RGB-D Scene Recognition with Object-to-Object Relation

Xinhang Song 1Key Lab of Intel. Inf. Proc., Inst. of Comput. Tech., Chinese Academy of Sciences, Beijing, 100190, China 2University of Chinese Academy of Sciences, Beijing, China xinhang.song@vipl.ict.ac.cn

Chengpeng Chen 1Key Lab of Intel. Inf. Proc., Inst. of Comput. Tech., Chinese Academy of Sciences, Beijing, 100190, China 2University of Chinese Academy of Sciences, Beijing, China chengpeng.chen@vipl.ict.ac.cn

Shuqiang Jiang 1Key Lab of Intel. Inf. Proc., Inst. of Comput. Tech., Chinese Academy of Sciences, Beijing, 100190, China 2University of Chinese Academy of Sciences, Beijing, China sqjiang@ict.ac.cn

ABSTRACT
A scene is usually abstract that consists of several less abstract entities such as objects or themes. It is very difficult to reason scenes from visual features due to the semantic gap between the abstract scenes and low-level visual features. Some alternative works recognize scenes with a two-step framework by representing images with intermediate representations of objects or themes. However, the object co-occurrences between scenes may lead to ambiguity for scene recognition. In this paper, we propose a framework to represent images with intermediate (object) representations with spatial layout, i.e., object-to-object relation (OOR) representation. In order to better capture the spatial information, the proposed OOR is adapted to RGB-D data. In the proposed framework, we first apply object detection technique on RGB and depth images separately. Then the detected results of both modalities are combined with a RGB-D proposal fusion process. Based on the detected results, we extract semantic feature OOR and regional convolutional neural network (CNN) features located by bounding boxes. Finally, different features are concatenated to feed to the classifier for scene recognition. The experimental results on SUN RGB-D and NYUD2 datasets illustrate the efficiency of the proposed method.

KEYWORDS
Intermediate representation; spatial layout; object detection; RGB-D; scene recognition

1 INTRODUCTION
The goal of scene recognition is to annotate images with scene categories. Humans have innate talent to recognize those abstract scenes without hard training, while it is typically challenging for computer, since most scenes are abstract representations composed of many less abstract entities in local regions (e.g., water, wall, people, chairs), which require reasoning from digital pixels to the abstract scenes. Scene categories can be inferred from low-level visual descriptors [29, 31, 40, 41], with the “bag” model of local visual descriptors. However, it’s difficult to obtain the required statistical knowledge for inferring scene categories from low-level visual features, due to the semantic gap [36].

An alternative approach is to split the scene reasoning in two steps with smaller semantic gap (including pixels to objects, objects to scenes), with the intermediate representations, such as object banks [23], latent topic models [4, 11, 24, 25, 27], mid-level parts [21] and regional (in multi-scale) features of convolutional neural network (CNN) [6, 16]. However, due to the object co-occurrences between different scenes, only representing images with such intermediate representations suffers from the problem of ambiguity for recognizing scenes.

A more discriminative approach is to represent images by intermediate representations with spatial layout. Spatial layout is helpful to distinguish some particular scenes that are confused by intermediate presentations (see Fig. 1, different scenes such as “dining room” and “classroom” may contain similar objects like “table” and “chair”). When only using the intermediate representations, the images of those two categories are difficult to be distinguished. However, when considering the intermediate representations with spatial layout, those scenes can be distinguished, i.e., in “dining room”, the table is usually surrounded by the chairs, while in “classroom” the chairs are always behind the “table”.

One intuitive way to obtain both intermediate representations and spatial layout is using object detection techniques [14, 15, 28], which can simultaneously provide object labels and the position of their bounding boxes. The intermediate representations are built on the statistical histogram of objects, and the spatial layout is analyzed by the position (coordinates) of the predicted bounding boxes. Particularly, some previous works [2, 13, 39] have implemented object detection techniques for scene recognition. However, these works either only use the object labels but not the spatial information [2, 13], or only locate the position of bounding boxes to extract local features [39] while ignore the object labels.

Particularly, the RGB-D data is helpful to better locate the objects. The low cost depth sensors, such as Microsoft Kinect, can capture RGB-D data, which extends traditional RGB recognition by including depth information. Depth camera can provide spatial information to detect object boundaries and understand the global layout of objects in the scene. Combining RGB with depth images to recognize scenes usually achieves better performance than only using RGB or depth images. Previously, depth information is modeled using handcrafted features. Although automatic feature learning from the data with CNN can provide more discriminative representations, however, the lack of large RGB-D datasets makes the complex CNN models not applicable. Recently, Song et al. [33]...
2 RELATED WORK

2.1 Intermediate representation

Vogel and Schiele [38] proposed to represent natural scenes with regional intermediate representation of local concepts such as water, rocks or foliage. Similarly, Object bank [23] trains classifiers with multi-scale images from ImageNet to obtain a more descriptive representation. The classemes representation [3] is based on a set of fixed basis classes. Attributes [10, 22] follow a similar idea, where classifiers are trained to detect whether certain attributes are present or not. Attributes can be modeled at both local and global levels, and defined for both objects [10] and scenes [26]. However, without exact object detection, the intermediate representation is not that reliable to obtain good performance for scene recognition.

2.2 Object detection

By leveraging the successful deep neural networks from object classification, Girshick et al. [15] propose an R-CNN framework for object detection, which aggregates the region proposal technique [37] with CNN classifier. Although R-CNN improves the previous works [7, 12] with a large margin in accuracy, the repeated CNN feature extraction (for each proposal) still limits R-CNN for efficiency. Thus, Girshick et al. [14] propose the Fast R-CNN framework to improve the R-CNN with ROI pooling layer and multi-task regression layer. The former accelerates R-CNN by pooling on feature maps rather than image pixels, which only requires the CNN feature extraction once for all the proposals of one image. The later jointly train the object classifiers and bounding boxes regressor. While, this Fast R-CNN still relies on the external region proposal mechanism. Recently, Ren et al. [28] propose the Faster R-CNN framework to address the region proposal problem with a region proposal network (RPN), which establishes the end-to-end training for object detection.

More recently, some works [2, 13, 39] have implemented object detection technique for scene recognitions. George et al. [13] propose to represent scene images by object distributions based on object detection, which is then optimized to distinguish fine-grained scenes by semantic clustering. Bappy et al. [2] combines object detection with manual annotation for active learning of scene recognition. Only representing images with object distributions lacks of spatial information. Wang et al. [39] extract local features located by the object detection, and local features are embedded with Fisher Vector.

In this paper, we leverage both object distributions and spatial information to represent images for scene recognition.

2.3 RGB-D recognition

Handcrafted features [1, 18] are engineered to capture some specific properties that the engineer has observed to be useful for RGB-D image recognition. Also, Socher et al. [32] use a single layer CNN trained on patches. The network is learned in an unsupervised way, and then combined with a recurrent convolutional network (RNN). After the availability of large datasets, the interest has shifted towards deep networks pretrained with RGB data from ImageNet or Places, which have shown better performance than previous smaller CNN models [19, 35]. Gupta et al. [19] use R-CNN on depth images to detect objects in indoor scenes. In order to address the problem of limited training data, they augment the training set by rendering additional synthetic scenes.

Fine-tuning or transferring the RGB pretrained CNN models to depth data are proposed in recent works [20, 39, 43]. Wang et al. [39] fine-tune the pretrained CNN with depth images, and extract
features on both local regions and whole images, then combine all features of RGB and depth patches/images in a component aware fusion method. Zhu et al. [43] jointly fine-tune the RGB and depth CNN models by including a multi-model fusion layer, simultaneously considering inter- and intra-modality correlation, meanwhile regularizing the learned features to be compact and discriminative. Alternatively, Gupta et al. [20] propose to transfer RGB CNN model to the depth data according to the RGB and depth image pairs. While Song et al. [34] do not depend on fine tuning or transferring, they train CNN with depth patches by weak-supervision, which achieves even better performance than fine tuning from RGB.

3 IMAGE REPRESENTATIONS BASED ON CONTENT RELATION

We first introduce the proposed intermediate representations with spatial layout, and then compare different types of image representations based on object distributions.

Figure 2: Co-occurrences between objects and scenes, (a) correlation matrix, (b) correlation matrix after reordering with clustering, where the "(#)" represents the cluster ID.

Figure 3: Feature visualization of a toy example, (a) one image of "dinning_room" from SUN RGB-D, and the object annotations are from the ground truth, (b) the visualization of $P^I_O$, which is normalized by the counting of appearances, we select 3 object categories for this toy example (c) the visualization of $P^I_{OO}$, the counting of object co-occurrences, (d) the visualization of $P^I_{OOR}$, 4 types of relative relations between the centers of object bounding boxes are selected here, i.e., the "left-up" means the co-occurrences between objects with such relation. Note that (c) and (d) are the visualizations of feature, and for training classifiers, the features are stretched to vectors.
3.1 Object-to-object relation representation

The common intermediate representations are object distributions, which can be represented as $P_O^I = [p_{i1}^O, p_{i2}^O, \ldots, p_{|O|}^O]$ (see Fig. 3b), where $p_{ij}^O$ is the appearance frequency of object $i$ observed in the image $I$, and $O$ is the object vocabulary. With this type of representation, the object co-occurrences between scenes may confuse the recognition. A more discriminative way is representing images with the object-to-object co-occurrences, $P_{OO}^I = \begin{bmatrix} p_{11}^O & \cdots & p_{1|O|}^O \\ \vdots & \ddots & \vdots \\ p_{|O|1}^O & \cdots & p_{|O| |O|}^O \end{bmatrix}$ (also see Fig. 3c), where each element $p_{ij}^O$ represents the appearance frequency of co-occurrence between object $i$ and $j$.

The proposed intermediate representation is composed of objects and relative relations, which is represented as the triplet $(\text{object}, \text{relation}, \text{object})$, also denoted as object-to-object relation (OOR) representation. The proposed OOR can be formulated as a tensor $P_{OOR}^I \in \mathbb{R}^{(|V|)^2 |V| |V|}$ (see Fig. 3d), where $V$ is the vocabulary of relative relations between objects. Particularly, we define the relative relations based on the coordinates of object bounding boxes $b = [x_l, y_u, x_r, y_d]$, where $[x_l, y_u]$ is left-top corner, and $[x_r, y_d]$ is the right-down corner. The relation between object $i$ and $j$ are represented as

$$V(i, j) = \begin{bmatrix} f\left(b_i^l - b_j^l\right) \\ f\left(b_i^u - b_j^u\right) \\ f\left(b_i^r - b_j^r\right) \\ f\left(b_i^d - b_j^d\right) \end{bmatrix}$$

(1)

where $f(x) = \begin{cases} 0, & \text{if } x \leq 0 \\ 1, & \text{if } x > 0 \end{cases}$. There are $|V| = 2^4 = 16$ different types of relative relations between objects.

3.2 Insights of OOR representation

We analyze the problem of object co-occurrences between scenes, and compare the proposed OOR with others on the SUN RGB-D [33] dataset. This dataset contains 40 categories with 10335 RGB-D images. Following the publicly available split in [33, 39], 19 most common categories are selected, consisting of 4,845/4659 images for training/test. Also, 19 popular object categories are selected in [33] for the object detection, whose annotations are given with the bounding box coordinates and object labels.

3.2.1 Object co-occurrences between scenes. We illustrate the statistics of the co-occurrences (appearing in the same image) between objects and scenes in a correlated matrix $W$, which is visualized in Fig. 2a. Each element $W_{ao}$ in the matrix is the co-occurrence frequency between scene $s$ and object $o$. With such matrix $W$ and the object distributions $P_O^I$, the scene probability distribution can be obtained as:

$$P_S^I = P_O^I W^T$$

(2)

With $P_S^I$, we can predict the scene label by finding the scene category with maximum probability. Also, the scene label can be predicted by training SVM classifiers based on the representations of $P_O^I$, $P_{OO}^I$, and $P_{OOR}^I$.

3.2.2 Insights in the analysis of confusion. Confusion matrices of different representations are compared in the Fig. 4, where the accuracy is calculated by the mean of diagonal. Note that the accuracy of $P_S^I$ (Fig. 4a) is much lower than others. Particularly, many scenes such as “dining area,” “conference room” and “dining room” are misclassified to “classroom”. This confused problem is mainly caused by similar object co-occurrences among these scenes. In order to better visualize such confusion between scenes, the correlation matrix in Fig. 2a is re-organized with the spectral clustering [5] algorithm. The Fig. 2b visualizes the correlation matrix after re-ordering the rows and columns. In practice, the number of cluster is $k = 5$, the scenes (rows) are reordered by the cluster ID, and the objects (columns) are reordered according to the correlation to the scenes. It can be observed that many scene categories such as “classroom”, “dining area”, “dining room”, and “conference room” are clustered together because of the co-occurring of objects like “chairs” and “tables” (see the first two columns in Fig. 2b), this is also the reason of the confusion between scenes in Fig. 4a.

Rather than using Eq. 2 to predict scene labels, we also recognize scenes with SVM classifier on different types of object representation of $P_O^I$, $P_{OO}^I$, and $P_{OOR}^I$, whose confusion matrices are visualized in Fig. 4b-d. The results of Fig. 4a and b are based on the same object representation $P_O^I$, but using different classifiers, i.e., Eq. 2 and SVM. The comparison between them shows the efficiency of more discriminative classifier SVM. Moreover, using $P_{OOR}^I$ with SVM (in Fig. 4d) results even better performance than $P_O^I$ and $P_{OO}^I$, including better overall accuracy and less confused problems in the confusion matrix. For instance, the confusion between “classroom” and “dining room” in Fig. 4d is much better than that in Fig. 4a, where the rate of misclassifying from “dining room” to “classroom” gets lower from 87% to 20%. This also supports our hypothesis (in Fig. 1) that the scenes with object co-occurrences also can distinguished by including the relative relations into the intermediate representation.

The above evaluations are based on the ground truth of annotated objects. While for the real world scene recognition, the annotations of object are not available. Thus, we implement object detection technique to obtain the labels and bounding boxes of objects, which will be introduced in next section.

4 DETECTION OF IMAGE CONTENT

In order to include object information for more accurate scene recognition, we apply object detection technique on each image, and represent the images with features based on the detected results. In particular, we adapt Faster R-CNN model [28] to the RGB-D data, and the RGB and depth modalities are combined at proposals generating process, more details are introduced in this section.

4.1 RGB-D object detection

Being a region based method for object detection, Faster R-CNN includes a branch, named region proposal network (RPN), to generate candidates of bounding box. On each candidate region, the CNN hidden features are first extracted by region of interest (RoI) pooling layer, then fed to the classifying layers, which consist of two classifiers, a softmax classifier for object labels and a regressor for the coordinates of bounding box.
4.2 RGB-D proposal fusion

With the RPN branches, the proposals $B_{rgb} = \{B_{rgb}^{(1)}, \ldots, B_{rgb}^{(n)}\}$ and $B_{depth} = \{B_{depth}^{(1)}, \ldots, B_{depth}^{(n)}\}$ are obtained on each pair of RGB and depth images, where each $B_{rgb}^{(i)}$ and $B_{depth}^{(i)}$, $i = [1, \ldots, n]$, contain the proposal information in form of bounding box, including the coordinates $r_{rgb}^{(i)}, d_{depth}^{(i)}$ and confidence score $c_{rgb}^{(i)}$ and $c_{depth}^{(i)}$, respectively. Besides, we set

$$N_{rgb}^{(i)} = \min \left\{ \frac{c_{rgb}^{(i)}}{\alpha}, \lambda \right\}$$
$$N_{depth}^{(i)} = \min \left\{ \frac{c_{depth}^{(i)}}{\alpha}, \lambda \right\}$$

Where $\left\lfloor \frac{c_{rgb}^{(i)}}{\alpha} \right\rfloor$ denotes the number of proposals with value $c_{rgb}^{(i)}$ larger than $\alpha$, which is considered as the confidence threshold, and $\lambda$ is an empirical value of the minimal number of proposals, ensuring enough proposals. The hyper-parameters $\alpha$ and $\lambda$ are evaluated in the following section. Note that these two hyper-parameters $\alpha$ and $\lambda$ decide the top $N_{rgb}^{(i)}$ ($N_{depth}^{(i)}$) proposals selected from $B_{rgb}^{(i)}$ ($B_{depth}^{(i)}$) based on $c_{rgb}^{(i)}$ ($c_{depth}^{(i)}$).

Intuitively, the proposals obtained from RGB and depth models should be combined to improve the detection performance. By directly combining the proposals of $B_{rgb}^{(i)}$ and $B_{depth}^{(i)}$, we get $B_{rgb, depth}^{(i)} = B_{rgb}^{(i)} \cup B_{depth}^{(i)}$. However, it is unavoidable to lead to the overlapping between $B_{rgb}^{(i)}$ and $B_{depth}^{(i)}$, making the features of overlapping region redundant for further scene recognition task. To avoid overlapping, a pooling process on bounding boxes level is performed on these proposals. Particularly, NMS is applied to get the pooled proposals $B_{pooled} = \{B_{pooled}^{(1)}, \ldots, B_{pooled}^{(n)}\}$, in which

$$B_{pooled}^{(i)} = NMS \left\{ B_{rgb}^{(i)} \cup B_{depth}^{(i)} \right\}$$

where $\beta$ is the Intersection-over-Union (IoU) overlapping threshold for non-maximum suppression. That means the regions with relatively overlapping area larger than $\beta$ will be merged.

5 MULTI-MODAL SCENE RECOGNITION

Some previous RGB-D works [33, 34, 43] recognize scene by training the CNN models with images, and extracting global features of CNN activation (e.g., fc7 or fc8) based on the whole image. While an alternative is to extract local features with object detection [39], and the local and global features are combined with Fisher Vector encoding. In [39], the object detection is mainly used for locating the regions and extracting features, while the object labels are ignored. Thus, besides extracting the local and global hidden features of CNN activation, we also learn OOR representation $P_{OOR}^{(i)}$. All these features (hidden features of CNN activation and semantic features OOR) are concatenated to feed to the classifier of scenes.

Multi-modal global features. Rather than the very deep and complex architectures, Song et al. [34] propose a relatively shallow and simple architecture for depth data that trained with weak-supervision. Then the authors concatenate depth and RGB CNNs with fully connected layers. In order to extract multi-modal global features, we apply the DCNN model in [34] to extract depth features, and combine RGB and depth data with RGB-D-CNN in [34].
with RPN, the local features of CNN (fc7) activation are extracted (objects). Finally, the concatenated features are fed to the SVM. Detected results are considered to be true or false positives.

Multi-modal multi-feature combination. The global and local features are extracted from CNN activation, which are further combined with an extended multi-layer network, training with the loss of scene labels. Thus, the local and global CNN features are combined to a global CNN feature, which is then concatenated with OOR to include the spatial layout information of image content (objects). Finally, the concatenated features are fed to the SVM classifier for scene recognition.

6 EXPERIMENTS

6.1 Setting

Dataset. Our approaches are evaluated on two datasets: NYU Depth Dataset version 2 (NYUD2) [30] and SUN RGB-D [33]. The former consists of 27 indoor categories. Following the original training/test split of images 795/654 in [30], all 27 categories are reorganized into 10 categories, where some of categories with few images are combined into a joint category "other". The latter contains 10335 RGB-D images in 40 categories. Following the public split in [33, 39], the 19 most common categories are selected, consisting of 4,845/4,659 images for training/test. The split is provided in the toolbox of SUN RGB-D dataset. For the object detection evaluation, we follow the same split of scene recognition, since the object detection further serves for the scene recognition. All depth images are encoded to HHA images using the code in [19].

Classifier. We optionally use class-specific weights (weight is an option implemented in liblinear[9]) to compensate the imbalance in training data. The weight \( w = \{w_1 \ldots w_M\} \) of each category \( m \) is computed as \( w_m = (\min_{i \in M} N_i \setminus N_m)^\beta \), where \( N_m \) is the amount of training images of the \( m^{th} \) category. We practically set \( \beta = 1 \).

Evaluation metric. Following [33, 39], we report the average class accuracy for the scene recognition. We follow the evaluation method of [8] to report the average precision (AP) for object detection. Detected results are considered to be true or false positives according to the overlap area with ground truth bounding boxes. To be considered a correct detection, the overlap area \( a_o \) between the predicted bounding box \( B_p \) and ground truth bounding box \( B_{gt} \) must exceed 50% by the formula:

\[
a_o = \frac{\text{area}(B_p \cap B_{gt})}{\text{area}(B_p \cup B_{gt})}
\]

6.2 Object detection

6.2.1 Implementation. For the object detection model training, both FRCN-RGB and FRCN-Depth use the ZF net [42] as the pre-trained model, and follow the empirical parameter setting of Faster RCNN.

For the training of FRCN-RGBD model, all the convolutional layers are copied from FRCN-RGB and FRCN-Depth, and the hidden layer outputs of the region proposals \( B_{\text{pool}} \) are concatenated to further train the new fully-connected and loss layers. The weights of the previous models are used for initializing the corresponding new layers, which speeds up model training significantly. Particularly, since the concatenation of the outputs of hidden layers results in double amount of weights between the concatenated layer and the first fully-connected layer, the corresponding weights of FRCN-RGB and FRCN-Depth are concatenated for the initialization as well, while those weights are divided by 2 to avoid activation saturation. For the other two fully-connected layers, the sizes of weight are same as that in FRCN-RGB and FRCN-Depth. Here, we take the average of weights from both models to initiate those two fully-connected layers of FRCN-RGBD model. With such trick of weights initialization, the model converges in a few thousand iterations.

For the object detection task, we set \( \beta = 200 \) for both FRCN-RGB and FRCN-Depth models, and \( \alpha = 0.3, \beta = 0.6 \) following the empirical setting of Faster RCNN work. Note that this setting is only used for object detection. For the scene recognition task, we practically find the best parameters.

6.2.2 Detection performance. We evaluate the performance of object detection on SUN RGB-D dataset, and the comparisons between different models of different modalities are illustrated in Table 1. Comparing between RGB and depth modalities, depth model works better on the object categories such as "bathtub", "bed", "chair", "pillow" and "table", that contain enough depth information in shapes, while works much worse than RGB model on that objects such as "door", "dresser", "monitor", "tv", that barely have depth in shapes (thin and flat in shape). However, with the RGB and depth fusion, the performances of all category are improved.

![Table 1: Object detection AP (%) of SUN RGB-D](image-url)

<table>
<thead>
<tr>
<th>Model</th>
<th>bathtub</th>
<th>bed</th>
<th>bookshelf</th>
<th>box</th>
<th>chair</th>
<th>counter</th>
<th>desk</th>
<th>door</th>
<th>dresser</th>
<th>garbage_bin</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRCN-RGB</td>
<td>34.4</td>
<td>63.2</td>
<td>39.8</td>
<td>12.5</td>
<td>43.9</td>
<td>42.2</td>
<td>20.3</td>
<td>30.7</td>
<td>30.0</td>
<td>40.0</td>
</tr>
<tr>
<td>FRCN-Depth</td>
<td>54.5</td>
<td>71.6</td>
<td>25.5</td>
<td>5.0</td>
<td>45.4</td>
<td>39.5</td>
<td>22.2</td>
<td>10.5</td>
<td>18.0</td>
<td>34.2</td>
</tr>
<tr>
<td>FRCN-RGBD</td>
<td>57.5</td>
<td>75.6</td>
<td>44.2</td>
<td>17.7</td>
<td>49.6</td>
<td>48.9</td>
<td>25.4</td>
<td>33.6</td>
<td>40.2</td>
<td>49.2</td>
</tr>
</tbody>
</table>

Table 1: Object detection AP (%) of SUN RGB-D
and the overall result (mAP) of RGB-D model outperforms RGB and depth models with a large margin about 10%.

6.3 Scene recognition

6.3.1 Parameter evaluation of object detection for scene recognition. For the scene recognition, the detected information of objects are used for feature extraction. We first obtain region proposals with RPN, then extract fc7 (last but one fully-connected layer) activation for each proposal, and finally the regional fc7 activation are combined to a global feature vector by max pooling on all proposals from one image. With FRCN-RGBD model, the combined feature is denoted as $F_{fc-rgbd}$. For comparison, we also consider another way of RGB-D fusion. For each image, we concatenate activation from FRCN-RGB and FRCN-Depth models to get a 8192-dimensional (two fc7 4096-dimensional activation) feature vector, denoted as $F_{fc-concat}$. Besides, to speed up classification model training, the general dimension reduction method PCA, is perform on $F_{fc-concat}$ and $F_{fc-rgbd}$, which projects them to 512-dimensions. Note that, this dimension reduction process speeds up the classifier training barely with accuracy loss.

The evaluations of parameter $\alpha$ and $\lambda$ are illustrated in Fig. 5. Note that, for the scene recognition, the larger $\lambda$ results more overlapped area between proposals, redundant local features and proposals with lower confidence. Thus, we select much smaller $\lambda$ for scene recognition. For both $F_{fc-concat}$ and $F_{fc-rgbd}$ in Fig. 5 a and b, the smaller $\alpha$ and larger $\lambda$ lead to better results, and best results are obtained when $\alpha = 0.3$ and $\lambda = 13$. The average number of detected objects of each image are illustrated in Fig. 6. For the larger $\alpha$ (e.g., $\alpha = 0.4$ or $\alpha = 0.5$), it shows that the average number of detected objects obviously increases with $\lambda$, which is also the reason of accuracy improvement in Fig. 5, particularly for $\alpha = 0.5$. Based on the selection of best results, we set $\alpha = 0.3$ and $\lambda = 13$ of object detection model for the rest experiments of scene recognition.

6.3.2 Intermediate representation. We build the intermediate representations of different modalities based on the object detection results. The comparisons between different intermediate representations are illustrated in Table 2. Note that, the main difference between the results in this table and Fig 4 is the source of object labels and their bounding boxes. The intermediate representations of RGB, depth and RGB-D in this Table 2 are based on the detected results of FRCN-RGB, FRCN-Depth and FRCN-RGBD, while the results in Fig. 4 are relied on ground truth annotations. Compared with the intermediate representations in Fig. 4, the $P^I_S$ are much worse due to the lack of confidence for the object detection results. However, the other results of $P^I_O$, $P^{OO}_O$ and $P^{OOR}_O$ outperform the ones in Fig. 4, especially the ones built on the detected results after RGB-D fusion, obtaining the best result with 36.3%. Comparing with different representations, the $P^{OOR}_O$ achieves best results in each modality, which outperforms $P^{OO}_O$ from 0.8% (RGB) to 2.9% (RGB-D), where the main improvement benefits from the depth data.

6.3.3 Hidden features. The results of scene recognition with local and global features of CNN activation on different modal data

![Figure 5: Parameter evaluation of object detection for scene recognition, evaluated on the SUN RGB-D dataset, (a) $F_{fc-concat}$, concatenation of fc7 activation of FRCN-RGB and FRCN-Depth, (b) $F_{fc-rgbd}$, fc7 activation of FRCN-RGBD.](image)

![Figure 6: Average number of objects detected from each image on SUN RGB-D.](image)

<table>
<thead>
<tr>
<th>Intermediate representations</th>
<th>RGB</th>
<th>Depth</th>
<th>RGB-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P^I_S$</td>
<td>16.8</td>
<td>13.9</td>
<td>17.8</td>
</tr>
<tr>
<td>$P^I_O$</td>
<td>31.4</td>
<td>26.5</td>
<td>31.9</td>
</tr>
<tr>
<td>$P^{OO}_O$</td>
<td>32.7</td>
<td>28.7</td>
<td>33.4</td>
</tr>
<tr>
<td>$P^{OOR}_O$</td>
<td>33.5</td>
<td>30.0</td>
<td>36.3</td>
</tr>
</tbody>
</table>
Table 3: Scene recognition accuracy (%) with CNN activation

<table>
<thead>
<tr>
<th>Feature</th>
<th>RGB</th>
<th>Depth</th>
<th>RGB-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local (object)</td>
<td>43.5</td>
<td>40.0</td>
<td>46.3</td>
</tr>
<tr>
<td>Global</td>
<td>41.4</td>
<td>41.1</td>
<td>51.5</td>
</tr>
</tbody>
</table>

Table 4: Comparisons on SUN RGB-D in accuracy (%)

<table>
<thead>
<tr>
<th>Method</th>
<th>RGB-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local-OOR</td>
<td>50.3</td>
</tr>
<tr>
<td>Proposed</td>
<td></td>
</tr>
<tr>
<td>Global+Local</td>
<td>52.6</td>
</tr>
<tr>
<td>Global+Local-OOR</td>
<td>54.0</td>
</tr>
<tr>
<td>State-of-the-art</td>
<td></td>
</tr>
<tr>
<td>Song et al. [33]</td>
<td>39.0</td>
</tr>
<tr>
<td>Zhu et al. [43]</td>
<td>41.5</td>
</tr>
<tr>
<td>Wang et al. [39]</td>
<td>48.1</td>
</tr>
<tr>
<td>Song et al. [34]</td>
<td>52.4</td>
</tr>
</tbody>
</table>

Figure 7: Confusion matrix of Global+Local-OOR.

are illustrated in Table 3. With the RGB and depth multi-modal feature fusion, the RGB-D features outperform single-modal features with the gains 2.8%/10.1% of local/global features in accuracy.

6.3.4 Comparison to the state-of-the-art. Different types of feature are combined to compare with the state-of-the-art works [33, 34, 39, 43], and the comparison results are illustrated in Table 4. Some works [33, 43] only depend on the global features, while Wang et al. [39] propose to extract both global features and local features for scene recognition. Although the work of [34] does not explicitly extract local features, the depth model (DCNN) of this work is trained based on local patches with weak-supervision. Somehow, the local information is implicitly included in this work. Based on the object detection results, by concatenating the local feature and OOR representation, our method Local-OOR obtains 50.3% and outperforms the work of [39] more than 2%. Note that the proposed Local-OOR only extracts the local features (OOR) is also a kind of local features), while [39] combines both local and global features. Although the single global feature in Table 3 does not work as well as that in [34], by combining local and global features and concatenating with OOR, our Global+Local-OOR achieves the state-of-the-art result with 54.0%, outperforming [34] with 1.6%, also see Fig. 7 for the confusion matrix of Global+Local-OOR.

Table 5: Comparisons on NYUD2 in accuracy (%)

<table>
<thead>
<tr>
<th>Method</th>
<th>RGB</th>
<th>Depth</th>
<th>RGB-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Global</td>
<td>57.3</td>
<td>54.1</td>
<td>64.0</td>
</tr>
<tr>
<td>Local-OOR</td>
<td>-</td>
<td>-</td>
<td>60.1</td>
</tr>
<tr>
<td>Global+Local-OOR</td>
<td>-</td>
<td>66.9</td>
<td></td>
</tr>
<tr>
<td>State-of-the-art</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gupta et al. [17]</td>
<td>54.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wang et al. [39]</td>
<td>63.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Song et al. [34]</td>
<td>65.8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

that we take fc7 (with 4096 dimension) activation as the global feature, in contrast to the fc8 (with 19 dimension) activation in [34].

6.4 NYUD2

We do not train new CNN models for NYUD2 dataset. Since this dataset contains much less data than SUN RGB-D, we mainly fine tune the pretrained models of SUN RGB-D to the NYUD2. Particularly, the Faster R-CNN models of object detection of SUN RGB-D is directly applied to this NYUD2 dataset. Since using RGB-D features obtains much better performance, we mainly report results of RGB-D features of Local-OOR and Global+Local-OOR. Comparing to other works [17, 34, 39], our proposed Global+Local-OOR achieves the state-of-the-art performance of 66.9%.

7 CONCLUSION

Different scenes contain different object co-occurrences, so representing images with object based intermediate representation may result in ambiguity for scene recognition. By analyzing the limitation of intermediate representation, we propose to represent images with objects and their relative relations, i.e., object-to-object relation (OOR) representation. Particularly, we show that OOR representation is more discriminative than object based representation for the scenes with similar object co-occurrences. Meanwhile, in order to better model the spatial information, the OOR is built on RGB-D data. And the depth data is shown to be helpful for both object detection and scene recognition tasks, especially for the objects with depth in shape. However, more discriminative representations with relative relations , the end-to-end training and predicting framework for scene recognition with OOR are also the goals of our future works.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 61532018 and 61322212, in part by the Beijing Municipal Commission of Science and Technology under Grant D161100001816001, in part by the Lenovo Outstanding Young Scientists Program, in part by National Program for Special Support of Eminent Professionals and National Program for Support of Top-notch Young Professionals. This work was supported by the National Postdoctoral Program for Innovative Talents under Grant BX201700255.
REFERENCES


