# Recognition of Objects and Places in 3D Point Clouds for Robotic Applications

Robert Cupec, Emmanuel Karlo Nyarko, Damir Filko, Luka Markasović
Faculty of Electrical Engineering Osijek
Josip Juraj Strossmayer University of Osijek
Osijek, Croatia
robert.cupec@etfos.hr

*Abstract*— **In this paper, object and place recognition are considered as the same problem with two applications in robotics: recognition of objects for robot manipulation and recognition of places for mobile robot localization. The approaches which use 3D point clouds obtained by 3D sensors as input are discussed. The paper focuses on feature-based recognition methods which are suitable for robotic applications since they provide a precise pose of either an object of interest or a mobile robot in its environment. Several state-of-the-art object and place recognition approaches are reviewed. An indoor place recognition system designed by the authors is presented as an example.**

***Keywords— computer vision; object recognition; place recognition; 3D sensors***

## I. Introduction

In great majority of the current practical robotic applications, robots perform a finite number of programmed motions along predefined trajectories in highly structured environments. At the same time, extensive research has been conducted in the field of robot perception, which is the key to more intelligent behavior of future robots and significant expansion of their field of application.

Recognition of objects of interest is one of the basic abilities an intelligent robot should have. Without this ability, the working environment of the robot must be organized in such a way that the objects the robot manipulates with are positioned accurately in predefined locations. This requirement is usually met by additional mechanical equipment and sensors, which significantly complicates the robot's application.

Environment perception is also a vital property of mobile robots. Mobile robots are of special interest in the robotics research community because their mobility allows them to expand their operating space to entire buildings or production floors. In order to exploit this ability to solve tasks which include moving from its current position to a given goal position, a mobile robot must be capable of localizing itself in a previously built map of its operating environment. Several variants of mobile robot localization problem are considered in the literature, from motion estimation[5] and local pose tracking[6][31] to global localization[7], loop closing[9] and kidnapped robot problem[8]. The last three problems are related to the place recognition problem, i. e. the problem of identifying the current robot location in a map without any prior knowledge about its previous location or motion. In this context, the term *place* is used to denote a particular location in the considered environment.

If a static part of an environment, e. g. floor, walls, door and ceiling of a room, is regarded as a rigid object, then the place recognition problem can be considered as a special case of the object recognition problem. Therefore, the same methodology can be used for both object and place recognition. In both cases, the applied recognition methods must be robust to changing viewpoint, varying lighting conditions as well as occlusion.

Most of object and place recognition approaches can be classified in two groups: appearance based methods[1][2] and the methods based on feature registration[3][4][10][11][15][23][24][25][28]. The methods based on feature registration are more interesting for robotic applications since they provide accurate object pose.

This paper considers object and place recognition approaches which use 3D point clouds obtained by a 3D sensor as input. In most of the research reviewed in this paper, RGB-D cameras like Microsoft Kinect or PrimeSense sensor are used. A RGB-D image consists of two images, a 'standard' RGB image and a depth image. The depth image assigns to each image pixel depth information from which 3D position of the point represented by this pixel can be computed. Hence, RGB-D image can be regarded as a colored 3D point cloud.

The paper is organized as follows. The general object recognition approach is defined in Section II and a commonly applied strategy for solving this problem is described. This strategy is based on registration of geometric features extracted form point clouds. Several approaches designed primarily for recognizing relatively small movable objects based on feature registration are described in Section III. Two methods for recognizing places in indoor environments which use the same strategy are described in Section IV. The paper is concluded in Section V.

## II. Recognition Based on Feature Registration

We begin this section by defining the problem of recognizing objects in 3D point clouds using a formulation similar to the one used in [10]. Let *S* be a set of 3D points i.e. a point cloud acquired by a 3D sensor, referred to in the following as a *scene*, and let $\mathcal{M} = \{M_1, M_2, ..., M_n\}$ be a set of models of objects to be recognized in that point cloud. Each model $M_i$ represents a point cloud obtained by scanning an object of interest by a 3D sensor from a particular viewpoint or by fusion of point clouds acquired from multiple views. Each model $M_i$ is assigned a reference frame in which the point coordinates are represented. Analogously, the points in the set $S$ are represented in the sensor reference frame. The considered problem is to determine a correct *interpretation* of the scene $S$, i. e. the set of hypotheses $\mathcal{I} = \{H_1, H_2, \ldots, H_r\}$, where each $H_i$ is a hypothesis that an object from the set $\mathcal{M}$ appears in the scene $S$ in a particular pose. A hypothesis can be represented by a pair $H_i = (o_i, \mathbf{T}_i)$, where $o_i$ is the object index and $\mathbf{T}_i$ is a homogenous transformation matrix describing the pose of the object relative to the sensor reference frame.

In this paper, approaches for object and place recognition in colored 3D point clouds, which are based on registration of features of various types, are considered. The basic pipeline of these approaches which is executed in the online recognition phase consists of the following steps:

- feature detection;
- generating hypotheses from feature matches;
- hypothesis evaluation.

Features which are used in the considered recognition approaches represent geometric elements such as a 3D point, a pair of oriented 3D points, a line or a planar surface segment. Each feature is assigned a *local descriptor* representing a vector of values which describe the local neighborhood of the feature.

In the *hypothesis generation* stage, the features detected in the scene $S$ are matched to the features of the same type extracted from all models $M_i \in \mathcal{M}$. Feature matching is performed according to the local descriptors assigned to the features. If the descriptor of a scene feature is sufficiently similar to a descriptor of a model feature, according to a certain similarity measure, the parameters of these two features defining their pose relative to the respective reference frames are used to compute the pose of a model reference frame relative to the sensor reference frame. If a single feature match does not contain information sufficient for estimating full 6DoF object pose, groups of features are matched, where geometric relations between the features in a group are used in the matching process together with the local descriptors. The object pose computed from a feature match or by matching two groups of features represents a hypothesis that this particular object is present in the scene in the computed pose.

Since many features are usually detected in point clouds, a large number of hypotheses are generated and only some of them are correct. Therefore, a suitable criterion must be used to decide which of the generated hypotheses can be accepted as correct and which should be rejected. This final step is referred to herein as *hypothesis evaluation*.

## III. Object Recognition

In this section several state-of-the-art object recognition approaches based on the general strategy described in Section II are reviewed. The approaches differ in selection of features, method applied to generate hypotheses and the criterion used for hypothesis evaluation.

### A. *Hypothesis Generation*

The method proposed in [11] can be used to recognize objects in 3D point clouds in cluttered scenes. The method is tested by recognizing a set of objects in 3D point clouds obtained by a laser range finder from a single viewpoint. The objects to be recognized are modeled by high resolution 3D point clouds covering the entire object surface. The features used in this approach are pairs of oriented points. This type of feature is originally proposed in [12]. An oriented point is obtained by assigning to a 3D point the unit vector perpendicular to the local surface in the close neighborhood of that point. Each oriented 3D point is defined by five parameters: three coordinates of the point and two parameters defining the orientation of the assigned unit vector. A pair of oriented 3D points can, therefore, be described by 10 parameters. Six of these 10 parameters define the 6DoF pose of this feature relative to the point cloud reference frame, while the remaining four parameters are used to form a local descriptor of the feature. In order to reduce the number of hypotheses, only the pairs of oriented points with a user defined distance are used in [11]. Since the point distance is constant, it is not used in the local descriptor as proposed in [12] and therefore the local descriptor is a three element vector. Hypotheses are generated using a RANSAC[13]-based approach. In order to achieve fast feature matching, a hash table is created from all models in $\mathcal{M}$. The entries in the hash table are created from the features extracted from the models. Each entry represents the information about the model and the pose of the feature relative to the model reference frame. The address of an entry is computed from the feature's local descriptor. In the online recognition phase, for each feature extracted from $S$, the feature with a similar local descriptor is fetched from the hash table. In order to compute surface normals needed to obtain oriented points efficiently, a fast identification of the neighboring points is critical. This is achieved by representing $S$ by an octree[14].

In the approach presented in [15], 3D points at uniformly sampled positions on the surfaces of models and the scene are used as features and each of these points is assigned a SHOT local descriptor proposed in [16]. Fast descriptor matching is accomplished by indexing implemented using FLANN [17]. Hypotheses are generated by a *correspondence grouping* algorithm. This algorithm starts from a seed correspondence, i. e. a pair of features with sufficiently similar local descriptors, and for each such correspondence it forms a group of feature correspondences which satisfy a particular geometric constraint, described in the following. Let $(\mathbf{p}_i, \mathbf{p}'_j)$ be the seed correspondence, where $\mathbf{p}_i$ is a scene feature point and $\mathbf{p}'_j$ is the corresponding model feature point. Another correspondence $(\mathbf{p}_k, \mathbf{p}'_l)$ is added to the group if

$$\left| \|\mathbf{p}_i - \mathbf{p}_k\| - \|\mathbf{p}'_j - \mathbf{p}'_l\| \right| < \varepsilon ,$$

where $\varepsilon$ is a user defined tolerance. For each correspondence group, a RANSAC-based algorithm is applied to all the correspondences in the group resulting in a 6DoF pose hypothesis. The pose obtained by the RANSAC algorithm are then refined using ICP procedure [18]. A sample result of the described method is shown in Fig. 1, where the hypothesis with the greatest number of feature correspondences is displayed. However, this simple hypothesis evaluation criterion is not suitable for cluttered scenes. A more advanced hypothesis evaluation method is proposed in [15]. This method is reviewed in Section III.B.

The method presented in [15] is augmented in [10] by two additional hypothesis generation pipelines, one which uses point features detected in the grayscale image with assigned SIFT[19] descriptors and the other which uses a semi-global 3D descriptor representing an extension of the OUR-CVFH approach [20] based on the color, shape and object size cues. Point features extracted from the grayscale image are projected onto the depth image in order to obtain 3D points coordinates.

The method presented in [10] is adapted in [21] for object recognition from multiple views.

The approach proposed in [22] assumes that the objects of interest lie on a flat surface which is detected using RANSAC and removed from *S*. The remaining points in *S* are then clustered and each cluster is compared to the models from $\mathcal{M}$ using global hue descriptors. For each matching pair of objects pose hypotheses are generated using SIFT features and RANSAC.

The approach proposed in [24] uses Color Point Pair Features, obtained by augmenting the descriptor proposed in [12] with color information. Hypotheses are generated by evidence accumulation followed by pose clustering. The hypotheses are sorted according to the number of accumulated votes and a fixed number of the highest ranked hypotheses are returned as the final result.

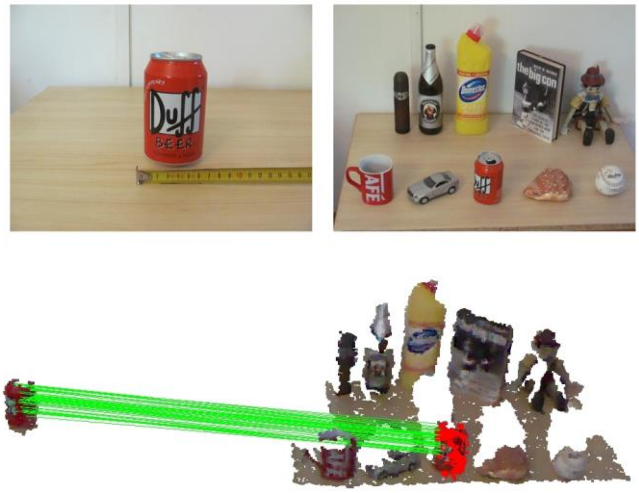

Figure 1. Object recognition in a colored 3D point cloud. Top row shows the RGB image of an object model (left) and a scene (right). Bottom row shows the hypothesis with the greatest number of feature matches, where green lines connect model features with the corresponding scene features.

### B. Hypothesis Evaluation

In general, the result of the hypothesis generation stage is a hypotheses set $\mathcal{H} = \{H_1, H_2, \dots, H_r\}$. Some of those hypotheses are correct and some are not. Selection of correct hypotheses according to a particular criterion is performed in the hypothesis evaluation stage.

The approach presented in [22] detects objects as clusters of 3D points remaining after the supporting plane is removed from *S*. This approach generates hypotheses separately for each cluster and selects a single most probable hypothesis for each cluster. The hypotheses are evaluated by projecting the SIFT features detected in the scene onto the corresponding model and searching for the corresponding feature in local neighborhood. The hypothesis are ranked according to the number of matched SIFT features. The method presented in [22] is improved in [23] by including additional cues in the hypothesis evaluation stage: color, shape context features and SIFT features. The scores of the said cues are blended feature-weighted linear stacking approach. Furthermore, standard ranking support vector machine is applied to determine which object corresponds to each cluster from *S*.

In the approach proposed in [11] hypothesis evaluation is performed in two stages. In the first stage, the model is transformed in the sensor reference frame by the transformation $\mathbf{T}_i$ of the evaluated hypothesis $H_i$. The hypothesis is rejected if the percentage of model points which are sufficiently close to scene points is below a user defined threshold or the percentage of model points which occlude scene points exceeds a user defined threshold. In the second stage, for each hypothesis, the number of points from *S* which are sufficiently close to the transformed model points is determined, referred to in the following as the hypothesis *support*. Each hypothesis which conflicts a hypothesis with a larger support is rejected. Two hypotheses are conflicting if the intersection of their supports is not an empty set.

In the approach presented in [15], each hypothesis $H_i$ is assigned a boolean variable $x_i \in \mathbb{B} = \{0,1\}$ which has a value of 1 if the hypothesis is correct and 0 if the hypothesis is false. Therefore, a possible solution can be represented by a sequence $\mathcal{X} = \{x_1, x_2, \dots, x_r\}$. Hypotheses $H_i$ for which $x_i = 1$ are referred to in the following as *active* hypotheses. In the hypothesis evaluation stage, the solution space $\mathbb{B}^r$ is searched for a solution which minimizes a cost function using a simulated annealing approach. The cost function is based on several cues. The first cue is the number of scene points which are explained by any hypothesis, where the term *explained* has the following meaning. A scene point is explained by a hypothesis $H_i$ if there is a point belonging to the model $M_{o_i}$ which is, after being transformed by $\mathbf{T}_i$, sufficiently close to the considered scene point. In order for a scene point to be considered as explained, in [15] is also required that the orientation of the scene point is sufficiently similar to the orientation of the corresponding model point. Each scene point explained by an active hypothesis contributes to the cost function by a value computed from the distance to the corresponding model point and the similarity between the orientations of these two points. This value is negative for close points with similar orientation, which means that each explained point decreases the cost function. The second cue is the number of visible model points of every active hypothesis which do not explain any scene point. This number increases the cost function. The third cue is the number of conflicting hypotheses for each scene point, i. e. the number of active hypotheses $H_i$ which explain the same scene point. This number also increases the cost function. The fourth cue is the number of unexplained scene points that are likely to belong to the same surface as nearby explained points. In order to compute this number, smooth clusters of points in $S$ are identified. Each cluster is assumed to belong to the same object. Hence, all unexplained scene points belonging to a cluster which also contains explained scene points increase the cost function.

In [10] the hypothesis evaluation method proposed in [15] is extended with two additional cues. The first is a color cue, which measures the similarity of the color of scene point and the corresponding model points. The second cue penalizes hypotheses according to which objects are partially positioned below the table.

## IV. Place Recognition

Most place recognition methods are based on bag-of-words (BoW) technique[32]. In addition to recognizing places in standard camera images the BoW approach is also used for place recognition based on point clouds[33]. Nevertheless, this approach determines which model image is most similar to the currently acquired image, but it does not provide accurate information about the current camera pose in the map.

In this section, place recognition approaches based on feature registration technique described in Section II are considered. Point features with local descriptors encoding information about the local shape of the object surface in the close vicinity of the corresponding point features, such as SHOT, are suitable for recognition of small objects. However, stable landmarks which can be used for place recognition in indoor environments are mostly planar surfaces such as floor, ceiling, walls, or large furniture whose position in the environment is fixed. Therefore, a reasonable choice of features to be used for indoor place recognition is planar surface segments. If the place recognition problem is regarded in analogy to the general object recognition problem discussed in Section II, then the environment map can be represented by a set $\mathcal{M} = \{M_1, M_2, ..., M_n\}$, where $M_i$ are local models representing particular locations in the map, which are referred to in this paper as *places*.

A place recognition approach which uses planar surface segments as features is proposed in [25]. The environment map used in [25] represents a set of planar surface segments described by the parameters defining their position with respect to a common reference frame. The planar surface segments are extracted from a depth image acquired by a RGB-D camera using a region growing technique proposed in [26]. The sensor data, according to which the current camera location in the map is identified, is not a single RGB-D image, but a sequence of images acquired while the camera is moving along a path in the considered environment. This can be regarded as fusion of a sequence of point clouds into a point cloud which corresponds to the notion of scene $S$ as it was defined in Section II. Since matching of a planar surface segment extracted from a scene to a planar surface segment in a map does not provide sufficient information for determining the 6DoF camera pose relative to the map, groups of planar surface segments must be matched in order to compute the camera pose. Both the map and the scene are represented by a graph, whose nodes are surface segments, which are connected by neighborhood relations. In the hypothesis generation stage subgraphs are formed consisting of one surface segment, representing the reference node of a subgraph, and all segments connected to this node. Hypotheses are generated by matching subgraphs of the scene with subgraphs of the map using the interpretation tree approach [27]. The result of matching two subgraphs is a set surface segment correspondences. From these correspondences 6DoF camera pose is computed by minimizing a cost function which measures the adjustment error of each matched plane pair. This pose computation is formulated as a least squares problem which is solved using Gauss-Newton optimization. A subgraph match is accepted if the parameters of the scene surface segments transformed by the estimated camera pose to the model reference frame are sufficiently similar to the parameters of the corresponding model surface segments.

Another place recognition approach based on planar surface segments is proposed in [28]. In addition to planar surface segments, straight object edges are also used as features. Planar surface segments are extracted form an organized point cloud, i. e. a depth image, using the method based on recursive Delaunay triangulation [29] to create a triangular mesh from the point cloud and the hierarchical surface merging approach proposed in [30] to merge triangles into planar segments.

The hypothesis generation used in [28] is also based on building an interpretation tree. However, the interpretation tree is built differently. First, the scene and model surface segments are sorted according to *information content factor* (ICF) [31], representing a measure of useful information for camera pose estimation provided by a surface segment. A queue of surface segment pairs is created, where the first element of the pair is a scene surface segment and the second element is a model surface segment. This queue is sorted according to the ICFs of the surface segments forming the pairs. Finally, the interpretation tree is constructed recursively by appending new nodes corresponding to surface segment pairs. The pairs are taken from the queue in the sorting order starting with the pair of surface segments having the highest ICFs. This strategy is aimed at generating a correct hypothesis in the early stage of interpretation tree construction, thereby reducing the computation time. Each path connecting a leaf of the interpretation tree to the root of the tree represents a 5DoF pose hypothesis, defining the orientation and two translational DoFs of the camera pose. This 5DoF pose is computed from the sequence of surface segment correspondences along the path from the leaf to the root of the interpretation tree using EKF. The remaining translational DoF is estimated by an evidence accumulation approach described in [28].

While the method described in [25] performs fusion of point clouds acquired during camera motion, the method presented in [28] uses a single depth image as a scene *S*. Each local model $M_i \in \mathcal{M}$ represents likewise a single depth image.

In [28], a probabilistic approach is used for hypothesis evaluation. The proposed method is based on the assumption that the prior probability of accidental appearance of planar surfaces in any particular geometric arrangement is rather low. Therefore, if a set of planar surfaces detected in the scene are in the same geometric arrangement as a set of planar surfaces in the environment model, the probability of the camera being in a particular pose which aligns the scene feature set to the model feature set is high. Moreover, the more scene surfaces are aligned with model surfaces the higher this probability is. Assuming that exactly one of the hypotheses from a considered hypothesis set $\mathcal{H}$ is correct and that the prior probability of all hypotheses is equal, the probability of a hypothesis $H_k$ can be considered proportional to the likelihood $p(Z \mid H_k)$, where $Z = \{F_1, F_2, \ldots\}$ denotes the set of planar surface segments $F_i$ extracted from *S*. Assuming that each surface segment represents an independent measurement, this likelihood can be computed as

$$p(Z \mid H_k) = \prod_{F_i \in Z} p(F_i \mid H_k), \tag{1}$$

where $p(F_i \mid H_k)$ is probability density function (PDF) of detecting a surface segment with particular parameters if the hypothesis $H_k$ is correct. Computation of $p(F_i \mid H_k)$ is performed by transforming all surface segments of the local model $M_{o_i}$ using the homogenous transformation matrix $\mathbf{T}_i$, and matching the scene surface segments extracted from *S* to the model surface segments extracted from $M_{o_i}$. If a scene surface segment $F_i$ is matched to a model surface segment $F'_j$, then $p(F_i \mid H_k)$ is computed using the uncertainty model of the orientation of the segments $F_i$ and $F'_j$, as explained in [28]. The smaller the difference between the parameters of the matched surface segments the higher the value $p(F_i \mid H_k)$. Furthermore, since the uncertainty of the orientation is lower for larger surface segments, this computation gives higher values of $p(F_i \mid H_k)$ for larger surface segments $F_i$ and $F'_j$. If a scene surface segment $F_i$ is not matched to any model surface segment, then a prior PDF corresponding to an accidental occurrence of a surface segment in the scene with particular parameters is used for $p(F_i \mid H_k)$ in (1). A scene surface segment is not considered in the matching process if it occludes a model surface segment, because it is assumed that a transparent surface cannot be detected by the 3D camera. Analogously, a model surface segment is rejected from the matching process if it occludes a scene surface segment. This approach yields a similar effect as the penalization of the model points which occlude scene points in [11] and the fourth cue in the hypothesis evaluation approach in [15].

An example of place recognition achieved by the approach proposed in [28] is shown in Fig. 2. The red cones represent the camera poses from which the model point clouds are acquired, while the blue cones represent the camera poses corresponding to correct hypotheses. The results shown in Fig. 2 are obtained by an improved version of the method described in [28]. The main improvement is that instead of using a single depth image as a scene *S* and to create each local model $M_i \in \mathcal{M}$, each scene and model point cloud is obtained by fusion of a sequence of depth images acquired form a particular location for varying pan and tilt angles of the camera.

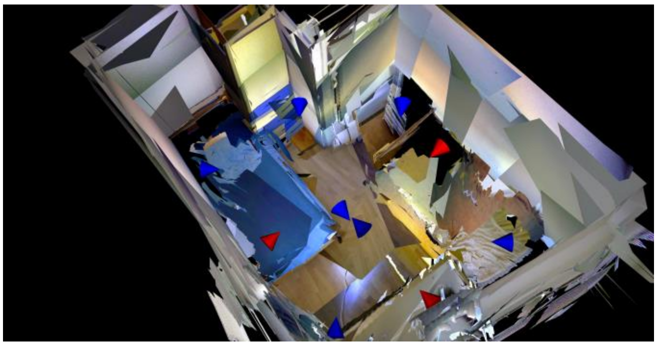

Figure 2. Place recognition based on registration of planar surface segments. For visualization purposes, textures are mapped onto the surface segments. The red cones represent the camera poses from which the model point clouds are acquired, while the blue cones represent the camera poses corresponding to correct hypotheses

## V. Conclusion

In this paper, a review of object recognition method based on registration of geometric features extracted from 3D point clouds is presented. The basic pipeline common to the considered methods is explained and different implementations of the three steps of this pipeline: feature detection, hypothesis generation and hypothesis evaluation, are discussed. Place recognition problem is formulated as a special case of the general object recognition problem. Two methods which use planar surface segments as features are briefly described. A sample result obtained by of one of these two methods is provided as an illustration of the discussed methodology.

## References

[1] M. J. Swain and D. H. Ballard, Color Indexing, International Journal of Computer Vision, 7,1,pp. 11–32, 1991.

[2] I. Ulrich and I. Nourbakhsh, "Appearance-Based Place Recognition for Topological Localization," In Proc. IEEE Int. Conf. on Robotics and Automation, April 2000, pp. 1023–1029.

[3] D. Lowe, "Object Recognition from Local Scale-Invariant Features", in Proc. Int. Conf. on Computer Vision (ICCV), 1999, pp. 1150–1157.

[4] S. Se, D.G. Lowe, and J.J. Little, “Vision-based global localization and mapping for mobile robots,” IEEE Transactions on Robotics, vol. 21, Jun. 2005, pp. 364–375.

[5] A.S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Fox, and N. Roy, “Visual Odometry and Mapping for Autonomous Flight Using an RGB-D Camera,” *15th International Symposium on Robotics Research (ISRR)*, 2011, pp. 1–16.

[6] M.F. Fallon, H. Johannsson, and J.J. Leonard, “Efficient scene simulation for robust monte carlo localization using an RGB-D camera,” *In Proc. IEEE International Conference on Robotics and Automation*, May. 2012, pp. 1663–1670.

[7] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, “Robust Monte Carlo localization for mobile robots,” Artificial Intelligence, vol. 128, May. 2001, pp. 99–141.

[8] S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics, The MIT Press, 2005.

[9] K. Leong Ho and P. Newman, “Detecting Loop Closure with Scene Sequences”, International Journal of Computer Vision, September 2007, Volume 74, Issue 3, pp 261-286.

[10] A. Aldoma, F. Tombari, J. Prankl, A. Richtsfeld, L. Di Stefano, and M. Vincze, “Multimodal cue integration through Hypotheses Verification for RGB-D object recognition and 6DoF pose estimation,” in ICRA, 2013, pp. 2104–2111.

[11] C. Papazov and D.Burschka, An Efficient RANSAC for 3D Object Recognition in Noisy and Occluded Scenes. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, Springer, Heidelberg, pp. 135–148, 2011.

[12] S. Winkelbach, S. Molkenstruck and F.M. Wahl, Low-Cost Laser Range Scanner and Fast Surface Registration Approach. In: Pattern Recognition, 28th DAGM Symposium, Proceedings. pp. 718–728, 2006.

[13] M. A. Fischler and R. C. Bolles, Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography, Graphics and Image Processing, vol. 24, no. 6, pp. 381–395, 1981.

[14] M. de Berg, O. Cheong, M. van Kreveld and M. Overmars, "Computational Geometry: Algorithms and Applications", Springer-Verlag, Berlin, Heidelberg, 2010.

[15] A. Aldoma, F. Tombari, L. Di Stefano, and M. Vincze, “A global hypothesis verification method for 3d object recognition,” in European Conference on Computer Vision (ECCV), 2012.

[16] F. Tombari, S. Salti, and L. Di Stefano, “Unique signatures of Histograms for local surface description,” in Proc. 11th European Conference on Computer Vision (ECCV), 2010.

[17] M. Muja and D. G. Lowe, "Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration", in International Conference on Computer Vision Theory and Applications (VISAPP), 2009.

[18] P. Besl, N. McKay, A Method for Registration of 3-D Shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, 1992.

[19] D. G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, The International Journal of Computer Vision, vol. 60, no. 2, pp. 91 – 110, 2004.

[20] A. Aldoma, F. Tombari, R. Rusu, and M. Vincze, "OUR-CVFH – Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation, " in Joint DAGM-OAGM Pattern Recognition Symposium, 2012.

[21] A. Aldoma, T. Fäulhammer and Markus Vincze, "Automation of “Ground Truth” Annotation for Multi-View RGB-D Object Instance Recognition Datasets," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2014, pp. 5016–5023.

[22] J. Tang, S. Miller, A. Singh, and P. Abbeel, "A textured object recognition pipeline for color and depth image data," in 2012 IEEE International Conference on Robotics and Automation (ICRA), 2012, pp. 3467-3474.

[23] Z. Xie, A. Singh, J. Uang, K. S. Narayan, and P. Abbeel, "Multimodal blending for high-accuracy instance recognition," in 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2013, pp. 2214–2221.

[24] C. Choi, H. I. Christensen, "3D Pose Estimation of Daily Objects Using an RGB-D Camera," in 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2013, pp. 3342–3349.

[25] E. Fernandez-Moral, W. Mayol-Cuevas, V. Arevalo and J. Gonzalez-Jimenez, "Fast place recognition with plane-based maps," in *Proc. IEEE Int. Conf. on Robotics and Automation*, May 2013, pp. 2719 – 2724.

[26] D. Holz and S. Behnke, "Fast Range Image Segmentation and Smoothing using Approximate Surface Reconstruction and Region Growing," in Proceedings of the International *Conference on Intelligent Autonomous Systems (IAS),* Jeju Island, Korea, 2012.

[27] W. E. L. Grimson, *Object Recognition by Computer - The role of Geometric Constraints*, MIT Press, Cambridge, MA. 1990.

[28] R. Cupec, E. K. Nyarko, D. Filko, A. Kitanov and I. Petrović, "Place Recognition Based on Matching of Planar Surfaces and Line Segments," *Int. J. Robotics Research*, to be published.

[29] F. Schmitt and X. Chen, "Fast segmentation of range images into planar regions," In *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 1991, pp. 710–711.

[30] M. Garland, A. Willmott and P. S. Heckbert, "Hierarchical Face Clustering on Polygonal Surfaces," In *Proc. ACM Symposium on Interactive 3D Graphics*, 2001.

[31] R. Cupec, E. K. Nyarko, D. Filko and I. Petrović, "Fast Pose Tracking Based on Ranked 3D Planar Patch Correspondences." In *Proc. IFAC Symposium on Robot Control*, Dubrovnik, Croatia, 2012.

[32] M. Cummins and P. Newman, "Appearance-only SLAM at large scale with FAB-MAP 2.0," International Journal of Robotics Research, August 2011, vol. 30 no. 9, pp.1100-1123.

[33] J. Behley, V. Steinhage and A.B. Cremers, "Laser-based segment classification using a mixture of bag-of-words," In Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Nov. 2013, pp.4195 -4200.