Effective Distinction Of Transparent And Specular Reflective Objects In Point Clouds Of A Multi-Echo Laser Scanner

Rainer Koch and Stefan May
Technische Hochschule Nuremberg
90489 Nuremberg, Germany
www.th-nuernberg.de
{Rainer.Koch, Stefan.May}@th-nuernberg.de

Andreas Nüchter
Informatics VII – Robotics and Telematics
Julius-Maximilians University Würzburg
Am Hubland, 97074 Würzburg, Germany
www.uni-wuerzburg.de
andreas.nuechter@uni-wuerzburg.de

Abstract—A favoured sensor for mapping is a 3D laser scanner since it allows a wide scanning range, precise measurements, and is usable indoor and outdoor. Hence, a mapping module delivers detailed and high resolution maps which makes it possible to navigate safely. Difficulties result from transparent and specular reflective objects which cause erroneous and dubious measurements. At such objects, based on the incident angle, measurements result from the object surface, an object behind the transparent surface, or an object mirrored with respect to the reflective surface. This paper describes an enhanced Pre-Filter-Module to distinguish between these cases. Two experiments demonstrate the usability and show that for single scans the identification of mentioned objects in 3D is possible. The first experiment was made in an empty room with a mirror. The second experiment was made in a stairway which contains a glass door. Further, results show that a discrimination between a specular reflective and a transparent object is possible. Especially for transparent objects the detected size is restricted to the incident angle. That is why future work concentrates on implementing a post-filter module. Gained experience shows that collecting the data of multiple scans and postprocess them as soon as the object was bypassed will improve the map.

Index Terms—3D laser scanning, Object, Identification, Segmentation, Error correction

I. INTRODUCTION

Mapping is an essential task for mobile robots. Based on the map, robots are capable to navigate through rough terrain, fulfil advanced manipulation tasks and inspection jobs. Therefore, a precise and detailed 3D map is required. Laser scanners are favoured sensors as they are suitable for indoor and outdoor applications, deliver precise measurements, and have a wide scanning range. Unfortunately, they are expensive and suffer when scanning transparent or specular reflective surfaces, e.g. glass, mirrors, or shiny metal, cf. Figure 1.

In case of specular reflective objects, e.g., mirrors, the laser beams get reflected and rerouted to an object located in front of the mirror. Therefore, the return measurements result from a mirrored object. Hence, the measured location of the object is wrong, cf. Figure 2. Besides, the mirroring surface is not mapped.

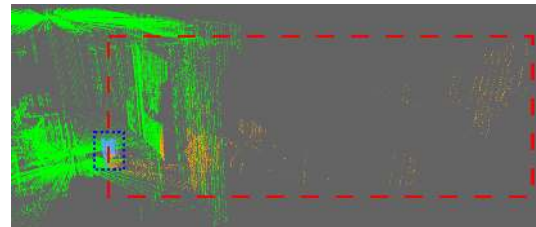Fig. 1: Robot equipped with the 3D-Hokuyo multi-echo laser scanner facing an unframed mirror.

Fig. 2: Resulting point cloud with erroneous data (marked by a red dashed line) caused by the mirror (marked by a blue dotted line).

In case of transparent objects, e.g. glass doors, the measurements of the laser beams partly result from the transparent surface and partly from the objects behind the surface, depending on the incident angle of the laser beam. It is understood that such erroneous measurements lead to a wrong map and therefore difficulties in navigation. If the object surface is not seen at all or seen only occasionally the map does not include it. That is why the robot might manoeuvre into the object and crashes.

Unfortunately, most environments include transparent or specular reflective objects like glass doors, windows, mirrors, or shiny metal surfaces. To prevent erroneous measurements from such objects there are two commonly used techniques. The first technique is to customize the environment and cover these objects. This is unwanted, since it changes the "real" environment. Besides, it is not always possible, e.g. when operating in rescue scenario. Because of this, many approaches employ the second technique – a sensor fusion[1], [2], [3], [4]. Here, a second sensor principle, like ultrasonic arrays, is fused with the laser scanner to respect these

troublesome objects. Ultrasonic arrays are capable to detect specular reflective and transparent objects but they suffer from imprecise measurements and low measurement range. Besides, it is necessary to deal with two sensor units. This results in extra costs, an additional source of hardware failure, and requires a calibration. Hence, this method is unwanted as well.

Contrary, this paper presents the 3D-Reflection-Pre-Filter, an enhanced version of the 3D-Mirror-Pre-Filter [5], that relies only on a multi-echo laser scanner. It is online applicable in order to pre-filter laser point clouds to reduce above mentioned effects. Section II outlines related work. Following, Section III describes the 3D-Reflection-Pre-Filter Approach and in Section IV two experiments demonstrate the applicability to environments with reflective and transparent objects. Finally, Section V summarizes the results and gives an outlook for future work.

II. RELATED WORK

For stationary systems several approaches are presented to eliminate transparent and specular reflective influences at laser scanners. In this case, it is possible to customize the environment since it is known. Therefore, stationary systems represent a special case. This section concentrates on approaches dealing with transparent and specular reflective object influences for mobile robots. Most approaches cover the 2D case. They are still important as this represents a simplification of the 3D case.

A sensor fusion of a laser scanner and an ultrasonic sensor is a common method to prevent robots to crash into transparent or specular reflective objects. In a first step, Yang et al. [1] used such a fusion for a 2D mapping approach to avoid the need to cover surfaces. Based on the assumption that reflective objects are flat and framed, the data from the two sensors are compared with respect to consistency. The approach detects and tracks mirrors online, while resulting errors are recalculated only offline.

In a second step, Yang et al. [6] extended their algorithm to identify mirror images. The advanced mirror detection does not rely on an ultrasonic sensor anymore, but rather only on a laser scanner. The approach assumes each gap in the wall of a laser scan to be a mirror. Once such a gap is detected, the space behind the candidate is analysed for a mirrored image, e.g. the search for similarity between both sides of the opening. In case of a positive result the opening is marked as a mirror and the points behind are erased. A drawback of this method is that objects with symmetry with respect to a line might be identified wrongly. Besides, it requires much processing power to search the gaps.

In contrast, the online applicable 2D approach, implemented by Forster et al. [7], searches for specific angles. At these angles reflections can be identified on the returning intensity of the laser scanner. Therefore, the algorithm tracks a subset of these angles. Based on the laser beam's intensity the algorithm eliminates the points. This also works for transparent objects, but fails if a diffuse reflective object is placed directly behind a transparent object. Furthermore, Forster et al. had no intention to distinguish between specular reflective or transparent objects.

A 3D approach which recognised framed mirrors with a predefined size was presented by Käshammer et al. [8]. The algorithm uses a point cloud to generate a panorama range image and searches for jumping edges. In case of a positive search it extracts the contour. Based on size and shape objects are verified. Therefore, this approach only locates squared mirrors with a predefined size. Transparent or other reflective objects are not considered. Points behind the plane are erased by the approach. There is no effort to back-project them to their original location and use them to improve mapping.

Tatogulu et al. [9] used the most suitable illumination model to modulate the surface of 3D models. The approach uses Lambertian diffuse reflection models, Blinn-Phong models [10], Gaussian models [11], and Beckmann specular reflection models [12] and matches them to the data of the scanned surface to identify the characteristics. The approach works effectively for diffuse surfaces, but does not cover specular reflections.

Koch et al. [13] presented a 2D-Mirror-Detector-Approach based on an Hokuyo UTM-30LX-EW multi-echo laser scanner. It comprises a pre-filter module, a post-filter module, and two mapping modules. The pre-filter module as well as the first mapping module apply laser scans on the fly. The Pre-Filter reduces identified points behind a transparent or specular reflective object in current scans. The first mapping stage builds a map with the pre-filtered scans. The identification of transparent and specular reflective objects can not be guaranteed since the measurements depend on the incident angle of the laser beam. Therefore, the Post-Filter creates a history of these pre-filtered scans. It evaluates all scans, triggered as soon as the reflective or transparent surface has been passed. Similar to the Pre-Filter each point of the scan gets evaluated, if it is a regular scan point, a point on the surface, or a point behind the surface. Finally, the second mapping module builds a refined map based on the evaluated scans. The algorithm identifies planar unframed transparent and specular reflective objects independent of their size. Points behind the objects are identified but not used.

In further research Koch et al. [14] extended their 2D algorithm. The Reflection-Classifier-Approach distinguishes between transparent and specular reflective objects. Thus, the intensity values of identified objects are analysed. Based on the type of object scan points on the object plane and behind the object plane are masked. Further points behind a reflective object are excluded while points behind a transparent object remain in the laser scan. The extended algorithm uses only

one TSD mapping module which gets updated by the results of the Post-Filter.

A 3D-Mirror-Pre-Filter-Approach applicable to multi-echo laser scanners in order to identify and filter transparent and specular reflective objects was presented by Koch et al. [5]. In contrast to above mentioned 2D approaches this online running Pre-Filter identifies and filters transparent and specular reflective objects in 3D point clouds. It recognises frameless and free-standing objects regardless their size. Furthermore, it identifies transparent and specular reflective objects. Hence, it back-projects points behind the object plane and fits them to the "valid" points based on an ICP algorithm. This method suffers from two drawbacks. First, locations with symmetry might result in a wrong identification. Second, the function assumes that the mirrored object is included in the "valid" point cloud as well. If only the mirrored object is seen the function will result in a wrong identification.

This paper presents the 3D-Mirror-Pre-Filter which is an enhanced version of the 3D-Mirror-Pre-Filter of Koch et al. [5]. In addition, it uses intensity values to distinguish between transparent and specular reflective objects. It is online applicable in order to recognise and pre-filter unframed planar objects in multi-echo laser scans.

III. APPROACH

Algorithm 1 shows the sequence of the 3D-Mirror-Pre-Filter-node. It is subsequently processed as soon as it receives a point cloud tuple $\vec{P}$ from the Hokuyo3D-node. First, erroneous distance values, points around the robot, and outliers get filtered out. Subsequently, the distance values of Echo 1 and Echo 2 are compared to identify discontinuities. In case of a positive result these points get extracted and the approach searches for planar square objects. Identified planes are further analysed to distinguish between transparent and specular reflective objects. Finally, the resulting point cloud tuple $\vec{P}_{out}$ is broadcasted to a mapping-node.

This paper presents an enhanced distinction between specular reflective and transparent objects. Thus, this section concentrates on the subfunction *separateObject()*, shown in Algorithm 2, in which the distinction proceeds.

For $n_{affected}$ greater than n_{object} the function *separateObjects()* starts subsequently to search for planar square objects. First, the subfunction *identifyObjects()* extracts planes based on the Point-Cloud-Library Sample-Consensus-Segmentation (PCL-SAC-Segmentation)[15]. For each identified plane its corners are determined with the Point-Cloud-Library-Feature-Extractor (PCL-feature-extractor). It is based on a principal component analysis (pca-analysis) and returns the four corners as well as the dimensions of a plane. Second, the subfunction *analyzeObjectType()* uses the extracted planes, the intensities, and the masked point cloud to distinguish between transparent or specular reflective objects shown in Algorithm 3. Finally, all points are masked according to their location and object type.

Algorithm 1 3D-Pre-Filter

Input: $\vec{P}$:
 $\vec{P}$ includes the scan clouds with points of Echo 1 and Echo 2 their corresponding values. Each point p_i has a distance d_i, an angle α_i, xyz-coordinates, a normal-vector $\vec{n}_{0_i}$, an intensity I_i, and a mask m_i.

Output: $\vec{P}_{valid,surface,affected}$:
 The message $\vec{P}_{valid}$ includes the valid scan points with their corresponding attributes and scan points on the surface (d, α, xyz, $\vec{n}_0$, I, and m). $\vec{P}_{surface}$ includes the scan points, located on the surface of the transparent or specular reflective object, with their corresponding values. $\vec{P}_{affected}$ includes the scan points, located behind the surface of the transparent or specular reflective object, with their corresponding values.

1: **procedure** 3D-MIRROR-PRE-FILTER
2:     $\vec{P}_{in} \leftarrow$ receiveScanTuple($\vec{P}$)
3:     $\vec{P}_1 \leftarrow$ distanceThresFilter($\vec{P}_{in}$, d_{min}, d_{max})
4:     $\vec{P}_2 \leftarrow$ boxFilter($\vec{P}_1$, $d_{x,y,z_{min}}$, $d_{x,y,z_{max}}$)
5:     $\vec{P}_3 \leftarrow$ outlierFilter($\vec{P}_2$, n_{inlier}, r, "*unchecked*")
6:     $\vec{P}_4, n_{affected} \leftarrow$ identifyReflections($\vec{P}_3$, n_{inlier}, r, "*errorSurface*")
7:     **if** ($n_{affected} \geq n_{object}$) **then**
8:         $\vec{P}_5 \leftarrow$ outlierFilter($\vec{P}_4$, "*errorSurface*")
9:         $\vec{o}, \leftarrow$ separateObject($\vec{P}_5$, $\vec{V}_{distinction}$, n_{object})
     ▷ function described in Algorithm 2
10:         $\vec{P}_{out} \leftarrow$ cleanScanTuple($\vec{P}_5$, $\vec{o}$, d_{thres_plane}, $d_{thres\_visionCone}$)
11:     **else**
12:         $\vec{P}_{out} = \vec{P}_3$
13:     sendScanTuple($\vec{P}_{out}$)

Algorithm 2 3D-Pre-Filter - separateObject()

Input: $\vec{P}_5$, $\vec{V}_{distinction}$, n_{object}:
 $\vec{P}_5$ is the scan cloud tuple resulting from the function *outlierFilter()*. n_{object} is the minimal amount of points to identify an object. $\vec{V}_{distinction}$ is a vector containing all variables for the object distinction.

Output: $\vec{o}$:
 $\vec{o}$ is a vector of objects with its properties (xyz-coordinates of corners, type of object, width, length, plane function parameters).

1: **function** SEPARATEOBJECT()
2:     **while** ($n_{affected} \geq n_{object}$) **do**
3:         $\vec{o} \leftarrow$ indentifyObjects($\vec{P}_5$, d_{thres_plane}, n_{object})
     ▷ function described in Algorithm 3
4:         $m_{objectType} \leftarrow$ analyzeObjectType($\vec{P}_5$, $\vec{V}_{distinction}$,)
5:         mask($\vec{P}_5$)

The subfunction *analyzeObjectType()* determines the type of the object based on three different methods each of which is processed in a subfunction (*meanIntensFactorCheck()*, *transformationCheck()*, and *intensVariation()*). Finally the function *evaluateResults()* compares the results, determines the final object type, and returns the result to the function *separateObject()*.

Algorithm 3 3D-Pre-Filter - identifyObjects()

Input: $\vec{P}_5$, $\vec{V}_{\text{distinction}}$, n_{object}:
$\vec{P}_5$ is the scan cloud tuple. $\vec{V}_{\text{distinction}}$, is a vector containing all variables for the object distinction.

Output: $m_{\text{objectType}}$:
$m_{\text{objectType}}$ is the type of object.

1: **function** IDENTIFYOBJECTS()
2:     $m_{\text{IntensFact}} \leftarrow$ meanIntensFactorCheck($\vec{P}_5$, $\vec{V}_{\text{distinction}}$)
3:     $m_{\text{Transf}} \leftarrow$ transformationCheck($\vec{P}_5$, $\vec{V}_{\text{distinction}}$)
4:     $m_{\text{intensVariation}} \leftarrow$ intensVariation($\vec{P}_5$, $\vec{V}_{\text{distinction}}$)
5:     $m_{\text{objectType}} \leftarrow$ evaluateResults($m_{\text{IntensFact}}$, m_{Transf}, $m_{\text{intensVariation}}$, $m_{\text{PhongCurve}}$)

A. Determination of the mean intensity factor

The mean intensity factor describes the relation of intensity values on the plane and intensity values behind the plane ("error"). For transparent objects most of the intensity values behind the plane are grater than the intensity values on the plane, cf. Figure 3. For specular reflective objects they are similar, cf. Figure 4. Therefore, an arithmetic mean intensity $\hat{I}_{Echo\_1}$ and $\hat{I}_{Echo\_2}$ is calculated by

$$\hat{I}_{Echo\_1} = \frac{\sum_{i=1}^{N} I_{Echo\_1,i}}{N} \quad \text{and} \quad \hat{I}_{Echo\_2} = \frac{\sum_{i=1}^{N} I_{Echo\_2,i}}{N} \tag{1}$$

with the intensity $I_{Echo\_1,i}$ and $I_{Echo\_1,i}$ as well as the amount of intensity values N.

Following the factor $f_{material}$ is calculated by

$$f_{material} = \frac{\hat{I}_{Echo\_2}}{\hat{I}_{Echo\_1}} \tag{2}$$

Based on a threshold $thres\_IntensFactor$ the object is rated to be transparent or specular reflective and the function returns $m_{IntensFact}$.

B. Evaluation of a valid transformation

The function *transformationCheck()* examines if there is a symmetry with respect to the identified object. Therefore, it uses the points located behind the identified object plane ("affected") and back-projects them with respect to the plane. Following, the points $D = \{d_i | i = 1..N_D\}$ and the "valid" points $M = \{d_i | i = 1..N_M\}$ are matched by an Iterartive-Closest-Point-algorithm (ICP-algorithm). In case of a positive result, cf. Figure 5, the translation T and the rotation R of the
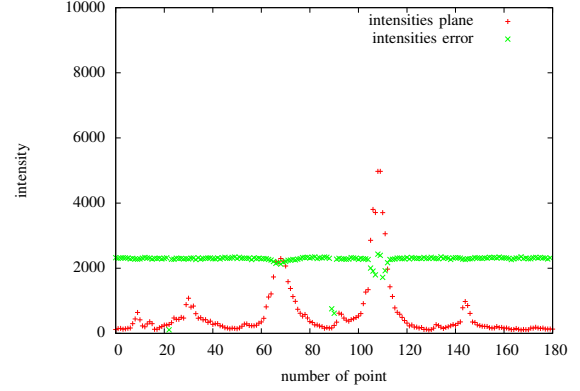

Fig. 3: Intensity values of an transparent object. Green are intensity values of points behind the transparent plane ("error") and red are intensity values of points on the transparent plane.
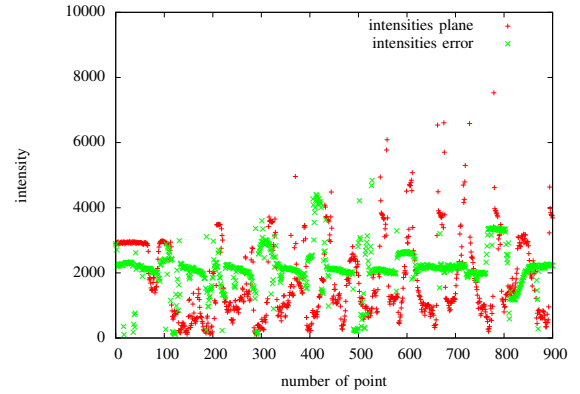

Fig. 4: Intensity values of an specular reflective object: Green are intensity values of points behind the mirror plane ("error") and red are intensity values of points on the mirror plane. For a better illustration of the differences between the intensity values on the plane ("mirror") and behind the plane ("error") all values above 10000 were cut off. The maximum intensity value on the plane ("mirror") was 42000.

resulting transformation matrix $Trans$. The resulting vector $\vec{Tr} = (\Phi, \Theta, \Psi, \Delta x, \Delta y, \Delta z)$ with its angles and distances is compared to a threshold. In case of small displacements it is assumed that the plane has reflective properties. Finally, the function returns the result m_{Transf} and stores the Transformation matrix with the back-projected points.

This function suffers from two drawbacks. First, locations with symmetry might result in a wrong identification. Second, the function assumes that the mirrored object is included in the "valid" point cloud as well. If only the mirrored object is seen the function will result in a wrong identification.

C. Verification of a consistent intensity value

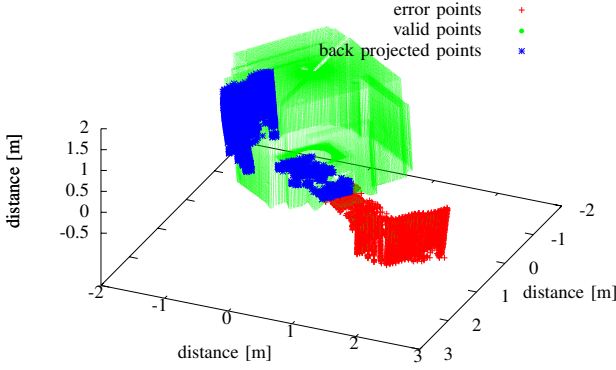Intensity values of points behind a transparent and a specular reflective object vary differently. Therefore, the

Fig. 5: Back-projected and ICP fitted points. There are still "valid" points in the area of the "error" points. That is because the function *cleanScanTuple()*, see Algorithm 1, was not processed yet.

function *intensVariation()* analyses the curve of the intensities of such points. Figure 6 shows the intensity values behind a transparent object which remain almost constant. In contrast, the intensity values of a mirrored object vary as shown in Figure 7.
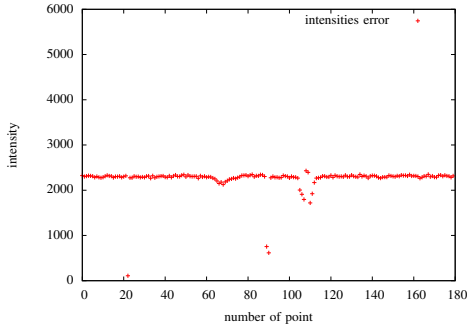

Fig. 6: Intensity values of points behind a transparent object.

To analyse the intensities the function calculates the median intensity $\tilde{int}$

$$\tilde{int} = \begin{cases} int_{\frac{N_{error}+1}{2}} & \text{for } N_{error} \text{ uneven} \\ \frac{1}{2}(int_{\frac{N_{error}}{2}} + int_{\frac{N_{error}+1}{2}}), & \text{for } N_{error} \text{ even} \end{cases} \quad (3)$$

with N_{error} is the amount of points.

Further it identifies the object type based on the amount of intensity values m_{close} which are close to the median intensity int. Therefore, it uses the factor $thres\_similar$ to identify if the point is close as described in Algorithm 4. Finally, the function returns the object type $m_{intensVariation}$.

D. Evaluation of final object type

Function *evaluateResults()* evaluates the results of the object type distinction functions (*meanIntensFactorCheck()*,
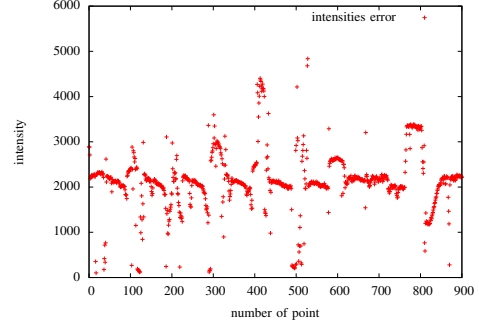

Fig. 7: Intensity values of mirrored points. For a better illustration intensity values above 6000 were cut off. The maximum intensity value of an mirrored point was 13000.

Algorithm 4 Identification of object type based on the amount of intensity values.

1: **for** $i = 0; i < N_{error}; i++$ **do**
2:     **if** $|intensity - \tilde{int}| < thres\_similar * \tilde{int}$ **then**
3:         $m_{close}++$;
4: **if** $(thres\_vary * N_{error}) < (m_{close}/N_{error})$ **then**
5:     $m_{intensVariation} \leftarrow$ mirror
6: **else**
7:     $m_{intensVariation} \leftarrow$ transparent

---

transformationCheck(), and *intensVariation()*) to determine the type of object.

IV. EXPERIMENTS AND RESULTS

This chapter consists of two experiments to demonstrate the usability of the 3D-Reflection-Pre-Filter Approach. Experiment 1 uses an empty room containing a mirror, Experiment 2 uses a stairway with a glass door. The mirror is unframed, planar, and square. The 3D scanner, a rotating Hokuyo UTM-30LX-EW, was in stationary mode. Hence, the robot was located at the middle of the room (about $2\ m$ from the object of interest). The scanner was rotating with $10rpm$ which results in a point cloud of 115000 scan points. Because of stationary mode of the scanner no mapping module was running.

(a) Setup of Experiment 1.   (b) Setup of Experiment 2.

Fig. 8: Scene of Experiment 1 and Experiment 2.

1) Experiment 1: Empty room with a mirror: The mirror in Experiment 1 standing on the floor and leaning on the wall with a minimal angle, c.f. Figure 8. The size of the mirror as well as the determined size are illustrated in TableIV-.2. Besides the table shows the result of the function *intensVariation()* and the range of the intensity values of Echo 2 of the median intensity.

The ICP determined the transformation matrix between the mirrored points and the rest of the point cloud. The vector $\vec{Tr}$ results as:

$\vec{Tr} = (4.8°, 0.4°, -0.4°, 0.01\ m, 0.01\ m, 0.10\ m).$

2) Experiment 2: Stairway with a glass door: The stairway in Experiment 2 consists of two glass doors, c.f. Figure 8. Due to the strong dependency of the laser beams incident angle on the surface only a small area. The detected area is around the laser beam hitting the surface perpendicular. That is why the robot will not crash into the glass door. Nonetheless, it is necessary to fuse multiple scans, from different locations to determine the complete glass area. Therefore, a Post-Filter module is required. Regarding the object type distinction the results of all three methods are correct. Similar to Experiment 1 the results are illustrated in Table IV-.2.

The ICP determined the transformation matrix between the mirrored points and the rest of the point cloud. The vector $\vec{Tr}$ results as:

$\vec{Tr} = (-15.0°, -13.7°, 19.4°, -0.86\ m, 0.62\ m, -0.07\ m).$

Experiment	mirror	glass
real size [cm]	60×40	2 doors 88×198
measured size [cm]	58.4×31.6	24.2×20.9
$f_{material}$	1.78	0.00073
intensVariation()	38%	51%
range of Echo 2	$\pm 20\%$	$\pm 20\%$

V. Conclusions and Future Work

This paper presents an enhanced 3D-Reflection-Pre-Filter Approach to pre-process point clouds of a 3D multi-echo laser scanner to reduce transparent and specular reflective influences. The Pre-Filter searches the point clouds for mismatches in distance between the corresponding echoes. On occurrence, these points get extracted, a function searches for planar objects, and the object corners are extracted. Afterwards, the type of object is determined by three methods. The first method analyses the intensity values of Echo 1 versus Echo 2. The second method creates a back-transformation of "affected" points w.r.t. the object plane and fits them by an ICP to determine a valid mirrored object. The last function evaluates the variation of the intensities of the "affected" points. Based on the results the identified object plane and the points behind the plane are classified, masked, and broadcasted.

Two experiments demonstrate the usability of this object distinction for an enhanced 3D-Reflection-Pre-Filter. The first experiment contains a mirror in an empty room. The second experiment contains a glass door located in a stairway. In both experiments the object type was determined correctly. The results show that it is possible for the Pre-Filter to distinguish between transparent and specular reflective objects in 3D. In contrast, this was not possible in 2D. The experiments also show that for transparent objects the dependency of the incident angle of the laser beam is higher than for specular reflective objects. Therefore, their visibility is limited. To assure a full identification of their dimensions multiple scans are required. That is why future work will concentrate in implementing a Post-Filter module.

References

[1] Yang, S.W., Wang, C.C.: Dealing with laser scanner failure: Mirrors and windows. In: IEEE International Conference on Robotics and Automation (ICRA), Pasadena, California (May 2008) 3009–3015

[2] Ali, H., Ahmed, B., Paar, G.: Robust window detection from 3d laser scanner data. 2008 Congress on Image and Signal Processing **1** (2008) 115–118

[3] Quintana, B., Prieto, S.A., Adn, A., Bosch, F.: Door detection in 3d colored laser scans for autonomous indoor navigation. In: 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN). (Oct 2016) 1–8

[4] Diosi, A., Kleeman, L.: Advanced sonar and laser range finder fusion for simultaneous localization and mapping. In: Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems. (2004)

[5] Koch, R., May, S., Nüchter, A.: Detection and purging of specular reflective and transparent object influences in 3d range measurements. In: 3D-Arch - 3D Virtual Reconstruction and Visualization of Complex Architetures and Scenarios (3D-Arch), 2016 International Workshop. (03/17 2017)

[6] Yang, S.W., Wang, C.C.: On solving mirror reflection in lidar sensing. Mechatronics, IEEE/ASME Transactions on **16**(2) (2011) 255–265

[7] Foster, P., Sun, Z., Park, J.J., Kuipers, B.: Visagge: Visible angle grid for glass environments. In: Robotics and Automation (ICRA), 2013 IEEE International Conference on. (2013) 2213–2220

[8] Käshammer, P.F., Nüchter, A.: Mirror identification and correction of 3d point clouds. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences **XL-5/W4** (2015) 109–114

[9] Tatoglu, A., Pochiraju, K.: Point cloud segmentation with lidar reflection intensity behavior. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on. (2012) 786–790

[10] Blinn, J.F.: Models of light reflection for computer synthesized pictures. In: Proceedings of the 4th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '77, New York, NY, USA, ACM (1977) 192–198

[11] Torrance, K.E., Sparrow, E.M.: Theory for off-specular reflection from roughened surfaces. J. Opt. Soc. Am. **57**(9) (Sep 1967) 1105–1112

[12] Beckmann, P., Spizzichino, A.: The scattering of electromagnetic waves from rough surfaces. Pergamon Press (1963)

[13] Koch, R., May, S., Koch, P., Kühn, M., Nüchter, A.: Detection of specular reflections in range measurements for faultless robotic slam. In Reis, L.P., Moreira, A.P., Lima, P.U., Montano, L., Muoz-Martinez, V., eds.: Robot 2015: Second Iberian Robotics Conference. Volume 417 of Advances in Intelligent Systems and Computing. Springer International Publishing (2016) 133–145

[14] Koch, R., May, S., Murmann, P., Nchter, A.: Identification of transparent and specular reflective material in laser scans to discriminate affected measurements for faultless robotic {SLAM}. Robotics and Autonomous Systems **87** (2017) 296 – 312

[15] PCL: Pcl. http://pointclouds.org (09 2016) (20 Sep. 2016).