Third International Conference on Information Science and Technology
March 23-25, 2013; Yangzhou, Jiangsu, China

Probabilistic 3D ICP Algorithm Based on ORB Feature

Zhe Ji, Fan Zhou, Xiang Tian, Rongxin Jiang and Yaowu Chen

Abstract—**Aligning the 3D point clouds is considered as a crucial step to build consistent maps from unknown environment in SLAM problem of mobile robotics. However, ICP algorithms for aligning the point clouds usually ignore the valuable visual information contained in the point clouds and only model surface structure from the "model" scan. This paper presents a new ICP algorithm incorporating the visual ORB feature. This approach is based on a probabilistic ICP algorithm that takes into account both scans for RGB images along with the depth information from Kinect sensor. Experiments are carried out on real world scenes and results show that the new approach improves the accuracy and saves time of the registration.**

I. Introduction

AUTOMATICALLY and incrementally building the consistent 3D maps of the unknown environment is the main objective of SLAM(simultaneous localization and mapping) in mobile robotics [1]. 3D point clouds, which are often generated by the laser scanner, are suitable for retrieving the 3D map information and the camera trajectory from the environment. However, they discard many valuable features represented by the images. In contrast, many state-of-the-art monocular camera image algorithms can be applied in RGB images which contain rich visual information. However, a single RGB image has no depth information. It would be an extra computational burden to extract the depth information from the image pairs. It's even harder when the light condition is very poor, such as indoors.

Thanks to Microsoft RGB-D camera named Kinect, we now can obtain the RGB image and the depth image synchronously and easily. Kinect camera can capture the registered images in 8-bit VGA resolution 640×480 pixels and the depth image at a frame rate of 30Hz. As an input sensor, RGB-D cameras can be used to build dense 3D consistent maps [2].

ICP(iterative closest point) algorithm is a robust and efficient approach to handle the point clouds registration problem. ICP was firstly introduced by Besl and McKay [3]. They directly proposed the registration problem of 3D shapes described either geometrically or with point clouds can be solved by minimizing the Euclidean distance between the corresponding points of the two scans. Almost at the same time, there were other two papers that introduce the ICP independently. Chen and Medioni [4] further introduced the point-to-plane variant of ICP by considering the fact that most range data are sampled from a locally planar surface. Zhang [5] also described ICP by adding a robust approach of removing outliers in the corresponding matching points between the two scans.

One of the main advantages of the ICP methods is simplicity. Furthermore, the speed of searching the corresponding points of ICP algorithm is greatly improved when implemented by k-d trees. On the other hand, a main drawback of ICP is because of the implicit assumption of the theoretical requirement that the points are taken from a known geometric surface rather than measured [3]. For this reason, Chen and Medioni [4] suggested using a point-to-plane distance error as the objective function instead of point-to-point distance.

Due to a substantial theoretical work, many ICP variants are integrated with the probabilistic methods. Because sensors and data generated from range scanners are noisy, the probabilistic methods of modeling the real world give better performance than the deterministic methods. Biber et al. [6] introduced a probabilistic model by assuming that the second scan is generated from the first through a random process. Haehnel and Burgard [7] propose a ray-tracing operation to correctly handle occlusions. Their model incorporates measurement noise and random noise in order to deal with errors typically found in 3D range scans. Segal et al [8] proposed a generalization of the ICP algorithm which assumes all measured points are drawn from Gaussians. Maximum likelihood estimation method is then used to estimate the transformation. They have shown good results with 3D point clouds scanned by a laser scanner.

This paper based on the proposed probabilistic ICP algorithms for point clouds which captured by the 3D scanners such as a ToF 3D laser scanner, put forward a new probabilistic 3D ICP algorithm to align RGB-D images. First, we detect and compute the ORB [9] features of the two RGB images. Afterwards, the 2D ORB features are mapped into 3D with the depth information, and then we can easily estimate the transformation in 3D point clouds from the matching ORB features. Finally, the transformation obtained previously is used as an initial guess to compute the final transformation between the two 3D point clouds by the means of an improved ICP algorithm that introduces a probabilistic method.

II. Generating Transformation based on ORB Feature of RGB-D Images

Manuscript received December 30, 2012. This paper is supported by the National Natural Science Foundation of China under Grant 40927001, the project of Key Scientific and Technological Innovation Team of Zhejiang Province under Grant 2011R09021-02 and the Fundamental Research Funds for the Central Universities.

The authors are with the Institute of Advanced Digital Technology and Instrument, Zhejiang University, Hangzhou 310027, China, and Zhejiang Provincial Key Laboratory for Network Multimedia Technologies, Hangzhou 310027, China (e-mail: jz@zju.edu.cn; fanzhou@mail.bme.zju.edu.cn; xiang.t@163.com; rongxinj@zju.edu.cn; cyw@mail.bme.zju.edu.cn)

A. Computation Of Image Feature Using ORB

Feature detection and matching is basic topic in the field of computer vision. The SIFT [10] key point detector and descriptor that proposed for many years is an effective method to handle this problem. However, SIFT requires a large quantity of computation. ORB descriptor, which proposed by Ethan Rublee et al [9] has demonstrated much more effective computation compared with SIFT and SURF [11] and shows the same performance.

ORB applies FAST [12] detector on every level in the pyramid of the image to produce the features. Moreover, ORB adds extra orientation information for FAST. The corner orientation is computed based on the intensity centroid as the measure. Given the second-order moments of an image patch as:

$$\boldsymbol{M}_{ij} = \sum_x \sum_y x^i y^j I(x,y) \tag{1}$$

where $I(x,y)$ means the image intensity at location (x,y). The centroid is:

$$c_x = \frac{\boldsymbol{M}_{10}}{\boldsymbol{M}_{00}}, c_y = \frac{\boldsymbol{M}_{01}}{\boldsymbol{M}_{00}} \tag{2}$$

Then the vector from the corner's center to the centroid of the patch is constructed. The orientation of the patch can be described as

$$\begin{aligned} \theta &= \arctan\left(\frac{c_y}{c_x}\right) && \text{if color} = \text{bright} \\ \theta &= \arctan\left(\frac{c_y}{c_x}\right) + 180^\circ && \text{if color} = \text{dark} \end{aligned} \tag{3}$$

Next, ORB uses the BRIEF [13] to extract the descriptors along the orientation of the patch defined by (3). Given the matrix representing any feature set of n binary tests at location (x,y):

$$\boldsymbol{B} = \begin{pmatrix} x_1 & \ldots, & x_n \\ y_1 & \ldots, & y_n \end{pmatrix} \tag{4}$$

and the rotation matrix:

$$\boldsymbol{R} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \tag{5}$$

we get the oriented binary feature:

$$\boldsymbol{B}_o = \boldsymbol{RB} \tag{6}$$

From (6), we construct the ORB feature:

$$ORB(patch,\theta,n) = BRIEF(patch,n) \big| (x_i, y_i) \in \boldsymbol{B}_o \tag{7}$$

where $BRIEF(patch,n)$ means the n-dimensional BRIEF descriptor for *patch* of the image.

Fig. 1. The matching ORB features of the two images.

After finding the less correlated correspondence with brute-force matching, we get the final matching ORB feature points as shown in Fig. 1.

B. Mapping 2D ORB Feature To 3D

Pixel of the RGB image has one-to-one correspondence to the 3D point. Given the 2D ORB feature point $ORB_{2D}(x,y)$, we can get the corresponding 3D ORB point along with the depth information which captured by Kinect camera as follow:

$$ORB_{3D}(x,y,z) = ORB_{2D}(x,y) + depth(x,y) \tag{8}$$

Based on this information we get a coarse transformation guess by solving a regular Least-Square problem.

III. Probabilistic ICP based on ORB Feature

By the projective transformation, we can generate the point clouds from the RGB image and its depth image. The process is easy and straightforward. The example of colorful point clouds is shown in Fig. 2.

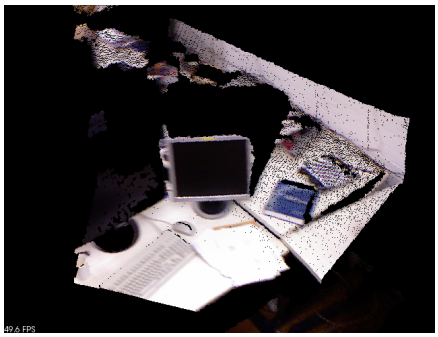

Fig. 2. The 3D RGB point clouds generated by the RGB-D image.

ICP technique is accurate and reliable method applied for image registration. Aside from ICP, feature-based image registration algorithm relying on the visual information alone, depends on the camera resolution and light condition of the image, camera calibration and so on. It is limited compared with ICP.

The goal of ICP is to estimate the rigid body transformation iteratively that best align a cloud of scene points with a geometric model that are both arbitrary form surfaces. The algorithm begins with two data sets, an initial estimate for the transformation and a stopping criterion for the iterations. It outputs an improved estimate of the matching transformation [1]. Furthermore, an improved ICP algorithm introducing the probabilistic method and using a new distance relating to Gaussian distribution as the metric error is proposed.

A. Classic ICP

Given two overlapping point sets in N dimension in $\mathbb{R}^m$, a scene data $P \triangleq \{\boldsymbol{p}_i\}_{i=1}^{N_p} \left(N_p \in \mathbb{N}\right)$ and a model data $Q \triangleq \{\boldsymbol{q}_j\}_{j=1}^{N_q} \left(N_q \in \mathbb{N}\right)$. Our problem is to compute an appropriate transformation $\boldsymbol{T}$, with which the scene data P best aligns the model data Q. The key process of the standard ICP algorithm can be summarized in the following steps.

1) Let T_0 be the initial transformation, usually assigned as identity matrix
2) Repeat $i = 1 \rightarrow N$ or the result is converged to the

matching transformation

a) Select points from scene data P and try to match these points with the model data Q to build up the correspondences.

b) Compute the closed-form matching transformation by minimizing distance between the corresponding points.

The minimizing distance process can be written as the following Least-Square problem

$$\boldsymbol{T} = \underset{T,j\in\{1,2,\ldots,N_q\}}{\arg\min}\left(\sum_{i=1}^{N_p}\| \boldsymbol{T}(\boldsymbol{p}_i)-\boldsymbol{q}_j \|_2^2\right) \tag{9}$$

This objective function is a general registration function. The transformation $\boldsymbol{T}$ can represent rigid transformation, scaling transformation, affine registration, etc.

Let $\boldsymbol{T}$ in (9) be combined as a rotation matrix and a translation vector, we minimize the mean squared error instead. Therefore, the registration represented by (9) can be described as

$$\boldsymbol{T} = \underset{T,j\in\{1,2,\ldots,N_q\}}{\arg\min}\left(\frac{1}{N_p}\sum_{i=1}^{N_p}\| \boldsymbol{R}\boldsymbol{p}_i+\boldsymbol{t}-\boldsymbol{q}_j \|_2^2\right) \tag{10}$$

$$\text{s.t. } \boldsymbol{R}^{\mathrm{T}}\boldsymbol{R}=\boldsymbol{I},\ \det(\boldsymbol{R})=1$$

where $R\in\mathbb{R}^{m\times m}$ is a rotation matrix, and $\boldsymbol{t}\in\mathbb{R}^{\mathrm{m}}$ is a translation vector. From minimizing (10), the closed form solution can be obtained.

B. *Point-to-plane ICP*

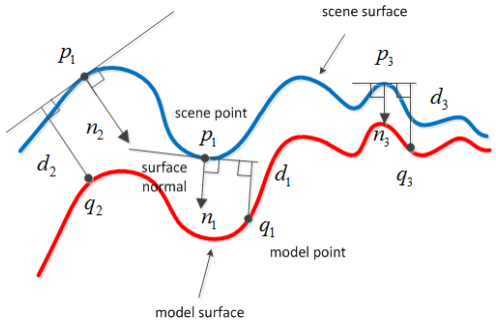

Fig. 3. Point-to-plane distance between two surfaces.

Point-to-plane ICP is a more robust and accurate variant more than classic ICP. Unlike the classic point-to-point ICP using the sum of the squared distance between points in each corresponding pair, the point-to-plane ICP minimized the metric error that represented the sum of the squared distance between a point and the tangent plane at its corresponding point as shown in Fig. 3. The point-to-plane ICP usually uses nonlinear least square methods to solve the metric error. The improvement is to replace objective function (10) with:

$$\boldsymbol{T} = \underset{T,j\in\{1,2,\ldots,N_q\}}{\arg\min}\left(\frac{1}{N_p}\sum_{i=1}^{N_p}\| (\boldsymbol{R}\boldsymbol{p}_i+\boldsymbol{t}-\boldsymbol{q}_j)\bullet\boldsymbol{n}_j \|_2^2\right) \tag{11}$$

where $\boldsymbol{n}_j$ is the unit surface normal vector at $\boldsymbol{q}_j$.

A. *Probabilistic Point-To-Plane ICP Incorporating ORB Features*

Our ICP method to refine the point clouds generated by the combination of 2D RGB information and depth information is based on generalized ICP (GICP) algorithm proposed by Segal et al [8]. The GICP algorithm combines the classic ICP and point-to-plane ICP algorithms into a single probabilistic framework. Then GICP incorporates the information of the planar surface structure from both scans instead of just the model scan as the normal point-to-plane ICP does.

For applying the probabilistic method to ICP, it is considered that the scene point clouds $P=\{\boldsymbol{p}_i\}_{i=1,\ldots,N}$ and the model point clouds $Q=\{\boldsymbol{q}_i\}_{i=1,\ldots,N}$, which are already indexed according to their correspondences, follow the Gaussian distribution respectively $\boldsymbol{p}_i\sim N\left(\hat{\boldsymbol{p}}_i,\boldsymbol{C}_i^{\boldsymbol{P}}\right)$ and $\boldsymbol{q}_i\sim N\left(\hat{\boldsymbol{q}}_i,\boldsymbol{C}_i^{\boldsymbol{Q}}\right)$, where $\left\{\boldsymbol{C}_i^{\boldsymbol{P}}\right\}_{i=1,\ldots,N}$ and $\left\{\boldsymbol{C}_i^{\boldsymbol{Q}}\right\}_{i=1,\ldots,N}$ are the covariance matrices associated with the measured points. The ideal transformation is $\boldsymbol{T}^*$, which satisfy the equation $\hat{\boldsymbol{q}}_i=\boldsymbol{T}^*\hat{\boldsymbol{p}}_i$. The distance error between the two point sets then can be described as

$$\begin{aligned}\boldsymbol{d}_i^{T^*} &\sim N\left(\hat{\boldsymbol{q}}_i-\left(\boldsymbol{T}^*\right)\hat{\boldsymbol{p}}_i,\boldsymbol{C}_i^{\boldsymbol{Q}}+\left(\boldsymbol{T}^*\right)\boldsymbol{C}_i^{\boldsymbol{P}}\left(\boldsymbol{T}^*\right)^{\boldsymbol{T}}\right)\\ &= N\left(0,\boldsymbol{C}_i^{\boldsymbol{Q}}+\left(\boldsymbol{T}^*\right)\boldsymbol{C}_i^{\boldsymbol{P}}\left(\boldsymbol{T}^*\right)^{\boldsymbol{T}}\right)\end{aligned} \tag{12}$$

By assuming $\boldsymbol{p}_i$ and $\boldsymbol{q}_i$ to be independent, we can using the Maximum Likelihood Estimation (MLE) to compute the transformation, $\boldsymbol{T}$ can be represented as

$$\boldsymbol{T} = \underset{T,i\in\{1,2,\ldots,N\}}{\arg\max}\prod_i p\left(\boldsymbol{d}_i^{T^*}\right) = \underset{T,i\in\{1,2,\ldots,N\}}{\arg\max}\sum_i \log p\left(\boldsymbol{d}_i^{T^*}\right) \tag{13}$$

which can be simplified:

$$\boldsymbol{T} = \arg\min\sum_i \boldsymbol{d}_i^{T}\left(\boldsymbol{C}_i^{\boldsymbol{Q}}+(\boldsymbol{T})\boldsymbol{C}_i^{\boldsymbol{P}}(\boldsymbol{T})^{T}\right)^{-1}\boldsymbol{d}_i^{T} \tag{14}$$

In fact, the point set is not only an arbitrary surface but a collection of surface sampled by the sensor. In a word, it's a sample 2-manifold in 3-space. Therefore, the covariance in the surface normal direction is considered very low. A point with q has the covariance matrix along its surface normal direction:

$$\begin{pmatrix}\varepsilon & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}$$

where ε is a very small constant.

Letting $\boldsymbol{\mu}_i$ and $\boldsymbol{v}_i$ be the normal vectors at the point at $\boldsymbol{p}_i$ and $\boldsymbol{q}_i$, $\boldsymbol{R}_x$ be the rotation that transform the normal vector to the point vector, then the covariance matrix can be computed as:

$$\boldsymbol{C}_i^{\boldsymbol{P}} = \boldsymbol{R}_{\mu_i}\cdot\begin{pmatrix}\varepsilon & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\cdot\boldsymbol{R}_{\mu_i}^{T} \tag{15}$$

$$\boldsymbol{C}_i^{\boldsymbol{Q}} = \boldsymbol{R}_{v_i}\cdot\begin{pmatrix}\varepsilon & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\cdot\boldsymbol{R}_{v_i}^{T} \tag{16}$$

With (15) and (16), we can compute the final transformation $\boldsymbol{T}$ from (14).

Summarizing the above, we propose the whole algorithm

of probabilistic point-to-plane ICP based on ORB features. We have utilized the notations below for representing the different data associated within the algorithm. A is referred to the data sets as a model, and B is referred to the data sets to be aligned.

$R_A = \{ r_a^i \in \mathbb{R}^3, i = 1,...,n \}$ RGB image as a model

$D_A = \{ d_a^i \in \mathbb{R}, i = 1,...,n \}$ depth image as a model

$P = \{ p^i \in \mathbb{R}^3, i = 1,...,n \}$ point cloud as a model

$Q = \{ q^i \in \mathbb{R}^3, i = 1,...,n \}$ point cloud as a scene

The cost function $\boldsymbol{T} = \arg\min \sum_i \| \boldsymbol{m}_i - \boldsymbol{T}\boldsymbol{p}_a^i \|$ of the classic ICP algorithm is replaced by the probabilistic objective function defined by (14).

TABLE I

PROBABILISTIC 3D ICP ALGORITHM INCORPORATING ORB FEATURE

Input: R_A, D_A, R_B, D_B

Output: The estimated rigid transformation $\boldsymbol{T}$

1: Detect and Extract the ORB descriptors of R_A and R_B.

2: Match the corresponding ORB descriptors between R_A and R_B, and save the results in the list L.

3: Compute the transformation $\boldsymbol{T}_0$ based on the list L.

4: $\boldsymbol{T} = \boldsymbol{T}_0$

5: **while** iter < MAX_ITERATIONS && not converged **do**

6: **for** $i = 1 \rightarrow N$ **do**

7: $m_i \leftarrow$ search the closest point in Q using $K-d$ tree with the distance $\boldsymbol{T}\boldsymbol{p}^i$

8: **if** $\| \boldsymbol{m}_i - \boldsymbol{T}\boldsymbol{p}^i \| < d_{\max}$ **then**

9: Save this point;

10: **else**

11: Discard this point;

12: **end if**

13: **end for**

14: $T = \arg\min \sum_i \boldsymbol{d}_i^{T^T} \left(\boldsymbol{C}_i^Q + (\boldsymbol{T}) \boldsymbol{C}_i^P (\boldsymbol{T})^T \right)^{-1} \boldsymbol{d}_i^T$

15: **end while**

IV. EXPERIMENTS

Our approach is compared with three other ICP algorithms, original GICP, GICP incorporating SIFT features, GICP incorporating SURF features, implemented for RGB-D images to verify the performance. Performance is analyzed in three aspects: RMSE error, convergence iterations and running time. The point clouds, which are generated by RGB-D image pair, contain about 250,000 valid points of all 307,200 points. We limit our ICP algorithm parameters as maximum iterations of 100, and 0.5 for max distance of matching corresponding points.

The example of image using our approach to align is shown in Fig. 4. In addition to the increased accuracy, the new algorithm not only takes advantages of visual ORB features of images, but also considers both images when computing the transformation. We can find more overlapping parts when alignment ended in Fig. 4(b) than it shows before in Fig. 4(a). Quantitative results are given below.

(a) Initial alignment (b) After alignment

Fig. 4. Example results of RGB-D image pair.

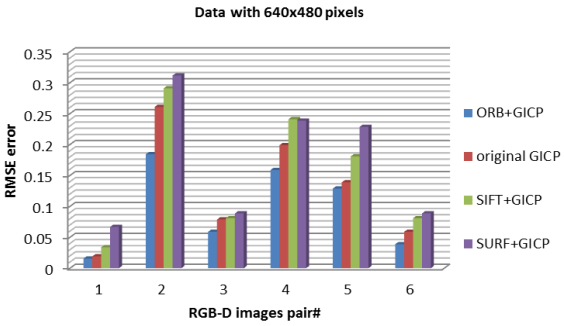

Fig. 5. Comparison of RMSE error.

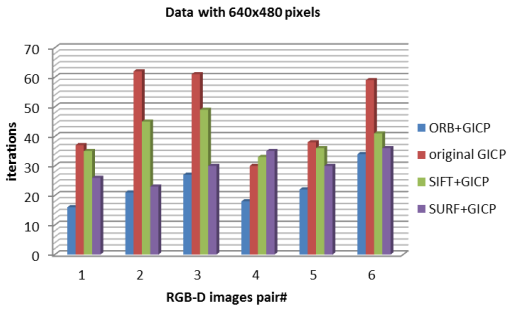

Fig. 6. Comparison of iterations of the convergence.

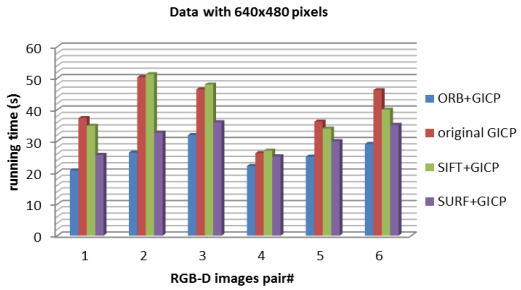

Fig. 7. Comparison of running time.

Six pairs of RGB-D images of real scene are used to demonstrate the performance of our algorithm. Fig. 5 shows RMSE error of images using different algorithms for the registration. From Fig. 5, it can be seen that our algorithm is more accurate than three others. Because our algorithm incorporating the visual ORB features, it is more accurate than original GICP algorithm. This is expected because we make use of the information captured by the Kinect camera more comprehensively. Fig. 6 shows that our approach makes the process of ICP greatly shorten. Because our method takes the coarse transformation generated by matching ORB features as an initial alignment, the cases with large distance

error are avoided at the beginning of the process of iteration. It can also be drawn that SURF descriptor is quicker than SIFT mentioned in [14] in detail. Original GICP without incorporating any visual feature is slowest in our test. Because of the lack of a reasonable initial guess, it costs much more iterations to converge. We can see the speed improvement of our approach more intuitively from Fig. 7. It is almost half-time of original GICP. GICP with SIFT and SURF are also much slower than our algorithm. Especially for GICP with SIFT, it almost consumes the same time with original GICP. This is not surprised if SIFT matching features generated bad initial transformation because of matching some corresponding key points in error.

V. CONCLUSION

This paper has presented a new ICP algorithm for aligning RGB-D images which can be applied in map alignment of SLAM. In the lack of a good initial guess, the performance of the ICP algorithm is not optimal if the registration is processed by using only the point clouds. Thus, the key idea of this approach is that we incorporate the visual ORB feature of the RGB images which can be captured by RGB-D camera like Kinect in our paper. Then we take this coarse transformation as a good initial for the next ICP process. Considering the noise in the data sampled from real world and the useful information of both scans, we implement our probabilistic ICP algorithm incorporating ORB feature based on Generalized ICP. Experimental results have validated the characteristics of our proposed algorithm.

For future work to utilize RGB-D camera more thoroughly, the color information of RGB-D image will be considered to be incorporated to get more robust ICP algorithm.

REFERENCES

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous Localization and Mapping : Part I," *Robotics Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99-110, June 2006.

[2] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "RGB-D mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 647–663, Feb. 2012.

[3] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[4] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, no. April, pp. 2724–2729, 1991.

[5] Z. Zhang. "Iterative point matching for registration of free- form curves and surfaces," *International Journal of Computer Vision*, vol. 13, no. 2, pp. 119–152, 1994.

[6] P. Biber, S. Fleck, W. Strasser. "A Probabilistic Framework for Robust and Accurate Matching of Point Clouds," *Pattern Recognition, Lecture Notes in Computer Science*, vol. 3175/2004, pp. 280-487, 2004.

[7] D. Haehnel, W. Burgard. "Probabilistic Matching for 3D Scan Registra- tion," *Proc. of the VDI-Conference Robotik*, 2002.

[8] A. V. Segal, D. Haehnel, and S. Thrun, "Generalized-icp," in *Robotics: Science and Systems*, 2009.

[9] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2564–2571, 2011.

[10] D. G. Lowe. "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[11] H. Bay, T. Tuytelaars, and L. Van Gool. "Surf: Speeded up robust features," in *European Conference on Computer Vision*, May 2006.

[12] E. Rosten and T. Drummond. "Machine learning for highspeed corner detection," in *European Conference on Computer Vision*, vol. 1, 2006.

[13] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. "Brief: Binary robust independent elementary features," in *European Conference on Computer Vision*, 2010.

[14] L. Juan and O. Gwun, "A Comparison of SIFT, PCA-SIFT and SURF," *International Journal of Image Processing*, vol. 3, no. 4, pp. 143–152, 2009.