Project IP-2014-09-3155

Advanced 3D Perception for Mobile Robot Manipulators

FEATURES FOR OBJECT RECOGNITION IN 3D POINT CLOUDS BASED ON PLANAR PATCHES

Technical Report

ARP3D.TR5

version 1.0

Robert Cupec, Ivan Vidović

Josip Juraj Strossmayer University of Osijek

Faculty of Electrical Engineering Osijek

Osijek, 2016.

Ovaj rad je financirala/sufinancirala Hrvatska zaklada za znanost projektom IP-2014-09-3155.

Mišljenja, nalazi i zaključci ili preporuke navedene u ovom materijalu odnose se na autora i ne odražavaju nužno stajališta Hrvatske zaklade za znanost.

This work has been fully supported by/supported in part by the Croatian Science Foundation under the project number IP-2014-09-3155.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of Croatian Science Foundation.

1. Motivation

Vision-based object recognition problem can be formulated in several ways. Let's start with the simplest definition. Given a camera image or a 3D point cloud acquired by a 3D sensor and a suitable object model stored in the computer memory, the vision system must decide whether an object of interest is present on the image/point cloud or not. The discussion presented in this paper is limited to object recognition in 3D point clouds, with the focus on point clouds acquired by RGB-D cameras. These cameras provide depth information for each image pixel. Hence, an image obtained by a RGB-D camera can be easily transformed into a colored 3D point cloud.

A more demanding object recognition task is to detect a bounding box in the image/point cloud which contains the object of interest. If multiple instances of the object of interest are expected to appear in the image/point cloud, then the output of the vision system are multiple bounding boxes. Another version of the object recognition problem is the task of assigning every point in the point cloud to one instance of the object of interest or to the background.

If recognition of objects of fixed shape with application in robot manipulation is considered, then the vision system must provide accurate information about the object pose, which is then used for planning the robot manipulator trajectory.

Object recognition process is basically a process of comparison of features extracted from an observed scene with the features of the object in a model database. Which form of object model will be used in a particular object recognition task depends on the specific requirements of that task.

3D object recognition methods can be classified in **two broad classes**: (i) local approaches and (ii) global approaches. A survey of methods for object recognition in 3D point clouds is given in (Cupec, Grbić and Nyarko, 2016).

Local approaches are mostly based on registration of features of various types. The basic pipeline of these approaches which is executed in the online recognition phase consists of the following steps:

1. feature detection;
2. generating hypotheses from feature matches;
3. 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. 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*.

Global approaches compute one single descriptor for each object encompassing the whole object surface (Aldoma et al., 2012a). The global approach is characterized by a smaller complexity in the description and matching stage with respect to local methods, since each surface is characterized by one single (or a few for multivariate semi-global features) descriptor. However, global approaches are less effective in presence of partial object occlusions and require points in the observed scene to be segmented into different clusters, so that descriptors can be computed on each object cluster separately (Narayanany and Likhachev, 2016).

Features. In this paper, local object recognition approaches are considered, which rely on features extracted form a scene which are compared to the model features of the same type.

Desirable properties of features used for object recognition are

1. A feature must provide a unique reference frame whose pose relative to the object reference frame is stable with respect to noise and viewing angle.
2. The local object shape in the neighborhood of the feature should be distinguishable from the local neighborhoods of other features of the same object and features of other objects which are expected to appear on the scene.

The most common features used for object recognition are point features. A point feature consists of a *keypoint* or *interest point,* which defines the location of the feature in a point cloud and *local descriptor* which describes the local neighborhood of the keypoint. Some of the keypoint detectors used for object recognition in 3D point clouds are:

1. Normal Aligned Radial Feature (NARF) (Steder et al., 2010) and
2. Intrinsic Shape Signatures (ISS) (Zhong Y, 2009).

The task of the local descriptor is to detect points on the object surfaces in the scene with unique geometric properties in their local neighborhood. Furthermore, a desirable property of a feature is that the geometry of its local neighborhood is such that it uniquely defines three orthogonal directions, which can be used as a stable *feature reference frame* (Tombari, 2013).

Feature reference frame. This reference frame is used in the hypothesis generation stage for estimation of the object pose in the scene relative to the scene reference frame. A common approach for identification of the feature reference frame is to use the point and surface normal distribution in the local neighborhood of the feature's keypoint to define the orientations of the reference frame axes (Zhong Y, 2009). There are two main problems with this approach:

1. The shape of the object of interest can be such that only a few keypoints with a unique reference frame can be detected on its surface. If such object is positioned in a cluttered scene, these few keypoints could be occluded.

2. The uncertainty of the reference frame determined using a small local neighborhood could be rather high. This uncertainty is increased by the sensor noise.

In order to cope with the said problems, we have developed a novel feature, which is based on segmentation of the object's surface into planar surface segments.

2. Segmentation to Planar Surface Segments

Many segmentation algorithm perform oversegmentation to superpixels or supervoxels as a preprocessing step (Gupta, Arbeláez and Malik, 2013). The introduction of a low-level preprocessing step to oversegment images into superpixels – relatively small regions whose boundaries agree with those of the semantic entities in the scene – has enabled advances in segmentation by reducing the number of elements to be labeled from hundreds of thousands, or millions, to a just few hundred (Simari, Picciau and De Floriani, 2014). Probably the most popular method for oversegmentation of 2D images into superpixels is SLIC (Achanta et al., 2012). The method is based on k-means clustering. The same idea is adapted to 3D point clouds by (Papon et al., 2013). They designed a method which represents a 3D point cloud by rectangular voxel grid and then groups neighboring voxels into supervoxels.

A drawback of the method proposed in (Papon et al., 2013) is that the size of the obtained supervoxels is defined by a user specified parameter, which has to be selected according to previously known properties of the scene or objects of interest.

We developed a method which segments a 3D point cloud into approximately planar surface segments, where a user can specify the maximum deviation of points grouped in a segment form a plane assigned to this segment. This segmentation has the same underlying principle as the segmentation method proposed in (Holz and Behnke, 2014). It is based on region growing and it requires a 3D triangular mesh as input. A seed point is randomly selected from the input mesh. The region growing procedure is initialized by creating a region consisting of the selected seed point and growing this region by adding neighboring points which satisfy the criterion that the point must lie on the plane defined by the seed point and its normal within a predefined threshold. After the region growing procedure is completed, i.e. if no more points which satisfied the planarity criterion can be added to the grown region, a new randomly selected mesh point, which currently isn't assigned to any planar segment, is used as the seed for the next region growing process. The advantage of this approach in comparison to the method proposed in (Papon et al., 2013) is that the size of the obtained segments is determined by the scene geometry. Large planar surfaces in a given scene are represented by few large segments, while highly curved surfaces are represented by multiple small segments. Another advantage in comparison to (Papon et al., 2013) is that the points of a surfel lie on a planar surface within a user specified tolerance.

In order to create a triangular mesh from an input point cloud, our approach uses the fast mesh construction algorithm presented in (Holz and Behnke, 2012), which is included in the Point Cloud Library (Rusu and Cousins, 2011). The main difference between our approach and the method

presented in (Holz and Behnke, 2014) is that the point cloud segmentation obtained by our approach can be easily transformed into a polygonal representation of the given point cloud with relatively small number of vertices in comparison to triangular mesh of the same precision. However, generating of a polygonal representation from the segmentation obtained by our method is not implemented yet. The details of our algorithm are not presented in this paper. The complete description of the proposed approach will be published in a journal or conference paper.

Several examples of segmentations obtained by applying our method to noisy mesh models from TUW database (Aldoma et al., 2012b) are shown in Fig. 1.

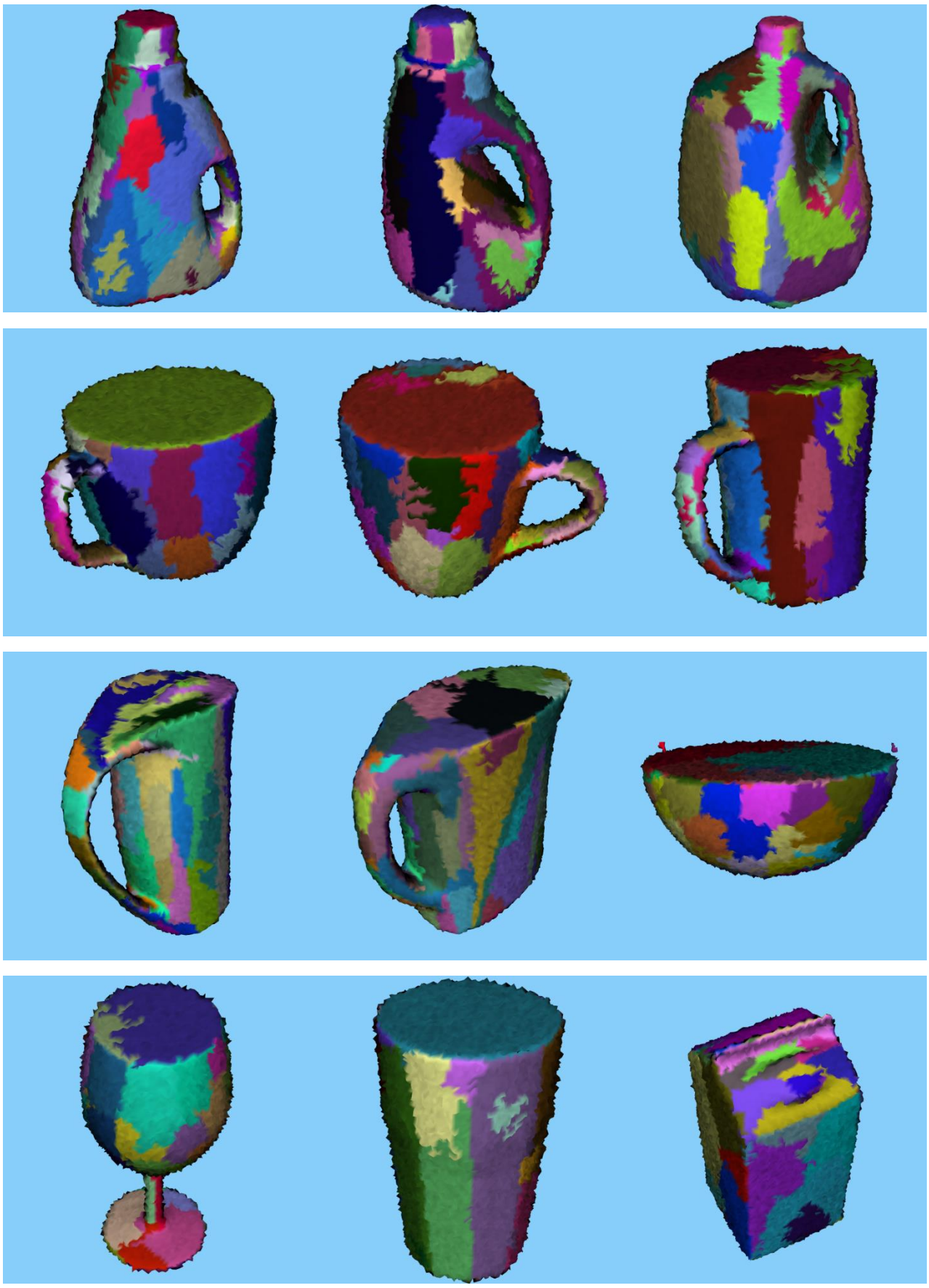

Fig. 1. Examples of planar surface segments obtained by applying the proposed method to noisy 3D models. Each segment is displayed with different color.

3. Object Recognition Problem

Let's consider the case where an object of interest is represented by a 3D point cloud, i.e. a set of points $M = \{P'_1, \quad P'_2, \quad \ldots, \quad P'_m\}$. Let the position of each point P'_j relative to the model reference frame S_M be defined by vector ${}^{M}\mathbf{p}'_j \in \mathbb{R}^3$. Let us now consider the task of finding the object of interest in a scene represented by a 3D point cloud. The task of finding the object of interest in the considered scene can be formulated as identifying a subset of the scene point cloud which resembles the object model point cloud according to a particular similarity criterion. More precisely, we search for the transformation between the model reference frame S_M and the scene reference frame S_C which maximizes the number of the transformed model points in the close vicinity of the scene points. Let $C = \{P_1, \quad P_2, \quad \ldots, \quad P_c\}$ be the scene point cloud. The considered problem can be formulated as maximization of the following *matching score*

$$\mathfrak{I}\left({}^{C}\mathbf{R}_M, {}^{C}\mathbf{t}_M\right) = \sum_{j=1}^{m} \mathbf{1}\left(\min_{i=1,\ldots,c} \left\| {}^{C}\mathbf{p}_i - \left({}^{C}\mathbf{R}_M \cdot {}^{M}\mathbf{p}'_j + {}^{C}\mathbf{t}_M\right)\right\| \le r\right) \tag{1}$$

where ${}^{C}\mathbf{R}_M$ and ${}^{C}\mathbf{t}_M$ represent the rotation matrix and translation vector defining the transformation between S_M and S_C respectively, ${}^{C}\mathbf{p}_i$ is a vector representing the position of point $P_i \in C$ relative to S_C, r is a threshold and **1**(*condition*) is the function which returns value 1 if *condition* is satisfied and 0 otherwise. If transformation $\left({}^{C}\mathbf{R}_M, {}^{C}\mathbf{t}_M\right)$ is found for which $\mathfrak{I}\left({}^{C}\mathbf{R}_M, {}^{C}\mathbf{t}_M\right)$ exceeds a predefined threshold, this indicates that the object of interest is present in the considered scene and that $\left({}^{C}\mathbf{R}_M, {}^{C}\mathbf{t}_M\right)$ represents a good estimate of its pose relative to the scene reference frame S_C.

The search for the transformation which maximizes matching score (1) can be made more efficient by subsampling the model point cloud, i.e. by representing the model with a reduced number of points and assigning each sample point the local surface normal. Let each point $P'_j \in M$ be assigned a unit vector ${}^{M}\mathbf{n}'_j$ representing the local object surface normal at that point, i.e. the vector orthogonal to the object surface in the close vicinity of the point P'_j. A pair $\left({}^{M}\mathbf{p}'_j, {}^{M}\mathbf{n}'_j\right)$ can be regarded as an *oriented point*. Since oriented points are more informative than vectors ${}^{M}\mathbf{p}'_j$ alone, a smaller number of oriented points can be used to describe the shape of an object.

By assigning surface normal to scene points, a set of oriented 3D points $\left({}^{C}\mathbf{p}_i, {}^{C}\mathbf{n}_i\right)$ is obtained. This additional surface normal information can be used in the object recognition process by replacing the matching score (1) with the following matching score

$$\mathfrak{I}\left({}^{C}\mathbf{R}_M, {}^{C}\mathbf{t}_M\right) = \sum_{j=1}^{m} \max_{i=1,\ldots,c} \left\{\mathbf{1}\left(\left\| {}^{C}\mathbf{p}_i - \left({}^{C}\mathbf{R}_M \cdot {}^{M}\mathbf{p}'_j + {}^{C}\mathbf{t}_M\right)\right\| \le r \wedge {}^{C}\mathbf{n}_i^T \cdot {}^{C}\mathbf{R}_M \cdot {}^{M}\mathbf{n}'_j \ge \varepsilon_n\right)\right\}, \tag{2}$$

where ε_n is a predefined threshold, and searching for transformation $\left({}^{C}\mathbf{R}_M, {}^{C}\mathbf{t}_M\right)$ which maximizes matching score (2).

4. Features for Object Recognition Based on Planar Surface Segments

The most common features used for object recognition are point features. We designed a novel feature, which is based on planar surface segments discussed in Section 2. Our feature detection method provides keypoints with stable reference frames. The proposed feature is named *Plane/Line Intersection Point (PLIP)*. The feature detection method considered in this paper takes as input a 3D triangular mesh and produces a set of PLIPs with assigned reference frames. The triangular mesh can be obtained from a point cloud using one of the available methods. The details of our approach are not presented in this paper. The complete description of the proposed approach will be published in a journal or conference paper. In this section, the analysis of the stability of the PLIP reference frame (RF) and its applicability in object recognition is presented.

The applicability of PLIP for object recognition can be evaluated by detecting PLIPs in model and scene point clouds and computing transformations which align model PLIPs with scene PLIPs. The PLIP method is applicable for object recognition if at least one of the obtained transformations is correct. The correctness of the transformations computed by PLIP alignment can be evaluated by comparing them to the known ground truth object poses.

Let's consider a PLIP detected on the model mesh, with the frame $S_{F'}$ whose origin is identical to the PLIP. Let the position of the origin of $S_{F'}$ relative to the model reference frame S_M be represented by vector ${}^{M}\mathbf{t}_{F'} \in \mathbb{R}^3$ and its orientation by rotation matrix ${}^{M}\mathbf{R}_{F'} \in \mathrm{SO}(3)$. Assuming that a PLIP detected on the surface of the object of interest placed on a scene has the same position and orientation relative to this object as one of the PLIPs detected on the surface of the object model, the alignment of the reference frame S_F of the scene PLIP and the reference frame $S_{F'}$ of the model PLIP result in the object pose relative to the scene reference frame S_C. In this case, the object pose can be computed by

$$ {}^{C}\mathbf{T}_{M} = {}^{C}\mathbf{T}_{F} \cdot {}^{M}\mathbf{T}_{F'}^{T} \ , \tag{3} $$

where ${}^{B}\mathbf{T}_{A}$ represents the homogenous transform matrix corresponding to the pose of a reference frame S_A relative to a reference frame S_A defined by

$$ {}^{B}\mathbf{T}_{A} = \left[\begin{array}{ccc|c} & {}^{B}\mathbf{R}_{A} & & {}^{B}\mathbf{t}_{A} \\ \hline 0 & 0 & 0 & 1 \end{array}\right]. $$

Matrix ${}^{C}\mathbf{T}_{F}$ in (3) represents the pose of S_F relative to S_C.

PLIPs can be used in combination with existing local descriptors such as SHOT (Tombari, Salti and Di Stefano, 2010), FPFH (Rusu, Blodow and Beetz, 2009), spin image (Johnson and Hebert, 1999) etc. In our future research, we plan to develop a hypothesis generation approach which is based on the matching score (2). In order to evaluate the discriminating power of the considered approach, we performed experiments in which object pose hypotheses are generated by PLIP alignment and evaluated using a descriptor based on the matching score (2).

The descriptor used in this paper is a set of oriented points, i.e. randomly selected vertices $P'_j \in M$ with assigned normals. It can be defined as set $D = \left\{\left(\mathbf{p}_{d,1}, \mathbf{n}_{d,1}\right), \ \left(\mathbf{p}_{d,2}, \mathbf{n}_{d,2}\right), \ \ldots, \ \left(\mathbf{p}_{d,n}, \mathbf{n}_{d,n}\right)\right\}$,

where $\mathbf{p}_{d,l}$ and $\mathbf{n}_{d,l}$ are the position vector and normal of a randomly selected model point represented with respect to the reference frame $S_{F'}$, i.e. for each l there is $j \in \{1, 2, \ldots, m\}$ such that

$$\mathbf{p}_{d,l} = {}^{M}\mathbf{R}_{F'}^{T}\left({}^{M}\mathbf{p}'_{j} - {}^{M}\mathbf{t}_{F'}\right),$$

$$\mathbf{n}_{d,l} = {}^{M}\mathbf{R}_{F'}^{T}\,{}^{M}\mathbf{n}'_{j}.$$

The discussed descriptor is formed from model points which are relatively close to the origin of $S_{F'}$, i.e. for $1 \le l \le n$

$$\left\|\mathbf{p}_{d,l}\right\| \le r_F,$$

where r_F is a distance threshold. The described descriptor is referred to in this paper as *Random Oriented Point (ROP) descriptor*.

An object recognition procedure based on PLIP feature detector and the proposed descriptor is given in Algorithm 1. In this algorithm, the object of interest is represented by a single PLIP feature and the corresponding ROP descriptor. The algorithm returns a hypothesis list. Each hypothesis H in that list represents a pair $\left({}^{C}\mathbf{T}_{M}, {}^{C}\mathbf{T}_{F,i}, s\right)$, where ${}^{C}\mathbf{T}_{M}$ is the pose of the object in the scene, ${}^{C}\mathbf{T}_{F,i}$ is the pose of the PLIP RF and s is a matching score analogous to (2). The hypothesis on the top of the list can be considered as the most probable one.

Algorithm 1 *Object recognition based on PLIP and ROP descriptor*

Input: depth image acquired by a 3D sensor, r, ε_n
model PLIP F' represented by transformation matrix ${}^{M}\mathbf{T}_{F'}$ and descriptor $D = \left\{\left(\mathbf{p}_{d,l}, \mathbf{n}_{d,l}\right)\right\}$, $l = 1, 2, \ldots, n$

Output: list of hypotheses H sorted according to matching score s

Convert depth image into a triangular mesh.
Detect PLIPs in the scene mesh. Each scene PLIP F_i is assigned a transformation matrix ${}^{C}\mathbf{T}_{F,i}$.
Create empty hypothesis list.
For each detected scene PLIP
    $s \leftarrow 0$
    For $l \leftarrow 1$ **to** n
        $\mathbf{p}' \leftarrow {}^{C}\mathbf{R}_{F,i} \cdot \mathbf{p}_{d,l} + {}^{C}\mathbf{t}_{F,i}$
        $\mathbf{n}' \leftarrow {}^{C}\mathbf{R}_{F,i} \cdot \mathbf{n}_{d,l}$
        If there is a scene point $\mathbf{p}$ with normal $\mathbf{n}$ such that $\left\|\mathbf{p} - \mathbf{p}'\right\| \le r$ and $\mathbf{n}^{T} \cdot \mathbf{n}' \le \varepsilon_n$ **then**
            $s \leftarrow s + 1$
        end if
    end for
    ${}^{C}\mathbf{T}_{M} \leftarrow {}^{C}\mathbf{T}_{F,i} \cdot {}^{M}\mathbf{T}_{F'}^{T}$
    $H_i \leftarrow \left({}^{C}\mathbf{T}_{M}, {}^{C}\mathbf{T}_{F,i}, s\right)$
    Add H_i to the hypothesis list
end for
Sort hypothesis list according to s in descending order.
return hypothesis list

4.1. Reference Frame Stability

In order to test the stability of PLIP RF with respect to the measurement noise, we performed an experiment with a 3D object model taken from the TUW database used in (Aldoma et al., 2012b). The object used in the discussed experiment is a bottle shown in Fig. 2. The original mesh is uniformly sampled and normals are estimated by MeshLab software (MeshLab). A single PLIP is extracted from the model mesh, a ROP descriptor consisting of 1000 oriented points is generated and this feature is used as the model feature. Then, Gaussian noise with standard deviation of 1 mm is superimposed on the same model mesh and the resulting mesh is used as a scene. An example of noisy model obtained by superimposing Gaussian noise to the model in Fig. 2 is shown in the top left image of Fig. 1.

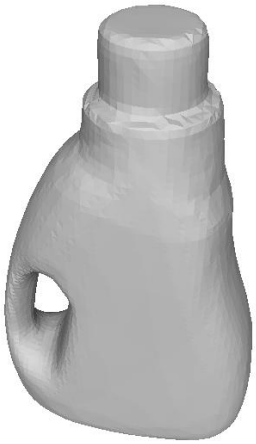

Fig. 2. 3D mesh from TUW database used as a model of an object of interest.

Furthermore, the normals assigned to mesh vertices are corrupted by adding a 3D vector sampled from Gaussian distribution with standard deviation of 0.05 and normalizing the obtained vector to unit vector. The superimposed noise causes the applied segmentation algorithm to generate different segments from those obtained by applying the same segmentation approach to the original mesh. The purpose of this noise addition is to simulate the measurement noise which is commonly present in point clouds acquired by a real sensor and to test the robustness of the PLIP detection method to the measurement noise. The stability of the PLIP RF is evaluated by measuring the deviation of the PLIP RF obtained after the noise addition to the original PLIP RF. This deviation is measured by the Euclidean distance between the reference frame origins and the rotation angle needed to align the axes of the two reference frames.

The experiment is performed by adding noise to the given model, detecting PLIPs in that noisy model and applying Algorithm 1 to that model. The parameters of this algorithm were set to the following values: r = 10 mm, ε_n = 0.866, r_F = 200 mm. The highest ranked hypothesis from the list generated by this algorithm is taken as the solution. The pose of the PLIP RF S_F corresponding to this hypothesis in the noisy model relative to the pose of the PLIP RF $S_{F'}$ in the original model is computed by

$$ {}^{F'}\mathbf{T}_{F,i} = {}^{M}\mathbf{T}_{F'}^{T} \cdot {}^{C}\mathbf{T}_{F,i} \cdot $$

The position error is computed by

$$e_t = \left\| {}^{F'}\mathbf{t}_F \right\|.$$

The orientation error is computed by determining the axis and the angle for which $S_{F'}$ must be rotated about this axis to align with S_F. The histogram of the position and orientation error resulting from 110 experiments are shown in Fig. 3. Since the noise added to the original model is generated by random sampling from Gaussian distribution, a different model is obtained in each experiment.

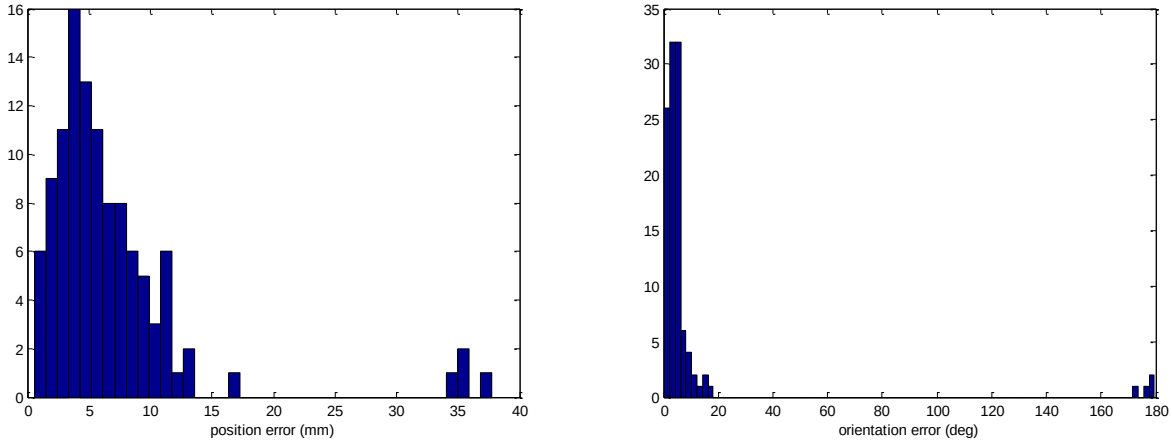

Fig. 3. Histograms of the position and orientation error recorded from 100 experiments of matching a PLIP extracted from the model shown in Fig. 2 to the PLIPs extracted from the same model corrupted by noise.

The normalized cumulative histograms[1] of the position and orientation error are shown in Fig. 4.

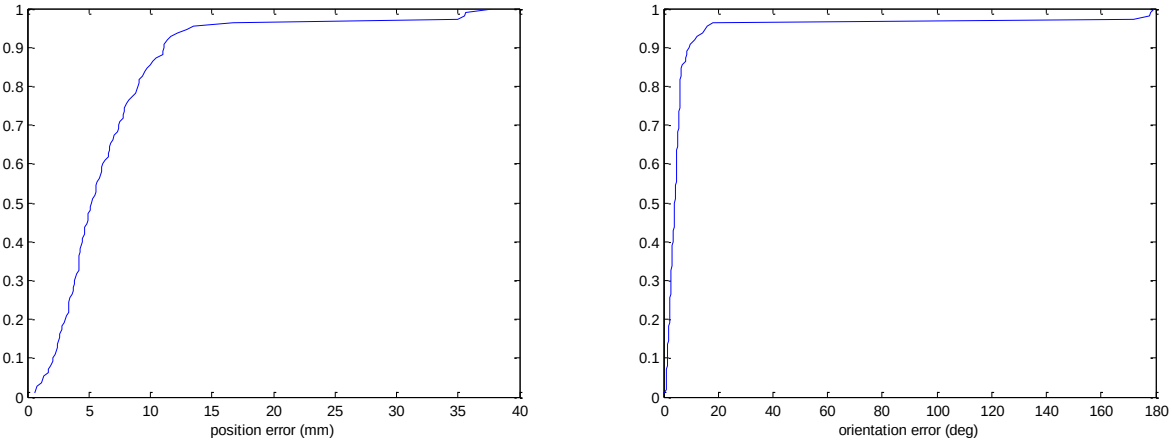

Fig. 4. Normalized cumulative histograms of the position and orientation error recorded from 110 experiments of matching a PLIP extracted from the model shown in Fig. 2 to the PLIPs extracted from the same model corrupted by noise.

In 95% experiments the position error was below 14 mm and the orientation error was below 16° in the same percentage of experiments. The results of four experiments were outliers, i.e. the cases where the PLIP RF of the highest ranked hypothesis differ for almost 180° from the model PLIP RF.

[1] A normalized cumulative histogram is a data representation where the horizontal axis corresponds to values of a measured variable x and the vertical axis represents the percentage of measurements which are $\geq x$.

4.2. Precision and Recall

The combination of PLIP feature detector and ROP descriptor was evaluated by experiments in which Algorithm 1 was applied to a set of 23 models from TUW database, where all these models are matched to the reference model shown in Fig. 2, which was also used in the experiment described in Section 4.1. Analogously to the experiment described in Section 4.1, a single PLIP with its ROP descriptor is used to describe the reference model. The considered models are corrupted by Gaussian noise in the same way as in Section 4.1. From each of 22 models, 5 noisy models were generated, while for the reference model 110 noisy models are generated. Hence, the total of 220 noisy models was generated, where half of them represent the object of interest, while the others represent other objects from the TUW database. Algorithm 1 returns a list of hypotheses, where each hypothesis is assigned a matching score. The purpose of the experiment reported in this section is to determine the discriminative power of the combination of PLIP feature detector and ROP descriptor, i.e. to determine if there is a threshold for matching score which separates the object of interest from other objects. The results of Algorithm 1 are categorized according to a given threshold τ in four sets:

- true positives (TP): noisy models of the object of interest for which the highest match score is $\geq \tau$;
- false negatives (FN): noisy models of the object of interest for which the highest match score is $< \tau$;
- false positives (FP): noisy models of other objects for which the highest match score is $\geq \tau$;
- true negatives (TN): noisy models of other objects for which the highest match score is $< \tau$.

The precision is computed by

$$PR = \frac{TP}{TP + FP},$$

and recall by

$$RC = \frac{TP}{TP + FN}.$$

By computing precision and recall for different values of the threshold τ precision-recall curves are obtained. In order to investigate how the discriminating power of the ROP descriptor depends on its size, we computed precision-recall curves for 5 different descriptor sizes, where descriptor size corresponds to the number of oriented points used to build the descriptor: 1000, 400, 150, 50 and 20. The resulting precision-recall curves are shown in Fig. 5.

Values of the threshold τ for which 100% precision is achieved as well as the values for which 100% recall is achieved are shown in Table 1. It can be concluded that for descriptor sizes 1000, 400 and 50 there is an interval of threshold values which clearly separate the object of interest from the other objects.

It should be noted that this analysis is performed on synthetically created meshes, which represent complete object shapes. In reality, however, objects are only partially visible and often occluded by

other objects. Nevertheless, the presented analysis demonstrates the potential of the investigated approach.

Table 1. Values of the threshold τ for which 100% precision and 100% recall is achieved.

Descriptor size	100% precision	100% recall
1000	≥ 703	≤ 761
400	≥ 287	≤ 298
150	≥ 109	≤ 99
50	≥ 37	≤ 37
20	≥ 16	≤ 15

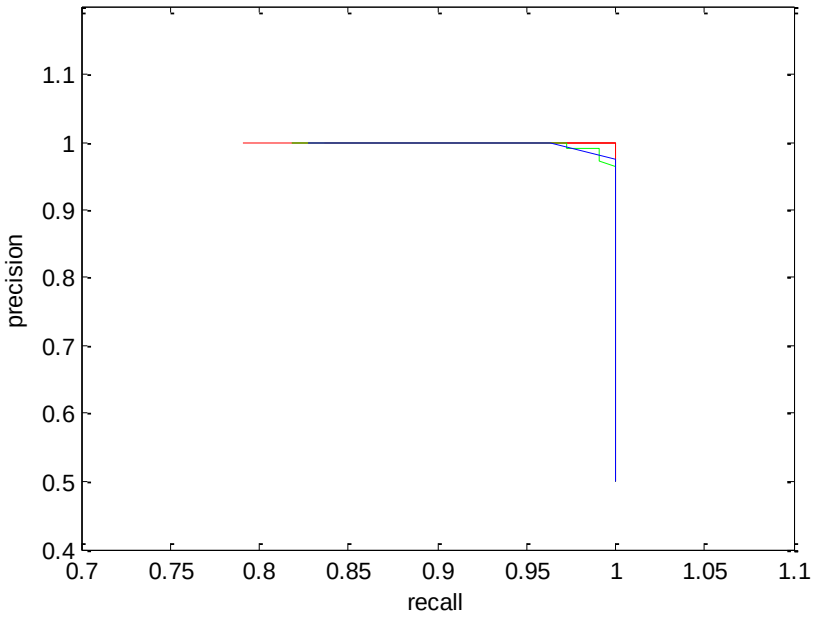

Fig. 5. Precision-recall curves for different descriptor sizes: 1000, 400, 50 (red), 150 (green) and 20 (blue).

References

Achanta R, Shaji A, Smith K, Lucchi A, Fua P and Süsstrunk S (2012) SLIC Superpixels Compared to State-of-the-Art Superpixel Methods, IEEE Transactions On Pattern Analysis And Machine Intelligence, vol. 34, no. 11, pp. 2274–2281

Aldoma A, Tombari F, Rusu RB, and Vincze M (2012a) OUR-CVFH – Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation, *Joint DAGM-OAGM Pattern Recognition Symposium*.

Aldoma A, Tombari F, Di Stefano L and Vincze M (2012b) A global hypothesis verification method for 3d object recognition, *European Conference on Computer Vision (ECCV)*

Cupec R, Grbić R and Nyarko EK (2016) Survey of State-of-the-Art Methods for Object Recognition in 3D Point Clouds, Technical Report ARP3D.TR2

Gupta S, Arbeláez P, and Malik J (2013) Perceptual Organization and Recognition of Indoor Scenes from RGB-D Images, Computer Vision and Pattern Recognition (CVPR)

Holz D and Behnke S (2012) Fast Range Image Segmentation and Smoothing using Approximate Surface Reconstruction and Region Growing, *International Conference on Intelligent Autonomous Systems (IAS)*, Jeju Island, Korea

Holz D and Behnke S (2014) Approximate Triangulation and Region Growing for Efficient Segmentation and Smoothing of Range Images, Robotics and Autonomous Systems, vol. 62, no. 9, pp. 1282–1293

Johnson A and Hebert M (1999) Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , vol. 21, pp. 433–449

MeshLab, Visual Computing Lab - ISTI – CNR, http://meshlab.sourceforge.net/

Narayanany V and Likhachev M (2016) PERCH: Perception via Search for Multi-Object Recognition and Localization, *IEEE International Conference on Robotics and Automation (ICRA)*

Papon J, Abramov A, Schoeler M and Wörgötter F (2013) Voxel Cloud Connectivity Segmentation - Supervoxels for Point Clouds, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2027–2034.

Rusu RB and Cousins S (2011) 3D is here: Point Cloud Library (PCL), *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China

Rusu RB, Blodow N and Beetz M (2009) Fast Point Feature Histograms (FPFH) for 3D Registration, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3212–3217

Simari P, Picciau G and De Floriani (2014) Fast and scalable mesh superfacets. Computer Graphics Forum, vol. 33, no. 7, pp. 181–190.

Steder B, Rusu RB, Konolige K and Burgard W (2010) NARF: 3D Range Image Features for Object Recognition, *Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan.

Tombari F (2013) How does a good feature look like? *IEEE International Conference on Robotics and Automation (ICRA), PCL Tutorial*

Tombari F, Salti S and Di Stefano L (2010) Unique signatures of Histograms for local surface description, *European Conference on Computer Vision (ECCV)*