Learning Semantic Interaction among Graspable Objects

Swagatika Panda, A.H. Abdul Hafez, and C.V. Jawahar

International Institute of Information Technology, Hyderabad, India

Abstract. In this work, we aim at understanding semantic interaction among graspable objects in both direct and indirect physical contact for robotic manipulation tasks. Given an object of interest, its support relationship with other graspable objects is inferred hierarchically. The support relationship is used to predict the "support order" or the order in which the surrounding objects need to be removed in order to manipulate the target object. We believe, this can extend the scope of robotic manipulation tasks to typical clutter involving physical contact, overlap and objects of generic shapes and sizes. We have created an RGBD dataset consisting of various objects present in clutter using Kinect. We conducted our experimentation and analysed the performance of our work on the images from the same dataset.

Keywords: Robotic Vision, Support Relation, Support Order, Semantic Interaction, RGBD.

1 Introduction

Understanding semantic interaction among the objects plays an important role in various robotic manipulation tasks in clutter. However, most of the robotic grasping and manipulation tasks remain confined to isolated objects, and often lying on planar surfaces [2, 8]. Recently, there has been a few attempts to infer pairwise support relationship among objects in clutter. Rosman *et al.* [9] predict three types of support relations: "on", "adjacent to", "both adjacent and on" using Kernel SVM. However, they assume that background is already segmented from the data. Sjöö and Jensfelt [11] infer four types of relations among each pair of object, viz. casual support, support force, protection and constraint using logistic regression classifier. But they work on simulated environment and hence remain limited to the inherent imperfections of simulation. Silberman *et al.* [10] perform inter-class support inference among regions of four major structure classes, viz. floor, wall, furniture and props using linear programming. They work in cluttered indoor settings. However, their work does not exploit the spatial relationship among different objects that overlap onto each other. Unlike the previous works, the aim of our work is to perform support inference in clutter involving both graspable and non-graspable objects, infer both direct and indirect support relations and then predict the "support order" in which the supported objects should be removed so that the object of interest can be manipulated without causing damage to the environment.

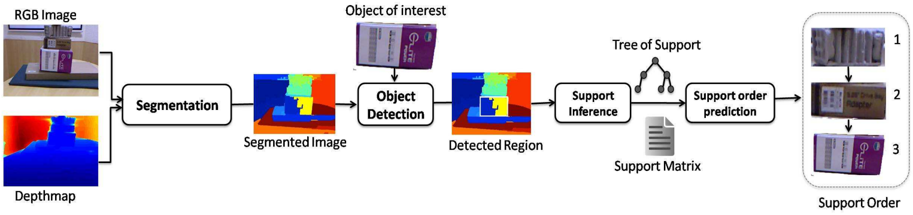


Fig. 1. Overview of our framework: Given the RGB and depth images and target object, our algorithm predicts the Support Matrix and the Support Order of the objects.

In our proposed framework, we learn the semantic interaction among the objects in clutter in three phases. At first, the regions corresponding to the graspable objects are separated out from the background entities in a cluttered indoor scene. Then, a target object is detected in clutter and its support relationship w.r.t. other objects is inferred hierarchically. Four kinds of support relations, i.e., "support from below", "support from side", "containment" or "none" are inferred. A "tree of support" is simultaneously built during support inference to encode the support relationship. Finally, the tree of support is traversed to predict the "support order". During tree traversal, special scenarios are identified and addressed so that minimal damage occurs when the objects are removed according to the predicted support order. We use different geometric features using both RGB and depth information to effectively capture the physical properties of objects. We validate the inferred support relationship and support order on various images from our RGBD dataset captured using Kinect in different indoor settings. In contrast to our previous work [7], where support relationship among every pair of objects present in the scene is obtained prior to detecting the target object and performing support order prediction, in the proposed work, support inference for all pairs of objects is avoided by posing it as an object-centric task. Here, support inference is performed hierarchically w.r.t. the object of interest. Using this approach, the number of comparisons required is reduced from $O(n^2)$ to $O(nlog(n))$. We believe, the predicted support order can be very useful in many applications such as grasping and object manipulation tasks in clutter involving overlaps and support by multiple objects. The overall framework of our work is explained through the block diagram shown in Fig. 1.

2 Support Inference

2.1 Segmentation & Object Detection

The image is segmented into different regions and target object is detected. Hierarchical segmentation method of Hoiem *et al.* [4] is used to segment the scene. This method uses superpixels segmented using Arbelaez's method [1]. Both 2D and 3D features of images are used for segmentation. The segmented regions are provided as input to the object detection and support inference modules. In the object detection module, SIFT feature matching [3, 6, 12] between the template image of the object of interest and the input image is performed. The outliers are discarded by applying RANSAC. The segmented regions corresponding to the matched points of the input image are merged into one region and chosen as the region corresponding to object of interest O, i.e., the object to be grasped. This approach ensures that the entire object region is chosen for grasping. Given the region corresponding to the target object, support relationship among different regions is predicted hierarchically.

2.2 Learning & Inference

A cascade of classifiers is applied to the segmented regions and the target object for support inference. The learning methods, classifiers and approach for inference is discussed in detail below.

Structure Class Classification An indoor environment typically consists of entities with distinct structural properties such as floor, walls as well as entities with highly diverse structural properties such as furniture. Given a cluttered scene, it is important to first determine which objects are graspable and segregate them from other non-graspable entities. A logistic regression classifier is trained to predict the structure class (floor/wall/furniture/graspable objects) [10] of each region. The regions predicted as graspable objects are selected for support inference while other regions are discarded as background. By discarding regions corresponding to walls, floor and furniture, unnecessary comparisons between all the regions in the image are avoided. In our experimentation, logistic regression is performed using stochastic gradient descent algorithm over the normalized features. The values of learning rate, batch size and maximum number of updates are empirically set to 10^{-5}, 100 and 7000, respectively.

Support Class Classification In clutter where objects overlap on one another, an object supports multiple other objects. In order to access any object O without causing damage to the environment, all objects that O supports must be identified and removed. Our goal is to predict these supported objects. Pairwise support relationship among these objects is inferred using a 3-layer feed-forward neural network based support classifier. We perform support inference hierarchically instead of comparing all regions with each other for efficiency. Given a pair of regions (A, B), the support class classifier predicts if B supports A "from below", "from side", "contains" it or "not related" to it.

The support features for the regions corresponding to only the graspable objects are extracted and normalized for training the neural network. Sigmoidal activation function is used for the four output units to limit all the outputs to a fixed range([0 1]) so that the outputs can be interpreted as probabilities. The number of hidden nodes is kept as one tenth of the size of the training data to avoid over-fitting. Stochastic gradient descent algorithm is used to minimize the cross-entropy loss function which is appropriate for dealing with probabilities. The most probable support class is assigned to the pair of regions given as input. In our experimentation, the values of number of hidden nodes, learning rate, batch size and maximum number of updates are empirically set to 50, 10^{-4}, 100 and 10,000, respectively.

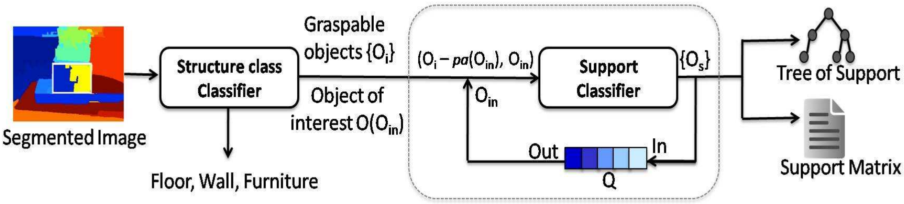


Fig. 2. Support Inference Module: The graspable object regions are filtered using structure class classifier. Support inference is performed hierarchically. tree of support and Support Matrix are generated as final output.

Hierarchical Support Inference An object can support multiple objects both directly and indirectly through other objects it directly supports. To ensure minimal damage during manipulation, support inference is performed hierarchically beginning with the target object.

Suppose, $S = \{O_i | i = 1, \ldots, n\}$ is the set of n graspable objects/regions in the scene and $O \in S$ is the target object. A tree of support T is built with the target object in the root to encode the predicted support relationship. (The details of traversal of T are discussed in Section 3). Instead of inferring for all possible pairs of objects which will require $n(n-1)/2$ comparisons, support relations of all the objects that O directly supports are inferred at first. Every object is paired with O and support relations of each of the pairs $\{(O_i, O) | O_i \in S - \{O\}\}$ are inferred from the support classifier. The objects $S_s = \{O_s\}$ predicted as objects supported by O are stored in a FIFO queue Q. They are also inserted into the tree T as child nodes of O. The object pulled from the queue Q is again fed to the support classifier to predict the objects that it supports in turn. All other regions except the supporting regions(s) or parent(s) $pa(O_s)$ and grandparents $pa(pa(O_s))$ of O_s, are paired with O_s for the prediction. Note that special care is taken to discard all the parents and grandparents of the object O_s in order to avoid loops which may lead to damage in practical scenario. The support inference is performed hierarchically until all the outcomes of the classifier are negative, i.e., all the support relationships are predicted as “not related”and consequently the queue Q is empty.

The advantage of performing support inference hierarchically instead of inferring for all pairs of objects is that this approach significantly reduces the number of comparisons required from $O(n^2)$ to $O(nlogn)$. Fig. 2 illustrates the process of support inference.

2.3 Features

For structure class classification, features proposed by Silberman *et al.* [10] are used. The features include SIFT features, histograms of surface normals, 2D and 3D bounding box dimensions, color histograms and relative depth.

For support classifiers, various features that capture the pairwise relationship of regions are used. Support feature $f(A, B)$ determine if region A is supported by B, but not the reverse. Hence, these features are not symmetric in nature. We adapt support features from [10]. The features include SIFT features of both regions to capture their individual characteristics, location features that capture absolute 3D positions of the regions and geometric features that capture different geometric relations between the two regions. In addition to this, we introduce five more new geometric features for improvement in encoding the pairwise relationships briefly discussed below. (i) **Proximity** between two objects $f_p(A, B)$ is measured as the ratio of distance in 3D between the centroids of the two objects and the sum of radii of the spheres circumscribing the two regions. The radius of the circumscribing sphere is approximated as the distance between the centroid and the point farthest from the centroid. (ii) The amount of overlap $f_{br}(A, B)$ is measured by **Boundary Ratio**, i.e., the ratio of the length of the boundary of the supported object A in contact with B with the perimeter of A.(iii) **Depth Boundary** $f_d(A, B)$ is used to differentiate visual occlusion and actual contact. It is the average distance between the two objects from the contact boundary. (iv) **Containment** $f_c(A, B)$ is the percentage of volume of the supported object A lying within the 3D convex hull of B. This feature determines if an object is on another object or inside it. (v) A stable object has higher probability of supporting an unstable object. If the gravity line of an object is in alignment with the baseline, it is considered as stable. **Relative stability** $f_s(A, B)$ is defined as 1 if supporting object is stable and supported object is unstable, -1 for the reverse and 0 if both are stable or unstable.

3 Support Order Prediction

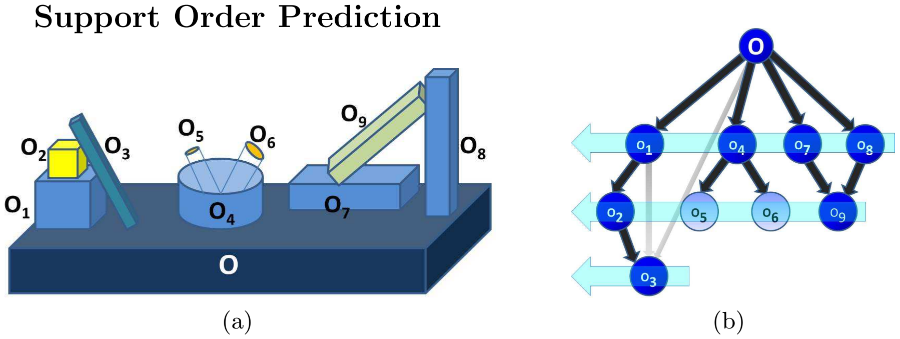


Fig. 3. Support Order Prediction. (a)$\{O, O_1, O_2, O_3\}$: support in multiple hierarchy; $\{O_4, O_5, O_6\}$: containment, $\{O_7, O_8, O_9\}$: simultaneous support by multiple objects. (b)Reverse Level Order Traversal of Tree of Support. Edges in gray: Pruned edges. Nodes in light blue: skipped nodes.

The tree of support T is traversed using reverse level order traversal to predict the support order. The objects present at the leaf nodes are the ones not supporting any other object. So they are picked up first and then the upper layer is traversed and the process repeats until we reach the root node, i.e., the target object. Various special cases are identified and handled during tree traversal to ensure minimal damage during manipulation. In case of support by multiple

Fig. 4. RGBD dataset for Support Inference: the dataset consists of images with different types of support, heavy clutter and occlusion.

hierarchy, the child node corresponding to the supported object is connected to multiple parent nodes from different layers. It is not feasible to retain all edges connecting to the child node. Retaining any of the edges in the upper layer(s) implies that the object corresponding to the child node will be searched even after its removal. On the other hand, pruning the edges in the lower layer(s) will increase the chance of damage since the presence of the supported object will be ignored while picking the parent node(s) in the lower layer(s). Therefore, the edge(s) between the child node and the parent node(s) at the lowest layer are retained while pruning off edges connected to parent node(s) in the upper layer(s). If any object contains other objects inside it instead of merely supporting it, then it is grasped directly without the need to remove the contained objects. So prior to retrieving any node during traversal, if the support type for a node is found to be "containment", then, this node is not retrieved. Fig. 3(b) graphically demonstrates the tree traversal and support order prediction for objects shown in Fig. 3(a). We traverse from the leaf nodes towards the root node. The support order is predicted as $O_3 \rightarrow O_9 \rightarrow O_2 \rightarrow O_8 \rightarrow O_7 \rightarrow O_4 \rightarrow O_1 \rightarrow O$.

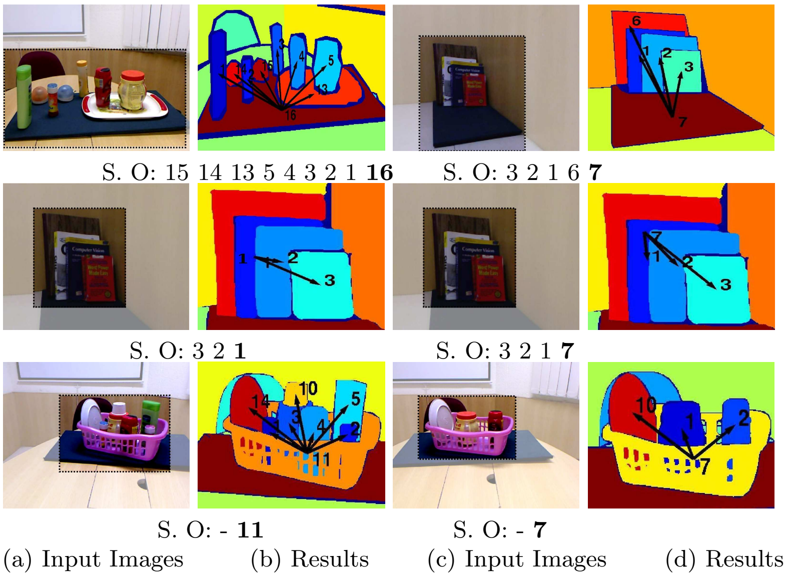
4 Experiments and Results

We have collected a dataset consisting of 50 images, point clouds and depthmaps with different levels of clutter using Kinect and have manually created annotation for each image, and ground truth support relationship for each pair of regions in all images (Refer Fig.4). The raw depthmaps are smoothened using an adaptation of colorization method by Levin *et al.* [5]. A 5-stage hierarchical segmentation approach proposed by Arbelaez *et al.* [1] is used for segmenting the images.

Table 1. Accuracy: Structure class & Support Inference

Inference	Structure class Inference		Support Class Inference	
Type	Training	Testing	Training	Testing
Ground Truth Regions	100	97.02	73.42	64.72
Segmented Regions	97.79	83.88	53.00	49.17

The output of the structure classifier affects support inference, since only the regions predicted as graspable objects are given as input to the support classifier. The support relationship of the regions missed by structure classifier can not be established. And, if a region is falsely classified as graspable object, we may end up inferring an infeasible support relation. Hence, it is important



S. O: 15 14 13 5 4 3 2 1 **16**   S. O: 3 2 1 6 **7**

S. O: 3 2 **1**   S. O: 3 2 1 **7**

S. O: - **11**   S. O: - **7**

(a) Input Images (b) Results (c) Input Images (d) Results

Fig. 5. Results of inference: The highlighted section in input images in column (a) & (c) are zoomed in columns(b) & (d) respectively for better view. The arrows point from target object to objects directly and indirectly supported by it. Support Order (S.O.) for each result is shown below the respective images.

to achieve high accuracy in structure class prediction. The accuracy of structure class classification is shown in Table 1. High accuracy in structure class inference can be due to similar background entities in our dataset.

The result of support inference is shown in Fig. 5. The predicted support order corresponding to the images are shown below each image. The results in row 1, 2 and 3 depict support from below, support from side and containment respectively. The results in row 1 show support relation from below to multiple objects. In row 2, we can observe the hierarchical support relationship. Book 7 supports book 1 directly and book 1 supports books 2 and 3. Therefore, book 7 also indirectly supports books 2 and 3. The results in row 3 show containment. All the objects contained in the basket are shown as supported by the basket. But due to containment, they need not be removed in order to remove the basket, as evident in the support order shown for images in row 3.

The accuracy of support class classification is given in Table 1. We achieve 64.72% accuracy on the ground truth regions. We observe that, support inference fails in a few situations. Often in frontal view, when the entire surface area of the supporting object is not visible and the contact to the supporting surface is not visible, the support relation is not inferred. In a few cases, due to error in structure class prediction, some valid graspable object regions are missed whereas some furniture regions are misclassified as graspable objects causing incorrect support inference. Sometimes, different planes of one object are segmented into different regions giving multiple regions for one object. In these cases also, infeasible support relations are inferred. Inaccurate segmentation errors also affect the support inference and thus support order prediction. As evident in Table 1, the accuracy of support class classification using segmented regions is 49.17%, which is significantly lower than compared to ground truth regions. This leaves lot of scope for improvement in this direction in future.

5 Conclusions

In this paper, we learned semantic interaction among objects in clutter by inferring support relationships and using them to predict the support order in which surrounding objects should be removed to access the target object. Our work extends the scope of semantic interaction to complex situations involving overlap, physical contact and objects of varied shape and size. We created a dataset consisting of different objects used in household and office environment and performed our experimentation on the same. In future, we plan to work on improving the performance of segmentation on RGBD data so that support relations are learned more accurately.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. Pattern Analysis and Machine Intelligence, IEEE Transactions on 33(5), 898–916 (2011)
2. Dogar, M., Hsiao, K., Ciocarlie, M., Srinivasa, S.: Physics-based grasp planning through clutter. In: RSS VIII (July 2012)
3. Harris, C., Stephens, M.: A combined corner and edge detector. In: Alvey vision conference. Manchester, UK (1988)
4. Hoiem, D., Efros, A.A., Hebert, M.: Recovering occlusion boundaries from an image. International Journal of Computer Vision 91(3), 328–346 (2011)
5. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. ACM Transactions on Graphics (TOG) 23(3), 689–694 (2004)
6. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision 60(2), 91–110 (2004)
7. Panda, S., Abdul Hafez, A.H., Jawahar, C.V.: Learning support order for manipulation in clutter. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2013)
8. Rasolzadeh, B., Björkman, M., Huebner, K., Kragic, D.: An active vision system for detecting, fixating and manipulating objects in the real world. The International Journal of Robotics Research 29(2-3), 133–154 (2010)
9. Rosman, B., Ramamoorthy, S.: Learning spatial relationships between objects. The International Journal of Robotics Research 30(11), 1328–1342 (2011)
10. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV 2012, pp. 746–760. Springer (2012)
11. Sjoo, K., Jensfelt, P.: Learning spatial relations from functional simulation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2011. pp. 1513–1519. IEEE (2011)
12. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/` (2008)