2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Daejeon Convention Center
October 9-14, 2016, Daejeon, Korea

RGB-D Multi-View Object Detection with Object Proposals and Shape Context

Georgios Georgakis and Md. Alimoor Reza and Jana Košecka

***Abstract*— We propose a novel approach for multi-view object detection in 3D scenes reconstructed from RGB-D sensor. We utilize shape based representation using local shape context descriptors along with the voting strategy which is supported by unsupervised object proposals generated from 3D point cloud data. Our algorithm starts with a single-view object detection where object proposals generated in 3D space are combined with object specific hypotheses generated by the voting strategy. To tackle the multi-view setting, the data association between multiple views enabled view registration and 3D object proposals. The evidence from multiple views is combined in simple bayesian setting. The approach is evaluated on the Washington RGB-D scenes datasets [1], [2] containing several classes of objects in a table top setting. We evaluated our approach against the other state-of-the-art methods and demonstrated superior performance on the same dataset.**

I. INTRODUCTION

Object detection is a key ingredient of scene understanding, which is leveraged by other high-level service robotics tasks, such as fetch and delivery of objects, object manipulation and object search. While this problem is widely studied with image only sensing modality, it has been demonstrated that the availability of the depth information can be effectively utilized to improve performance and the robustness of the existing systems [2], [3]. The majority of the existing approaches use the traditional object detection and recognition pipelines in the RGB-D setting treating the depth as an additional channel. These include sliding window detectors [4], [1] or methods based on local feature descriptors [5], [6], [7]. In the proposed approach we utilize local shape based descriptors extracted from image contours and implicit shape models to generate the object hypotheses. In the testing stage, the shape descriptors are extracted along the depth and contour edges supported by object proposals generated by mean-shift clustering [8] in 3D. In the single-view object detection many of the misses or false positives are often caused by occlusions or view-dependent ambiguities, which can be often resolved in a multi-view setting. Towards this end, we propose to integrate the single view detections, exploiting the fact that the views are registered together in a common reference frame. In summary, we show that unsupervised 3D object proposals support the implicit shape models favorably, reducing the number of false positives in single-view object detection. We further demonstrate that the performance of object detection can be significantly improved by integrating the evidence from multiple views. We validate the approach on the benchmark Washington RGB-D scenes dataset [1], [2] and demonstrate superior performance compared to previous methods.

The authors are with the Department of Computer Science, George Mason University, Fairfax, VA, USA. {ggeorgak, mreza, kosecka}@gmu.edu

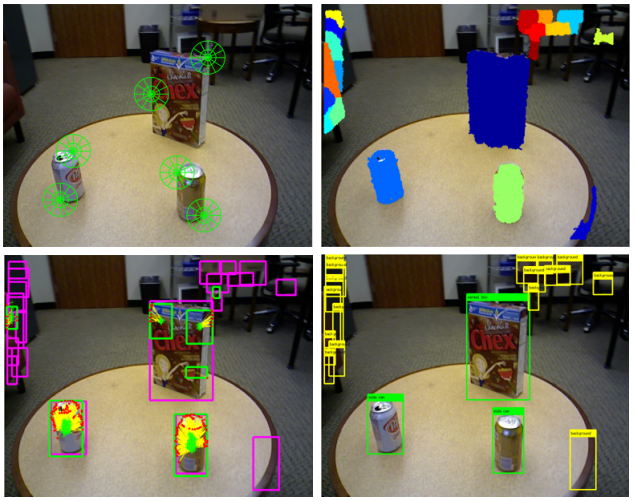


Fig. 1: Brief overview of our approach. Top row: Extraction of shape contexts and generation of object proposals. Bottom row: Example of the soda can detector overlaid with the object proposals (left) and final result (right) after integration of all detectors and multi-view information.

II. RELATED WORK

The problem of object detection and categorization has been studied extensively both in RGB and RGB-D settings. Here we briefly review the closest works to ours with respect to object proposals, local shape descriptors, voting and multi-view detection techniques. In an attempt to reduce the search space of the traditional sliding window techniques such as [9], some recent works have concentrated on generating category-independent object region proposals. Pre-trained classifiers are then evaluated for these regions [10] or various matching and clustering techniques are used to extract object models in an unsupervised or weakly supervised manner [11]. In the table-top RGB-D settings, Mishra et al. [12] uses object boundaries to guide the detection of fixation points that denote the presence of objects, while Karpathy et al. [13] performs object discovery by ranking 3D mesh segments based on objectness scores. Additional approaches of unsupervised object hypothesis generation are described in [14] and references therein.

Effective use of local object descriptors has been demonstrated in the works of [5], [6] focusing mostly on the

978-1-5090-3761-2/16/$31.00 ©2016 IEEE

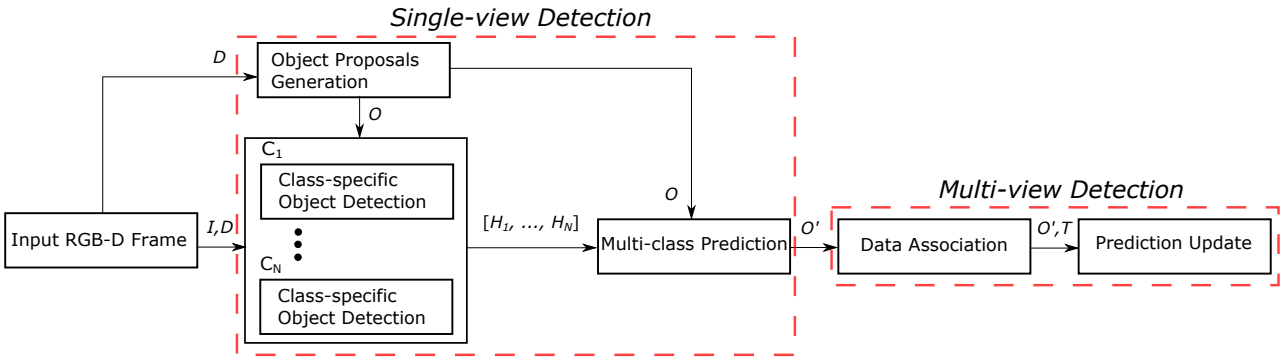


Fig. 2: Outline of our approach. Given the image frame I and the depth map D as input we first obtain the set of object proposals O and apply the class-specific object detectors for all object categories $[C_1, ..., C_N]$. For each category C we obtain a set of hypotheses. The hypotheses scores are then normalized and associated with the object proposals to get the multi-class prediction O' on the frame. Each proposal in O' now holds a probability distribution over the object categories. We then associate the proposals with the object tracks T in the scene, and update the objects probability distribution using the history of observations in the tracks.

object instance detection and textured objects. Since we are interested in detecting both textured and non-textured objects, we take advantage of objects' shape properties as encoded by the shape context descriptor. Shape Contexts were used in the past by [15] for the detection of pedestrians, cars, and bikes, and by [16] for estimating the pose of objects in cluttered RGB-D images. In the detection setting, they were combined with voting based strategies initially introduced by Implicit Shape models [17].

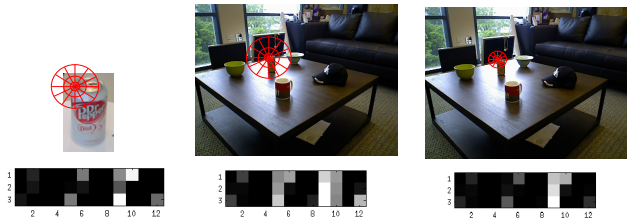
In robotic settings, where the sensor has the capability of moving, it is particularly attractive to consider multi-view detection. There are strategies where multiple views are used to register the cameras and obtain denser 3D reconstruction used for 3D object detection or categorization [2], [18], or methods where multi-view models were used to capture the correspondence between multiple viewpoints of an object [19], [20]. In our work, we will exploit the 3D reconstructions and poses to generate and associate the object proposals across multiple views, but proceed with the detection and evaluation in 2D images.

III. Approach

Figure 2 shows an overview of our approach. We discuss in more detail the object proposals generation and single view object detection in Section III-A, and the multi-view object detection in Section III-B.

A. Object Proposals and Single View Detection

In the table-top scenes, like the ones in WRGB-D scenes dataset [1], [2], small objects lie on top of the planar surfaces. Large horizontal planar surfaces are first detected and 3D points belonging to them are not considered in the proposal generation stage. The remaining 3D points are clustered using mean-shift clustering [8]. Figure 1 shows examples of clusters in different colors in the image. The radius parameter for the mean-shift is adjusted for different RGB-D frames based on the median depth value of the detected



(a) (b) (c)

Fig. 3: Effect of the Shape Context radius scaling during detection in testing stage. (a) Training image. Notice how the descriptor in (c) is more similar to the descriptor in (a) compared to (b) when the scaling is applied.

planar surfaces. The clustered 3D points are projected on the image space, followed by a connected components analysis to ensure region continuity. Each object proposal $o_k = \{pc_k, m_k\}$ is characterized by its 3D point cloud pc_k and its segmentation mask m_k in the image space. In WRGB-D scenes dataset, eliminating dominant planar surfaces performs well since the objects of interest are comprised of small non-planar segments. More advanced techniques for labeling support surfaces can also be employed [21].

In the training stage, we collect a set of shape contexts [15] for each object category from class-specific training images. We extract the *shape contexts* $d = (n_\theta, n_\phi, n_r)$, where n_θ is the number of orientations of the edge map, n_ϕ is the number of angular bins and n_r is the number of radial bins. The descriptors are extracted on our combined edge map with depth edges found using Canny edge detector and boundaries found by *gPb* detector [22] in the RGB image. In our case, we used $n_\theta = 4$, $n_\phi = 12$, and $n_r = 3$, resulting in a 144 dimensional descriptor. Additional parameter r is the radius[1]

[1]Objects sizes vary in training images and the *shape context* descriptor's radius parameter needs to be conformed to this size variance. We tackle this size variance by determining the appropriate radius of the descriptor for each object on an independent validation set.

of the shape context, defining the extent of the radial bins in image space. Each descriptor $f_i = (d_i, q_i)$ is defined by the shape context vector d_i and 2D vector q_i denoting the relative position of point i with respect to the center of the object.

Sampling for local descriptors during testing: Test images contain multiple-object instances besides background, unlike training images that consist of a single object of interest. Instead of dense sampling of edge points from the entire image (Figure 4a), we utilize the set $O = \{o_1, ..., o_K\}$ of generated object proposals to guide the descriptor sampling. The union of all proposal segmentation masks (Figure 4b) is used for uniform sampling of the shape descriptors (Figure 4c). We follow the same edge-map generation strategy as in our training stage. There is often large scale variation of objects in images compared to training images, which could lead to very different shape context representations. We address this by scaling the shape context radius as $r = r_t z_t / z$, where r_t is the radius used in training, z_t is the median depth of the object in training images and z is the sampled depth from the test image. Figure 3 illustrates an example where the scaling of the shape context radius helps the detection of a soda can.

Hypothesis generation: Each descriptor extracted from the test image at location l is compared to all shape contexts $\{d_i\}$ from the training set using the χ^2 distance choosing K best matches. For each match, the relative position towards the center of the object in the training image q_i is used to cast a vote on the test image making a prediction about the object's center (Figure 4d). Each vote is weighted using the matching score, accumulated at each location and smoothed with a Gaussian kernel to create the heat map shown in Figure 4e. To adjust for the scale difference we use the scaled offset vector $q_i z / z_l$, where z is the median depth of the training image where f_i was extracted from, and z_l is the sampled depth from location l. The scaled vectors are shown in Figure 4f. Additionally, in order to ensure that all the votes have directions towards the center of objects, we prune those for which the locations l_i and $l_i + q_i z / z_l$ belong to different object proposal masks:

$$vote_i = \begin{cases} 1 & l_i \in m_u,\ l_i + q_i z / z_l \in m_v,\ u = v \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $vote_i$ is a variable indicating the validity of each vote. Only valid votes contribute to the generation of the hypotheses. Figure 4g shows the overlap of the votes with the proposal masks, Figure 4h presents the votes after the pruning, and Figure 4i illustrates the updated heat map. We finally form the set of hypotheses as local maxima in the updated heat map, where each hypothesis consists of $h_j = (x_j, s_j, v_j)$, where x_j is the location of the hypothesis, s_j is the score, and v_j is the list of votes that contributed to the hypothesis. Figure 4j shows an example of the formed hypotheses.

Verification using object proposals: For each hypothesis, we find the closest object proposal based on the overlapping area between the bounding boxes of the hypotheses and the object proposals calculated as the intersection over union (IoU). Hypotheses with $IoU < 0.5$ and low scores are removed. Figure 4k shows the hypotheses overlaid on the object proposals, while Figure 4l depicts the surviving hypotheses after verification. In the case of over/under-segmentation during the proposal generation stage, as long as a sufficient number of correct contour points are sampled, the obtained hypothesis closest to the over/under-segmented proposal will survive the verification stage.

Score normalization and multi-class prediction: The responses of the class-specific object detectors are combined to get the multi-class prediction. The score of each hypothesis depends on the number and the quality of the votes and on the size of the object. For each object category c, we get a sample of the number of edge points from each of its training images, and find the mean μ^c and standard deviation σ^c of the samples. The scores are then normalized as $\bar{s}_j^c = \frac{s_j^c - \mu^c}{\sigma^c}$, where s_j^c is the score of hypothesis j for category c. The hypotheses from all detectors are associated with their closest object proposal in order to get a score for each category. The proposal's label is determined by the category with the highest normalized score. Proposals that are not associated with any hypothesis are labelled as background. For a single frame, each object proposal $o_k = \{pc_k, m_k, w_k, y_k\}$ is now augmented with w_k, which is the normalized score distribution over the object categories, and with y_k, which is the predicted category label.

B. Multi-view Object Detection

In the robotic setting, we often have multiple viewpoints available as the robots move around. Our multi-view object detection approach consists of a data association step, where we link object proposals in the sequence into tracks, and a prediction refinement step where we update the probability distribution over the classes for each object proposal using Bayesian update rule. We assume the first frame of the sequence to be the reference frame, and initialize a new track for each of its object proposals. When a new frame is observed, the 3D centroid of each object proposal is associated with the closest 3D centroid in the reference frame, if their distance is less than a threshold. If an object proposal cannot be associated with any proposal in the reference frame, then a new track is initialized. Each track t will consist of a set of object proposals from consecutive frames. When an object goes in and out of view due to occlusion, a new track is started. This new track can be associated to the previous track of the same object by checking the position of the proposal in the reference frame.

Class distribution update: For each proposal o in a single-view frame, we normalize its score distribution w to get the probability distribution $p(y)$, where y is a random variable corresponding to proposal o defined over the category labels. The range of the score distribution is identified by running the single-view detector on a validation set, in order to get a correspondence between a score value and a probability. A probability distribution $p(C_t)$ for each track t is maintained, where C_t is a random variable associated with

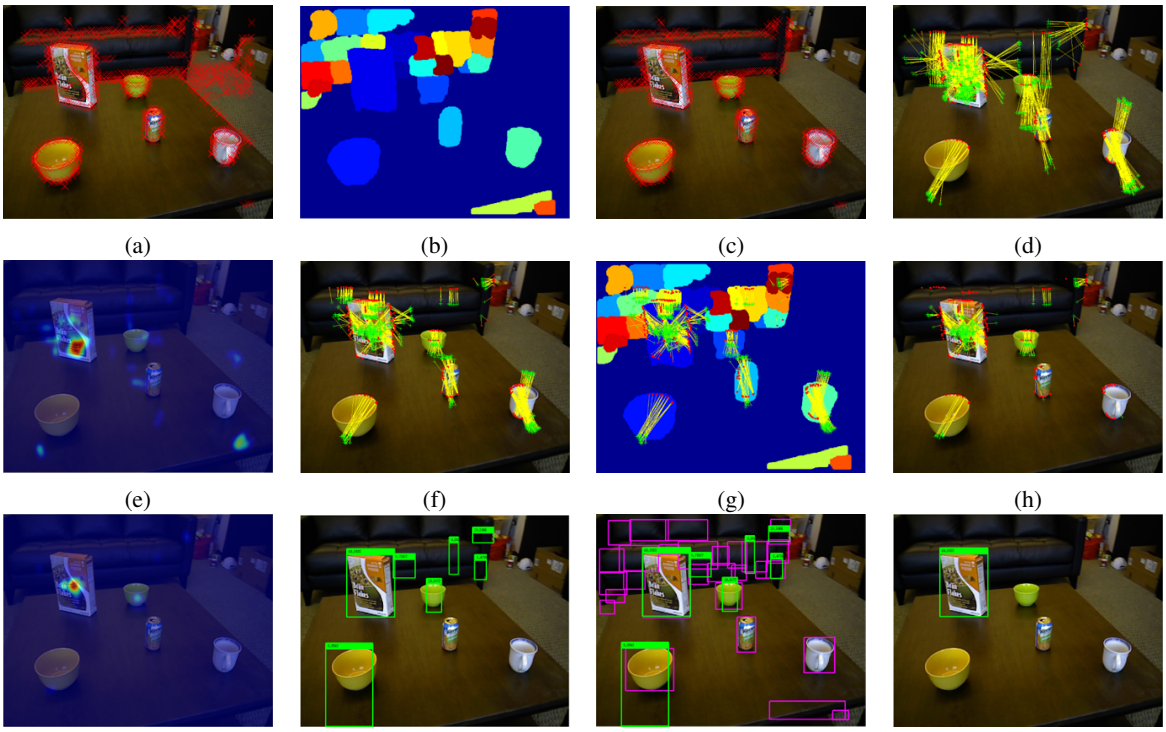


Fig. 4: Illustration of the class specific object detection approach explained in detail in Section III-A, where the object of interest is a cereal box. (a) initial sampling points, (b) segmentation masks of proposals, (c) filtered sampling points, (d) cast of votes after matching and (e) resulting heat map, (f) scaling of relative position for each vote, (g) overlay of votes over the object proposals and pruning following equation 1, (h) recasting of the remaining scaled votes and (i) the updated heat map, (j) generated hypotheses, (k) verification of the hypotheses, and (l) remaining hypothesis.

track t and it is defined over the object category labels. We begin with a uniform distribution for $p(C_t)$, and when a new proposal is associated to the track, $p(C_t)$ is updated given the history of the predictions of the proposals $\{y^1, y^2, ..., y^n\}$ using Bayes rule:

$$\begin{aligned} p(C_t|y^{1:n}) &= \frac{p(y^n|C_t, y^{1:n-1})p(C_t|y^{1:n-1})}{p(y^n|y^{1:n-1})} \\ &= \frac{p(y^n|C_t)p(C_t|y^{1:n-1})}{p(y^n|y^{1:n-1})} \end{aligned} \tag{2}$$

where $y^{1:n}$ is a short-hand notation for $\{y^1, y^2, ..., y^n\}$. The first order Markov assumption is used, such that $p(y^n|C_t, y^{1:n-1}) = p(y^n|C_t)$. For $p(y^n|C_t)$, we use the probability distribution $p(y)$ of the associated proposal computed in the single-view frame n, and $p(y^n|y^{1:n-1}) = \sum_{C_t} p(y^n|C_t)p(C_t|y^{1:n-1})$ is the normalizing factor. The category label that maximizes the posterior probability distribution $p(C_t)$ is assigned to all views in the track.

IV. Experiments

We evaluate our approach on the WRGB-D object and scenes datasets v1 [1] and v2 [2]. The WRGB-D v1 is divided in two parts. The first contains cropped images of 300 instances of objects of 51 categories, and the second is comprised of eight video scene sequences that contain some of these object instances seen from various viewpoints and with a considerable amount of occlusion. A second version of this dataset (v2) [2] contains fourteen larger video scenes and offers more variability in viewpoints. The cropped images were used for training, while the testing was performed on the video scenes.

Four experiments are conducted: 1) Class-specific object detection is evaluated on scenes from v1, while 2) single-view multi-class, 3) multi-view object detection, and 4) 3D point labeling are evaluated on v2 scenes. In all experiments, we sub-sample the videos and run the detection on every 5th frame. In the experiment on the v1 scenes, the objects of interest are *bowl*, *cap*, *cereal box*, *coffee mug*, *soda can*, and *flashlight*. We use the same set of objects except *flashlight* in v2 scenes since it is not present. The evaluation for the first three experiments is being done at the bounding box level, counting as a true positive a bounding box with an IoU larger than 0.5. For the 3D point labeling experiment, the evaluation was performed on a per-point basis.

Class-specific object detection: Each object detector is evaluated individually on the v1 scenes by calculating the Average Precision (AP) at the level of object categories. As shown in Table I, we compare to two other methods, which consist of sliding window detectors and present superior performance in almost all categories. On average, 9.5%

	Bowl	Cap	Cereal Box	Coffee Mug	Soda Can	Flashlight	Average
Tang et al. [23](HOG)	51.6	33.3	21.4	54.1	**71.0**	32.1	43.9
Tang et al. [23](HH)	71.6	71.4	50.0	61.8	60.6	44.4	60.0
Ours	**75.1**	**74.5**	**61.2**	**62.8**	69.5	**73.6**	**69.5**

TABLE I: Illustration of class-specific object detection results obtained on the WRGB-D v1 scenes dataset [1]. Comparison of Average Precision (%) for each object category and the average over all classes. HH is the combination of HOG and HONV, the feature that is introduced in [23].

	Bowl	Cap	Cereal Box	Coffee Mug	Soda Can	Background	Average
Single-View							
Pillai et al. [18]	**88.6/71.6**	85.2/**62.0**	83.8/75.4	70.8/**50.8**	78.3/42.0	**95.0**/90.0	81.5/59.4
Ours	70.7/56.8	**87.2**/49.0	**84.6/83.3**	**83.7**/34.3	**85.6/55.6**	89.0/**98.1**	**83.5/62.8**
Multi-View							
Pillai et al. [18]	88.7/70.2	89.4/72.0	**95.6**/84.3	80.1/64.1	**89.1**/75.6	96.6/96.8	89.8/72.0
Ours	**92.7/89.8**	**96.9/81.0**	87.4/**97.8**	**88.4/87.0**	86.7/**84.2**	**97.3/98.0**	**91.6/89.6**
3D Point Labeling							
HMP2D+3D [2]	**97.0**/89.1	**82.7/99.0**	**96.2/99.3**	81.0/92.6	**97.7/98.0**	95.8/95.0	**91.7**/95.5
Ours	88.5/**95.1**	79.3/95.6	91.0/98.6	**85.0/95.3**	85.8/93.4	**99.6/98.7**	88.2/**96.1**

TABLE II: Precision/recall (%) results for the single-view multi-class, multi-view, and 3D point labeling experiments on the WRGB-D v2 scenes dataset [2]. For the first two, we compare against [18] who does not exploit the depth channel and uses object proposals generated from a reconstructed scene rather than a single frame. Superior average results are shown in both single-view and multi-view detections. For the 3D point labeling experiment, we compare against [2], which is an approach tailored towards the 3D scene labeling task, and show comparable results.

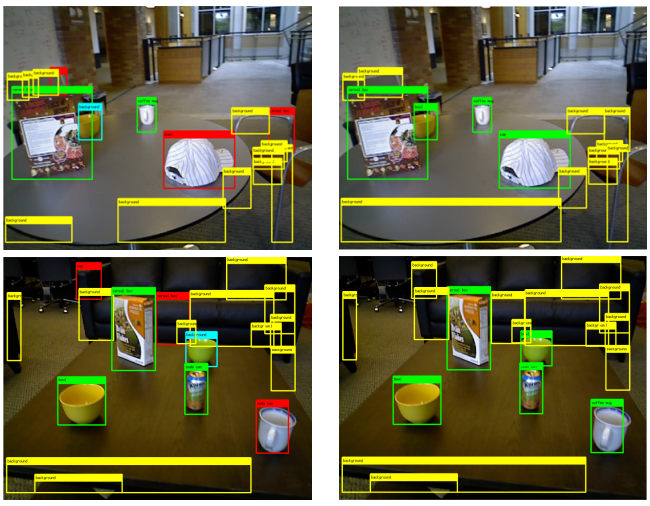


Fig. 5: A qualitative comparison between single-view (left column) and multi-view (right column) detections. Green bounding boxes are correct detections, yellow are background, cyan are missed detections, and the red are false detections. Notice that in the single-view examples we have missing and false detections which are recovered and fixed respectively in the multi-view approach.

increase in performance is achieved, when compared to the methods presented in Tang et al. [23].

Single-view multi-class detection: The multi-class predictions are evaluated on individual frames. Using the hypotheses generated from the class-specific detectors, a class label for each object proposal is selected. An empirical threshold is utilized to discard low scoring predictions and precision/recall is reported for each object category in rows

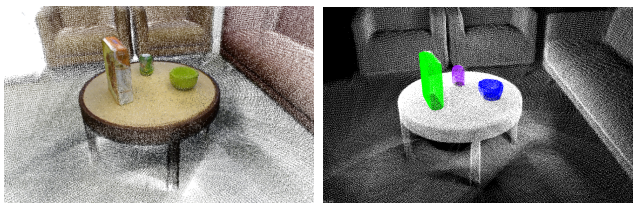


Fig. 6: Example output of our 3D labeling experiment. The coloring of the objects is: cereal box(green), soda can (magenta), and bowl(blue). Points colored in white belong to the background.

$2, 3$, and 4 of Table II. We compare our results to the state-of-the-art Pillai et al. [18] and achieve higher average performance, with an increase of 2% in precision and 3.4% in recall. Examples of single-view multi-class detections are illustrated in Figure 5.

Multi-view object detection: The performance of the multi-view object detection approach is investigated on the WRGB-D v2 scenes dataset [2]. In order to perform the data association step, we use the ground truth poses provided by the dataset. In an on-line setting, this can be done by estimating the relative poses between the views using one of the state-of-the-art methods. The consistency of our object proposals from frame-to-frame allows for the creation of long tracks throughout the scene. Longer tracks produce more reliable results, therefore, we eliminate any track which is present in less than 60% of the total number of frames in each scene. This threshold was set empirically, however, we have found that any value above 40% produces very similar results.

Using the probability distribution over the categories for each proposal obtained from our single-view multi-

class detection, we refine our predictions through sequential Bayesian estimation as discussed in Section III-B. We compare our results to the state-of-the-art[2] Pillai et al. [18] which in contrast to our work, aggregates the detector's responses over multiple frames to make a prediction. Rows $5, 6,$ and 7 of Table II illustrate our superior results in almost all object categories, with an average increase of 1.8% in precision, and 17.6% in recall. A high increase in performance when compared to the single-view detections is also noticeable. Missing detections in the single-view approach can be recovered from the history of previous observations, and similarly false detections get discarded. Figure 5 qualitatively compares single-view with multi-view detections and depicts the strength of the latter.

3D point labeling: Finally, the task of 3D point labeling is evaluated in the 14 reconstructed scenes of the WRGB-D v2 scenes dataset [2]. The 3D points of the reconstructed map are projected on the images and the segmentation masks of the proposals of our multi-view object detection output are used to label each point. We consider the same five table-top object categories and label everything else as background. Table II (rows 8,9,10) presents a comparison to the state-of-art approach of HMP2D+3D [2], for which 3D scene labeling is the primary task, while our approach focused on obtaining bounding boxes for the objects. Nevertheless, we present comparable results in terms of precision, and marginally better in terms of recall. Figure 6 illustrates a qualitative result of our 3D point labeling.

V. CONCLUSIONS

We have demonstrated a novel approach for multi-view object detection in RGB-D table-top settings, using shape-based representation with a voting strategy to generate object hypotheses. It has been shown that in single-view object detection, many false-positive hypotheses can be reduced using 3D object proposals generated from 3D point clouds with mean-shift clustering. Another approach to generate 3D object proposals would be to use the reconstructed scene map, however, we avoid this since it is more computationally demanding and increases the running time of the algorithm. In addition, it is demonstrated that the evidence from multiple views can further improve the object detection's performance. Our method achieves state-of-the-art performance in multi-view object detection on the WRGB-D scenes datasets. In the future, we plan to perform more extensive experiments with a larger number of object categories in complex scenes.

REFERENCES

[1] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[2] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3D scene labeling," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[2]In the work of [3], the authors focus on object recognition with clear background on the same dataset, and report only one experiment for detection on a subset of the available test data, therefore we do not provide a comparison.

[3] L. Bo, X. Ren, and D. Fox, "Learning hierarchical sparse features for RGB-D object recognition," in *International Journal of Robotics Research (IJRR)*, 2014.

[4] L. Spinello and K. O. Arras, "People detection in RGB-D data," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011.

[5] A. Collet, M. Martinez, and S. S. Srinivasa, "The MOPED framework: Object recognition and pose estimation for manipulation," in *International Journal of Robotics Research (IJRR)*, 2011.

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

[7] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3D recognition and pose using the viewpoint feature histogram," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[8] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," in *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2002.

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," in *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2010.

[10] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," in *International Journal of Computer Vision (IJCV)*, 2013.

[11] M. M. Cheng, Z. Zhang, W. Y. Lin, and P. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," in *IEEE Conference Computer Vision and Pattern Recognition (CVPR)*, 2014.

[12] A. Mishra, A. Shrivastava, and Y. Aloimonos, "Segmenting "simple" objects using RGB-D," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

[13] A. Karpathy, S. Miller, , and L. Fei-Fei, "Object discovery in 3D scenes via shape analysis," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[14] R. Triebel, J. Shin, and R. Siegwart, "Segmentation and unsupervised part-based discovery of repetitive objects," in *Proceeding of Robotics: Science and Systems (RSS)*, 2010.

[15] L. Wang, J. Shi, G. Song, and I. Shen, "Object detection combining recognition and segmentation," in *Asian Conference on Computer Vision (ACCV)*, 2007.

[16] C. Teo, A. Myers, C. Fermuller, and Y. Aloimonos, "Embedding high-level information into low level vision: Efficient object search in clutter," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[17] B. Liebe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," in *International Journal of Computer Vision (IJCV)*, 2008.

[18] S. Pillai and J. Leonard, "Monocular slam supported object recognition," in *Proceeding of Robotics: Science and Systems (RSS)*, 2015.

[19] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and V. Gool, "Towards multi-view object class detection,," in *IEEE Conference Computer Vision and Pattern Recognition (CVPR)*, 2006.

[20] M. Sun, H. Su, S. Savarese, and L. Fei-Fei, "A multi-view probabilistic model for 3D object classes," in *IEEE Conference Computer Vision and Pattern Recognition (CVPR)*, 2009.

[21] R. Guo and D. Hoiem, "Support surface prediction in indoor scenes," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013.

[22] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," in *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2004.

[23] S. Tang, X. Wang, X. Lv, T. X. Han, and J. Keller, "Histogram of oriented normal vectors for object recognition with a depth sensor," in *Asian Conference on Computer Vision (ACCV)*, 2012.