Proceedings of the 3rd Annual
IEEE Conference on Automation Science and Engineering
Scottsdale, AZ, USA, Sept 22-25, 2007

Object localization in range data for robotic bin picking

K. Boehnke, *Student Member, IEEE*

Abstract— **This paper describes an approach to solve the *bin picking problem*. In many industrial processes, product parts, which have to be assembled, are delivered scrambled in boxes. Usually these parts have to be picked out of the box manually to feed them into an automated process. Using an industrial robot for this task is very difficult. This problem is not solved in general up to now. Our flexible approach uses knowledge about the form of the objects to find them in range data. We compare the 2.5D-appearance of simulated object poses with the real range data in two different steps, and find the best matching pose of the object. This approach can handle many different kinds of objects and takes features of range sensors into consideration to improve the accuracy and robustness of the object localization.**

I. INTRODUCTION

THINKING about the task to pick an object out of a bin every human is able to learn this in the first years of his life. Imaging a 2-years-old baby taking his favorite toy car out of a box full of toy cars of different shapes, sizes, and colors; everyone can agree with this: This is an easy task for the baby. At first glance, everyone takes this for granted.

So why is this task so difficult for a robot? Taking unknown objects out of a bin is still an unsolved problem in the field of robotic automation. Independent from the field of research, most of known solutions to solve this problem are limited often to shape based or single objects[1],[2],[3]; sometimes to a few simple kinds of objects[4],[5].Due to the fact, that every object differs in form, position, and orientation, a general solution of bin picking problem is hard to find. To understand the main targets of the problem we have to separate the whole task into different process steps.

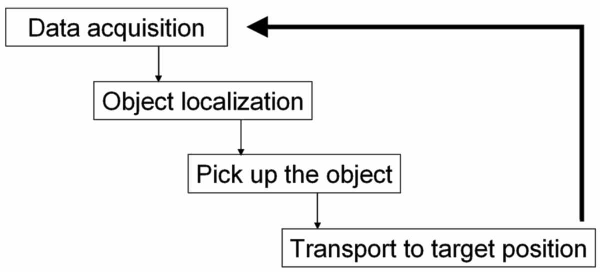

Fig. 1. The iterative robotic bin picking process starts with data acquisition. After this it follows the object localization which is the most important and complex step in this process. When the object position is known, the robot picks up the object and transports it to the target position.

The Fig. 1 depicts common steps of a bin picking automation process in industrial applications. First of all a visual capture device takes a picture of an industrial scene. The most important component of the bin picking problem is the algorithm to localize an object in the scene. In order to pick up the object with the gripper, the robot has to know the exact position of the object. Therefore, an adequate grasp point must be defined. After that the system has to detect possible collision points with the surrounding environment and find a way to guide the robot to the target position, where the object has to be placed. This process is repeated for each object in the bin.

The approach in this paper focuses on the object localization step, which is the most challenging step in the whole process. Object recognition and object localization has a long history in two dimensional image processing[6],[7]. Due to the lack of the third dimension in an image the position of an object in the scene can not be fully determined. By using range data the distance to the camera is known, so our approach is able to use this information to find objects in a 3-dimensional scene with high accuracy. In the approach from [8], 3D-Models are projected in the image plane and compared to the image of the scene in order to estimate a possible pose. The proposed work in this paper extends this approach to range images. We introduce a simulation of a full laser scanning process. Industrial laser range sensors are modeled to transfer a cad aided design (CAD) model to a 2.5D range data representation. This virtual range data is compared to the real range data of the scene.

Our main contribution is to handle nearly all kind of objects without restrictions. We compare the appearance of a single object in a simulated scene with the real scene. Therefore we simulate a laser range sensor with all its features. This leads to a better accuracy and robustness of the whole system. Our second contribution is the definition of a flexible coarse-to-fine algorithm. Depending on the needed position accuracy, we can adjust the process time of the system by changing a threshold between the coarse pose estimation and the refinement process.

After the overview in section 2 the coarse pose estimation is introduced in section 3 to reject most of the impossible poses of the object in the scene. In the refinement step we use a modified Iterative Closest Point (ICP) algorithm to derive optimal solution with high accuracy. We conclude with upcoming extensions of our approach.

Manuscript received April 30, 2007. This work was supported in part by the company VMT GmbH Weinheim, Germany.

Kay Boehnke is a PHD student at the Communication Department, University Polytechnica Timisoara, Romania (e-mail: kay.boehnke@gmail.com).

II. System Overview

An overview of the proposed object localization system is shown in the Fig. 2. The object localization is separated into pose estimation and refinement. Because the refinement process has high computational cost, we introduce the pose estimation to reduce the number of possible poses. This hierarchical object localization is related to hypotheses and verification approaches[5] or a two step Coarse-to-fine algorithm [9].

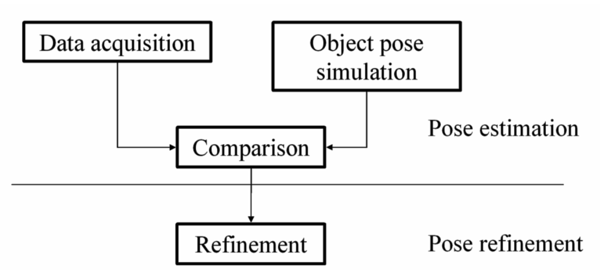

Fig. 2. Our object localization is divided into two different steps. The pose estimation delivers approximate positions of the objects. These candidates are verified in the refinement step.

Looking at the pose estimation, the input data from range sensors is compared to simulated range data. Therefore a simulated sensor delivers a virtual range image of the pose of an object representation. The result of the pose estimation is used to define the start positions for the pose refinement. The refinement step uses a modified registration algorithm to increase the accuracy. The components of our system are introduced in detail in the following chapters.

III. Object Pose Estimation

The purpose of the object pose estimation is to find adequate coarse positions of an object in the scene. This pre-selection is made to decrease the number of candidates for the refinement process. Because of this, the acquired data from the range sensor is compared to a simulated scene.

A. *Data acquisition*

One of our contributions is that we take features of real range sensors into consideration to adapt the simulated range sensor to real range sensors. Therefore this chapter will shortly introduce the data acquisition with industrial range sensors.

The most common data sources for industrial applications are still passive camera systems. Cameras provide a 2-dimensional projection of a scene, so no depth information can be obtained without any further processing [10]. With the help of stereo cameras or the solution of structured light sensors distance values can be determined[11]. Unfortunately many camera based solutions to get range data are having problems with their robustness and sensitivity on lightning conditions[5]. In [12] non- contact industrial laser range sensors are introduced. At the moment these active sensors are superior to other industrial measurement methods regarding their accuracy, costs and robustness beside stereo camera systems[5],[12]. For active laser distance measuring two major principles –the triangulation and time-of-flight (TOF)– are used in industrial applications. More or less TOF and phase measurement methods are long range technologies (over 1.0 meter) and triangulation based methods belong to close range methods. Most of these industrial sensors deliver a two-dimensional distance contour. In order to get a whole 2.5D scene representation as shown in Fig. 4(b) this sensor has to be moved over the real scene preferably in a linear way [13]. The distance axis of the sensors is directed towards the objects in the bin as shown in Fig. 4(a). The sensor moves from the start point to the end point with specific incremented steps. The step width is connected with the scan frequency of the sensor. A range image made with this setup is shown in Fig 5.

B. *Object pose simulation*

The object pose simulation creates a virtual range image (VRI) with help of a simulated sensor and a virtual scene. The components of the object pose estimation are shown in Fig. 3.

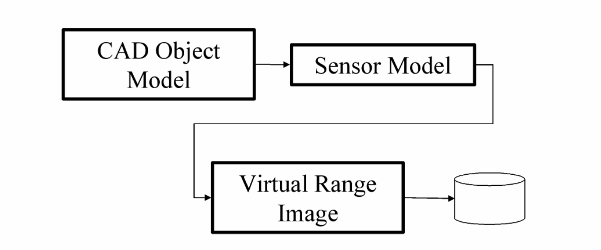

Fig. 3. A virtual range image(VRI) is generated from the CAD-model of the object and the model of the used sensor in the real scene. The database of the virtually scanned CAD- models is generated offline to increase the performance.

An object model is placed in a virtual scene. In most industrial applications CAD-models of the objects already exist. If not, the object model can be created manually. So in this work we assume the object is known as CAD-model. A common format to store CAD-models relies on triangulated points. This triangle mesh is stored in the often used and very common STL (Structured Triangle List) file format. A big advantage of a triangulated mesh representation is the simplified calculation in the sensor simulation. The CAD-based object model is used to generate virtual range images with the help of the simulated range sensor. The sensor models adopt all properties of the real sensors. Like mentioned above, range sensors deliver a contour so a linear movement is also necessary for the sensor simulation to get a full 2.5D range image. To compare the results from simulation to the real image the resolution of the data in moving direction should be similar. Therefore, the properties of the scanning process of the objects in the scene must be known. This includes the distance between ground and sensor (in Z) and the direction of scanning (in Y). Moreover, all parameters and properties of the sensors must be known.

Time-of-flight (TOF) laser distance sensors measure the distance between the object and the light source along a light beam. This beam is moved incrementally with a parameterized angle step width in the orientation of x. The result is a radial distance. For each angle, a distance value is measured. To ensure the comparability between different sensor data these values are converted into Cartesian coordinates, but this is not necessary in many cases. The simulated light beams start in the origin of the sensor coordinate system and the distance values are calculated between this point and the closest point to the object. The angle steps of the light beam can be found in the rotation of the Y-axis (R_y).

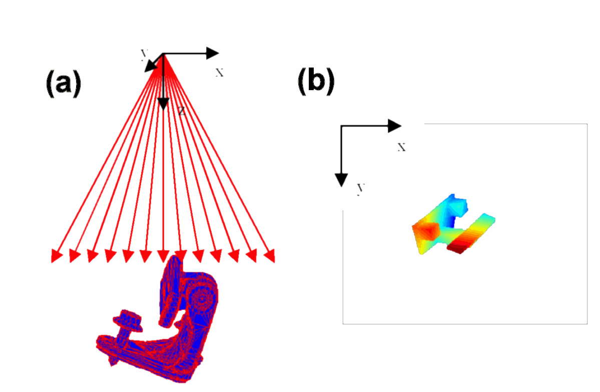

Fig. 4. (a) This figure shows the scanning process of a virtual TOF sensor. The sensor is moved along the y-axis.
(b) The result of the virtual scan is the shown VRI. This distance z is converted to the color space in this image.

Triangulation based laser sheet of light sensors consist of a stripe projector to emit a laser line to the object. A camera grabs the projected line and with the help of the geometric configuration the distance can be acquired. More details can be found in [12].

We have to separate the sensor simulation for TOF laser distance sensors and the triangulation based laser sensors, because both measurement principles deliver different range images even if they are observing the same scene. The characteristic of triangulation based laser distance sensors is the fact, that their receiving device is not placed in the same position like the laser, which sends a laser line to the scene. Because of this displacement most depth images of triangulation based laser distance sensors suffer from occlusions coming from their measurement setup. So in most of our application scanning TOF laser distance sensors are used.

With its simulated geometric configuration the sensor model determines the distance between the object and the laser source in the same way. The distance is calculated for every maximum value in the camera row index. This results in a distance vector for every projected and acquired laser line. By virtually moving the simulated sensor this simulation results in an equidistance depth map of the scene. The sensor models for industrial TOF- and triangulation based sensors produce a distance vector per scanning step. These distance vectors can be interpreted as rows of a 2.5D range image. The sensor model virtually scans the object and produces a range image in the same way like the real scene.

If a CAD-model is put in the measurement range of this virtual sensor and the virtual sensor is moved over this model in parameterized steps, a virtual 2.5D range image is produced in this virtual scanning process. This virtual range image contains one surface of the 3D-Model. This VRI shows this surface of the model, which surface normals are orientated towards the sensor. Such a virtual 2.5D scan is shown in Fig 4(b).

For every possible position and orientation of the object model a VRI is produced. This VRI is indexed with a known position and orientation of the model in the sensor measurement space and stored to a database. The process of virtual scanning is very time consuming. Therefore, it is necessary to generate all range images offline. This is called the VRI database. The database contains scanned models in the form of range images for defined positions and orientations. Depending on the application, the positions and orientations we need exorbitant space to store these range images. In most cases we set limits to the degrees-of-freedom and store only VRI of defined step widths. But in general this process must be done for all needed positions and orientations for every kind of object. A VRI is generated for every position of every object in scene. These positions differ only in a few millimeters steps depending on the resolution of the real sensor and the size of the real object and the scene. For every position exist 3 possible rotations for the object. Due to the limit of computer power and storage, the proposed simulation is limited to “coarse” or “rough” poses. The number of the coarse poses in the database mainly depends on the needed accuracy, the needed robustness, the sensor resolution and the resources of the system. Using this VRI database the model based approach changed to a view centered approach.

C. *Comparison*

The aim of coarse pose estimation is the reduction of possible solutions which will be found in the pose refinement.

The position and orientation of the object is estimated by range data comparison. Every VRI is compared to the real range image (RRI) with the following error function:

$$Error = \frac{1}{N}\sum_{i=0}^{X}\sum_{j=0}^{Y}\left|Z_1(i,j)-Z_2(i,j)\right| \qquad (1)$$

The error is defined as the mean of the difference between every distance value Z_1 of the simulated object and the distance value Z_2 of the scene. The error depends mainly on the position X, Y, Z and the rotation around the axis R_x, R_y,

R_z of the simulated object and the limits of degrees-of-freedom of the object. Different VRI's for different kind of objects are compared to the RRI in the same way. So the object classification is integrated in the step of object localization. In this step of coarse pose estimation we used a fixed increment of ΔX, ΔY, ΔZ, which depends mainly on a-priori knowledge of the object position in the scene. For example, in the case of the door joints in Fig. 5 we assumed that the distance Z do not have to be changed in major steps, because all door joints laying on the bottom of the box. The VRI consisting of the surfaces representation of the 3D model is compared to the real range image (RRI) of the scene made by the sensor which is shown in Fig. 5.

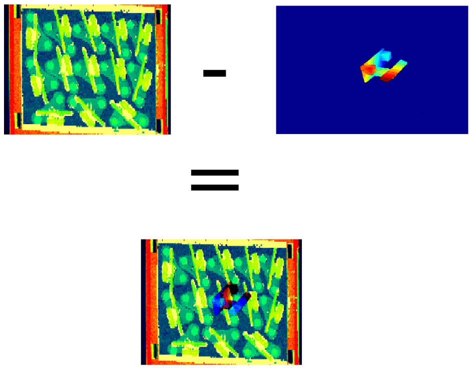

Fig. 5. The distance values of the VRI are compared to the distance values of the real range image in their position. The result is a matching error for this VRI.

One advantage of this pre-selection of matching positions is the fact, that all VRI can be calculated offline and stored in a database. So the process for our coarse pose estimation can be summarized in that way:

- the RRI is delivered by the sensor
- all VRI in the database (one for each possible pose) are compared to the RRI
- the best VRI candidates are selected for pose refinement

The here used error function returns an error value. If the error value is low the VRI matches with the RRI. Because of the fact, that all VRI are compared to the RRI, each VRI gets an error value. The VRI candidates with the lowest error values are selected for pose refinement. The number of best matching VRI's can be limited by an error-threshold or a fixed number of VRI candidates. We decide to use a combination of both: Assuming we know the maximum number of objects in the scene n_o, our experiments show that a good choice to limit the number n_c of VRI candidates to

$$n_C = 1.2 \cdot n_O \tag{2}$$

The error threshold depends on the object size and the increment size. We take the VRI candidates within the best 10-20% of all error values in the coarse pose estimation process which have to be a good initial threshold proven in our experiments.

This pre-selection results in 10-15 VRI candidates in the application. These VRI candidates are delivered to the pose refinement process, starting with the best matching candidate.

IV. Object Pose Refinement

In the previous chapter the coarse pose estimation creates an error value for every pose. The best VRI candidates were chosen and used as input for the pose refinement to find the best matching candidate.

The task of the pose refinement is to find a nearly exact match between the object in the scene and the simulated image. The pose refinement in our case is very similar to the registration process which generally deals with the determination of a transformation. The registration process aligns a representation of an object to another pose of this object. In this chapter the idea of "distance matching" of the pose estimation is extended to an iterative comparison to find the exact position for all VRI candidates.

A. ICP

The classical and most commonly used algorithm for rigid transformations is the Iterative Closest Point algorithm (ICP) [14],[15]. Important works are [16],[17]. Because of the slow convergence speed, the ICP was improved by many researchers [18],[19]. Nevertheless the ICP is still one of the most popular registration algorithms. Real time implementations [18],[20] show the potential of the ICP-Algorithm.

As the name already implies, the ICP is an iterative algorithm which determines its parameters with the help of a mathematical minimization. The algorithm minimizes the mean square error of the point distance in several iteration steps. To abort the iterations a threshold or a maximum number of iterations is implemented. The algorithm converts monotonously to a local minimum. Therefore, the knowledge of an approximate initial solution is important for the success of the method, which is provided in the first step of our approach: the coarse object pose estimation.

The Iterative ICP algorithm is a registration method to transform the view of an object to another view of the same object or another object.

The set of points *M* of the model is defined as:

$$M = \{m_i\} \quad m_i = (x_i, y_i, z_i), i = 0,1,...N_i \tag{3}$$

The set of points *P* of the scene is defined as:

$$P = \{p_j\} \quad with \quad p_j = (x_j, y_j, z_j), j = 0,1,...N_j \tag{4}$$

The set *M* consists of points with the coordinates x,y,z in the coordinate system of our simulated sensor. Due to the fact that the simulated and real coordinate systems are equal, we get the rigid transformation between the VRI and RRI by minimizing the error E

$$E = \sum_i \left\| m_i - R(p_i) - t \right\|^2 \tag{5}$$

To find the rotation R and the translation t we use the closed form solution with the help of unit quaternions[21].

An iteration of the ICP is separated into three steps. At first, every point of one dataset is assigned to a point in the other dataset. Hereby, every point is assigned to the Euclidean closest point of the model. The corresponding points of two datasets are usually not known, so the ICP guesses the corresponding points by determining the smallest distance between two points in the two datasets. According to Besl and McKay[16], a point-to-point distance is used for the calculation of the transformation. Chen and Medioni[17] developed simultaneously a similar algorithm for point-to-surface distance. We use a point-to-point metric and implemented a kD-Tree to increase the processing time of the ICP algorithm[18]. The elements of the normalized eigenvector of the largest positive eigenvalue of the matrix Q

$$Q(\textstyle\sum_{PM}) = \begin{bmatrix} tr(\sum_{PM}) & \Delta^T \\ \Delta & \sum_{PM} + \sum_{PM}^T - tr(\sum_{PM}) I_3 \end{bmatrix} \tag{6}$$

correspond to the elements of the unit quaternion. (Please refer to [16],[18] for details).

We improve the approach of Horn by solving this eigenvalue problem with a QR-decomposition, which expose to be a fast algorithm in our previous experiments [22]. According to Horn [21] we get the rotation matrix R from resulting unit quaternion. The result is a rigid transformation in the used coordinate system. An iteration step finishes by applying the resulting transformation to one dataset.

The ICP algorithm is used for every VRI candidate. The resulting error value after applying the ICP is calculated according to (5). The best VRI candidate is selected and the object coordinates in the sensor coordinate system and used as the final result in our object localization step.

V. Further Extensions and Conclusion

This work focuses on range images provided by range sensors and uses a model-based scalable hierarchical system without the need of segmentation or feature extraction. Our approach was successfully implemented and tested in its first version to handle a real industrial application and to proof the potential of our concept. We achieve an overall accuracy of ±1mm in translation and a rotation accuracy lesser than ±5 degrees. Our intention is to provide a basic system, which can be used with any kind of object. But this object has to be known preferred as CAD- representation. Industrial robotic bin picking includes -beside the object localization- an adequate sensor selection, an application-invariant localization algorithm, a robot control interface, a grasp point definition and a collision avoidance strategy. Merging all these components in a system will be one of our challenging tasks in the very next future.

Our approach has potential to meet different requirements and solve many problems with its universality. The system does not use any segmentation algorithms, but uses 3D information which is aligned to the input data in a hierarchical system.

The system framework offers many possible extensions. Due to the incremental process object positions can be verified and tracked over all steps of the process. This increases the robustness and reduces the computational costs. By using range data the complexity of the object localization process is increased. The complexity of the object localization depends also on the chosen algorithm and the complexity of the object. The main problem of our algorithm is the high computational cost, if we have very complex objects models and sensors with high resolutions. But depending on the application and the used PC, we can change this computational time-memory-trade off by increasing the number of pre-calculated VRI's in our database. We are not limited to only one object, because we can store as many objects in our database as we want. This is one big advantage of the whole system.

The scalability of our approach offers a great potential in the future. Starting with the coarse pose estimation process, we are able to adjust the needed accuracy with simple changes in the position and orientation step width. New sensors with higher resolution can be modeled without any problem, even if sub sampling is necessary due to system limits.

The approach of coarse pose estimation could be extended to an iterative process and could be used as a refinement method. This minimization process can be accelerated applying minimization methods like Hill climbing, simplex minimization and other minimization algorithms.

One of the promising improvements in the near future will be the modification of the ICP algorithm using the features of the used sensors. Therefore we will change the point-to-point metric to the so called reversed calibration metric introduced by Blais and Levine[23].

All real range sensors deliver range images with measurement errors, reflections, and noise[12]. To increase our accuracy we will take these additional features into consideration. As mentioned above our error function in the coarse estimation step can be changed to increase the accuracy, which has to be analyzed in our next research activities. Our approach is characterized by a flexible coarse-to-fine algorithm, which can be used to find any kind of objects in range data provided that these objects are known to our system. The density of positions and the rotational degree of freedom of the object to create a VRI is the most important parameter to adjust the level in our flexible coarse-to-fine algorithm. The simulation of real scenes offers the possibility to use our approach in many scenarios. The described two-step object localization will be used in a flexible system to cover a high percentage of applications in robotic bin picking.

ACKNOWLEDGMENT

The author wants to thank M. Otesteanu, P. Roebrock, M. Kleinkes, W. Neddermeyer, W. Winkler, and K. Lehmann for their support in this project.

REFERENCES

[1] M. Hashimoto, K. Sumi, "3d object recognition based on integration of range image and grey-scale image", *in 1999 Proc. British Machine Vision Conf.*, pp. 253–262

[2] M. Kavoussanos, A. Pouliezos, "Visionary automation of sack handling and emptying", *IEEE Robotics and Automation Magazine*, vol 7, pp. 44-49, 2000

[3] B.K.P. Horn and K. Ikeuchi , "Picking parts out of a bin", MIT, Cambridge Res. Lab., Rep. AIM-746, 1983

[4] A. Kak, J. Edwards, "Experimental state of the art in 3D object recognition and localization using range data*", in 1995 Proc. of the IEEE IROS workshop on vision for robots*, 1995

[5] D. Katsoulas, "Robust recovery of piled box-like objects in range images", Ph.D. dissertation, Dept. Computer Science, Freiburg Univ., Germany, 2004

[6] J. Andrade-Cetto and A. C. Kak, "Object recognition", in *Wiley Encyclopedia of Electrical and Electronics Engineering*, J. G. Webster Ed. New York: John Wiley & Sons, 2000, pp. 449-470.

[7] S. Dickinson, "Object Representation and Recognition", in *What is Cognitive Science?,* E. Lepore and Z. Pylyshyn Ed. Oxford: Basil Blackwell, 1999, pp. 172-207.

[8] D.G. Lowe, "Fitting parameterized three-dimensional models to images", *IEEE Trans. Pattern Anal. and Machine Intell.*, vol 13, pp. 441-450, 1991

[9] R. Campbell and P. Flynn, "A Survey of Free-form Object Representation and Recognition Techniques", in *Computer Vision and Image Understanding*, Vol 81, pp. 166-210, 2001

[10] M. Sonka,V. Hlavac,R. Boyle, "Image Processing, Analysis and Machine Vision", Chapman & Hall Computing, 1993

[11] K. Rahardja and A. Kosaka: "Vision-Based Bin-Picking: Recognition and Localization of Multiple Complex Objects Using Simple Visual Cues", in *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Osaka, 1996, pp. 1448-1457

[12] K. Boehnke, "3D Sensors", Seminar paper, Univ. Timisoara, Romania, 2006

[13] J. Park and G.N. DeSouza, "3D Modeling of Real-World Objects Using Range and Intensity Images ", in *Innovations in Machine Intelligence and Robot Perception*, S. Patnaik, L.C. Jain, G. Tzafestas and V. Bannore Ed.,New York: Springer-Verlag, 2005

[14] D. Simon, "Fast and Accurate Shape-Based Registration", Ph.D. dissertation, tech. report CMU-RI-TR-96-45, Robotics Institute, Carnegie Mellon University, 1996

[15] M. Wheeler M, "Automatic modeling and localization for object recognition", Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, PA, 1996

[16] J.P. Besl, N.D. McKay, "A Method for Registration of 3-D Shapes", in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol 14, pp. 239-256, 1992

[17] Y. Chen, G. Medioni, "Object Modelling by Registration of Multiple Range Images" in *Image and Vision Computing*, vol. 10, pp. 145-155, 1992.

[18] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm", in *Proc. of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pp. 145–152, 2001

[19] K. Boehnke, "ICP Algorithms", Seminar paper, University Timisoara, Romania, 2006

[20] T. Jost and H. Huegli, "A Multi-Resolution ICP with Heuristic Closest Point Search for Fast and Robust 3D Registration of Range Images", in *4th Inter. Conf.on 3-D Digital Imaging and Modeling*, 2003, pp. 427-433

[21] B. Horn, „Closed–form solution of absolute orientation using unit quaternions", in *Journal of the Optical Society of America*, vol. 4, pp. 629-642, 1987

[22] A. Zenzes and K. Boehnke, "Implementierung des ICP-Algorithmus", seminar paper, Dep. Electron. Eng., Univ. of Coop. Education Mannheim, 2006

[23] G. Blais, M.D. Levine, "Registering Multiview Range Data to Create 3D Computer Graphics" in *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, pp. 820-824, 1995