Out of lab calibration of a rotating 2D scanner for 3D mapping

Conference Paper · June 2017

DOI: 10.1117/12.2270298

Rainer Koch
Technische Hochschule Nürnberg Georg Simon Ohm

Stefan May
Technische Hochschule Nürnberg Georg Simon Ohm

Andreas Nuchter
University of Wuerzburg

Out of lab calibration of a rotating 2D Scanner for 3D mapping

Rainer Koch[a], Lena Böttcher[a], Maximilian Jahrsdörfer[a], Johannes Maier[a], Malte Trommer[a], Stefan May[a] and Andreas Nüchter[b]

[a]Technische Hochschule Nürnberg Georg Simon Ohm
Kesslerplatz 12, 90489 Nürnberg, Germany
[b]Informatics VII – Robotics and Telematics, Julius-Maximilians University Würzburg, Am Hubland, 97074 Würzburg, Germany

ABSTRACT

Mapping is an essential task in mobile robotics. To fulfil advanced navigation and manipulation tasks a 3D representation of the environment is required. Applying stereo cameras or Time-of-flight cameras (TOF cameras) are one way to archive this requirement. Unfortunately, they suffer from drawbacks which makes it difficult to map properly. Therefore, costly 3D laser scanners are applied. An inexpensive way to build a 3D representation is to use a 2D laser scanner and rotate the scan plane around an additional axis.

A 3D point cloud acquired with such a custom device consists of multiple 2D line scans. Therefore the scanner pose of each line scan need to be determined as well as parameters resulting from a calibration to generate a 3D point cloud. Using external sensor systems are a common method to determine these calibration parameters. This is costly and difficult when the robot needs to be calibrated outside the lab. Thus, this work presents a calibration method applied on a rotating 2D laser scanner. It uses a hardware setup to identify the required parameters for calibration. This hardware setup is light, small, and easy to transport. Hence, an out of lab calibration is possible. Additional a theoretical model was created to test the algorithm and analyse impact of the scanner accuracy.

The hardware components of the 3D scanner system are an HOKUYO UTM-30LX-EW 2D laser scanner, a Dynamixel servo-motor, and a control unit. The calibration system consists of an hemisphere. In the inner of the hemisphere a circular plate is mounted. The algorithm needs to be provided with a dataset of a single rotation from the laser scanner. To achieve a proper calibration result the scanner needs to be located in the middle of the hemisphere. By means of geometric formulas the algorithms determine the individual deviations of the placed laser scanner. In order to minimize errors, the algorithm solves the formulas in an iterative process.

First, the calibration algorithm was tested with an ideal hemisphere model created in Matlab. Second, laser scanner was mounted differently, the scanner position and the rotation axis was modified. In doing so, every deviation, was compared with the algorithm results. Several measurement settings were tested repeatedly with the 3D scanner system and the calibration system. The results show that the length accuracy of the laser scanner is most critical. It influences the required size of the hemisphere and the calibration accuracy.

Keywords: calibration, 3D laser scanner, out-of-lab, algorithm

1. INTRODUCTION

Mapping is an essential task in mobile robotics. To fulfil advanced navigation and manipulation tasks a 3D representation of the environment is required. Applying stereo cameras or Time-of-flight cameras (TOF cameras) are one way to archive this requirement. Unfortunately, stereo cameras have drawbacks when scanning low textured areas or lightning is insufficient. In contrast, TOF cameras have a low measurement range and suffer from strong light influences, e.g., sunlight. This makes it difficult to use these sensors out of the box.

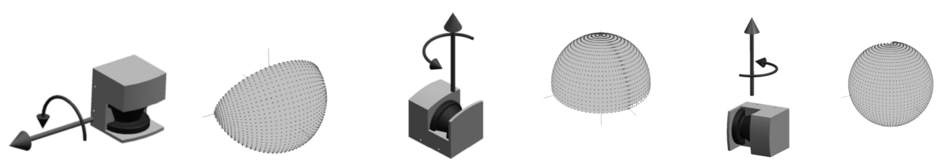
Further author information: (Send correspondence to Rainer Koch)
Rainer Koch: E-mail: rainer.koch@th-nuernberg.de, Telephone: +49 911 5880-1411

In many cases the environment is unknown and therefore, the robot should be capable to fulfil indoor and outdoor missions without any restrictions. Therefore, costly 3D laser scanners are used even they have limitations when scanning transparent and specular reflective objects. An inexpensive way to build a 3D representation is to use a 2D laser scanner and move the scan plane.

Figures 1 illustrates three methods to archive such a 3D representation with a 2D laser scanner. First, c.f. Figure 1a, the scan plane is pitched, e.g., by a servo motor. After each step a scan is taken. This brings up a drawback since a synchronisation between movement and scanning is needed. Only if the scanner has reached his position a scan could be taken. The resulting point cloud equals the peace of a ball.

Second, c.f. Figure 1b, the laser scanner is rotated with a continuous velocity. The middle of the scan plane is aligned with the rotation axis. Since the movement is constant and much slower than the scanning speed it can be assumed that the angle between the scans is constant. Depend on the scanning angle of the 2D laser scanner the resulting point cloud equals a half ball.

The third method, c.f. Figure 1c, also rotates the scan plane with a constant velocity. In compare to the second method the scan plane is orientated parallel to the rotation axis. Hence, the resulting point cloud equals a full ball. It is understood that this configuration will be preferred since it allows the maximum field of view.



(a) Nicking scanner: scan plane is moved up and down.

(b) Rotating scanner: rotating axis is aligned with the middle of scan plane.

(c) Rotating scanner: scan plane is aligned horizontal to rotating axis.

Figure 1: Methods to archive a 3D scan by rotating a 2D laser scanner.[1]

The drawback of all these methods lay in the alignment of the scanner. Hence, such a system needs to be calibrated to result in a proper 3D point cloud. It is understood that the calibration is only valid as long the system remains unchanged. Unfortunately, this is not always the case. Especially when operating in harsh environment where it is necessary to adapt or fix the robot, e.g., when operating in a rescue scenario. In such a case it is hardly possible to bring the robot back to the lab to calibrate the sensor. A handy calibration system, easy to use, would be needed.

2. RELATED WORK

Using external sensor systems is a common method to determine calibration parameters. This is costly and difficult, especially when the robot needs to be calibrated outside the lab. Most research concern calibration between 2D and 3D sensors, e.g., a RGB-Camera with a 3D laser scanner.

For calibration Qayyum et al.[2] use the depth image against the color image of a an omni-directional camera. The user has to select common features between depth and color image in order to obtain rigid body transformation between laser and camera. The rigid transformation estimated earlier from different resolution sensors is provided as a initial guess to a non-linear least square estimation.

In compare, Niola et al.[3] determine the relative positions between laser module and camera for calibration procedure. By using a cylindrical lens a plane light is projected and an optical filter segments the laser stripes from the rest of the scene. These calibration method is based on the modelling of the geometrical relationship between the 3D coordinates of the laser stripe on the target and its digital coordinates in the image plane.

Martinez et al.[4] need a set of consecutive scans of a small room with stairs for experimental calibration data. In the first place data synchronization is performed. Then, geometric parameters can be obtained.

Yang et al.[5] randomly place directional pattern in the view-field of the imaging system. Then, a flexible planar calibration pattern was used (arbitrarily placed) to extract known feature points in order to construct other unknown features points.

In compare the presented algorithm is designed to calibrate the hardware setup of a rotating 2D laser scanner to archive a 3D point cloud.

3. HARDWARE

Following section describes the hardware setup of the 3D scanner system as well as the setup of the calibration system. The scanner, as well as the calibration algorithm are implemented as a ROS-node.[6]

3.1 3D laser scanner

The hardware components of the 3D scanner system are an HOKUYO UTM-30LX-EW[7] 2D laser scanner, a Dynamixel servo-motor, and a control unit. The scanner uses a laser source ($\lambda = 905\,nm$) to scan a 270° semicircular field. A Dynamixel controller regulates the rotation speed of the servo-motor by receiving speed commands via USB from a ROS-node, called *Hokuyo3D-node*. The scanner rotates around the z-axis which is marked by a beige filled semicircle in Figure 2. This equals the second methode presented by Wulf et al.,[1] cf. Figure 1. When rotating the scanner, it supplies scans via ethernet with a frequency of 25 Hz to the *Hokuyo3D-node*. The node stitches the single scans together and builds a point cloud. The scanning plane of the Hokuyo is in parallel with the z-plane and marked by red line semicircle. Therefore, at one data take of the Hokuyo, the amount of points in the β-angle is fixed to $N_\beta = 1080$. The amount of data takes on the α-angle varies with the speed of the scanner. The total amount of scan points result to $N = r \cdot N_\beta$ where N is the total amount of scan points, r is the amount data takes per half rotation, and N_β are the amount of points takes per scan of Hokuyo.

A message containing the point cloud with a tuple $\vec{P}$ is published after the scanner has rotated $\alpha = 180°$. $\vec{P}$ consists two datasets $\vec{e}_1$ and $\vec{e}_2$ resulting from the first and the second echo of the scanner.

$$\vec{P} = \{\vec{e}_1, \vec{e}_2\} \qquad \text{with} \qquad \vec{e}_k = \begin{cases} d_{k,i} | i = 1 \cdots N \\ \alpha_{k,i} | i = 1 \cdots N \\ x_{k,i} | i = 1 \cdots N \\ y_{k,i} | i = 1 \cdots N \\ z_{k,i} | i = 1 \cdots N \\ \vec{n}_{0_{k,i}} | i = 1 \cdots N \\ I_{k,i} | i = 1 \cdots N \\ m_{k,i} | i = 1 \cdots N \end{cases} \qquad \text{with} \qquad k = \{1, 2\}. \tag{1}$$

where $\vec{P}$ is the tuple of e_1 (Echo 1) and e_2 (Echo 2) with its points and attributes. $\vec{e_k}$ is a tuple with the point cloud and the corresponding attributes. Each point i has d for the distance from the point to the scanner, x, y, z for the coordinates of the point, α for the angle of the scanning plane, $\vec{n}_0$ for the surface normal vector, I for the intensity, and m to mask points.

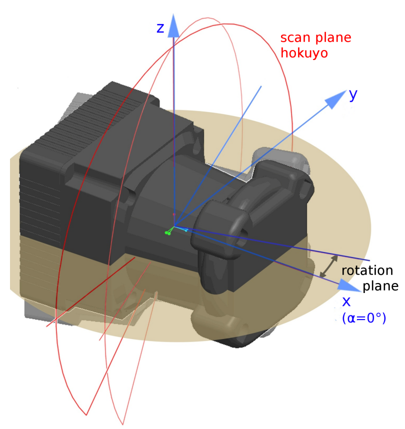
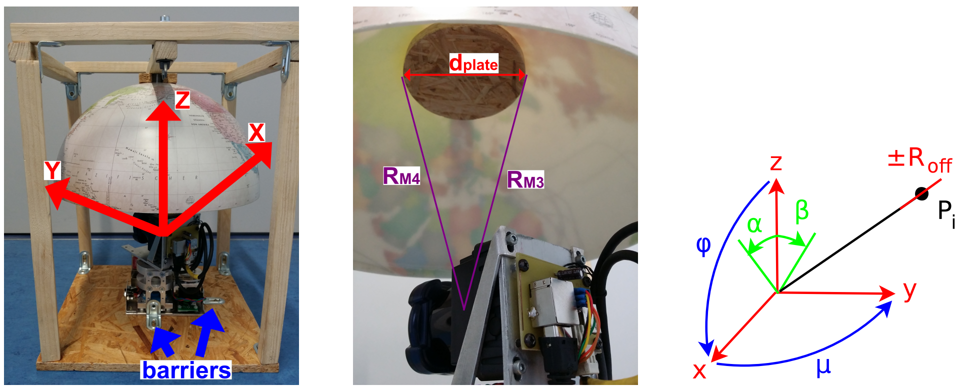


Figure 2: Koordinate system and scannig plane of rotating Hokuyo UTM-30LX-EW.

3.2 Setup calibration system

The calibration system consists of a hemisphere, cf. Figure 3a, suspended on the top to a framework. The hemisphere is not filled and has a inner diameter $D_M = 300\ mm$, a thickness of $d_{thick} = 1.5\ mm$, and is located $h_{hemi} = 193\ mm$ above the ground plane. In the inner of the hemisphere a circular plate $d_M = 102\ mm$ is mounted $h_{plate} = 323\ mm$, cf. Figure 3b. The 3D scanner has to be positioned exactly in the center of the hemisphere, such that the rotation axis is exactly aligned with the axis of the hemisphere. Therefore, three barriers, which are fixed to the ground plate of the calibration system, supports the $y > 0$ alignment of the scanner. Besides, following assumptions where made to simplify the calibration approach: $z > 0$, $\alpha < 45°$, and $\beta > 0°$.



(a) HOKUYO 3D laser scanner with calibration setup. The scanner is located roughly in the center of the hemisphere. Three barriers supports the alignment.

(b) The inner of the hemisphere with the round plate on the top.

(c) Coordinate system with calibration errors (α, β), scan rotation angles (φ, μ), and measurement P_i with its deviation $\pm R_{off}$.

Figure 3: Hokuyo 3D laser scanner with calibration setup.

4. APPROACH

The calibration approach, illustrated in Figure 4a, is divided into four steps: receive scan, prepare point cloud, calibration algorithm, and print out calibration file.

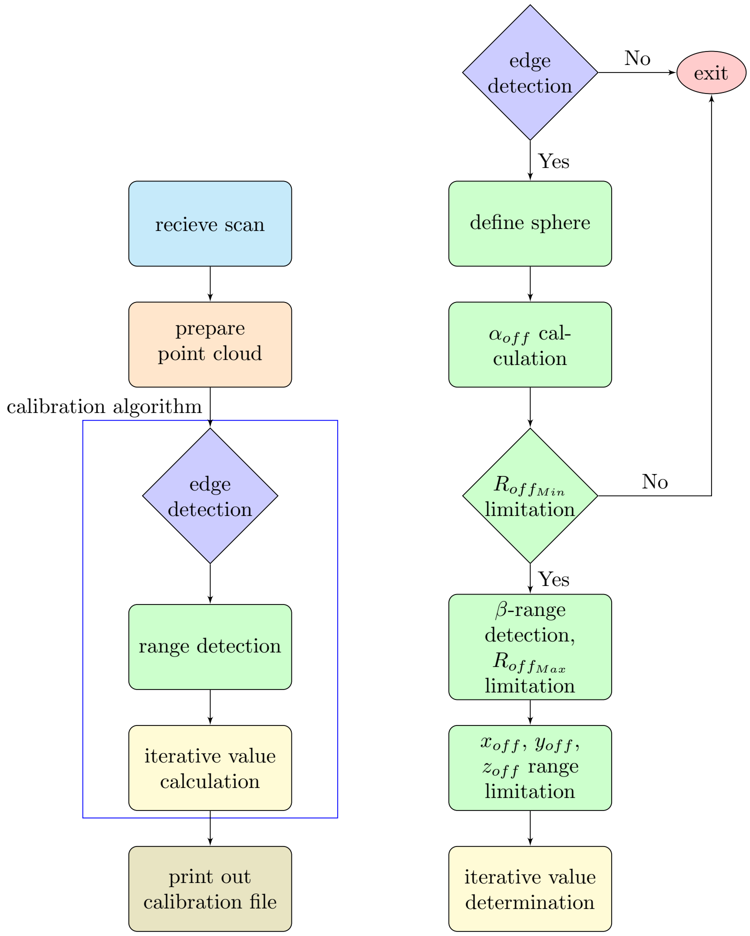
In the first step (highlighted turquoise) the algorithm receives a point cloud $\vec{P}$ (see Equation 3.1) from the *Hokuyo-node*.

In the following step (highlighted orange) the point cloud is preprocessed and customized. First, all points which are to far away, to be located on the surface, are limited. Second, the required angles φ_i are calculated and stored with the point cloud. Finally, an averaging of the points from multiple scans is done to eliminate outliers.

As illustrated in Figure 4 the calibration algorithm can be simplified into three sub-steps: detection, range definition, determine calibration values.

First sub-step (highlighted purple), the edges of the hemisphere and the plateau are detected and result the four corners C_1, C_2, C_3, and C_4, see Figure 5a. Therefore, the function *edgeDetection()* searches for jumping edges. It takes two values and combines them with each other. In case of a significant differences it assumes a corner. The corners are fundamental for the following sub-steps. Therefore, the algorithm exits if the corners could not be identified.

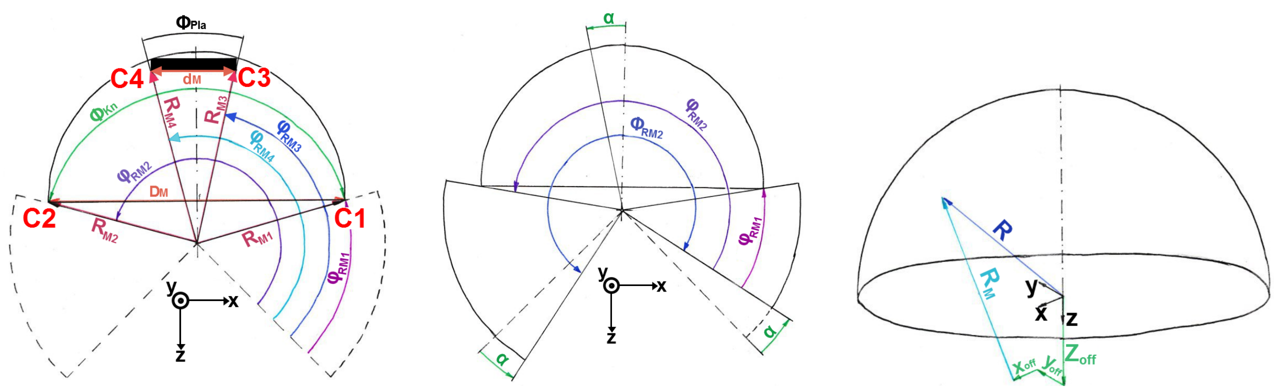
Second sub-step (highlighted green), the ranges (minimum and maximum) of the values R_{off}, α, β, x_{off}, y_{off}, and z_{off} are determined, c.f. Figure 3c. Therefore, the function *alphaRange()* determines the amount of possible measurement points between C_1 and C_2. With the angles of these points, which is directly related to the index (α = index/angular_resolution), the angle α, c.f. Figure 5b, could be calculated by



(a) Flow chart of the calibration approach.

(b) Flow chart of the calibration algorithm.

Figure 4: Flow chart of calibration approach and its integrated calibration algorithm.



(a) Illustration of the angles and corners.

(b) Determine of α.

(c) Illustration of the value R in the hemisphere.

Figure 5: Variables of the calibration setup used for the algorithm.

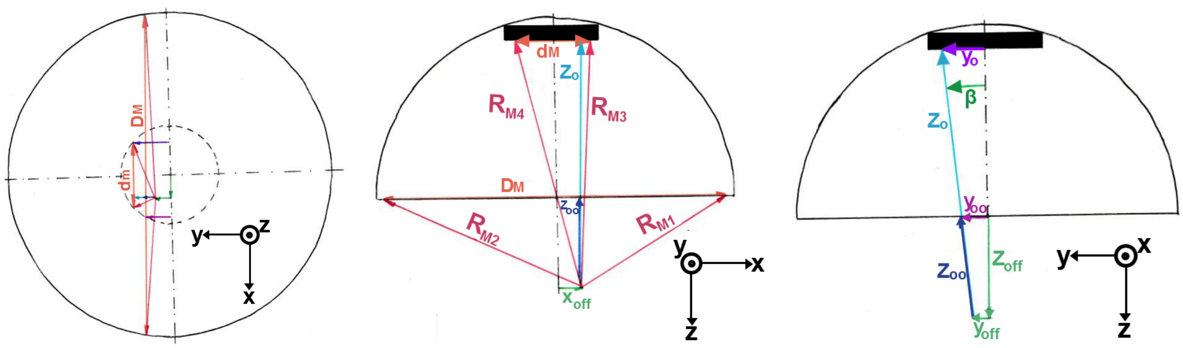
$$\alpha = (\Phi_{RM2} - \varphi_{RM2} - \varphi_{RM1})/2 \tag{2}$$

The scanner itself has an accuracy of $\pm 30mm$. It is assumed this value, further called R_{off}, for a single scan with invariant lightning and invariant surface conditions is constant. That is why each value can be described as

$$R_M = R_O + R_{off} \tag{3}$$

with R_M is the measured value and R_O is the true value. The function *rOffMinRangeLimitation()* calculates the minimal offset R_{off}.

Then, the range of β is determined in the function *betaRange()*. As described in the Section 3 the values of β and y_{off} assumed to be greater positive. That is why they have to be located in the front of the hemisphere, c.f. Figure 6a. In case of R_{off} is minimal β has to be maximal.



(a) Top view of hemisphere. (b) Front view of hemisphere. (c) Side view of hemisphere.

Figure 6: Illustration of the three views of the hemisphere.

Following, the function *rOffMaxRangeLimitation()* calculates the maximum value of R_{off} by respecting the minimal value of β.

In the end the function *positionOffsetRange()* determines the minimum and maximum values of x_{off}, y_{off}, and z_{off} by respecting the minimal and maximal value of R_{off} and β. This accelerates the following iterative process.

The third sub-step of the calibration algorithm (highlighted yellow), calculates the offset values α, β, x_{off}, y_{off}, and z_{off} in an iterative process. Therefore, the function *sphereRangeValues()* picks the surface values of the hemisphere between C_1 and C_3 and between C_2 and C_4. To fulfil

$$R_M = \sqrt{(x_M^2 + y_M^2 + z_M^2)}. \tag{4}$$

the offset values (x_{off}, y_{off}, z_{off}) are substracted from the ideal values (x, y, z). By assuming that R_M is equal to the real radius of the hemisphere, c.f. Figure 5c, equation

$$R_M = R = \sqrt{(x_M - x_{off})^2 + (y_M - y_{off})^2 + (z_M - z_{off})^2)} \tag{5}$$

is solved by an iterative loop to determine the offset values.

In the last step of the calibration approach (highlighted olive) all determined values are stored into a calibration file.

5. EXPERIMENTS AND RESULTS

For the first experiment an ideal model of the calibration setup was created in Matlab[8] and the algorithm was tested with predefined offsets values. In the second Experiment an hardware setup, described in Section 3, was built and the algorithm tested.

5.1 Experiment 1:

In the first experiment an ideal model of the hemisphere was used to test the algorithm with Matlab. Therefore, each offset value (further called $value_{off_{in}}$) was modified individual. The offset value was used to create a transformation Matrix T and the ideal hemisphere mode transformed. Then the resulting scan of the hemisphere was used as an input for a Matlab based version of the calibration algorithm. The input values ($value_{off_{in}}$) and the resulting calibration values ($value_{off_{out}}$) are illustrated in Table 1. Two different angular resolutions (0.25° and 0.125°) were used to illustrate the angular resolution dependency of the laser scanner. For β the biggest difference between input and output occurs. Further the resulting values show that a bigger offset value can be determined more precise.

Table 1: Predefined offset values with different angular resolution.

angular resolution:		0.25°		0.125°	
value	set	determined	|difference|	determined	|difference|
α [°]	2	1.88	0.13	1.94	0.06
	4	3.88	0.13	3.94	0.06
	6	5.88	0.13	5.94	0.06
	8	7.88	0.13	7.94	0.06
	10	9.88	0.13	9.94	0.06
β [°]	2	4.27	2.27	3.71	1.71
	4	5.07	1.07	4.69	0.69
	6	6.78	0.78	6.40	0.40
	8	8.28	0.28	8.18	0.18
	10	10.27	0.27	10.00	0.00
x_{off} $[mm]$	5	5.11	0.11	5.01	0.01
	10	10.03	0.03	9.93	0.07
	15	14.95	0.05	14.98	0.02
	20	20.13	0.13	20.02	0.02
y_{off} $[mm]$	5	13.19	8.19	10.72	5.72
	10	15.86	5.86	13.19	3.19
	15	18.10	3.10	17.02	2.02
	20	23.39	3.39	21.80	1.80

Table2 illustrates the results after modify multiple values ($value_{off_{in}}$). The values are similar to the values in Table 1.

Table 2: Multiple predefined offset values

Test 1	set	determined	|difference|	Test 2	set	determined	|difference|
α [°]	5	4.88	0.13	α [°]	10	7.31	2.69
β [°]	5	5.72	0.72	β [°]	10	9.49	0.51
x_{off} [mm]	5	5.06	0.06	x_{off} [mm]	20	18.92	1.08
y_{off} [mm]	5	5.01	0.01	y_{off} [mm]	20	20.97	0.97
Test 3	set	determined	|difference|	Test 4	set	determined	|difference|
α [°]	-10	-11.13	1.13	α [°]	-2	-0.13	1.88
β [°]	6	6.28	0.28	β [°]	10	9.62	0.38
x_{off} [mm]	5	4.83	0.17	x_{off} [mm]	-20	-20.05	0.05
y_{off} [mm]	10	9.92	0.08	y_{off} [mm]	15	15.78	0.78

5.2 Experiment 2:

The second experiment was applied on the hardware setup and the rotating laser scanner described in Section 3. First, calibration values (reference white column) of the 3D laser scanner are determined, see Table 3. Since there was no second reference system they were used as a reference for the tests. Next, each value (highlighted yellow) was modified individual . This results the "set" values (highlighted green). The determined calibration values (highlighted violet) are higher than suspected. This is due the fact that the distance accuracy of the laser scanner is high in compare to the diameter of the hemisphere.

Table 3: Calibration values determined with the hardware setup

Test:		$\alpha[°]$			$\beta[°]$		
	reference	set	determined	difference	set	determined	difference
α [°]	0,63	26,62	25,63	0,99	0,63	0,25	0,38
β [°]	0,06	0,06	0,00	0,06	5,96	0,06	5,9
x_{off} [mm]	-2,05	-2,05	-3,78	1,73	-2,05	2,76	-4,81
y_{off} [mm]	6,11	6,11	3,36	2,75	6,11	0,20	5,91
R_{off} [mm]	14,30	14,30	10,14	4,16	14,30	15,16	-0,86
Test:		x_{off} [mm]			y_{off} [mm]		
	reference	set	determined	difference	set	determined	difference
α [°]	0,63	0,63	-1,00	1,63	0,63	-0,50	1,13
β [°]	0,06	0,06	0,06	0	0,06	0,06	0
x_{off} [mm]	-2,05	5,95	8,33	-2,38	-2,05	0,00	-2,05
y_{off} [mm]	6,11	6,11	8,91	-2,8	19,61	8,53	11,08
R_{off} [mm]	14,30	14,30	13,97	0,33	14,30	13,82	0,48

6. CONCLUSIONS AND FUTURE WORK

This paper presents a calibration approach applies an calibration system for out of lab calibration. The approach determines the offset values for a rotating laser scanner. It uses an hemisphere, which is small and easy to transport, to determine the values in an iterative process.

The first experiment uses an implementation in Matlab to verify the approach. The results demonstrate that the determined values for bigger offsets are more precise than for small offsets. Besides, the angular accuracy of the laser scanner influences the accuracy of the determined values.

The second experiment applied the calibration system and the rotating Hokuyo. The results are not as accurate as the simulated values in Matlab. This is due the fact that the distance accuracy of the Hokuyo is high in compare to the diameter of the hemisphere. Nevertheless, the experiments verify the applied approach. Future work will concentrate to minimize the influences of the laser scanner accuracy.

REFERENCES

[1] Wulf, O. and Wagner, B., "Fast 3d-scanning methods for laser measurement systems," in [*in the 14th International Conference on Control Systems and Computer Science*], (2003).

[2] Qayyum, U., Martin, A., Kim, J., and Shim, D., "Omni-viser: 3d omni vision-laser scanner," (2012).

[3] Niola, V., Rossi, C., Savino, S., and Strano, S., "A method for the calibration of a 3-d laser scanner," *Robotics and Computer-Integrated Manufacturing* **27**(2), 479 – 484 (2011). Translational Research Where Engineering Meets Medicine.

[4] Martínez, J. L., Morales, J., Reina, A. J., Mandow, A., no Boter, A. P., and García-Cerezo, A., "Construction and calibration of a low-cost 3d laser scanner with 360 ; field of view for mobile robots," in [*2015 IEEE International Conference on Industrial Technology (ICIT)*], 149–154 (March 2015).

[5] Yang, Z., Wang, P., Li, X., and Sun, C., "Flexible calibration method for 3d laser scanner system," *Transactions of Tianjin University* **20**(1), 27–35 (2014).

[6] ROS, "Ros." http://www.ros.org. (13 April 2017).

[7] Hokuyo, "Hokuyo utm-30lx-ew." https://www.hokuyo-aut.jp. (13 April 2017).

[8] Matlab, "Matlab." https://de.mathworks.com/. (13 April 2017).