3D Multi-Sensor Data Fusion for Object Localization in Industrial Applications

C. Pfitzner, W. Antal, P. Heß, S. May, C. Merkl, P. Koch, R. Koch, M. Wagner
Nuremberg Institute of Technology Georg Simon Ohm, Germany

Abstract

This paper describes a data fusion approach for 3D sensors exploiting assets of the signed distance function. The object-oriented model is described as well as the algorithm design. We developed a framework respecting different modalities for multi-sensor fusion, 3D mapping and object localization. This approach is suitable for industrial applications having need for contact-less object localization like bin picking. In experiments we demonstrate 3D mapping as well as sensor fusion of a structured light sensor with a Time-of-Flight (ToF) camera.

1 Introduction

Gripping and handling of objects demand precise reconstruction of surfaces – surfaces to which robot gripper jaws need to adapt. Reconstruction is challenging when dealing with complex 3D shapes, especially if fine details are of interest or surfaces have specular reflection characteristics. For the localization of such objects sensors need to be selected w.r.t. the desired working range or the physical principle. In many cases, a combination of different sensors is beneficial. Optical sensors like 2D laser profile sensors stand out due to high precision but are also characterized by a small field of view and high costs. Furthermore, 3D perception is only possible while moving either sensor or object.

In general, the surface of a complex 3D shape is to be reconstructed by merging measurements of sensors from different perspective views. Depth measurements are needed as well as the sensor's pose. But often the manipulator can not guarantee high precision in every state of moving as shown by Stelzer et al. [17]. Higher degrees of freedom and the resulting non-linear equations cause position errors while moving at maximum speed. Commonly, high precision is only given while moving the sensor with low speed or in deadlock. In addition, the acquisition of encoder and 3D data needs to be synchronized.

Sensor fusion can help to reconstruct the environment. Every sensor has benefits and handicaps dedicating them for certain fields of application [8]. Dealing with multiple sensors raises the question: How much do I trust a certain sensor in the current situation? The challenge of multi-sensor data fusion and mapping lies in the variety of measurement characteristics. Differences in resolution, frame rate, range, accuracy or sensor noise makes need for specific mathematical sensor models.

A suitable application in industrial environments is bin picking for sorting of parts with focus on high flexibility in pick-and-place tasks: Via a sensor – mounted on the robot itself or from an external view point, cf. **Figure 1** – data from the environment is processed by a compute unit in order to classify objects and determine their pose. With known size, position and orientation grasping and manipulation of objects can be performed.

In this paper we present a representation for 3D multi-sensor data fusion with focus on object localization. Multiple measurements are fused in a generic truncated signed distance representation, from which smooth point clouds with minimal noise can be extracted. Object localization and classification is more robust on the basis of the fused data set [21].

This paper is structured as follows: In section 2 the related work in contact-less object localization in industrial applications is presented as well as 3D reconstruction from sensor data. Section 3 introduces the approach in software and algorithm development for data acquisition and sensor fusion. Experiments in section 4 show the accuracy in localization of a given scene in comparison to a high precise 2D laser range finder, cf. **Figure 2b**. Also sensor fusion with two rigid mounted sensors is shown in experiments. With the outlined approach RGB-D cameras can augment an accurate laser point cloud with color information. Finally, a short conclusion is given.

2 Related Work

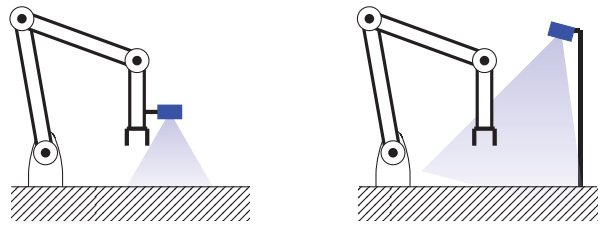
This section partitions related work in object localization, 3D reconstruction and mapping, and multi-sensor fusion.

2.1 Object Localization

Bin picking based on machine vision was developed decades ago for pick-and-place applications and is still an active research area. Gradually the sensors changed with different approaches.

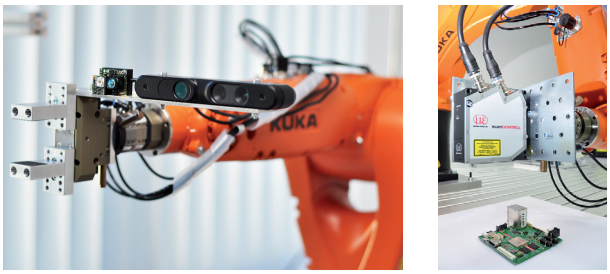
Horn and Ikeuchi [7] presented in 1983 one of the first bin picking approaches: By the shape from shading approach

ISBN 978-3-8007-3601-0



(a) (b)

Figure 1: Bin picking with sensor rigid attached to robot (a) and sensor with view on working area (b).



(a) (b)

Figure 2: Set up for contact-less localization with ToF camera and structured light sensor (a) and for reference measurements with a laser profile scanner (b).

torus objects were localized while the robot moved towards the estimated grasping point. This approach is based on a monocular camera.

Monocular cameras only work on separated objects with fixed geometry. Grasping stacked objects could fail with a collision because of uncertain object height. In 1990 Al-Hujazi and Sood [2] showed 3D range image segmentation based on edge detection and region growing. The algorithm determines the potential holdsites for gripping the object. On occasion different type of 3D sensors like Time-Of-Flight cameras [7] or structured light sensors like the Microsoft Kinect [15] are used for bin picking approaches.

Nieuwenhuisen et al. [15] demonstrated in 2013 an approach of extending the robot's workspace for the bin picking problem by replacing a stationary picking robot by an autonomous anthropomorphic mobile robot. 3D object recognition is based on graph matching of aligned point cloud scans.

2.2 3D Reconstruction and Mapping

3D Reconstruction can be done with different approaches: Taylor [19] showed the usage of feature based methods for the reconstruction of complex 3D shapes. In many fields of applications, iterative schemes are commonly used. The Normal Distribution Transform (NDT) [4, 11] and the Iterative Closest Points (ICP) algorithm are de facto standard for range image registration [3, 23]. With the release of the Microsoft Kinect camera in 2010, many researchers focused the localization and mapping with hand-held RGB-D cameras. On of the most considerable approaches with this type of sensor was published by Izadi et al. [9] under the name KinectFusion: Based on the signed distance function (SDF) [16] they showed 3D reconstruction in real-time, while exploiting massive parallelism on GPUs. Localized by ICP registration, the hand-held Kinect camera can be used to fill a defined volume while minimizing the errors of the depth image channel through data integration. Because of a high frame rate, due to the GPU implementation, the search for corresponding point pairs is done efficiently. Out of the defined volume, high-density 3D models can be extracted and used for further processing. In the *point cloud library* (PCL) an open-source implementation under the name KinFu [1] is available.

Sturm et al. used the KinectFusion approach to reconstruct persons in the sensors's field of view. The Kinect may even be used to create a 3D model of oneself. Sending this model to a 3D printer, one receives a copy of the own body [18].

Whelan et al. extended the KinectFusion approach to work on large scale environments [22]. The close-range reconstruction is done classically with KinectFusion. Areas that leave the predefined volume under sensor motion are subsequently extracted and meshed.

2.3 Multi-sensor Data Fusion

Multi-sensor data fusion is essential w.r.t. localization and identification of objects. Identification is based on the outstanding attribute of an object or a combination of its characteristics.

One reason for data fusion is to confirm the data of another sensor. As mentioned previously, reflective surfaces cause errors when using a laser scanner. On the other hand an ultrasonic sensor has a low precision. The combination of both is done by Fabrizi et al. [6].

Two rigid mounted monocular cameras can be used for 3D perception. With an overlapping field of view of two calibrated cameras depth calculation is feasible as shown by Zhang [24]. Intrinsic and extrinsic parameter estimation of this stereo arrangement is performed by a closed form solution. Depth estimation with stereo cameras works well for textured surfaces.

In contrast, Time-of-Flight (ToF) cameras work on structure-less surfaces, but have commonly less resolution and less working range. Additionally, a specific error model resulting from the measurement principle is needed to be designed, e.g., jumping edge errors or multi-path reflection. With the fusion of a stereo camera and a ToF camera drawbacks of both principles can be compensated. Nair et al. used a local fusion, based on stereo block matching and subsequently a variational fusion based on total variation to increase smoothness of data [13]. Using multiple ToF cameras increases depth data accuracy as shown by Kim et al. [10].

ISBN 978-3-8007-3601-0

To the best of our knowledge, a sensor fusion application for ToF and structured light has not been realized so far.

3 Approach

Algorithms section is divided in representation, the usable sensor models and the framework for sensor fusion.

3.1 Representation

The approach's basis – the signed distance function – represents the distance of a given point to a surface. The space from which a map is to be reconstructed, is divided in voxels for a 3D representation. Let $\mathbf{v}$ be the center of an arbitrary element, $\mathbf{p}$ the sensor's position and m the distance measurement determined in the direction of the given element, the signed distance function (**SDF**) reads:

$$d(\mathbf{v}) = m - ||\mathbf{p} - \mathbf{v}|| \tag{1}$$

If the SDF returns negative values, corresponding elements are not visible to the sensor due to occlusion. Thus, the values of the SDF are truncated or respectively multiplied with a weight w. The multiplication with the exponential model according to Bylow et al. [5] results in the truncated signed distance function (**TSDF**) and is described as followed.

$$w(d) = \begin{cases} 1 & \text{if } d \geq -\epsilon \\ e^{-\sigma(d-\epsilon)^2} & \text{if } d < -\epsilon \text{ and } d > -\rho \\ 0 & \text{if } d < -\rho \end{cases} \tag{2}$$

With focus on measurement noise and multiple view data integration, the representation by the TSDF can be considered for sensors with noisy data like structured light or ToF cameras. Further information relating to the TSDF can be found in the publications of KinectFusion [9, 14].

Figures 3 sketches the registration and integration process of new measurement data.

```
1:  procedure ONSENSORDATAREVEIVE(sensor)
2:      model ← RAYCAST(sensor)
3:      scene ← get data from sensor
4:      T_icp ← icp registration of model and scene
5:      T_sensor ← T_icp T_sensor        ▷ update sensor pose
6:      if (T_sensor T_last_push^-1 > thresh) then
7:          T_last_push ← T_sensor
8:          PUSH(sensor)
9:      end if
10: end procedure
```

Figure 3: Registration and integration of new measurement data.

The function *PUSH* is responsible for the localization of the sensor, as well for adding information to the voxel space and is comparable to Izadi et al. [9].

3.2 Sensor model

The here presented approach is suitable for several types of sensors. The most common sensor model for 3D devices is the pin hole camera model which works for RGB-D sensors as well as for ToF cameras.
The pinhole model is represented by a 3×4 projection matrix $\mathbf{P}$,

$$\mathbf{P} = \begin{pmatrix} f_u & 0 & t_u & 0 \\ 0 & f_v & t_v & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3}$$

$$\mathbf{P}\xi = (su, sv, s)^{\mathrm{T}} \rightarrow (u, v)^{\mathrm{T}}, \tag{4}$$

with f_u and f_v representing scaling parameters and t_u and t_v the coordinates of the principal point. The parameter s respects the fact that all points along a ray of sight are projected to the same pixel coordinate $(u, v)^{\mathrm{T}}$. This ambiguity is resolved with the measured distance.
Out of equations 3 and 4 the pixel-dependent line of sight can be calculated by inversion. In this case it reads:

$$x = \frac{1}{f_u} \cdot u - t_u \qquad y = \frac{1}{f_v} \cdot v - t_v \qquad z = 1 \tag{5}$$

Out of these definitions the assignment of an arbitrary coordinate to the measurement matrix and vice versa is possible. **Figure 5a** shows the raycasting model for projective sensors like the Asus Xtion or the ToF camera.
At this time, also a second model for 3D localization is implemented, suitable for a 2D laser scanner moved by a robot: The model for the 2D laser range finder is described by the conversion between polar and Cartesian coordinates. The line of sight in the 2D scanning plane of a Micro Epsilon device $(x'z')$ is determined by

$$x' = \sin\theta \qquad z' = \cos\theta, \tag{6}$$

where θ is the angle of the laser beam. The translation in one direction of a robot along a linear movement is described by an additional three degrees of freedom translation vector $\mathbf{t}^{\mathrm{T}} = \begin{pmatrix} t_x & t_y & t_z \end{pmatrix}$

$$x = x' + t_x \qquad y = t_y \qquad z = z' + t_z \tag{7}$$

The easiest way to get full 3D perception with such a 2D laser range finder is to move the scanner in y direction. Moving only in x or z direction does not deliver full 3D perception while moving the sensor above the scene for bin picking.
The back projection converts an arbitrary point $\mathbf{p} = (x\ y\ z)$ in polar representation as follows, provided that $t_x = t_z = 0$.

$$\theta = \arctan\frac{x}{z},\ \phi \in [-\theta; \theta] \qquad t = t_y \tag{8}$$

Figure 5b depicts the sensor model for the laser range finder.
Further sensor models can be adapted and merged in the existing sensor framework. If deduction in time is possible, the framework can also be implemented on power-saving CPUs or even embedded platforms.

ISBN 978-3-8007-3601-0

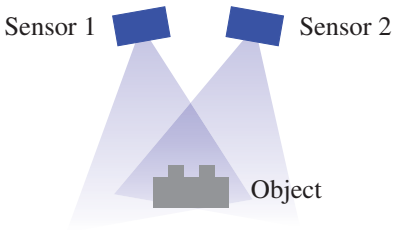


Figure 4: Sensor fusion with two rigid mounted sensors focusing on object in the field of view.

The framework can easily be adapted to other applications and sensors. Models for other types of sensors can be added. The software library is made available as open-source at
`https://github.com/stefanmay/obviously`.

1: **procedure** FUSESENSORS
2: **while** (TSDSpace = empty) **do**
3: ONSENSORDATARECEIVE($sensor1$)
4: **end while**
5: $model \leftarrow$ RAYCAST($sensor2$)
6: $scene \leftarrow$ get data from sensor1
7: $\mathbf{T}_{icp} \leftarrow$ icp registration of model and scene
8: $\mathbf{T}_{sensor2} \leftarrow \mathbf{T}_{icp}\mathbf{T}_{sensor1}$ ▷ update sensor pose
9: $\mathbf{T}_{fusion} \leftarrow \mathbf{T}_{icp}$
10: **end procedure**

Figure 6: Sensor fusion with two sensors based on the TSDSpace.

3.3 Sensor Fusion and Calibration

As explained in the introduction multi-sensor data fusion is challenging due to different sensor characteristics. The TSDSpace can solve this problem because of sensor corresponding raycasting from the voxel space.

Multi-sensor fusion can be done with two possibilities: Each sensor can map the environment independently, for example mounted on two separated robots. The other possibility can be achieved with two sensors mounted rigid to each other. The field of view of both sensors must have nearly the same orientation so the perception of the environment is partially overlapping, cf. **Figure 4**.

To estimate the transformation $\mathbf{T}_{fusion}$ between two sensors a scene providing unique features is mandatory. Therefore every object seen in the raw data of the sensors can be used.

While filling the TSDSpace with the first sensor the rigid sensor array can be moved freely in the scene. If the TSDSpace is augmented with enough filled voxels, both sensors should be fixed in a static position. After fixing the sensor array the transformation between those two sensors can be estimated by ICP. **Figure 6** demonstrates the sequence in sensor fusion based on the TSDSpace.

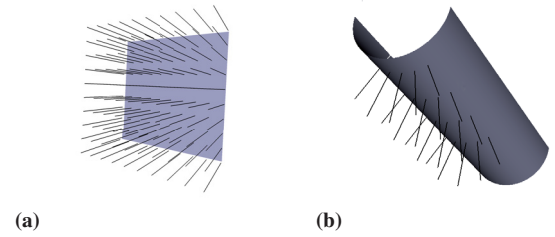


Figure 5: Raycasting models for different sensors with structured light sensor (a) and 2D laser range finder (b).

4 Experiments and Results

The here presented approach is tested on a scene containing small objects, cf. **Figure 7a**, which would be suitable for a bin picking application.

4.1 System overview

The Asus Xtion Pro Live delivers point clouds with a resolution of 320×240 and a granularity of 1 cm. The structured light sensor has a field of view of $58°$ H, $45°$ V. Additional to the depth sensor the Asus Xtion Pro Live includes additionally a pre-calibrated RGB camera. Because of the structured light principle the sensor can only deliver points with a distance higher than 0.4 m to the sensor.

The second sensor in our experiments is the PMD CamBoard Nano with a resolution of 160×120 and a granularity of 1.5 cm. In contrast to the Asus Xtion, the ToF camera can deal with objects in near field. Because of low illumination the CamBoard Nano has a working range of approximately 1.5 m.

Both sensor devices are mounted rigid to each other and attached to a Kuka KR 6 R900 sixx industrial robot. Localization and mapping was done in real-time on a modern Intel Core i7 CPU. The position of the robot was not taken into account for localization to demonstrate the registration of depth information. Sensor data was prefiltered with a bilateral filter [20] as well as with a threshold in magnitude. The data from the ToF camera was also filtered for jumping edge errors to minimize errors in localization [12].

For reference measurements the Micro Epsilon 2600-100 laser profile scanner was used. It features a field of view of about $20°$ while the operating range is limited to 26.5 cm. However it delivers depth information with a resolution of 12 µm. Point clouds were generated by moving the scanner attached on a robot over the scene with low speed while taking scan slices in millimeter steps, cf. **Figure 7b** and **7c**.

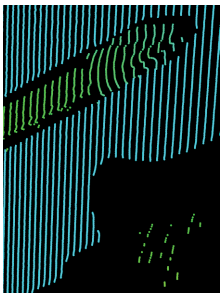
4.2 Registration

The first experiment demonstrates the mapping based on the truncated signed distance function as mentioned in section 3. Point clouds are colored in depth for better

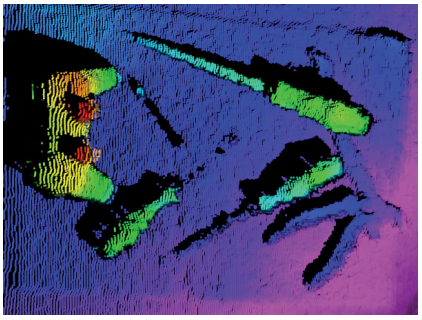
ISBN 978-3-8007-3601-0

(a) Scene for testing

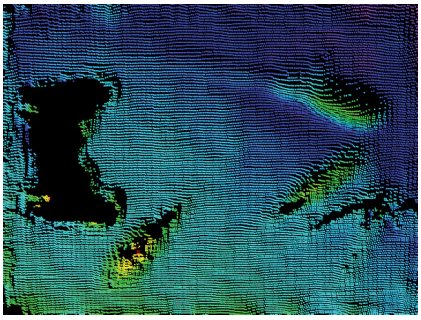
(b) Point cloud from 2D laser scanner



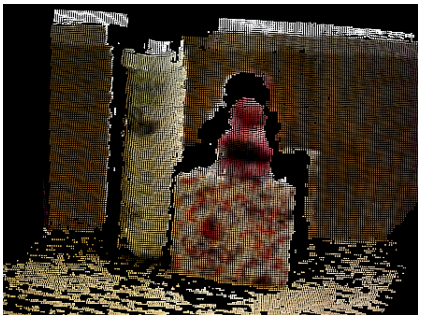
(c) Detail with screw driver tip



(d) Asus Xtion point cloud



(e) CamBoard Nano point cloud



(f) Fused point cloud

Figure 7: Set-up and results of experiments.

visualization. **Figure 7** demonstrates the results of mapping with different sensors: Needless to say the laser generates a precise model of the scene, cf. Figure 7c where even the tip of the screwdriver can be classified. The benefit of the truncated signed distance function can be seen in the amount of time for mapping. With the small field of view the scanner has to be moved several times depending on distance to the objects and the region the objects assign. So finding an object just with such a sensor is time consuming. Also on dark and inclined surfaces the laser scanner can not deliver all points of an object, as seen on the points of the game pad in Figure 7b.

Because of the wider field-of-view the structured light sensor and the ToF camera can see the whole scene in one vision pick-up without moving. Resulting point clouds in **Figure 7d** and **7e** were generated by less than ten sensor takes while slightly moving the sensor above the scene.

In contrast to the reconstruction from laser data, the game pad can clearly be recognized in the point cloud of the Asus Xtion. Even small details like the 2 mm height buttons are visible.

The ToF camera works best on rough optical surfaces. The shiny screw-wrench can hardly can be seen in the cloud of the ToF camera and the game pad can only be recognized by it's shadow. Both sensors have problems in representing the thin shiny parts of the screw driver.

4.3 Sensor Fusion

In a second experiment, the ToF camera and the structured light sensor were mounted rigid to each other, looking at a scene with unique geometric shapes from the sensors view. The TSDSpace is filled with sensor data from the ToF camera with few sensor takes. After that the transformation $\mathbf{T}_{fusion}$ is computed with the help of the back projection. **Figure 7f** shows the resulting sensor fusion. Due to the fact the ToF camera does not deliver any color information still some of the resulting points in the cloud are white. The augmented color from the structured light sensor is blurred because of the trilinear interpolation of the color.

5 Conclusion

In this paper we presented the approach of the truncated signed distance function for localization and registration. Furthermore we showed the fusion of different sensor types in an experiment with a ToF camera and a structured light sensor.

The benefit of the truncated signed distance function is shown in experiments: In comparison to the laser profile scanner, the here introduced approach can be used for bin picking applications with the need of a wide field of view for object detection. The accuracy gets better through the TSDSpace, than with raw data because of merging sensor data from slightly different views. The already existing RGB camera in the structured light sensor can be assistant in finding objects for bin picking applications due to their surface color.

ISBN 978-3-8007-3601-0

References

[1] Point cloud library (PCL). http://pointclouds.org, 2013. Accessed on 13/10/2013.

[2] E. Al-Hujazi and A. K. Sood. Range image segmentation combining edge-detection and region-growing techniques with applications sto robot bin-picking using vacuum gripper. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(6):1313–1325, 1990.

[3] P. Besl and N. McKay. A method for Registration of 3–D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239 – 256, February 1992.

[4] P. Biber and W. Straßer. The normal distributions transform: a new approach to laser scan matching. In *Proceedings of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 2743–2748, 2003.

[5] E. Bylow, J. Sturm, C. Kerl, F. Kahl, and D. Cremers. Real-time camera tracking and 3d reconstruction using signed distance functions. In *Robotics: Science and Systems Conference (RSS)*, June 2013.

[6] E. Fabrizi, G. Oriolo, and G. Ulivi. Accurate map building via fusion of laser and ultrasonic range measures. In *Fuzzy Logic Techniques for Autonomous Vehicle Navigation*, volume 61. Physica-Verlag HD, 2001.

[7] S. Fuchs, S. Haddadin, M. Keller, S. Parusel, A. Kolb, and M. Suppa. Cooperative bin-picking with time-of-flight camera and impedance controlled dlr lightweight robot iii. In *IROS*, pages 4862–4867. IEEE, 2010.

[8] D. L. Hall and J. Llinas. An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85:6–23, 1997.

[9] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, 2011.

[10] Y. M. Kim, C. Theobalt, J. Diebel, J. Kosecka, B. Miscusik, and S. Thrun. Multi-view image and tof sensor fusion for dense 3d reconstruction, 2009.

[11] M. Magnusson. *The Three-Dimensional Normal-Distributions Transform — an Efficient Representation for Registration, Surface Analysis, and Loop Detection*. PhD thesis, Örebro University, 2009. Örebro Studies in Technology 36.

[12] S. May, D. Droeschel, D. Holz, S. Fuchs, E. Malis, A. Nüchter, and J. Hertzberg. Three-dimensional mapping with time-of-flight cameras. *J. Field Robot.*, pages 934–965, 2009.

[13] R. Nair, F. Lenzen, S. Meister, H. Schäfer, C. S. Garbe, and D. Kondermann. High accuracy tof and stereo sensor fusion at interactive rates. In A. Fusiello, V. Murino, and R. Cucchiara, editors, *ECCV Workshops (2)*, volume 7584 of *Lecture Notes in Computer Science*, pages 1–11. Springer, 2012.

[14] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '11, pages 127–136. IEEE Computer Society, 2011.

[15] M. Nieuwenhuisen, D. Droeschel, D. Holz, J. Stückler, A. Berner, J. Li, R. Klein, and S. Behnke. Mobile bin picking with an anthropomorphic service robot. In *ICRA*, pages 2327–2334. IEEE, 2013.

[16] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces (Applied Mathematical Sciences)*. Springer, 2003 edition, 2002.

[17] M. Stelzer, Stryk, E. Abele, J. Bauer, and M. Weigold. High speed cutting with industrial robots: Towards model based compensation of deviations. In *Proceedings of Robotik*, 2008.

[18] J. Sturm, E. Bylow, F. Kahl, and D. Cremers. CopyMe3D: Scanning and printing persons in 3D. In *German Conference on Pattern Recognition (GCPR)*, Saarbrücken, Germany, September 2013.

[19] C. J. Taylor. Surface reconstruction from feature based stereo. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, pages 184–, Washington, DC, USA, 2003. IEEE Computer Society.

[20] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846, 1998.

[21] K. Umeda and T. Arai. Industrial vision system by fusing range image and intensity image. In *Multisensor Fusion and Integration for Intelligent Systems, 1994. IEEE International Conference on MFI '94.*, pages 337–344, 1994.

[22] T. Whelan, M. Kaess, J. Leonard, and J. McDonald. Deformation-based loop closure for large scale dense RGB-D SLAM. In *Proceedings of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[23] Z. Zhang. Iterative point matching for registration of free-form curves. Technical Report RR-1658, INRIA Sophia Antipolis, Valbonne Cedex, France, 1992.

[24] Z. Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, 2000.

ISBN 978-3-8007-3601-0