this drawback, [23] also designed a monochromatic 27×127 M-array based on five different geometric features. The basic feature was given by a square with four blank quadrants and the remaining four features were defined by filling one to four quadrants of the square.

To increase the robustness of code detection, [55] proposed a M-array which makes use of a voting approach and the Hamming distance between the codes. The used encoding is either based on color or geometric features and supports sub-matrices of dimensions 3×3 . To detect the code of an element in a captured M-array, the respective position is analyzed nine times because it is a member of nine different but overlapping 3×3 sub-matrices. Consequently, a vote can be conducted to determine the correct code.

The Hamming distance between two code words is defined as the number of positions in which they differ [66]. To further increase the robustness of code detection, [55] defined a minimum Hamming distance d which has to be satisfied between each adjacent sub-matrix during the creation of the M-array. In the case of an error that occurs in one of the code digits of a sub-matrix, the computed Hamming distance will be equal or greater than $d - 1$. The receiver knows that allowed Hamming distances are equal or greater d and thus incorrect Hamming distances given by $d - 1$ can be detected. A higher minimum Hamming distance d provides a better error detection. The reason is that the number of possible $d - 1$ Hamming distances will increase when d is increased. However, the increase of d also leads to the decrease of the supply of allowed sub-matrices which have to be unique in the M-array. Thus, the increase of d is limited.

[64] utilized the M-array approach proposed in [55] for robot positioning by means of *visual servoing*. The method uses a 20×20 M-array and sub-matrices of dimensions 3×3 which contain red, green and blue spots. Experimental results showed that the constructed M-array can be used to successfully position a six degrees of freedom (DOF) robot manipulator with respect to an object. The main drawback of the created M-array is that it does not provide a rotation invariant detection of the codes. The M-array constructed in [55] can only be used for cases that do not involve a high rotation of the grid in the camera image.

To avoid the drawbacks of color-based spatial multiplexing, [3] proposed a monochromatic M-array which is constructed by using three geometric features. The latter are given by a disc, a circle and a stripe (see Fig. 2.2). The encoding is realized by a M-array of dimensions 27×29 and sub-matrices of dimensions 3×3 . Furthermore, the approach

in [3] uses the constraint of a minimum Hamming distance $d = 4$ which has to be satisfied between each adjacent sub-matrix. The algorithm used for the construction of the M-array achieved that approximately 96% of the codes satisfy the specified constraint. The shape of the stripe feature shown in Fig. 2.2 offers the computation of the orientation of the captured M-array grid. A surface reconstruction is computed by using the center of mass of each geometric feature. The experimental results showed that the proposed M-array grid works robust in most cases. However, the effect of unnoticed partial pattern occlusion is still a problem.



Figure 2.2.: Geometric features used in [3].

2.1.3. Customized encoding approaches

This section covers techniques for spatial multiplexing which use types of non-orthodox encoding. [33] suggested a monochromatic approach based on the projection of a field of connected square cells. One out of three different intensity values was assigned to each square cell. The superimposed light at the intersection (node) of each set of four cells led to 18 possible intensity values. For the identification of a node in the image, the intensity value of the node and its four adjacent nodes was evaluated. To avoid ambiguity, [33] used the epipolar line constraint which incorporates that a projected node is associated with an epipolar line in the captured camera image. A node in the image can be verified by evaluating the distance to the epipolar line. As described, [33] utilized an intensity based encoding strategy. This approach clearly suffers from the same disadvantages as color based spatial multiplexing.

In [41], a coding strategy that purely relies on the epipolar constraint was realized. The projected pattern contains an array of black vertical slits on a white background. The black slits are crossed by an array of green diagonal slits. Each crossing between the black and green slits in the captured camera image is associated with an epipolar line. A defined angle between the black and green slits provides that the mapping between the captured crossings and the epipolar lines can be uniquely resolved. Consequently, the identification of a crossing does not depend on other crossings in the local neighborhood. In this manner, the problems of unnoticed partial pattern occlusion can be completely avoided. This is a big advantage compared to common techniques for spatial

multiplexing.

However, a disadvantage of the epipolar line based coding is given by the trade-off between reconstruction density and accuracy. A higher reconstruction density can be achieved by adding more green slits. Hence, the angle between the black and green slits has to be decreased. This strategy is limited since a decreasing angle also decreases the distance between the epipolar lines. When the distance between the epipolar lines is too short then the mapping between the captured crossings and the epipolar lines becomes ambiguous. The reason is the presence of image noise.

A different approach was realized in [51] which suggested a distance based encoding. An array of black slits on a white background is projected onto the surface of an object. To implement the distance encoding, each black slit is cut at random positions. In this manner, each slit consists of a number of slit segments which have a random length. The code detection is realized by analyzing the length of a slit segment and its six adjacent slit segments in the projected array. The explained method clearly implies the same disadvantage as the distance based de Bruijn method described in Sec. 2.1.1.

[38] proposed the projection of a grid of horizontal and vertical light slits. The color of all horizontal and vertical slits was given by blue and red, respectively. The encoding was realized by random based spacing between the horizontal slits. An algorithm for peak detection was utilized to localize the intersections of the horizontal and vertical slits. Afterwards, the intersections and the encoded distances in the local neighborhood were used to find the correspondence between the projected and captured grid. Since distance based encoding is utilized, the explained method suffers from the same disadvantage as the procedure in [51].

2.2. Time multiplexing

Time-multiplexed structured light covers methods which realize light encoding by successive pattern projection. By means of a k -ary code, the position of $N = k^n$ light elements can be encoded in n projected patterns. The radix k defines the possible projection states which are usually realized by means of change in intensity or different colors. A camera captures the reflection of each pattern in one single image. Consequently, the projection of n patterns results in n captured camera images. The following sections cover

general temporal codes, temporal hybrid codes and temporal shifting codes.

2.2.1. Temporal codes

An early work in the area of time multiplexing is proposed in [67]. An array of 2^n light stripes is encoded by means of the projection of n patterns. The binary encoding is realized by switching the light source on or off. The first projected pattern corresponds to the most significant bit (MSB). Consequently, the proposed approach represents a coarse-to-fine scheme. As mentioned in Sec. 2.1, light stripes are not separated by gaps and hence light stripes will merge when they provide the same projection state. The first projected pattern is coarse since the stripes merge to two blocks. The patterns that follow after the first one provide a finer partitioning of the light stripes since the alternation of the projection states becomes larger towards the least significant bit (LSB). In the last pattern the projection state alternates for each adjacent stripe. In this context, it should be mentioned that the order of the patterns has no influence. The same reconstruction results can be achieved when the projection starts with the LSB.

The code of the captured stripes can be retrieved as follows. At first, the stripe edges are detected in the last camera image. Afterwards, the position of the detected edges are utilized in all n images to analyze the intensity between them. The method explained in [67] can only be used for static applications which guarantee that the object whose surface has to be reconstructed does not move during the projection of the n patterns. This is a contrast to approaches which are based on spatial multiplexing. The use of a one-shot pattern provides fast surface reconstruction and is applicable for moving objects.

The advantage of time-multiplexed structured light is given by accuracy. As described in Sec. 2.1, the accuracy of the code detection for methods based on spatial multiplexing is affected in the case of unnoticed partial pattern occlusion. This problem does not exist for methods based on time multiplexing. This is due to the fact that the encoding of each projected light element is independent of adjacent light elements. The accuracy of the code detection for methods based on time multiplexing can only be influenced by the robustness of the algorithm that is responsible for the extraction of the captured light elements.

As proposed in [31], the robustness of binary time multiplexing can be further increased by introducing the Gray code [22]. By using the Gray code, the difference between the

code of adjacent light stripes is given by exactly one shuffled bit. This is equivalent to a Hamming distance of value 1 and provides a guaranteed detection in case of an error that occurs in one of the code digits. In the case of an array of light stripes, the conversion from the usual binary code to the Gray code is loss-free. This means that all 2^n light stripes can be reordered so that the Hamming distance between the code of adjacent light stripes is given by the value 1. Preserving the number of stripes is an advantage. Otherwise the number n of projected patterns would have to be increased in order to compensate for unusable stripes.

Another approach to further increase the robustness of binary time multiplexing is realized in [52]. The Hamming distance between the code of adjacent light stripes is maximized in order to detect or even correct erroneous codes that are extracted from the captured camera images. However, this approach also leads to the disadvantage that not all 2^n light stripes can satisfy the constraint. As described before, this leads to an increase of the number n of projected patterns.

[12] proposed the use of color encoding to increase the number of projected stripes. As described in Sec. 2.1, the use of color encoding can be problematic because of the albedo of the object. To avoid this problem, [12] suggested an intensive color calibration for the optical system. The following calibration steps are executed offline. A color calibration for the camera is carried out to compute the transfer function. Furthermore, an image of the scene is acquired which only contains the ambient light. The next calibration step has to be carried out online because it depends on the color properties of the object whose surface has to be reconstructed. A white pattern is projected onto the surface of the object and the reflection is captured by the camera. The albedo of a colored object can change the reflected light from white to different colors. Using the white light information and the measured colors in the camera image, a transfer function can be calculated for the relevant pixels in the camera image.

After calibration, the projection of the color encoded light stripes can be carried out. Using the calibrated parameters, the color of the relevant pixels in the n camera images can be corrected to compensate for the influence of the camera, the ambient light and the albedo of the object. Consequently, the corrected colors provide a very close approximation of the projected colors. In this manner, the code detection can not be affected by the use of color encoding. The disadvantage of this approach is given by the fact that the described calibration is time consuming. Moreover, the calibration of the albedo transfer function has to be carried out online for every new object. This is achieved by using an

additional pattern which contains white light. Hence, the number n of projected patterns is extended by 1.

2.2.2. Temporal hybrid and shifting codes

[32] proposed a hybrid approach which combines spatial and time multiplexing. A one-shot pattern provides fast surface reconstruction and it is applicable for moving objects. On the other hand, time multiplexing provides robustness in the case of partial pattern occlusion. The method described in [32] makes an effort to combine the advantages of both approaches.

The basis is given by the binary encoding of 2^n light stripes. To provide spatial multiplexing, a projected pattern contains the complete code of length n for each binary stripe. The code of a stripe is projected along the x-axis and the codes of all stripes are stacked along the y-axis. The time multiplexing is realized by the successive projection of n patterns. Each new pattern contains the previous pattern which is modified by means of a code shifting of 1 bit along the x-axis. In this manner, the algorithm can automatically switch between spatial and time multiplexing. It uses time multiplexing if the object whose surface has to be reconstructed is not moving. In this situation, a highly accurate code detection can be achieved. The algorithm switches to spatial multiplexing as soon as the object starts to move.

Since the same stripe encoding is used for both approaches, the switching to spatial multiplexing does not change the sequence of projected patterns. As soon as the algorithm detects that the object does not move anymore it switches back to time multiplexing. Code detection errors in the case of spatial multiplexing are reduced since the Gray code is utilized in [32].

Another approach for time multiplexing is given by the shifting of the projected patterns. The goal is to obtain a reconstruction with a higher density. [73] proposed a shifting method based on the projection of binary stripe patterns. The method utilizes the Gray code and the projection order follows a coarse-to-fine scheme from MSB to LSB. After the projection of all patterns, the LSB pattern is repeated three times. Each time this pattern is slightly shifted and thus encoded regions have to be tracked in the camera image. By means of the shifting, the stripe edges incorporate a translation and hence a higher reconstruction density can be achieved.

3. Establishing point correspondence

The measurement system shown in Fig. 1.1 projects a grid of discrete laser spots onto the forehead of a person. The triangulation camera captures the reflection of the laser spots in order to reconstruct their centers. For the purpose of reconstruction, the correspondence between the projected and captured laser spots has to be established. Due to partial pattern occlusion, it can not be guaranteed that all laser spots of the projected grid are captured by the camera. The occlusion of parts of the projected pattern can either be caused by the individual shape of the persons's forehead or wrinkles of the skin. Partial pattern occlusion also occurs when the projected grid of laser spots exceeds the forehead area.

Besides partial pattern occlusion, the processing of the camera images has to be considered. The successful detection of the captured laser spots depends on the robustness of the used detection algorithm. Due to partial pattern occlusion and detection issues, the correspondence problem can not be solved directly. For this reason, a structured light encoding technique is developed in the following. Furthermore, algorithms for the detection of interest points are addressed.

3.1. Preliminary considerations

As a basis for the development of a structured light method, the most important advantages and disadvantages of the discrete encoding methods described in Sec. 2 are summarized. Spatial multiplexing offers the projection of a one-shot pattern and the encoding relies on the local neighborhood of each projected light element. Spatial multiplexing allows for a high reconstruction rate and can also be applied for moving objects. The two major classes of spatial multiplexing are given by de Bruijn sequences and M-arrays. Methods based on de Bruijn sequences mainly use the projection of an array of

multiple light slits or stripes. Moreover, some of the proposed methods based on de Bruijn sequences also use the projection of a grid of horizontal and vertical slits. The matrix structure of M-arrays presents a scheme which is applicable for the encoding of a grid of discrete spots. Another class of spatial multiplexing is covered by customized encoding approaches which do not follow a formal encoding paradigm. Proposed methods in this area offer pattern projection based on spots or slits.

Most spatial multiplexing approaches utilize color encoding. However, the albedo of the object whose surface has to be reconstructed can lead to code detection errors. To decrease this effect an intense color calibration has to be carried out. Other spatial multiplexing approaches rely on the use of distance information for the encoding. A drawback is given by the distortion of the encoded distances in the camera image. A more robust encoding scheme is given by the use of geometric features. The latter can be used for M-arrays since the properties of these features imply the encoding of a grid of discrete spots. The use of appropriate geometric features also offers the advantage that the orientation of the M-array can be detected in the camera image. A general disadvantage of all spatial multiplexing approaches is the lack of robustness against unnoticed partial pattern occlusion.

The use of a M-array based on binary encoded spots is another alternative to avoid the drawbacks of color encoding. Nevertheless, a M-array based on binary encoded spots also necessitates the projection of a second pattern in order to detect the encoded spots that use the same color as the background. By means of a specified minimum Hamming distance between the code words, the robustness of the code detection can be further increased for all spatial multiplexing approaches. Another strategy to further increase the robustness of the code detection is given by means of epipolar lines.

This technique also offers the possibility to develop a spatial multiplexing approach that solely relies on epipolar lines. In this case, the pattern projection is designed so that each interest point in the camera image is uniquely mapped to an epipolar line. This principle offers the big advantage that the set of interest points can be encoded without the use of the local neighborhood. In this manner, the consequences of unnoticed partial pattern occlusion can be completely avoided. Hence, the sole use of epipolar lines is a promising strategy to construct a very robust one shot pattern. Nonetheless, also this approach contains a critical drawback. In the presence of image noise, epipolar lines have to be located at least 10 pixels apart from each other. Only in this manner, a unique mapping between interest points and epipolar lines can be guaranteed. Consequently, only a small

amount of epipolar lines can be utilized which ultimately results in a low reconstruction density.

Next to spatial multiplexing, the second branch of discrete encoding approaches is defined by time multiplexing. The latter incorporates light encoding by means of the successive projection of n patterns. The successive pattern projection enables that each projected light element is encoded independently of the adjacent light elements. Consequently, partial pattern occlusion has no effect on the code detection and hence a highly accurate surface reconstruction can be achieved. This is a big advantage compared to spatial multiplexing. On the other hand, n patterns have to be projected in order to detect the code of each light element. Consequently, surface reconstruction based on time multiplexing incorporates a higher acquisition time. After the detection of the code words, surface reconstruction can be carried out. Typically, the positional information of the last captured pattern is utilized for surface reconstruction.

Moreover, existing time multiplexing approaches can only be used for static applications which ensure that the object does not move during the projection of the n patterns. As in the case of spatial multiplexing, the robustness of time multiplexing approaches can be further increased by incorporating the use of the Hamming distance between the code words. Furthermore, the robustness can be increased by using the epipolar line constraint. In the literature, time multiplexing is mainly used for the projection of an array of multiple light stripes. Since the encoding of each projected light element is independent from the adjacent light elements, time multiplexing can be easily transferred to the encoding of a grid of discrete spots.

Next to the class of pure temporal codes, time multiplexing also covers temporal hybrid codes and temporal shifting codes. The former is useful for applications which involve an object that is moving on an alternating basis. A hybrid approach uses time multiplexing when the object is not moving. In this situation, a highly accurate surface reconstruction can be obtained. When the object starts to move then the method switches to spatial multiplexing. Approaches which follow the paradigm of temporal shifting codes are an extension of pure temporal codes and can be utilized to further increase the reconstruction density.

3.2. Design of a structured light pattern

To establish point correspondence for the grid projection of the measurement system shown in Fig. 1.1, binary time multiplexing is utilized in this work. The reason to decide against spatial multiplexing is that all spatial multiplexing approaches lack robustness against unnoticed partial pattern occlusion. To achieve a highly accurate surface reconstruction, the only proper spatial multiplexing approach would be the sole use of epipolar lines. Nevertheless, the disadvantage is that only a small amount of interest points can be uniquely mapped to epipolar lines. Consequently, this approach leads to a sparse surface reconstruction. A successive projection of shifted patterns could solve this problem. However, a successive projection can drastically affect the coherence of reconstructed surface points if the head of the person is moving.

Instead, binary time multiplexing is utilized in this work. Also this approach involves successive pattern projection. The difference is that time multiplexing utilizes the image data from only one captured pattern for surface reconstruction. The image data of the remaining patterns is used for code detection. Thus, the coherence of reconstructed surface points solely depends on one projected pattern.

Previous time multiplexing approaches can not be used in the case of moving objects. Consequently, previous time multiplexing approaches can not be used in the case of optical head tracking for radiation therapy. The time multiplexing method developed in this work reduces this restriction. The developed method can be used for objects that move slowly. Moreover, the developed method offers a high reconstruction rate. The constraint of slow movement is satisfied during optical head tracking for radiation therapy. The head of a person rests inside a head mold which offers only little space for head movement.

The developed time multiplexing method utilizes binary encoding. Consequently, two encoding states exist which are realized by switching the laser source on or off. As an alternative, the laser source could also be used to create laser spots of different intensities. In this manner, the amount of encoding states could be greater than two. For example, three encoding states given by 0 %, 50 % and 100 % of laser intensity could be realized. In the context of time multiplexing, a larger amount of encoding states k offers a reduction of the amount n of projected patterns. However, an intensity based encoding suffers from the same problems as discussed for color based encoding. Similar to color calibration, an

intensity calibration could be carried out for an intensity based encoding. However, this is not a practical approach in the scope of head tracking. Instead, binary encoding is utilized to guarantee a clear distinction between the encoding states.

Before the developed time multiplexing method is described, the structural and time-related properties of the grid projection are addressed. The projection of a single grid of laser spots requires a certain time period t which lasts from the moment when the first laser spot hits the surface to the moment when the last laser spot hits the surface. The value of t depends on the used projection hardware. In this work, the projection system is given by a point-based laser source and two controllable mirrors which redirect the laser beam.

The used projection system is able to project a grid of $N = 1000$ discrete laser spots in approximately $t = 0.1$ s. Using a grid resolution of 40×25 laser spots, the dimensions of the projected grid are given by approximately 16×10 cm². These dimensions apply for a projection distance of 60 cm which is also the specified projection distance for the future head tracking system.

The grid dimensions of 16×10 cm² are used as an estimate of the average forehead size of a person. The chosen grid dimensions depend on the number N of projected laser spots and the spot size. The gap between the laser spots is as small as possible and guarantees that the projected laser spots are clearly distinguishable in the camera image. The size of the laser spots depends on the optical properties of human skin. Due to the high amount of diffuse reflection, the spot size on human skin is larger than compared to the spot size on synthetic material. At a projection distance of 60 cm, the diameter of each laser spot is given by 2 mm. The gap between the laser spots is defined by 2 mm in both directions.

The time period t depends on the velocity of the mirrors and how fast the laser source can be switched on and off. In this work, the laser source is switched during the movement of the mirrors. The movement of the mirrors is carried out in intervals. This is necessary since the slow switching characteristics of the used laser source prevent a high velocity of the mirrors. If the mirrors would not stop at the end of an interval then the projection would result in laser line fragments instead of laser spots. During each movement interval of the mirrors, five laser spots can be projected.

An increase of the number N of projected laser spots linearly increases the time period t as well as the dimensions of the projected grid. Exemplary configurations that can be

3. Establishing point correspondence

achieved by means of the utilized projection system are given in Table 3.1.

Number of spots N	Grid dimensions [cm ²]	time period t [s]
40×25 (1000)	16×10	0.1
80×50 (4000)	32×20	0.4

Table 3.1.: This table exemplarily presents possible configurations for the projection of a grid of laser spots. The quantity t defines the time period that lasts from the moment when the first laser spot hits the surface to the moment when the last laser spot hits the surface. The grid dimensions apply for a projection distance of 60 cm.

The declared values for t are only valid in the case that the measurement system shown in Fig. 1.1 is solely used for surface reconstruction. If the measurement of the optical features described in Sec. 1.2 is carried out simultaneously then the time period t increases. This is due to the fact that the laser source has to be switched on for a longer time in order to ensure that enough information bearing light can be captured by the HDR camera. Thus, the mirrors of the projection system have to stand still during the projection of every single laser spot. In this scenario, the projection of a grid of 1000 laser spots can take up to several seconds. In most cases, the simultaneous measurement of optical features is not carried out in the scope of this work. The reason is that this work is focused on the acquisition of point-based surface reconstructions. Hence, for most experiments conducted in this work, the time periods declared in Table 3.1 are valid. If the simultaneous measurement of optical features was necessary in a conducted experiment then the deviating time period is explicitly specified.

In the following, the developed time multiplexing method is described. By means of the binary code, the position of N laser spots is encoded in n projected grids. The number N of laser spots and the code length n are related by:

$$N = 2^n. \quad (3.1)$$

Thus, given a number of N laser spots that have to be encoded, the required code length n is calculated by:

$$n = \log_2(N). \quad (3.2)$$

If this equation results in a value n that is not a natural number then n has to be rounded up in order to provide a code length that is sufficient for the encoding of N laser spots. The triangulation camera operates in trigger mode and captures the reflection of each grid in one single image. Consequently, the projection of n grids results in n captured camera images. In the following, the n captured camera images are referred to as code frames. Since the head of a person will move during head tracking, a high number of consecutive zeros has to be avoided in the projected code words. This is necessary in order to ensure that the code words can be tracked in the captured camera images. For this reason, bit stuffing is used to extend the code frames by so-called full frames. When the camera captures a full frame then it is ensured that all laser spots of the grid are projected. The fusion of code frames and full frames is referred to as extended code sequence.

As shown in Fig. 3.1, an extended code sequence starts with a full frame. In this manner, the system can acquire the initial pixel location of the laser spots that are visible in the image. The pixel location of a laser spot is represented by its center. In this work, the following nomenclature is used to avoid unclarity. The term laser spot describes a laser profile and the term laser center refers to a point information. The detection of a center of a laser spot is also known as interest point detection and is described later in Sec. 3.3.

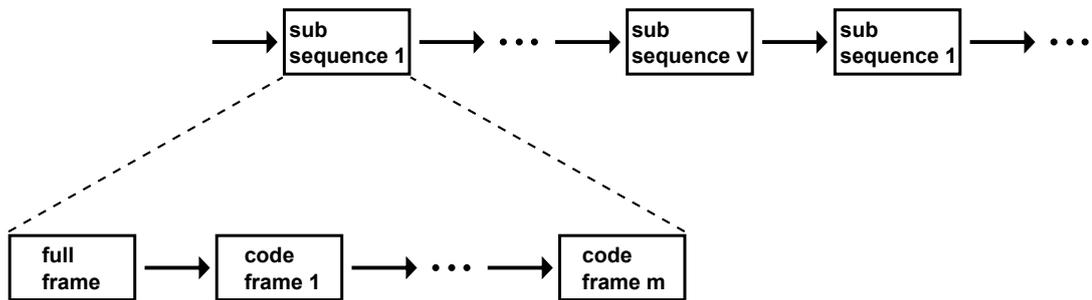


Figure 3.1.: Extended code sequence containing v sub-sequences.

An extended code sequence is arranged in v sub-sequences. A sub-sequence has a configurable length of j frames and always starts with a full frame. The remainder of a sub-sequence is given by m code frames. The processing of the code frames starts with the LSB and ends with the MSB of the projected code words. Depending on the value n , not all sub-sequences necessarily possess the same amount of code frames. If an even partitioning of the code frames is not possible then the last sub-sequence possesses a different amount of code frames. A length $j = 2$ means that a sub-sequence includes one full frame which is followed by one code frame (see Fig. 3.2) and a length $j = 3$ means

that a sub-sequence includes one full frame which is followed by two code frames. The projection of an extended code sequence is repeated permanently.

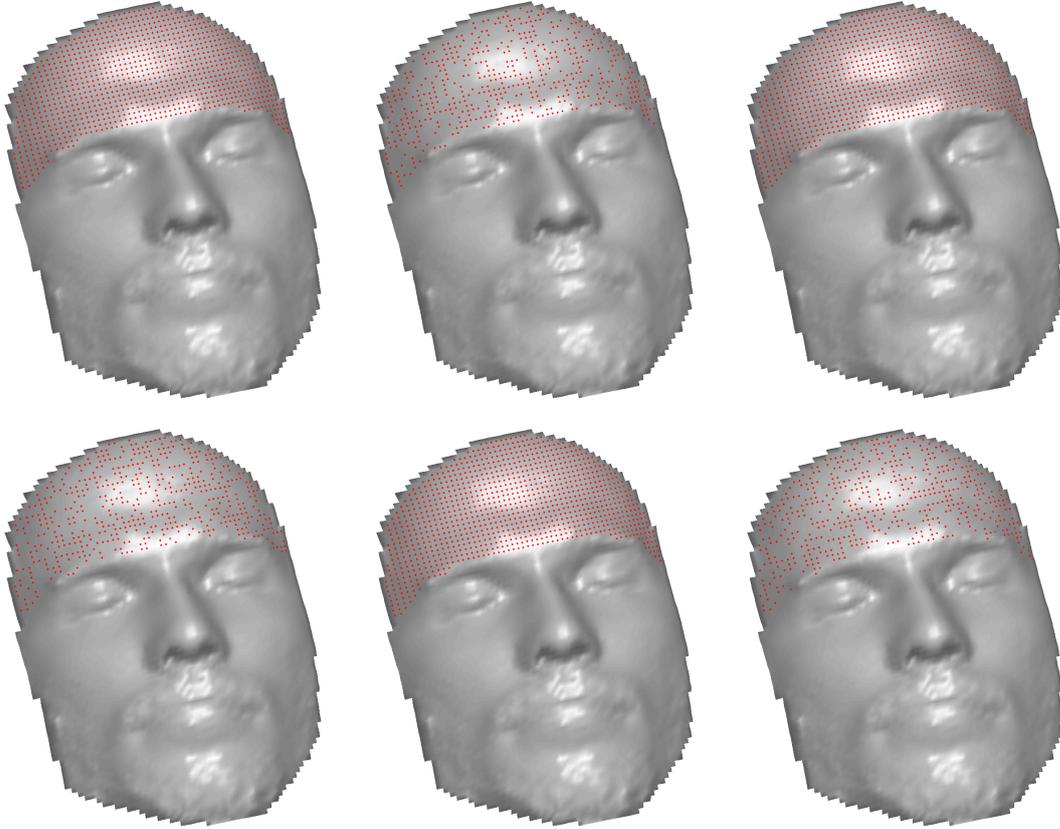


Figure 3.2.: This figure exemplarily shows 1024 laser spots which are projected onto the forehead of a person. The binary representation of the indices of the laser spots (range: 0 to 1023) defines the code words. The encoding is realized by means of $n = 10$ code frames. The location of the code words in the projected grids is computed by a random permutation. The length of each sub-sequence is defined by $j = 2$ and thus the number of sub-sequences is given by $v = 10$. The images show the projection of the first three sub-sequences of an extended code sequence. The order is given by top-left to bottom-right.

The tracking of the code words in a captured extended code sequence is carried out for each laser spot that is visible in the first full frame. The latter yields the initial pixel locations of the laser centers. The pixel locations of the laser centers in the subsequent code frame are used to build a KD tree as described in Sec. A.7. Using this KD tree, a radius based nearest-neighbor search is performed for each initial pixel location (query points) from the first full frame.

The radius r is individually calculated for each query point and is given by half of the distance between the query point and the closest adjacent query point in the first full frame. A successful tracking of the code words requires that the pixel locations of the laser centers in the subsequent code frame are located within the radius of the query points. This implies that only a slight head movement is allowed between the acquisition of two camera images.

If the radius based search returns a nearest neighbor for a query point, then the tracked code information is given by the bit value 1. Otherwise the tracked code information is given by the bit value 0. The subsequent frames within the extended code sequence are processed in the same manner. Here, a special case is given by the radius based search for the full frames. A full frame yields an update of the location of the query points and the individual radius for each query point is recomputed. Furthermore, the full frames do not contribute to the tracked code information. If a full frame does not yield an update for the location of a query point then the tracking of the code word is cancelled for the respective laser spot.

After all v sub-sequences of an extended code sequence have been processed, all tracked laser spots have been identified by means of their respective code words. The first extended code sequence is referred to as initialization phase. A surface reconstruction can be carried out for each full frame that is captured after the initialization phase. Code frames are not used for surface reconstruction since they would only yield a sparse surface reconstruction. The correspondence between the projected and captured light information is established by utilizing the tracked code words of the preceding extended code sequence. Laser spots are permanently tracked in the repeated extended code sequences.

Unidentified laser spots that are visible in the first full frame of a new extended code sequence are tracked as well. Unidentified laser spots can be caused by head movements which appear during the processing of the preceding extended code sequence. Head movements can lead to varying partial pattern occlusions and thus laser spots can disappear and reappear in the camera images. Consequently, partial pattern occlusions can cause the cancellation of the tracking of the affected spots. Laser spots which reappear after a while are treated as unidentified spots. The tracking of unidentified laser spots is always initiated in the beginning of an extended code sequence.

As described, the tracking of laser spots is carried out permanently in the repeated extended code sequences. The code words of already identified spots are also extracted in new

3. Establishing point correspondence

extended code sequences. In this manner, it is guaranteed that identified spots are re-identified. Consequently, erroneous codes can be corrected. As presented in Fig. 3.2, the location of the code words in the projected grids is computed by a random permutation. In this manner, homogeneous areas are avoided in the projected grids. After the code word of a laser spot is identified, further code verifications are performed. An identified code word is rejected if it is out of range. Moreover, if the same code word was assigned to two or more laser spots then the code word is rejected as well. This procedure ensures that the following stages of code verification are not perturbed.

In the scope of time multiplexing, the Gray code is often used to verify the detected code words (see Sec. 2.2). However, a loss-free conversion from the binary code to the Gray code is only possible in the case of an array of light stripes. In this work, the shape of the projected patterns is given by a grid of discrete laser spots. Thus, the majority of the laser spots possess eight adjacent spots. The Gray code defines a Hamming distance of value 1 between adjacent code words. This leads to the fact that not all original code words can be utilized to construct a grid which satisfies the Gray code constraint. In order to compensate for unusable code words, the code length n would have to be increased.

However, an increase of the code length n also leads to an increase of the described initialization phase. For this reason, the Gray code is not used in this work. Instead, the explicit code words of the adjacent laser spots are used to verify the code of a laser spot. This strategy involves higher computational costs than the utilization of a Hamming distance. Nonetheless, the used strategy is more accurate and does not lead to an increase of the code length n . Moreover, experiments revealed that the respective increase of computational costs is affordable. The developed software is anyway able to process a captured camera image before the next camera image arrives.

The next stage of code verification is realized by means of epipolar lines. Every projected laser spot corresponds to an epipolar line in the camera image. Due to head movements, a captured laser spot will move along its epipolar line. To verify the code of a captured laser spot, the orthogonal distance of the respective laser center is computed with respect to the corresponding epipolar line. The laser spot is rejected if the computed distance exceeds a threshold of 1 pixel. The described computation depends on the projective geometry of the calibrated laser scanning system and is explained later in Sec. 4.5.

The developed time multiplexing approach embeds full frames into the projected code information. This ensures that the laser spots in the camera images can be tracked when

it comes to slight movements of the person’s head. The use of full frames also provides a high reconstruction rate. The highest reconstruction rate can be achieved when the length j of each sub-sequence is given by the value 2. This means that only one code frame follows after a full frame. As shown in Table 3.1, a grid of 1000 laser spots can be projected in approximately $t = 0.1$ s. Hence, using a length of $j = 2$ for each sub-sequence, a surface reconstruction of 1000 points can be obtained every 0.2 s. This results in five surface reconstructions per second.

The only drawback of the proposed method is given by the circumstance that the first extended code sequence has to be processed before the computation of surface reconstructions can start. The first extended code sequence is referred to as initialization phase and depends on the amount N of projected laser spots as well as the length j of each sub-sequence. Table 3.2 shows two exemplary configurations for N using different values of j . An increase of j leads to a decrease of the length of the extended code sequences. In this manner, a shorter initialization phase can be achieved.

Nevertheless, an increase of j also means that the occurrence of full frames is decreased. Thus, the allowed velocity of the object whose surface has to be reconstructed is reduced as well. Furthermore, a decrease of the full frames also leads to a reduced reconstruction rate. Consequently, there exists a trade-off between the length of the initialization phase on the one hand and the allowed velocity of the object as well as the reconstruction rate on the other hand.

Number N of spots	Code length n	Length j of each sub-sequence	Number v of sub-sequences	Length of an extended code sequence
1000	10	2	10	20
		3	5	15
		6	2	12
4000	12	2	12	24
		3	6	18
		7	2	14

Table 3.2.: Exemplary parameter configurations for the projection of extended code sequences.

As explained, a grid of 1000 laser spots can be projected in approximately $t = 0.1$ s. By

using a length of $j = 2$ frames for each sub-sequence, the length of an extended code sequence is given by 20 frames. This results in a short initialization phase of 2 s. In the case of 4000 laser spots, the initialization phase takes longer. As presented in Table 3.1, a grid of 4000 laser spots can be projected in approximately $t = 0.4$ s. Using again a length of $j = 2$ frames for each sub-sequence, the length of an extended code sequence is given by 24 frames. This results in an initialization phase of 9.6 s.

As described in the beginning of this section, a moving object affects the coherence of the spatial points contained in an acquired surface reconstruction. In the case of the developed time multiplexing approach, the coherence of the reconstructed spatial points directly depends on the time period t . A surface reconstruction is computed by using the detected laser centers in a full frame. The time which is required for the projection of a full frame is defined by the time period t . When the object is moving during this time period then the captured full frame will contain the respective movement artifacts. Since the surface reconstruction depends on the detected laser centers, these artifacts are also transferred to the surface reconstruction. As a consequence, a short time period t has to be chosen for the grid projection in the context of head tracking.

The process of surface reconstruction is carried out by triangulation. The latter depends on the detected laser centers and the extrinsic parameters of the laser scanning system. The calibration of the extrinsic parameters of the laser scanning system and the triangulation of spatial points is described later in Sec. 5. Beforehand, algorithms for the detection of interest points are introduced.

3.3. Detection of interest points

The code tracking procedure described in Sec. 3.2 necessitates the accurate detection of the centers of the laser spots in the captured images. The center of a captured laser spot can be represented in two manners. An intuitive representation is given by the center of the elliptical contour of the spot. A second representation is given by the maximum intensity of the spot. The maximum intensity is not necessarily located at the same point as the center of the elliptical contour. This is due to the incident angle of the respective laser ray relative to the object's surface. The more the incident angle differs from 90° , the more the point of maximum intensity will differ from the center of the elliptical contour. Which of the two representations is utilized depends on the algorithm for interest point

detection. In this section, template matching, ellipse fitting and paraboloid fitting are introduced. Later in Sec. 6, all three detection algorithms are evaluated in terms of the accuracy of the triangulation.

Figure 3.3 shows a typical grayscale image that contains a grid of 72×56 (4032) laser spots that were projected on a planar surface. Figure 3.4 exemplarily shows a close up of nine laser spots that are included in the grid in Fig. 3.3. It is noticeable that the captured profile varies for each laser spot. Moreover, image noise is clearly visible. Depending on the properties of the utilized algorithm, both factors can lead to detection inaccuracies.

3.3.1. Template matching

Template matching utilizes a template mask T of size $(M \times N)$ which is slid across an image I . It computes the similarity between T and the equally sized image section at image position $I(x, y)$. Common similarity measures are given by *sum of squared difference* (SSD), *sum of absolute difference* (SAD), and *cross-correlation* (CC) [29]. The CC measure is the most commonly used similarity measure since it offers an optimal signal-to-noise ratio estimation [79]. According to [50], the CC measure can be defined by the following convolution of the template mask T and the respective image section at image position $I(x, y)$:

$$CC(x, y) = \sum_{v=0}^M \sum_{u=0}^N I(x+u, y+v)T(u, v). \quad (3.3)$$

A more advanced similarity measure is given by *normalized cross-correlation* (NCC) which is robust against linear changes in the amplitude of illumination [82]. The NCC measure can be defined as follows [50]:

$$NCC(x, y) = \frac{\sum_{v=0}^M \sum_{u=0}^N I(x+u, y+v)T(u, v)}{\sqrt{\sum_{v=0}^M \sum_{u=0}^N I(x+u, y+v)^2} \sqrt{\sum_{v=0}^M \sum_{u=0}^N T(u, v)^2}}. \quad (3.4)$$

The described robustness of the NCC measure is an advantage since the profile of each captured laser spot has slightly individual brightness characteristics (see Fig. 3.4). Fur-

3. Establishing point correspondence

thermore, measurements revealed that the brightness characteristics of the profile of a laser spot also vary moderately over time. For template matching, a template mask T showing one representative laser spot is extracted from a camera image that is captured offline. The result of Eq. 3.4 is given by a new image that contains several local maxima. The latter represent the highest matching similarity and yield the sought centers of the captured laser spots.

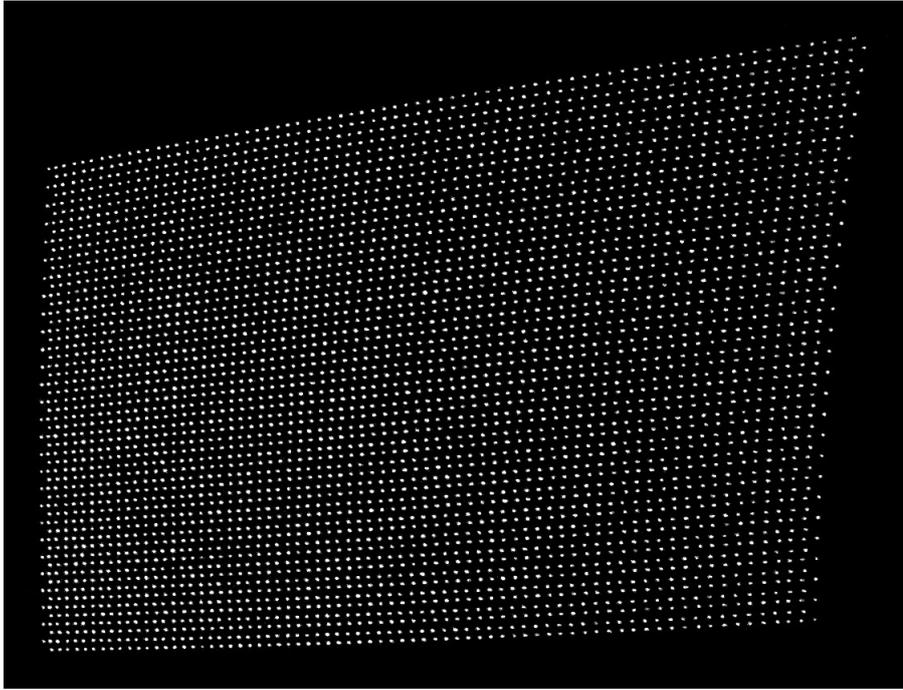


Figure 3.3.: Captured grid of 72×56 (4032) laser spots which were projected on a planar surface.

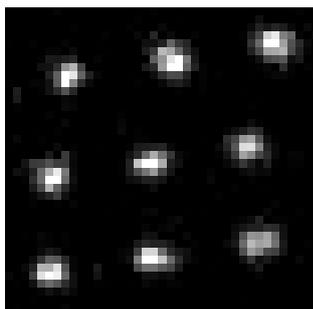


Figure 3.4.: Exemplary close up of nine captured laser spots contained in the grid in Fig. 3.3.

The matching similarity is given in the range $[0, 1]$ and a globally fixed threshold is utilized for the detection of the local maxima. The detection algorithm starts in the upper left corner of the image which results from the convolution. The algorithm compares the intensity of every pixel to the specified threshold. The search is carried out row-wise. When the intensity is equal or greater than the threshold, the algorithm assumes that the current pixel belongs to a laser spot. To find the local maximum, the algorithm uses a region of interest (ROI) that is centered at the current pixel. The ROI has the same dimensions as the utilized template mask. The maximum intensity inside the ROI represents the sought local maximum. Afterwards, the value 0 is assigned to every pixel that belongs to the ROI. Subsequently, the algorithm continues the search in the described manner by comparing the intensity of the next pixel in the current row against the threshold.

Template matching includes several disadvantages. Detection errors occur as soon as a captured laser spot differs from the template mask. The NCC measure guarantees robustness against linear changes in the amplitude of illumination. But it is not robust against scaling and perspective distortion of the captured laser spots. Hence, a new template mask has to be defined when it comes to a significant change of the pose of the camera with respect to the object whose surface has to be reconstructed. A general disadvantage of template matching is the lack of sub-pixel localization.

3.3.2. Ellipse fitting

Another method for the detection of interest points is given by ellipse fitting. Figure 3.4 shows that the contour of each captured laser spot can be described by an ellipse. Thus, the center of a captured laser spot can be determined by calculating the center of the respective ellipse. The first step of this approach is defined by the binarization of a captured grayscale image. A grayscale image which contains a grid of projected laser spots is exemplarily shown in Fig. 3.3. In this work, Otsu's method [62] is utilized for binarization. This method assumes a bi-modal image histogram that contains two classes of pixels given by back- and foreground. The histogram is analyzed to compute the grayscale threshold which optimally separates both classes of pixels. Subsequently, the computed threshold is applied to the captured grayscale image. The process of binarization yields an image in which back- and foreground are separated by means of the values 0 (black) and 255 (white), respectively. The resulting binary image contains the laser spots (filled with the color white) which are separated from the black background.

3. Establishing point correspondence

In the next step, the laser spots are detected by means of the region labeling method described in [80]. A detected spot is represented by means of all included image points. On this basis, the contour of each laser spot can be extracted. The contour of a spot is defined by the image points p_i and is used to carry out a least-squares (LS) fitting of an ellipse model. [20] covers an overview of algebraic and geometric fitting procedures. Both approaches are used to minimize the sum of the squared errors. In algebraic fitting, the error is given with respect to the axis of the ordinate. In contrast, geometric fitting, also known as orthogonal distance fitting or best fitting [2], defines the error as the orthogonal distance between a data point and the curved shape of the ellipse. For this reason, methods for geometric fitting are considered to yield more accurate results. However, geometric fitting requires an iterative solution which increases the computational cost.

In this work, a single camera image typically contains around 4000 laser spots. Hence, high computational costs for the fitting of the respective ellipses have to be avoided. Due to this reason, an algebraic fitting of ellipses is used in this work. More precisely, a linear regression based model for conic sections is applied. The procedure used in this work is similar to the technique specified in [97]. The general form of a conic section is expressed by the formula

$$0 = f(x, y) = ax^2 + bxy + cy^2 + dx + ey + f. \quad (3.5)$$

The intersection of a plane with a cone can lead to a circle, an ellipse, a parabola or a hyperbola [44]. Consequently, these types of curves are also called conic sections. The model in Eq. 3.5 is a generic formula whose parameters can be used to represent any of the described conic sections. Moreover, it provides a linear estimation of an ellipse with arbitrary translation and rotation. In this work, the model for conic sections is used in the image space and hence it utilizes image points of type $p = [x \ y]^T$. As described, the extracted contour of a laser spot is given by the image points p_i . To fit an ellipse to the contour p_i , the model in Eq. 3.5 is utilized to define the following system of linear equations:

$$\mathbf{0}_{n \times 1} = \underbrace{\begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^2 & x_n y_n & y_n^2 & x_n & y_n & 1 \end{bmatrix}}_A \underbrace{[a \ b \ c \ d \ e \ f]^T}_k. \quad (3.6)$$

In order to avoid the trivial solution $\mathbf{k} = \mathbf{0}$, the estimation of \mathbf{k} is formulated by a constrained minimization problem:

$$\begin{aligned} \min_{\mathbf{k}} \quad & \| \mathbf{A}\mathbf{k} \|_2^2, \\ \text{subject to} \quad & \| \mathbf{k} \|_2^2 = 1. \end{aligned} \tag{3.7}$$

This minimization problem can be solved by means of the *method of Lagrange multipliers*. After some steps the following eigenvalue problem can be constructed [27]:

$$\mathbf{A}^T \mathbf{A} \mathbf{k} = \lambda \mathbf{k}. \tag{3.8}$$

The symbol λ describes a *Lagrange multiplier*. A solution \mathbf{k} that differs from the null vector is an eigenvector of the matrix $\mathbf{A}^T \mathbf{A}$. Moreover, the related λ is an eigenvalue of this matrix. The sought solution for \mathbf{k} is given by the eigenvector that corresponds to the smallest eigenvalue. This solution can be comfortably computed by applying the *singular value decomposition* (SVD) to the matrix \mathbf{A} (see Sec. A.4):

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T = \mathbf{U} \text{diag}(\sigma_1, \dots, \sigma_6) \mathbf{V}^T. \tag{3.9}$$

The columns of \mathbf{V} are the eigenvectors of $\mathbf{A}^T \mathbf{A}$ and are referred to as the set v_j with $j \in [1, 6]$. The singular values of \mathbf{A} are contained on the diagonal of the matrix $\mathbf{\Sigma}$ and are referred to as the set σ_j with $j \in [1, 6]$. The number of extracted singular values and eigenvectors is equivalent to the number of columns of the matrix \mathbf{A} . The sought solution for \mathbf{k} is given by the eigenvector that corresponds to the smallest singular value [27]. Since the singular values are given in descending order, the sought solution for \mathbf{k} is defined by the eigenvector v_6 .

The estimated vector \mathbf{k} includes the parameters of the conic section defined in Eq. 3.5. Since the conic section model was fitted to an elliptical contour, the generic parameters can be converted to the parameters of an ellipse. As described in the beginning of this section, the sought parameters are given by the center of an ellipse. The following calculation of the center is based on the descriptions in [65]. Using the parameters a to e of the estimated vector \mathbf{k} , the center c of an ellipse is computed as follows:

$$\mathbf{c} = \begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{Q}^{-1}\mathbf{m} = \begin{bmatrix} 2a & b \\ b & 2c \end{bmatrix}^{-1} \begin{bmatrix} -d \\ -e \end{bmatrix}. \quad (3.10)$$

Ellipse fitting represents a robust technique to detect the center of a laser spot in the presence of image noise. Compared to template matching, the fitting of an ellipse does not depend on a reference signal. The fitting is carried out individually for each laser spot. Consequently, a varying laser profile in the camera image does not affect the detection accuracy. Furthermore, the center computation provides sub-pixel localization.

3.3.3. Paraboloid fitting

As proposed in [70], the detection of interest points can also be realized by paraboloid fitting. This method assumes that the center of a laser spot is defined by its maximum intensity. Fitting a paraboloid to a captured laser spot can compensate for image noise and the imperfection of the laser source. In this context, the maximum intensity of a captured laser spot is used as an initial guess. After paraboloid fitting, the sought center of a laser spot is given by the maximum of the paraboloid.

Fitting a paraboloid to a captured laser spot is carried out as follows. At first, the described initial guess has to be detected. This can be achieved by means of template matching described in Sec. 3.3.1. This method computes an image point s which represents the maximum similarity between the template mask and the captured laser spot. The maximum intensity of the captured laser spot is located in the local neighborhood of s . The maximum intensity of the captured laser spot is detected inside a ROI which is centered at s . This ROI has the same dimensions as the template mask that was used before. The fitting of a paraboloid is carried out by using the following bi-squared function:

$$z = f(x, y) = ax^2 + bxy + cy^2 + dx + ey + f. \quad (3.11)$$

This function uses the coordinates of an image point $\mathbf{p} = [x \ y]^T$ and the associated intensity z . Accordingly, the computed initial guess and its eight neighbors are utilized to construct a system of linear equations:

$$\underbrace{\begin{bmatrix} z_1 \\ \vdots \\ z_9 \end{bmatrix}}_w = \underbrace{\begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_9^2 & x_9 y_9 & y_9^2 & x_9 & y_9 & 1 \end{bmatrix}}_A \underbrace{[a \ b \ c \ d \ e \ f]^T}_k. \quad (3.12)$$

In the respective image section, the points p_1 to p_9 are ordered from upper left to lower right. Hence, the initial guess is defined by p_5 . By using the initial guess as the origin of a translated coordinate system, the matrix A can be converted into a constant matrix:

$$A = \begin{bmatrix} 1 & 1 & 1 & -1 & -1 & 1 \\ 0 & 1 & 0 & 0 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}. \quad (3.13)$$

The system of linear equations defined by Eqs. 3.12 and 3.13 can be solved by means of the *Moore-Penrose pseudoinverse* described in Sec. A.3. Consequently, the vector k is computed by:

$$k = A^+ w. \quad (3.14)$$

The explained paraboloid fitting uses the pixel values of a 3×3 sub-matrix in the camera image. In this work, paraboloid fitting is also carried out by means of a 5×5 sub-matrix. The respective adjustment of matrix A in Eq. 3.13 is straightforward and is not further described. As mentioned in the beginning of this section, the sought quantity is given by the maximum of the fitted paraboloid. The maximum of the fitted paraboloid in the translated coordinate system is given by:

3. Establishing point correspondence

$$\begin{aligned} u &= \frac{2cd - be}{b^2 - 4ac} , \\ v &= \frac{2ae - bd}{b^2 - 4ac} . \end{aligned} \tag{3.15}$$

Using the following translation, the maximum of the fitted paraboloid is converted to the original image coordinate system:

$$\mathbf{c} = \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x_5 + u \\ y_5 + v \end{bmatrix} . \tag{3.16}$$

This translation represents an update of the initial guess $\mathbf{p}_5 = [x_5 \ y_5]^T$. The quantity \mathbf{c} defines the detected center of the laser spot.

The robustness of paraboloid fitting against image noise clearly depends on the dimensions of the sub-matrix. A larger sub-matrix can lead to more accurate results. But the use of a larger sub-matrix also increases the computational costs. Compared to template matching, paraboloid fitting does not depend on a reference signal. The fitting is carried out individually for each laser spot. Consequently, a varying laser profile in the camera image does not affect the detection accuracy. Furthermore, the center computation provides sub-pixel localization.

4. Projective geometry

This section describes the models of projective geometry which are used for the calibration of the laser scanning system and for surface reconstruction in Sec. 5. The utilized models of projective geometry cover a camera model, the homography matrix, pose estimation, triangulation and the epipolar geometry.

4.1. Camera model

4.1.1. Model of central projection

The projection of a spatial point $\mathbf{p}^c = [x_c \ y_c \ z_c]^T$ on an image point $\mathbf{p}^s = [x_s \ y_s]^T$ is defined by the intercept theorem [27]:

$$\mathbf{p}^s = \frac{f}{z_c} \begin{bmatrix} x_c \\ y_c \end{bmatrix}. \quad (4.1)$$

The spatial point \mathbf{p}^c is given in the camera coordinate system $(CS)_c$ and the image point \mathbf{p}^s is given in the sensor coordinate system $(CS)_s$. The projective mapping defined in Eq. 4.1 is also shown in Fig. 4.1. The z-axis of $(CS)_c$ is called optical axis and represents the camera's direction of view. The optical axis orthogonally intersects the sensor plane. The point of intersection defines the origin of the sensor coordinate system $(CS)_s$. The distance between the origin of $(CS)_c$ and the origin of $(CS)_s$ is expressed by f and presents an intrinsic parameter of the camera.

In the context of the intercept theorem, the origin of $(CS)_s$ is located in front of the camera. Contrary to this, the origin of $(CS)_s$ in a real physical camera is located on the

negative z -axis of $(CS)_c$. For this reason, the sensor plane in a real physical camera is rotated by 180° around the x -axis of $(CS)_s$. Subsequently, the sensor plane is rotated by 180° around the the y -axis of $(CS)_s$. Hence, a projective ray will intersect with the camera's center before reaching the sensor plane. The camera's center is defined by the origin of $(CS)_c$ and a projective ray connects the spatial point p^c with the image point p^s . Changing the pose of $(CS)_s$ with respect to $(CS)_c$ in the described manner, leads to the fact that the projective mapping in Eq. 4.1 is also valid for real physical cameras. By using homogeneous points \tilde{p}^c and \tilde{p}^s , the projective mapping in Eq. 4.1 can be converted into matrix form:

$$\tilde{p}^s = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tilde{p}^c. \quad (4.2)$$

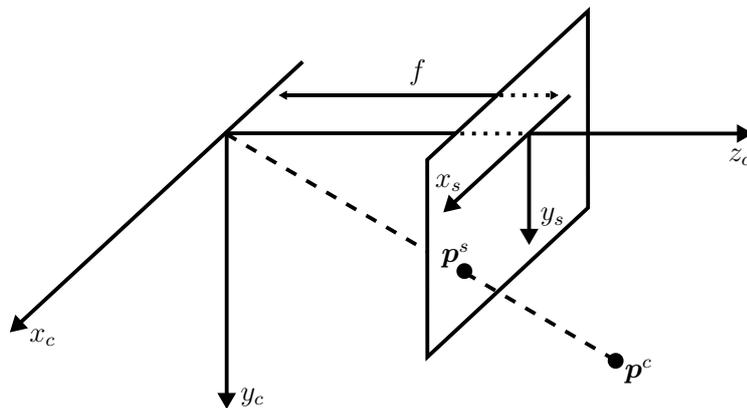


Figure 4.1.: Central projection.

In an image processing system, the origin of the image coordinate system $(CS)_b$ is located in the top-left corner of the camera image. For this reason, an image point p^s has to be converted into an image point $p^b = [x_b \ y_b]^T$ with respect to the image coordinate system $(CS)_b$. This conversion is achieved by means of a translation which uses the principal point $[x_0 \ y_0]^T$:

$$\tilde{p}^b = \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tilde{p}^s. \quad (4.3)$$

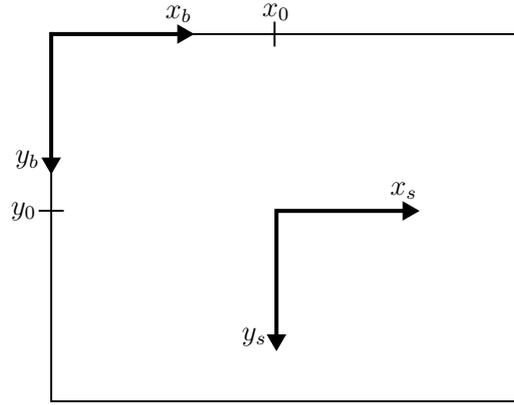


Figure 4.2.: Sensor and image coordinate system.

The principal point represents the origin of the sensor coordinate system $(CS)_s$ with respect to the image coordinate system $(CS)_b$ (see Fig. 4.2). Equations 4.2 and 4.3 are merged as follows:

$$\tilde{\mathbf{p}}^b = \begin{bmatrix} f & 0 & x_0 & 0 \\ 0 & f & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tilde{\mathbf{p}}^c = \underbrace{\begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 1} \end{bmatrix} \tilde{\mathbf{p}}^c. \quad (4.4)$$

\mathbf{K} defines the calibration matrix and contains all intrinsic parameters of the camera. The symbol $\mathbf{I}_{3 \times 3}$ defines the (3×3) identity matrix. So far, a spatial point was given by \mathbf{p}^c with respect to the camera coordinate system. However, spatial points are often given with respect to an object coordinate system $(CS)_g$. By means of the pose ${}^c\mathbf{T}_g$, a point $\mathbf{p}^g = [x_g \ y_g \ z_g]^T$ can be transformed into a point \mathbf{p}^c (see Sec. A.2). The transformation matrix ${}^c\mathbf{T}_g$ contains the (3×1) translation vector ${}^c\mathbf{t}_g$ and the (3×3) rotation matrix ${}^c\mathbf{R}_g$. These quantities define the extrinsic parameters of the camera. Using the extrinsic parameters of the camera and Eq. 4.4, the projection of \mathbf{p}^g on \mathbf{p}^b is given by:

$$\tilde{\mathbf{p}}^b = \underbrace{\mathbf{K} \begin{bmatrix} {}^c\mathbf{R}_g & {}^c\mathbf{t}_g \end{bmatrix}}_{\mathbf{P}} \tilde{\mathbf{p}}^g. \quad (4.5)$$

\mathbf{P} is called projection matrix. Besides f , x_0 and y_0 , more intrinsic parameters have to be incorporated in order to model the projection behavior of a *charge-coupled device* (CCD)

camera. First of all, a scaling m_x and a scaling m_y have to be considered for the direction x and y , respectively. Both scaling factors specify the ratio between the number of pixels and the edge lengths of the sensor. Hence, it is also possible to model non quadratic pixels. Usually, a CCD sensor includes quadratic pixels. In this case, m_x and m_y are equal. Moreover, a factor s_k for shear mapping has to be incorporated. This parameter models the angle between the x -axis and y -axis of the sensor. Commonly, the sensor's axes are orthogonal and hence the shear mapping is given by $s_k = 0$. The calibration matrix incorporates the additional intrinsic camera parameters as follows:

$$\mathbf{K} = \begin{bmatrix} fm_x & fm_x s_k & x_0 m_x \\ 0 & fm_y & y_0 m_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & \alpha_x s_k & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (4.6)$$

The intrinsic camera parameters f , x_0 , y_0 , m_x and m_y are combined by α_x , α_y , u_0 and v_0 .

4.1.2. Lens distortion

The camera's lens system causes a distortion of image points. For this reason, the model of central projection described in Sec. 4.1.1 is extended by a model of lens distortion. On this basis, a rectification of image points can be defined later in Sec. 4.1.3. The model of lens distortion considers the following types of distortion [89]:

- **Radial distortion:** The imperfection of lenses leads to a radial distortion δ_{rad} of the image points. Radial distortion is defined relative to the center of distortion which is given by the principal point presented in Fig. 4.2. The effect of radial distortion is given by either an outward or inward distortion. The higher the distance between an image point and the center of distortion, the higher the radial distortion. The effect of radial distortion is exemplarily shown for the undistorted lines of the grid in Fig. 4.3 a. The result is given by either a pincushion distortion (outward distortion) or a barrel distortion (inward distortion) in Fig. 4.3 b and c, respectively.
- **Decentering distortion:** In the ideal case, the optical center of the lenses are collinearly located on the optical axis. Deviations lead to a decentering distortion δ_{dec} which consists of a radial distortion and a tangential distortion. As exemplarily presented in Fig. 4.4, the effect of tangential distortion is given by a sideward dis-

tortion. All axes shown in Fig. 4.4 also represent undistorted lines which become bent due to tangential distortion.

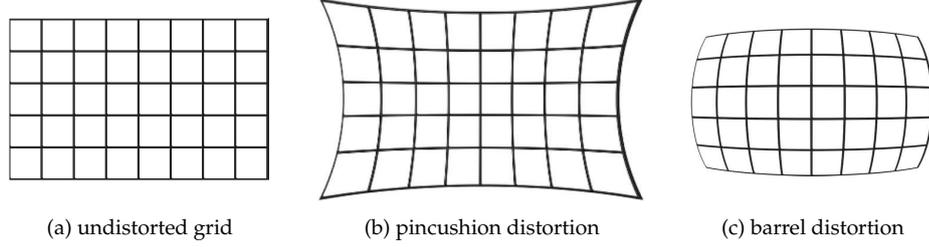


Figure 4.3.: Types of radial distortion.

The basis for the model of lens distortion is given by a normalized image point $\mathbf{p}^n = [x_n \ y_n]^T$:

$$\mathbf{p}^n = \frac{1}{z_c} \begin{bmatrix} x_c \\ y_c \end{bmatrix}. \quad (4.7)$$

Eq. 4.7 is equivalent to Eq. 4.1 for the case $f = 1$. The squared L2-norm of \mathbf{p}^n is defined by:

$$r^2 = x_n^2 + y_n^2. \quad (4.8)$$

Using Eq. 4.8, the radial distortion is computed as follows [28]:

$$\delta_{rad} = (k_{r1} r^2 + k_{r2} r^4 + k_{r3} r^6) \mathbf{p}^n. \quad (4.9)$$

Furthermore, the decentering distortion is expressed by:

$$\delta_{dec} = \begin{bmatrix} 2k_{t1}x_ny_n + k_{t2}(r^2 + 2x_n^2) \\ k_{t1}(r^2 + 2y_n^2) + 2k_{t2}x_ny_n \end{bmatrix}. \quad (4.10)$$

The parameters of radial distortion k_{r1} , k_{r2} and k_{r3} as well as the parameters of tangential distortion k_{t1} and k_{t2} represent further intrinsic parameters of the camera. Applying

Eqs. 4.9 and 4.10 to a normalized image point \mathbf{p}^n leads to a distorted image point $\mathbf{p}^d = [x_d \ y_d]^T$:

$$\mathbf{p}^d = \mathbf{p}^n + \delta_{rad} + \delta_{dec}. \quad (4.11)$$

As mentioned in Sec. 4.1.1, an image point in the image processing system is given by \mathbf{p}^b . For this reason, the calibration matrix \mathbf{K} in Eq. 4.6 is utilized to transform a distorted image point \mathbf{p}^d into an image point \mathbf{p}^b :

$$\tilde{\mathbf{p}}^b = \mathbf{K}\tilde{\mathbf{p}}^d. \quad (4.12)$$

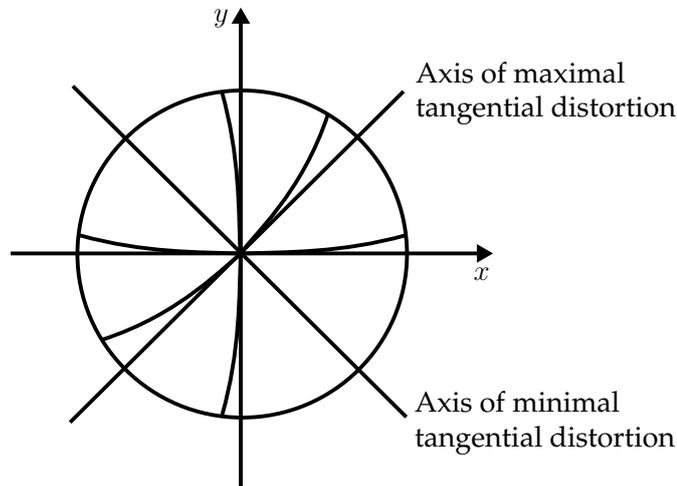


Figure 4.4.: Tangential distortion of lines.

4.1.3. Rectification of image points

A rectification of image points is achieved by compensating for the effects of lens distortion. A rectification is necessary in order to guarantee that image points are given with respect to the model of central projection described in Sec. 4.1.1. This is a prerequisite for the models of projective geometry that are utilized in the subsequent sections. By means of rectification, a given image point \mathbf{p}^b is transformed into a normalized image point \mathbf{p}^n . In the first step, Eq. 4.12 is expressed in Euclidean form:

$$\mathbf{p}^b = \begin{bmatrix} \alpha_x(x_d + s_k y_d) + u_0 \\ \alpha_y y_d + v_0 \end{bmatrix}. \quad (4.13)$$

Hence, the inverse computation of the components of the distorted image point \mathbf{p}^d is defined by:

$$\begin{aligned} y_d &= \frac{y_b - v_0}{\alpha_y}, \\ x_d &= \frac{x_b - u_0}{\alpha_x} - s_k y_d. \end{aligned} \quad (4.14)$$

Using the distorted image point \mathbf{p}^d , the inversion of Eq. 4.11 leads to the normalized image point \mathbf{p}^n . Since Eqs. 4.9 and 4.10 contain \mathbf{p}^n as well, a direct inversion is not possible. Instead, the *Gauss-Newton method* [10] can be used to find the root of the function

$$\mathbf{f}(\mathbf{p}^n) = \mathbf{p}^n + \delta_{rad} + \delta_{dec} - \mathbf{p}^d. \quad (4.15)$$

The *Gauss-Newton method* is an iterative procedure and requires a starting point for the sought solution. The starting point is given by the distorted image point \mathbf{p}^d . The described rectification of image points uses the complete set of intrinsic parameters of the camera. The latter can be estimated by means of a camera calibration. In this work, the intrinsic parameters were calibrated by means of the widely-used *Camera Calibration Toolbox for Matlab* by Jean-Yves Bouguet which is partially inspired by the calibration method of Zhang [98]. As explained, the subsequent models of projective geometry necessitate that image points are rectified. For this reason, the following sections solely refer to normalized image points.

4.2. Homography matrix

The homography matrix \mathbf{H} defines the mapping between corresponding image points. As exemplarily shown in Fig. 4.5, a point \mathbf{p}^n in the image I corresponds with a point \mathbf{p}^{n*} in the image I^* if both points represent the 2D projection of the same spatial point \mathbf{p}^g . Us-

ing the (3×3) homography matrix \mathbf{H} , an image point $\tilde{\mathbf{p}}^n$ is mapped to the corresponding image point $\tilde{\mathbf{p}}^{n*}$ as follows [27]:

$$\tilde{\mathbf{p}}^{n*} = \mathbf{H}\tilde{\mathbf{p}}^n = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \tilde{\mathbf{p}}^n. \quad (4.16)$$

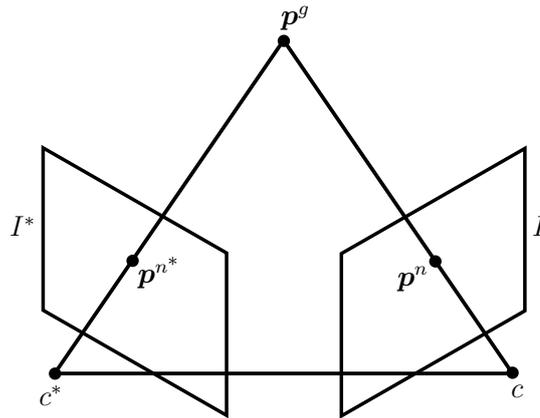


Figure 4.5.: This figure shows the geometric relationships for one pair of corresponding image points p^n and p^{n*} which are given in the image I and I^* , respectively. The spatial point p^g is given in the object coordinate system $(CS)_g$ and is projected on p^n and p^{n*} . The connection between the camera center c and the camera center c^* represents the baseline.

\mathbf{H} is a nonsingular matrix and hence the inverse mapping is given by the inversion of \mathbf{H} :

$$\tilde{\mathbf{p}}^n = \mathbf{H}^{-1} \tilde{\mathbf{p}}^{n*}. \quad (4.17)$$

Using the *direct linear transformation* (DLT), the homography matrix \mathbf{H} can be estimated. The basis for this estimation is given by Eq. 4.16. Since \mathbf{H} is a homogeneous matrix, Eq. 4.16 can be extended by a scale factor $s \neq 0$ [27]:

$$\tilde{\mathbf{p}}^{n*} = s\mathbf{H}\tilde{\mathbf{p}}^n. \quad (4.18)$$

The scale factor s emphasizes that the vector on the left side and the vector on the right side of Eq. 4.18 have the same direction but not necessarily the same length. Using Eq.

4.16, this relationship can be expressed by the cross product:

$$\mathbf{0}_{3 \times 1} = \tilde{\mathbf{p}}^{n*} \times \mathbf{H} \tilde{\mathbf{p}}^n. \quad (4.19)$$

By reordering Eq. 4.19, the following system of linear equations can be defined:

$$\mathbf{0}_{2 \times 1} = \underbrace{\begin{bmatrix} \mathbf{0}_{1 \times 3} & -(\mathbf{p}^n)^T & -1 & y_{n*}(\mathbf{p}^n)^T & y_{n*} \\ (\mathbf{p}^n)^T & 1 & \mathbf{0}_{1 \times 3} & -x_{n*}(\mathbf{p}^n)^T & -x_{n*} \end{bmatrix}}_{\mathbf{A}} \underbrace{[h_1 \ \dots \ h_9]^T}_{\mathbf{h}}. \quad (4.20)$$

In this system, one linearly dependent equation has been removed. Furthermore, the value 1 has been assigned to the third component of $\tilde{\mathbf{p}}^{n*}$ and $\tilde{\mathbf{p}}^n$. Hence, Eq. 4.20 only includes the Euclidean coordinates x_{n*} and y_{n*} as well as the Euclidean point \mathbf{p}^n . The components of the vector \mathbf{h} in Eq. 4.20 correspond to the components of \mathbf{H} in Eq. 4.16. The order of the components is given by top-left to bottom-right. Equation 4.20 uses one pair of corresponding image points. By means of n pairs of corresponding image points, the system of linear equations becomes

$$\mathbf{0} = \underbrace{\begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_n \end{bmatrix}}_{\mathbf{A}_{res}} \mathbf{h}. \quad (4.21)$$

For the sake of simplicity, \mathbf{A}_{res} will be referred to as \mathbf{A} . According to [27], $n \geq 4$ pairs of corresponding image points have to be available to estimate \mathbf{h} . In order to avoid the trivial solution $\mathbf{h} = \mathbf{0}$, the estimation of \mathbf{h} is formulated by a constrained minimization problem:

$$\begin{aligned} \min_{\mathbf{h}} \quad & \|\mathbf{A}\mathbf{h}\|_2^2, \\ \text{subject to} \quad & \|\mathbf{h}\|_2^2 = 1. \end{aligned} \quad (4.22)$$

As explained in Sec. 3.3.2, this minimization problem can be solved by applying the *method of Lagrange multipliers*. After some steps the following eigenvalue problem can be

defined [27]:

$$\mathbf{A}^T \mathbf{A} \mathbf{h} = \lambda \mathbf{h}. \quad (4.23)$$

A solution \mathbf{h} that differs from the null vector is an eigenvector of the matrix $\mathbf{A}^T \mathbf{A}$. The sought solution for \mathbf{h} is given by the eigenvector that corresponds to the smallest eigenvalue. This solution can be comfortably computed by applying the SVD to the matrix \mathbf{A} (see Sec. A.4):

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T. \quad (4.24)$$

The sought solution for \mathbf{h} is given by the eigenvector that is located in the last column of \mathbf{V} [27].

4.3. Pose estimation

In this work, the pose cT_g of a planar calibration body has to be estimated with respect to a camera. The concept of the pose of a rigid body is explained in Secs. A.1 and A.2. To construct the calibration body, the calibration pattern presented in Fig. 4.6 is printed on a planar board. In order to estimate the pose cT_g , the coordinate system $(CS)_g$ of the calibration body has to be defined. One freely selectable corner of the squares of the calibration pattern is used as the origin of $(CS)_g$. The x-axis and the y-axis of $(CS)_g$ are aligned along the edges of the squares. All squares of the calibration pattern have the same size and the edge length is known. Hence, the coordinates of the corners of the squares are known in the x-y plane of $(CS)_g$. A 2D point in the x-y plane of $(CS)_g$ is referred to as $\mathbf{p}^g = [x_q \ y_q]^T$. Furthermore, the z-axis of $(CS)_g$ is a normal vector on the calibration pattern which points towards the viewer. The relationship between a 2D point \mathbf{p}^g and its 3D representation \mathbf{p}^g relative to $(CS)_g$ is given by:

$$\mathbf{p}^g = [x_g \ y_g \ z_g]^T = [x_q \ y_q \ 0]^T. \quad (4.25)$$

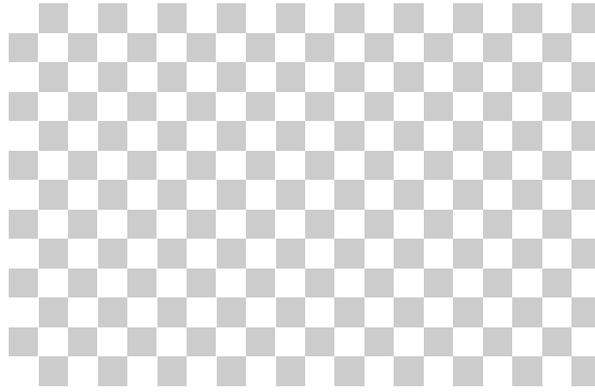


Figure 4.6.: Calibration pattern.

The pose cT_g of the calibration body is estimated by computing an initial solution which is improved by a subsequent refinement procedure.

4.3.1. Initial solution

The computation of an initial solution for cT_g is based on a captured camera image which contains the projection of the constructed calibration body. Based on the *Harris corner detector* [25], the corners of the inner squares of the calibration pattern are detected within sub-pixel domain. The resulting set of detected corners is referred to as \mathbf{p}_j^n with $j \in [1, J]$ where J defines the amount of detected corners. As mentioned, the coordinates of the corners are also known with respect to $(CS)_g$. In the following, \mathbf{p}_j^q describes the set of corners of the inner squares in the x-y plane of $(CS)_g$. Two points \mathbf{p}^n and \mathbf{p}^q are referred to as corresponding points if they incorporate the same index j . By using the corresponding sets \mathbf{p}_j^n and \mathbf{p}_j^q , the homography matrix \mathbf{H} can be estimated (see Sec. 4.2). Using \mathbf{H} , a point \mathbf{p}^q is mapped to the corresponding point \mathbf{p}^n as follows:

$$\tilde{\mathbf{p}}^n = \mathbf{H}\tilde{\mathbf{p}}^q. \quad (4.26)$$

An initial solution for the pose cT_g is computed by means of decomposing the estimated homography matrix \mathbf{H} into a translation vector and a rotation matrix [46]. First of all, \mathbf{H} has to be normalized. For this purpose, the SVD is applied to the matrix \mathbf{H} (see Sec. A.4):

$$\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{U} \text{diag}(\sigma_1, \sigma_2, \sigma_3) \mathbf{V}^T. \quad (4.27)$$

Subsequently, each component of \mathbf{H} is divided by the second singular value σ_2 :

$$\mathbf{H}_1 = \frac{1}{\sigma_2} \mathbf{H}. \quad (4.28)$$

The final normalization step is given by a sign correction of \mathbf{H}_1 . For this purpose, the sign of the dot product in Eq. 4.29 has to be evaluated for each pair of corresponding points. The sign function is described in Eq. 4.30. The sign of \mathbf{H}_1 is only corrected if the sign of each dot product in Eq. 4.29 is negative:

$$\mathbf{H}_2 = \begin{cases} -\mathbf{H}_1 & \text{if } \text{sgn}((\mathbf{p}^n)^T \mathbf{H}_1 \mathbf{p}^q) = -1 \forall j \\ \mathbf{H}_1 & \text{otherwise.} \end{cases} \quad (4.29)$$

$$\text{sgn}(x) = \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases} \quad (4.30)$$

In the following, the normalized homography matrix \mathbf{H}_2 is decomposed into a translation vector and a rotation matrix. At first, the SVD is applied to the matrix \mathbf{H}_2 :

$$\mathbf{H}_2 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{U} \text{diag}(\sigma_1, \sigma_2, \sigma_3) [\mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3]^T. \quad (4.31)$$

The columns of the matrix \mathbf{V} represent the orthonormal (3×1) eigenvectors $\mathbf{v}_1, \mathbf{v}_2$ and \mathbf{v}_3 . The components of the SVD in Eq. 4.31 are used to calculate the (3×1) vectors \mathbf{m}_1 and \mathbf{m}_2 as follows:

$$\begin{aligned}
\mathbf{m}_1 &= \frac{\sqrt{1 - \sigma_3^2} \mathbf{v}_1 + \sqrt{\sigma_1^2 - 1} \mathbf{v}_3}{\sqrt{\sigma_1^2 - \sigma_3^2}}, \\
\mathbf{m}_2 &= \frac{\sqrt{1 - \sigma_3^2} \mathbf{v}_1 - \sqrt{\sigma_1^2 - 1} \mathbf{v}_3}{\sqrt{\sigma_1^2 - \sigma_3^2}}.
\end{aligned} \tag{4.32}$$

Since \mathbf{v}_1 and \mathbf{v}_3 are orthonormal, the resulting vector in the numerator of \mathbf{m}_1 is normalized by using the term $\sqrt{\sigma_1^2 - \sigma_3^2}$ in the denominator. Consequently, \mathbf{m}_1 is a unit vector. Furthermore, \mathbf{m}_1 is orthonormal to \mathbf{v}_2 . The same relationships apply for the vector \mathbf{m}_2 . By means of the cross product, the vectors \mathbf{m}_1 and \mathbf{m}_2 are used to compute the orthogonal matrices \mathbf{B}_1 and \mathbf{B}_2 , respectively:

$$\begin{aligned}
\mathbf{B}_1 &= \begin{bmatrix} \mathbf{v}_2 & \mathbf{m}_1 & \mathbf{v}_2 \times \mathbf{m}_1 \end{bmatrix}, \\
\mathbf{B}_2 &= \begin{bmatrix} \mathbf{v}_2 & \mathbf{m}_2 & \mathbf{v}_2 \times \mathbf{m}_2 \end{bmatrix}.
\end{aligned} \tag{4.33}$$

The normalized homography matrix \mathbf{H}_2 as well as the vectors \mathbf{m}_1 and \mathbf{m}_2 are utilized to compute a second set of orthogonal matrices \mathbf{W}_1 and \mathbf{W}_2 :

$$\begin{aligned}
\mathbf{W}_1 &= \begin{bmatrix} \mathbf{H}_2 \mathbf{v}_2 & \mathbf{H}_2 \mathbf{m}_1 & (\mathbf{H}_2 \mathbf{v}_2) \times (\mathbf{H}_2 \mathbf{m}_1) \end{bmatrix}, \\
\mathbf{W}_2 &= \begin{bmatrix} \mathbf{H}_2 \mathbf{v}_2 & \mathbf{H}_2 \mathbf{m}_2 & (\mathbf{H}_2 \mathbf{v}_2) \times (\mathbf{H}_2 \mathbf{m}_2) \end{bmatrix}.
\end{aligned} \tag{4.34}$$

As explained, the decomposition of the estimated homography matrix yields an initial solution for the pose ${}^c\mathbf{T}_g$ of the calibration body with respect to the camera. The quantities that were computed so far, are utilized in Eqs. 4.35 to 4.38 in order to define four different solutions for the decomposition of the estimated homography matrix. Each solution contains a translation vector \mathbf{t} , a rotation matrix \mathbf{R} and a normal vector \mathbf{n} . The latter is located on the calibration body. The physically possible solutions are selected by imposing the following positive depth constraint. If the z-component of the translation vector \mathbf{t} is positive then the calibration body is located in front of the camera. Two of the four solutions satisfy this constraint and the remaining two solutions are rejected. Both physically possible solutions represent an initial solution for the pose ${}^c\mathbf{T}_g$. Both initial solutions lead to the same result in the following refinement procedure.

$$\begin{aligned}
 \mathbf{n}_1 &= \mathbf{v}_2 \times \mathbf{m}_1 \\
 \mathbf{R}_1 &= \mathbf{W}_1 \mathbf{B}_1^T \\
 \mathbf{t}_1 &= (\mathbf{H}_{N_2} - \mathbf{R}_1)\mathbf{n}_1
 \end{aligned} \tag{4.35}$$

$$\begin{aligned}
 \mathbf{n}_2 &= \mathbf{v}_2 \times \mathbf{m}_2 \\
 \mathbf{R}_2 &= \mathbf{W}_2 \mathbf{B}_2^T \\
 \mathbf{t}_2 &= (\mathbf{H}_{N_2} - \mathbf{R}_2)\mathbf{n}_2
 \end{aligned} \tag{4.36}$$

$$\begin{aligned}
 \mathbf{n}_3 &= -\mathbf{n}_1 \\
 \mathbf{R}_3 &= \mathbf{R}_1 \\
 \mathbf{t}_3 &= -\mathbf{t}_1
 \end{aligned} \tag{4.37}$$

$$\begin{aligned}
 \mathbf{n}_4 &= -\mathbf{n}_2 \\
 \mathbf{R}_4 &= \mathbf{R}_2 \\
 \mathbf{t}_4 &= -\mathbf{t}_2
 \end{aligned} \tag{4.38}$$

4.3.2. Refinement

In order to formulate the refinement procedure, the projection matrix \mathbf{P} in Eq. 4.5 is defined for a normalized image point $\tilde{\mathbf{p}}^n$:

$$\tilde{\mathbf{p}}^n = \underbrace{\begin{bmatrix} {}^c\mathbf{R}_g & {}^c\mathbf{t}_g \end{bmatrix}}_{\mathbf{P}} \tilde{\mathbf{p}}^g = \underbrace{\begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix}}_{\mathbf{P}} \tilde{\mathbf{p}}^g. \tag{4.39}$$

The vectors \mathbf{P}_1 , \mathbf{P}_2 and \mathbf{P}_3 define the rows of the projection matrix \mathbf{P} . The latter contains the translation vector ${}^c\mathbf{t}_g$ and the rotation matrix ${}^c\mathbf{R}_g$ of the sought pose ${}^c\mathbf{T}_g$. Using the projective mapping in Eq. 4.39, the non-linear least-squares problem in Eq. 4.40 can be formulated. In this manner, the parameters contained in \mathbf{P} can be refined. The initial solution is given by the translation vector ${}^c\mathbf{t}_g$ and the rotation matrix ${}^c\mathbf{R}_g$ which are obtained from the decomposition of the homography matrix described in Sec. 4.3.1.

Furthermore, the corresponding sets \mathbf{p}_j^n and \mathbf{p}_j^g of 2D corners from Sec. 4.3.1 are utilized. For the refinement procedure, Eq. 4.25 is used in order to convert the set \mathbf{p}_j^g into the set \mathbf{p}_j^n . The minimization problem in Eq. 4.40 can be solved by using the *Levenberg-Marquardt algorithm* [43, 49].

$$\min_{\mathbf{P}} \sum_{j=1}^n \left\| \begin{bmatrix} \frac{P_1 \tilde{\mathbf{p}}_j^g}{P_3 \tilde{\mathbf{p}}_j^g} \\ \frac{P_2 \tilde{\mathbf{p}}_j^g}{P_3 \tilde{\mathbf{p}}_j^g} \end{bmatrix} - \mathbf{p}_j^n \right\|_2^2 \quad (4.40)$$

4.4. Triangulation

The reconstruction of the surface of an object is carried out by triangulation. In the following, a linear and a non-linear triangulation method is described. The non-linear triangulation uses the result of the linear triangulation as an initial guess.

4.4.1. Linear triangulation

The triangulation method used in this work is based on the model of a calibrated stereo camera rig. The triangulation of a spatial point \mathbf{p}^g is carried out by using the corresponding image points \mathbf{p}^n and \mathbf{p}^{n^*} which are given in the image I and I^* , respectively (see Fig. 4.5). Furthermore, the pose between the two cameras is utilized. The triangulation model is formulated by means of the projection matrix of both cameras. As explained in Sec. 4.3.2, the projection matrix \mathbf{P} for a normalized image point \mathbf{p}^n is defined by Eq. 4.39. The projection matrix of the first camera is defined as follows. For triangulation, the object coordinate system $(CS)_g$ coincides with the camera coordinate system $(CS)_c$. Hence, the translation vector ${}^c\mathbf{t}_g$ becomes the null vector $\mathbf{0}_{3 \times 1}$ and the rotation matrix ${}^c\mathbf{R}_g$ becomes the identity matrix $\mathbf{I}_{3 \times 3}$:

$$\tilde{\mathbf{p}}^n = \underbrace{\begin{bmatrix} {}^c\mathbf{R}_g & {}^c\mathbf{t}_g \end{bmatrix}}_{\mathbf{P}} \tilde{\mathbf{p}}^g = \underbrace{\begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 1} \end{bmatrix}}_{\mathbf{P}} \tilde{\mathbf{p}}^g. \quad (4.41)$$

Since $(CS)_g$ coincides with $(CS)_c$, the projection matrix \mathbf{P}^* of the second camera contains

the pose of the first camera coordinate system $(CS)_c$ with respect to the second camera coordinate system $(CS)_{c^*}$. This pose is defined by the translation vector ${}^{c^*}t_c$ and the rotation matrix ${}^{c^*}R_c$:

$$\tilde{\mathbf{p}}^{n^*} = \underbrace{\begin{bmatrix} {}^{c^*}R_g & {}^{c^*}t_g \end{bmatrix}}_{P^*} \tilde{\mathbf{p}}^g = \underbrace{\begin{bmatrix} {}^{c^*}R_c & {}^{c^*}t_c \end{bmatrix}}_{P^*} \tilde{\mathbf{p}}^g. \quad (4.42)$$

The triangulation of a spatial point \mathbf{p}^g is solved by means of the DLT method described in Sec. 4.2. Consequently, the relationships in Eqs. 4.41 and 4.42 can be expressed by the cross product:

$$\begin{aligned} \mathbf{0}_{3 \times 1} &= \tilde{\mathbf{p}}^n \times P \tilde{\mathbf{p}}^g, \\ \mathbf{0}_{3 \times 1} &= \tilde{\mathbf{p}}^{n^*} \times P^* \tilde{\mathbf{p}}^g. \end{aligned} \quad (4.43)$$

By reordering the resulting equations, the following system of linear equations can be defined:

$$\mathbf{0}_{4 \times 1} = \underbrace{\begin{bmatrix} x_n P_3 - P_1 \\ y_n P_3 - P_2 \\ x_{n^*} P_3^* - P_1^* \\ y_{n^*} P_3^* - P_2^* \end{bmatrix}}_A \tilde{\mathbf{p}}^g. \quad (4.44)$$

The value 1 is assigned to the third component of $\tilde{\mathbf{p}}^{n^*}$ and $\tilde{\mathbf{p}}^n$ in Eq. 4.43. Thus, Eq. 4.44 only includes the Euclidean coordinates x_n, y_n, x_{n^*} and y_{n^*} . Moreover, the vectors P_1, P_2 and P_3 define the rows of the projection matrix P . Accordingly, the vectors P_1^*, P_2^* and P_3^* define the rows of the projection matrix P^* . In order to avoid the trivial solution $\tilde{\mathbf{p}}^g = \mathbf{0}$, the estimation of $\tilde{\mathbf{p}}^g$ is formulated by a constrained minimization problem:

$$\begin{aligned} \min_{\tilde{\mathbf{p}}^g} & \| A \tilde{\mathbf{p}}^g \|_2^2, \\ \text{subject to} & \| \tilde{\mathbf{p}}^g \|_2^2 = 1. \end{aligned} \quad (4.45)$$

As explained in Sec. 3.3.2, this minimization problem can be solved by applying the

method of Lagrange multipliers. After some steps the following eigenvalue problem can be defined [27]:

$$\mathbf{A}^T \mathbf{A} \tilde{\mathbf{p}}^g = \lambda \tilde{\mathbf{p}}^g. \quad (4.46)$$

A solution $\tilde{\mathbf{p}}^g$ that differs from the null vector is an eigenvector of the matrix $\mathbf{A}^T \mathbf{A}$. The sought solution for $\tilde{\mathbf{p}}^g$ is given by the eigenvector that corresponds to the smallest eigenvalue. This solution can be comfortably computed by applying the SVD to the matrix \mathbf{A} (see Sec. A.4):

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T. \quad (4.47)$$

The sought solution for $\tilde{\mathbf{p}}^g$ is given by the eigenvector that is located in the last column of \mathbf{V} [27]. In the literature, the described linear triangulation is also referred to as *linear-eigen method* [26].

4.4.2. Non-linear triangulation

By means of the projective mappings in Eqs. 4.41 and 4.42, the non-linear least-squares problem in Eq. 4.48 can be formulated. The initial guess for the sought spatial point \mathbf{p}^g is given by the result of the linear triangulation described in Sec. 4.4.1. The minimization problem in Eq. 4.48 can be solved by using the *Levenberg-Marquardt algorithm* [43, 49].

$$\min_{\mathbf{p}^g} \left\| \begin{bmatrix} \frac{P_1}{P_3} \tilde{\mathbf{p}}^g \\ \frac{P_2}{P_3} \tilde{\mathbf{p}}^g \end{bmatrix} - \mathbf{p}^n \right\|_2^2 + \left\| \begin{bmatrix} \frac{P_1^*}{P_3^*} \tilde{\mathbf{p}}^g \\ \frac{P_2^*}{P_3^*} \tilde{\mathbf{p}}^g \end{bmatrix} - \mathbf{p}^{n^*} \right\|_2^2 \quad (4.48)$$

4.5. Epipolar geometry

As described in Sec. 4.4, the triangulation of a spatial point \mathbf{p}^g is carried out by using the corresponding image points \mathbf{p}^n and \mathbf{p}^{n^*} which are located in the image I and I^* ,

respectively. In this work, the second camera is replaced by a mirror based projection system that projects a grid of laser spots onto the surface of an object. In this context, each laser ray can be considered as an inverse camera. Furthermore, each inverse camera is associated with a virtual image I^* .

When a laser spot is projected onto the surface of an object then it's position p^{n^*} in the virtual image I^* is known. This relationship will be explained later in the context of the calibration of the laser scanning system in Sec. 5. After the projection of the laser spot, the corresponding image point p^n has to be detected in the image I . For this purpose, the structured light encoding method described in Sec. 3.2 is utilized. To verify that an image point p^n was identified correctly, the concept of epipolar geometry [27] is utilized in this work.

The epipolar geometry for two cameras is illustrated in Fig. 4.7. A spatial point p^g is projected on the corresponding image points p^n and p^{n^*} which are given in the image I and I^* , respectively. The points p^g and p^{n^*} are connected by a projective ray that intersects with the camera center c^* . The 2D representation of this projective ray is given by the epipolar line l in the image I . Consequently, the image point p^n is located on the epipolar line l . This relationship is also known as the epipolar constraint.

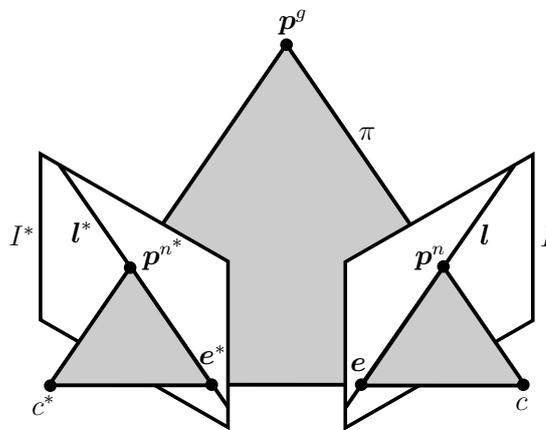


Figure 4.7.: Epipolar geometry.

The described principle also holds for the second camera which leads to an epipolar line l^* in the image I^* . The plane that contains the epipolar lines l and l^* is referred to as the epipolar plane π . Further quantities in the scope of epipolar geometry are given by the epipoles e and e^* . As explained, the camera center c^* is located on the projective ray that connects the points p^g and p^{n^*} . The 2D representation of the camera center c^* is

referred to as the epipol e which is located on the epipolar line l . Since each projective ray of the second camera intersects with the camera center c^* , the epipol e is the point of intersection of all epipolar lines in the image I . The described principle also holds for the second camera which leads to the epipol e^* in the image I^* .

The mathematical representation of the epipolar geometry is defined by the (3×3) fundamental matrix F [27]. The fundamental matrix in Eq. 4.49 is computed by means of the pose of the second camera coordinate system $(CS)_{c^*}$ with respect to the first camera coordinate system $(CS)_c$. This pose is given by the translation vector ${}^c t_{c^*}$ and the rotation matrix ${}^c R_{c^*}$.

$$F = S({}^c t_{c^*}) {}^c R_{c^*} \quad (4.49)$$

The term $S(\cdot)$ defines the skew-symmetric matrix of a vector. The matrix F is used to map an image point p^{n*} to the epipolar line $l = [a \ b \ c]^T$:

$$l = F \tilde{p}^{n*}. \quad (4.50)$$

By means of the epipolar line l , the following homogeneous linear equation can be defined:

$$0 = (\tilde{p}^n)^T l. \quad (4.51)$$

Equation 4.51 is satisfied by the image point p^n since it corresponds to the image point p^{n*} . As explained in the beginning of this section, the epipolar geometry is utilized to verify that an image point p^n was identified correctly. This is realized by calculating the orthogonal distance d of p^n with respect to the epipolar line l . For this purpose, two arbitrary points g_1^n and g_2^n on the epipolar line l are selected. The computation of the orthogonal distance d is given by [40]:

$$d = \frac{|\det([\ g_2^n - g_1^n \quad p^n - g_1^n])|}{\|g_2^n - g_1^n\|_2}. \quad (4.52)$$

A point p^n is considered to be identified correctly if the orthogonal distance is in the

4. Projective geometry

range $d \leq 1$ pixel. A tolerance is necessary due to image noise.

5. Calibration and surface reconstruction

This section describes the calibration of the extrinsic parameters of the laser scanning system. Furthermore, this section describes how the extrinsic parameters are utilized for the reconstruction of the centers of the projected laser spots.

For many 3D scanning applications it is common to use an off-the-shelf light projection system such as a LCD or DLP projector. As explained in [5] and [56], the model of central projection described in Sec. 4.1.1 can be used to model the projective mapping of those systems. In this context, a light projection system is considered as an inverse camera. In this case, the extrinsic parameters are given by the pose between the camera and the inverse camera. Hence, the calibration procedure is similar to the calibration of a stereo camera rig. The inverse camera approach is appropriate since the projective rays incorporate a point where all projective rays are intersecting.

However, in this work the light projection is realized by means of an infrared laser beam and two rotatable galvanometer driven mirrors (see Fig. 1.2). As described in Sec. 3.2, the projection of a grid of laser spots is realized by means of the movement of the mirrors and the switching of the laser source. The laser source sends the laser beam to the first mirror which redirects the beam towards the second mirror. The latter finally redirects the laser beam towards the surface of the object. Considering the projection of all laser spots, a bundle of laser rays exists which originates from the second mirror and which is sent towards the surface of the object. Depending on the orientation of the laser source and the orientation of the mirrors, this bundle of laser rays does not necessarily include a point where all laser rays intersect.

To achieve a point of intersection, a specific mechanical alignment of the laser source and the mirrors would be required. However, this mechanical alignment would be highly time consuming. Moreover, an ideal point of intersection can not be accomplished. Consequently, this approach is not utilized in this work. This also implies that the model of

central projection can not be used to describe the projective mapping of the galvanometric laser projection system.

A more general extrinsic calibration can be realized by using a physical-mathematical model of a galvanometric laser projection system. The model in [48] describes the orientation of the two galvanometer driven mirrors and the laser source. Depending on the turning position of both mirrors, the model is used to compute the path of a laser ray. [48] also describes a calibration method to compute the parameters of the proposed model. The drawback of this approach is given by the fact that the parameters of the model can not describe irregularities of the projection system. An example is given by imperfections of the mirror surfaces. The model in [48] assumes that the surface of both mirrors is given by an ideal plane. But in practice, small surface deviations will arise naturally during the fabrication of a mirror. Furthermore, it is very likely that slight amounts of debris can be found on the surface of a mirror after it's mechanical installation.

All imperfections of the mirror surfaces will lead to slight changes of the path of a laser ray. These changes are not considered by the model described in [48]. Hence, the calibration can lead to a coarse estimation of the model parameters. In [91], an experimental analysis of the triangulation accuracy was carried out on the basis of the calibration of the model described in [48]. The accuracy analysis was carried out by calculating the deviation of the triangulated points with respect to a planar surface. The achieved triangulation accuracy was given by a root mean square (RMS) error of around 1.5 mm. Consequently, the model described in [48] yields a very coarse triangulation accuracy.

To avoid such problems, a machine learning approach can be utilized. [91] proposed a machine learning approach to calibrate the parameters of a galvanometric laser projection system. The approach is based on the turning positions of the mirrors and the laser centers in the camera image. [91] utilized three different machine learning approaches given by an *artificial neural network* (ANN) [9], a *support vector regression* (SVR) [77] and a *Gaussian process* (GP) [69]. As before, the accuracy analysis was carried out by calculating the deviation of the triangulated points with respect to a planar surface. The analysis of the triangulation accuracy revealed RMS errors between 0.4 to 0.5 mm. The best results were achieved by means of the SVR and GP approach.

The drawback of a machine learning approach is given by high computational costs. As reported in [91], the training of an ANN or a SVR model can take several minutes. The training of a GP model can even take one to two days. Another disadvantage of

machine learning is given by the fact that an accurate generalization of the model can not be guaranteed. The interpolation result for an a priori unseen feature vector can incorporate unexpected inaccuracies.

To avoid all of the previously described disadvantages, this work follows a different approach. As described, the laser scanning system projects a grid of discrete laser spots onto the surface of an object. Thus, a set of laser rays exists which originates from the second mirror and which is sent towards the surface of the object. To achieve a highly accurate estimation of the sought extrinsic parameters, each laser ray is calibrated independently. In this manner, the true path of each laser ray can be estimated. The extrinsic calibration of a laser ray is carried out by computing the translation and the direction of the laser ray with respect to the camera. In the following, this procedure is called *explicit laser ray calibration*.

The developed calibration procedure is used to achieve a highly accurate surface reconstruction. In this context, only one set of laser rays has to be calibrated. When new turning positions of the mirrors have to be utilized then the parameters of the corresponding laser rays are computed by means of interpolation. In this manner, a time consuming acquisition of new calibration data can be avoided. A new calibration is only required if the position or orientation of one of the optical components in the laser scanning system is changed during reassembly. The developed calibration procedure requires only very low computational costs and terminates after a few seconds. The same applies for the interpolation of laser rays. Solely the acquisition of calibration data is time consuming. However, this is also the case for all previously described calibration approaches.

5.1. Explicit laser ray calibration

For the calibration of a set of laser rays, a calibration body is constructed as explained in Sec. 4.3. The calibration pattern presented in Fig. 4.6 is printed on a planar board. Usually, a calibration pattern contains black squares to provide a high contrast. However, a grid of laser spots is projected onto the calibration body. In order to minimize the absorption of light energy, the used calibration pattern contains gray squares.

In the following, the acquisition of calibration data is explained. One extended code sequence (see Sec. 3.2) is projected onto the calibration body which is placed parallel to

the front side of the laser scanning system. The utilized projection distance is referred to as d_1 . In order to generate a high amount of full frames, the length of each sub-sequence is defined by two frames. This configuration is useful for the calibration since the image data of the full frames is used for averaging. The described configuration is only fixed for the calibration. After calibration, the length of the sub-sequences can be changed to meet individual requirements during the process of surface reconstruction.

The camera captures the extended code sequence that was projected onto the calibration body. As exemplarily shown in Fig. 5.1, each projected grid is captured in one camera image. Besides one extended code sequence, the camera captures one additional image which shows the whole calibration body without projected laser spots. In the following, this additional image is called chessboard frame.

The described data acquisition is repeated in the same manner for further projection distances. The second projection distance is given by $d_2 = d_1 - 1$ cm and the third projection distance is given by $d_3 = d_1 + 1$ cm. This alternation is continued and each time the shifting value is increased by 1 cm. Consequently, the fourth projection distance is defined by $d_4 = d_1 - 2$ cm and the fifth projection distance is given by $d_5 = d_1 + 2$ cm. Subsequent projection distances are calculated in the same manner.

The set of utilized projection distances is referred to as d_i with $i \in [1, I]$ where I defines the amount of projection distances. The definition of I is explained later in Sec. 6. In the following, the processing of the captured camera images is described with regards to one projection distance. This procedure involves the processing of one captured extended code sequence and one chessboard frame. The procedure is carried out in the same manner for each of the I projection distances. At first, the captured extended code sequence is processed by the code detection procedure explained in Sec. 3.2. This results in the identification of the laser spots that are tracked in the captured camera images. For the purpose of calibration, the centers of the tracked laser spots are stored for each full frame. Since a tracked laser spot is seen in all full frames, the stored data is used to compute the mean center p^n for each tracked laser spot. The resulting set of mean laser centers is referred to as p_j^n with $j \in [1, J]$ where J defines the number of mean laser centers.

The computation of mean laser centers is carried out in order to remove the noise from the data. Figure 5.2 shows a typical distribution of the pixel error of all stored laser centers. The pixel error is given by the mean free laser centers. Figure 5.2 shows that the pixel error follows a Gaussian distribution. The maximum L2-norm of the pixel error is given

by approximately 0.5 pixel. The pixel error of the laser centers is caused by image noise. Another reason is given by the positioning accuracy of the galvanometer driven mirrors.

The processing of the chessboard frame is carried out according to the pose estimation procedure described in Sec. 4.3. As a result of this procedure, the quantities cT_g and \mathbf{H} are given. cT_g defines the estimated pose of the calibration body with respect to the camera. \mathbf{H} describes the homography matrix which maps the 2D corners of the calibration body to the corresponding 2D corners in the chessboard frame. By using the inverse of the homography matrix \mathbf{H} , the mean laser centers \mathbf{p}_j^n are mapped to the corresponding laser centers \mathbf{p}_j^q in the x-y plane of the calibration body:

$$\tilde{\mathbf{p}}^q = \mathbf{H}^{-1} \tilde{\mathbf{p}}^n. \quad (5.1)$$

Using the relationship described in Eq. 4.25, the laser centers \mathbf{p}_j^q in the x-y plane of $(CS)_g$ are mapped to their 3D representation \mathbf{p}_j^g . By means of the estimated pose cT_g , the laser centers \mathbf{p}_j^g are transformed to the laser centers \mathbf{p}_j^c with respect to the camera (see Eq. A.5).

The explained processing of the captured extended code sequence and the chessboard frame is carried out in the same manner for each of the I projection distances. Consequently, a set \mathbf{p}_j^c of spatial laser centers is calculated for each of the I projection distances. Each set \mathbf{p}_j^c is sorted according to the identified code words. For the calibration of the laser rays, the sorted sets are combined in the collection $\mathbf{p}_{i,j}^c$ which is structured as a set of sets. Due to the described sorting, the collection $\mathbf{p}_{i,j}^c$ can be used to access a set \mathbf{p}_i^c of spatial laser centers. A set \mathbf{p}_i^c is related to one code word and includes the spatial laser centers for all I projection distances. Thus, a set \mathbf{p}_i^c is related to one single laser ray.

On this basis, a set of laser rays can be calibrated with respect to the camera. The calibration of a laser ray is carried out by means of *principal component analysis* (PCA). Given a set \mathbf{p}_i^c , the PCA calculates the covariance matrix of this set (see Sec. A.5). The result of the PCA is given by the eigenvectors and the eigenvalues of the covariance matrix. The set of eigenvectors is defined by \mathbf{u}_w and the set of eigenvalues is defined by λ_w with $w \in [1, 3]$. The amount of eigenvectors and eigenvalues is equivalent to the dimension of the spatial laser centers in the set \mathbf{p}_i^c . The eigenvalues λ_w are arranged in descending order $\lambda_1 \geq \dots \geq \lambda_3 > 0$. The corresponding (3×1) eigenvectors \mathbf{u}_w are orthonormal vectors and define the principal components of the spatial laser centers \mathbf{p}_i^c .

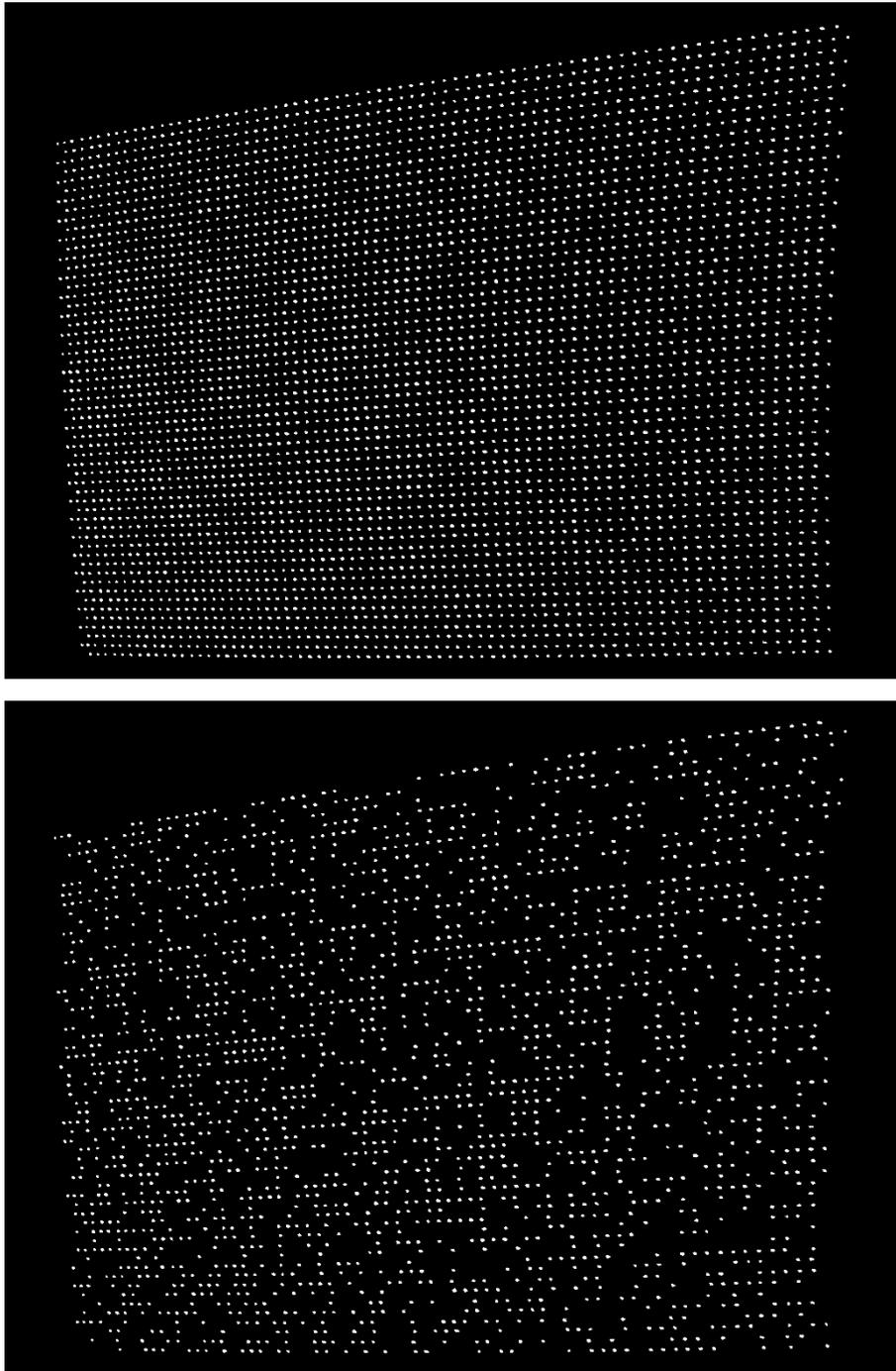


Figure 5.1.: This figure exemplarily shows a grid of 72×56 (4032) laser spots which is projected onto the calibration body. The binary representation of the indices of the laser rays (range: 0 to 4031) defines the code words. The encoding is realized by means of $n = 12$ code frames. The length of each sub-sequence is defined by $j = 2$ and thus the number of sub-sequences is given by $v = 12$. The images show the projection of the first sub-sequence of an extended code sequence.

As mentioned, the extrinsic calibration of a laser ray is carried out by computing the translation and the direction of the laser ray with respect to the camera. The sought direction of the laser ray is given by the direction of highest variation in the associated point set p_i^c . Hence, the sought direction of the laser ray is defined by the eigenvector in u_w that corresponds to the largest eigenvalue in λ_w . Due to the descending order of the eigenvalues λ_w , the direction of the laser ray is given by the eigenvector u_1 .

Usually, the orientation of a laser ray is sufficiently described by a single vector. Nevertheless, the calibrated laser rays are the basis for triangulation which involves the use of projection matrices. For this reason, the orientation of a laser ray has to be described by a (3×3) rotation matrix. In this context, a laser ray is modeled as an inverse camera which projects one single point along the optical axis. An inverse camera is referred to as laser coordinate system $(CS)_l$. The orientation of $(CS)_l$ with respect to the camera is defined by the rotation matrix cR_l .

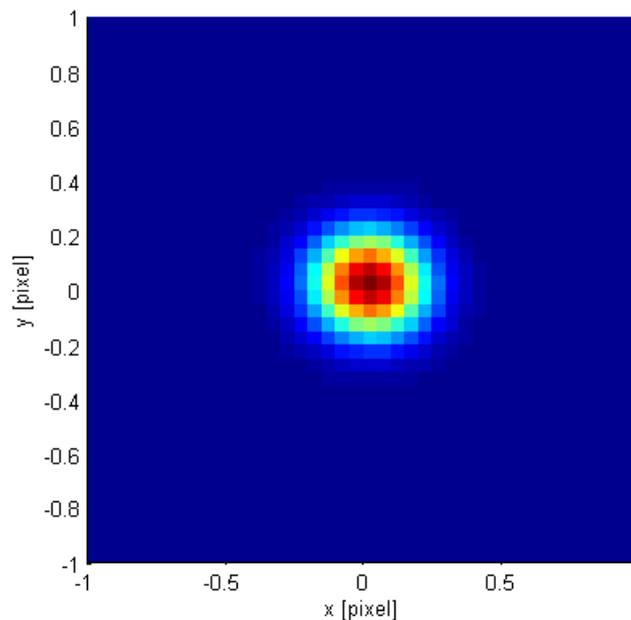


Figure 5.2.: Typical Gaussian distribution of the pixel error of all laser centers that are used for the calibration of the laser rays.

The rotation matrix cR_l of a laser ray is determined by the following procedure. As described, the direction of a laser ray is given by the eigenvector u_1 . This eigenvector is used as the optical axis of the laser coordinate system $(CS)_l$ and defines the z-axis of the

rotation matrix ${}^c\mathbf{R}_l$. The eigenvectors \mathbf{u}_w which result from the PCA already represent an orthonormal basis and can be used to determine the remaining entries of ${}^c\mathbf{R}_l$. The eigenvector \mathbf{u}_2 is utilized as the x-axis of ${}^c\mathbf{R}_l$. The cross product of \mathbf{u}_1 and \mathbf{u}_2 is used to determine the y-axis. Hence, the complete rotation matrix ${}^c\mathbf{R}_l$ is given by:

$${}^c\mathbf{R}_l = [\mathbf{u}_2 \quad \mathbf{u}_1 \times \mathbf{u}_2 \quad \mathbf{u}_1]. \quad (5.2)$$

The (3×1) translation vector ${}^c\mathbf{t}_l$ of a laser ray is computed by the mean vector of the respective set \mathbf{p}_i^c of spatial laser centers:

$${}^c\mathbf{t}_l = \frac{1}{I} \sum_{i=1}^I \mathbf{p}_i^c. \quad (5.3)$$

In order to calibrate all laser rays, the described procedure is carried out to process each set \mathbf{p}_i^c that is contained in the collection $\mathbf{p}_{i,j}^c$. The result is given by the set of translation vectors ${}^c\mathbf{t}_{l(j)}$ and the set of rotation matrices ${}^c\mathbf{R}_{l(j)}$. The notation $l(j)$ is used to describe the index $j \in [1, J]$ which is associated to a laser coordinate system $(CS)_l$.

5.2. Interpolation of laser rays

Given a set of calibrated laser rays, new laser rays can be interpolated. An interpolation of laser rays is necessary if new turning positions of the galvanometer driven mirrors have to be utilized. The turning position of both mirrors is determined by a program that uses a 16-bit register for the steering of each mirror. Consequently, each laser ray is associated with a register value pair $\mathbf{v} = [\text{reg}_x \quad \text{reg}_y]^T \in \mathbb{N}^2$. Regarding a set of calibrated laser rays, the set of associated register value pairs defines a grid with equidistant steps in both directions. As explained in Sec. 5.1, a set of calibrated laser rays is described by the set of translation vectors ${}^c\mathbf{t}_{l(j)}$ and the set of rotation matrices ${}^c\mathbf{R}_{l(j)}$ with $j \in [1, J]$. For the sake of clarity, the given set of calibrated laser rays is referred to as L_1 . The set of corresponding register value pairs is referred to as $V_1 = \mathbf{v}_j$. The elements of the sets V_1 and L_1 define the supporting points of the interpolation algorithm.

As exemplarily shown in Fig. 5.3, the interpolation of laser rays is carried out for new

register value pairs which are centered between the elements of V_1 . A centered position of the new register value pairs is chosen for the sake of simplicity. The following interpolation of laser rays can also be configured for arbitrary register value pairs which are located between the elements of the grid V_1 . The set of new register value pairs is referred to as $V_2 = v_k$ with $k \in [1, K]$. The quantity K defines the amount of new register value pairs. Since the new register value pairs are centered between the elements of V_1 , it follows that $K < J$. Using one element of the set V_2 , the interpolation of a laser ray is defined as follows. Since the interpolation is realized in an equidistant grid, the respective four enclosing elements of the set V_1 can be described by the corners of the cross product $[0, 1] \times [0, 1]$. Consequently, the interpolation is carried out for the register value pair $[0.5 \ 0.5]^T$. The interpolation algorithm utilizes the four calibrated laser rays of the set L_1 that correspond to the four used elements of V_1 . The algorithm requires two points on each of the four calibrated laser rays. The first point is already given by the translation vector ${}^c t_l$ and an additional point ${}^c g_l$ can be calculated on the optical axis:

$${}^c g_l = {}^c R_l [0 \ 0 \ 1]^T + {}^c t_l. \quad (5.4)$$

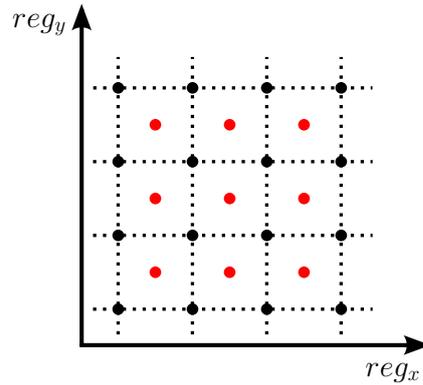


Figure 5.3.: The black dots represent the register value pairs of the given set V_1 . The red dots represent the new register value pairs from the set V_2 that are used as interpolation points. The dashed lines are given for the purpose of orientation.

Subsequently, a bilinear interpolation is carried out for the x , y and z component of the four used translation vectors $W_1 = \{ t_1, t_2, t_3, t_4 \}$. The same procedure is carried out for the four additional points $Q_1 = \{ g_1, g_2, g_3, g_4 \}$. Thus, the interpolation of a laser ray is based on six bilinear interpolations. In the following, the bilinear interpolation is defined in a general manner. For this purpose, the quantities μ_1, μ_2, μ_3 and μ_4 are utilized to represent either the x , y or z components of the elements of either W_1 or

Q_1 . Using the register value pairs $[0 \ 0]^T$, $[1 \ 0]^T$, $[0 \ 1]^T$ and $[1 \ 1]^T$, a bilinear interpolation for the register value pair $[0.5 \ 0.5]^T$ is carried out as follows. At first, a polynomial of degree one is computed which passes through the supporting points $[0 \ \mu_1]^T$ and $[1 \ \mu_2]^T$:

$$\begin{aligned} m_1 &= \mu_2 - \mu_1, \\ b_1 &= \mu_1. \end{aligned} \tag{5.5}$$

The parameter m_1 is the slope and the parameter b_1 is the axis intercept of the polynomial. The parameters are used to calculate the polynomial's function value s_1 for the value 0.5:

$$s_1 = 0.5m_1 + b_1. \tag{5.6}$$

Afterwards, a second polynomial of degree one is computed which passes through the supporting points $[0 \ \mu_3]^T$ and $[1 \ \mu_4]^T$:

$$\begin{aligned} m_2 &= \mu_4 - \mu_3, \\ b_2 &= \mu_3. \end{aligned} \tag{5.7}$$

The parameters are used to calculate the polynomial's function value s_2 for the value 0.5:

$$s_2 = 0.5m_2 + b_2. \tag{5.8}$$

Subsequently, a third polynomial of degree one is computed which passes through the supporting points $[0 \ s_1]^T$ and $[1 \ s_2]^T$:

$$\begin{aligned} m_3 &= s_2 - s_1, \\ b_3 &= s_1. \end{aligned} \tag{5.9}$$

Again, the parameters are used to calculate the polynomial's function value s_3 for the value 0.5:

$$s_3 = 0.5m_3 + b_3. \quad (5.10)$$

The value s_3 is the sought result of the bilinear interpolation for the register value pair $[0.5 \ 0.5]^T$. The described bilinear interpolation is carried out for the x , y and z component of the four translation vectors in the set W_1 . The same procedure is carried out for the four additional points in the set Q_1 . The result is given by two points ${}^c t_l$ and ${}^c g_l$ which represent the interpolated laser ray. The latter is associated to the register value pair $[0.5 \ 0.5]^T$. The point ${}^c t_l$ defines the translation vector of the interpolated laser ray with respect to the camera. The normalized difference of ${}^c t_l$ and ${}^c g_l$ represents the direction of the interpolated laser ray. Thus, this normalized difference also represents the z -axis of the coordinate system $(CS)_l$ of the interpolated laser ray:

$$\mathbf{z} = \frac{{}^c \mathbf{g}_l - {}^c \mathbf{t}_l}{\|{}^c \mathbf{g}_l - {}^c \mathbf{t}_l\|_2}. \quad (5.11)$$

The rotation matrix ${}^c \mathbf{R}_l$ of the interpolated laser ray is given by:

$${}^c \mathbf{R}_l = [\mathbf{x} \ \mathbf{y} \ \mathbf{z}]^T. \quad (5.12)$$

The x -axis and the y -axis of ${}^c \mathbf{R}_l$ are freely selectable with the restriction that the three axes have to represent an orthonormal basis. An axis vector $\mathbf{x} = [x_1 \ y_1 \ z_1]^T$ which is orthogonal to the axis vector $\mathbf{z} = [x_3 \ y_3 \ z_3]^T$ can be calculated by means of the dot product:

$$\mathbf{x}^T \mathbf{z} = |\mathbf{x}| |\mathbf{z}| \cos(90^\circ) = 0. \quad (5.13)$$

By choosing $\mathbf{x} = [1 \ 0 \ z_1]^T$, the only unknown value in Eq. 5.13 is given by z_1 . Thus, the value z_1 can be computed by:

$$z_1 = \frac{-x_3}{z_3}. \quad (5.14)$$

Subsequently, the following normalization is utilized to convert the resulting axis vector \boldsymbol{x} to a unit vector:

$$\boldsymbol{x} = \frac{\boldsymbol{x}}{|\boldsymbol{x}|}. \quad (5.15)$$

The axis vector \boldsymbol{y} is expressed by the following cross product:

$$\boldsymbol{y} = \boldsymbol{z} \times \boldsymbol{x}. \quad (5.16)$$

Hence, all axes of the rotation matrix ${}^c\boldsymbol{R}_l$ are calculated. The interpolated laser ray is given by the translation vector ${}^c\boldsymbol{t}_l$ and the rotation matrix ${}^c\boldsymbol{R}_l$ with respect to the camera. Furthermore, the interpolated laser ray is associated with the register value pair of the set V_2 that was used for interpolation. The described procedure is utilized to interpolate all laser rays that are associated with the register value pairs in the set V_2 . The set of interpolated laser rays is referred to as L_2 and includes the set of translation vectors ${}^c\boldsymbol{t}_{l(k)}$ and the set of rotation matrices ${}^c\boldsymbol{R}_{l(k)}$ with $k \in [1, K]$.

5.3. Surface reconstruction

As explained in Sec. 3.2, the laser scanning system permanently projects encoded grids of laser spots onto the surface of an object. After the captured full frames and code frames of the first extended code sequence are processed, the tracked laser spots are identified. Consequently, each full frame that is captured after the first extended code sequence is used for the reconstruction of the object's surface.

The reconstruction of the surface of an object is carried out by triangulation. The triangulation of a spatial laser center depends on the center of the captured laser spot and the corresponding calibrated laser ray. The calibration of the laser rays is carried out offline before the laser scanning system is used for surface reconstruction. Moreover, each calibrated laser ray is associated with its respective code word. Hence, the correspondence between the calibrated laser rays and the captured laser spots is established by means of the code detection procedure explained in Sec. 3.2.

Using the center of a captured laser spot and the corresponding calibrated laser ray, the triangulation of a spatial laser center is formulated as follows. In Sec. 4.4, the triangulation of spatial points was formulated for a calibrated stereo camera rig. The same triangulation method is applied to the calibrated laser scanning system. As described in Sec. 5.1, the triangulation method requires that a laser ray is modeled as an inverse camera which projects one single point along the optical axis. An inverse camera is referred to as laser coordinate system $(CS)_l$. As described in Sec. 5.1, a calibrated laser is given by the translation vector ${}^c\mathbf{t}_l$ and the rotation matrix ${}^c\mathbf{R}_l$ with respect to the camera. In the scope of triangulation, the first camera is described by the camera coordinate system $(CS)_c$ and the second camera is given by a laser coordinate system $(CS)_l$.

The center of a captured laser spot is described by the point \mathbf{p}^n . Furthermore, the projection matrix for the first camera is given in Eq. 4.41. The projection matrix for the laser coordinate system can be formulated by using Eq. 4.42. At first, Eq. 4.7 is utilized to extend the calibrated laser ray by a virtual model of central projection:

$$\mathbf{p}^{n*} = \frac{1}{z_l} \begin{bmatrix} x_l \\ y_l \end{bmatrix}. \quad (5.17)$$

Eq. 5.17 describes the projection of a spatial point $\mathbf{p}^l = [x_l \ y_l \ z_l]^T$ on the virtual image point $\mathbf{p}^{n*} = [x_{n*} \ y_{n*}]^T$. The spatial point \mathbf{p}^l is given in the laser coordinate system $(CS)_l$. As explained, a laser ray is modeled as an inverse camera which projects one single point along the optical axis. Thus, only one virtual image point exists. This point is given by $\mathbf{p}^{n*} = \mathbf{0}_{2 \times 1}$ and describes the point of intersection of the virtual image plane and the z-axis of the laser coordinate system.

The projection model in Eq. 4.42 uses the pose of the first camera with respect to the second camera. In the context of the laser scanning system, this relationship is now given by the translation vector ${}^l\mathbf{t}_c$ and the rotation matrix ${}^l\mathbf{R}_c$ with respect to the laser coordinate system $(CS)_l$. However, as described in Sec. 5.1, a calibrated laser ray is given by the translation vector ${}^c\mathbf{t}_l$ and the rotation matrix ${}^c\mathbf{R}_l$ with respect to the camera coordinate system $(CS)_c$. Hence, the required pose is calculated by (see Sec. A.2):

$$\begin{aligned} {}^l\mathbf{R}_c &= ({}^c\mathbf{R}_l)^T, \\ {}^l\mathbf{t}_c &= -({}^c\mathbf{R}_l)^T {}^c\mathbf{t}_l. \end{aligned} \quad (5.18)$$

Accordingly, Eq. 4.42 can now be formulated as follows:

$$[0 \ 0 \ 1]^T = \underbrace{\begin{bmatrix} {}^l\mathbf{R}_c & {}^l\mathbf{t}_c \end{bmatrix}}_{\mathbf{P}^*} \tilde{\mathbf{p}}^g. \quad (5.19)$$

The left side of Eq. 5.19 incorporates the homogeneous representation of the virtual image point $\mathbf{p}^{n^*} = \mathbf{0}_{2 \times 1}$. Using the projection model in Eqs. 4.41 and 5.19, the spatial laser center \mathbf{p}^g can be triangulated. The methods for linear and non-linear triangulation are described in Sec. 4.4. As explained in Sec. 4.5, the epipolar geometry is used to verify the code words of the captured laser spots. Using a calibrated laser ray, the computation of the fundamental matrix \mathbf{F} in Eq. 4.49 is now defined as:

$$\mathbf{F} = S({}^c\mathbf{t}_l) {}^c\mathbf{R}_l. \quad (5.20)$$

Using the virtual image point $\mathbf{p}^{n^*} = \mathbf{0}_{2 \times 1}$, the computation of the corresponding epipolar line in Eq. 4.50 is now given by:

$$\mathbf{l} = \mathbf{F}\tilde{\mathbf{p}}^{n^*} = \mathbf{F}[0 \ 0 \ 1]^T. \quad (5.21)$$

6. Accuracy analysis

The accuracy of triangulated laser centers depends on the deviations that occur in the developed laser scanning system. In this section, the sources of error of the laser scanning system are briefly described. Afterwards, an experimental analysis of the triangulation accuracy is carried out. The triangulation accuracy is analyzed for two different rigid phantoms and three human foreheads. Moreover, this section also covers an accuracy analysis for tracking. The developed laser scanning system was used for the tracking of different rigid phantoms and a human head. In this manner, it is possible to determine how the process of tracking is influenced by remaining inaccuracies of the triangulation. The experimental results presented in this section have also been partially published in [85–88].

6.1. Sources of error

Noise in the camera image leads to slight deviations during the detection of the laser centers. According to [1], the two common types of noise in a camera image are given by Gaussian white noise as well as salt and pepper noise. Reasons for image noise are given by high temperature or transmission errors in the electronic circuit. Gaussian white noise follows a Gaussian distribution whose parameters do not change over time. White noise is also referred to as a stationary process. Salt and pepper noise creates peaks in the camera image. In this case, the affected pixels are driven to their minimum or maximum intensity value. In general, image noise occurs at random pixel positions and thus it can not be predicted. Numerous filtering approaches exist to reduce image noise. However, the camera images in this work are only slightly affected by image noise. The captured laser spots are clearly visible and can be detected with high accuracy. For this reason, no filters are utilized to reduce image noise. In this manner, the computational costs for image processing are not increased unnecessarily. Furthermore, side effects of noise

reduction filters can be avoided.

A deviation of the detected laser centers can also be caused by image blur. The latter is due to the spherical aberration of the optical system. Spherical aberration is described as follows. Light rays which run parallel to the optical axis of a lens are ideally refracted if they intersect in one single point [42]. This point is defined as the focal point. A spherical lens is described by a surface section of a sphere and thus a single focal point does not exist. As a consequence, the camera image appears slightly blurry. It has to be mentioned that spherical aberration is independent from the effect of defocusing. An object which is perfectly focussed by the camera, can still occur slightly blurry in the camera image due to spherical aberration. According to [36], spherical aberration can be minimized but it can not be eliminated completely. To minimize spherical aberration, an aspheric lens can be used [54]. However, the production of an aspheric lens is expensive. Instead, it is more common to use a system of different spherical lenses. This system of simple lenses is also referred to as compound lens or camera lens.

Another error source of the laser scanning system is given by the two galvanometer driven mirrors. The positioning of the mirrors is achieved by a closed-loop controller and the desired turning positions are determined by a program that uses a 16-bit register for the steering of each mirror. When a grid of laser spots is projected, the program uses a configured register value pair for each laser spot. Ideally, the closed-loop controller would position the scanning mirrors as specified by the register value pairs. However, in practice the achieved mirror positions slightly deviate from the desired positions. During the calibration of the laser rays, this effect is compensated by means of averaging. Furthermore, the calibration method processes a set of spatial laser centers by means of a LS fitting procedure. Hence, the calibrated laser rays are supposed to represent the ideal laser rays that could be generated by an ideal mirror positioning. Accordingly, each calibrated laser ray corresponds to the desired register value pair.

As described, the achieved mirror positions slightly deviate from the desired positions. These mirror positioning errors affect the accuracy of triangulation. As soon as the controller causes an inaccurate mirror position, the generated laser ray slightly differs from the calibrated laser ray. The camera captures the reflection of the respective laser spot and the calibrated laser ray is used for triangulation. Hence, the calibrated laser ray affects the accuracy of triangulation if a positioning error is present for one of the mirrors. The calibrated laser ray has to be used since the generated laser ray is unknown.

A further error source of the laser scanning system is given by the intrinsic parameters of the camera which are computed by means of calibration (see Sec. 4.1.2). The calibration of the intrinsic parameters is based on the detection of corner features in the camera image. Thus, the calculated intrinsic parameters will slightly deviate from their true values.

As described in [34, 71], the accuracy of triangulation depends on the length of the baseline. The latter virtually connects the centers of the two camera coordinate systems. An increase of the baseline causes an increase of the triangulation accuracy. Consequently, the distance between the light projection system and the triangulation camera should be as high as possible. In the scope of this work, the length of the baseline is given by approximately 35 cm. Moreover, the accuracy of triangulation is also affected by the distance of the object's surface with respect to the laser scanning system. An increase of the object's distance causes a decrease of the triangulation accuracy. Besides other reasons, this effect is based on the image resolution. When the object is more far away then the resolution of the ROI is reduced. In this case, the detection of the laser centers becomes less accurate.

6.2. Triangulation accuracy for rigid objects

In order to analyze the triangulation accuracy of the developed laser scanning system, the surface of two different rigid phantoms is reconstructed. The first phantom is given by a planar surface and the second phantom is given by a printed stair model. The latter allows a more detailed analysis of the triangulation accuracy. The achieved triangulation accuracy is compared for calibrated and interpolated laser rays. Finally, Microsoft's Kinect camera is used to reconstruct the surface of the two phantoms. The resulting triangulation accuracy is compared to the triangulation accuracy of the developed laser scanning system.

Microsoft's Kinect camera represents an alternative for fast point-based surface reconstruction. The Kinect camera is mainly used as an input control in the entertainment sector. Nevertheless, the Kinect camera was also used for several medical motion monitoring applications [74, 95]. The advantage of the Kinect camera is that it offers a high reconstruction rate of 30 Hz and a high reconstruction density.

Commercially available devices for surface reconstruction often use the *single slit* ap-

proach which is described in Sec. 2. In the scope of this approach, a light slit is swept across the object's surface and the camera captures an image for every single slit. This approach can offer a high triangulation accuracy but the stepwise scanning leads to a very low reconstruction rate of far less than 1 Hz. In contrast, the use of light encoding offers a high reconstruction rate but it also leads to the fact that the surface reconstruction is more prone to errors.

Since the developed laser scanning system is based on light encoding, it would not be appropriate to compare the achieved triangulation accuracy with a commercially available device which follows the *single slit* approach. Instead, the Kinect camera is utilized for comparison. The Kinect camera uses an infrared light projector to project a specific one-shot pattern on the surface of an object [95]. Available images of the projected pattern show a grid of discrete light spots which vary in intensity and distance to each other. Hence, it is very likely that the Kinect camera follows the M-array encoding principle described in Sec. 2.1.2.

6.2.1. Triangulation accuracy for a plane

In order to analyze the accuracy of the triangulation, a planar surface is reconstructed. The error measure for triangulation is given by the perpendicular distance of the triangulated points with respect to the plane. This measure is also referred to as point-to-plane distance. The highest error of a triangulated point is given by the deviation along the optical axis of the camera. For this reason, the planar surface is placed parallel to the front side of the laser scanning system.

A triangulated point is given by \mathbf{p}^g with respect to an object coordinate system. In the scope of the triangulation procedure described in Sec. 5.3, the object coordinate system coincides with the camera coordinate system. For this reason, a triangulated point can also be referred to as \mathbf{p}^c with respect to the camera coordinate system. A reconstruction of the planar surface is given by the point set \mathbf{p}_i^c with $i \in [1, I]$ where I defines the number of triangulated points.

In the following, a plane is fitted to the point set \mathbf{p}_i^c . Subsequently, the sought perpendicular distance of the triangulated points with respect to the fitted plane is computed. The fitting of the plane is realized by means of the PCA which is explained in Sec. A.5. In this manner, the fitted plane can be described by a coordinate system. The PCA calculates the

covariance matrix of the point set \mathbf{p}_i^c . The result of the PCA is given by the eigenvectors and the eigenvalues of the covariance matrix. The set of eigenvectors is defined by \mathbf{u}_w and the set of eigenvalues is defined by λ_w with $w \in [1, 3]$. The amount of eigenvalues and eigenvectors is equivalent to the dimension of the spatial points contained in the set \mathbf{p}_i^c . The eigenvalues λ_w are arranged in descending order $\lambda_1 \geq \dots \geq \lambda_3 > 0$. The corresponding (3×1) eigenvectors \mathbf{u}_w are orthonormal vectors and define the principal components of the point set \mathbf{p}_i^c .

The eigenvectors \mathbf{u}_w are used to determine a rotation matrix ${}^c\mathbf{R}_h$ of the fitted plane with respect to the camera. The z-axis of ${}^c\mathbf{R}_h$ is given by the direction of smallest variation in the respective point set \mathbf{p}_i^c . Hence, the z-axis is defined by the eigenvector \mathbf{u}_3 . The x-axis and the y-axis of ${}^c\mathbf{R}_h$ span the fitted plane and are calculated by using a second eigenvector. The eigenvector \mathbf{u}_2 is utilized as the x-axis of ${}^c\mathbf{R}_h$. The cross product of \mathbf{u}_3 and \mathbf{u}_2 is used to determine the y-axis and hence the complete rotation matrix ${}^c\mathbf{R}_h$ is given by:

$${}^c\mathbf{R}_h = [\mathbf{u}_2 \quad \mathbf{u}_3 \times \mathbf{u}_2 \quad \mathbf{u}_3]. \quad (6.1)$$

The translation vector ${}^c\mathbf{t}_h$ of the fitted plane is defined as the mean vector of the point set \mathbf{p}_i^c :

$${}^c\mathbf{t}_h = \frac{1}{I} \sum_{i=1}^I \mathbf{p}_i^c. \quad (6.2)$$

The computed translation vector ${}^c\mathbf{t}_h$ and the rotation matrix ${}^c\mathbf{R}_h$ are used to define the homogeneous (4×4) transformation matrix ${}^c\mathbf{T}_h$ (see Eq. A.5). By means of the inverse transformation matrix ${}^h\mathbf{T}_c$, Eq. A.6 is used to transform the set of triangulated points \mathbf{p}_i^c to the point set \mathbf{p}_i^h with respect to the fitted plane. The third component of the resulting points \mathbf{p}_i^h defines the sought perpendicular distance of the triangulated points with respect to the fitted plane. The computed perpendicular distances are referred to as the 1-dimensional set e_i .

The triangulation accuracy for a planar surface is defined by the RMS of the computed triangulation errors e_i . The latter is a set of mean free error values which follow a Gaussian distribution. Consequently, the RMS is equivalent to the standard deviation. The

triangulation accuracy for a planar surface is computed as follows:

$$RMS = \sqrt{\frac{1}{I} \sum_{i=1}^I e_i^2}. \quad (6.3)$$

6.2.2. Triangulation accuracy for a stair phantom

The second phantom for the analysis of triangulation accuracy is given by a printed stair model. The stair model enables that triangulation errors can be considered for all three spatial directions. Thus, the triangulation accuracy can be analyzed in more detail than in the case of a planar surface. The stair phantom is presented in Fig. 6.1. It consists out of plastic material and was created by means of a 3D printer. The analysis of the triangulation accuracy for the stair phantom requires a ground truth point set of the phantom's surface.

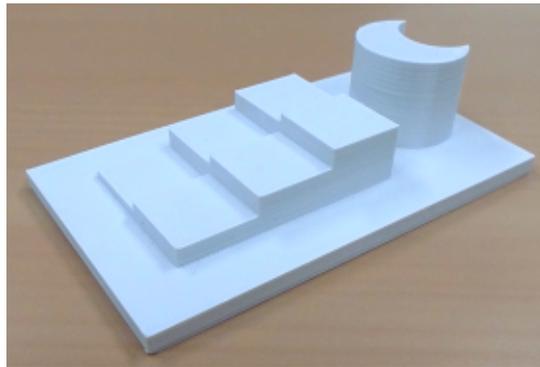


Figure 6.1.: Stair phantom of dimensions $12 \times 6 \text{ cm}^2$.

For this purpose, the stair phantom was reconstructed by means of a CT scanner (Siemens *SOMATOM Definition AS+*, voxel size $0.359 \times 0.359 \times 0.6 \text{ mm}^3$). Subsequently, a surface point set of the stair model was extracted from the CT imaging data. The extracted surface point set is used as the required ground truth. To analyze the triangulation accuracy for the stair phantom, a surface reconstruction is acquired by using the developed laser scanning system. The error measure for triangulation is given by the perpendicular distance of the triangulated points with respect to the ground truth surface. In order to calculate the triangulation errors, the set of triangulated points is registered to the ground truth point set. The registration process is defined by two steps given by a pre-alignment

and a refinement. The pre-alignment is carried out by manually registering the set of triangulated points to the ground truth point set. The resulting coarse registration is the initial solution for the subsequent refinement step.

The refining registration is carried out by means of the *iterative closest point* (ICP) algorithm. The most common variant of the ICP algorithm uses the point-to-point distance measure for registration. However, the set of triangulated points and the ground truth point set contain no or only few corresponding points. Thus, the use of the point-to-point based ICP would lead to registration errors. To avoid these registration errors, another variant of the ICP algorithm is used in this work. This variant uses the point-to-plane distance measure as explained in Sec. A.6.2. After the convergence of the ICP algorithm, the set of triangulated points and the ground truth point set are in a registered state. Consequently, the triangulation errors can be computed by using the difference vectors \mathbf{d}_i in Eq. A.17. Calculating the L2-norm of each difference vector \mathbf{d} in the set \mathbf{d}_i yields the sought perpendicular distance of the triangulated points with respect to the ground truth surface:

$$e = \|\mathbf{d}\|_2. \quad (6.4)$$

The triangulation errors are referred to as the 1-dimensional set e_i . The triangulation accuracy for the stair phantom is defined by the RMS of the triangulation errors e_i (see Eq. 6.3). The triangulation errors e_i are defined by the length of the difference vectors \mathbf{d}_i . Thus, the triangulation errors e_i are given by values ≥ 0 . As a matter of principle, the triangulated points are located before and behind the ground truth surface. This can also be revealed by the evaluation of the direction of the difference vectors \mathbf{d}_i . Consequently, it is also possible to describe the triangulation errors by positive and negative values. This leads to a set of mean free error values which follow a Gaussian distribution. However, the computation of the RMS value would lead to the same result as compared to the originally defined triangulation errors e_i . This is due to the fact that the computation of the RMS value involves the squaring of each triangulation error. For this reason, it is simpler to compute the RMS value by means of the originally defined triangulation errors e_i .

A disadvantage of the stair phantom is given by the circumstance that the ground truth point set is affected by the accuracy of the CT scanner. Hence, it can be assumed that the computed triangulation accuracy for the stair phantom is slightly worse than the true

triangulation accuracy. Nevertheless, the stair phantom is useful since triangulation errors can be considered for all three spatial directions. Since the triangulation accuracy for the stair phantom is affected, it is recommended to compute the triangulation accuracy for the planar surface as well. Since the use of a planar surface considers triangulation errors only in one direction, it can be assumed that the resulting triangulation accuracy is slightly better than the true triangulation accuracy. As a consequence, it can be assumed that the true triangulation accuracy is given by a value between the result of both methods.

6.2.3. Results and discussion

The laser scanning system was configured to project a grid of 72×56 (4032) laser spots. According to Table 3.1, this amount of laser spots is projected in approximately 0.4 s. At a projection distance of 60 cm, the dimensions of the projected grid are given by approximately $28.8 \times 22.4 \text{ cm}^2$. The system projects one extended code sequence on the surface of a phantom. 12 code frames are required to encode 4032 laser spots and a length of two frames is utilized for each sub-sequence. Thus, the length of the projected extended code sequence is given by 24 frames. After the captured full frames and code frames of the extended code sequence are processed, the tracked laser spots are identified. Subsequently, the laser scanning system captures one more full frame which is used for the reconstruction of the surface of the phantom. The set of triangulated points can be used to compute the accuracy of the triangulation.

Before a surface reconstruction can be carried out, a set of laser rays has to be calibrated. As explained in Sec. 5.1, a certain amount I of projection distances has to be utilized for the calibration. The projection distances are given by d_i with $i \in [1, I]$. The amount I of projection distances has an influence on the accuracy of the triangulation. An increase of I leads to a higher calibration accuracy and thus the resulting triangulation accuracy is increased as well. Nevertheless, there is a limiting value for I where an increase does not lead to a further increase of the triangulation accuracy.

Since the limiting value for I is not known beforehand, an iterative calibration of the laser rays is utilized in this work. In the first iteration $k = 1$, the first three projection distances d_1, d_2 and d_3 are used to calibrate the set of laser rays. The first projection distance is given by $d_1 = 60 \text{ cm}$ which is also the specified projection distance for the future head tracking

system. The second projection distance is given by $d_2 = d_1 - 1$ cm and the third projection distance is given by $d_3 = d_1 + 1$ cm. After calibration, the surface of the stair phantom is reconstructed to compute the triangulation accuracy as explained in Sec. 6.2.2. The stair phantom is utilized for the iterative calibration since it enables that triangulation errors can be considered for all three spatial directions.

In order to decide for the position of the stair phantom, the following circumstances have to be considered. The highest triangulation accuracy is achieved approximately in the middle of the used projection distances d_i . This is due to the fact that each laser ray is fitted to a set of spatial points which is associated with the used projection distances (see Sec. 5.1). Consequently, the stair phantom is placed at projection distance d_1 parallel to the front side of the laser scanning system. The reader might think that this is problematic since a part of the training data for calibration is also acquired at projection distance d_1 . However, the image data for calibration is acquired by using the planar calibration body described in Sec. 5.1. In contrast to the planar calibration body, the stair phantom incorporates variations in height. Thus, the resulting image data is different from the image data which is acquired for calibration.

In every following iteration $k \in [2; K]$, the projection distance d_{k+2} and all preceding projection distances d_1 to d_{k+1} are used to calibrate the laser rays. Hence, in each iteration, the number of used projection distances is increased by one. The computation of the projection distances is explained in Sec. 5.1. At the end of each iteration, the surface of the stair phantom is reconstructed to compute the current triangulation accuracy. In the scope of the iterative calibration, the triangulation accuracy is always evaluated for the projection distance $d_1 = 60$ cm. For this reason, the image data for the reconstruction of the surface of the stair phantom has to be acquired only once during iteration $k = 1$. This image data can be reused in every following iteration. The described iterative process terminates as soon as the triangulation accuracy for the stair phantom can not be increased anymore. The termination of the process is given at iteration $k = K$. Since the iterative calibration starts by using three projection distances, the final amount of used projection distances is given by:

$$I = K + 2. \tag{6.5}$$

To realize the positioning of the calibration body and the stair phantom at the specified projection distances, a robot manipulator was utilized. The laser scanning system was

mounted to the end-effector of an *Adept Viper s850* robot. To realize the specified projection distances, the end-effector of the robot was positioned accordingly.

The described iterative calibration of a set of laser rays was carried out for four configurations. Each configuration utilized a different method to detect the centers of the laser spots in the camera image. The detection methods are given by template matching, ellipse fitting as well as paraboloid fitting based on a 3×3 and a 5×5 sub-matrix. As explained in Sec. 3.3, each detection method has different advantages and disadvantages which affect the accuracy of the center detection. For each of the four configurations, the same acquired image data was used.

As explained in Sec. 5.3, a surface reconstruction can be acquired by means of the formulated linear and non-linear triangulation. For each of the four configurations, both triangulation methods are utilized in the scope of the computation of the triangulation accuracy for the stair phantom. In this manner, the behavior of both methods can be analyzed. The termination of each iterative calibration was determined by the evaluation of the increase of the triangulation accuracy. The termination is solely based on the triangulation accuracy of the linear method.

The results of the iterative calibration for all four configurations are shown in Figs. 6.2 to 6.5. For each configuration, the calibration terminates at a different final iteration K . Table 6.1 presents the triangulation accuracy at K for all four configurations. The use of ellipse fitting yields the highest triangulation accuracy. Furthermore, ellipse fitting achieves the fastest convergence of the iterative calibration.

For all four configurations, the following behavior can be seen in Figs. 6.2 to 6.5. Comparing the linear and the non-linear triangulation at iteration $k = 1$, the non-linear triangulation achieves an increase of accuracy between 0.005 mm and 0.008 mm. The iterative process increases the triangulation accuracy for both triangulation methods. This increase is larger for the linear triangulation and towards termination the linear triangulation achieves nearly the same accuracy as the non-linear triangulation. In fact, at the final iteration K , the result of both methods only differs between two to five tenths of a micrometer (see Table 6.1).

For the purpose of comparison, the curves for the linear triangulation are presented in Fig. 6.6 for all four configurations. It can be seen that paraboloid fitting causes the strongest curve oscillations and it also leads to the smallest triangulation accuracy for

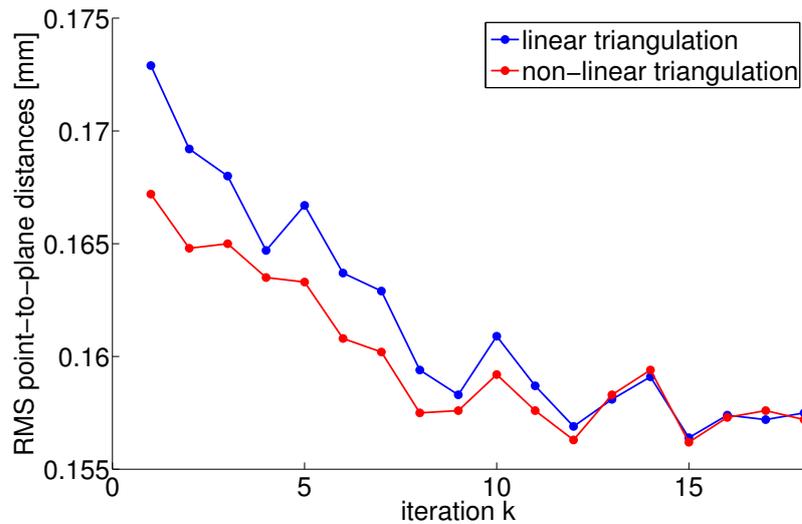


Figure 6.2.: Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of template matching.

every iteration. This is not surprising since image noise has a strong effect on paraboloid fitting. A better compensation for local variations can be achieved by using a larger sub-matrix. The utilized paraboloid fitting considers a sub-matrix of dimensions 3×3 or 5×5 . As can be seen, the sub-matrix of dimensions 5×5 already achieves a higher triangulation accuracy than the sub-matrix of dimensions 3×3 .

However, experiments for even larger sub-matrices showed that only a slight increase of the triangulation accuracy can be achieved and that template matching and ellipse fitting still outperforms paraboloid fitting. Actually, a large increase of the dimensions of the sub-matrix even leads to a counterproductive result given by a decreasing triangulation accuracy. The reason is that the model of a paraboloid is not the best choice when a large part of a captured laser profile is considered. For the sake of visual clarity, the curves for larger sub-matrices are not presented.

As can be seen in Fig. 6.6, the curve for template matching incorporates less oscillations than the curves for paraboloid fitting. Furthermore, template matching also leads to a higher triangulation accuracy for every iteration. This is due to the fact that the influence of image noise is reduced for template matching. The reason is that template matching processes the whole signal of each captured laser spot. Nevertheless, template matching uses a static reference signal to detect the highest similarity. For this reason, template matching can not compensate for the varying shape and scale of the captured laser spots.

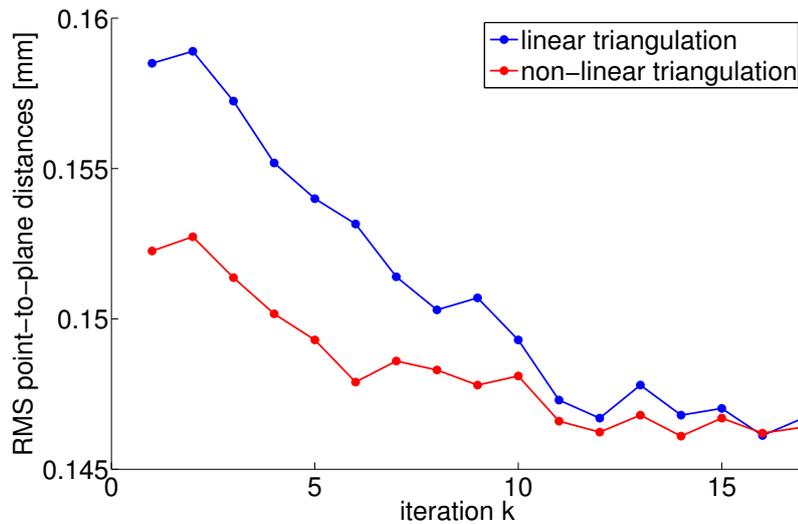


Figure 6.3.: Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of ellipse fitting.

The curve for ellipse fitting in Fig. 6.6 incorporates the least oscillations of all four configurations. Moreover, ellipse fitting yields the highest triangulation accuracy for every iteration. Just as paraboloid fitting, ellipse fitting does not depend on a reference signal and hence each laser spot is processed individually. Compared to paraboloid fitting, ellipse fitting can handle image noise in a more robust manner. The reason for this is given by the circumstance that ellipse fitting uses the extracted contour of a captured laser spot.

In Fig. 6.6, the results of the four configurations are compared by means of the linear

	Final iteration K	Linear triangulation	Non-linear triangulation
Template matching	18	0.1575	0.1572
Ellipse fitting	17	0.1467	0.1464
Paraboloid fitting 3×3	20	0.1837	0.1839
Paraboloid fitting 5×5	21	0.1702	0.1697

Table 6.1.: Triangulation accuracy in [mm] for the stair phantom at final iteration K .

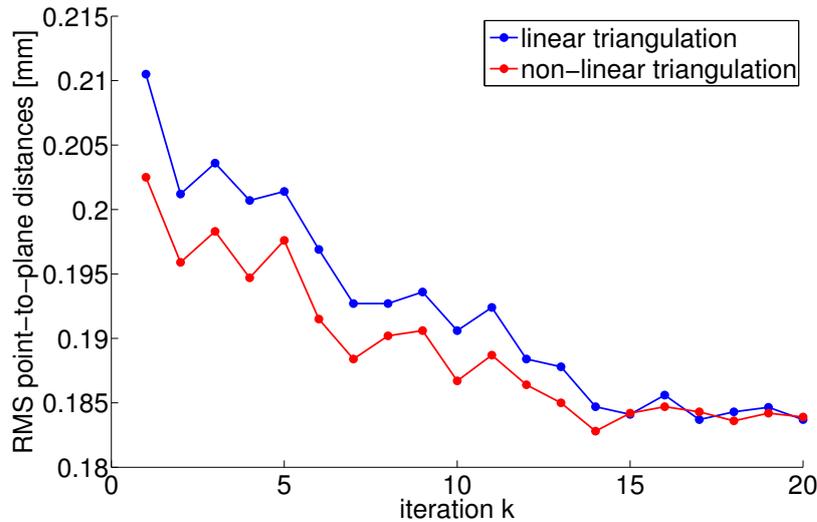


Figure 6.4.: Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of paraboloid fitting based on a 3×3 sub-matrix.

triangulation. A comparison of the curves for the non-linear triangulation leads to the same conclusions. For this reason, the curves for the non-linear triangulation are not shown together in one figure.

The results have shown that the linear and the non-linear triangulation achieve a very similar triangulation accuracy. At the termination of the iterative calibration, the results only differ by a few tenths of a micrometer. Furthermore, ellipse fitting achieved the highest triangulation accuracy. For this reason, ellipse fitting is chosen as the preferred method for the detection of the centers of the captured laser spots. Using ellipse fitting, the final iteration of the calibration was given by $K = 17$. The same iterative calibration based on ellipse fitting was repeated multiple times and the resulting final iteration was always given between $K = 15$ and $K = 18$.

According to Eq. 6.5, a maximum value of $K = 18$ implies that $I = 20$ projection distances are always sufficient for the calibration of a set of laser rays. This value is valid for a calibration setup that uses the projection distance $d_1 = 60$ cm. The latter was defined in the beginning of this section. Using $I = 20$ projection distances, a set of laser rays can be calibrated directly in one run. In this manner, an iterative calibration can be avoided.

The defined projection distance $d_1 = 60$ cm is also the specified projection distance for

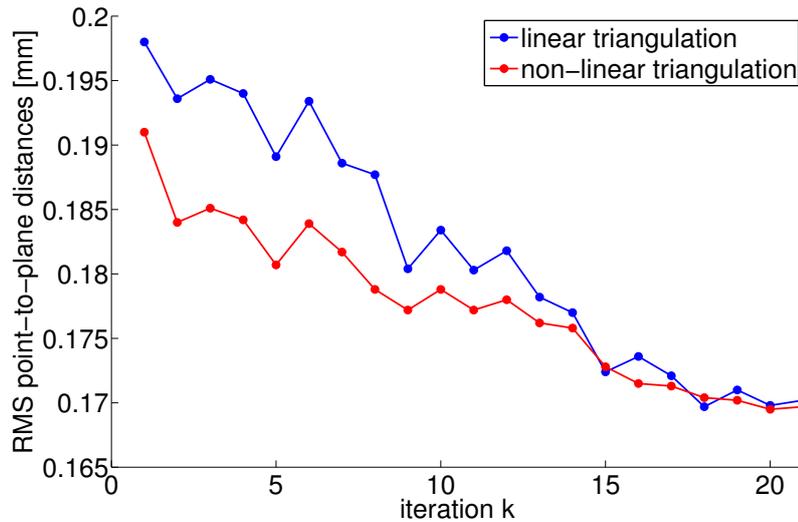


Figure 6.5: Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of paraboloid fitting based on a 5×5 sub-matrix.

the future head tracking system. If this distance has to be changed then the amount I of projection distances has to be redetermined by means of an iterative calibration. A change of d_1 clearly leads to a different triangulation accuracy. The further away d_1 is located the less accurate the triangulation accuracy becomes. This is due to the fact that the detection accuracy for the centers of the captured laser spots becomes less accurate if the used projection distances are located more far away (see Sec. 6.1). To compensate for this effect, a larger baseline can be considered for the laser scanning system. A larger baseline leads to a higher triangulation accuracy.

So far, the accuracy of the triangulation was always evaluated at projection distance $d_1 = 60$ cm. At the end of the iterative calibration, a final triangulation accuracy was achieved. After the termination of the calibration, it is also important to analyze the triangulation accuracy for the relevant scanning space. For this reason, the following experiment was carried out.

A set of laser rays was calibrated by using the defined amount of $I = 20$ projection distances. The first projection distance was again given by $d_1 = 60$ cm. To analyze the triangulation accuracy for the relevant scanning space, the stair phantom was placed at seven different positions parallel to the front side of the laser scanning system. The positions are defined by the set g_j with $j \in [1; 7]$. The positions are chosen so that they

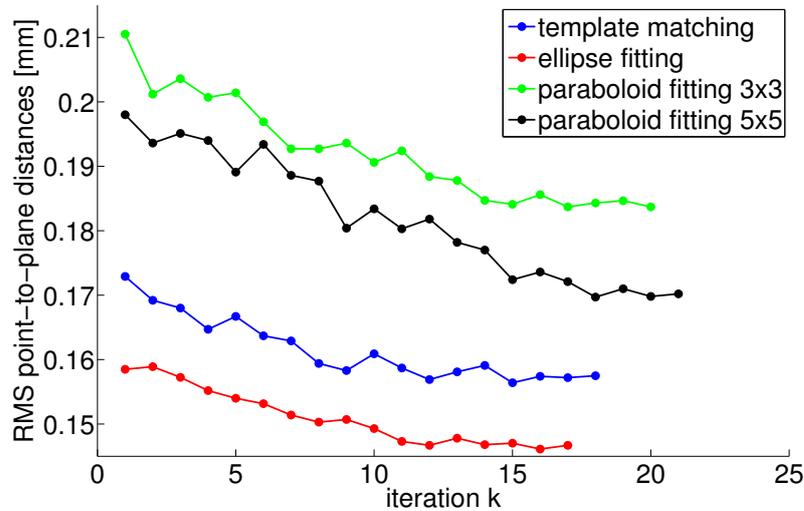


Figure 6.6.: Triangulation accuracy for the stair phantom. The curves show the results of the linear triangulation for all four center detection methods from Figs. 6.2 to 6.5.

differ from the projections distances d_i that were used for calibration. This is necessary since the following experiment was also carried out with a planar surface. Since the acquisition of the image data for calibration is based on a planar calibration body, the set g_j and the set d_i have to be different. Later in this section, the results for the stair phantom are compared to the results for the planar surface.

The first position is given by $g_1 = 47.5$ cm which is incremented by 4 cm for every subsequent position. Consequently, the last position is given by $g_7 = 71.5$ cm. In this manner, it is guaranteed that the positions g_1 and g_7 are located outside of the area that is spanned by the projection distances d_i . Since $I = 20$ projection distances are used, this area is defined between the distances 50 cm and 69 cm.

The triangulation accuracy for the stair phantom is presented in Fig. 6.7. At first, the blue and the red curve are analyzed which present the triangulation accuracy based on calibrated laser rays. The curves show the triangulation accuracy for all seven positions contained in g_j . The blue curve and the red curve show the accuracy which is based on the linear triangulation (LT) and the non-linear triangulation (NT), respectively. As can be seen, the highest triangulation accuracy is achieved for position $g_4 = 59.5$ cm. The latter defines the middle of the area that is spanned by the projection distances d_i . For the linear triangulation, the highest accuracy is given by 0.1477 mm. Compared to the outer positions g_1 and g_7 , the accuracy for the linear method only decreases by 0.0053 mm and

0.0071 mm, respectively. Position g_7 causes a larger decrease since it is located further away from the camera.

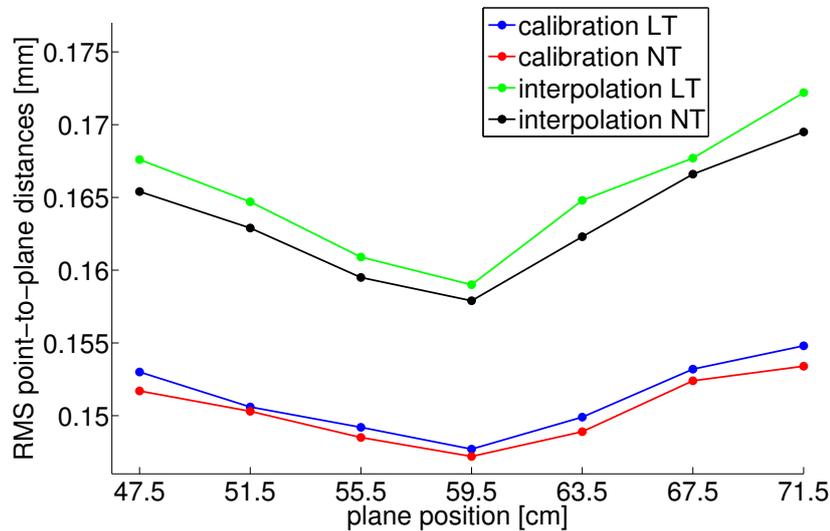


Figure 6.7.: Triangulation accuracy for the stair phantom. The blue and the red curve show the accuracy based on a set of calibrated laser rays. For comparison, the green and the black curve show the accuracy for the same set of laser rays which is computed by interpolation.

The accuracy of the triangulation behaves according to the principle of the calibration of the laser rays. As explained in Sec. 5.1, each laser ray is fitted to a set of spatial points which is associated with the used projection distances d_i . This implies that the highest triangulation accuracy can be achieved if a surface is placed approximately in the middle of the used projection distances.

The blue and the red curve in Fig. 6.7 show that the non-linear triangulation achieves only slightly better results compared to the linear method. In fact, the differences are marginal and vary around a few tenths of a micrometer. The smallest difference between both methods is achieved at position g_4 . Towards the outer positions g_1 and g_7 , the difference increases. Hence, the non-linear triangulation is able to slightly compensate for the decreasing accuracy of the linear method. However, the difference of the achieved accuracy is marginal. Consequently, it is sufficient to use the linear triangulation for surface reconstruction. The non-linear triangulation uses the result of the linear triangulation as an initial solution. By avoiding the calculation of the non-linear triangulation, computational costs can be reduced.

For comparison, the described experiment has also been carried out for interpolated laser rays. The interpolation of laser rays is explained in Sec. 5.2. A set of laser rays was calibrated and afterwards it was used for the interpolation of a new set of laser rays. This new set is an estimation of the set of calibrated laser rays which was used to produce the blue and the red curve in Fig. 6.7. The interpolated laser rays were used for the same surface reconstruction at the positions contained in g_j . To guarantee a direct comparison of the triangulation accuracy, the image data that has been used for surface reconstruction based on calibrated laser rays has been reused for surface reconstruction based on interpolated laser rays.

The triangulation accuracy for the stair phantom is presented in Fig. 6.7. The green and the black curve present the triangulation accuracy based on interpolated laser rays. The curves show the triangulation accuracy for all seven positions contained in g_j . The green curve and the black curve show the accuracy which is based on the linear triangulation (LT) and the non-linear triangulation (NT), respectively. At first, the accuracy of the linear method is compared for the set of calibrated laser rays and the set of interpolated laser rays. The respective differences between the blue curve and the green curve are presented in Table 6.2. The mean difference for both modalities is given by 0.0141 mm.

Plane position [cm]	Difference LT [mm]	Difference LT / calibration LT [%]
47.5	0.0146	9.54
51.5	0.0141	9.36
55.5	0.0117	7.84
59.5	0.0113	7.65
63.5	0.0149	9.94
67.5	0.0145	9.46
71.5	0.0174	11.24

Table 6.2.: This table presents the difference between the accuracy of the linear triangulation for calibrated and interpolated laser rays from Fig. 6.7.

Table 6.2 also presents the difference of the accuracy for both modalities in percentage. Here, the triangulation accuracy based on calibrated laser rays is utilized as the basis. The mean value is given by 9.29%. Consequently, the use of interpolation causes only a very small decrease of the triangulation accuracy. This implies that the use of a set of inter-

polated laser rays is an appropriate approach for surface reconstruction. In this manner, only one set of laser rays has to be computed by means of calibration. If a new set of laser rays has to be provided, the set of calibrated laser rays can be utilized for interpolation. Hence, a time consuming acquisition of new calibration data can be avoided.

Fig. 6.7 shows that the highest accuracy based on interpolated laser rays is again achieved for position g_4 . For the linear triangulation, the highest accuracy is given by 0.159 mm. Compared to the outer positions g_1 and g_7 , the accuracy for the linear method decreases by 0.0086 mm and 0.0132 mm, respectively. This decrease is larger than compared to the blue curve which is based on the calibrated laser rays. The reason is given by the fact that the deviation between the interpolated laser rays and the true path of the laser rays is increased. Thus, higher triangulation errors are generated.

Next, the linear and the non-linear triangulation based on interpolated laser rays are compared in Fig. 6.7. The curves show that the non-linear triangulation achieves slightly better results than the linear method. The differences vary around one to three micrometers. The smallest difference for both methods is achieved at position g_4 . Towards the outer positions g_1 and g_7 , the difference increases. Hence, the non-linear triangulation is able to slightly compensate for the decreasing accuracy of the linear method. This compensation is higher than in the case of the calibrated laser rays. An improvement of one to three micrometers can already be useful. Thus, the use of the non-linear triangulation can be considered for the case of interpolated laser rays. Nevertheless, it has to be taken into account that the execution of the non-linear triangulation can take up to three times longer than the execution of the linear triangulation.

As explained in Sec. 6.2.2, a disadvantage of the stair phantom is given by the fact that the ground truth point set is already affected by the accuracy of the CT scanner. Hence, it can be assumed that the computed triangulation accuracy for the stair phantom is slightly worse than the true triangulation accuracy. For this reason, the triangulation accuracy is also analyzed for a planar surface as described in Sec. 6.2.1. Since the use of a planar surface considers triangulation errors only in one direction, it can be assumed that the resulting triangulation accuracy is slightly better than the true triangulation accuracy. As a consequence, it can be assumed that the true triangulation accuracy of the laser scanning system is given by a value between the result of both methods.

The experiment for the planar surface was carried out in the same manner as for the stair phantom. The results for the planar surface are presented in Fig. 6.8. At first, the

blue and the red curve are analyzed which present the triangulation accuracy based on calibrated laser rays. The highest triangulation accuracy is again achieved for position g_4 . The highest accuracy for the linear triangulation is given by 0.1037 mm. For the stair phantom in Fig. 6.7, the highest accuracy for the linear method based on calibrated laser rays is given by 0.1477 mm. As explained, it can be assumed that the true accuracy for the linear method is located between these two values. The difference between both values is given by 0.044 mm.

In Fig. 6.8, the accuracy of the linear method is compared between position g_4 and the outer positions g_1 and g_7 . The accuracy of the linear method only decreases by 0.005 mm and 0.0067 mm, respectively. As in the case of the stair phantom, the non-linear triangulation achieves only slightly better results compared to the linear method. The differences between the blue and the red curve in Fig. 6.8 are marginal and vary around a few tenths of a micrometer.

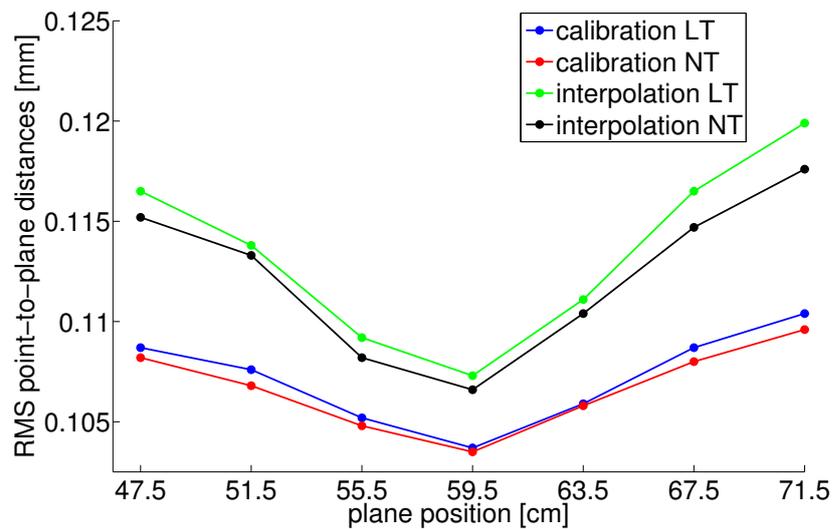


Figure 6.8.: Triangulation accuracy for a plane. The blue and the red curve show the accuracy based on a set of calibrated laser rays. For comparison, the green and the black curve show the accuracy for the same set of laser rays which is computed by interpolation.

Figure 6.8 also presents the triangulation accuracy which is based on interpolated laser rays. The highest triangulation accuracy is again achieved for position g_4 . The highest accuracy for the linear triangulation is given by 0.1073 mm. For the stair phantom in Fig. 6.7, the highest accuracy for the linear method based on interpolated laser rays is given by 0.159 mm. As explained, it can be assumed that the true accuracy for the linear

6. Accuracy analysis

method is located between these two values. The difference between both values is given by 0.0517 mm.

The blue curve and the green curve in Fig. 6.8 present the accuracy for the linear triangulation based on calibrated and interpolated laser rays, respectively. The differences between the blue and the green curve are presented in Table 6.3. The mean difference for both modalities is given by 0.0062 mm. Table 6.3 also presents the difference of the accuracy for both modalities in percentage. Here, the triangulation accuracy based on calibrated laser rays is utilized as the basis. The mean value is given by 5.74%. Consequently, the use of interpolation causes only a very small decrease of the triangulation accuracy.

Plane position [cm]	Difference LT [mm]	Difference LT / calibration LT [%]
47.5	0.0075	6.88
51.5	0.0062	5.76
55.5	0.0040	3.80
59.5	0.0036	3.47
63.5	0.0052	4.91
67.5	0.0078	7.18
71.5	0.0091	8.21

Table 6.3: This table presents the difference between the accuracy of the linear triangulation for calibrated and interpolated laser rays from Fig. 6.8.

Next, the results for the linear and the non-linear triangulation based on interpolated laser rays are compared in Fig. 6.8. As described, the highest triangulation accuracy is achieved for position g_4 . For the linear triangulation, the highest accuracy is given by 0.1073 mm. Compared to the outer positions g_1 and g_7 , the accuracy for the linear method only decreases by 0.0092 mm and 0.0126 mm, respectively. The non-linear triangulation achieves slightly better results than the linear method. The differences vary around one to two micrometers. As in the case of the stair phantom, the use of the non-linear triangulation can be considered for the case of interpolated laser rays.

In the last experiment of this section, Microsoft's Kinect camera was used to compute a reconstruction of the surface of the stair phantom and a reconstruction of a planar surface. The resulting triangulation accuracy is compared to the achieved triangulation accuracy

of the developed laser scanning system. The values for position g_4 in Figs. 6.7 and 6.8 are utilized for the comparison. For the purpose of surface reconstruction, the stair phantom was placed at the same position g_4 parallel to the front side of the Kinect camera. Afterwards, the triangulation accuracy for the stair phantom was computed according to Sec. 6.2.2.

The same procedure was carried out for the reconstruction of a planar surface. The respective triangulation accuracy was computed according to Sec. 6.2.1. The resulting triangulation accuracy for the developed laser scanning system and the Kinect camera are presented in Table 6.4. For the developed laser scanning system, four different methods are distinguished. The methods *calibration LT* and *calibration NT* describe the use of calibrated laser rays in combination with linear or non-linear triangulation, respectively. Moreover, the methods *interpolation LT* and *interpolation NT* describe the use of interpolated laser rays in combination with linear or non-linear triangulation, respectively.

As can be seen in Table 6.4, all four methods clearly outperform the triangulation accuracy of the Kinect camera. This is valid in the case of the planar surface as well as the stair phantom. Figure 6.9 shows the reconstruction of a planar surface that was acquired with the developed laser scanning system (a) and the Kinect camera (b). The former was acquired by means of the *calibration LT* method. The respective triangulation accuracies are presented in Table 6.4. Fig. 6.9 (a) shows that the developed laser scanning system computes a very accurate reconstruction of the planar surface. In contrast, Fig. 6.9 (b) shows that the Kinect camera generates a surface reconstruction which incorporates many small peaks.

As presented, the Kinect camera only achieves a very coarse triangulation accuracy. The main reason is given by the fact that the Kinect camera utilizes spatial multiplexing for the encoding of the projected light. As explained in Sec. 2, spatial multiplexing can lead to a higher amount of code detection errors. As a consequence, higher triangulation errors are generated.

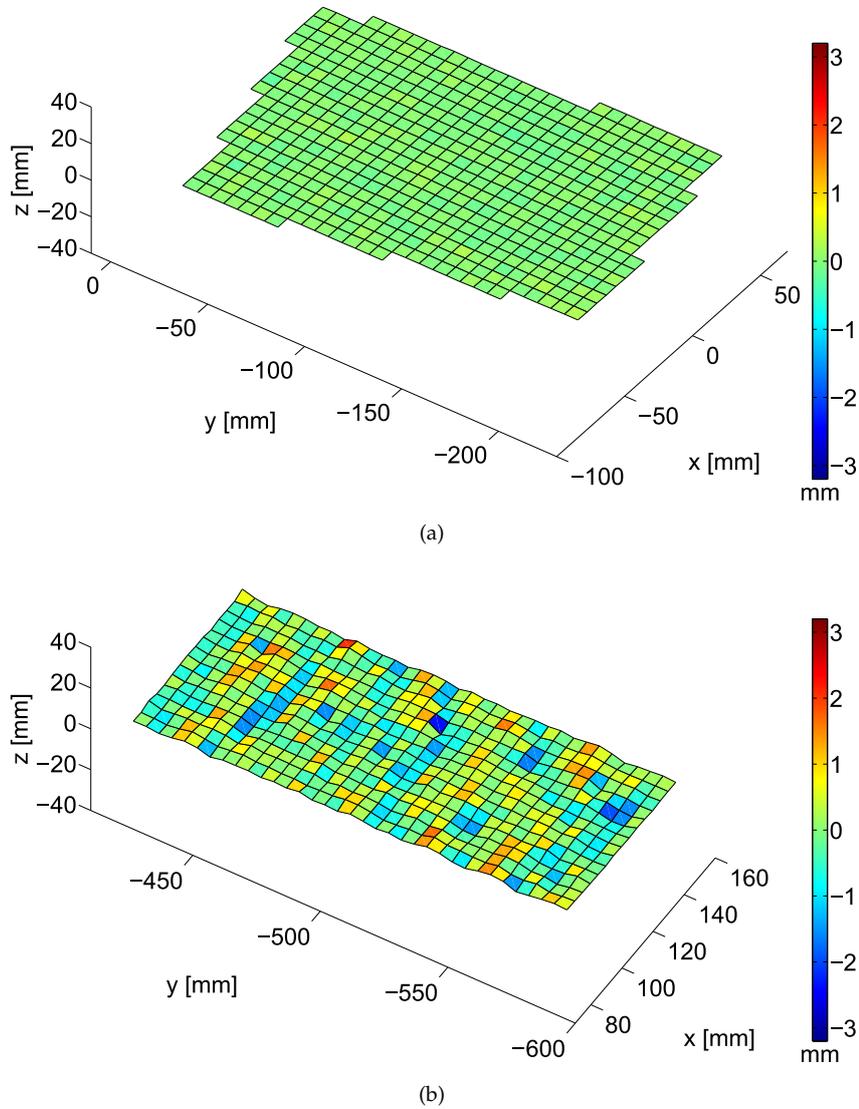


Figure 6.9.: Reconstruction of a plane which was acquired by means of the *calibration LT* method (a) and Microsoft's Kinect camera (b). The triangulation errors are mapped to the color bars. The area that was reconstructed by the Kinect camera is smaller since the device provides a higher reconstruction density. Both surface plots are based on an equivalent amount of approximately 4000 triangulated points.

Device	Method	Triangulation accuracy for a plane [mm]	Triangulation accuracy for the stair phantom [mm]
Developed laserscanner	Calibration LT	0.1037	0.1477
	Calibration NT	0.1035	0.1472
	Interpolation LT	0.1073	0.1590
	Interpolation NT	0.1066	0.1579
Kinect camera	N/A	0.7580	0.8905

Table 6.4.: Comparison of the triangulation accuracy for the developed laser scanning system and Microsoft’s Kinect camera.

6.3. Triangulation accuracy for human foreheads

In case of a rigid object, laser light does not penetrate far into the material. Hence, the reflected light mainly depends on the structure of the surface. In contrast, laser light penetrates far into soft tissue. This is especially true for the utilized infrared laser light. Due to light interaction, a considerable amount of the reflected light comes from the inner layers of the soft tissue. Thus, the reflected light is very diffuse in the case of soft tissue.

This implies that the captured reflection includes more variability. Consequently, the diffuse reflection can influence the detection of the centers of the laser spots. Hence, it can be assumed that the triangulation accuracy is decreased for the reconstruction of the surface of a person’s forehead. To quantify this decrease, the triangulation accuracy of the laser scanning system was determined for the surface reconstruction of three human foreheads. Before the results are analyzed, a measure for the triangulation error is defined.

6.3.1. Definition of an error measure

To define the triangulation accuracy for the surface of a forehead, a ground truth point set can not be used. The reason is that the skin and muscles of the forehead are deformable. As an alternative, a mathematical model of a forehead surface could be fitted to

the acquired surface reconstruction. However, it is problematic to find a general model which is able to incorporate the individual shape of any arbitrary forehead surface. If a mathematical model allows too much variability then it will simply compensate for the triangulation errors.

To avoid this problem, a different approach is utilized in this work. An acquired surface reconstruction is partitioned into small patches which contain a specified number of triangulated points. Afterwards, the procedure described in Sec. 6.2.1 is used to fit a plane to each patch. The triangulation errors for a patch are defined by the perpendicular distance of the associated points with respect to the fitted plane. This measure is also referred to as point-to-plane distance. The triangulation accuracy for a forehead is defined by the RMS of the triangulation errors for all patches (see Eq. 6.3).

As described, a plane is fitted to each patch. The quality of the surface approximation for each patch depends on the number of utilized points. For the experiments in this work, eight triangulated points were used for each patch. This value was determined by means of a qualitative comparison of the fitting results. Smaller values tend to compensate for triangulation errors and higher values lead to a coarse approximation of curved patches. Although a tuned value is used for the number of points, the approximation of curved patches is clearly affected. Consequently, it can be assumed that the resulting triangulation accuracy is slightly worse than the true triangulation accuracy.

6.3.2. Results and discussion

The laser scanning system was configured to project a grid of 32×32 (1024) laser spots. According to Table 3.1, this amount of laser spots is normally projected in approximately 0.1 s. However, for the following experiment, a simultaneous measurement of the optical features described in Sec. 1.2 was carried out. The simultaneous measurement of the optical features was carried out in order to establish a realistic measurement scenario. The reason is that the acquisition of a surface reconstruction and optical features has to be carried out simultaneously in the future head tracking system. As described in Sec. 3.2, the simultaneous measurement of optical features increases the projection time for a grid of laser spots. For the described configuration, the projection time for one grid of laser spots was given by approximately 15 s. The evaluation of the optical features is not carried out in this work.

The system projects one extended code sequence on the surface of a forehead. 10 code frames are required to encode 1024 laser spots and a length of two frames is utilized for each sub-sequence. Thus, the length of the extended code sequence is given by 20 frames. After the captured full frames and code frames of the extended code sequence are processed, the tracked laser spots are identified. Subsequently, the laser scanning system captures one more full frame which is used for the reconstruction of the surface of the forehead. Finally, the set of triangulated points can be used to compute the accuracy of the triangulation.

The triangulation accuracy of the laser scanning system was evaluated for three different subjects. Ellipse fitting achieved the highest triangulation accuracy for the reconstruction of the surface of rigid objects. This is due to the fact that ellipse fitting offers a robust detection of the centers of the captured laser spots. A comparison of the triangulation accuracy revealed that ellipse fitting also achieves the highest triangulation accuracy in the case of soft tissue. For the reason of clarity, the results are not presented for all laser center detection methods. Since the highest triangulation accuracy was achieved by means of ellipse fitting, the following analysis only considers ellipse fitting.

For surface reconstruction, the forehead of each of the three subjects was located at a projection distance of approximately 60 cm. For comparison, the experiments were carried out on the basis of calibrated and interpolated laser rays. Furthermore, linear and non-linear triangulation was utilized for both modalities. At first, the results are discussed for calibrated laser rays and the use of linear triangulation. For each subject, the surface reconstruction and a histogram of the absolute triangulation errors are presented.

Fig. 6.10 shows a front view on the reconstruction of the forehead surface of subject 1. The blue points define the triangulated points. The latter were used to interpolate the copper surface. For illustration purposes, an illumination model and a shading model were used for the interpolated surface. The same view is presented in Figs. 6.11 and 6.12 for the forehead of subject 2 and 3, respectively. Comparing each reconstructed forehead surface with the reconstruction of a planar surface in Fig. 6.9 (a) on a qualitative basis, it can be seen that the reconstructed forehead surfaces are less accurate.

As mentioned, one reason is given by the fact that the detection of the centers of the captured laser spots is less accurate for soft tissue. Another reason is that the head of a subject moves slightly during the acquisition of a surface reconstruction. In this context, the projection time has to be considered. As described in the beginning of this section,

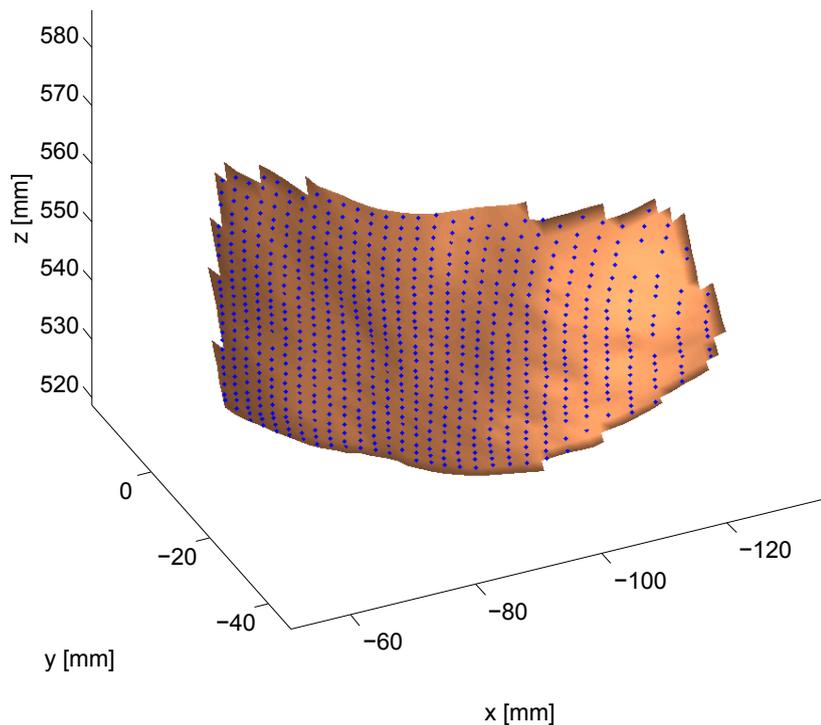


Figure 6.10.: Front view on the reconstructed forehead surface of subject 1. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.

the projection time for one grid of laser spots was given by approximately 15s. The head movement causes the small buckles that are visible in the surface reconstructions presented in Figs. 6.10 to 6.12. To minimize head movement during the acquisition of a surface reconstruction, the head of each subject was placed in a head mold. However, slight head movements are still possible inside the head mold. Another reason for the reduced accuracy is that the soft tissue of a forehead is deformable. This leads to the same artifacts as in the case of head movement.

In Figs. 6.13 to 6.15, a histogram of the absolute triangulation errors is presented for the reconstructed forehead surface of subject 1 to 3, respectively. Each of the three reconstructions contain around 900 triangulated points. As described in the beginning of this section, the laser scanning system was configured to project 1024 laser spots. Hence, a loss of approximately 10 % of the points occurred for each of the three reconstructions. This is contrary to the surface reconstructions of rigid objects where almost always 100 % of the points were reconstructed.

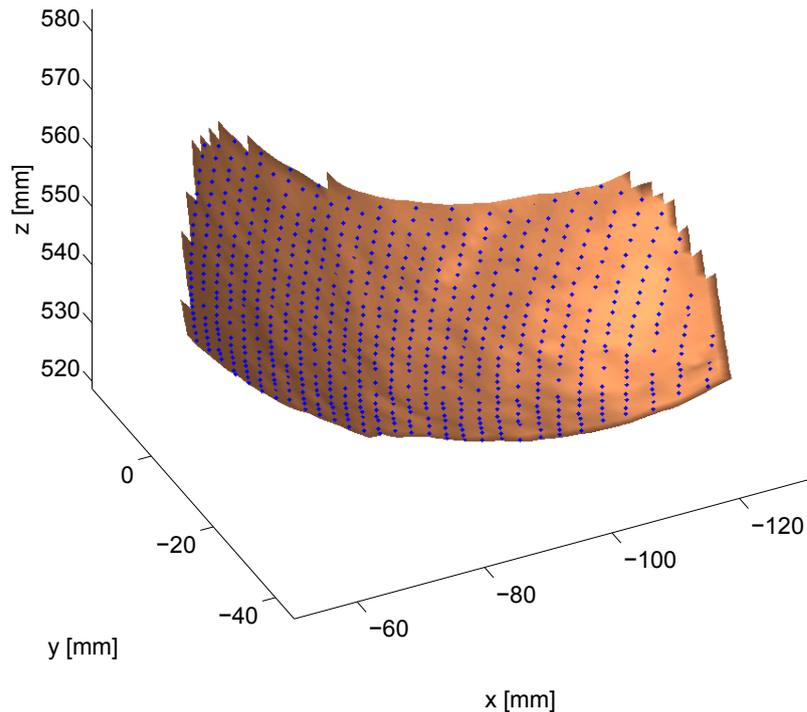


Figure 6.11.: Front view on the reconstructed forehead surface of subject 2. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.

One reason for the loss of points is that a small amount of the laser spots appear very weak inside the camera images. In this case, the ellipse fitting algorithm fails to distinguish the affected laser spots from the background of the camera images. Another reason for the loss of points is that the detection accuracy for the centers of the captured laser spots is reduced in the case of soft tissue. This affects the verification of the laser centers in the scope of epipolar lines. As described in Sec. 5.3, the orthogonal distance of a detected laser center is computed with respect to its corresponding epipolar line. If the orthogonal distance is greater than 1 pixel, the detected laser center is rejected in order to prevent high inaccuracies in the triangulation result.

The histograms in Figs. 6.13 to 6.15 show that the maximum triangulation error is given between 0.55 mm and 0.65 mm. However, the histograms also show that the probability for a triangulation error greater than 0.3 mm is low. For subject 1 to 3 this probability is given by 4.8 %, 5.6 % and 10.2 %, respectively. As explained in Sec. 6.3.1, triangulation accuracy is defined by the RMS of the triangulation errors. The triangulation accuracy for the reconstructed forehead surface of all three subjects is presented in Table 6.5.

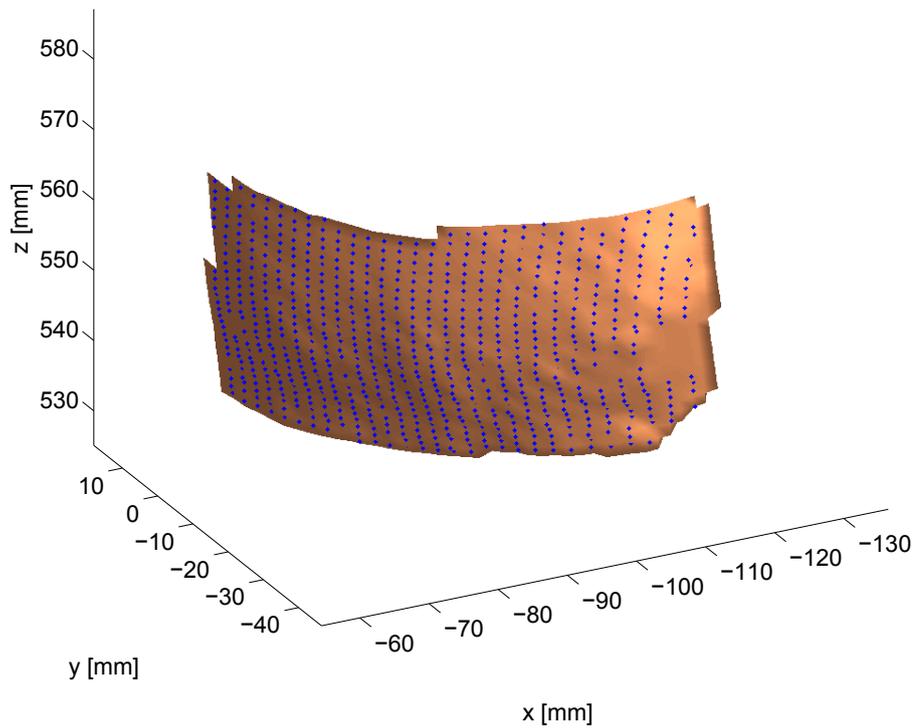


Figure 6.12.: Front view on the reconstructed forehead surface of subject 3. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.

As explained, the results in Figs. 6.10 to 6.15 are based on the use of calibrated laser rays and linear triangulation. This modality is referred to as *calibration LT* and the respective triangulation accuracies are shown in the first row of Table 6.5. The triangulation accuracy for subject 1 and 2 is very similar whereas the triangulation accuracy for subject 3 is given by an increased value of 0.1832 mm. Compared to subject 1 and 2, the increase is given by approximately 0.03 mm. This shows that the error influences can vary indi-

Method	Subject 1	Subject 2	Subject 3
Calibration LT	0.1526	0.1560	0.1832
Calibration NT	0.1515	0.1557	0.1823
Interpolation LT	0.1647	0.1652	0.1891
Interpolation NT	0.1623	0.1634	0.1871

Table 6.5.: Triangulation accuracy in [mm] for the reconstructed forehead surface of all three subjects.

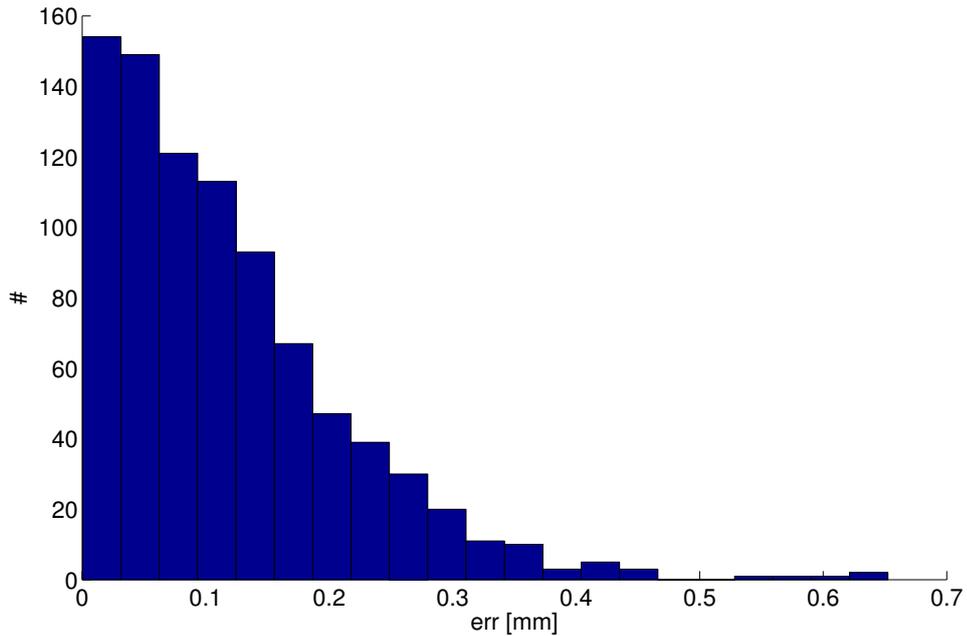


Figure 6.13.: Histogram of the absolute triangulation errors for the reconstructed forehead surface of subject 1.

vidually for each subject. This is not surprising since the interaction between laser light and soft tissue is clearly individual for each subject. The same applies for the described movement artifacts.

The use of calibrated laser rays and non-linear triangulation is referred to as *calibration NT*. As can be seen in Table 6.5, this modality only leads to a marginal accuracy increase of a few tenths of a micrometer for each subject. Table 6.5 also presents the triangulation accuracy based on the use of interpolated laser rays. The combination with linear and non-linear triangulation is referred to as *interpolation LT* and *interpolation NT*, respectively. The mean value of the differences between the accuracy for *calibration LT* and *interpolation LT* is given by 0.0091 mm. Using the accuracy for *calibration LT* as a basis, the differences between both modalities were also calculated in percentage. The mean value is given by 5.68%. Consequently, the use of interpolated laser rays causes only a very small decrease of triangulation accuracy. This implies that the use of interpolated laser rays is an appropriate approach for the reconstruction of a forehead surface.

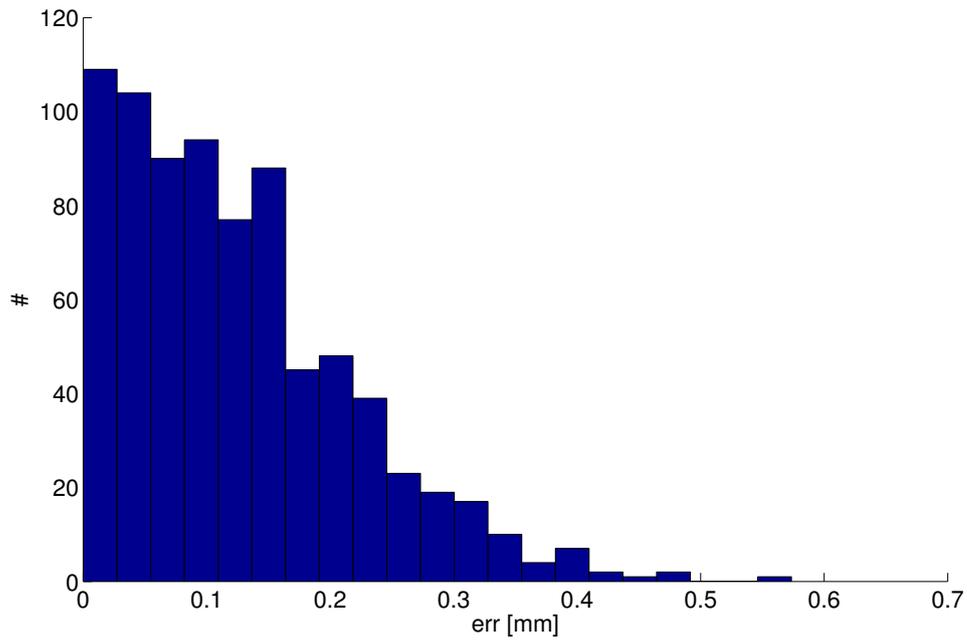


Figure 6.14.: Histogram of the absolute triangulation errors for the reconstructed forehead surface of subject 2.

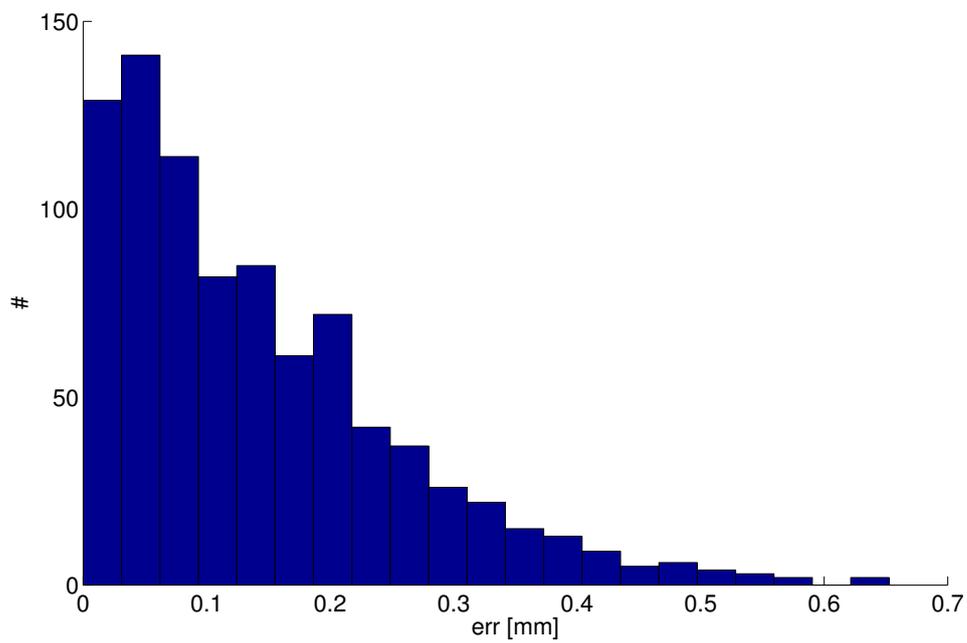


Figure 6.15.: Histogram of the absolute triangulation errors for the reconstructed forehead surface of subject 3.

Comparing the accuracy for *interpolation LT* and *interpolation NT* reveals that the non-linear triangulation achieves an accuracy increase of around two micrometers for each subject. An improvement of two micrometers can already be useful. Thus, the use of the non-linear triangulation can be considered for the case of interpolated laser rays.

By comparing the accuracies shown in Tables 6.5 and 6.4, it can be seen that the accuracy for all three subjects is only slightly reduced with respect to the accuracy for the stair phantom. For the modality *calibration LT*, the difference is given by 0.0049 mm, 0.0083 mm and 0.0355 mm for subject 1 to 3, respectively. For the modality *interpolation LT*, the difference is given by 0.0057 mm, 0.0062 mm and 0.0301 mm for subject 1 to 3, respectively.

As described in the beginning of this section, the projection time for one grid of laser spots was given by approximately 15 s. For the purpose of comparison, the reconstruction of a forehead surface was also carried out without the simultaneous measurement of optical features. In this manner, the projection of a grid of 1024 laser spots can be carried out in approximately 0.1 s (see Table 3.1). Fig. 6.16 shows a front view on the reconstruction of the forehead surface of subject 2. Comparing Figs. 6.11 and 6.16 on a qualitative basis, it can be clearly seen that the reconstructed forehead surface in Fig. 6.16 is more accurate. Since the projection of a grid of laser spots is now carried out in approximately 0.1 s, the artifacts that are related to head movement and the deformation of soft tissue do not exist in Fig. 6.16. The triangulation accuracy for the reconstructed forehead surface of subject 2 is presented in Table 6.6. Comparing Table 6.6 to the second column of Table 6.5 reveals that the triangulation accuracy for the forehead surface in Fig. 6.16 is increased. For the modality *calibration LT*, the difference is given by 0.0236 mm.

Method	Subject 2
Calibration LT	0.1324
Calibration NT	0.1319
Interpolation LT	0.1432
Interpolation NT	0.1417

Table 6.6.: Triangulation accuracy in [mm] for the reconstructed forehead surface of subject 2 shown in Fig. 6.16.

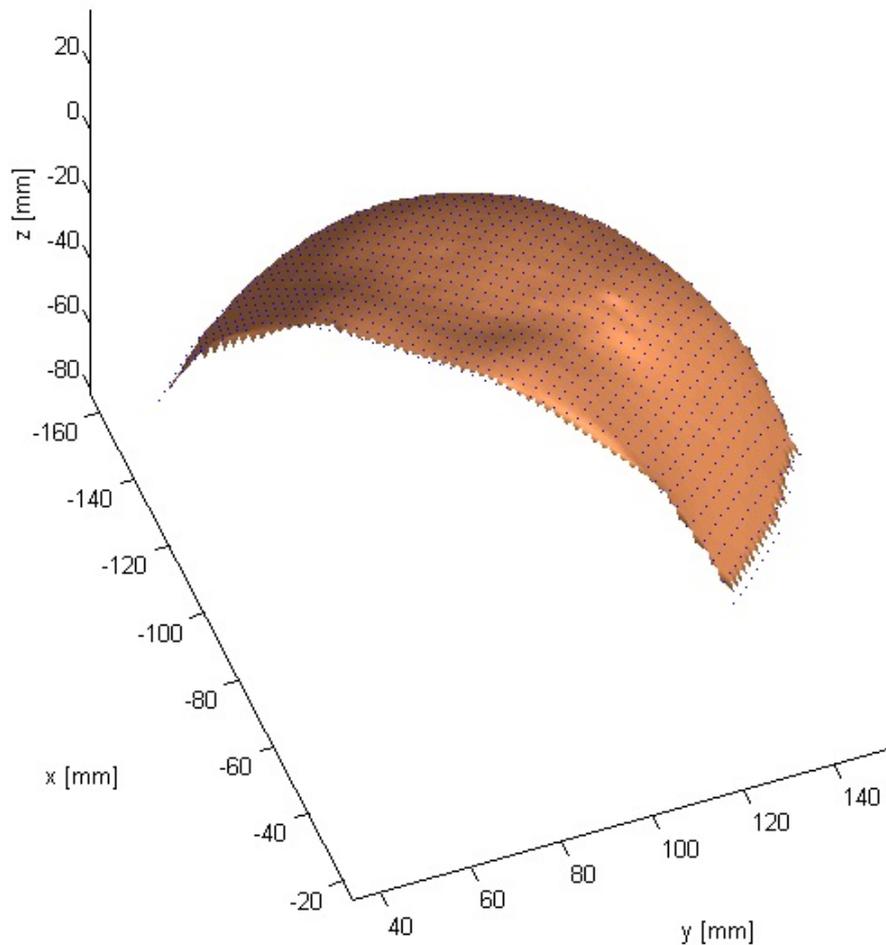


Figure 6.16.: Front view on the reconstructed forehead surface of subject 2. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.

6.4. Tracking accuracy

In the previous sections, the triangulation accuracy was determined for rigid phantoms and human foreheads. In future works, the developed laser scanning system shall be utilized for the tracking of human heads. Consequently, the triangulation accuracy will influence the accuracy of tracking. As explained in Sec. 1, the tracking of a person's head shall be realized by combining two datasets. The first dataset is given by a surface reconstruction of the forehead and the second dataset includes the optical features which correlate to the thickness of the soft tissue. The correlation properties of the optical fea-

tures shall be used to compensate for the soft tissue. The final goal is to offer a highly accurate registration with the rigid skull surface.

As described, the final tracking system is part of future developments. Consequently, the influence of the triangulation accuracy on the final tracking accuracy can not be analyzed in this work. However, the developed laser scanning system was utilized to carry out tracking experiments which are solely based on surface reconstructions. The resulting tracking accuracy can be interpreted as a tendency for the tracking accuracy which has to be analyzed in the scope of the future tracking system.

6.4.1. Definition of tracking accuracy

The tracking accuracy was determined for two rigid objects and a human head. The rigid objects are given by the stair phantom in Fig. 6.1 and the polystyrene head phantom shown in Fig. 6.17. For the purpose of tracking, the developed laser scanning system was mounted to the end-effector of an *Adept Viper s850* robot. In this manner, the movement of the robot can be utilized as a ground truth for the tracking.



Figure 6.17.: Polystyrene head phantom.

The tracking accuracy for an object is determined by the following procedure. At first, a reconstruction of the surface of the object is acquired. This reconstruction is referred to as the target point set. During the acquisition of the target point set, the end-effector is located in the initial position. After the acquisition of the target point set, translational

displacements are applied to the robot end-effector. The space which includes all applied end-effector displacements is described by a sphere and the initial position of the end-effector defines the center of this sphere. Regarding the projection of extended code sequences, one grid of laser spots is projected after each displacement. Due to the acquisition of the target point set, the tracked laser spots are already identified. Hence, a new reconstruction of the surface of the object is obtained for each displacement that incorporates a captured full frame.

The tracking itself is realized by means of the ICP algorithm which registers the target point set to each newly obtained point set. The utilized ICP variant uses the point-to-plane distance measure as explained in Sec. A.6.2. The result of each registration is given by an estimate of the pose cT_g of the coordinate system of the target point set with respect to the camera coordinate system (see Eq. A.18). The pose cT_g consists of the translation vector ${}^c\mathbf{t}_g$ and the rotation matrix cR_g .

Each registration result has to be compared to a ground truth. The ideal ground truth for tracking would be given by the initial pose of the camera with respect to each shifted camera. This is due to the fact that each reconstructed point set is given with respect to the camera coordinate system. However, this approach would necessitate an eye-in-hand calibration in order to compute the constant pose of the camera with respect to the end-effector. The problem is that an eye-in-hand calibration introduces high inaccuracies. To avoid this problem, the ground truth for tracking is defined by solely using the applied translational end-effector displacements. In this manner, the ground truth solely depends on the high positioning accuracy of the robot end-effector.

In the following, a translational end-effector displacement is referred to as ${}^e\mathbf{t}_{e^*}$. The latter describes the translation of the initial end-effector coordinate system $(CS)_{e^*}$ with respect to a shifted end-effector coordinate system $(CS)_e$. Using the ground truth translation ${}^e\mathbf{t}_{e^*}$ and the registered translation ${}^c\mathbf{t}_g$, the translational tracking error can be defined as:

$$d = \|{}^e\mathbf{t}_{e^*}\|_2 - \|{}^c\mathbf{t}_g\|_2. \quad (6.6)$$

Since both translation vectors are defined in a different manner, the translational tracking error can not be defined by means of their difference vector. Instead, the L2-norm of both translation vectors has to be compared in Eq. 6.6. As described, the displacements of the end-effector are given by a pure translation. Consequently, the rotational tracking error

solely depends on the rotation matrix ${}^c\mathbf{R}_g$ which results from the registration process. In this work, the rotational tracking error is defined by the angle θ . The latter is calculated by converting ${}^c\mathbf{R}_g$ to the angle-axis representation [75]. Finally, the tracking accuracy is defined by the RMS of the translational tracking errors and the RMS of the rotational tracking errors.

6.4.2. Results and discussion

The laser scanning system was configured to project a grid of 72×56 (4032) laser spots. According to Table 3.1, this amount of laser spots is projected in approximately 0.4 s. In order to encode 4032 laser spots, 12 code frames are required. The length of the subsequences depends on the individual tracking experiment. In the scope of the tracking experiments, calibrated laser rays and the linear triangulation are utilized for the generation of surface reconstructions.

In the first experiment, the tracking was carried out for the polystyrene head phantom shown in Fig. 6.17. The surface reconstructions of the whole face were utilized for the tracking. In this manner, a high amount of distinctive information is incorporated. As explained in Sec. 6.4.1, a target point set is acquired when the robot end-effector is located in the initial position. The target point set for the face of the polystyrene head phantom is presented in Fig. 6.18.

The translational displacements of the robot end-effector are carried out consecutively by using a constant robot step size g . Figure 6.19 shows an exemplary surface reconstruction of the phantom's face which was obtained after a translational end-effector displacement was carried out. Figure 6.20 exemplarily presents the result of the registration of both point sets.

As explained, the tracking of the polystyrene head phantom was carried out by means of the whole face of the phantom. The tracking was carried out for three configurations. Each configuration uses two parameters. The first parameter is given by the length j of each sub-sequence included in a projected extended code sequence. The second parameter defines the robot step size g . Both parameters determine the pixel shift between two full frames. As explained in Sec. 3.2, the tracked pixel position of a laser spot is updated for every full frame. If the pixel shift between two full frames is too far then the tracking of a laser spot can fail. In the worst case, a wrong laser spot is tracked. This is

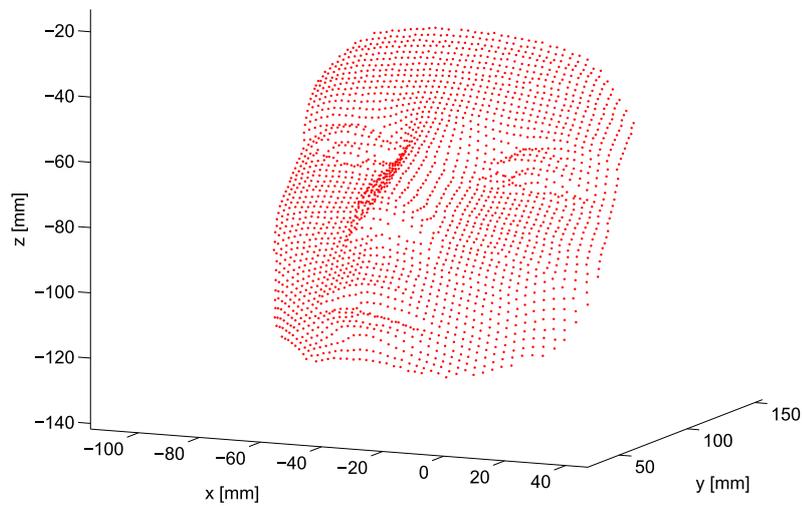


Figure 6.18.: Target point set of the polystyrene head phantom.

due to the fact that the tracking of laser spots is realized by a radius based search. An incorrect tracking of the captured laser spots can lead to code detection errors. The latter can lead to triangulation errors which ultimately influence the accuracy of the tracking of the head phantom.

To avoid this problem, a small value j can be utilized. As explained in Sec. 3.2, a small value j also leads to the advantage that a high reconstruction rate is achieved. On the other hand, a small value j also leads to an increased initialization phase. Besides j , the pixel shift between two full frames also depends on the robot step size g . A large value g clearly causes a large pixel shift between two full frames.

The tracking accuracy for the three configurations is presented in Table 6.7. For each of the three configurations, 264 tracking results were acquired. The tracking accuracy is given by the RMS of the translational tracking errors and the RMS of the rotational tracking errors. The highest tracking accuracy was achieved for the configuration which uses a length of $j = 2$ frames for each sub-sequence and a robot step size of $g = 0.2$ mm. In this case, the described pixel shift is very small and the correct tracking of the captured laser spots is guaranteed. This is not the case in the following configurations. Increasing either the length j or the robot step size g leads to a decreasing tracking accuracy. Increasing only j from 2 to 3 already leads to a RMS of the translational tracking errors that is loc-

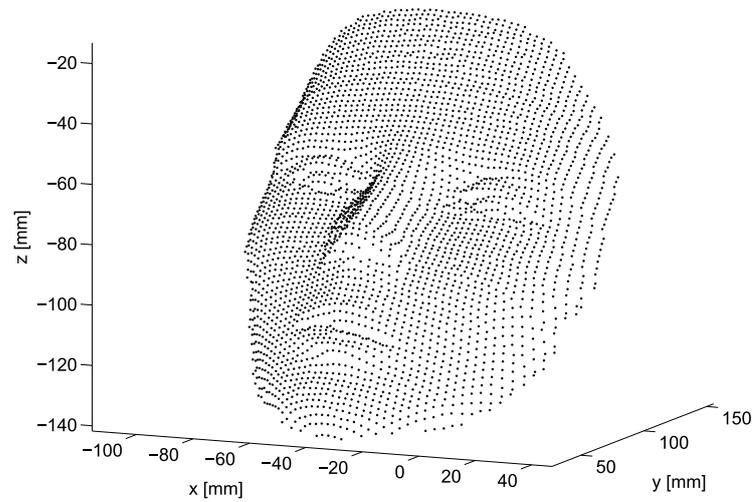


Figure 6.19.: Exemplary surface reconstruction of the face of the polystyrene head phantom. The reconstruction was obtained after a translational displacement was applied to the robot end-effector.

ated outside of the submillimeter range. Increasing only g from 0.2 mm to 0.5 mm leads to a tracking accuracy that is around four times worse than the tracking accuracy of the first configuration.

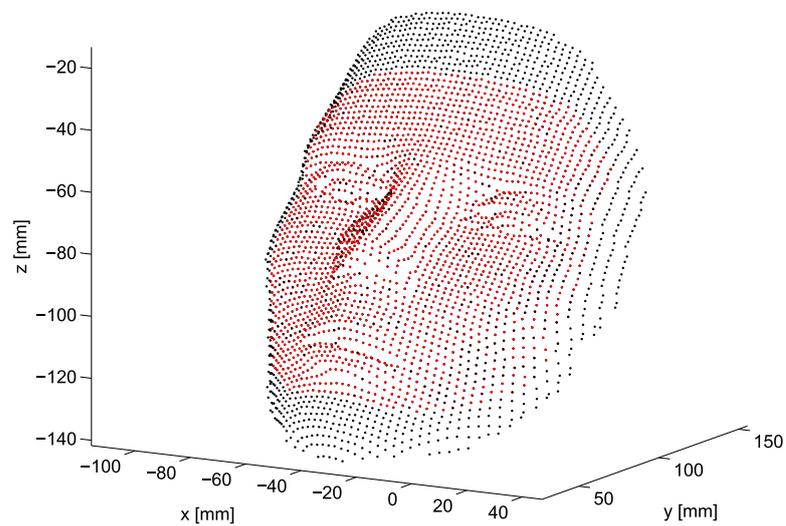


Figure 6.20.: Registration result for the target point set and the surface reconstruction shown in Figs. 6.18 and 6.19, respectively.

6. Accuracy analysis

Length j of each sub-sequence	Robot step size g [mm]	RMS translational tracking errors [mm]	RMS rotational tracking errors [degree]
2	0.2	0.75	0.18
3	0.2	1.01	0.26
2	0.5	2.93	1.37

Table 6.7.: Tracking accuracy for the polystyrene head phantom. The tracking was carried out by using surface reconstructions of the whole face of the phantom.

Nevertheless, the tracking accuracy for the first configuration shows that a high tracking accuracy can be achieved if small values are chosen for the parameters j and g . For this reason, a length of $j = 2$ frames for each sub-sequence and a robot step size of $g = 0.2$ mm are utilized for all following tracking experiments. For the purpose of comparison, the described tracking experiment was also carried out for the stair phantom shown in Fig. 6.1. In contrast to the polystyrene head phantom, the stair phantom incorporates an even surface. This leads to the circumstance that the detection of the centers of the captured laser spots is carried out with higher accuracy. The tracking accuracy for the stair phantom is presented in the first row of Table 6.8. By comparing these results with the first row in Table 6.7, it can be seen that a much higher tracking accuracy can be achieved by means of the stair phantom. In fact, the tracking accuracy is around twice as good.

Object	Length j of each sub-sequence	Robot step size g [mm]	RMS translational tracking errors [mm]	RMS rotational tracking errors [degree]
Stair phantom	2	0.2	0.33	0.12
Forehead of the polystyrene head phantom	2	0.2	1.56	0.41
Forehead of a human head	2	0.2	7.43	1.11

Table 6.8.: Tracking accuracy for two rigid objects and the forehead of a human head.

As described in the beginning of this section, the future head tracking system will incorporate a surface reconstruction of the forehead of a person. For this reason, the described tracking experiment was also carried out for the forehead of the polystyrene head phantom. The resulting tracking accuracy is presented in the second row of Table 6.8. By comparing this result to the first row in Table 6.7, it can be seen that using only the forehead of the head phantom leads to a tracking accuracy which is around two times worse. This is due to the fact that a surface reconstruction of a forehead contains less distinctive information than a surface reconstruction of a whole face. Consequently, the use of more information could be considered for the future development of the tracking of person's head. Especially the use of the surfaces around the pointed cheekbones and the nasal bone can improve the tracking accuracy considerably. These areas belong to the most distinctive parts in a facial surface reconstruction.

The last tracking experiment was carried out for a human head. To achieve the same conditions for the tracking setup as in the case of the head phantom, the head of the test person was fixed with a stereotactic frame. In this manner, it was guaranteed that the head of the subject was not able to move. As in the case of the head phantom, the only allowed movement is carried out by the robot end-effector. A side effect of the stereotactic frame is that the frame occluded most of the parts of the lower facial structure. For this reason, the head tracking could only be carried out by using the surface of the forehead.

The accuracy for the tracking of a human head is presented in the third row of Table 6.8. The high inaccuracy is caused by two reasons. First of all, only surface reconstructions of the forehead are utilized for the tracking. Consequently, a reduced amount of distinctive information is incorporated. Furthermore, the soft tissue of a forehead is deformable and structural changes are caused by facial expressions as well as variations in temperature. Consequently, it is not surprising that the achieved tracking accuracy is much reduced compared to the tracking accuracy for rigid objects.

This result confirms that the tracking of a person's head can not be carried out by the sole use of a reconstruction of the deformable skin surface. In order to achieve a high tracking accuracy, it is necessary to use the reconstruction of a rigid surface. As described in Sec. 1, this is the goal for future developments. The goal is to combine the surface reconstruction of a forehead with optical features which correlate to the thickness of the soft tissue. In this manner, the influence of the deformable soft tissue shall be compensated in order to offer a highly accurate registration with respect to the surface of the rigid skull.

7. Simulation

As explained in Sec. 6.2.2, the stair phantom incorporates a ground truth point set that is affected by the accuracy of the CT scanner. Hence, it can be assumed that the triangulation accuracy for the stair phantom is slightly worse than the true triangulation accuracy. In contrast, the use of a planar surface considers triangulation errors only in one direction. Thus, it can be assumed that the triangulation accuracy for a planar surface is slightly better than the true triangulation accuracy. As a consequence, it can be assumed that the true triangulation accuracy of the laser scanning system is given by a value between the result of both methods.

The correctness of the computed triangulation accuracy depends on the correctness of the ground truth point set. To generate an ideal ground truth point set, this section covers a simulation of the developed laser scanning system. In order to analyze the triangulation accuracy, the sources of error described in Sec. 6.1 are included in the simulated laser scanning system. Finally, the triangulation accuracy of the simulated laser scanning system is utilized to verify the triangulation accuracy for rigid objects reported in Sec. 6.2.3.

7.1. A simulation model

The model in [48] is utilized to simulate a laser ray which is deviated by two controllable mirrors. A graphical representation of this model is shown in Fig. 7.1. The origin of the projector coordinate system $(CS)_f$ is located on the rotation axis of the first mirror. The x-axis of $(CS)_f$ is parallel to the rotation axis of the second mirror and the y-axis of $(CS)_f$ is perpendicular to it. The laser ray which originates from the laser diode is defined by its (3×1) origin s and the (3×1) direction l . In the following, this laser ray is referred to as the incoming laser ray.

The point of intersection of the incoming laser ray and the first mirror is given by:

$$\mathbf{b} = \mathbf{s} + \frac{\langle \mathbf{n}, -\mathbf{s} \rangle}{\langle \mathbf{n}, \mathbf{l} \rangle} \mathbf{l}. \quad (7.3)$$

The notation $\langle \cdot, \cdot \rangle$ denotes the dot product. A laser ray which is reflected by the first mirror is defined by its origin \mathbf{b} and the direction \mathbf{d} . The latter is expressed by:

$$\mathbf{d} = 2 \langle \mathbf{n}, -\mathbf{l} \rangle \mathbf{n} + \mathbf{l}. \quad (7.4)$$

The position of the second mirror is described by means of $\mathbf{q} = [0 \ r \ 0]^T$. The quantity r is the distance between the origin \mathbf{a} and the position \mathbf{q} of the second mirror. The orientation of the second mirror is defined by the normal vector \mathbf{k} which depends on the angle ψ . The latter defines the controllable rotation of the second mirror around the x-axis of $(CS)_f$. Hence, the normal vector of the second mirror is given by $\mathbf{k} = [0 \ \cos(\psi) \ \sin(\psi)]^T$. The point of intersection of a reflected laser ray and the second mirror is defined by:

$$\mathbf{h} = \mathbf{b} + \frac{\langle \mathbf{k}, \mathbf{q} - \mathbf{b} \rangle}{\langle \mathbf{k}, \mathbf{d} \rangle} \mathbf{d}. \quad (7.5)$$

A laser ray which is reflected by the second mirror is described by its origin \mathbf{h} and the direction \mathbf{m} . The latter is computed by:

$$\mathbf{m} = -2 \langle \mathbf{k}, \mathbf{d} \rangle \mathbf{k} + \mathbf{d}. \quad (7.6)$$

A laser ray that is reflected by the second mirror is referred to as an outgoing laser ray. For the simulation, two sets of laser rays have to be considered. The first set simulates the calibrated laser rays and the second set simulates the erroneous laser rays during surface reconstruction. As described in Sec. 6.1, the positioning of the mirrors is error-prone in the physical laser scanning system. Thus, during the reconstruction of a surface, the laser rays slightly deviate from the calibrated laser rays. The simulation of this deviation is achieved by introducing a positioning error for both controllable mirrors.

The actual positioning accuracy of the galvanometer driven mirrors in the physical laser scanning system is unknown. However, comparable products (scanlab.de) achieve a positioning accuracy of approximately $1 \mu\text{rad}$ (RMS). For this reason, a standard deviation of $1 \mu\text{rad}$ is utilized to introduce a realistic Gaussian distributed positioning error. In order to obtain independent positioning errors for both mirrors, two equal Gaussian distributions are utilized. A positioning error for the first mirror is added to the angle ϕ and a positioning error for the second mirror is added to the angle ψ .

For the simulation, a planar surface is placed at a specified distance with respect to the projector coordinate system $(CS)_f$. The goal is to project a set of laser points onto the planar surface. The position of the planar surface is described by its origin $\mathbf{u} = [0 \ 0 \ j]^T$. The orientation of the planar surface is described by the constant normal vector $\mathbf{v} = [0 \ 0 \ 1]^T$ which also represents the z-axis of $(CS)_f$. In order to project a laser point onto the planar surface, the intersection point \mathbf{p}^f between the planar surface and an outgoing laser ray is computed. The calculation of a point \mathbf{p}^f is carried out as follows:

$$\mathbf{p}^f = \mathbf{h} + \frac{\langle \mathbf{u}, \mathbf{v} \rangle - \langle \mathbf{h}, \mathbf{v} \rangle}{\langle \mathbf{m}, \mathbf{v} \rangle} \mathbf{m}. \quad (7.7)$$

The described projection utilizes the laser rays which depend on the explained positioning error. The result is given by a set of laser points which are projected onto the planar surface. This set is described by \mathbf{p}_i^f with $i \in [1, I]$ where I defines the amount of points.

Next, the camera coordinate system $(CS)_c$ is placed with respect to the projector coordinate system $(CS)_f$. The translation vector of $(CS)_c$ is given by ${}^f\mathbf{t}_c = [g \ 0 \ 0]^T$. The rotation matrix of $(CS)_c$ is given by ${}^f\mathbf{R}_c$ and is computed by means of XYZ Euler angles as described in Eq. A.4. The translation vector ${}^f\mathbf{t}_c$ and the rotation matrix ${}^f\mathbf{R}_c$ are utilized to construct the homogeneous (4×4) transformation matrix ${}^f\mathbf{T}_c$ (see Eq. A.5).

As described, a projected laser point \mathbf{p}^f on the planar surface is given with respect to the projector coordinate system $(CS)_f$. Using the inverse transformation matrix ${}^c\mathbf{T}_f$, a projected laser point \mathbf{p}^f can be transformed to a projected laser point \mathbf{p}^c with respect to the camera coordinate system $(CS)_c$. Thus, the set of all projected laser points with respect to the camera is defined by \mathbf{p}_i^c . The obtained set \mathbf{p}_i^c defines the ground truth point set for the triangulation of spatial points.

A deviation between a triangulated point set and the ground truth point set is caused by the following error sources. For triangulation, the calibrated laser rays are utilized. Furthermore, a pixel error is added to the laser points which are projected into the sensor coordinate system of the placed camera. The projection of a point \mathbf{p}^c into the sensor coordinate system is defined by Eq. 4.7. By applying Eq. 4.7 to all points in the set \mathbf{p}_i^c , a set of normalized image points \mathbf{p}_i^n is obtained.

Subsequently, a pixel error is added to each point contained in the set \mathbf{p}_i^n . As explained in Sec. 5.1, the pixel error in the physical camera follows a Gaussian distribution and the maximum L2-norm of the pixel error is given by approximately 0.5 pixel. The same error distribution is applied for the pixel error of the set \mathbf{p}_i^n in the simulation. However, the set \mathbf{p}_i^n describes image points which are located in the sensor coordinate system. Thus, the unit of these image points is not given by pixels. In order to apply the described pixel error, the calibrated intrinsic parameters of the physical camera are utilized to transform the pixel error to the sensor coordinate system. The applied pixel error simulates the deviation that occurs during the detection of the centers of the captured laser spots.

Finally, the set \mathbf{p}_i^n of image points and the calibrated laser rays are utilized as the input parameters of the triangulation procedure described in Sec. 5.3. For the simulation experiments, the linear triangulation method is utilized. The result is given by the set $\mathbf{p}_i^{c'}$ of triangulated points with respect to the camera coordinate system. Here, the index c' is utilized to prevent confusion with the ground truth point set \mathbf{p}_i^c . The triangulation error e for one point $\mathbf{p}^{c'}$ is defined by the L2-norm of the difference vector between $\mathbf{p}^{c'}$ and the corresponding point \mathbf{p}^c in the ground truth point set:

$$e = \left\| \mathbf{p}^{c'} - \mathbf{p}^c \right\|_2. \quad (7.8)$$

The triangulation accuracy is defined as the RMS of all triangulation errors e_i and is computed by means of Eq. 6.3.

7.2. Results and discussion

In the first experiment, the simulation model of Sec. 7.1 is utilized to verify the triangulation accuracy for rigid objects reported in Sec. 6.2.3. For this purpose, the parameters of

the simulated laser scanning system are determined by a manual estimation of the corresponding parameters in the physical laser scanning system. The intrinsic parameters of the galvanometric laser projection system are given by the angle ω of the first mirror, the distance r between the two mirrors as well as the origin s and the direction l of the incoming laser ray. The utilized extrinsic parameters are given by the translation g and the XYZ Euler angles of the camera with respect to the projector coordinate system.

As in Sec. 6.2.3, the simulated laser scanning system projects a grid of 72×56 (4032) laser points. For the purpose of comparison, the planar surface is placed at a projection distance $j = 59.5$ cm. This is the projection distance which is associated with the highest triangulation accuracy for the physical laser scanning system in Figs. 6.7 and 6.8. A histogram of the triangulation errors for the simulated laser scanning system is presented in Fig. 7.2. The triangulation accuracy is defined as the RMS of all triangulation errors and is given by 0.1184 mm. In Table 7.1, the resulting triangulation accuracy is compared to the triangulation accuracies of the physical laser scanning system. The results for the physical system are taken from the first row of Table 6.4 and represent the triangulation accuracy for the stair phantom and a planar surface.

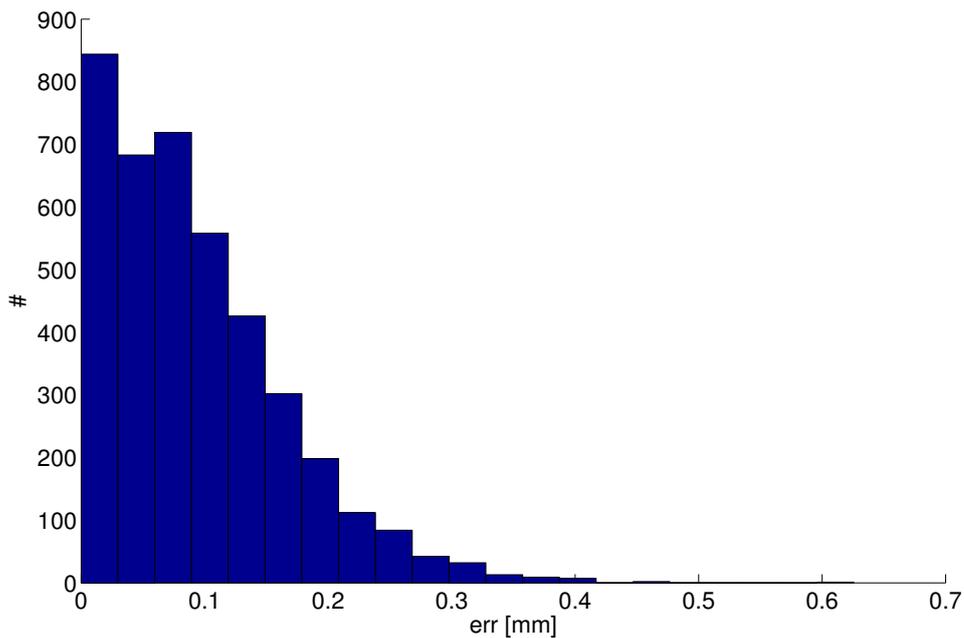


Figure 7.2.: Histogram of the triangulation errors for the reconstructed planar surface.

As can be seen in Table 7.1, the triangulation accuracy of the simulated system is located between the two triangulation accuracies of the physical system. This corresponds to the

Device	Reconstructed surface	Triangulation accuracy [mm]
Physical laserscanner	Plane	0.1037
	Stair phantom	0.1477
Simulated laserscanner	Plane	0.1184

Table 7.1.: Comparison of the triangulation accuracy for the physical and the simulated laser scanning system.

initial assumption that the true triangulation accuracy of the physical system is given by a value which is located between the triangulation accuracy for the stair phantom and a planar surface. The difference between the triangulation accuracy of the simulated system and the triangulation accuracy for a planar surface in the physical system is given by 0.0147 mm. The difference between the triangulation accuracy of the simulated system and the triangulation accuracy for the stair phantom in the physical system is given by 0.0293 mm.

In the second experiment, the simulation model is used to analyze the influence of the distance between the camera and the galvanometric laser projection system. In the first experiment, the translation of the camera with respect to the projector coordinate system $(CS)_f$ was given by $g = 40$ cm. For comparison, the triangulation accuracy is computed by using a translation g of 20 cm, 30 cm, 50 cm and 60 cm. All other parameters of the simulation model are the same as in the first experiment.

Camera translation g [cm]	Triangulation accuracy for a plane [mm]
20	0.1341
30	0.1269
40	0.1184
50	0.1102
60	0.1033

Table 7.2.: Comparison of the triangulation accuracy for different camera translations in the simulated laser scanning system.

The triangulation accuracy for all camera translations is compared in Table 7.2. As ex-

7. Simulation

plained in Sec. 6.1, the triangulation accuracy increases when the distance between the camera and the galvanometric laser projection system increases as well. This behavior is also valid for the triangulation accuracies in Table 7.2. The relationship between the camera translation and the triangulation accuracy can be approximated by a linear function.

8. Conclusion and future works

Optical head tracking for robot-guided radiosurgery is a very promising approach. It is based on the use of harmless laser light and it offers high scan rates. However, optical head tracking is inaccurate since the registration process is influenced by deformable structures which are given by the skin and the muscles. In order to overcome the problems of optical head tracking, our research group developed a novel measurement system. The system projects a grid of discrete infrared laser spots onto the forehead of a person. The captured reflection of the grid is utilized to acquire a point-based reconstruction of the forehead's surface. Furthermore, the captured reflection of the grid is also used to calculate a set of optical features for each captured laser spot. The novelty of this approach is given by the correlation of each feature set with the thickness of the soft tissue. The goal for future developments is to combine the point-based surface reconstruction with the optical features. In this manner, the measurement system is intended to offer highly accurate registration with respect to the rigid skull surface. Consequently, tracking errors that are related to the deformable skin and muscles can be avoided.

The point-based reconstruction of the surface of a forehead is acquired by means of triangulation. The latter requires the mapping between the projected and the captured light information. This is also known as the correspondence problem. The reflection of a grid of laser spots is captured in one single camera image. Due to partial pattern occlusion and undetected laser spots, the correspondence problem can not be solved directly. For this reason, binary time multiplexing is utilized for the encoding of the projected grid. The advantage of time multiplexed structured light is that each laser spot is encoded independently. In this manner, code detection errors can be minimized and a highly accurate reconstruction of spatial surface points can be achieved.

The proposed method for binary time multiplexing reduces the general restrictions of previous time multiplexing methods. The developed method can be used for objects that move slowly. Moreover, the method makes an improvement to offer a high reconstruction

tion rate. Hence, the proposed method is appropriate for surface reconstruction in the context of head tracking for radiation therapy. The only disadvantage of the developed time multiplexing method is the fact that an initialization phase is required. After the initialization phase has passed, the acquisition of surface reconstructions can begin. The encoding of 1000 laser spots results in a reasonably short initialization phase of 2 s. However, if the simultaneous computation of optical features is required than the integration time of the camera has to be increased. In this case, the projection of a single grid can take up to several seconds and the initialization phase can take up to more than one minute.

The triangulation of surface points also requires the calibration of the extrinsic parameters of the laser scanning system. To achieve a high calibration accuracy, this work proposes an explicit laser ray calibration. The latter incorporates the calibration of each single laser ray that is used for the projection of a grid of laser spots. Furthermore, a method for the interpolation of laser rays is proposed. Using a set of laser rays that was calibrated in advance, new laser rays can be computed by means of interpolation. In this manner, a time consuming acquisition of new calibration data can be avoided.

In order to analyze the triangulation accuracy of the developed laser scanning system, the surface of a plane and the surface of a stair phantom were reconstructed. The triangulation accuracy is given by a RMS error of 0.1037 mm and 0.1477 mm, respectively. These results were also verified by means of a simulation of the laser scanning system. The revealed triangulation accuracy is given by a RMS error of 0.1184 mm. In order to compare the triangulation accuracy with a commercially available scanning system, the surface of the plane and the surface of the stair phantom were also reconstructed by means of Microsoft's Kinect camera. The results show that the developed laser scanning system outperforms the Kinect camera by a factor of six.

Furthermore, the surface of the forehead of the three different subjects was reconstructed. The resulting triangulation accuracy is given between RMS errors of 0.1526 mm and 0.1832 mm. Due to the interaction of the infrared laser light with soft tissue, the detection of the centers of the captured laser spots is less accurate than in the case of the rigid phantoms. Moreover, the head of a subject moves slightly during the acquisition of a surface reconstruction. Both circumstances lead to a triangulation accuracy that is slightly decreased compared to the triangulation accuracy for the rigid phantoms.

The developed laser scanning system was also utilized to carry out tracking experiments. For this purpose, the laser scanning system was attached to the end-effector of a robot

manipulator. The goal was to analyze how the remaining triangulation errors influence the accuracy of tracking. The highest accuracy was achieved for the tracking of the stair phantom. The RMS of the translational tracking errors and the RMS of the rotational tracking errors is given by 0.33 mm and 0.12 °, respectively. The tracking of a polystyrene head phantom revealed a slightly decreased accuracy. The RMS of the translational tracking errors and the RMS of the rotational tracking errors is given by 0.75 mm and 0.18 °, respectively.

The tracking of the forehead of a human head revealed a highly inaccurate result. The RMS of the translational tracking errors and the RMS of the rotational tracking errors is given by 7.43 mm and 1.11 °, respectively. This result clearly shows that the sole use of a surface reconstruction of the forehead is not appropriate for tracking. The result supports the initial idea to combine the surface reconstruction of a forehead with the optical features which correlate with the thickness of the soft tissue. In this manner, the influence of the deformable soft tissue shall be compensated.

The developed measurement system obtains a point-based reconstruction of the surface of the forehead of a person. Furthermore, the measurement system acquires optical features which correlate to the thickness of the soft tissue. For both measurements, an infrared laser source with a wavelength of 850 nm is utilized. The interaction of the infrared laser light with the soft tissue leads to the fact that the detection of the centers of the captured laser spots is less accurate than in the case of a rigid object. Thus, the triangulation accuracy is slightly decreased for the reconstruction of the surface of a person's forehead.

To avoid this problem in future developments, the triangulation part of the measurement system will use a blue laser source with a wavelength of 405 nm. This wavelength guarantees that the light absorption of the chromophores in the human skin is much higher than compared to infrared laser light at 850 nm [4]. A high light absorption by the chromophores reduces the light penetration depth and the effects of light scattering in the skin. As a consequence, the skin specific variation of the captured laser light can be reduced by means of a blue laser source. In this manner, the accuracy of the detection of the centers of the captured laser spots shall be increased for soft tissue.

Another open point for further improvements is given by the initialization phase of the proposed time multiplexing method. In future works, alternative strategies for the encoding of the projected laser spots can be evaluated. An alternative encoding strategy should either decrease the duration of the initialization phase or it should completely re-

8. Conclusion and future works

move the necessity of an initialization phase. A first idea to decrease the duration of the initialization phase is given by the repetition of codes. The resulting ambiguity can be resolved by means of a partitioning of the camera image into code regions.

A. Mathematical supplement

A.1. Pose of a rigid body

Fig. A.1 shows a rigid body with its orthonormal coordinate system $(CS)_1$ which is given with respect to the orthonormal reference coordinate system $(CS)_0$. A generalized description is given by $(CS)_j$ for the coordinate system of the rigid body and $(CS)_i$ for the reference coordinate system. The pose of a rigid body is defined relative to $(CS)_i$ and consists of a position and an orientation [75]. The position is expressed by the (3×1) translation vector ${}^i\mathbf{t}_j$:

$${}^i\mathbf{t}_j = [x \ y \ z]^T. \quad (\text{A.1})$$

The latter describes the translation of the origin of $(CS)_j$ relative to $(CS)_i$. The orientation is given by the (3×3) rotation matrix ${}^i\mathbf{R}_j$:

$${}^i\mathbf{R}_j = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}. \quad (\text{A.2})$$

The latter defines the rotation of $(CS)_j$ relative to $(CS)_i$. The columns of ${}^i\mathbf{R}_j$ represent orthonormal vectors given in $(CS)_i$ and define the x-, y- and z-axis of $(CS)_j$. Since the three axes span an orthonormal basis, ${}^i\mathbf{R}_j$ is called orthogonal matrix.

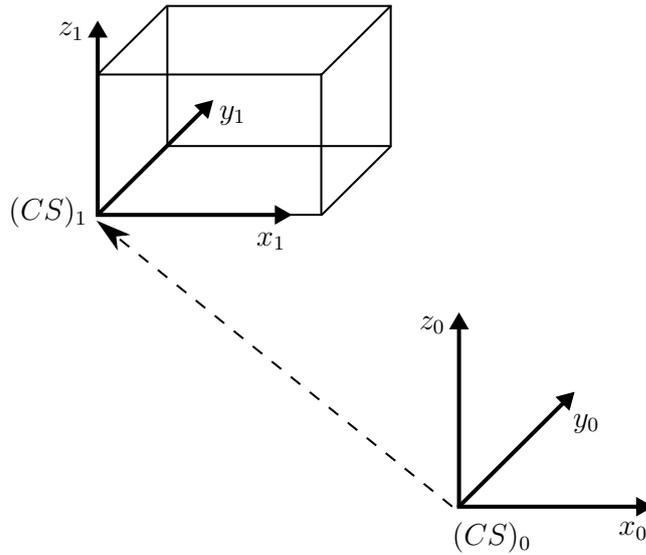


Figure A.1.: Pose of a rigid body.

A.1.1. Rotation matrix

In the following, the computation of the elementary rotation matrices is described. Subsequently, the elementary rotation matrices are used to compose the rotation matrix which describes the orientation of a rigid body with respect to a reference coordinate system. The angles α , β and γ describe the elementary rotations of a coordinate system $(CS)_j$ around the x-, y- and z-axis of a reference coordinate system $(CS)_i$, respectively. By means of the angles α , β and γ , the elementary rotation matrices can be expressed as follows [75]:

$$\begin{aligned}
 \mathbf{R}_x(\alpha) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}, \\
 \mathbf{R}_y(\beta) &= \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}, \\
 \mathbf{R}_z(\gamma) &= \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}.
 \end{aligned} \tag{A.3}$$

A compact representation of the orientation of a rigid body is given by the XYZ Euler angles. This representation is defined by the following successive rotations. The coordinate system $(CS)_j$ is rotated around the x-axis of the reference coordinate system $(CS)_i$. Afterwards, $(CS)_j$ is rotated around the new y' -axis of the previously rotated coordinate system. Subsequently, $(CS)_j$ is rotated around the new z'' -axis of the previously rotated coordinate system. The corresponding rotation matrix is computed by multiplying the elementary rotation matrices from Eq. A.3 as follows [75]:

$$\begin{aligned} \mathbf{R}(\alpha, \beta, \gamma) &= \mathbf{R}_x(\alpha)\mathbf{R}_{y'}(\beta)\mathbf{R}_{z''}(\gamma) \\ &= \begin{bmatrix} c_\alpha c_\gamma - s_\alpha c_\beta s_\gamma & -c_\alpha s_\gamma - s_\alpha c_\beta c_\gamma & s_\alpha s_\beta \\ s_\alpha c_\gamma + c_\alpha c_\beta s_\gamma & -s_\alpha s_\gamma + c_\alpha c_\beta c_\gamma & -c_\alpha s_\beta \\ s_\beta s_\gamma & s_\beta c_\gamma & c_\beta \end{bmatrix}. \end{aligned} \quad (\text{A.4})$$

The notations $c_{(\cdot)}$ and $s_{(\cdot)}$ represent the functions $\cos(\cdot)$ and $\sin(\cdot)$, respectively.

A.2. Linear coordinate transformation

In the following, the pose definition from Sec. A.1 is used to describe the linear coordinate transformation based on translation and rotation. This transformation is also known as rigid transformation and is a subclass of the more general affine linear transformation which also incorporates scaling and shear mapping. A rigid transformation offers the changing of the coordinate system for a point $\mathbf{p} = [x \ y \ z]^T$. By means of homogeneous coordinates, a rigid transformation can be expressed in matrix form. By using the homogeneous (4×4) transformation matrix ${}^i\mathbf{T}_j$, a homogeneous point $\tilde{\mathbf{p}}^j$ given in the coordinate system $(CS)_j$ is transformed into a homogeneous point $\tilde{\mathbf{p}}^i$ given in the coordinate system $(CS)_i$ [75]:

$$\tilde{\mathbf{p}}^i = {}^i\mathbf{T}_j \tilde{\mathbf{p}}^j = \begin{bmatrix} {}^i\mathbf{R}_j & {}^i\mathbf{t}_j \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \tilde{\mathbf{p}}^j. \quad (\text{A.5})$$

The inverse transformation is used to transform the point $\tilde{\mathbf{p}}^i$ into the point $\tilde{\mathbf{p}}^j$:

$$\tilde{\mathbf{p}}^j = ({}^i\mathbf{T}_j)^{-1} \tilde{\mathbf{p}}^i = {}^j\mathbf{T}_i \tilde{\mathbf{p}}^i. \quad (\text{A.6})$$

Since the rotation matrix ${}^i\mathbf{R}_j$ is an orthogonal matrix, the inverse of ${}^i\mathbf{R}_j$ is given by its transpose. Hence, ${}^j\mathbf{T}_i$ is defined by:

$${}^j\mathbf{T}_i = \begin{bmatrix} ({}^i\mathbf{R}_j)^T & -({}^i\mathbf{R}_j)^T {}^i\mathbf{t}_j \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} = \begin{bmatrix} {}^j\mathbf{R}_i & -{}^j\mathbf{R}_i {}^i\mathbf{t}_j \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} = \begin{bmatrix} {}^j\mathbf{R}_i & {}^j\mathbf{t}_i \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}. \quad (\text{A.7})$$

A.3. Moore-Penrose pseudoinverse

According to [75], the *Moore-Penrose pseudoinverse* \mathbf{A}^+ of a matrix \mathbf{A} is calculated as follows:

$$\begin{aligned} \mathbf{b} &= \mathbf{A}\mathbf{x} \\ \mathbf{A}^T\mathbf{b} &= \mathbf{A}^T\mathbf{A}\mathbf{x} \\ (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b} &= (\mathbf{A}^T\mathbf{A})^{-1}(\mathbf{A}^T\mathbf{A})\mathbf{x} \\ \underbrace{(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T}_{\mathbf{A}^+}\mathbf{b} &= \mathbf{x}. \end{aligned} \quad (\text{A.8})$$

\mathbf{A}^+ only exists if the rank of \mathbf{A} is equal to the number of columns in \mathbf{A} .

A.4. Singular value decomposition

Given a $(m \times n)$ matrix \mathbf{A} , the *singular value decomposition* (SVD) of \mathbf{A} is given as follows [21]:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{U} \text{diag}(\sigma_1, \dots, \sigma_k)\mathbf{V}^T. \quad (\text{A.9})$$

The singular values of \mathbf{A} are contained on the diagonal of the $(m \times n)$ matrix $\mathbf{\Sigma}$. The singular values are defined by the set σ_j with $j \in [1, k]$ where $k = \min\{m, n\}$. The

singular values are given in descending order $\sigma_1 \geq \dots \geq \sigma_k \geq 0$. The rank of the matrix \mathbf{A} is determined by $r = \text{rank}(\mathbf{A})$. The rank r is equivalent to the number of non-zero singular values contained on the diagonal of $\mathbf{\Sigma}$. Thus, the remaining singular values are given by $\sigma_{r+1} = \sigma_{r+2} = \dots = \sigma_k = 0$. If \mathbf{A} is a matrix of full rank then all singular values σ_j are non-zero.

The $(m \times m)$ matrix \mathbf{U} and the $(n \times n)$ matrix \mathbf{V} are orthogonal matrices. The columns of the matrix \mathbf{U} and \mathbf{V} define the left-singular vectors and the right-singular vectors of the matrix \mathbf{A} , respectively. Moreover, the columns of the matrix \mathbf{U} and \mathbf{V} also define the eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$, respectively. By means of the singular values σ_j , the corresponding eigenvalues of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$ can be calculated. The relationship between a singular value σ and an eigenvalue λ is defined as follows:

$$\lambda = \sigma^2. \tag{A.10}$$

A.5. Principal component analysis

Given a set of I d -dimensional data points, the *principal component analysis* (PCA) is used to compute the principal axes (principal components) of the data points [37]. A set of d -dimensional data points is referred to as \mathbf{p}_i with $i \in [1, I]$ where I defines the number of data points. The d -dimensional mean vector \mathbf{m} is calculated by:

$$\mathbf{m} = \frac{1}{I} \sum_{i=1}^I \mathbf{p}_i. \tag{A.11}$$

Subsequently, the matrix $\mathbf{\Psi}$ is computed as follows:

$$\mathbf{\Psi} = [\mathbf{p}_1 - \mathbf{m}, \mathbf{p}_2 - \mathbf{m}, \dots, \mathbf{p}_I - \mathbf{m}]. \tag{A.12}$$

The matrix $\mathbf{\Psi}$ is used to calculate the $(d \times d)$ covariance matrix \mathbf{C} :

$$\mathbf{C} = \frac{1}{I-1} \mathbf{\Psi} \mathbf{\Psi}^T. \quad (\text{A.13})$$

The principal components of the data points \mathbf{p}_i are given by the eigenvectors of the covariance matrix \mathbf{C} . In order to use the principal components it is also necessary to compute the eigenvalues of \mathbf{C} . The eigenvector that corresponds to the largest eigenvalue is the direction of greatest variation. The eigenvector that corresponds to the second largest eigenvalue is the direction which incorporates the next highest variation. The same principle applies for all remaining eigenvectors. In order to calculate the eigenvectors and the eigenvalues of \mathbf{C} , the SVD of the matrix $\mathbf{\Psi}$ has to be computed (see Sec. A.4):

$$\mathbf{\Psi} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T = \mathbf{U} \text{diag}(\sigma_1, \dots, \sigma_d) \mathbf{V}^T. \quad (\text{A.14})$$

The columns of the $(d \times d)$ matrix \mathbf{U} are the eigenvectors of $\mathbf{\Psi} \mathbf{\Psi}^T$. The columns of \mathbf{U} are referred to as the set \mathbf{u}_j with $j \in [1, d]$. The eigenvalues of the matrix $\mathbf{\Psi} \mathbf{\Psi}^T$ are determined by using the singular values of the matrix $\mathbf{\Psi}$. These singular values are located on the diagonal of the computed $(d \times I)$ matrix $\mathbf{\Sigma}$ and are referred to as the set σ_j with $j \in [1, d]$. The eigenvalues of the matrix $\mathbf{\Psi} \mathbf{\Psi}^T$ are defined by the squared singular values σ_j^2 (see Eq. A.10). By using the following eigenvalue problem for the matrix \mathbf{C} , the sought eigenvectors and eigenvalues of the covariance matrix \mathbf{C} can be determined:

$$\underbrace{\frac{1}{I-1} \mathbf{\Psi} \mathbf{\Psi}^T}_{\mathbf{C}} \mathbf{u} = \underbrace{\frac{1}{I-1} \sigma^2}_{\lambda} \mathbf{u}. \quad (\text{A.15})$$

Thus, the computed eigenvectors \mathbf{u}_j also define the eigenvectors of the covariance matrix \mathbf{C} . As defined in Eq. A.15, the eigenvalues of \mathbf{C} are given by the set λ_j with $j \in [1, d]$. Since the computation of the eigenvalues λ_j is based on the SVD, the eigenvalues λ_j are arranged in descending order $\lambda_1 \geq \dots \geq \lambda_d > 0$. The corresponding set of eigenvectors \mathbf{u}_j are orthonormal vectors and define the principal components of the data points \mathbf{p}_i .

A.6. Iterative closest point algorithm

A set of 3D points given in an object coordinate system $(CS)_g$ is described by \mathbf{p}_m^g with $m \in [1, M]$ where M is the amount of points. A set of 3D points given in the camera coordinate system $(CS)_c$ is described by \mathbf{p}_n^c with $n \in [1, N]$ where N is the amount of points. The amount of points in both sets can be different and the point correspondence between both sets is not given. For the sake of clarity, the point sets \mathbf{p}_m^g and \mathbf{p}_n^c are referred to as target point set and input point set, respectively.

Using the *iterative closest point* (ICP) algorithm [8], the target point set \mathbf{p}_m^g can be registered to the input point set \mathbf{p}_n^c . The result is given by an estimate of the pose cT_g of the object coordinate system $(CS)_g$ with respect to the camera coordinate system $(CS)_c$. The two mostly used variants of the ICP algorithm use either the point-to-point distance measure or the point-to-plane distance measure.

A.6.1. Point-to-point distance

Each iteration of the ICP algorithm includes the following three steps. In the first step, the points in the input point set \mathbf{p}_n^c are used to build a KD tree as described in Sec. A.7. Using this KD tree, a nearest-neighbor search is performed for the points contained in the target point set \mathbf{p}_m^g . In this manner, points from the target point set are mapped to points from the input point set. Here, a duplicate mapping to a point from the input point set has to be avoided. The amount I of mapped points is given by the minimum of the initial amounts M and N :

$$I = \min\{M, N\}. \quad (\text{A.16})$$

The result of the nearest-neighbor search is given by the mapped point sets \mathbf{p}_i^g and \mathbf{p}_i^c with $i \in [1, I]$. The second step of the ICP algorithm is given by the minimization of the point-to-point distances. This can be expressed by the following linear least-squares (LS) problem which fits the translation vector \mathbf{t} and the rotation matrix \mathbf{R} to the mapped points:

$$\min_{\mathbf{R}, \mathbf{t}} E = \sum_{i=1}^I \|\mathbf{p}_i^c - (\mathbf{R}\mathbf{p}_i^g + \mathbf{t})\|_2^2 = \sum_{i=1}^I \|\mathbf{d}_i\|_2^2. \quad (\text{A.17})$$

In [8], the LS problem in Eq. A.17 is solved by the unit quaternion-based method described in [30]. In the third step of the ICP algorithm, the estimated parameters \mathbf{t} and \mathbf{R} are used to transform the points contained in the target point set \mathbf{p}_m^g to points with respect to the camera coordinate system $(CS)_c$ (see Eq. A.5). The resulting set of points is referred to as the new target point set. Afterwards, the three described steps of the ICP algorithm are carried out again. In each new iteration, the new target point set replaces the target point set of the previous iteration. The termination criteria of the ICP algorithm can be defined by comparing the sum E of squared differences in Eq. A.17 for the current and previous iteration. If the change of the error sum E is lower than a user-defined threshold then the ICP algorithm has converged. After convergence, the algorithm is terminated after step two of the current iteration.

As described in the beginning of this section, the ICP algorithm is used to estimate the pose ${}^c\mathbf{T}_g$. After the convergence of the ICP algorithm, the parameters of this pose can be computed by means of the result of Eq. A.17. The latter yields a translation vector \mathbf{t} and a rotation matrix \mathbf{R} for each iteration of the ICP algorithm. Thus, a set $\mathbf{t}_{(k)}$ of translations and a set $\mathbf{R}_{(k)}$ of rotation matrices have been computed. The quantity k is the iteration index with $k \in [1, K-1]$ where K defines the number of executed ICP iterations. Iteration K is not considered for $\mathbf{t}_{(k)}$ and $\mathbf{R}_{(k)}$ since step three is not executed for the final ICP iteration. As explained in Sec. A.2, a pose can be described by a homogeneous (4×4) transformation matrix \mathbf{T} which contains a translation vector \mathbf{t} and a rotation matrix \mathbf{R} . Thus, the sets $\mathbf{t}_{(k)}$ and $\mathbf{R}_{(k)}$ can be represented by the set $\mathbf{T}_{(k)}$ of poses. The sought transformation matrix ${}^c\mathbf{T}_g$ is defined by multiplying the elements of the set $\mathbf{T}_{(k)}$:

$${}^c\mathbf{T}_g = \mathbf{T}_{(K-1)}\mathbf{T}_{(K-2)} \cdots \mathbf{T}_{(1)}. \quad (\text{A.18})$$

The ICP algorithm only converges in the global minimum if the initial target point set \mathbf{p}_m^g and the input point set \mathbf{p}_n^c are already close to each other before the algorithm starts. Otherwise, the algorithm can converge in a local minimum. In a local minimum, the two point sets \mathbf{p}_m^g and \mathbf{p}_n^c are incorrectly aligned. In the scope of this work, the ICP algorithm is used for the registration of point sets which are only a few millimeters away from each other. In this manner, a successful registration is always guaranteed.

A.6.2. Point-to-plane distance

Another variant of the ICP algorithm uses the point-to-plane distance measure. This variant uses the same algorithmic structure as the point-to-point based ICP which is described in Sec. A.6.1. The point-to-plane based ICP extends the first step of the point-to-point based ICP. As described in Sec. A.6.1, the result of the first step of an ICP iteration is given by the mapped point sets \mathbf{p}_i^g and \mathbf{p}_i^c with $i \in [1, I]$. This result is the basis for the following extension.

In order to establish the point-to-plane distance measure, the points included in the input point set \mathbf{p}_n^c are used to build a KD tree as described in Sec. A.7. Subsequently, every point in the set \mathbf{p}_i^c is processed by the following procedure. A point contained in the set \mathbf{p}_i^c is defined as the query point \mathbf{p}^c . By means of the KD tree, a nearest-neighbor search is carried out to find the J nearest neighbors of \mathbf{p}^c in the set \mathbf{p}_n^c . The J nearest neighbors of \mathbf{p}^c are defined by the set \mathbf{p}_j^c with $j \in [1, J]$. Since the query point \mathbf{p}^c is contained in the KD tree as well, it is also contained in the set \mathbf{p}_j^c .

In the following, a plane is fitted to the points included in the set \mathbf{p}_j^c . The set \mathbf{p}_j^c contains the point \mathbf{p}^g which was mapped to the current query point \mathbf{p}^c . The goal is to compute the point $\hat{\mathbf{p}}^c$ on the fitted plane which is orthogonally connected to the point \mathbf{p}^g . The point $\hat{\mathbf{p}}^c$ is referred to as the orthogonal base point of \mathbf{p}^g . The point \mathbf{p}^g and its orthogonal base point $\hat{\mathbf{p}}^c$ establish the sought point-to-plane distance.

The fitting of a plane to the point set \mathbf{p}_j^c is computed by means of PCA (see Sec. A.5). In this manner, the fitted plane can be described by a coordinate system. The PCA computes the covariance matrix of the point set \mathbf{p}_j^c . The result of the PCA is given by the eigenvectors and the eigenvalues of the covariance matrix. The set of eigenvectors is defined by \mathbf{u}_w and the set of eigenvalues is defined by λ_w with $w \in [1, 3]$. The amount of eigenvalues and eigenvectors is equivalent to the dimension of the spatial points contained in the set \mathbf{p}_j^c . The eigenvalues λ_w are arranged in descending order $\lambda_1 \geq \dots \geq \lambda_3 > 0$. The corresponding (3×1) eigenvectors \mathbf{u}_w are orthonormal vectors and define the principal components of the point set \mathbf{p}_j^c .

The eigenvectors \mathbf{u}_w are used to determine a rotation matrix ${}^c\mathbf{R}_h$ of the fitted plane with respect to the camera. The z-axis of ${}^c\mathbf{R}_h$ is given by the direction of smallest variation of the respective point set \mathbf{p}_j^c . Hence, the z-axis is defined by the eigenvector \mathbf{u}_3 . The

x-axis and the y-axis of ${}^c\mathbf{R}_h$ span the fitted plane and are calculated by using a second eigenvector. The eigenvector \mathbf{u}_2 is utilized as the x-axis of ${}^c\mathbf{R}_h$. The cross product of \mathbf{u}_3 and \mathbf{u}_2 is used to determine the y-axis and hence the complete rotation matrix ${}^c\mathbf{R}_h$ is given by:

$${}^c\mathbf{R}_h = [\mathbf{u}_2 \quad \mathbf{u}_3 \times \mathbf{u}_2 \quad \mathbf{u}_3]. \quad (\text{A.19})$$

The translation vector ${}^c\mathbf{t}_h$ of the fitted plane is defined as the mean vector of the point set \mathbf{p}_j^c :

$${}^c\mathbf{t}_h = \frac{1}{J} \sum_{j=1}^J \mathbf{p}_j^c. \quad (\text{A.20})$$

The computed translation vector ${}^c\mathbf{t}_h$ and the rotation matrix ${}^c\mathbf{R}_h$ are used to define the homogeneous (4×4) transformation matrix ${}^c\mathbf{T}_h$ (see Eq. A.5). By means of the inverse transformation matrix ${}^h\mathbf{T}_c$, Eq. A.6 is used to transform the point \mathbf{p}^g to the point $\mathbf{p}^h = [x_h \quad y_h \quad z_h]^T$ with respect to the fitted plane. The point $\hat{\mathbf{p}}^h$ is the sought orthogonal base point represented with respect to the fitted plane. The point $\hat{\mathbf{p}}^h$ is computed by using the first two components of the point \mathbf{p}^h . Furthermore, the value 0 is utilized as the third component:

$$\hat{\mathbf{p}}^h = [x_h \quad y_h \quad 0]^T. \quad (\text{A.21})$$

The transformation matrix ${}^c\mathbf{T}_h$ is used to transform the point $\hat{\mathbf{p}}^h$ to the sought orthogonal base point $\hat{\mathbf{p}}^c$ with respect to the camera. The point \mathbf{p}^g and it's orthogonal base point $\hat{\mathbf{p}}^c$ establish the sought point-to-plane distance. Each query point in the set \mathbf{p}_i^c is processed by the explained procedure. The result is given by the set $\hat{\mathbf{p}}_i^c$ of orthogonal base points. Subsequently, the mapped sets $\hat{\mathbf{p}}_i^c$ and \mathbf{p}_i^g are used to solve Eq. A.17. The latter now incorporates the minimization of the point-to-plane distances.

The point-to-plane based ICP is especially useful when two point sets have to be registered which contain only few or no corresponding points. In this case, the point-to-point based ICP would lead to registration errors. By means of the described orthogonal base points, these registration errors can be reduced. The explained parameter J is a user

defined constant which determines the amount of nearest neighbors that is used to fit a plane model. In general, J should be a small value in order to achieve a more accurate surface approximation by means of the fitted planes.

A.7. KD trees

A computationally efficient method for nearest-neighbor search is given by KD trees. Given a 2-dimensional point $\mathbf{p} = [x \ y]^T$, its nearest neighbor in a set of N 2-dimensional points can be found by using a KD tree [7]. A KD tree is a binary search tree which is constructed by successive partitioning of the given set of N 2-dimensional points. An exemplary box-based representation of a KD tree for a set of 20 2-dimensional points is given in Fig. A.2. The partitioning starts with the x coordinates of the points and leaves an equal number of points on both sides of the cut. The number of points on both sides differ by one if the partitioned set contains an odd number of points. The next partitioning step is carried out along the y coordinates of both new subsets. In this manner, the construction algorithm continues to alternate the partitioning along the x and y coordinates of new subsets. The process terminates if each point is separated from each other.

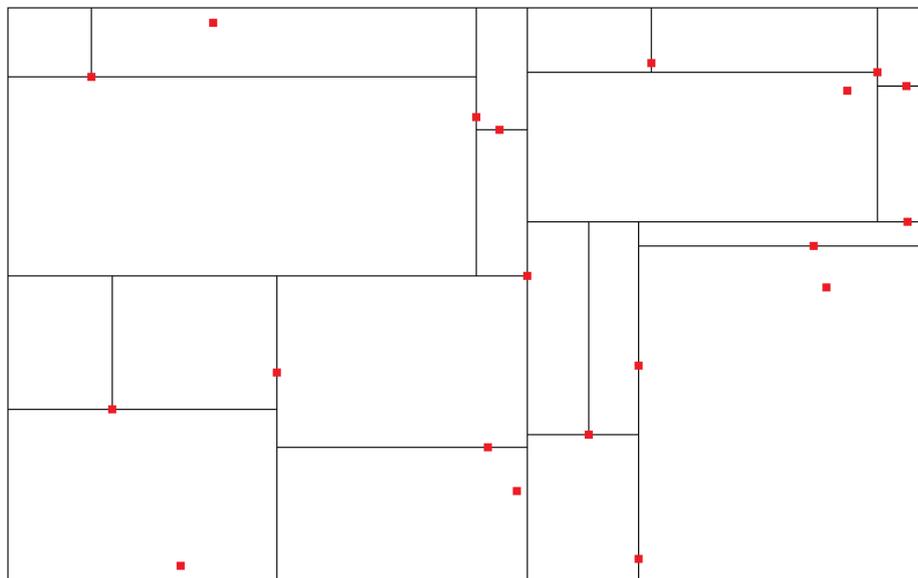


Figure A.2.: Exemplary KD tree constructed from 20 2-dimensional points.

The corresponding complete KD tree contains N leaf nodes and each leaf node uniquely represents one of the N points from the initial set. A nearest-neighbor search for a point p is carried out by traversing the KD tree with respect to the computed partitioning steps. The search terminates in one of the leaf nodes. The found leaf node represents the sought nearest neighbor of p . According to [68], the time complexity for the construction of a KD tree including N points is given by $O(N \log N)$. The time complexity for a nearest-neighbor search is defined by $O(\log N)$. Consequently, it takes $O(N \log N)$ to carry out a nearest neighbor search for all points contained in a KD tree. In this work, a KD tree implementation based on fully automated parameter selection is utilized [59].

Bibliography

- [1] R. Adhami, P. Meenen, and D. Hite. *Fundamental concepts in electrical and computer engineering with practical design problems*. Universal Publishers, 2nd edition, 2007. ISBN 978-1581129717.
- [2] S. Ahn. *Least squares orthogonal distance fitting of curves and surfaces in space*, volume 3151 of *Lecture Notes in Computer Science*. Springer, 2004. ISBN 978-3-540-28627-1.
- [3] C. Albitar, P. Graebing, and C. Doignon. Robust structured light coding for 3d reconstruction. In *IEEE 11th International Conference on Computer Vision (ICCV)*, 2007, pages 1–6.
- [4] R. Anderson and J. Parrish. The optics of human skin. *Journal of Investigative Dermatology*, **77**(1):13–19, 1981.
- [5] S. Audet and M. Okutomi. A user-friendly method to geometrically calibrate projector-camera systems. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, 2009, pages 47–54.
- [6] J. Battle, E. Mouaddib, and J. Salvi. Recent progress in coded structured light as a technique to solve the correspondence problem: A survey. *Pattern Recognition*, **31**(7):963–982, 1998.
- [7] J. Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, **18**(9):509–517, 1975.
- [8] P. Besl and H. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(2):239–256, 1992.
- [9] C. Bishop. *Neural networks for pattern recognition*. Oxford University Press, Inc, 1995. ISBN 978-0198538646.
- [10] A. Björck. *Numerical methods for least squares problems*. SIAM, Philadelphia, 1996. ISBN 0-89871-360-9.
- [11] K. Boyer and A. Kak. Color-encoded structured light for rapid active ranging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(1):14–28, 1987.
- [12] D. Caspi, N. Kiryati, and J. Shamir. Range imaging with adaptive color structured light. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**(5):470–480, 1998.
- [13] N. de Bruijn. A combinatorial problem. In *Proc. Koninklijke Nederlandse Akademie v. Wetenschappen*, 1946, pages 758–764.
- [14] M. Descovich, P. Sneed, N. Barbaro, M. McDermott, C. Chuang, I. Barani, J. Nakamura, and L. Ma. A dosimetric comparison between Gamma Knife and CyberKnife treatment plans for trigeminal neuralgia. *Journal of Neurosurgery*, **113**(Special Supplements):199–206, 2010.
- [15] T. Etzion. Constructions for perfect maps and pseudorandom arrays. *IEEE Transactions on Information Theory*, **34**(5):1308–1316, 1988.
- [16] P. Fechteler and P. Eisert. Adaptive color classification for structured light systems. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2008, pages 1–7.
- [17] D. Fofi, T. Sliwa, and Y. Voisin. A comparative survey on invisible structured light. In *Proc. SPIE 5303*,

- Machine Vision Applications in Industrial Inspection XII*, 2004, pages 90–98.
- [18] F. Forster. A high-resolution and high accuracy real-time 3d sensor based on structured light. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission*, 2006, pages 208–215.
- [19] D. Fu, G. Kuduvalli, V. Mitrovic, W. Main, and L. Thomson. Automated skull tracking for the CyberKnife image-guided radiosurgery system. In *Proc. SPIE 5744*, 2005, pages 366–377.
- [20] W. Gander, G. Golub, and R. Strebler. Least-squares fitting of circles and ellipses. *BIT Numerical Mathematics*, **34**(4):558–578, 1994.
- [21] G. Golub and C. Van Loan. *Matrix computations*. Johns Hopkins University Press, 3rd edition, 1996. ISBN 0-8018-5414-8.
- [22] F. Gray. Pulse code communication. US Patent 2632058, 1953.
- [23] P. Griffin, L. Narasimhan, and S. Yee. Generation of uniquely encoded light patterns for range data acquisition. *Pattern Recognition*, **25**(6):609–616, 1992.
- [24] C. Guan, L. Hassebrook, and D. Lau. Composite structured light pattern for three-dimensional video. *Opt. Express*, **11**(5):406–417, 2003.
- [25] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Fourth Alvey Vision Conference*, 1988, pages 147–151.
- [26] R. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding*, **68**(2):146–157, 1997.
- [27] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2nd edition, 2003. ISBN 0-521-54051-8.
- [28] J. Heikkilä and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pages 1106–1112.
- [29] A. Hofhauser, C. Steger, and N. Navab. Edge-based template matching and tracking for perspective distorted planar objects. In *4th International Symposium on Visual Computing, Las Vegas, Nevada, USA*, 2008.
- [30] B. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, **4**(4):629–642, 1987.
- [31] S. Inokuchi, K. Sato, and F. Matsuda. Range imaging system for 3-d object recognition. In *IEEE 7th International Conference on Pattern Recognition (ICPR)*, 1984, pages 806–808.
- [32] I. Ishii, K. Yamamoto, K. Doi, and T. Tsuji. High-speed 3d image acquisition using coded structured light projection. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2007, pages 925–930.
- [33] M. Ito and A. Ishii. A three-level checkerboard pattern (TCP) projection method for curved surface measurement. *Pattern Recognition*, **28**(1):27–40, 1995.
- [34] R. Jain, R. Kasturi, and B. Schunck. *Machine vision*. McGraw-Hill, Inc., 1995. ISBN 978-0070320185.
- [35] C. Je, S. Lee, and R. Park. High-contrast color-stripe pattern for rapid structured-light range imaging. *Computer Vision - ECCV 2004, Lecture Notes in Computer Science*, **3021**:95–107, 2004.
- [36] A. Jha. *A textbook of applied physics*, volume 1. I K International Publishing House Pvt. Ltd. ISBN 978-9380026770.
- [37] I. Jolliffe. *Principal component analysis*. Springer, 2nd edition, 2002. ISBN 978-0387954424.
- [38] H. Kawasaki, R. Furukawa, R. Sagawa, and Y. Yagi. Dynamic scene shape reconstruction using a single structured light pattern. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pages 1–8.
- [39] G. Kim, T. Pawlicki, Q. Le, and G. Luxton. Linac-based on-board imaging feasibility and the dosimetric

- consequences of head roll in head-and-neck IMRT plans. *Medical Dosimetry*, **33**(1):93–99, 2008.
- [40] C. Kimberling. *Triangle centers and central triangles*. Utilitas Mathematica Publishing Incorporated, 1998. Congressus Numerantium 129.
- [41] T. Koninckx and L. Van Gool. Real-time range acquisition by adaptive structured light. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**(3):432–445, 2006.
- [42] W. Kresse and D. Danko, editors. *Springer handbook of geographic information*. Springer Publishing Company, Incorporated, 2012. ISBN 978-3-540-72680-7.
- [43] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, **2**:164–168, 1944.
- [44] N. Lobontiu, J. Paine, E. Garcia, and M. Goldfarb. Design of symmetric conic-section flexure hinges based on closed-form compliance equations. *Mechanism and Machine Theory*, **37**(5):477–498, 2002.
- [45] H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, **293**(5828):133–135, 1981.
- [46] Y. Ma, S. Soatto, J. Kosecká, and S. Sastry. *An invitation to 3-D vision*, volume 26 of *Interdisciplinary Applied Mathematics*. Springer Science+Business Media New York, 1st edition, 2004. ISBN 978-1-4419-1846-8.
- [47] F. MacWilliams and N. Sloane. Pseudo-random sequences and arrays. *Proceedings of the IEEE*, **64**(12):1715–1729, 1976.
- [48] A. Manakov, H. Seidel, and I. Ihrke. A mathematical model and calibration procedure for galvanometric laser scanning systems. In *Proceedings of the 16th International Workshop on Vision, Modeling and Visualization (VMV)*, 2011, pages 207–214.
- [49] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, **11**(2):431–441, 1963.
- [50] J. Martin and J. Crowley. Experimental comparison of correlation techniques. In *International Conference on Intelligent Autonomous Systems (IAS-4)*, 1995.
- [51] M. Maruyama and S. Abe. Range sensing by projecting multiple slits with random cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(6):647–651, 1993.
- [52] M. Minou, T. Kanade, and T. Sakai. A method of time-coded parallel planes of light for depth measurement. *Transactions of the Institute of Electronics and Communication Engineers of Japan (IECE)*, **E64**(8):521–528, 1981.
- [53] T. Monks, J. Carter, and C. Shadle. Colour-encoded structured light for digitisation of real-time 3d data. In *International Conference on Image Processing and its Applications*, 1992, pages 327–330.
- [54] J. Moore, C. Davis, M. Coplan, and S. Greer. *Building scientific apparatus*. Cambridge University Press, 4th edition, 2009. ISBN 978-0521878586.
- [55] R. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissano. Structured light using pseudorandom codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**(3):322–327, 1998.
- [56] D. Moreno and G. Taubin. Simple, accurate, and robust projector-camera calibration. In *Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, 2012, pages 464–471.
- [57] H. Morita, K. Yajima, and S. Sakata. Reconstruction of surfaces of 3-d objects by M-array pattern projection method. In *Second International Conference on Computer Vision*, 1988, pages 468–473.
- [58] T. Moser, G. Habl, M. Uhl, K. Schubert, G. Sroka-Perez, J. Debus, K. Herfarth, and C. Karger. Clinical evaluation of a laser surface scanning system in 120 patients for improving daily setup accuracy in fractionated radiation therapy. *International journal of radiation oncology, biology, physics*, **85**(3):846–853, 2013.

- [59] M. Muja and D. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *International Conference on Computer Vision Theory and Applications*, 2009, pages 331–340.
- [60] N. Naff. CyberKnife radiosurgery for the treatment of brain and spine disorders: Part I: Technology and treatment process. *Contemporary Neurosurgery*, **30**(26):1–5, 2008.
- [61] S. Nath, J. Lawson, J. Wang, D. R. Simpson, C. B. Newman, J. Alksne, A. Mundt, and K. Murphy. Optically-guided frameless linac-based radiosurgery for brain metastases: Clinical experience. *Journal of Neuro-Oncology*, **97**(1):67–72, 2010.
- [62] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, **9**(1):62–66, 1979.
- [63] J. Pages, J. Salvi, and J. Forest. A new optimised de Bruijn coding strategy for structured light patterns. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, 2004, pages 284–287.
- [64] J. Pages, C. Collewet, F. Chaumette, and J. Salvi. An approach to visual servoing based on coded light. In *Proceedings 2006 IEEE International Conference on Robotics and Automation (ICRA)*, 2006, pages 4118–4123.
- [65] T. Pavlidis. Curve fitting with conic splines. *ACM Transactions on Graphics (TOG)*, **2**(1):1–31, 1983.
- [66] W. Peterson and E. Weldon. *Error-correcting codes*. MIT Press, 1972. ISBN 978-0262160391.
- [67] J. Posdamer and M. Altschuler. Surface measurement by space-encoded projected beam systems. *Computer Graphics and Image Processing*, **18**(1):1–17, 1982.
- [68] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical recipes: The art of scientific computing*. Cambridge University Press, 3rd edition, 2007. ISBN 978-0521880688.
- [69] C. Rasmussen and C. Williams. *Gaussian processes for machine learning*. MIT Press, 2006. ISBN 978-0262182539.
- [70] V. Rodehorst and A. Koschan. Comparison and evaluation of feature point detectors. In *Proc. of the 5th Int. Symposium - Turkish-German Joint Geodetic Days*, 2006. ISBN 3980903044.
- [71] J. Salvi, J. Batlle, and E. Mouaddib. A robust-coded pattern projection for dynamic 3d scene measurement. *Pattern Recognition Letters*, **19**(11):1055–1065, 1998.
- [72] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado. A state of the art in structured light patterns for surface profilometry. *Pattern Recognition*, **43**(8):2666–2680, 2010.
- [73] G. Sansoni, M. Carocci, and R. Rodella. Calibration and performance evaluation of a 3-d imaging sensor based on the projection of structured light. *IEEE Transactions on Instrumentation and Measurement*, **49**(3): 628–636, 2000.
- [74] A. Santhanam, D. Low, and P. Kupelian. TH-C-BRC-11: 3d tracking of interfraction and intrafraction head and neck anatomy during radiotherapy using multiple Kinect sensors. *Medical Physics*, **38**(6):3858, 2011.
- [75] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics. Modelling, planning and control*. Advanced Textbooks in Control and Signal Processing. Springer London, 2009. ISBN 978-1-84628-641-4.
- [76] D. Simpson, J. Lawson, S. Nath, B. Rose, A. Mundt, and L. Mell. A survey of image-guided radiation therapy use in the United States. *Cancer*, **116**(6):3953–3960, 2010.
- [77] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, **14**(3): 199–222, 2004.
- [78] M. Snell, F. Bova, D. Larson, D. Leavitt, W. Lutz, E. Podgorsak, and A. Wu. Stereotactic radiosurgery. Report of TG42 (Medical Physics, WI, 1995).
- [79] C. Sun. Fast stereo matching using rectangular subregioning and 3d maximum-surface techniques. *International Journal of Computer Vision*, **47**(1-3):99–117, 2002.

- [80] S. Suzuki and K. Abe. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, **30**(1):32–46, 1985.
- [81] E. Tryggestad, M. Christian, E. Ford, C. Kut, Y. Le, Y. Sanguineti, D. Song, and L. Kleinberg. Inter- and intrafraction patient positioning uncertainties for intracranial radiotherapy: A study of four frameless, thermoplastic mask-based immobilization strategies using daily cone-beam CT. *International journal of radiation oncology, biology, physics*, **80**(1):281–290, 2011.
- [82] D. Tsai and C. Lin. Fast normalized cross correlation for defect detection. *Pattern Recognition Letters*, **24**(15):2625–2631, 2003.
- [83] A. Ulusoy, F. Calakli, and G. Taubin. One-shot scanning using de Bruijn spaced grids. In *IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops)*, 2009, pages 1786–1792.
- [84] H. van Tilborg and S. Jajodia, editors. *Encyclopedia of cryptography and security*. Springer, 2nd edition, 2011. ISBN 978-1-4419-5905-8.
- [85] B. Wagner, P. Stüber, T. Wissel, R. Bruder, A. Schweikard, and F. Ernst. Accuracy of object tracking based on time-multiplexed structured light. In *12th Annual Meeting of the German Society for Computer and Robot assisted Surgery (CURAC)*, 2013, pages 139–142.
- [86] B. Wagner, P. Stüber, T. Wissel, R. Bruder, A. Schweikard, and F. Ernst. Time-multiplexed structured light for head tracking. In *44th Annual Meeting of the German Society for Medical Physics (DGMP)*, 2013, pages 199–202.
- [87] B. Wagner, P. Stüber, T. Wissel, R. Bruder, A. Schweikard, and F. Ernst. Accuracy analysis for triangulation and tracking based on time-multiplexed structured light. *Medical Physics*, **41**(8), 2014.
- [88] B. Wagner, P. Stüber, T. Wissel, R. Bruder, A. Schweikard, and F. Ernst. Ray interpolation for generic triangulation based on a galvanometric laser scanning system. In *IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, 2015. accepted.
- [89] J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(10):965–980, 1992.
- [90] S. Winkelbach, S. Molkenstruck, and F. Wahl. Low-cost laser range scanner and fast surface registration approach. *Pattern Recognition, Lecture Notes in Computer Science*, **4174**:718–728, 2006.
- [91] T. Wissel, B. Wagner, R. Bruder, P. Stüber, A. Schweikard, and F. Ernst. Data-driven learning for calibrating galvanometric laser scanners. *IEEE Sensors Journal*. submitted.
- [92] T. Wissel, P. Stüber, B. Wagner, R. Bruder, A. Schweikard, and F. Ernst. Preliminary study on optical feature detection for head tracking in radiation therapy. In *IEEE 13th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2013, pages 1–5.
- [93] T. Wissel, P. Stüber, B. Wagner, R. Dürichen, R. Bruder, A. Schweikard, and F. Ernst. Tissue thickness estimation for high precision head-tracking using a galvanometric laser scanner - a case study. In *36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2014, pages 3106–3109.
- [94] A. Wu, G. Lindner, A. H. Maitz, A. M. Kalend, L. D. Lunsford, J. C. Flickinger, and W. D. Bloomer. Physics of gamma knife approach on convergent beams in stereotactic radiosurgery. *Int. J. Radiat. Oncol., Biol., Phys.*, **18**(4):941–949, 1990.
- [95] J. Xia and R. Siochi. A real-time respiratory motion monitoring system using Kinect: Proof of concept. *Medical Physics*, **39**(5):2682–2685, 2012.
- [96] L. Zhang, B. Curless, and S. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *Proc. First International Symposium on 3D Data Processing Visualization and Transmission*, 2002, pages 24–36.
- [97] Z. Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. *Image and Vision Computing*, **15**:59–76, 1997.

Bibliography

- [98] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Proc. IEEE International Conference on Computer Vision*, 1999, pages 666–673.

List of Figures

1.1	Measurement system.	4
1.2	This figure shows a diagram of the laser scanning system. The laser source generates an infrared laser beam which is redirected by two galvanometer driven mirrors. The outgoing laser beam hits the surface of an object and the reflection of the beam is captured by the triangulation camera.	5
2.1	Classification of structured light systems: (a) single spot, (b) single slit, (c) multiple slits, (d) grid. Here, image (d) only contains the projected and the captured grid without showing the path of the light rays.	8
2.2	Geometric features used in [3].	15
3.1	Extended code sequence containing v sub-sequences.	27
3.2	This figure exemplarily shows 1024 laser spots which are projected onto the forehead of a person. The binary representation of the indices of the laser spots (range: 0 to 1023) defines the code words. The encoding is realized by means of $n = 10$ code frames. The location of the code words in the projected grids is computed by a random permutation. The length of each sub-sequence is defined by $j = 2$ and thus the number of sub-sequences is given by $v = 10$. The images show the projection of the first three sub-sequences of an extended code sequence. The order is given by top-left to bottom-right.	28
3.3	Captured grid of 72×56 (4032) laser spots which were projected on a planar surface.	34
3.4	Exemplary close up of nine captured laser spots contained in the grid in Fig. 3.3.	34
4.1	Central projection.	42
4.2	Sensor and image coordinate system.	43
4.3	Types of radial distortion.	45

4.4	Tangential distortion of lines.	46
4.5	This figure shows the geometric relationships for one pair of corresponding image points p^n and p^{n^*} which are given in the image I and I^* , respectively. The spatial point p^g is given in the object coordinate system $(CS)_g$ and is projected on p^n and p^{n^*} . The connection between the camera center c and the camera center c^* represents the baseline. . .	48
4.6	Calibration pattern.	51
4.7	Epipolar geometry.	58
5.1	This figure exemplarily shows a grid of 72×56 (4032) laser spots which is projected onto the calibration body. The binary representation of the indices of the laser rays (range: 0 to 4031) defines the code words. The encoding is realized by means of $n = 12$ code frames. The length of each sub-sequence is defined by $j = 2$ and thus the number of sub-sequences is given by $v = 12$. The images show the projection of the first sub-sequence of an extended code sequence.	66
5.2	Typical Gaussian distribution of the pixel error of all laser centers that are used for the calibration of the laser rays.	67
5.3	The black dots represent the register value pairs of the given set V_1 . The red dots represent the new register value pairs from the set V_2 that are used as interpolation points. The dashed lines are given for the purpose of orientation.	69
6.1	Stair phantom of dimensions 12×6 cm ²	80
6.2	Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of template matching.	85
6.3	Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of ellipse fitting.	86
6.4	Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of paraboloid fitting based on a 3×3 sub-matrix.	87
6.5	Triangulation accuracy for the stair phantom at every iteration k starting at $k = 1$. The detection of the centers of the laser spots is carried out by means of paraboloid fitting based on a 5×5 sub-matrix.	88

6.6	Triangulation accuracy for the stair phantom. The curves show the results of the linear triangulation for all four center detection methods from Figs. 6.2 to 6.5.	89
6.7	Triangulation accuracy for the stair phantom. The blue and the red curve show the accuracy based on a set of calibrated laser rays. For comparison, the green and the black curve show the accuracy for the same set of laser rays which is computed by interpolation.	90
6.8	Triangulation accuracy for a plane. The blue and the red curve show the accuracy based on a set of calibrated laser rays. For comparison, the green and the black curve show the accuracy for the same set of laser rays which is computed by interpolation.	93
6.9	Reconstruction of a plane which was acquired by means of the <i>calibration LT</i> method (a) and Microsoft's Kinect camera (b). The triangulation errors are mapped to the color bars. The area that was reconstructed by the Kinect camera is smaller since the device provides a higher reconstruction density. Both surface plots are based on an equivalent amount of approximately 4000 triangulated points.	96
6.10	Front view on the reconstructed forehead surface of subject 1. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.	100
6.11	Front view on the reconstructed forehead surface of subject 2. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.	101
6.12	Front view on the reconstructed forehead surface of subject 3. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.	102
6.13	Histogram of the absolute triangulation errors for the reconstructed forehead surface of subject 1.	103
6.14	Histogram of the absolute triangulation errors for the reconstructed forehead surface of subject 2.	104
6.15	Histogram of the absolute triangulation errors for the reconstructed forehead surface of subject 3.	104
6.16	Front view on the reconstructed forehead surface of subject 2. The blue points represent the triangulated points. The latter were used to interpolate the copper surface.	106
6.17	Polystyrene head phantom.	107

6.18	Target point set of the polystyrene head phantom.	110
6.19	Exemplary surface reconstruction of the face of the polystyrene head phantom. The reconstruction was obtained after a translational displacement was applied to the robot end-effector.	111
6.20	Registration result for the target point set and the surface reconstruction shown in Figs. 6.18 and 6.19, respectively.	111
7.1	Model for a galvanometric laser projection system.	116
7.2	Histogram of the triangulation errors for the reconstructed planar surface.	120
A.1	Pose of a rigid body.	128
A.2	Exemplary KD tree constructed from 20 2-dimensional points.	137

List of Tables

3.1	This table exemplarily presents possible configurations for the projection of a grid of laser spots. The quantity t defines the time period that lasts from the moment when the first laser spot hits the surface to the moment when the last laser spot hits the surface. The grid dimensions apply for a projection distance of 60 cm.	26
3.2	Exemplary parameter configurations for the projection of extended code sequences.	31
6.1	Triangulation accuracy in [mm] for the stair phantom at final iteration K .	86
6.2	This table presents the difference between the accuracy of the linear triangulation for calibrated and interpolated laser rays from Fig. 6.7.	91
6.3	This table presents the difference between the accuracy of the linear triangulation for calibrated and interpolated laser rays from Fig. 6.8.	94
6.4	Comparison of the triangulation accuracy for the developed laser scanning system and Microsoft's Kinect camera.	97
6.5	Triangulation accuracy in [mm] for the reconstructed forehead surface of all three subjects.	102
6.6	Triangulation accuracy in [mm] for the reconstructed forehead surface of subject 2 shown in Fig. 6.16.	105
6.7	Tracking accuracy for the polystyrene head phantom. The tracking was carried out by using surface reconstructions of the whole face of the phantom.	112
6.8	Tracking accuracy for two rigid objects and the forehead of a human head.	112
7.1	Comparison of the triangulation accuracy for the physical and the simulated laser scanning system.	121
7.2	Comparison of the triangulation accuracy for different camera translations in the simulated laser scanning system.	121

List of Abbreviations

ANN Artificial neural network

CBCT Cone-beam computed tomography

CC Cross-correlation

CCD Charge-coupled device

CT Computed tomography

DLP Digital light processing

DLT Direct linear transformation

DOE Diffractive optical element

DOF Degrees of freedom

GP Gaussian process

HDR High dynamic range

HSI (Hue, saturation, intensity) space

HSV (Hue, saturation, value) space

ICP Iterative closest point

IGRT Image-guided radiation therapy

LCD Liquid-crystal display

LS Least-squares

LSB Least significant bit

MSB Most significant bit

MVCT Megavoltage computed tomography

NCC Normalized cross-correlation

OBI On-board Imager

PCA Principal component analysis

RGB (Red, green, blue) space

RMS Root mean square

ROI Region of interest

SAD Sum of absolute difference

SRT Stereotactic radiotherapy

SSD Sum of squared difference

SVD Singular value decomposition

SVR Support vector regression

Acknowledgements

I would particularly like to thank my supervisor, Prof. Dr.-Ing. Achim Schweikard, who gave me the opportunity to work in the field of tracking in radiation therapy. I would like to express my gratitude to him for enabling me to freely realize my ideas.

Furthermore, I would like to thank my colleagues with whom I worked together in the head tracking project. Especially, I would like to thank Dr. rer. nat. Floris Ernst for managing our project, reviewing my work and providing profitable discussions. In this scope, I would also like to mention Patrick Stüber, Tobias Wissel and Ralf Bruder who offered productive team work and who developed crucial parts of our prototype. Moreover, I would like to mention Cornelia Rieckhoff and Olga Schachmatova who organized the administrative work and who always provided help for administrative questions.

Finally, I would like to thank my mother Tosca Wagner and my girlfriend Rocío Aldana Figueroa for encouragement and proofreading.