Hierarchical Object Geometric Categorization and Appearance Classification for Mobile Manipulation

Zoltan-Csaba Marton, Dejan Pangercic, Radu Bogdan Rusu, Andreas Holzbach, Michael Beetz

Abstract—In this paper we present a comprehensive object categorization and classification system, of great importance for mobile manipulation applications in indoor environments. In detail, we tackle the problem of recognizing everyday objects that are useful for a personal robotic assistant in fulfilling its tasks, using a hierarchical multi-modal 3D-2D processing and classification system. The acquired 3D data is used to estimate geometric labels (plane, cylinder, edge, rim, sphere) at each voxel cell using the Radius-based Surface Descriptor (RSD). Then, we propose the use of a Global RSD feature (GRSD) to categorize point clusters that are geometrically identical into one of the object categories (bowl, box, plate, mug, pan, etc.). Once a geometric category and a 3-DOF position is obtained for each object cluster, we extract the region of interest in the camera image and compute a 2D SURF feature vector for each patch. We obtain the orientation around the up-right axis, and the exact object instance from the appearance. The resultant system provides a hierarchical categorization of objects into basic classes from their geometry and identifies objects and their poses based on their appearance, with near real-time performance. We validate our approach on an extensive database of objects that we acquired using real sensing devices, and show classification of 98.63% for 3D object category recognition and close to 98% for visual feature classification.

I. INTRODUCTION

The use of accurate object models enables personal robotic agents doing everyday manipulation in indoor environments to perform their tasks more reliably, flexibly, and efficiently. As these robots get more sophisticated manipulation capabilities, they require more expressive and comprehensive object models, beyond their position and appearance, including information about their precise shape or additional semantic information that is useful with respect to the robot tasks.

Though the set of objects of daily use that a personal robot could encounter in its tasks is unlimited, there are certain regularities that can be exploited with respect to the object shapes, textures, or uses. Therefore, in some sense, the perception system can specialize itself to a specific set of objects that are usually present in the world, while at the same time retain a certain degree of flexibility with respect to the incorporation of novel objects in its internal models. For example, a new flavor of iced tea should be recognized as an instance of a tea box from its geometry (shape) or use, even though the robot has never seen it before, and therefore could not understand the semantics of the object from its visual appearance.

This paper proposes a comprehensive multi-modal perception system comprised of hierarchical object geometric categorization [1] and appearance classification for personal robots manipulating in indoor environments. Our focus is on robustly identifying objects of interest supported by planar surfaces such as tables that can be manipulated by the robot. Since the goal is to have personal robotic assistants working and operating in the same environments for months or years without interruption, a clear need to learn accurate models of the objects that are to be manipulated again and again arises. This requires the creation of efficient object classifiers that can discriminate but also generalize between the objects present in the world over time.

We motivate the additional use of geometry along appearance (i.e., texture) in our approach for three equally important problems. On one hand, our experience is that texture alone can lead to false positive matches such as the example shown in Figure 2. Given a template picture of some object that we need to find out and match in a scene, say a beer bottle, the system can mistakenly identify it on parts of the world which contain similar pictures of that object, a mug in this case. Simply put, without geometry, texture can be deceiving. Though the case presented here is slightly exaggerated, the point still stands.

On the other hand, there are several cases where for a general purpose problem such as the one of identifying objects in indoor environments, the space of possible solutions becomes so big that the model that is to be learned has to support thousands of different classes or more. This leads to increased training or update times for the model, but can also result in decreased classification accuracies. By considering geometry as well, we simplify the problem...
and create a more logical categorization and classification framework, by creating separate models for groups of objects that have the same or similar geometry (e.g., cereal boxes, mugs, etc). This results in an overall optimization (in terms of the computational complexity) for supervised learning mechanisms.

Finally, there are objects which have no texture, in the sense that they are uniformly colored and do not exhibit texture properties that can be identified by regular keypoint descriptors in computer vision.

Without appearance however, it would be very hard to discover the orientation of an object, especially for objects with a near-symmetrical geometry. Though it might be argued that the orientation of a symmetrical object is not very important, in some cases (for example if the iced tea should be poured out of the tetrapak) this subtle disambiguation is mandatory for mobile manipulation. Additionally, we might want to discern between objects which have the same geometry but a different appearance and purpose, like for example a ketchup bottle versus a mustard bottle.

In detail, the approach presented herein combines our previously proposed 3D Radius-based Surface Descriptor (RSD) [2] and Global Point Feature Histograms (GFPFH) [1] with the 2D Speeded Up Robust Features (SURF) features [3] to create a hierarchical geometrical-appearance supervised classification scheme. Using the rough geometrical shape of the object we cluster objects in distinct categories and use that to influence the problem of classification and modelling based on appearance. The key contributions of the research reported in this paper thus include the following:

- the use of a fast 3D feature computation method to annotate surfaces and object hypotheses with geometric labels;
- the proposal of a powerful global descriptor (GRSD) that can generalize over objects with geometric similarities to limit the possibilities of which object instance could a cluster be;
- the synergy of depth and visual processing and learning techniques in a multi-modal hierarchical architecture for the problem of robustly identifying discriminative object classes and their rough orientations with respect to the robot camera view.

The structure of this paper is organized as follows. Related work is described in Section II. Next, we give a brief description of our system architecture in Section III. The acquisition of a database of models is presented in Section IV. We present the geometric and appearance processing pipelines in Section V followed by a discussion of experimental results in Section VI. We conclude in Section VII.

II. RELATED WORK

The are two principal mainstream lines in the area of the object recognition related research: one aiming at recognition of objects in camera images, and one using 3D depth data acquired through range scanning devices. Combining both of them leads to a hybrid approach and our work falls into this category. Depending on the type of perception data, various different 2D (e.g. [4]) and 3D (e.g. [5]) distinctive local features have been developed. Taken individually however, these are still insufficient to solve the full object recognition problem as both are prone to failure in situation where texture-less objects are present or depth data is too noisy or ambiguous. That is why different research initiatives have decided to combine sets of local features and cluster them together using different metrics (kernels), in order to be able to infer the global identifiers for objects.

In [6] objects are recognized and reconstructed using image databases. The overall approach is based on finding the consistent matches in the subsets of all images. Following a structure and motion of each object is solved using a Sparse Bundle Adjustment algorithm. Fergus et al. [7] is proposing an unsupervised scale-invariant learning scheme, in order to detect objects on a wide range of images. Objects therein are modeled as flexible constellations of parts using a probabilistic representation for all significant aspects of the object. The work exploits the expectation-maximization algorithm in a maximum-likelihood setting. The method in [8] estimates 6-DOF object poses in cluttered scenes by matching local descriptors to stored models. Since the objects present in household environments are most often texture-less, our approach constitutes an important advantage over the above proposed research initiatives, which fail to work in the absence of good textured objects.

The work in [9] uses an iterative matching procedure to merge similar models in an unsupervised manner, while a spectral clustering of similarity matrix is used to terminate the merging convergence. However, it is unclear how well the proposed algorithm would i) generalize to unknown, novel objects and ii) infer semantic properties of those. Lai et al. [10] perform outdoor laser scans classification combining manual labeling and data downloaded from the Internet in an effort coined domain adaption. While their presented recall curves outperform others, the number of objects is relatively low and household objects are less distinct. In [11], the authors investigate the extraction of GOODSAAC point features and object recognition from range images, that are in turn computed from point cloud datasets. These object models are, as in our case, created from real 3D data but processed using the work in [9].

The combination of depth information with camera images is addressed in [12]. The authors calculate depth information for each pixel in the scene by applying laser-line triangulation with a rotating vertical laser and a camera. To obtain

![Fig. 2](https://via.placeholder.com/150)

Fig. 2. An example of a good model match using SURF features extracted from 2D images (left), where a beer bottle template is successfully identified in an image. However, zooming out from the image, we observe that the bottle of beer is in fact another picture stitched to a completely different 3D object (in this case a mug). The semantics of the objects in this case are thus completely wrong.
III. SYSTEM ARCHITECTURE

The platform used for the acquisition of models is briefly described in Figure 1, and consists of a B21 mobile base with Amtec Powercube 6-DOF arms and sensors such as a SICK LMS400 laser device and Basler Scout stereo cameras.\(^1\) To facilitate the assembly of a large database of object models, we have created a rotating table using a DP PTU47 pan-tilt unit that is controlled by the robot over the network. Objects placed on this rotating table are scanned and geometric and appearance models are created for them automatically using supervised learning techniques. The resultant database of object models is then used to categorize and classify objects found in natural table setting scenes while performing manipulation tasks. Our approach is built as part of the Robot Operating System (ROS)\(^2\) open source initiative, and makes use of modular and robust components that can be reused on other robotic systems different than ours.

The general architecture of our framework together with the geometric and processing pipelines is illustrated in Figure 3. For a better understanding of the underlying blocks and the connections between them, the overall system is divided into three major components, namely: acquisition, geometric processing, and appearance processing. The hardware devices that are used by our system are shown in orange boxes, while the processing steps are depicted with rounded blue boxes, and the outputs are represented as yellow dotted boxes.

The acquisition component is responsible for acquiring the 3D depth data and the 2D RGB images that are used by the processing components. A task executive running on the robot controls the four hardware devices used, triggers the assembly of 3D point cloud models \(\mathcal{P}\), and takes image snapshots when needed. The process is described in greater detail in Section IV.

The point cloud models \(\mathcal{P}\) are then processed through a series of geometric reasoning steps including the segmentation of the supporting table planes, the clustering of object candidates into independent data structures, gross outlier removal, and normal estimation [16]. Using the RSD descriptors [2], we describe the underlying surface geometry at every surface unit as plane, cylinder, edge, rim and sphere, as detailed in Section V-A. In a next step, for each cluster, a GRSD descriptor is estimated using the previously acquired voxel labels, and an SVM model is used to categorize clusters into following categories: 1) bowl, 2-3) medium and small box, 4-6) tall, short and small cylinder, 7-8) big and small flat boxes, 9) mug, 10) pan, 11) plate and 12) tetrapak. This completes the first classification layer \(L_1\) in our geometric processing scheme (see Figure 8). The output of the geometric processing component is a set of annotated object clusters, categorized into classes which give hints with regards to their geometric structure.

For each object cluster obtained, a Region Of Interest (ROI) segmentation step is applied on the acquired camera images, in order to obtain a smaller, cropped image that represents that particular object. Note that for this step to be successful, we employ a precise camera to laser calibration step offline based on [17]. Then, for each image patch representing the object of interest we estimate SURF features for points of interest in the image, resulting in a vector of \(n\)
features. Based on the results obtained from the geometric global cluster annotation, the SURF feature vector is tested against a subset of images in the database that could possibly represent appearance models for it. This constitutes the second classification layer \( (L_2) \).

IV. OBJECT DATABASE ACQUISITION

Our database of 3D objects, available at http://semantic-3d.cs.tum.edu, was obtained using the hardware devices mentioned in Section I.

The set of objects encompasses the ones commonly used in a typical household environment (mugs, utensils, books, etc) and is envisioned for a larger expansion in the future. In a pursue to account for a wide variety of view angles, we rotated the objects on the rotating table with a given angle-step \( (30^\circ \text{ in the preliminary version}) \) and acquired partial snapshots from a human-eye perspective, i.e. the ones that the best approximate the robot’s view point during its working cycle. We consider this to be an important point as opposed to similar initiatives (e.g., [14]) where the datasets are i) acquired using high-precision but non-affordable, fixed sensors and thus ii) not useful for robotics applications such as ours.

A. Datasets Qualification

The images in the dataset have been acquired using Basler Scout scA1390 stereo cameras at a resolution of 1390x1038 pixels. The 3D depth data was recorded using a SICK LMS400 range scanner with 0.5\(^\circ\) angular resolution, resulting in point clouds with roughly 200-1000 points per object after the gross statistical removal procedure. The range scanner was tilted with 30 radians/s during one scanning cycle.

V. GEOMETRIC AND APPEARANCE PROCESSING

The input of the geometric processing step is a set of partial 3D point cloud models \( \mathcal{P} \), acquired from the tilting LMS400 laser sensor installed on the Powercube arm.

The robot then proceeds at extracting supporting horizontal planes from \( \mathcal{P} \), with our assumption being that the robot is already close or in the vicinity of a table. If this is not already the case, in a more general sense, we make use of global planar segmentation techniques such as the ones we previously proposed in [16] on the entire dataset \( \mathcal{P} \) thus makes little sense, as the mean density of \( \mathcal{P} \) will affect the point clusters located at large distances from \( \mathbf{v} \). Instead, we apply a statistical gross outlier removal procedure on each separate point cluster \( \mathcal{O}_i \) in parallel, thus decoupling the relationship between the point density of \( \mathcal{O}_i \) and \( \mathbf{v} \), or in other words enabling the filtering step to be viewpoint and density independent.

Additional smoothing is performed by displacing the points to lie on the regression plane of their local neighborhood, as this approach is much faster than a full-blown surface estimation (by for example using MLS), yet it reduces the “thickness” of scanned surfaces considerably.

A. Voxel Annotation

To speed up the local surface classification, we label the surface units needed for the global classification (see next subsection) directly, instead of taking the dominant point-based label for each voxel. This reduces the complexity proportionally to the average number of points in a voxel. Also, the neighborhood can be directly constructed using the points in the current and surrounding voxels.

Once the neighborhoods is constructed, we compute the RSD features, i.e. the radius of the highest and lowest curvature in the local neighborhood, as described in [2]. As a short overview, from the distribution of normal angles by distance we take the minimum and maximum normal variations by distance, and solve the equation:

\[
d(\alpha) = \sqrt{2r_\text{max}^2 + O(\alpha^5)}
\]

which greatly reduces the problem of finding \( r_{\text{min}} \) and \( r_{\text{max}} \).

Since these values have physical meaning, we can categorize surfaces using simple, intuitive rules, into: planes (large \( r_{\text{min}} \)), cylinders (medium \( r_{\text{min}} \), large \( r_{\text{max}} \)), edges (small \( r_{\text{min}} \) and \( r_{\text{max}} \)), rims (small \( r_{\text{min}} \), medium to large \( r_{\text{max}} \)), and spheres (similar \( r_{\text{min}} \) and \( r_{\text{max}} \)). Figure 4 and the top right part of Figure 1 show point clouds annotated with the surface type.

B. Object Categorization

Once all voxels are annotated locally using a geometric class, our processing pipeline constructs a global feature space that can produce a unique signature for each object cluster. This space is based on the idea that, for a set of labeled voxels, a global feature can be constructed by observing the relationships between all these local labels (and the encapsulated free space). Since the labels represent geometric classes obtained from the classification of RSD descriptors, we call this new feature the Global Radius-based Surface Descriptor (GRSD).
The computation of GRSD is similar to GFPFH [1], with the exception that we sum up the individual $H_{f_{ij}}$ histograms instead of computing their distribution, to further reduce computational complexity. This way, the complete processing of a cluster (correcting, estimating normals, computing the voxelized RSD values, labeling voxels and constructing the GRSD) takes between 0.2 and 0.5 seconds (depending on object size) on a single core (using voxel size of 1.5cm as in the presented examples).

![Fig. 4. Example of RSD classes and GRSD histograms for a big flat box (i.e. book, upper row) and a mug (bottom row). The histogram bin values are scaled between -1 and 1 according to the training data, and the colors represent the following local surfaces: red - sharp edge (or noise), yellow - plane, green - cylinder, light blue - sphere (not present), and dark blue - rim (i.e. boundary, transition between surfaces). For the accurate color information we kindly direct the reader to the electronic version of the paper.](image)

Figure 4 shows two sets of histograms of different objects generated by the GRSD estimation. We then selected 12 overall categories, namely: bowl, box-medium, box-small, cylinder-big, cylinder_short, cylinder_small, flat_big_box, flat_small_box, mug, pan, plate and tetrapak. Although the classes were picked by hand, they match general categories of geometric objects and are intuitive with respect to the underlying geometry. Therefore this representation is consistently useful and applicable for grasping, because each object can be grasped similarly to the objects in its class. An SVM model is then trained using the global histograms and the predefined classes.

C. Visual Feature Classification

The visual feature detection is carried out only on the region of interest (ROI) in the image, in order to avoid false positive matches with the background. To obtain the ROI, we use the 3D boundary points [16] detected in the point cloud cluster, which we then project onto the image. The convex hull of these projected points is calculated and used as boundary to cut off the background.

To obtain complete 360° models for each object, we decided to divide the entire interval at every 30°, therefore resulting in 12 datasets per object. For each image, we extracted the ROIs representing the objects of interest and computed a vector of SURF features. The next step is to quantize the feature vector, that is, cluster it into a Bag of Features using standard K-Means techniques. This step is needed in order to obtain the constant bag size necessary for an SVM classifier and can be at best thought of as a histogram with the number of features in each cluster represented as tabulated frequencies. The classification was again performed using an SVM classifier with an RBF Laplacian kernel, and the model is used to determine object orientation for test objects. Figure 5 presents two simple examples of matching SURF features for a scene containing an iced tea box with the same orientation (left), and different orientation (right).

VI. DISCUSSIONS AND EXPERIMENTAL RESULTS

To validate our proposed framework, we have performed several experiments on global (object), as well as appearance-based data, using the processing pipelines presented in Section V. Overall, we have gathered over 300 datasets, i.e. partial views of 29 objects from the database (shown in Figure 6).

![Fig. 5. Example of matching SURF features for an iced tea object.](image)

![Fig. 6. A subset of the collection of objects used in our experiments.](image)

To evaluate the overall performance of our approach we carried out three types of object recognition test: i) test with the training dataset, ii) preliminary test with 13 of the objects in regular table settings, and iii) test with 6 previously unseen objects on the table. While the best accuracy (98.63%, see bottom row of Figure 7) was obtained in the first case, the accuracies for the remaining tests (76.92% and 50.00% respectively) were still encouraging. We are, however, striving to perform more tests to obtain a representative sampling for evaluation, as very few examples of objects (1-2 in most cases) were tested so far.
From an implementation point of view, our approach makes use of the efficient Radius-based Surface Descriptor to label individual voxels in 3D data. SVM machine learning algorithm to classify object clusters using GRSD and appearance models using SURF. The preliminary results indicates that the architecture is promising.

As future work we plan to extend our object database and perform additional tests with the proposed approach by extending the feature spaces used, as well as create and fit suitable 3D models for grasping applications. Though at a preliminary stage, we are releasing both the 3D database and our code as part of the ROS initiative.

Acknowledgments This work was supported by the CoTeSys (Cognition for Technical Systems) cluster of excellence at the Technische Universität München.

REFERENCES


Fig. 8. Representation of our hierarchical categorization (Layer 1) and classification (Layer 2) approach. Angle range used: [-180 to +180]°.

