Transparent Object Detection and Reconstruction on a Mobile Platform

Ulrich Klank, Daniel Carton, Michael Beetz
Intelligent Autonomous Systems, Technische Universität München,
Boltzmannstr. 3, Garching bei München, 85748, Germany
klank@cs.tum.edu

***Abstract*— In this paper we propose a novel approach to detect and reconstruct transparent objects. This approach makes use of the fact that many transparent objects, especially the ones consisting of usual glass, absorb light in certain wavelengths [1]. Given a controlled illumination, this absorbtion is measurable in the intensity response by comparison to the background. We show the usage of a standard infrared emitter and the intensity sensor of a time of flight (ToF) camera to reconstruct the structure, which can not be measured by the usual 3D measurements of the ToF camera, given we have a second view point.**

We take advantage of this fact by deriving this internal sensory contradiction from two ToF images and reconstruct an approximated surface of the original transparent object. Therefor we are using a perspectively invariant matching in the intensity channels from the first to the second view of initially acquired candidates. For each matched pixel in the first view a 3D movement can be predicted given their original 3D measurement and the known distance to the second camera position. If their line of sight did not pass a transparent object or suffered any other major defect, this prediction will highly correspond to the actual measured 3D points of the second view. Otherwise, if a detectable error occurs, we approximate a more exact point to point matching technique and reconstruct the original shape by triangulating the points in the stereo setup. We tested our approach using a mobile platform with one Swissranger SR4k. As this platform is mobile, we were able to create a stereo setup by moving it. Our results show a detection of transparent objects on tables while simultaneously identifying opaque objects that also existed in the test setup. The viability of our results is demonstrated by a successful automated manipulation of the respective transparent object.

Fig. 1. The robot grasps a transparent object, which was reconstructed using a SR4k camera.

I. Introduction

For the scenario of a robot in a domestic environment solutions for as well object detection as reconstruction include often the usage of ToF sensors and stereo cameras. In the state of the art literature, we find approaches which allow the avoidance on the one hand and manipulation on the other hand of nearly all kind of objects. Most of those methods are meant to be applied to a special task, and a globally applicable solution can be most probably only achieved by a combination of a set of those methods.

We want discuss in this paper the task of detection and reconstruction of transparent objects, for which the state of the art lacks a robust solution [2]. Especially, current sensors and the respective methods have problems either to detect the objects or to detect their transparency.

But in domestic environments the recognition of such objects for robots that perform manipulation tasks is particularly interesting. Glass objects are quite common in such environments in the form of drinking glasses, vases or bottles, which is reasoned in their transparency allowing to see the content directly. Without the robust detection of any kind of object, a robot would fail on any everyday task in a household including the manipulation or recognition of such objects. Apart from the problem that a transparent object could not be grasped, the missing recognition leads to more difficulties. A collision for example would be inevitable if the robot tried to manipulate a known object but would be incapable of recognizing the transparent object located aside.

The challenges that transparency implies to common sensors and algorithms seem simple [3]. Nevertheless, no general solution could be found, yet [4]. Laser beams, e.g. emitted by LIDAR-Sensors, are usually partly reflected and refracted several times before they hit any surface, which leads to false or no 3D information at all [1], [5]. For normal camera systems transparent objects are almost invisible except for specularities that can be used to deduce shape information if well defined preconditions are met [6], [7]. A 3D reconstruction using common stereovision approaches is difficult due to the lack of stable features on transparent objects [8], [9]. Tests with structured light [10] that were carried out at the beginning of our work showed only poor results on objects like drinking glasses or plastic bottles. On the other hand, some of the reconstructions were good enough to fit a 3D shape model into the point cloud. But the success of these approaches were still heavily depending on various factors such as lightening environment and object

shape. Decent results could only be obtained under very constrained conditions. Therefore, recent studies concentrate on the development of algorithms which consider the special properties of translucent objects [2], [8]. Additionally, research into various sensors is being done, to maybe obtain more useful measurements [11], [12].

Our approach provides a foundation to expand available object recognition systems by transparency, leading to more robustness in the robots environment perception. Apart from that, we took the household-robot as the use case for the actual system. Accordingly, our method is supposed to enable a robot not only to detect but also to manipulate transparent objects, which requires a reconstruction. Our key contributions are:

1) A robust segmentation for various background types (bright uniform/wood, dark uniform)
2) A specialized matching with a series of validity tests to distinguish between opaque and transparent objects
3) A probabilistic reconstruction of possibly occupied area by a transparent object
4) A robust grasping mechanism based on [13]

The structure of this paper is as follows: The following section explains the architecture of our system and its integration into the robotic system. Section II presents the Problem and the effects which are observed in the sensor reading of a SR4k for transparent objects. The following sections describe the method for Segmentation in section III and the Matching procedure in section IV followed by the description of the special behavior of transparent objects in SR4000 point clouds in Section V which also depicts the related check. Subsequently we discuss the reconstruction of the 3D points in Section VI. The results of our test on our hardware setup follow in Section VII. Lastly we draw our conclusions in Section VIII.

A. System architecture

As for development purposes we mainly used the MVTec HALCON image processing library and embedded our code into ROS [14]. The architecture of our software is shown in Figure 2.

The inputs we use for our method is the SR4000 point cloud data which is also converted into a distance and an intensity map of the size 176x144. The system itself is divided into two steps with the robot's movement in between. In the first part the data from the SwissRanger is read, the obtained intensity images are optimized in their contrasts [15] and are segmented by extracting darker areas with an enhanced thresholding [16]. This procedure yields possible candidate areas for transparent objects. These candidates support not only 2D intensities but also 3D information corresponding to every image point. The defined candidate areas are then returned to the operating system together with the corresponding distance and intensity image as well as the 3D point cloud data.

Secondly the robot performs a movement within a certain range which provides pose parameters that can be used for a 3D point transformation between the first and the second

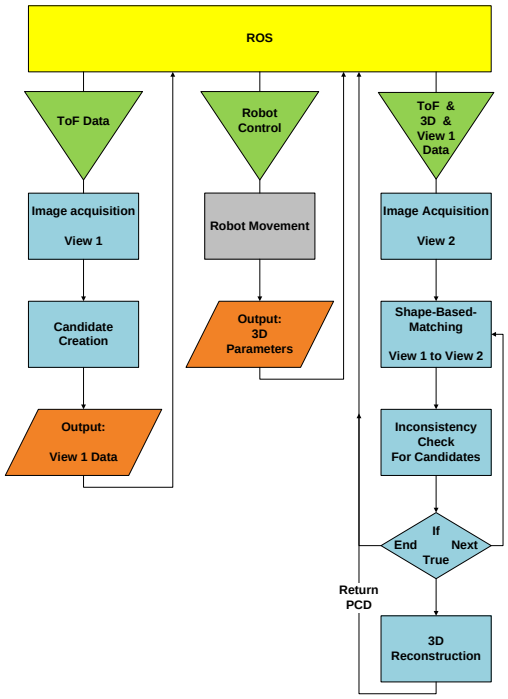


Fig. 2. System Architecture

view on the scene. These parameters are acquired from the operating system ROS by comparing the two robot positions for each view which originate from an AMCL driven self-localisation supported by two laser sensors. To ensure that the candidates remain in the field of view of the ToF camera, every candidate has an approximated world coordinate pose attached to it such that the platform can focus this pose and run the last step.

Now a second view is generated as well as a second segmentation with the same procedure as the first one. The inputs are then complemented by the 3D transformation data and the ToF data from the first view. The method then processes every candidate and checks whether it has the characteristic of a transparent objects when comparing the two views. In order to perform this check, we first establish 2D image correspondences by applying a perspectively invariant matching in the intensity channels [17] for the respective candidate. In the next step the algorithm ascertains whether a candidate is a transparent object or not by checking for inconsistencies in its 2D and 3D points when comparing the two views. If the check is positive, a 3D reconstruction is carried out and the new 3D points are transformed into a suitable form for later grasping or path planning algorithms. More detailed information about each of the mentioned steps will be provided in the following sections.

II. Problem Statement

To begin with, the available data will be described briefly. As shown in Figure 3(a) a transparent object can be seen in a usual camera image if the light environment fulfills certain

conditions. Here we have no direct light from above and a distinct amount of ambient light coming from our ToF sensor. Figure 3(b) shows the same camera image where bright light is illuminating the scene from above. The translucent object is barely recognizable with human eyes. As a result, common stereovision approaches fail to even perceive the object.

(a) Barely illuminated by ambient light from the side and Sr4k light.

(b) Highly illuminated with bright light coming from above.

Fig. 3. Camera Images of the same scene with different illuminations.

Accordingly a sensor to perceive those objects should be mostly invariant to the light environment. Here we propose the use of a SR4k ToF camera, which is mostly invariant to ambient light and therefore fulfills our demands. When a transparent object is positioned in the view of a ToF-camera an absorbtion of the IR-light is measurable in the intensity channel. Yet it provides improper 3D data for non-opaque objects. Figure 4(a) and 4(b) show two point clouds with the intensity values associated to each point that was generated by the SR4000 while observing the same scenes with changing light environment described above. The scene shows a table with several opaque objects and two transparent objects in the middle and on the left.

(a) A scarcely illuminated scene.

(b) A highly illuminated scene.

Fig. 4. ToF point cloud colored with intensities showing different, yet not distinguishable, illuminations

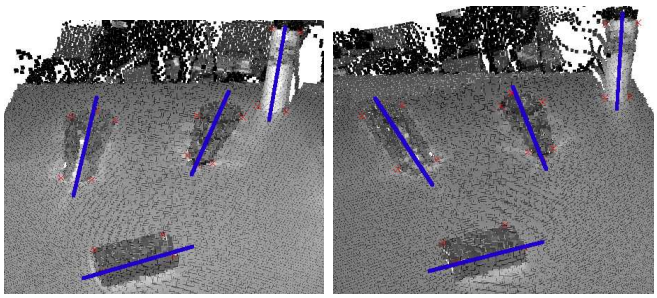
These views seem to show a measurement of the transparent objects, which are marked with red crosses. As one can see, the glass's shape is perceivable as a darker area on the table. The reason for this is that the infrared light emitted by the SR4000 camera is partly absorbed by the glass [1]. The surrounding table surface on the other hand yields much higher intensities because the infrared light is reflected strongly. A very striking fact can be seen when observing the point clouds from a different angle. The 3D points of the glass are as flat as the table's surface on which the object was placed. This effect is illustrated in the Figures 5(a) and 5(b). Moving the camera leads to a shadow-like behaviour of the transparent objects, what means that the wrongly estimated

(a) from front left

(b) from front right

Fig. 5. The red crosses mark transparent object that are as flat as the surface because the ToF camera can not measure the shape correctly

3D positions fall away from the light source that illuminates the scene onto the next surface which is the table in our setup. According to the results in [4] we suspect that one part of the infrared light is reflected because the intensity images contain specularities as well. After the light has propagated through the object it supposedly hits the first opaque surface, which is the table, and is reflected back to the camera. Since the light was mainly reflected by the table, the distance estimation of the ToF-camera is in accordance to the surface behind the glass, explaining the flattened 3D estimation. The following set of images in Figures 6(a) and 6(b) depict the shadow-like behaviour. The robot moves to the right and the point clouds of the transparent objects move into the opposed direction.



(a) ToF Camera in first position

(b) ToF Camera in second position

Fig. 6. Observed from a fixed point of view the transparent objects (marked with red cross) are moving like a shadow over the surface when the ToF camera is moved. The blue axes indicate the shadow-like movement.

All the objects in the Figures 6(a) and 6(b) are marked with red crosses to illustrate their alignment. The point of view is stable as well as the table on which the objects are placed. The only movement is performed by our platform such that the ToF-camera position changes. The blue axes indicate that the opaque objects on the bottom and on the right keep their alignment whereas the two transparent objects in the middle and middle left of the picture perform a twist to the left. The aforementioned effects are taken into consideration by our method to distinguish between opaque and non-opaque objects. If a transparent object is found, the real 3D shape has to be recovered to enable our platform to manipulate it.

III. Segmentation

In order to avoid calculating a complete image to image correspondence we apply a segmentation as a preprocessing step. A complete image to image correspondence would lead to too many ambiguities in the matching since the features observed in the SR4k are usually weak. Fortunately, transparent objects absorb enough light of the ToF camera's emitted

flash, that they appear darker than their background except from some specularities which are significantly brighter. Objects with a proper 3D response in the camera could be segmented using geometric cues [18] but this method is not functional for translucent obstacles. Accordingly our method performs a segmentation in 2D in order to find areas with possible inaccuracies that have to be restored by a reconstruction procedure to support the reliability of platform functions such as grasp-position planning.

A. Intensity-based initial Segmentation

In order to generate candidate regions that our method can focus on, the lowered intensities have to be found. First, we enhance the contrasts [15] between environment and transparent object which leads to emphasized boundaries between objects and background. The high contrasts allow the usage of an optimized threshold [16] which, applied to the intensity image of the ToF camera, extracts pixel-areas that represent lower intensities.

This extraction is carried out on a higher pyramid level as well and after that, the results of both levels are combined what removes most of the false candidates. This leads to a better separation of the regions because thin structures in the environment do not appear anymore. The extracted regions are then refined with morphing operators to remove noise, separate areas and obtain solid regions.

B. Automatic Dark Background Adaption

In the course of the development we also considered a dark gray table as a surface. The segmentation procedure partly failed here because the table did not reflect as much of the IR-light anymore. The specularities actually seem to overcome the reflections of the table back through the objects which may originate from the heavily lowered reflectivity. However if the infrared light only passes one transparent layer the intensities keep their attribute of being darker than the table's surface. For two layers we have to deal with higher intensities now as the specularities are brighter than the background. Nevertheless our segmentation adapts to darker backgrounds by assuming that the average image intensity complies with the background intensity. The average intensity is then interpreted as a dark or bright background.

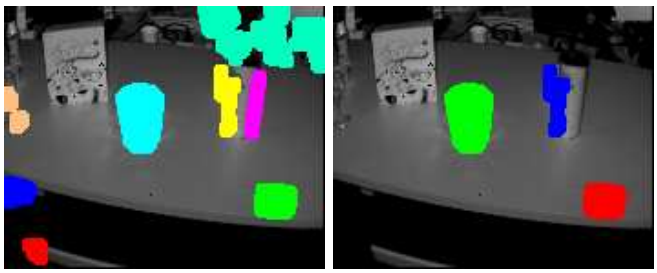
C. Splitting and Merging

Occasionally, we found the regions of non-opaque objects still connected with opaque ones due to distinct ways of scene setups and view angles. As this leads to problems in the later matching phase, we added a second segmentation step that splits these regions. By generating a histogram that assigns column or row values to occurrences. After a smoothing procedure the function representing this histogram is derived for the dimensions of row and column to extract local maxima. Those maxima each represent a new centroid for splitting candidates. All split candidates are tested against nearby candidates for a possible merge to avoid oversegmentation.

D. Candidate Omission

The resulting candidates are then evaluated based on their region convexity. If this factor is above a certain threshold, we calculate the deviations in x, y and z direction for the associated 3D points. We use the deviations as an approximation of the volume that is occupied by the candidate. From the number of the points and this volume we may now calculate the point density. Applying this procedure we are able to omit candidates that do not yield a solid region or high density. Especially noise that occurs at the edges of the view has very low density.

The results of the initial segmentation are exemplarily displayed in Figure 7(a). The refinement of this segmentation using the splitting, merging and omitting steps is shown in Figure7(b). Both images show regions overlayed over the intensity image in different colors for the single segments.



(a) Initial segmentation. (b) After splitting and omission.

Fig. 7. Result of the whole segmentation phase with a solid transparent object in the middle.

IV. Matching

The matching procedure is crucial to find out where the candidate region moved to in the second view in order to be able to triangulate between the two views. We solve this correspondence question by introducing a planarity assumption for the images we observe. We assume the objects to be shadows, or more precisely projections of an object onto a flat surface. This assumption allows us to match the objects with a perspective invariant matching while introducing errors that are discussed later.

A. Initial Planar Matching

Thus, we use a perspective invariant matching, which is robust against noise and local deformations, presented by Hofhauser et. al. in [17]. This matching builds a candidate based on a grayvalue template and returns a planar homography for the best match from the first to the second view. The distance measure used for the matching score is based on corresponding image derivatives in magnitude and direction. A prerequisite for the method is planarity of the template region. However, the matching is robust enough even if the background behind the transparent object is not planar. We use the candidate regions gathered by the segmentation in the first view as template images. The search is then applied to all parts of the second view that were found by the same segmentation step that are close enough to the predicted position. The prediction contains estimates of the

maximal expected rotations and perspective distortions. On success, the matching results in a 2D homography which is applied to the candidate's image points. By transforming a region we get another region in the second view that covers the candidate's match, which leads directly to an initial point to point correspondence in 2D as well as 3D. In order to visualize this correspondence we introduce the following symbols: the segments in the first view will be represented by $S^1 = [s_0, s_1, \ldots, s_n]$, the segments of the second view are $S^2 = [s'_0, s'_1, \ldots, s'_m]$, each consisting of a set of measured points with a 2D coordinate p, a 3D coordinate q and an intensity value v: $s_i = [Q_{i,0}, Q_{i,1}, \ldots Q_{i,r}]$, $Q_{i,j} = \left[p_{i,j} = [x, y]^T, q_{i,j} = [X, Y, Z]^T, v_{i_j}\right]$. We estimate a planar homography H between the two views for each segment s_i and its resulting correspondence in the second view s'_k.

The best discrete pixel correspondence C defined by the indices j and l between the first and the second segment is then just the closest pixel to the transformed point using the homography:

$$C = \left[(0, l_0), \ldots, (r, l_r) | \ \arg\min_{l_j} \|p_{i,j} - H{p'}_{k,l_j}\|\right] \quad (1)$$

$$p_{i,j} \in Q_{i,j} \in s_i, {p'}_{k,l_j} \in {Q'}_{k,l_j} \in {s'}_k$$

Figures 8(a) and 8(b) visualize this relation.

(a) Candidate region in first view (b) Matched and transformed region in second view

Fig. 8. The planar matching provides a perspective 2D homography that defines the pixel to pixel matching.

B. Introduced Errors by Planarity Assumption

[17] also discusses prerequisites for this type of matching. Clearly, our method has to deal with a certain error as the transparent objects are not planar. Our results show that the triangulation of the center of the region is a valid approximation of the center of mass of the transparent object. Figure 9 shows the effect of the planar assumptions: using the correspondences given by Eq. IV-A a plane in space can be reconstructed (right part of the picture in blue). This plane approximates a mean plane of the two silhouettes observed in the two views as shadows on the table. Therefore, every correspondence bears an error e_{move} depending on the distance to the center of the reconstruction. Additionally, we underestimate the depth of the object with the planar assumption by e_{object}.

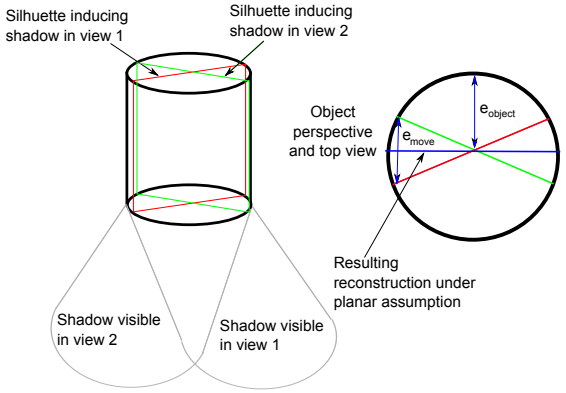


Fig. 9. Visualization of the errors introduced by the planar assumption, left is a perspective view of the scene and on the right a top view of the object with two kinds of errors: e_{object} introduced by the planar assumption, and e_{move} introduced by the changing silhouette due to the cameras movement.

V. Inconsistency Check

The characteristics which we expect to distinguish transparent from opaque objects or dark patches on a flat surface are inconsistencies in their ToF 3D data to the 2D image data. To begin with, we depict the 2D related checks. Using the known 3D transformation P and the projection K, which consists of the known internal camera parameters, we predict the position of the set S^1 in the second view making it comparable to the set S^2. Given a correct 3D measurement, the transformed 3D data should result in the same points $\hat{q}_{i,j} = [X, Y, Z]^T \in s_i \in S^2$ found by the matching procedure if the object is opaque. Then the transformed points $\hat{q}_{i,j} = Pq_{i,j}$ are projected into the image plane by applying K. The resulting 2D points $\hat{p}$ should also be consistent with the 2D points p' of the candidate's match in the second view if the object is opaque. Here, we perform the first check that estimates the extent of offset within the 2D data which should be higher for transparent objects.

$$\sum_{\forall (j,l) \in C} \|\hat{p}_j - p'_l\| \quad (2)$$

This offset in 2D can be derived from the twist a transparent object undergoes which was illustrated in 6(a) and 6(b). Not only causes this twist an offset but also a rotation between the reprojected and the matched candidate region what leads to the second check that determines whether this rotation is above a defined magnitude. From the correspondence between p' and $\hat{p}$ we deduce a similarity transformation in R^2 with 5 degrees of freedom (dof): the rotation R (2 dof), the translation t (2 dof) and the scaling s.

$$\arg\min_{R,t,s} \sum_{\forall (j,l) \in C} {p'_l}^T \begin{pmatrix} sR & t \end{pmatrix} \hat{p}_j \quad (3)$$

This similarity should be resulting in an identity given a correct 3D data, otherwise the sheering introduced by the shadow effect can be measured in the R component. We can define a threshold depending on the movement performed by the camera between the two views which has to be

exceeded by a transparent object. If the candidate data passed these thresholds, the 3D data is tested for inconsistencies to further assure transparency. Given again the 2D point to point correspondence C, we can triangulate the actual 3D position of a point r in space. Assuming the 3D ray given by q, q' expresses and the camera movement P the triangulation of the 3D values can be seen as the following minimization:

$$\underset{m_1,m_2}{\arg\min} \left(m_1 q_{i,j}\right)^T P \left(m_2 {q'}_{k,l}\right) \tag{4}$$

where $m_1, m_2 > 0$ are scalar factors. From this optimization a new 3D point can be calculated using the gained factor m_1 and m_2:

$${q'}_{triangulated} = \frac{\left(m_1 q_{i,j}\right) + P\left(m_2 {q'}_{k,l}\right)}{2} \tag{5}$$

The greater the distance of the two factors m_1 and m_2 from 1, the more likely there is an inconsistency between the triangulation and the 3D measurement. If we triangulate all points in the segments we get approximately a planar structure in 3D which is in correspondence with the measured 3D points, as already mentioned in Figure 9.

The planar body has certain attributes that are correlated to the scalar factors m_1 and m_2 which our first 3D related checks are based on. First the position of the triangulated points is analyzed which should be between the camera and the measured 3D points.

$$q_{i,j} - {q'}_{triangulated,i,j} > 0 \tag{6}$$

The second property is that the planar reconstruction should be nearly orthogonal to the flat shadow of a transparent object. We reassess this fact by fitting a plane into both the measured and the reconstructed points and calculate the angle α between.

$$\alpha = \arccos\left(\frac{n^T \cdot n'}{|n||n'|}\right) \tag{7}$$

with the normals n and n':

$$n = \underset{a,b,c,d}{\arg\min} \left(q_{i,j}\right)^T \begin{pmatrix} a & b & c & d \end{pmatrix}^T \tag{8}$$

$$n' = \underset{a',b',c',d'}{\arg\min} \left(q'_{triangulated,i,j}\right)^T \begin{pmatrix} a' & b' & c' & d' \end{pmatrix}^T \tag{9}$$

After the shadow-like behaviour was checked by means of its 2D inconsistency and its planarity, a last procedure checks whether the mentioned 3D movement of the shadow actually took place. As the movement should be determined by a rotation of the planar candidate point cloud, we identify its screw parameters by estimating the 3D transformation parameters that align the points q and Pq'. In order to obtain this information our method makes use of a single step of the ICP-algorithm [19] applying a least-squares-minimization to the following:

$$\underset{R_{icp}T_{icp}}{\arg\min} \sum_{\forall (j,l) \in C} q_j^T R_{icp} T_{icp} P {q'}_l \tag{10}$$

As a matter of fact the angles of rotation around the three axes are much bigger for the translucent obstacles than for opaque ones. From the rotation matrix R_{icp} we can deduce the 3D twist the respective candidate has undergone. The overall magnitude of rotation between the point clouds q and Pq' indicates a transparent objects with a high value and an opaque objects with a very low value even if the measurement is erroneous. All together our method compares the two views of each candidate by means of image point distance and twist between the object's point clouds to determine whether an observed low intensity area is caused by an opaque or a translucent object.

VI. Reconstruction

To finally be able to manipulate a transparent object, its real 3D occurrence must be recovered. As explained above, the initially reconstructed 3D points are located in a plane. Although the reconstruction mentioned in V is closer to an object standing upright it does not describe the real volumetric shape well enough to form a manipulable point cloud. Given the errors introduced in Figure 9, we can compensate for the error e_{object} and e_{move} by appending a set of additional possible intersections inside the object to the triangulation. The correct triangulation should occur in a set of reconstructions between $q_{k,k}$ and $P \cdot q'_{k,i}$ with k being a fixed number and i running from the first to the last column of the image row $p'_{k,i}$ that corresponds to $q_{k,i}$ and $q'_{k,i}$.

$${q'}_{triangulated,i} = \frac{\left(m_1 q_{k,k}\right) + P\left(m_2 {q'}_{k,i}\right)}{2} \tag{11}$$

Figure 10 illustrates such a triangulation performed on one horizontal slice of the objects shadow.

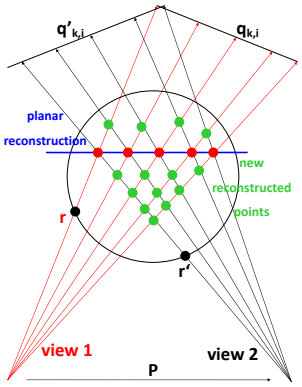


Fig. 10. The possible intersection points when triangulating between one image point of view 2 and all image points in view 1 that are located on a single image row.

The circle in figure 10 indicates a transparent object. The points r and r' satisfy the planar matching error explained in section IV which also leads to the planar reconstruction displayed by the blue line and the red dots. In order to compensate this error we triangulate new points shown as green dots using the matched points $q_{k,k}$ and $q'_{k,k}$ as well

as the points in $q'_{k,i}$ that yield a point $q_{triangulated,k,i}$ that is closer to the second view position than the planar reconstruction. Yet the method does not intersect $q_{k,k}$ with all points $q'_{k,i}$ of the respective image row. The last point used fulfils the following:

$$|i| = 0.5 \cdot |p_{k,i}| \tag{12}$$

This approach yields a 3D blob shaped like the 2D region $p'_{i,j}$ that embodies an average point $\overline{q}_{triangulated,i,j}$ and the deviations around it which closely approximate the real position of the transparent object. The quality of this approach will be evaluated in the following section.

VII. Results

To demonstrate the performance of our system we want to focus on two facts: First, does the system detect transparent objects without any hallucinating and without detecting any color changes in the background or even opaque objects as transparent objects. Second, we want to show that the 3D reconstruction is good enough to try a grasp of this object without further modeling.

A. Testing Environments

To rate the performance of our approach, we tested our methods on a set of different objects shown in figure 11(a) and 11(b).

(a) Camera image of the objects illuminated by laboratory light with a high proportion of green.

(b) ToF point cloud of the objects.

Fig. 11. Objects that defined the testing environments.

The transparent objects were a plastic mug, a plastic wine glass, a drinking glass, a glass made of red plastic and a glass made of yellow plastic which appeared to be transparent for the SR4k. The opaque objects consisted of an ice tea container, a package of tea bags, a textured cup, a white cup, a white plastic cup, a light blue cup and a black thermos jug. We applied our procedure to every object individually having it placed on a gray kitchen table. The positions on the table altered in a way that the respective object was positioned in the middle or close to one of the edges of the table.

B. Accuracy and Grasping

How accurate our method performed overall on the test setup can bee seen in table I.

We tested every mentioned object five times at different relative positions to the robot on a table. For the test we used a dark table, since it was a newly installed kitchen in our lab and it represents the most challenging scenario for our approach: It is striking that we had no false positive detection implying that our method did not try to reconstruct any object with a decent response in the ToF-camera. Accordingly, all the opaque candidates were omitted by the inconsistency checks. On the other hand, we achieved a higher amount of false negatives showing that our thresholds were immoderate at times. Yet we successfully reconstructed 58 out of 105 transparent objects and grasped 41% in a way that our platform confirmed its capability to further manipulate, relocate or retrieve the reconstructed object.

		classified as	
type	count	transparent	no transparent object
transparent	105	55.24%	44.76%
opaque	44	0.00%	100.0%
grasp success		24/58	-

TABLE I

The rates of reconstructed transparent objects in the test with one to three objects in a scene.

Our grasping method first executes a probabilistic encaging and applies corrections by reactively repositioning the hand in case of collisions which are detected with force feedback sensors. The failed grasping attempts in our tests arouse from slight reconstruction errors that either overestimated the height of the object such that the grasping position was too high or introduced an inaccurate position estimation what lead to collisions between the grasper and the object. Those collision could eventually cause wrong corrections of the hand or they could be missed by the sensors in case of light objects like the plastic glasses, which would move the object out of the expceted position.

This test includes objects solely on the table and three objects at the same time nearby not affecting the test performance which was for the single experiments all about approximately 53% - 58%. The following figure 12 denotes which stage of our checks principally contributed to candidates being omitted.

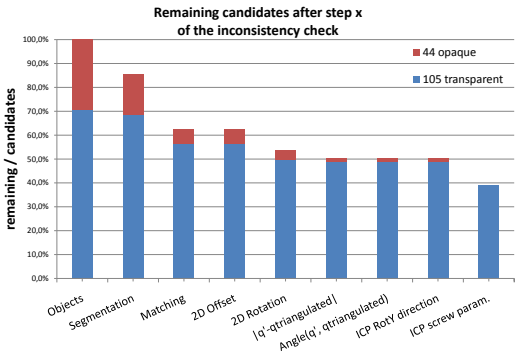


Fig. 12. Candidates remaining after a certain stage of the inconsistency checks.

As one can see, the candidates corresponding to opaque objects are mostly filtered by our segmentation method which originates from our initial idea of transparent objects yielding

lower intensities through the doubled absorption. Our method filters further false positives in the matching phase where the transparent object might not be found in the second view because it was omitted in the previous segmentation which of course has to be applied again to the second view data. The remaining opaque objects then get discarded because they do not bear the twist a transparent object creates. The amount of wrong negatives can be tracked in figure 12 as well. We only lost 3 positive candidates in our segmentation due to heavy occlusion but a fair amount due to unsuccessful matching expectedly. The loss occurring during the check of the 3D twist was identified to be an inconsistency in the calculations of our ICP method. Nevertheless, the method did not reject candidates wrongly for the rest of our checks.

VIII. Conclusion

We presented a method that extracts candidate regions, evaluates their attributes comparing two views and reconstructs the shape of transparent objects. The experiments showed that our candidate search as well as the determination between opaque and transparent object attributes is valid. Alongside, we proved that transparent objects hold the characteristics of a shadow when they are observed with a ToF camera. The validity of our reconstruction method was demonstrated by the successful manipulation of transparent objects. We were also able to maintain the claimed invariance to environmental lightening by exclusively relying on the active illumination of our ToF-camera. The unique behaviour of transparency in ToF data leads to an identification of translucent objects and yields a fundament for reconstruction approaches. Our method is already capable of converting this information into an object reconstruction that enables a robot platform to perform successful grasping as well as collision avoidance procedures. The drawbacks are currently the relatively high missing rate of actual transparent objects, which are mostly caused by the difficult conditions for the segmentation and matching induced by the sensor's low signal-noise ratio.

In the future we concentrate on refinements such that our approach can cope with cluttered scenes, occlusions and different surfaces. Eventually, transparent objects shall become manipulable in any situation that occurs in a real household environment.

Acknowledgments: This work was supported by CoTeSys (Cognition for Technical Systems) cluster of excellence at TUM and by MVTec Software GmbH, München.

References

[1] G. Eren, O. Aubreton, F. Meriaudeau, L. Secades, D. Fofi, A. Naskali, F. Truchetet, and A. Ercil, "Scanning from heating: 3D shape estimation of transparent objects from local surface heating," *Opt. Express*, vol. 17, pp. 11 457–11 468, 2009.

[2] I. Ihrke, K. Kutulakos, H. Lensch, M. Magnor, and W. Heidrich, "State of the art in transparent and specular object reconstruction," *STAR Proc. of Eurographics*, pp. 87–108, 2008.

[3] R. Rusu, A. Holzbach, R. Diankov, G. Bradski, and M. Beetz, "Perception for mobile manipulation and grasping using active stereo," in *9th IEEE-RAS International Conference on Humanoid Robots, 2009. Humanoids 2009*, 2009, pp. 632–638.

[4] A. Wallace, P. Csakany, G. Buller, A. Walker, and S. Edinburgh, "3D imaging of transparent objects," in *Proc. British Machine Vision Conf*. Citeseer, 2000, pp. 466–475.

[5] S. Yang and C. Wang, "Dealing with laser scanner failure: Mirrors and windows," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation*, 2008.

[6] K. Kutulakos and E. Steger, "A theory of refractive and specular 3D shape by light-path triangulation," *International Journal of Computer Vision*, vol. 76, no. 1, pp. 13–29, 2008.

[7] P. Lagger, M. Salzmann, V. Lepetit, and P. Fua, "3d pose refinement from reflections," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 0, pp. 1–8, 2008.

[8] S. Agarwal, S. Mallick, D. Kriegman, and S. Belongie, "On refractive optical flow," *Computer Vision-ECCV 2004*, pp. 483–494, 2004.

[9] M. Ben-Ezra and S. Nayar, "What Does Motion Reveal About Transparency?" in *Proceedings of the Ninth IEEE International Conference on Computer Vision-Volume 2*. IEEE Computer Society, 2003, p. 1025.

[10] S. Gudmundsson, *Robot vision applications using the CSEM swissranger camera*. Master's thesis. Informatics and Mathematical Modelling. Technical University of Denmark. Copenhagen. Denmark, 2006.

[11] D. Miyazaki, M. Saito, Y. Sato, and K. Ikeuchi, "Determining surface orientations of transparent objects based on polarization degrees in visible and infrared wavelengths," *JOSA A*, vol. 19, no. 4, pp. 687–694, 2002.

[12] J. Pelletier and X. Maldague, "Shape from heating: a two-dimensional approach for shape extraction in infrared images," *Optical engineering*, vol. 36, p. 370, 1997.

[13] A. Maldonado, U. Klank, and M. Beetz, "Robotic grasping of unmodeled objects using time-of-flight range data and finger torque information," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, October 18-22 2010.

[14] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," in *In IEEE International Conference on Robotics and Automation (ICRA 2009)*, 2009.

[15] R. Gonzalez and P. Wintz 2nd, "Digital Image Processing 2nd Edition Addison Wesley," *Reading, Mass*, 1987.

[16] W. Niblack, *An introduction to digital image processing*. Strandberg Publishing Company Birkeroed, Denmark, Denmark, 1985.

[17] A. Hofhauser, C. Steger, and N. Navab, "Perspective planar shape matching," in *Image Processing: Machine Vision Applications II*, ser. Proc. SPIE 7251, K. S. Niel and D. Fofi, Eds., 2009.

[18] R. Rusu, A. Holzbach, R. Diankov, G. Bradski, and M. Beetz, "Perception for mobile manipulation and grasping using active stereo," in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*. IEEE, 2010, pp. 632–638.

[19] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least square fitting of two 3-d point sets," in *IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 9, Issue 5*. IEEE, 1987, pp. 698–700.