A Point Cloud Alignment Algorithm Based on Stereo Vision Using Random Pattern Projection

Chin-Sheng Chen[1,*], Mei-Yu Huang[1], Chun-Wei Yeh[1], and Chien-Liang Huang[1]

[1]National Taipei University of Technology, Graduate Institute of Automation Technology, 1, Sec. 3, Zhongxiao E. Rd, Taipei, Taiwan R. O. C., 10608

(**Received** 30 September 2015; **Accepted** 27 November 2015; **Published on line** 1 March 2016)

*Corresponding author: saint@mail.ntut.edu.tw

DOI: 10.5875/ausmt.v6i1.1032

Abstract: This paper proposes a point cloud alignment algorithm based on stereo vision using Random Pattern Projection (RPP). In the application of stereo vision, it is rather difficult to find correspondences between stereo images of texture-less objects. To overcome this issue, RPP is used to enhance the object's features, thus increasing the accuracy of the identified correspondences of the stereo images. In the 3D alignment algorithm, the down sample technique is used to filter out the outliers of the point cloud data to improve system efficiency. Furthermore, the extracted features of the down sample point cloud data were applied in the matching process. Finally, the object's pose was estimated by the alignment algorithm based on object features. In experiments, the maximum error and standard deviation of rotation are respectively about 0.031°and 0.199°, while the maximum error and standard deviation of translation are respectively about 0.565 mm and 0.902 mm . The execution time for pose estimation is about 230ms.

Keywords: stereo vision random, pattern projection, object pose estimation, point cloud

Introduction

Over the past few years extensive improvements have been made to 3D vision technology, which has been widely applied in many fields including manufacturing and object recognition. To obtain 3D geometry information, binocular stereo vision systems use two cameras to view an object from different viewpoints, thus creating stereo images which can be used to calculate 3D geometry information through trigonometry. After obtaining the camera calibration parameters by observing different orientations of a 2D planar calibration pattern [1], the main task is to determine correspondences between the stereo images along the epipolar lines. Methods to determine such correspondence in stereo vision can be divided into two classes: global and local methods [2, 3], The global method relies on an iterative scheme and obtains the disparity on the basis of the minimization of a global cost function [4, 5]. This method can produce an accurate and dense disparity map, but at a high computational cost as the local method is based on the relation between each pixel and its adjacent pixels. Therefore, choosing a small window size is not suitable for texture-less objects because the low variation of intensity may produce an inferior match. Compared to the global method, the local method is less accurate, but can be deployed in many real-time applications. The main difficulty of the local method lies in choosing a window with an appropriate size and shape. Veksler [6] proposed an algorithm to choose a window size and shape by comparing the window cost for different window sizes. However, this approach requires the ignition of many parameters for window cost computation. Yoon et al. [7] proposed a method to adaptively adjust the weight of the pixels in a window. This method uses color similarity and geometric proximity to reduce image ambiguity at points of depth discontinuity. In the stereo vision, it is sometimes difficult to work especially when objects do not contain obvious features. To overcome this issue, structured light is combined with stereo vision. The correspondences between the stereo images can be identified more efficiently by observing how the projected pattern changes between the different viewpoints. Scharstein et al. [8] decoded light patterns to obtain unique codes at each pixel in each view to compute correspondences. Ishii et al. [9] proposed a method that measures the 3D shapes of moving objects using only a single projection pattern. Despite relatively low accuracy, this method is suitable for real-time applications. To improve 3D measurement accuracy, Jiang et al. [10] and Konolige [11] used a unique random speckle pattern. Jiang et al. also took advantage of temporal consistency to reduce the range of disparity updating to improve system efficiency. Base on the object's 3D information, the point cloud library (PCL) can be used to simulate and estimate the object's pose in 3D real world coordinates. Rusu and Cousins [12] showed that PCL is an advanced and extensive approach to improving 3D perception. PCL provides support for all the common 3D building blocks that applications need, including filtering, feature estimation, registration and segmentation. It has been widely used in 3D vision, CAD/CAM and machine vision, and has been deployed on Windows, Linux, and Android. In PCL applications, the key to pose estimation is to extract the features of the point cloud data through global or local methods. Alex and Adamson [13] showed how to use the local method to extract surface normal vector. Huang and You [14] proposed a way to compute normal vectors and curvatures to determine the similarity of point cloud data. Wahl et al. [15] proposed a four-dimensional feature called the point feature histogram (PFH) which calculates the angles and normal vectors of all neighboring points of the reference point to align the point cloud data [16-18]. To extend the PFH, Rusu et al. [19] proposed a method called the fast point feature histogram (FPFH) which reduced the angle calculation between neighboring points of the reference point, but at the cost of reduced feature precision. In the global method, Rusu et al. [20] proposed the viewpoint feature histogram (VFH), a descriptor of 3D point cloud data that encodes geometry and viewpoint. This method is derived from the extended FPFH. By computing the relative angles between each surface normal to the central viewpoint direction, it can be used to achieve object recognition and classification. Once the point cloud data features are extracted, one of the most popular registration methods is the iterative closest point (ICP) algorithm[21,22]. After comparing the features of point cloud data and determining the correspondences, ICP tries to determine the optimal transformation between all of correspondences by minimizing distance errors. Due to the ICP computation, Rusu et al. [19] proposed the sample consensus initial alignment (SAC-IA) method to give a first rough registration for ICP. The SAC-IA selects correspondences randomly and computes a rigid transformation to find the optimal result with a minimum distance error. In this paper, we propose a point cloud alignment method based on stereo vision. The active illumination solution, RPP, is used to enhance the accuracy of 3D point clouds. Method efficiency is improved using a down sample technique and the local features descriptor of the point clouds.

Chin-Sheng Chen received his PhD in mechanical engineering from National Chiao Tung University in 1999. Presently he is Professor and Director of the Graduate Institute of Automation Technology, National Taipei University of Technology. His research interests include machine vision systems and motion control.

Mei-Yu Huang received her MS in 2015 from the Graduate Institute of Automation Technology at National Taipei University of Technology. Her research mainly focuses on machine vision and 3D pose estimation.

Chun-Wei Yeh received his PhD in electronic, electrical and computer engineering at the University of Birmingham in 2015. Presently he is an Adjunct Assistant Professor at the Graduate Institute of Automation Technology, National Taipei University of Technology. He is also co-owner of MiM Tech. Inc., founded in 2014. His research interests include digital image processing and machine vision.

Chien-Liang Huang received his PhD from the Graduate Institute of Automation Technology at National Taipei University of Technology in 2015. His research interests mainly include machine vision and pattern recognition.

Architecture of the proposed method

Figure 1 shows an overview of the proposed method. The method can be divided into two phases: offline and online. Prior to applying the object pose estimation algorithm, the point clouds of the template and the target objects must be generated by the stereo vision module. The down sample technique and the feature extraction method are applied to describe the template and target objects. The pose of the target object is then estimated using the SAC-IA method [19]. Details of the proposed method are described in the following subsections.

Stereo vision module

Before estimating object's pose based on the point clouds, the point clouds must be known. The most important task in this module is the generation of an object's point clouds, which are then fed into the point cloud alignment method. In general, the surfaces of texture-less objects have no discriminative textures, which makes it difficult to generate accurate and reliable point clouds. Therefore, the point clouds are generated by observing the object with an active illumination solution, RPP, from two cameras set at different angles. RPP allows for the extraction of more reliable and discriminate features for texture-less objects. Another key task is camera calibration which uses the cameras' intrinsic and extrinsic parameters to generate accurate point clouds.

However, using RPP to project random laser points onto the object's surface also increases the noise in the captured images. To reduce the noise and enhance the features, we apply an image process called the histogram equalization method. The final step in the stereo vision module is the generation of point clouds based on the corresponding points in the stereo images.

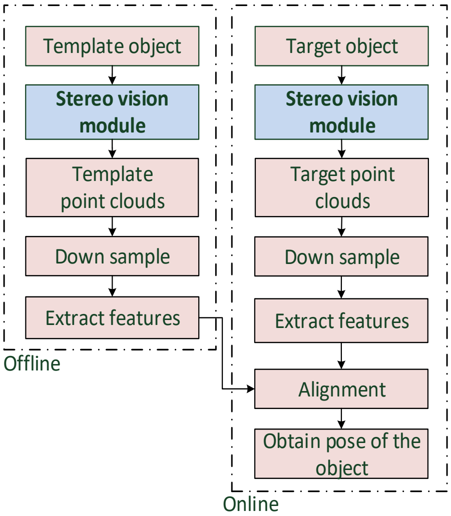

Figure 1. Architecture of the proposed method.

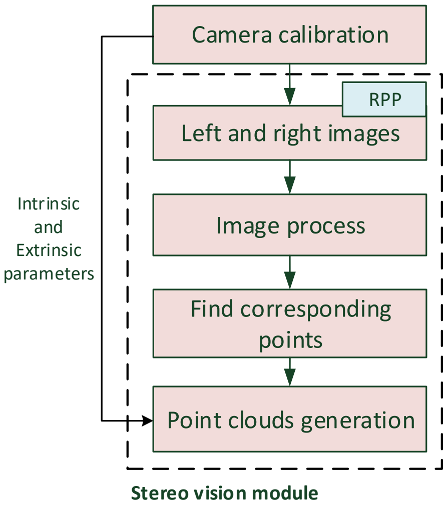

Figure 2. Stereo vision module.

To find the corresponding points between stereo images, a simple and efficient method called local stereo matching is applied. We assume that the left image is reference image l and right image is target image r. In local stereo matching, the correspondences can be computed by summing up the absolute differences (SAD) in a small region cropped around each point.

$$SAD(x,y,d)=\sum_{i=x-m}^{x+m}\sum_{j=y-n}^{y+n}\left|L(i,j)-R(i-d,j)\right| \tag{1}$$

where $L(i,j)$ and $R(i,j)$ are the respective intensity values of reference image l and target image r at position (i,j). m and n define the size of the matching window. d is the amount of window shift in r along a scan line. The correspondences can then be obtained by minimizing the error at each pixel along the scan line.

$$\underset{d\in[d_{\min},d_{\max}]}{\arg\min}\ SAD(x,y,d) \tag{2}$$

where $[d_{\min},d_{\max}]$ limits the search range. After finding the correspondences, the point clouds will be generated by triangulation.

$$Z=\frac{Bf}{x_l-x_r}=\frac{Bf}{d} \tag{3}$$

where B is the distance between two cameras. f is the focal length of the cameras. Z is the estimated depth value of the object. When the point clouds are found, the point clouds of an object can be generated by the commonly used triangular method.

Details of proposed method

According to Fig. 1, this method consists of offline and online parts. The main task of the offline phase is training the features extracted from the template object to estimate the target's pose in the latter phase. In the online phase, the features of the target object are also extracted to estimate its pose. The key problem with point clouds is they are not well suited for time-critical applications with huge data sets. For this reason, the down sample technique is designed to increase the computation advantage. There are three steps in the down sample technique:

1. Remove outliers: By calculating the number of neighbors in the point cloud data, the system removes the point whose number of neighbor is lower than predefined threshold.
2. Reduce the number of points: The huge point cloud data set requires a voxel grid to be created for each point. All points are then down sampled with their centroid in each voxel. The point cloud data still can maintain accurate surface measurements despite using significantly fewer points.
3. Plane model segmentation: In this paper, the object sits on a work table. The plane model of the work table does not include any features that need to be considered. Accordingly, this system segments and removes the plane model of the work table to improve system efficiency.

For the alignment method for estimating object pose, this paper extracts local features called FPFH, which is an extension of PFH. A PFH representation is based on the relationships between the points in the k-neighborhood and the surface normal vector. The relationship between two points D_s and D_t is shown in Figure 3. To compute the relative difference between two pints and normal N_s and N_t, the fixed coordinate frame UVW is defined as follows:

$$U = N_s, \tag{4}$$

$$V = \frac{(D_t - D_s)\times U}{\left\| D_t - D_s \right\|}, \tag{5}$$

$$W = U \times V, \tag{6}$$

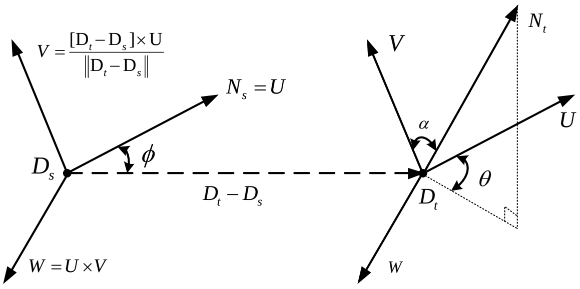

Figure 3. Relationship of the features of two points

The difference between the two normals N_s and N_t can be expressed as angular and distance features as follows using the UVW frame:

$$\alpha = \cos^{-1}(V \cdot N_t), \tag{7}$$

$$d = \left\| D_t - D_s \right\|, \tag{8}$$

$$\phi = \cos^{-1}(U \cdot \frac{D_t - D_s}{d}), \tag{9}$$

$$\theta = \tan^{-1}(W \cdot N_t, U \cdot N_t) \tag{10}$$

As shown in Fig. 4, the features of D_q can be expressed by calculating the four angular and distance features of all point pairs in the k-neighborhood. The four features are categorized using a histogram as follows:

$$\left.\begin{array}{l} f_1 = V \cdot N_t \\ f_2 = \left\| D_t - D_s \right\| \\ f_3 = U \cdot \dfrac{D_t - D_s}{d} \\ f_4 = \tan^{-1}(W \cdot N_t, U \cdot N_t) \end{array}\right\} idx = \sum_{i=1}^{4} step(s_i, f_i) \cdot 2^{i-1} \tag{11}$$

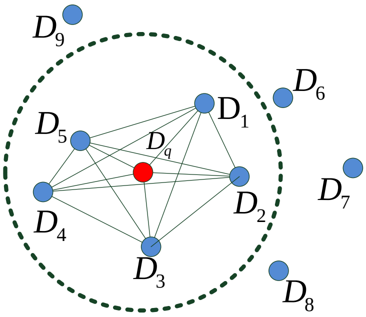

Figure 4. Neighborhood of PFH.

Here, f_2 means the distance between two points; f_1 and f_3 are dot products between normalized vectors. Their values are between ± 1 because of the cosine angles. The value of f_4 is between $\pm \pi/2$ because of the arctangent angle. The $step(s_i, f_i)$ is 0 if $f_i < s_i$ and 1 otherwise. This means that the algorithm classifies each feature in two parts by setting s_i to the center of the interval of f_i. By dividing the feature values into two parts, a 2^4 bins of combination between the four features can be obtained. Therefore, the value of idx is from 0 to 15. Furthermore, the FPFH is proposed to speed up the computation. As shown in Figure 5, the feature of D_q can be obtained by computing the four features between only itself and its neighbors to simplify the histogram feature computation. Then, the k-neighborhood is re-determined and the neighboring SPFH values also used to weight the final histogram of D_q :

$$FPFH(D_q) = SPFH(D_q) + \frac{1}{k}\sum_{i=1}^{k} \frac{1}{w_k} \cdot SPFH(D_k), \tag{12}$$

$$w_k = \sqrt{\exp\left\| D_q - D_k \right\|}, \tag{13}$$

where w_k represents the distance between D_q and its neighbor point D_k.

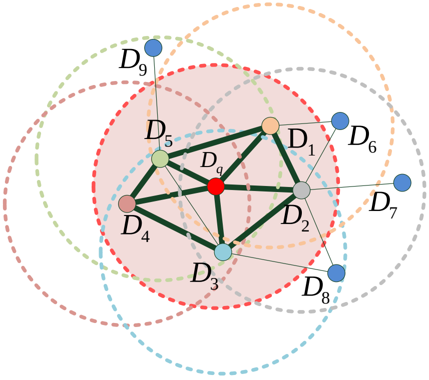

Figure 5. Neighborhood of FPFH.

Experimental results

A texture-less object, made of 30mm steel gage, is used to assess the rotation and the translation accuracy of the proposed approach (Fig. 6). All experiments were performed in Visual Studio 2010 on a personal computer (PC) with an Intel Core i5 2.8 GHz and 4GB memory. The hardware of the cameras and RPP are shown in Fig. 7. Figure 8 shows the RPP, modeled as Osela RPP016, and the projected pattern with a field of view (FOV) of $35° \times 35°$. The corresponding number of projected laser points is about 23,880. The cameras (Sony XCG-V60E) have a CCD resolution of 640×480 pixels. To estimate the rotation accuracy, the point cloud data is successively rotated from $0°$ to $90°$ in $5°$ step increments. To estimate translation accuracy, the point cloud data is successively translated from 0 to 90 mm in 5mm step increments. There are 6859 rotation and translation experimental data points. To evaluate overall accuracy, two performance indices, the average (Aver.) and the standard deviation (Std. dev.), are used to quantitatively show the performance of the proposed method. The rx, ry and rz respectively represent the error of rotation angle with respect to the *x*, *y* and *x* axes. The tx, ty and tz respectively represent the error of translation in the *x*, *y* and *x* directions. The experimental results are shown in Table 1.

Extracted feature	PFH			FPFH		
Aver. error of rotation(°)	rx	ry	rz	rx	ry	rz
	0.047	0.011	-0.042	0.031	0.016	-0.031
Aver. error of translation (mm)	tx	ty	tz	tx	ty	tz
	0.371	-0.109	0.865	-0.109	0.067	0.565
Std. dev. of rotation(°)	rx	ry	rz	rx	ry	rz
	0.309	0.029	0.312	0.193	0.030	0.199
Std. dev. of translation (mm)	tx	ty	tz	tx	ty	tz
	1.152	0.987	1.852	0.616	0.902	0.830
time (ms)	1064			230		

Table 1. Pose estimation errors.

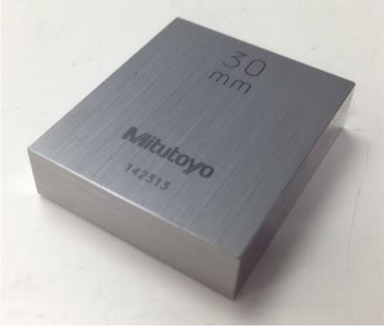

Figure 6. Texture-less gage.

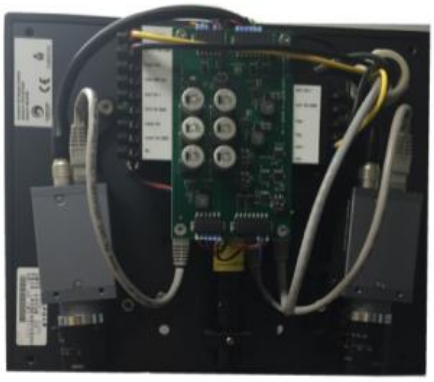

Figure 7. Camera and RPP hardware.

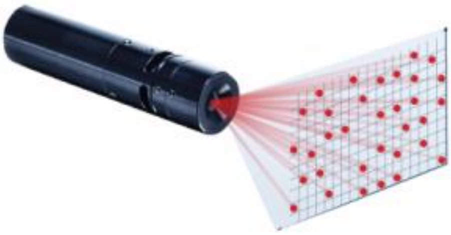

Figure 8. RPP and projected pattern.

As shown in Table 1, FPFH extraction results in a maximum error and standard deviation of rotation of about 0.031° (rx) and 0.199° (rz). The maximum error and standard deviation of translation are about 0.565 mm and 0.902 mm. The execution time of pose estimation is about 230 ms. The execution time of FPFH is about one-fourth that of PFH. FPFH also provides accurate and efficient pose estimation results by simplifying and extending the calculation of the neighboring points. These results indicate that the proposed method can be reliably used on real word 3D objects.

Conclusion

A method is proposed to apply an active illumination solution and RPP to increase reliable and discriminate features for texture-less objects. After obtaining the object's point cloud, the extracted FPFH features of the down sample point cloud data were applied to the 3D alignment algorithm. Experimental results found a maximum error and standard deviation of rotation of about 0.031° and 0.199°, while the maximum error and standard deviation of translation is about 0.565 mm and 0.902 mm, with a pose estimation execution time of about 230ms. Thus, the proposed method is an efficient and accurate point cloud alignment algorithm suitable for use in real-time applications.

References

[1] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence,* vol. 22, no. 11, pp. 1330-1334, 2000.
doi: 10.1109/34.888718

[2] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International journal of computer vision,* vol. 47, no. 1-3, pp. 7-42, 2002.
doi: 10.1023/A:1014573219977

[3] L. D. Stefano, M. Marchionni, and S. Mattoccia, "A fast area-based stereo matching algorithm," *Image and vision computing,* vol. 22, no. 12, pp. 983-1005, 2004.
doi: 10.1016/j.imavis.2004.03.009

[4] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," *Exploring artificial intelligence in the new millennium,* vol. 8, pp. 236-239, 2003.

[5] A. Klaus, M. Sormann, and K. Karner, "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure," in proceeding of *Pattern Recognition, 2006. ICPR 2006. 18th*

International Conference on, Hong Kong, 2006, pp. 15-18.
doi: 10.1109/ICPR.2006.1033

[6] O. Veksler, "Stereo correspondence with compact windows via minimum ratio cycle," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 24, no. 12, pp. 1654-1660, 2002.
doi: 10.1109/TPAMI.2002.1114859

[7] K.-J. Yoon and I.-S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.28, no. 4, pp. 650-656, 2006.
doi: 10.1109/TPAMI.2006.70

[8] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in proceeding of *Computer Vision and Pattern Recognition*, Madison, WI, 2003, pp. I-195-I-202.
doi: 10.1109/CVPR.2003.1211354

[9] I. Ishii, K. Yamamoto, and T. Tsuji, "High-speed 3D image acquisition using coded structured light projection," in proceeding of *Intelligent Robots and Systems*, San Diego, CA, 2007, pp. 925-930.
doi: 10.1109/IROS.2007.4399180

[10] J. Jiang, J. Cheng, and H. Zhao, "Stereo Matching Based on Random Speckle Projection for Dynamic 3D Sensing," in proceeding of *Machine Learning and Applications (ICMLA)* , Boca Raton, FL, 2012, pp. 191-196.
doi: 10.1109/ICMLA.2012.40

[11] K. Konolige, "Projected texture stereo," in proceeding of *Robotics and Automation (ICRA),* Menlo Park, CA, 2010, pp. 148-155.
doi: 10.1109/ROBOT.2010.5509796

[12] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in proceeding of *Robotics and Automation (ICRA),* Shanghai, China, May 9-13, 2011, pp. 1-4.
doi: 10.1109/ICRA.2011.5980567

[13] M. Alexa and A. Adamson, "On normals and projection operators for surfaces defined by point sets," in proceeding of *Eurographics conference on Point-Based Graphics*, Germany, 2004, pp. 149-155.
doi: 10.2312/SPBG/SPBG04/149-155

[14] J. Huang and S. You, "Point cloud matching based on 3D self-similarity," in proceeding of *Computer Vision and Pattern Recognition Workshops (CVPRW)*, Providence, RI, 2012, pp. 41-48.
doi: 10.1109/CVPRW.2012.6238913

[15] E. Wahl, U. Hillenbrand, and G. Hirzinger, "Surflet-pair-relation histograms: a statistical 3D-shape representation for rapid classification," in proceeding of *3-D Digital Imaging and Modeling,* Banff, Canada, 2003, pp. 474-481.
doi: 10.1109/IM.2003.1240284

[16] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in proceeding of *IEEE/RSJ 2008 Intelligent Robots and Systems*, Nice, France, 2008, pp. 3384-3391.
doi: 10.1109/IROS.2008.4650967

[17] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, "Learning informative point classes for the acquisition of object model maps," in proceeding of *2008 10th International Conference on Control, Automation, Robotics and Vision*, Hanoi, Vietnam, 2008, pp. 643-650.
doi: 10.1109/ICARCV.2008.4795593

[18] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, "Persistent point feature histograms for 3D point clouds," in proceeding of *10th Int Conf Intel Autonomous Syst (IAS-10),* Baden-Baden, Germany, 2008, pp. 119-128.
doi: 10.3233/978-1-58603-887-8-119

[19] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in proceeding of *2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009, pp. 3212-3217.
doi: 10.1109/ROBOT.2009.5152473

[20] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3d recognition and pose using the viewpoint feature histogram," in proceeding of *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, 2010, pp. 2155-2162.
doi: 10.1109/IROS.2010.5651280

[21] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in proceeding of *Robotics-DL tentative*, 1992, pp. 586-606.

[22] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in proceeding of *Third International Conference on 3-D Digital Imaging and Modeling*, Quebec City, Canada, 2001, pp. 145-152.