Project IP-2014-09-3155

Advanced 3D Perception for Mobile Robot Manipulators

SURVEY OF STATE-OF-THE-ART METHODS FOR OBJECT RECOGNITION IN 3D POINT CLOUDS

Technical Report

ARP3D.TR2

version 1.0

Robert Cupec, Ratko Grbić, Emmanuel Karlo Nyarko

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. Recognition of Fixed Shapes

The problem of recognition of objects of fixed shape can be formulated as in (Aldoma et al., 2013). 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.

3D object recognition methods can be classified in **two broad classes**: (i) local approaches and (ii) global approaches.

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). These approaches 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). Although less effective in presence of partial object occlusions, 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. Furthermore, this is beneficial also in terms of memory footprint, since a notably reduced amount of information needs to be stored to represent the model library. These properties make global pipelines appealing in scenarios where segmentation is feasible, objects do not present high levels of occlusions and efficiency represents a relevant constraint (Aldoma et al., 2012a).

The recognition of objects of fixed shape in 3D point clouds today is in a rather mature phase. State-of-the art algorithms which solve this problem achieve a high precision and recall. A comprehensive survey of 3D object recognition methods is given in **(Guo et al., 2014)**.

In **(Hinterstoisser et al., 2010)** a template matching approach to object recognition in RGB-D images is proposed. Each modelled object is represented by approximately 2000 templates, representing the object viewed from different angles and in different scales. An efficient implementation of template matching allows recognition of a single object in 0.1 s.

The method proposed in **(Papazov and Burschka, 2010)** 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. 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. Hypotheses are generated using RANSAC (Fischler and Bolles, 1981). 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 (de Berg M et al., 2010).

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 **(Aldoma et al., 2012b)**, 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 (Tombari, Salti and Di Stefano, 2010). Fast descriptor matching is accomplished by indexing implemented using FLANN (Muja and Lowe, 2009). 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 ε 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 the Iterative Closest Point (ICP) algorithm (Besl and McKay, 1992). 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 $\mathcal{X} = \{x_1, x_2, \ldots, x_r\}$. 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, it 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 using the method proposed in (Rabbani, van den Heuvel and Vosselman, 2006). 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.

The method presented in (Aldoma et al., 2012b) is augmented in **(Aldoma et al., 2013)** by two additional hypothesis generation pipelines, one which uses point features detected in the grayscale image with assigned SIFT (Lowe, 2004) descriptors and the other which uses a semi-global 3D descriptor representing an extension of the OUR-CVFH approach (Aldoma et al., 2012a) 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 hypothesis evaluation method proposed in (Aldoma et al., 2012b) 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.

The method presented in (Aldoma et al., 2013) is adapted in **(Aldoma, Fäulhammer and Vincze, 2014)** for object recognition from multiple views.

In **(Fäulhammer et al., 2015)**, an efficient online multi-view method which integrates information of the captured environment by merging individual single-view recognition outputs is proposed. The proposed approach is based on (Aldoma et al., 2014), where an online multi-view object instance recognition method merges single-view results in a batch to generate ground-truth data of a static environment. In typical every-day indoor environments, due to mainly stationary objects, the majority of the scene is static. Exploiting this static information, the authors propose a method using RGB-D data that continuously combines hypotheses constructed from various viewpoints into a common coordinate system. Thus, the system gathers the maximum amount of information for all objects in the scene, and overcomes the problem of single views that are particularly poor. Using a dynamic graph representation of the observed environment, the proposed approach enables the system to improve recognition with each new observation. The system recognizes pre-trained models and their respective 6DoF pose at each time step using information from current and previous observations. The proposed method showed a significant improvement in recall for the TUW dataset, consisting of heavily cluttered RGB-D frames with textured and non-textured objects, which are highly occluded in some frames. State-of-the-art results were also obtained for the Willow dataset at a reduced computational complexity.

The approach proposed in **(Tang et al., 2012)** 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 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 (Tang et al., 2012) is improved in **(Xie et al., 2013)** 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 (SVM) is applied to determine which object corresponds to each cluster from *S*.

The approach proposed in **(Choi and Christensen, 2013)** uses Color Point Pair Features, obtained by augmenting the descriptor proposed in (Winkelbach , Molkenstruck and Wahl, 2006) 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.

The approach proposed in **(Narayanany and Likhachev, 2016)** is capable of detecting multiple objects in the scene which can be mutually occluded. However, it relies on several unrealistic assumptions, which limit significantly its practical applicability. It is assumed that all objects in the scene are placed on a flat horizontal surface in the nominal upright position and uses RANSAC to determine the camera pose relative to that horizontal surface, so that only 3DoF object pose must be determined. Furthermore, it is assumed that the number of objects in the scene is known ahead and that only modelled objects are present in the scene. The problem of finding the correct scene interpretation is formulated as an optimization problem – minimizing an explanation cost function. The design of this cost function allows it to be decomposed over the set of objects searched in the scene. The search for the optimal solution is based on building of a monotone scene generation tree starting from the empty scene and adding objects one by one. Each object insertion corresponds to addition of a new node in the tree, i.e. each node represents a hypothetical scene which is rendered and evaluated according to the explanation cost function. The constraint in this process is that the addition of the new object does not occlude the scene generated by the previously inserted objects. Thereby, the algorithm generates and renders all feasible scenes in a discretized pose space, given the aforementioned constraints. The optimization is performed by multi-heuristic search using FOCAL-MHA* algorithm (Narayanan, Aine and Likhachev, 2015). For a scene, the mean planning time was 6.5 min.

Many object recognition methods such as (Papazov and Burschka, 2010; Aldoma et al., 2012b; Aldoma et al., 2013) require a 3D object model in form of a triangular mesh or a point cloud. Although such models can be obtained by 3D scaners, a useful ability of a mobile robot manipulator is to create 3D models of objects of interest automatically by moving around an object of interest and acquiring RGB-D images of the object from multiple views. An approach for creating 3D object modles suitable for object recognition by fusion of partial 3D models obtained by several 3D reconstruction sessions is proposed in **(Prankl et al., 2015)**. Each 3D reconstruction session assumes tracking of an object accross a sequence of RGB-D images using visual odometry based on tracked interest points. The camera positions are refined by means of bundle adjustment as well as an accurate multi-view ICP approach introduced in (Fantoni, Castellani and Fusiello, 2012). Segmentation of the object of interest from the background is attained by a multi-plane detection and a smooth clustering approach. Flat parts, larger than a certain threshold are modelled as planes and the remaining areas are recursively clustered depending on the deviation of the surface normals of neighbouring image points. Hence, smooth clusters "pop out" from the surrounding planar surfaces and need to be selected to form up a complete object. Multi-session data fusion is achieved by registration of partial 3D models, where Correspondences between bodies are obtained by matching SIFT and SHOT features (capturing thus both appearance and geometrical information), correspondence grouping stage followed by RANSAC and absolute orientation estimation. Allignment of partial 3D models is facilitated using the modelling constraint that objects lie on a planar surface. Aligning planes locks 3 of the 6 degrees of freedom involved in rigid body registration. The remaining 3 degrees of freedom (i.e. translation on the plane and rotation about the plane normal) are respectively approximated by centering the point cloud and by sampling rotations about the plane normal (every 30°). The initial alignments that are then refined by ICP. The authors

made the source code of their system named *RTM - Recognition, Tracking and Modelling Toolbox* publically available at http://www.acin.tuwien.ac.at/?id=450.

In **(Kouskouridas et al., 2016)**, a method for object detection and 6 DoF pose estimation in heavily cluttered and occluded scenarios is proposed. The Latent-Class Hough Forests, a novel patch-based approach to 3D object detection and pose estimation; It performs one-class learning at the training stage, and iteratively infers latent class distributions at test time. Two new more challenging public datasets for multi-instance 3D object detection and pose estimation are presented, comprising near and far range 2D and 3D clutter as well as foreground occlusions in domestic and industrial scenarios.

In **(Doumanoglou et al., 2015)**, object detection and pose estimation is considered but the paper also addresses the problem of next-best-view estimation. In the first part, the training objects are rendered and depth-invariant RGB-D patches are exctracted. The latter are given as input to a Sparse Autoencoder which learns a feature vector in an unsupervised manner. Using this feature representation, a Hough Forest is trained to recognize object patches in terms of class and 6D pose (translation and rotation). Given a test image, patches from the scene pass through the Autoencoder followed by the Hough forest, where the leaf nodes cast a vote in a 6D Hough space indicating the existence of an object.

In **(Savarimuthu et al., 2015)**, an integrated system for recognition, pose estimation and simultaneous tracking of multiple objects in 3D scenes is presented. The system solves the problem of complete semantic representation of dynamic scenes which requires three essential steps: recognition of objects, tracking their movement and identification of interactions between them. The focus is on maintain tracks of multiple interacting objects even in the presence of long durations of full occlusions (which usually appears in even simple assembly tasks) because losing object identify makes it impossible to recover an accurate semantic understanding of observed actions. The proposed tracking system is composed of four modules: object recognition, pose estimation, tracking and SEC (Semantic event Chains). The system is initialized by the first two modules in combination, i.e. each object in a scene is recognized and its pose is estimated. As stated in article, correct detection of objects is crucial to the performance of the running system, but also requires a high amount of computational resources. Therefore, at first frame of a sequence, robust object recognition and pose estimation techniques are applied to extract the identity and location of each object present in the scene. After the detection of objects, the real-time tracker maintains the pose of the objects.

Modules for object recognition and pose estimation operate on a specialized set of 3D feature points called texlets, which are used for representing local surfaces. At first objects in the scene have to be separable by segmentation procedure. Then, for each recognized segment in the scene, texlets are extracted. Two dimensional histogram is computed globally on an object surface represented by a texlets. This two dimensional histogram is placed into single array and thus forming input to Random Forest (Breiman, 2001) Classifier which is used for object recognition. For each recognized object, a robust feature based estimation is executed (method uses local histogram descriptors computed in a local region around each texlet, defined by a support radius)

The proposed method is compared with ICP and Particle Filter based trackers and to a "ground truth tracking" by magnetic sensors. The experiments are restricted to table top scenarios. Tracking is performed in real-time.

Article deficiencies: there is no information about random forest classifier building (the article focus is on tracking rather than object recognition).

2. Recognition of Object Classes

In **(Nan, Xie and Sharf, 2012)** an approach for indoor 3D scene interpretation is proposed which simultaneously segments the input point cloud and classifies segmented objects. The features used for object classification are segments obtained by analyzing point distribution along the upward axis and segmenting a point cloud into horizontal slabs according to this analysis. For each slab, the bounding box is computed and a feature vector is formed from the parameters of the bounding boxes of the slabs. In the learning phase, an RDF classifier is trained using supervised learning with various indoor objects. The applied training set consisted of manually segmented and labeled scans of roughly 1000 different objects (e.g. 20 beds, 110 cabinets, 510 chairs, 40 monitors, 250 tables, etc.). Both synthetic and scanned objects are used. In the recognition phase, the input point cloud is segmented into a set of piecewise smooth patches. An adjacency graph is computed with nodes corresponding to patches and edges connecting close patches. The search-classify procedure starts by selecting several random patch triplets representing the initial object. The classification likelihood is computed for each patch triplet. The triplets with very low classification likelihood are removed. The procedure continues by growing the initial object and computing the classification likelihood value for the grown object. The current object is grown by examining each neighboring patch (patch connected to an object patch), computing the classification likelihood defined by the union of the current object and the neighboring patch, selecting the patch corresponding to the highest classification likelihood and adding this patch to the current object. The growing procedure stops when the classification likelihood of the object obtained by adding a patch to the current object is less than 0.95 of the classification likelihood of the current object. The result of the search-classify procedure is a segmented object point cloud with assigned class to which a template fitting is applied. The template which is fitted to the object point cloud is a polygonal template with predefined scalable parts. The template fitting is performed by an iterative procedure in which two steps are alternated: computing the closest distances between points in the object point cloud and the template and deforming the template to minimize this distance. After each deformation step, the local deformation scales are optimized using pre-analyzed part information for each template model. The optimization iteratively restores scale relations such as symmetries and proximities between parts of the template while refitting it to the point cloud. Outliers are detected in the search-classify process as patches with a large Euclidean distance to the fitted template and removed from the object, thereby refining the segmentation.

The algorithm presented in **(Kim et al., 2013)** learns a probabilistic model of part-based template variations, given a set of 3D meshes of object of a particular class and an initial deformable model represented by a template encoding the types of parts expected in the repository and an initial guess for the locations and scales of each part. The algorithm simultaneously segments polygonal models into parts, learns a probabilistic model of part-based template variations, and establishes point-to-point surface correspondences in large collections of shapes.

In **(Mueller, Pathak and Birk, 2014)**, an approach for object class recognition using a part-based shape categorization in RGB-D images is proposed. Initially the RGB-D image of a scene is over-segmented with the Mean-Shift (Comaniciu, 2002). The result are homogeneous patches which are used as fundamental shape elements. These patches are grouped by a region growing procedure which results in a set of patches which can cover semantically reasonable parts of object instances. These merged patches are referred to as super patches. Each super patch is described by a shape descriptor proposed in (Mueller, Ploeger and Roscoe, 2012). The resulting description is quantized, using an unsupervised learning approach, in a dictionary containing symbols referred to as visual words. For training the dictionary, unlabeled super patches are collected from random scenes and applied to the clustering procedure. A fast k-Means clustering algorithm was used with the objective to group similarly-shaped super patches to the same words. A hierarchical top-down quantization approach is applied which allows to learn shape part constellation models on different levels of specificity (the specificity levels can be understood as a general-to-specific partition of the description space or, in the context of shape appearance, a rough-to-specific expression of detailedness). For each shape category, a graphical model is generated based on the appeared visual word constellations of training instances from object shape category. This graphical model encodes a set of unary, pairwise and higher order relationships between the observed visual words, for instance the appearance, euclidean neighborhood or alignment. The detection of the actual instance in a scene is solved by the convex-alignment assumption between super patches. Groups of neighboring patches which are aligned to each other in a convex pose are selected as potential object instance candidates. In the inference, the probability that an observed object instance belongs to a shape category given the constellation of visual words is computed. The final solution is obtained by solving the maximum a posteriori (MAP) problem, which is formulated as energy minimization problem. The energy function which is minimized is defined as the sum of costs of particular relations between visual words. These costs are determined using statistics and they can be interpreted as the confidence that a particular word or a particular relation between words appears in a particular category. The proposed approach is tested on recognition of sacks, barrels and parcels.

A significant volume of research in the field of recognition of object classes in RGB images is inspired by a successful application of Convolutional Neural Networks (CNN) in ImageNet Large Scale Visual Recognition Challenge **(Krizhevsky, Sutskever and Hinton, 2012)**. In comparison to standard neural networks with similarly sized layers, CNNs have much fewer connections and parameters and therefore are easier to train with maybe slightly worse performance. CNN used in (Krizhevsky, Sutskever and Hinton, 2012) consists of 5 convolutional and 3 fully connected layers and has approximately 60 million parameters. Usually very long training time of CNNs is shortened by high-performance C++/CUDA implementation.

Other significant work in the field of object recognition using CNNs is reported in (Sermanet et al., 2014; Girshick et al., 2014; Girshick, 2015; Ren et al., 2015; Razavian et al., 2014; Simonyan and Zisserman, 2015; He et al., 2015). The success of CNNs in object recognition in RGB images also motivated application of CNNs for object recognition in RGB-D images.

In **(Eitel et al., 2015)** an object recognition approach based on CNN is proposed. The proposed RGB-D architecture for object recognition consists of two separate CNN processing streams which are consecutively combined with a late fusion network. CNNs are pretrained by ImageNet (Russakovsky et al., 2015). Depth images are encoded as a rendered RGB images, spreading the

information contained in the depth data over all three RGB channels, and then a standard (pretrained) CNN is used for recognition. Due to lack of large scale labeled depth datasets, CNN pretrained on ImageNet (Krizhevsky, Sutskever and Hinton, 2012) are used. A novel data augmentation that aims at improving recognition in noisy real-world setups is proposed. The approach is experimentally evaluated using two datasets: Washington RGB-D Object Dataset and RGB-D Scenes dataset.

Another object recognition approach which uses deep CNN is proposed in **(Schwarz, Schulz and Behnke, 2015)**. It also uses CNN which is pre-trained for image categorization and provide a rich, semantically meaningful feature set. The depth information is incorporated by rendering objects from a canonical perspective and colorizing the depth channel according to distance from the object center.

In **(Rock et al., 2015)**, an approach for recovering a complete 3D model from a depth image of an object is proposed. The reason why it is included in this report is that it performs recovery of a similar object to the input object from a model database, which actually represents recognition of object from a particular class. Furthermore, the ability to infer complete 3D shape from a single view is important for grasping, as we often reach around an object to grasp its unseen surfaces. Guessing the shape of object is based on past experience. Input to the system is segmented depth image (query) for which a complete 3D mesh has to be produced. First step is finding a similar depth images from a training set of meshes. Random forests are used to produce candidates. The candidate with minimum surface-to-surface distance based on the query and retrieved depth images is selected to produce an exemplar mesh and camera viewpoint. The retrieved model may be of a different instance or category, and is likely to have a slightly different viewpoint. Therefore, the retrieved mesh is deformed to better aproximate the depth image. In final step, the deformed exemplar is used to complete the input depth image. For particular query, retrieving similar examplar 3D meshes from training dataset can be complex since traing dataset contains mesh rendered from many viewpoints. Random forest are used to partiton training data set into similar 3D shapes based in features of a depth image. Each tree of the forest maps input features to a set of training examples. The trained forest contains five trees where each leaf contain five or fewer examples. Leaf nodes tend to group similar objects. Random forest returns 15 to 25 potential matches. The match that has minimum surface-to-surface distance based on query and retrieved depth images is chosen. Paper introduces a synthetic dataset built from the SHREC12 mesh classification dataset.

3. Deformable Objects

Due to the high dimensionality of the state spaces of deformable objects, perceiving deformable objects is much more difficult than perceiving rigid objects. Often, self-occlusions make it impossible to infer the full states of deformable objects from a single view. In addition, many deformable objects of interest lack distinguishable key-points (Schulman et al., 2013).

In **(Schulman et al., 2013)** a probabilistic approach for tracking deformable objects is proposed. A deformable object is modelled by a set of K 3D points $\mathbf{m}_1$, $\mathbf{m}_2$, …, $\mathbf{m}_K$ connected in a mass-spring system on a triangular mesh. At each time step, the sensor hardware generates from the visible portion of the object a point cloud consisting of N 3D points $\mathbf{c}_1$, $\mathbf{c}_2$, …, $\mathbf{c}_N$. For each observation

(point cloud), the algorithm's task is to find the most probable node positions given the measurement, i.e., to calculate

$$\arg\max_{\mathbf{m}_{1:K}} p\left(\mathbf{m}_{1:K} \middle| \mathbf{c}_{1:N}\right)$$

To solve this maximum a posteriori (MAP) estimation problem, we use the expectation maximization (EM) algorithm. The EM algorithm alternates between generating a lower bound to the log-probability function based on an expectation over the latent variables (E step) and maximizing this lower bound with respect to the node positions (M step). The latent variables in this case are correspondences between the sensor points $\mathbf{c}_i$ and the model points $\mathbf{m}_j$. In E step, correspondences between the sensor points and model points are established. Each correspondence is given a weight proportional to the estimated probability that a particular sensor point represents a particular model point. This probability is computed according to the sensor measurement model and visibility constraint imposed by self-occlusion and occlusion by other objects in the scene. The maximization in M step is performed using physics-based simulation engine. An artificial observation potential energy is defined, which includes gravitational potential energy and bending energy as well as the energy resulting from virtual forces which attract model points towards sensor points. The attraction force is proportional to the weight of the point correspondences determined in E step. The algorithm is applied to tracking of rope and cloth under arbitrary manipulations including knot tying and folding. The reported implementation runs in real time and is suitable for closed-loop control in robotic manipulation tasks.

In **(Twardon and Ritter, 2015)** a textile manipulation system is described, which is based on identification of boundary components of clothes. Given a boundary component, an optimal grasping pose is determined. The boundary components are identified by a graph-based approach, where edge contours detected in the input depth image are represented by a graph, simple cycles are detected in this graph and a simple cycle is selected as a boundary component using a heuristic approach. The depth image is converted to a graph representation in three steps: (i) normal-based edge detection, (ii) thinning/skeletonization and (iii) contour graph creation. The applied skeletonization algorithm iteratively removes boundary points preserving the end points and pixel connectivity. The graph nodes are identified as points with at least three neighbors in the image and the contour edges linking the nodes are identified. The proposed approach is applied for solving the task of hanging up a knit cap on a hat-stand.

In **(Li, Chen and Allen, 2014)** an approach for recognition of clothes is proposed, which recognizes several previously modelled garments hanging by different grasping points and identifies the grasping point. The approach is based on bag-of-feature (BOF) idea. For each input garment model, a set of 20–50 grasping points is predefined in terms of the garment categories. For each of the grasping points, a simulation of draping under gravity is carried out and rendered when the garment achieves a stable state (e.g. no shaking). The simulated 90-cameras system captures depth images of the model from different viewpoints. SIFT is applied over the generated depth image set. A codebook is built by sparse coding approach. After sparse coding on the feature vector, spatial pyramid construction is applied to the feature vector to preserve spatial information. In the recognition phase, a two-layer hierarchical classifier is used. For both layers, features are extracted and the codebook is built up. Then by applying the sparse coding, the features are encoded using the codebook. Finally, SVM is uses to classify the garment category and pose.

4. Features for Object Recognition

In **(Winkelbach , Molkenstruck and Wahl, 2006)** pairs of oriented points are used as features. 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 **(Papazov and Burschka, 2010)** the feature proposed in (Winkelbach , Molkenstruck and Wahl, 2006) is modified in order to reduce the number of hypotheses. Only the pairs of oriented points with a user defined distance are used. Since the point distance is constant, it is not used in the local descriptor as proposed in (Winkelbach , Molkenstruck and Wahl, 2006) and therefore the local descriptor is a three element vector.

In **(Rusu et al., 2008)** a 3D descriptor named Point Feature Histogram (PFH) is proposed. The proposed method assigns a histogram of particular values describing the local surface properties to each point in the input point cloud. For each pair of points in a *k*-neighborhood of a query point, a source point p_s and the target point p_t are defined, the source being the one having the smaller angle between the associated normal and the line connecting the points. The Darboux frame is defined with the origin in the source point and the axes *u*, *v* and *w* defined by

$$u = n_s \ , \qquad v = \frac{(p_t - p_s) \times u}{\left\| (p_t - p_s) \times u \right\|} \ , \qquad w = u \times v \ ,$$

where n_s and n_t are the surface normals in the points p_s and p_t. Four features are computed from the points p_s and p_t and their normals:

$$f_0 = v^T n_t$$

$$f_1 = \left\| p_t - p_s \right\|$$

$$f_2 = \frac{u^T (p_t - p_s)}{f_1}$$

$$f_3 = \operatorname{atan} \frac{w^T n_t}{u^T n_t}$$

A 4D histogram is created from the values f_0, f_1, f_2 and f_3 of all point pairs. The authors used 3 bins per feature therefore obtaining the total of 81 bins. After creating the histogram, each bin is normalized with the total number of point pairs.

In **(Rusu, Blodow and Beetz, 2009)**, a modification of the descriptor proposed in (Rusu et al., 2008) is described, which can be computed faster. The descriptor is named Fast Point Feature Histogram (FPFH). The descriptors are generated in a two stage procedure. In the first step, for each query point the features f_0, f_2 and f_3 of PFH are computed (feature f_1 of PFH is omitted) only between the query point and its neighbors. The histogram of these features is created, which is

named the Simplified Point Feature Histogram (SPFH). In the second step, *k* neighbors are determined for each point and the final histogram (FPFH) is computed by combining SPFHs of *k* neighbors.

In **(Rusu et al., 2010)** a global descriptor based on FPFH is proposed, which is named Viewpoint Feature Histogram (VPH). The new feature consisting of two parts: (i) a viewpoint direction component and (ii) a surface shape component comprised of an extended FPFH. The viewpoint component is computed by collecting a histogram of the angles that the viewpoint direction makes with each normal. The second component measures the angles used as features in PFH but now measured between the viewpoint direction at the central point and each of the normals on the surface.

A modification of VPH, which is more robust to occlusion and measurement noise, is proposed in **(Aldoma et al., 2011)**. This descriptor is named Clustered Viewpoint Feature Histogram (CVFH). The main idea is that VPH is not computed over the whole object, but over stable surfaces of the object. The object surface is clustered to stable surfaces by a region growing procedure, which starts by selecting a random object point, creating the initial cluster containing only that point and growing the cluster by adding neighboring points with similar surface normal directions. For each stable surface, a histogram similar to VFH is computed. The histogram also includes a shape distribution component representing the distribution of the square distances of all points of the surface to the surface centroid normalized by their maximum value. Due to the invariance of CVFH with respect to rotations along the view direction of the camera (roll), the object and viewpoint recognition is determined up to an unknown rotation. In order to provide 6DoF object pose, for each CFVH, the camera's roll histogram is computed. The surface normals at each point are projected onto a plane that is orthogonal to the vector given by the camera center and the centroid of the stable region used to compute CVFH. A histogram of the projected normals is created (roll histogram). The orientation of the object in the scene relative to the camera optical axis is estimated by determining the shift between the model roll histogram and the roll histogram of the object in the scene using Discrete Fourrier Transform.

In **(Aldoma et al., 2012a)**, a modification of CVFH is proposed, which is named Oriented, Unique and Repeatable CVFH (OUR-CVFH). The proposed OUR-CVFH descriptor is based on Semi-Global Unique Reference Frames (SGURF), which are computed for each cluster, where clusters are detected as proposed in (Aldoma et al., 2011). The SGURF assigned to a particular cluster is determined using the eigenvectors of the weighted scatter matrix of the points in that cluster. CVFH is modified by removing one component and adding the component related to the distribution of points in 8 octants of the SGURF.

Signature of Histograms of OrienTations (SHOT) descriptor **(Tombari, Salti and Di Stefano, 2010)** is inspired by SIFT. The spherical neighborhood of the keypoint is subdivided into an isotropic spherical grid that encompasses partitions along the radial, azimuth and elevation axes, as sketched in Fig. 1. For each cell of the grid, a histogram is created by accumulating points inside the cell into bins according to cosine of the angle between the normal at each point and the normal at the keypoint.

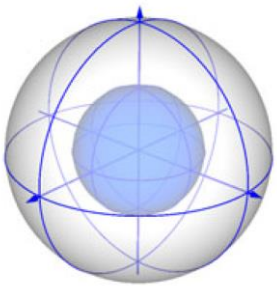

Fig. 1. Isotropic spherical grid used to form SHOT descriptor.

In order to cope with the boundary effects, the soft histogram approach is applied. For each point being accumulated into a specific local histogram bin, quadrilinear interpolation with its neighbors is performed, i.e. the neighboring bins in the local histogram and the bins having the same index in the local histograms corresponding to the neighboring cells of the grid. In particular, each entry is multiplied by a weight of $1 - d$ for each dimension, where d is the distance of the current entry from the central value of the bin. Along each dimension, d is measured in units of the histogram or grid spacing, i.e. it is normalized by the distance between two neighbor bins or cells. To achieve robustness to variations of the point density, the whole descriptor is normalized to sum up to 1.

In **(Shah, Bennamouna and Boussaid, 2015)**, a novel local surface description technique for automatic 3D object recognition in cluttered scenes is presented. The local surface description is based on a novel 3D keypoint detector which exploits the divergence of the vector field and detects highly repeatable or stable 3D keypoints. The vector field at each point (vertex) of a 3D surface is defined as the weighted average of the normals of its immediate neighboring triangles. The authors also propose a pruning step and a novel saliency measure to rank the keypoints and select the best points for the subsequent extraction of the novel local feature descriptor 3D-Vor. This descriptor, captures the vorticity at each point of the local surface to provide a highly discriminative surface representation. Extensive experimental testing on three popular datasets is performed and the results indicate that the proposed descriptor is highly descriptive and robust to resolution and noise. Based on the proposed keypoint detector and 3D-Vor descriptor, a fully automated 3D object recognition algorithm is also proposed. A major advantage of the proposed algorithm is that it requires only a single correct feature correspondence for object recognition. Extensive 3D object recognition experiments of complex scenes, in the presence of clutter and occlusion, on three publicly available datasets are performed. Results indicate that the proposed method outperforms existing methods and achieves recognition rates of at least 96%.

Another two poular features for object recognition in depth images are spin image **(Johnson and Hebert, 1999)** and NARF **(Steder et al., 2010)**.

References

Aldoma A, Blodow N, Gossow D, Gedikli S, Rusu RB, Vincze M and Bradski G (2011) CAD-Model Recognition and 6DOF Pose Estimation Using 3D Cues, *IEEE International Conference on Computer Vision (ICCV)*

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

Aldoma A, Tombari F, Prankl J, Richtsfeld A, Di Stefano L and Vincze M (2013) Multimodal cue integration through Hypotheses Verification for RGB-D object recognition and 6DoF pose estimation, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2104–2111

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)*

de Berg M, Cheong O, van Kreveld M and Overmars M (2010) Computational Geometry: Algorithms and Applications, *Springer-Verlag*, Berlin, Heidelberg.

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

Breiman L (2001) Random forests, *Machine learning*, 45(1), pp. 5–32

Choi C and Christensen HI (2013) 3D Pose Estimation of Daily Objects Using an RGB-D Camera, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3342–3349.

Comaniciu D (2002) Mean shift: A robust approach toward feature space analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619.

Doumanoglou A, Kouskouridas R, Malassiotis S and Kim TK (2015) 6D Object Detection and Next-Best-View Prediction in the Crowd, *arXiv:1512.07506*

Eitel A, Springenberg JT, Spinello L, Riedmiller M and Burgard W (2015) Multimodal Deep Learning for Robust RGB-D Object Recognition. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany.

Fantoni S, Castellani U and Fusiello A (2012) Accurate and automatic alignment of range surfaces, *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, pp. 73–80

Fischler MA and Bolles RC (1981) 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.

Girshick R (2015) Fast R-CNN, *IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448.

Girshick R, Donahue J, Darrell T and Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. *IEEE Computer Vision and Pattern Recognition*, pp. 580–587.

Guo Y, Bennamoun M, Sohel F, Lu M, and Wan J (2014) 3D Object Recognition in Cluttered Scenes with Local Surface Features: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2270–2287.

He K, Zhang X, Ren S and Sun J (2015) Deep Residual Learning for Image Recognition. ArxivOrg;7:171–180.

Hinterstoisser S, Lepetit V, Ilić S, Fua P and Navab N (2010) Dominant orientation templates for real-time detection of texture-less objects, *Computer Vision and Pattern Recognition (CVPR)*

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

Kim VG, Li W, Mitra NJ, Chaudhuri S, DiVerdi S and Funkhouser T (2013) Learning Part-based Templates from Large Collections of 3D Shapes, *ACM Transactions on Graphics (Proc. SIGGRAPH)*

Kouskouridas R, Tejani A, Doumanoglou A, Tang D and Kim TK (2016) Latent-Class Hough Forests for 6 DoF Object Pose Estimation, *arXiv:1602.01464*

Krizhevsky A, Sutskever I and Hinton GE (2012) ImageNet classification with deep convolutional neural networks. Annual Conference on Neural Information Processing Systems (NIPS)

Li Y, Chen CF and Allen PK (2014) Recognition of Deformable Object Category and Pose, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5558–5564

Lowe DG (2004) Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110.

Mueller CA, Pathak K and Birk A (2014) Object Shape Categorization in RGBD Images using Hierarchical Graph Constellation Models based on Unsupervisedly Learned Shape Parts described by a Set of Shape Specificity Levels, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Chicago, IL, USA, pp. 3053–3060.

Mueller CA, Ploeger P and Roscoe MS (2012) Towards Scalable 3D Object Shape Categorization, *Active Semantic Perception Workshop on IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Muja M and Lowe DG (2009) Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration, *International Conference on Computer Vision Theory and Applications (VISAPP)*

Nan L, Xie K and Sharf A (2012) A Search-Classify Approach for Cluttered Indoor Scene Understanding, *ACM SIGGRAPH Asia*

Narayanan V, Aine S and Likhachev M (2015) Improved Multi-Heuristic A* for Searching with Uncalibrated Heuristics, *Eighth Annual Symposium on Combinatorial Search*

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

Papazov C and Burschka D (2010) An Efficient RANSAC for 3D Object Recognition in Noisy and Occluded Scenes, *Asian Conference on Computer Vision (ACCV)*, Part I, pp. 135–148

Prankl J, Aldoma A, Svejda A and Vincze M (2015) RGB-D Object Modelling for Object Recognition and Tracking, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*

Rabbani T, van den Heuvel F and Vosselman G (2006) Segmentation of point clouds using smoothness constraint, Proceedings of the ISPRS Commission V Symposium 'Image Engineering and Vision Metrology', Dresden, Germany, pp. 248–253.

Razavian AS, Azizpour H, Sullivan J and Carlsson S (2014) CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. *IEEE Computer Vision and Pattern Recognition (CVPR)*, pp. 512–519.

Ren S, He K, Girshick R and Sun J. (2015) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal *Neural Information Processing Systems (NIPS)*.

Rock J, Gupta T, Thorsen J, Gwak JY, Shin D and Hoiem D (2015) Completing 3D Object Shape from One Depth Image, *IEEE Computer Vision and Pattern Recognition (CVPR)*

Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC and Fei-Fei L (2015) ImageNet Large Scale Visual Recognition Challenge, *International Journal of Computer Vision* (*IJCV)*.

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

Rusu RB, Bradski G, Thibaux R and Hsu J (2010) Fast 3D recognition and pose using the viewpoint feature histogram. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan

Rusu RB, Marton ZC, Blodow N and Beetz M (2008) Learning Informative Point Classes for the Acquisition of Object Model Maps, *International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Hanoi, Vietnam

Savarimuthu TR, Papon J, Buch AG, Aksoy EE, Mustafa W, Wörgötter F and Krüger (2015) An Online Vision System for Understanding Complex Assembly Tasks. *International Conference on Computer Vision Theory and Applications*, pp. 1–8.

Schulman JD, Lee AX, Ho J and Abbeel P (2013) Tracking Deformable Objects with Point Clouds, *IEEE International Conference on Robotics and Automation (ICRA)*

Schwarz M, Schulz H and Behnke S (2015) RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features. *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1329–1335.

Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R and LeCun Y (2014) OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks, *International Conference on Learning Representations (ICLR)*, pp. 1–16.

Shah SAA, Bennamouna M and Boussaid F (2015) A novel feature representation for automatic 3D object recognition in cluttered scenes, *Neurocomputing*

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.

Simonyan K and Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*, pp. 1–14.

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

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

Twardon L and Ritter H (2015) Interaction skills for a coat-check robot: identifying and handling the boundary components of clothes, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3682–3688

Winkelbach S, Molkenstruck S and Wahl FM (2006) Low-Cost Laser Range Scanner and Fast Surface Registration Approach, *Proceedings of Pattern Recognition: 28th DAGM Symposium*, pp. 718–728.

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