Multiview RGB-D Dataset for Object Instance Detection

Georgios Georgakis, Md Alimoor Reza, Arsalan Mousavian, Phi-Hung Le, Jana Košecká
Department of Computer Science
George Mason University
{ggeorgak,mreza,amousavi,ple13,kosecka}@gmu.edu

Abstract

This paper presents a new multi-view RGB-D dataset of nine kitchen scenes, each containing several objects in realistic cluttered environments including a subset of objects from the BigBird dataset [26]. The viewpoints of the scenes are densely sampled and objects in the scenes are annotated with bounding boxes and in the 3D point cloud. Also, an approach for detection and recognition is presented, which is comprised of two parts: i) a new multi-view 3D proposal generation method and ii) the development of several recognition baselines using AlexNet [14] to score our proposals, which is trained either on crops of the dataset or on synthetically composited training images. Finally, we compare the performance of the object proposals and a detection baseline to the Washington RGB-D Scenes (WRGB-D) dataset [15] and demonstrate that our Kitchen scenes dataset is more challenging for object detection and recognition. The dataset is available at: http://cs.gmu.edu/~robot/gmu-kitchens.html.

1. Introduction

The problem of detection and recognition of common household objects in realistic environments is one of the key enabling factors for service robotics. In this paper we present an approach for detection and recognition of object instances in cluttered kitchen scenes, whose scale and size affords manipulation by commonly used robotic grippers or hands. One of the driving forces of progress in detection and categorization is the use of machine learning techniques and subsequent performance evaluation of the proposed approaches on different datasets.

The benchmarks available in computer vision community are largely comprised of images from photo sharing sites with varying amount of clutter [18] [8]. With the inception of Microsoft Kinect sensor, several RGB-D datasets also have been released such as the NYUD-V2 [25], the Washington RGB-D (WRGB-D) scenes dataset [15], and the BigBird [26]. Each of these datasets has been collected for solving a specific task in mind such as categorization, pose estimation, or object segmentation. NYUD-V2 offers a large set of objects in a diverse number of scenes but lacks the multi-view aspect and puts less emphasis on the small hand-held objects. On the other hand, the WRGB-D scenes dataset focuses on the small objects, but is limited when it comes to the number of objects and the level of clutter in the scenes. We introduce a new dataset that addresses these shortcomings, by using objects from the BigBird and creating scenes that are more realistic in terms of number of objects available, clutter, and viewpoint variation, while our focus is on the detection of hand-held objects.

The contributions of the paper are as follows:

i) We present a new RGB-D multi-view kitchen scenes
dataset for instance detection and recognition of commonly encountered house-hold objects in realistic settings. The dataset is comprised of densely sampled views of several kitchen counter-top scenes and provides annotation in both 2D and 3D. The complexity of the dataset is demonstrated by comparing to the WRGB-D scenes dataset [15] on the task of object detection.

ii) We develop a multi-view object proposal generation method, which uses only 3D information from the reconstructed scenes. We show comparable results to existing proposal generation methods and demonstrate the effectiveness of the multi-view approach against a 3D single-view approach.

iii) Finally, we utilize our proposals to establish several detection baselines on the Kitchen scenes dataset that includes both Convolutional Neural Network (CNN) based approaches as well as a non CNN-based approach. During training of CNNs, we leverage different training strategies and show how they can affect the performance of the final object detection.

Figure 1 illustrates an overview of the approach.

2. Related Work

The problem of object detection has been studied extensively in a variety of domains using image only data or RGB-D images. To position our work, we review few representative approaches that have been applied in similar settings. Traditional methods for object detection in cluttered scenes follow the sliding window-based pipelines where efficient methods for feature computation and classifier evaluation were developed such as DPM [9]. Examples of using these models in the table top settings similar to ours include [16] [27]. Another commonly and effectively used strategy for object detection exploited the use of local features and correspondences between model reference image and the scene. Object detection and recognition systems that deal with textured household objects such as Collet et al. [6] and Tang et al. [30] take advantage of the discriminative nature of the local descriptors. A disadvantage of these local descriptors is that they usually perform poorly in the presence of non-textured objects, which led to alternative representations that capture object’s shape properties such as the Shape Context [32].

In an attempt to reduce the search space of the traditional sliding window techniques several recent works have concentrated in generating category-independent object proposals. Some representative works include Edge boxes [24], BING [5], and Selective search [31]. In the RGB-D settings, Mishra et al. [20] 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. Our 3D multi-view approach eliminates large planar surfaces in the scenes to facilitate the segmentation of the small objects. Recently, proposal generation methods based on convolutional neural networks (CNN) have been introduced such as the Multibox [29], the DeepMask [21], and the Region Proposal Network (RPN) in [23]. These methods perform very well on the settings they were trained for, but they require re-training in order to generalize to new settings.

Since the advent of deep learning methods, the choice of features for particular recognition tasks has been replaced by various alternatives for training or fine tuning deep CNNs or design of new architectures and optimization functions suited for various tasks. Early adoption of these techniques using CNNs such as R-CNN [11] for object detection uses object proposal methods [31] to find promising bounding boxes, extract the features using the network of [14] and trains SVM classifiers to classify each bounding box to different categories. Recently, methods such as YOLO [22], SSD [19], and Faster RCNN [23] drop the use of unsupervised proposal generation techniques and train their networks end-to-end to predict bounding boxes in addition to the classification score for each object category. Although these methods perform significantly well on challenging object detection benchmarks, they require large amounts of bounding box labeled training data.

In the RGB-D table-top settings, the availability of training sets with labeled object instances is limited. In addition, existing datasets are either captured in non realistic settings (Willow Garage [4]), do not focus on the small objects (NYU-V2 [25], SUN RGB-D [28]), or they do not provide a large set of objects in clutter (WRGB-D Scenes [15]). Our kitchen scenes dataset addresses these shortcomings by focusing on the small hand held objects, and by increasing the clutter and viewpoint variation in the scenes. More comprehensive discussion about additional RGB-D datasets can be found in [10]. Recent works such as Held et al. [12] have attempted to address the limitation of training data in these settings by proposing the strategy of pre-training AlexNet [14] on an auxiliary dataset, before training on a single image and performing object instance recognition. The problem considered, however, was the one of object recognition as opposed to detection in cluttered scenes.

In the following sections we describe the ingredients of our paper starting with a brief discussion of our Kitchen scenes dataset. Then we describe the 3D multi-view proposal generation strategy and its evaluation against 3D single-view and 2D proposals. Finally, we describe the object detection method leveraging the 3D multi-view proposals and establish several object detection baselines on the new Kitchen scene dataset.

3. Kitchen Scenes Dataset

Our Kitchen scenes dataset contains 9 videos from realistic kitchen environments. Table 1 shows scene statis-
Figure 2: Top row: Sparse reconstructions of our scenes using the state-of-the-art incremental Structure from Motion (SfM) algorithm [24]. Bottom row: Corresponding dense point clouds following the procedure in Section 5.

<table>
<thead>
<tr>
<th>Scene</th>
<th>Objects</th>
<th>Frames</th>
<th>3D points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>16</td>
<td>880</td>
<td>13.5x10^6</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>728</td>
<td>13.7x10^6</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>763</td>
<td>9.7x10^6</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>714</td>
<td>5.7x10^6</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>1316</td>
<td>11.6x10^6</td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td>451</td>
<td>7.7x10^6</td>
</tr>
<tr>
<td>7</td>
<td>11</td>
<td>740</td>
<td>13.2x10^6</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>398</td>
<td>13.5x10^6</td>
</tr>
<tr>
<td>9</td>
<td>13</td>
<td>745</td>
<td>8.3x10^6</td>
</tr>
</tbody>
</table>

Table 1: Kitchen scenes dataset statistics.

Figure 3: Annotation procedure for an object in a scene. Top row: Selected 3D points (shown in red in top left) of an object using Meshlab [1]. Bottom row: 2D bounding box annotations of the object after projecting the selected points to the images.

The video frames are first registered using RGB images utilizing an incremental Structure-from-Motion (SfM) software COLMAP [24]. COLMAP takes a set of images and produces camera poses \( C = \{C_i \in SE(3) \mid i = 1, 2, ..., N\} \) for registered images along with the sparse reconstruction of the scene as a set of points \( P = \{P_j \in R^3 \mid j = 1, 2, ..., M\} \), where each \( P_i \) is of the form \( P_i = [X_i, Y_i, Z_i] \). This registration process
yields sparse reconstruction of the scene up to a universal scale. To determine the scale parameter and to obtain the dense 3D point-cloud, we use the depth channel associated with the RGB-D frames. The camera poses from COLMAP are used to project 3D points from each RGB-D frame to the reference coordinate frame. The 3D point cloud of the depth channel is of the form \( p^d_i = [x_i, y_i, z_i] \). To determine the scale between \( Z_i \) and \( z_i \) values before projecting into the world coordinate, we use the correspondences between \( P^d_i \)'s and \( p_i \)'s. For reliable scale estimation we only consider 3D points whose \( Z_i \) values are between the \( 10^{th} \) and \( 90^{th} \) percentile. We then find the ratios between the \( Z_i \) and \( z_i \) values and keep their median as the representative scale factor \( \alpha \). The scale factor is then used to register the depth maps to the common reference frame. Figure 2 shows examples of sparse and dense point clouds of our scenes.

**Scene Annotation.** We pursue the strategy from [16] for annotating the objects in the scenes. We utilize a feature in the publicly available MeshLab tool in order to select groups of points in the registered 3D point cloud. Each group corresponds to a 3D segmentation of an object. We then crop these segmentations from the scenes and project them on the frames in which they are visible. The projected points are then used to define a bounding box over the object. Following this strategy, we are able to obtain both 3D segmentation level annotations along with 2D bounding boxes. Figure 3 shows the annotation procedure for an object.

**4. 3D Multi-view Proposal Generation**

The object detection strategy we adopt relies on generation of image proposals followed by a feature extraction and classification. We follow a simple algorithm to generate 3D object proposals from the dense point clouds of the scenes and demonstrate that they are superior to object proposals generated from a single RGB-D frame, and comparable to 2D image based RGB proposal generation methods. However, 3D multi-view object proposals can be potentially more useful in tasks such as pose estimation for robotic grasping, and can be more reliable in scenes when objects are occluded. Our algorithm starts with the removal of large planar surfaces, followed by mean shift clustering [17] of the remaining points, and finally removal of outlier points with a cuboid fitting strategy. Figure 4 highlights the procedure.

In order to generate proposals for our objects of interest in 3D, we first remove large supporting surfaces from the scene. This step removes a lot of the unwanted areas of the scene and in some cases it causes object regions to be discontinued in 3D space. A 3D Hough transform based plane detection algorithm is utilized to detect all planar surfaces in the 3D scene [14]. Figure 4 shows our detected planes of a 3D dense point cloud of a scene. The detected planes are sorted by their size and a certain percentage of them is removed. The exact number of planes that needs to be removed vary across scenes. Hence a number of different point clouds of a scene, each with a different number of planes removed, are generated. Specifically, we remove 50%, 33%, 25%, 15%, and 10% of the planes in order to create five new 3D point clouds. The plane fitting and filtering is followed by mean shift clustering on each new point cloud. Since mean shift is a density based algorithm it is more applicable to our setting than other approaches as objects in 3D tend to have compact representations. Mean shift clustering requires only one radius parameter. The clustering is applied several times, each time with a different value (ranging from 0.3 to 1.0) for the radius in order to capture objects of various scales.

The final step of our algorithm is to remove any outlier points in the resulting clusters to get a compact 3D proposals. Similar to [17], we use direct search, and achieve this by finding the tightest 3D cuboid for each cluster that includes at least 90% of the points. We relax the problem by not considering any rotation for the cuboid, and use just six degrees of freedom, three for shifting the origin and three for scaling the extent of the cuboid along each axis. The best cuboid is chosen based on the volume to number of included points ratio. The 3D points inside the best-fit cuboid comprise the object proposal.

**Evaluation.** The 3D proposals are evaluated on the images since the vast majority of object detection algorithms operate in image space. We compare our multi-view 3D proposals against a single-view 3D proposal generation algorithm, which computes object proposals using 3D point clouds from a single RGB-D frame rather than the dense point cloud of the scene. For a fair comparison, the same pipeline is followed for the generation of the single-view proposals. We also compare our 3D proposals against two widely used proposal generation algorithms, Selective search [31] and BING [5], and a CNN approach, the Faster R-CNN RPN [23]. Table 2 presents the results in terms of recall for an IoU overlap threshold of 0.5, while figure 6 illustrates recall given a range for the IoU overlap threshold. For all approaches, we have generated around 3000 proposals per image. The performance is reported on the 11 Big Bird objects [20] that have been included in our scenes. Table 2 reports also the average performance of the algorithms when the coke bottle object is not taken into consideration, since we have observed that our 3D proposals perform poorly for transparent objects due to limitations of the Kinect sensor.

The multi-view approach outperforms the single-view by a large margin since the latter cannot recover objects which are heavily occluded or lie on a surface that is not visible from a certain viewpoint. Also, the RPN shows an 11.8%
lower recall than the multi-view approach which signifies that fine-tuning the network might be necessary to achieve comparable results. In comparison to Selective search, multi-view has better recall by a small margin of 1.5%, when the coke bottle is not considered. Figure 5 shows failure cases of selective search in which the 3D multi-view approach successfully localizes the object. Overall, BING seems to outperform the other strategies, however, its performance drops rapidly while the IoU overlap threshold increases as shown in figure 6. This suggests that BING's proposals have poor overlap with the objects and they may not be suitable to use for classification.

To compare with the WRGB-D Scenes dataset, we test the performance of the 3D single-view proposal generation algorithm against its performance on our Kitchen scenes dataset. As can be seen in Table 3, for the WRGB-D dataset the algorithm achieves a recall result that is significantly higher compared to recall results in our Kitchen scenes dataset, with much less proposals per image. Note that in the case of the WRGB-D, we used just a single radius value when computing the mean shift clustering. This result is not surprising for the WRGB-D case since the algorithm detects the support surface in the vast majority of the frames and the clustering segments the objects successfully due to the low amount of clutter. Figure 7 shows examples of proposals generated on images from both datasets.

5. Object Detection

The object proposals from section 4 are category-agnostic, therefore further processing is needed to determine the object categories. Towards this goal, we establish four baselines for the object detection task on the proposed Kitchen scenes dataset. The baselines differ in the approach or the type of training data used.

1. Turntable. We use the turntable images provided by the BigBird dataset [26], which depict a single object against a
### Table 2: Recall (%) results for the proposal strategies on the kitchen scenes dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Recall(%) / No. Proposals</th>
</tr>
</thead>
<tbody>
<tr>
<td>WRGB-D [15]</td>
<td>89.3 / 17</td>
</tr>
<tr>
<td>Our Kitchen Scenes</td>
<td>62.4 / 2989</td>
</tr>
</tbody>
</table>

### Figure 6: Recall vs IoU overlap threshold for the proposal strategies on the kitchen scenes dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Recall(%) / No. Proposals</th>
</tr>
</thead>
<tbody>
<tr>
<td>WRGB-D [15]</td>
<td>89.3 / 17</td>
</tr>
<tr>
<td>Our Kitchen Scenes</td>
<td>62.4 / 2989</td>
</tr>
</tbody>
</table>

### Table 3: Recall results given an average number of proposals per frame when applying the single-view proposal generation algorithm from section 4 on the WRGB-D and our Kitchen Scene dataset. The recall was estimated using an IoU threshold of 0.5.

- For three of our baselines.

4. **HMP Scene Folds.** In addition to the three baselines, we follow the non CNN route of Hierarchical Matching Pursuit (HMP) for the fourth baseline [3].

#### Figure 5: Failure examples from Selective Search [31] (left column) where the multi-view is successful (right column). The blue bounding boxes are the ground truth, the green are the proposals with the highest overlap with the ground truth, and the red highlight the failure cases of selective search. The first two rows show missing objects due to occlusions, while in the last row two objects are merged in a single proposal.

#### Figure 6: Recall vs IoU overlap threshold for the proposal strategies on the kitchen scenes dataset.

2. **Turntable Background.** We use the turntable images provided by the BigBird dataset [26] superimposed on the backgrounds randomly sampled from the kitchen scenes.

3. **CNN Scene Folds.** We use folds of images from the dataset along with bounding boxes of the objects of interest as training examples. We follow recent trends in object detection and train a Convolutional Neural Network (CNN) for a clean background for training the CNN.

- For three of our baselines.

4. **HMP Scene Folds.** In addition to the three baselines, we follow the non CNN route of Hierarchical Matching Pursuit (HMP) for the fourth baseline [3].

**Turntable.** We use the images with single object placed on a clean background from the BigBird dataset [26]. We sub-sample these images where objects are observed from 60 viewpoints. Five random crops of each image are created in order to increase the variability of the training set. For
the background category, we randomly sample patches from kitchen scenes of the NYUD-V2 dataset [25]. The size of the generated training set is 3600 images. We feed these training images to a Convolutional Neural Network (CNN) for training. We refer to this baseline as Turntable Background.

**Turntable Background.** The second baseline extends the training set of Turntable by superimposing the object mask image from BigBird dataset on random background patches from NYUD-V2 Kitchen scenes [25]. These synthesized training images are prepared to make our detector more robust to different backgrounds. Examples of these synthetic patches are shown in Figure 8. Objects that have incomplete ground truth segmentation masks in the Big Bird due to sensor limitations are not superimposed on background patches. We train another CNN with this training set containing 6525 images and refer to this baseline as Turntable Background. AlexNet [14] is used for both of the Turntable and Turntable Background baselines as initialization, and train each fold separately for 30000 iterations with a learning rate of 0.0001.

**CNN Scene Folds.** This experiment investigates the performance of a detector that is trained on examples extracted from our Kitchen scenes images. We do a 3-Fold cross validation experiment using a random partition of the nine scenes into folds. Each training fold contains images from six out of our nine scenes. The images from the remaining three scenes are included in the test fold. The 3D multi-view proposals are used to generate examples from the training fold scenes. We follow the approach of R-CNN [11] and consider all proposals with an IoU overlap larger than 0.5 with the ground truth as instances of a particular object. Proposals having an IoU less than 0.3 are considered as background. We ignore the rest of the proposals having values in between these two ranges. We sub-sample proposals every 10 images to avoid high correlation between the examples and end up with a training set of around 70000 patches for each fold. Following the CNN architecture of Turntable and Turntable Background baselines, we use the AlexNet for initialization, and train each fold separately for 30000 iterations with a learning rate of 0.0001.

**HMP Scene Folds.** We also compare our detection against a non-CNN architecture based feature generation method. We selected Hierarchical Matching Pursuit (HMP) approach, which is used in the context of detecting, labeling, and segmenting objects similar to our Kitchen Scenes dataset environment [15, 3]. HMP is a sparse coding technique that learns features in an unsupervised manner from the training images. We applied HMP on our 3-Fold cross-validation experiment using the publicly available implementation provided by the authors [3]. In each fold, we learned dictionaries from both the grayscale and RGB channels of the proposal-patches of the training fold. The learned dictionaries are used to extract features, i.e., the sparse codes of the patches, which are then concatenated together. The dictionary sizes in the 1st and 2nd layer are 75 and 150 respectively. We used patch sizes of 5x5 and 4x4 respectively for learning the dictionaries in these two layers. A linear SVM is trained on the features of the training fold.

**Discussion.** The detection results for all baselines are illustrated in Table 4. CNN Scene Folds outperforms the next best baseline by 19.2% in terms of mean Average Precision (mAP), while the HMP Scene Folds and Turntable produce the lowest performance. It is evident that training on exam-
Turntable | 1.0 | 25.7 | 8.6 | 2.8 | 17.2 | 21.2 | 24.0 | 7.6 | 40.1 | 1.2 | 2.6 | 89.5 | 20.1
Turntable Background | 0.1 | 33.0 | 15.9 | 17.9 | 18.0 | 19.9 | 26.0 | 10.8 | 32.5 | 3.2 | 3.3 | 89.5 | 22.5
HMP Scene Folds | 0.0 | 26.8 | 22.8 | 13.2 | 2.7 | 33.7 | 17.2 | 4.1 | 14.3 | 11.5 | 8.0 | 86.5 | 20.1
CNN Scene Folds | 3.5 | 48.8 | 50.0 | 27.6 | 27.9 | 52.4 | 48.1 | 18.8 | 53.6 | 46.9 | 32.7 | 90.6 | 41.7

Table 4: Average precision (%) results for the object detection baselines on the kitchen scenes dataset.

WRGB-D [15] | bowl | cap | cereal box | coffee mug | soda can | background | mAP
--- | --- | --- | --- | --- | --- | --- | ---
Turntable | 42.2 | 71.7 | 72.3 | 49.0 | 57.7 | 82.5 | 62.6

Table 5: Object detection results for the WRGB-D Scenes dataset [15]. The detection was performed using the procedure of the Turntable baseline for this dataset.

6. Conclusions

We have presented a new RGB-D multi-view dataset for object instance detection and recognition of commonly encountered household objects in realistic settings. The dataset was utilized to demonstrate the effectiveness of a novel 3D multi-view object proposal method against a 3D single-view method, while at the same time achieving comparable results to established image-based proposal methods. The generated proposals were further used to establish and compare several object detection pipelines on the dataset. These include both deep-learning-based approaches and a non-deep-learning-based approach. Not surprisingly the performance of the CNN based strategies is superior to previously used methods. Also, the best performance is achieved by training and testing on the different folds from the kitchen dataset. Note that the performance drops significantly when the dataset used for training does not contain the type of backgrounds found in test, e.g., training with random backgrounds. Comparative experiments on the WRGB-D dataset reveal differences in nature of complexity with respect to the new Kitchens scenes dataset.

Acknowledgments. We acknowledge support from NSF NRI grant 1527208. Some of the experiments were run on ARGO, a research computing cluster provided by the Office of Research Computing at George Mason University, VA. (URL: http://orc.gmu.edu).
References