Complete 3D Scene Parsing from Single RGBD Image

Chuhang Zou
http://web.engr.illinois.edu/~czou4/
Zhizhong Li
http://zli115.web.engr.illinois.edu
Derek Hoiem
http://dhoiem.cs.illinois.edu

Department of Computer Science
University of Illinois at
Urbana-Champaign
USA

Abstract

Inferring the location, shape, and class of each object in a single image is an important task in computer vision. In this paper, we aim to predict the full 3D parse of both visible and occluded portions of the scene from one RGBD image. We parse the scene by modeling objects as detailed CAD models with class labels and layouts as 3D planes. Such an interpretation is useful for visual reasoning and robotics, but difficult to produce due to the high degree of occlusion and the diversity of object classes. We follow the recent approaches that retrieve shape candidates for each RGBD region proposal, transfer and align associated 3D models to compose a scene that is consistent with observations. We propose to use support inference to aid interpretation and propose a retrieval scheme that uses convolutional neural networks (CNNs) to classify regions and retrieve objects with similar shapes. We demonstrate the performance of our method compared with the state-of-the-art on our new NYUd v2 dataset annotations which are semi-automatically labelled with detailed 3D shapes for all the objects.

1 Introduction

In this paper, we aim to predict the complete 3D models of indoor objects and layout surfaces from single RGBD images. The prediction is represented by detailed CAD objects with class labels and 3D planes of layouts. This interpretation is useful for robotics, graphics, and human activity interpretation, but difficult to produce due to the high degree of occlusion and the diversity of object classes.

Our approach. We propose an approach to recover a 3D model of room layout and objects from an RGBD image. A major challenge is how to cope with the huge diversity of layouts and objects. Rather than restricting to a parametric model and a few detectable object classes, as in previous single-view reconstruction work, our models represent every layout surface and object with a 3D mesh that approximates the original depth image under projection. We take a data-driven approach that proposes a set of potential object regions, matches each region to a similar region in training images, and transfers and aligns the associated labeled 3D models while encouraging their agreement with observations. During the matching step, we use CNNs to retrieve objects of similar class and shape and further incorporate

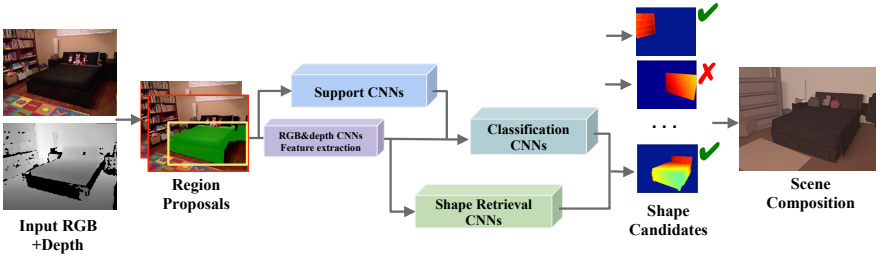

Figure 1: **Overview of our approach.** Given one RGBD image, our approach is to perform complete 3D parsing. We generate layout proposals and object proposals, predict each proposal's support height and class and retrieve a similar object shape. We then select a subset of layout and object shapes to compose the scene with complete 3D interpretation.

support estimation to aid interpretation. We hypothesize, and confirm in experiments, that support height information will help most for interpreting occluded objects because the full extent of an occluded object can be inferred from support height. The subset of proposed 3D objects and layouts that best represent the overall scene is then selected by our optimization method based on consistency with observed depth, coverage, and constraints on occupied space. The flexibility of our models is enabled through our approach (Fig. 1) to propose a large number of likely layout surfaces and objects and then compose a complete scene out of a subset of those proposals while accounting for occlusion, image appearance, depth, and layout consistency.

Detailed 3D labeling. Our approach requires a dataset with labeled 3D shape for region matching, shape retrieval, and evaluation. We make use of the NYUd v2 dataset [18] which consists of 1449 indoor scene RGBD images, with each image segmented and labeled with object instances and categories. Each segmented object also has a corresponding annotated 3D model, provided by Guo and Hoiem [4]. The 3D labeling provides groundtruth 3D scene representation with layout surfaces as 3D planar regions, furniture as CAD exemplars, and other objects as coarser polygonal shapes. However, the polygonal shapes are too coarse to enable comparison of object shapes. Therefore, we extend the labeling by Guo and Hoiem with more detailed 3D annotations in the object scale. Annotations are labeled automatically and are adjusted manually as described in Sec. 3. We evaluate our method on our newly annotated groundtruth. We measure success according to accuracy of depth prediction of complete layout surfaces, voxel occupancy accuracy, and semantic segmentation performance.

We evaluate our method on our newly annotated NYUd v2 dataset. Experiments on object retrieval demonstrate better region classification and shape estimation compared with the state-of-the-art. Our performance of scene composition shows better semantic segmentation results and competitive 3D estimation results compared with the state-of-the-art.

Our **contributions** are:

1. We refine the NYUd v2 dataset with detailed 3D shape annotations for all the objects in each image. The labelling process is performed by a semi-automatic approach and can be utilized for labelling other RGBD scene datasets.
2. We apply support inference to aid region classification in images.
3. We use CNNs to classify regions and retrieve objects with similar shapes;
4. We demonstrate better performance in full 3D scene parsing from single RGBD images compared with the state-of-the-art.

This paper is an extension of our previous work [5] (available only on arxiv) that predicts full 3D scene parsing from an RGBD image. Our main new contributions are refinement of the NYUd v2 dataset with detailed 3D shape annotations, use of CNNs to classify regions and retrieve object models with similar shapes to a region, and use of support inference to aid region classification. We also provide more detailed discussion and conduct more extensive experiments, demonstrating qualitative and quantitative improvement.

2 Related work

The most relavent work to our paper is Guo et al [5] which predicts the complete 3D models of indoor scene from single RGBD image as introduced above. Both our approach and Guo et al. recover complete models from a limited viewpoint, which is in contrast to the multiview whole-room 3D context interpretation method by Zhang et al. [24] that advocate making use of 360° full-view panoramas. We introduce other related topics as follows.

Inferring shape from single RGB (D) Image. Within RGB images, Lim et al. [12, 13] find furniture instances and Aubry et al. [1] recognize chairs using HOG-based part detectors. In RGBD images, Song and Xiao [19] search for main furniture by sliding 3D shapes and enumerating all possible poses. Similarly, Gupta et al. [8] fit shape models of 6 classes with poses to improve object detection. Our approach finds an approximate shape for any object and layout in the scene. We take an exemplar-based approach, apply region-to-region retrieval to transfer similar 3D shape from training region to query region.

Semantic segmentation for single RGBD Images. Silberman et al. [18] use both image and depth cues to jointly segment the objects into 4 categories and infer support relations. Gupta et al. [6] apply both generic and class-specific features to assign 40-class region labels. Their following work encodes both RGB and depth descriptor (HHA) [7]: height above ground, angle with gravity and the horizontal disparity into the CNNs structure for better region classification. Long et al. [15] introduce a fully convolutional network structure for learning better features. Our method make use of the RGB and HHA features to categorize objects into a larger variety of 81 classes to distinguish infrequent objects in the scenes, rather than restricting to a parametric model and a few detectable objects. Though semantic segmentation is not the main purpose of our method, we can infer region labels by projecting the 3D scene models to the 2D images.

Support height estimation. Guo and Hoiem [4] localize the height and full extent of support surfaces from one RGBD image. In addition, object height priors have shown to be crucial geometric cues for better object detection in both 2D [10, 21] and 3D [8, 14, 19]. Deng et al. [3] applies height above ground to distinguish objects. We propose to use object's support height to aid region class interpretation, which helps in classification of occluded regions by distinguishing objects that appear at different height levels: e.g. chair should be on the floor and alarm clock should be on the table.

3 Detailed 3D annotations for indoor scenes

We conduct our experiments on the NYUdv2 dataset [18], which provides complete 3D labeling of both objects and layouts of 1449 RGB-D indoor images. Each object and layout has a 2D segment labeling and a corresponding annotated 3D model, provided by Guo and Hoiem [4]. The 3D annotations use 30 models to represent 6 categories of furniture

Figure 2: **Samples of our semi-automatic 3D object annotations in NYUd v2 dataset**. Images from first to third row: input RGB (D) image, 3D annotations by Guo and Hoiem [4], our refined 3D annotations. Our annotations of the scene is much detailed in object shape scale.

that are most common and use extruded polygons to label all other objects. These models provide a good approximation of object extent but are often poor representations of object shape, as shown in Fig 2. Therefore, we extend the NYUd v2 dataset by replacing the extruded polygons with CAD models collected from ShapeNet [2] and ModelNet [22]. To align name-space between datasets, we manually map all model class labels to the 633-class 3D object labels in NYUd v2 dataset. The shape retrieval and alignment process is performed automatically and then adjusted manually, as follows.

Coarse alignment. For each groundtruth 2D region r_i in the NYUd v2 dataset, we retrieve model set $M = \{M_i\}$ from our collected models that have the same class label as r_i. We also include the region's original coarse 3D annotation by Guo and Hoiem [4] in the model set M, so that we can preserve the original labeling if no provided CAD models are better fit in depth. We initialize each M_i's 3D location as the world coordinate center of the 3D annotation labeled by Guo and Hoiem. We resize M_i to have the same height as the 3D annotation.

Fine alignment. Next, we align each retrieved 3D object model M_i to fit the available depth map of the corresponding 2D region r_i in the target scene. The initial alignment is often not in the correct scale and orientation; e.g., a region of a left-facing chair often resembles a right-facing chair and needs to be rotated. We found that using Iterative Closest Point to solve for all parameters did not yield good results. Instead, we enumerate 16 equally-spaced orientations from -180 to 180 from top-down view and allows 2 minor scale revision ratio as $\{1.0, 0.9\}$. We perform ICP to solve for translation initialized using scale and rotation, and pick the best ICP result based on the following cost function:

$$\begin{aligned}
&\text{FittingCost}(M_i, T_i) = \\
&C_{depth} \sum_{j \in r_i \cap s(M_i,T_i)} |\mathcal{I}_d(j) - \hat{d}(j; M_i, T_i)| \\
&+ \sum_{j \in r_i \cap \neg s(M_i,T_i)} C_{missing} \\
&+ C_{occ} \sum_{j \in \neg r_i \cap s(M_i,T_i)} \max(\hat{d}(j; M_i, T_i) - \mathcal{I}_d(j), 0)
\end{aligned} \tag{1}$$

where T_i represents scale, rotation, and translation, $s(.)$ is the mask of the rendered aligned object, $\mathcal{I}_d(j)$ denotes the observed depth at pixel j and $\hat{d}(j)$ means the rendered depth at j. The first term encourages depth similarity to the groundtruth RGBD region; the second

penalizes pixels in the proposed region that are not rendered, and the third term penalizes pixels in the rendered model that are closer than the observed depth image (so the model does not stick out into space known to be empty).

Based on the fitting cost of Eq. 1, our algorithm picks the model M_i with the best translation, orientation, and scale T_i. We set the term weights $C_{depth}, C_{missing}, C_{occ}$ as $1.0, 0.9, 0.5$ based on a grid search in the validation set. For efficiency, we first obtain the top 5 models based on the fitting cost, each maximized only over the 16 initial orientations before ICP. For each of these models, we then solve for the best translation T_i for each scale and rotation based on Eq. 1 and finally select the aligned model with the lowest fitting cost.

Post-processing. Automatic fitting may fail due to high occlusion or missing depth values. We manually conduct a post-processing check and refine bad-fitting models, which affects roughly 10% of models. Using a GUI, an annotator checks the automatically produced shape for each region. If the result is not satisfactory, the user compares to other top model fits, and if none of those are good matches, then the fitting optimization based on Eq. 1 is applied to the original polygonal 3D labeling. This helps to ensure that our detailed shape annotations are a strict improvement over the original course annotations.

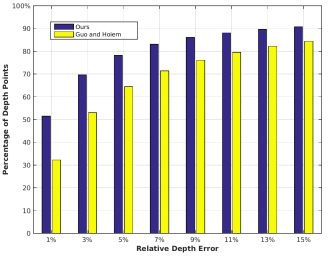

Figure 3: Cumulative relative depth error of our detailed 3D annotations and the 3D annotations by Guo and Hoiem [4] in NYUd v2 dataset.

Validation. Figure 3 reports the cumulative relative error of the rendered depth of our detailed 3D annotations compared with groundtruth depth in NYUd v2 dataset. The relative error r_D is computed as:

$$r_D = \frac{1}{|S_I|} \sum_{I \in S_I} \sum_{p \in I} \frac{|d_p - \hat{d}_p|}{d_p} \tag{2}$$

where $S_I = \{I_1, I_2, \ldots, I_N\}$ is all the RGBD images in the dataset; p represents a pixel in each image I; d_p is the groundtruth depth of pixel p from sensor; and $\hat{d}_p$ is the rendered depth of the 3D label annotation at pixel p. For comparison, we report the r_D of the 3D annotations by Guo and Hoiem [4]. Our annotations have more points with low relative depth error, and achieve a better modeling of depth for each image.

4 Generating object candidates

Given an RGBD image as input, we aim to find a set of layout and object models that fit RGB and depth observations and provide a likely explanation for the unobserved portion of the scene. To produce object candidate regions, we use the method for RGBD images by Gupta et al. [7] and extract top ranked 2000 region proposals for each image. Our experiments show that this region retrieval is more effective than the method based on Prims algorithm [16] used in our previous work [5]. Likely object categories and 3D shapes are then assigned to each candidate region.

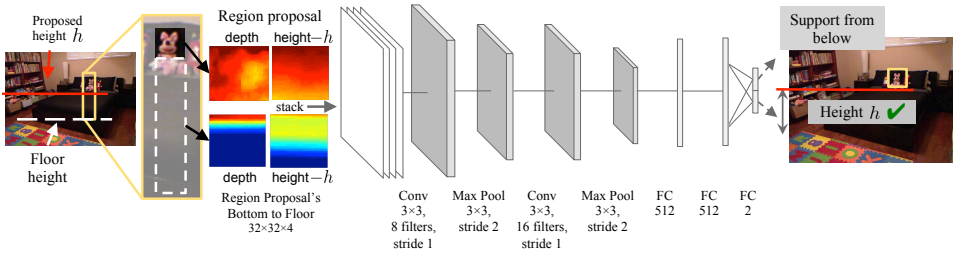

Figure 4: The CNN for predicting a candidate object's support height. We perform ReLU between the convolutional layer and the max pooling layer. Local response normalization is performed before the first fully connected (FC) layer. We add dropout with 0.5 before each FC layer during training.

4.1 Predicting region's support height and class

We train and use CNN networks to predict the object category and support height of each region, as shown in Fig. 4 and Fig. 5. The support height is used as a feature for the object classification. We also train and use a CNN with a Siamese network design to find the most similar 3D shape of a training object, based on region and depth features.

Support height prediction. We predict support height for each object with the aim of better predicting class and position. We first find candidate support heights using the method of Guo and Hoiem [4] and use a CNN to estimate which is most likely for a given object region based on crops of the depth maps and height maps of the region proposal and region that extends from bottom of the region proposal to the estimated floor position. The support height network also predicts whether the object is supported from below or behind. To create the feature vector, we subtract the candidate support height from the height cropped images, re-size all four crops to 32×32 patches, and concatenate them, as illustrated in Fig. 4.

In the test set, we identify the closest candidate support height with 92% accuracy, with an average distance error of 0.18m. As a feature for classification, we use the support height relative to the camera height, which leads to slightly better performance than using support height relative to the estimated ground likely because dataset images are taken from consistent heights but estimated ground height may be mistaken.

Categorization. Our classification network gets input of the region proposal's support height and type, along with CNN features from both RGB and depth. The network predicts the probability for each class as shown in Fig. 5. To model the various classes of shapes in indoor scene, we classify the regions into the 78 most common classes, which have at least 10 training

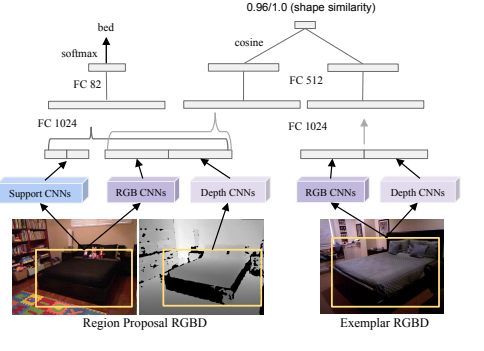

Figure 5: The CNNs for region classification (left) and similar shape retrieval (right). We perform ReLU and dropout with 0.5 after the first FC layer for both of the networks during training.

samples. Less common objects are classified as “other prop", “other furniture" and “other structure" based on rules by Silberman et al. [18]. In addition, we identify a region proposal that is not representative for an object shape (e.g. a piece of chair leg region when the whole chair region is visible) as a “bad region" class. This leads to our $78+3+1=82$-class classifier. The input support height and type are directly predicted by our support height prediction network. To create our classification features, we copy the two predicted support values 100 times each (a useful trick to reduce sensitivity to local optima for important low-dimensional features) and concatenate them to the region proposal's RGB and HHA features from Gupta et al. [7] in both the 2D bounding box and masked region as in [9]. Experiments show that using the predicted support type and the support height improves the classification accuracy by about 1%, with larger improvement for occluded objects.

4.2 Predicting region's 3D shape

Using a Siamese network (Fig. 5), we learn a region-to-region similarity measure that predicts 3D shape similarity of the corresponding objects. The network embeds the RGB and HHA features used in our classification network into a space where cosine distance correlates to shape similarity, as in [23]. In training, we use surface-to-surface distance [17] between mesh pairs as the ground truth similarity and train the network to penalize errors in shape similarity orderings. Each region pair's shape similarity score is compared with the next pair's among the randomly sampled batch in the current epoch and penalized only if the ordering disagrees with the groundtruth similarity. We attempted sharing embedding weights with the classification network but observed a 1% drop in classification performance. We also found predicted class probability to be unhelpful for predicting shape similarity.

Candidate region selection. We apply the above retrieval scheme to each of the 2000 region proposals in each image, obtaining shape similarity rank compared with all the training samples and 81-object class and non-object class probability for each region proposal. In order to reduce the number of retrieved candidates before the scene interpretation in Sec. 5, we first reduce the number of region proposals using non-maximal suppression based on non-object class probability and threshold on the non-object class probability. We set the threshold to obtain 190 region proposals for each image, on average. We select the two most probable classes for each remaining region proposal and select five most similar shapes for each class, leading to 10 shape candidates for each region proposal.

Then, we further refine these 10 retrieved shapes. We align each shape candidate to the target scene by translating the model using the offset between the depth point mass centers of the region proposal and the retrieved region. We then perform ICP, using a grid of initial values for rotation and scale and pick the best one for each shape based on the fitting energy in Eq. 1. We set the term weight: $C_{missing}$ as 0.6, C_{depth} as 1.0, C_{occ} as 0.9 based on a grid search on the validation set. We tried using the estimated support height for each region proposal for aligning the related 3D shape models but observed a worse performance in the scene composition result. This is because a relatively small error in object's support height estimation can cause a larger error in fitting.

Finally, we select the two most promising shape candidates based on the following energy function,

$$\begin{aligned} E_l(m_i) =& w_f E_{fitting}(m_i,t_i) \\ &+ w_c \log P(c_i|r_i) + w_b \log P(b_i|r_i) \end{aligned} \tag{3}$$

$E_{fitting}$ is the fitting energy defined in Eq. 1 that we used for alignment. $P(c_i|r_i)$ and $P(b_i|r_i)$ are the softmax class probability and the non-object class probability output by our classification network for the regon proposal r_i. We normalize $P(c_i|r_i)$ to sum to 1, in order not to penalize the non-object class twice in the energy function. We set the term weights $w_f = 1.0$, $w_c = -1500$, $w_b = 1300$ using a grid search. Note that $E_{fitting}$ is on the scale of the number of pixels in the region.

4.3 Training details

We first find meta-parameters on the validation set after training classifiers on the training set. Then, we retrain on both training and validation in order to report results on the test set. We train our networks with the region proposals that have the highest (and at least 0.5) 2D intersection-over-union (IoU) with each groundtruth region in the train set. We train the support height prediction network with the groundtruth support type for the region proposal and set the groundtruth support height as the closest support height candidate that is within 0.15 meters from the related 3D annotation's bottom. For training regions that are supported from behind, we do not penalize the support height estimation, since our support height candidates are for vertical support. For training the classification network, we also include the non-object class region proposals that have < 0.3 IoU with the groundtruth regions. We randomly sample the same number of the non-object regions as the total number of the object regions during training. To avoid unbalanced weights for different classes, we sample from the dataset the same number of training regions for each class in each epoch. When training the shape similarity network, we translate each 3D model to origin and resize to $200 \times 200 \times 200$-voxel cuboid before computing the surface-to-surface distance. We use ADAM [11] to train each network with the hyper-parameter of $\beta_1 = 0.9$, $\beta_2 = 0.999$. The learning rate for each network is: support height prediction 0.0008, classification 0.003 and Siamese network 0.0001.

5 Scene composition with candidate 3D shape

Given a set of retrieved candidate 3D shapes and the layout candidates from Guo et al. [5], we select a subset of the candidates that closely reproduce the original depth image when rendered, correspond to minimally overlapping 3D aligned models. The composition is hard because of the high degree of occlusions and large amount of objects in the scene. We apply the same method proposed by Guo et al. [5] to perform scene composition, this leads to our final scene interpretation.

6 Experiments

6.1 Experiments setting

We evaluate our method on our detailed 3D annotated NYUd v2 dataset. We also report the result of the state-of-the-art by Guo et al. [5] on our new annotations. Same as Gupta et al. [7], we tune the parameters on the validation set while training on the train set; we report our result on the test set by training on both the train and validation set.

Table 1: **Our 81-class classification accuracy on groundtruth 2D regions in the test set.** We compare two methods: our classification network with/without estimating support height. We compute the average accuracy for each class, average precision based on the predicted probability and the accuracy averaged over instances. The classification networks are trained and evaluated 10 times and the means and standard deviations (reflecting variation due to randomness in learning) are reported. 15 common object class results are also listed. Bold numbers signify better performance.

Method	avg per class	avg precision	avg over instance	picture	chair	cabinet	pillow	bottle	books	paper	table	box	window	door	sofa	bag	lamp	clothes
w/o support height	43.7± 0.3	37.7 ± 0.1	40.8± 0.3	57.5	46.1	39.5	66.5	**55.6**	30.0	40.7	36.7	10.6	61.4	54.9	63.0	14.9	64.6	25.3
w/ support height	**44.7± 0.3**	**39.7±0.1**	**42.7±0.2**	57.5	**53.1**	**44.35**	**69.0**	54.9	**33.5**	**43.5**	**39.5**	**14.1**	**62.4**	**57.8**	**65.9**	**15.1**	**65.9**	**26.9**

Table 3: Quantitative evaluation for our retrieval method compared with Guo et al's method.

Method	avg class accuracy (%)			avg 3D IoU			avg surface distance (m)		
	top 1	top 2	top 3	top 1	top 2	top 3	top 1	top 2	top 3
Guo et al. [5]	11.97	16.36	19.65	0.134	0.177	0.200	0.033	0.029	0.027
Ours	**41.56**	**54.47**	**62.07**	**0.191**	**0.231**	**0.249**	**0.026**	**0.024**	**0.023**

6.2 Evaluation of region classification and shape retrieval

Classification. We report the region classification accuracy on the groundtruth 2D regions in the test set by our classification network, as shown in Table 1. Overall, including support height prediction will improve the classification results. For certain classes, objects often appear on the floor level (e.g. chair, desk) will have better classification accuracy given object's height, while objects often appear on several heights (e.g. picture) do not have this benefit.

Table 2: Classification accuracy under different occlusion ratios of groundtruth regions in test set

Occlusion Ratio	< 0.5	> 0.5
w/ support height	45.8	38.5
w/o support height	44.2	35.7

Table. 2 shows the per instance classification accuracy under different occlusion ratios of the groundtruth regions in the test set. The improvement in classification accuracy is larger for highly occluded area, which conforms with our claim that estimating object's support height will help classify occluded regions.

Retrieval. We evaluate our candidate shape retrieval method compared with the state-of-the-art by Guo et al. [5] given groundtruth 2D regions in the dataset, as shown in Table 3. Since we use groundtuth regions, we do not perform candidate selection in Sec. 4.2 for this evaluation. We evaluate 1) top N retrieved class accuracy and 2) top N retrieved shape similarity, based on the shape intersect over union (IoU) and the surface-to-surface distance. In our experiment, we set $N = \{1,2,3\}$. To avoid rotation ambiguity in shape similarity measurement, we rotate each retrieved object to find the best shape similarity score with the groundtruth 3D shape. Our retrieval method outperforms the state-of-the-art under all the evaluation criteria.

6.3 Evaluation of scene composition

84-class semantic segmentation. We evaluate the 84-class semantic segmentation (81 object classes and 3 layout classes: wall, ceiling, floor) on the rendered 2D image of our scene composition result. We compare our method with Guo et al. [5] with both automatically generated region proposals and groundtruth regions. Table 4 shows the average class accuracy (avacc), average class accuracy weighted by frequency (fwavacc) and the average pixel ac-

Table 4: Results of 84-class semantic segmentation on both automatic region proposals and groundtruth regions.

(a) Overall results

Method	Mean coverage weighted	Mean coverage unweighted
Guo et al. [8]	48.12	29.75
MCG+CCA	50.43	32.91
Ours w/o support	50.43	34.34
Ours	50.23	33.92
Ours w/ GT support	50.52	34.32
Ours w/ GT-region	52.39	35.61
Detailed groundtruth labeling	65.42	49.15

(b) Each class's w/ support accuracy minus w/o support accuracy with automatic region proposals

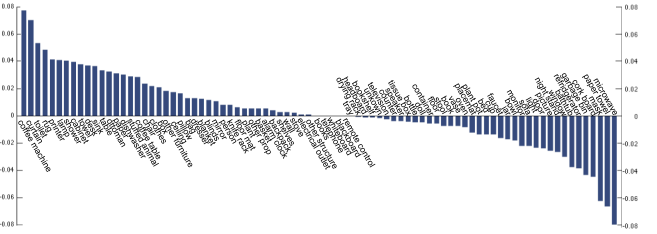

Table 5: Results of 84-class instance segmentation on both automatic region proposals and groundtruth regions.

Method	Mean coverage weighted	Mean coverage unweighted
Guo et al. [8]	48.12	29.75
MCG+CCA	50.43	32.91
Ours w/o support	50.43	34.34
Ours	50.23	33.92
Ours w/ GT support	50.52	34.32
Ours w/ GT-region	52.39	35.61
Detailed groundtruth labeling	65.42	49.15

curacy (pixacc). Our method has a better semantic segmentation result. We report our result with/without support height prediction and the result given the groundtruth support height as the upper bound. We see improvements, compared to Guo et al., due to both classification method (CNN vs. CCA) and region proposal method (MCG by Gupta et al. vs. prim's by Guo et al.). We also report the 40-class semantic segmentation compared with the state-of-the-art [15]: 47.7 pixacc / 18.3 avacc / 33.8 fwavacc vs. their 65.4/34.0/49.5. Note that our method focuses on predicting 3D geometry rather than semantic segmentation.

84-class instance segmentation. We report the instance segmentation result with the experiment setting same as semantic segmentation. The evaluation follows the protocol in RMRC [20]. Our method outperforms the state-of-the-art slightly. Note that the lower results for MeanCovU is caused by the large diversity of classes with less frequency.

3D estimation. We evaluate 3D estimation as in Guo et al. [8] with layout pixel-wise and depth prediction and freespace and occupancy estimation. As shown in Fig. 6, our method has competitive results in 3D estimation compared with the state-of-the-art.

Qualitative results. Sample qualitative results are shown in Fig. 6 and Fig. 7. Our method has better region classification and shape estimation property, which results in a slightly better scene composition result. Failure cases are caused by bad pruning in region proposals (last two column, Fig. 6) and confusion between similar class (top right, Fig. 7).

7 Conclusions

In this paper, we predict the full 3D parse of both visible and occluded portions of the scene from one RGBD image. We propose to use support inference to aid interpretation and pro-

Table 6: Results of 3D layout, freespace and occupancy estimation.

Method	Layout Pixel Error			Layout Depth Error			Freespace		Occupancy			
	overall	visible	occluded	overall	visible	occluded	precision	recall	precision	recall	precision-ϵ	recall-ϵ
Guo et al. [8]	10.9	14.0	4.8	0.166	0.204	0.074	0.954	0.914	**0.504**	0.380	**0.751**	0.646
Ours	**10.6**	**13.6**	4.8	**0.15**	**0.181**	0.074	0.954	**0.919**	0.478	**0.397**	0.741	**0.710**

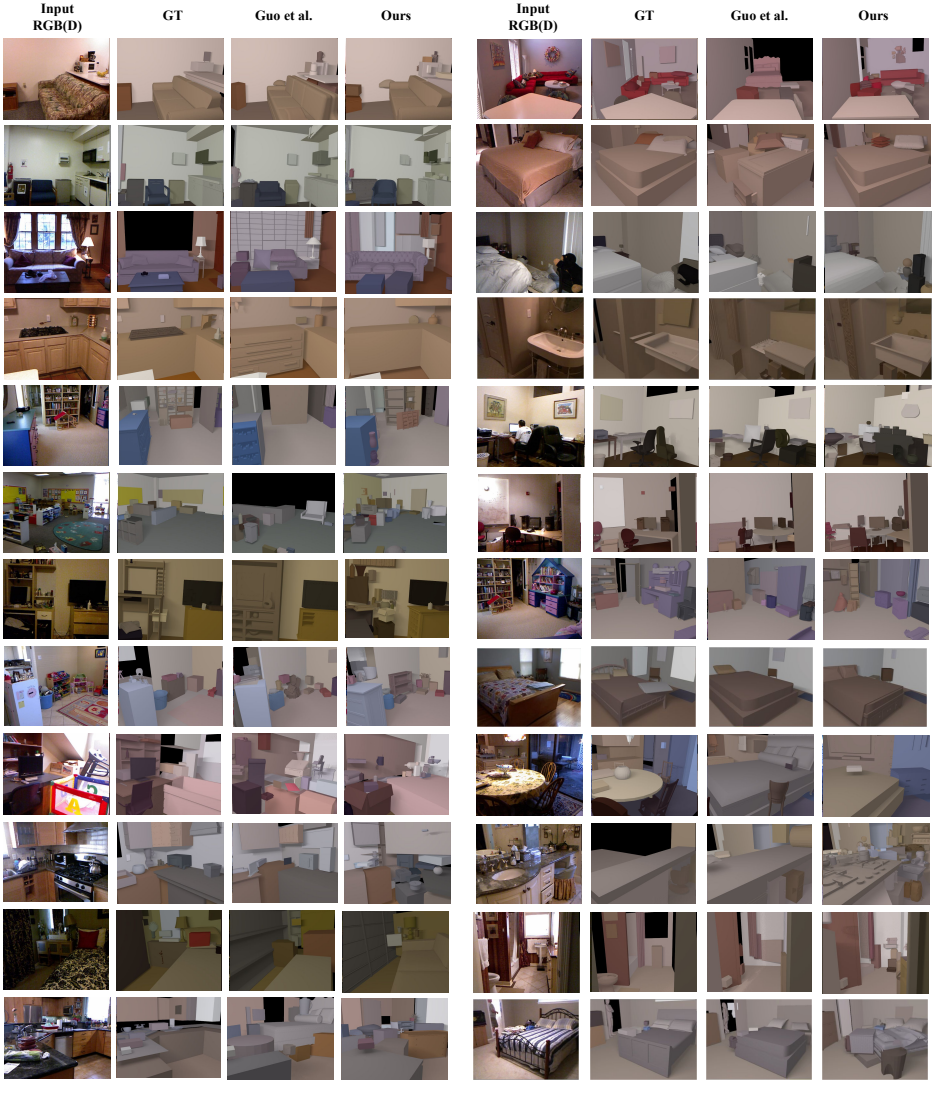

Figure 6: Qualitative results on scene composition with automatic region proposals. We randomly sample images from the top 25% (first four rows), medium 50% (row 6-9) and worst 25% (last four rows) based on 84-class semantic segmentation accuracy.

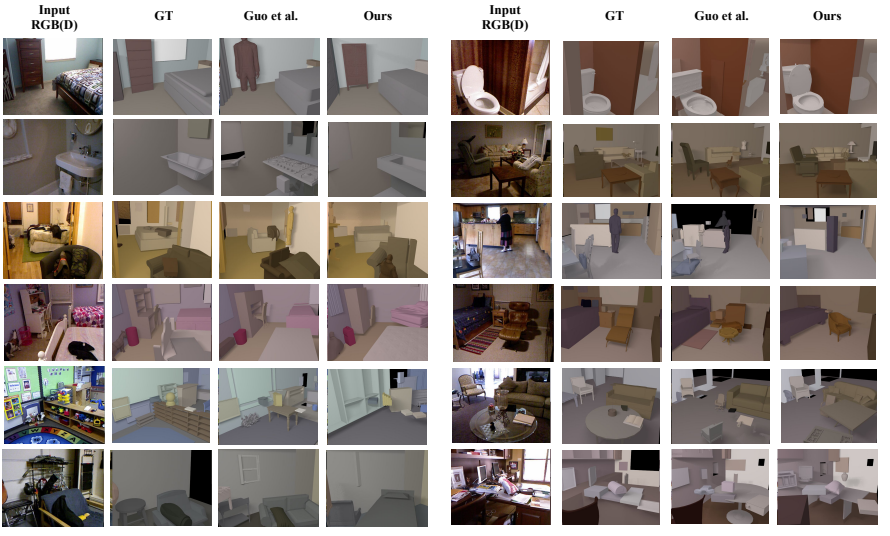

Figure 7: Qualitative results on scene composition given groundtruth 2D labeling as region proposals. We randomly sample images from the top 25% (first two rows), medium 50% (row 3-4) and worst 25% (last two rows) based on 84-class semantic segmentation accuracy.

pose a retrieval scheme that uses CNNs to classify regions and find objects with similar shapes. Experiments demonstrate better performance of our method in semantic segmentation and instance segmentation and competitive results in 3D scene estimation.

Acknowledgements

This research is supported in part by ONR MURI grant N000141010934 and ONR MURI grant N000141612007. We thank David Forsyth for insightful comments and discussion.

References

[1] M. Aubry, D. Maturana, A. A. Efros, B. C. Russell, and J. Sivic. Seeing 3d chairs: Exemplar part-based 2d-3d alignment using a large dataset of cad models. In *CVPR*, 2014.

[2] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[3] Zhuo Deng, Sinisa Todorovic, and Longin Jan Latecki. Semantic segmentation of rgbd images with mutex constraints. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1733–1741, 2015.

[4] Ruiqi Guo and Derek Hoiem. Support surface prediction in indoor scenes. In *ICCV*, 2013.

[5] Ruiqi Guo, Chuhang Zou, and Derek Hoiem. Predicting complete 3d models of indoor scenes. *arXiv preprint arXiv:1504.02437*, 2015.

[6] Saurabh Gupta, Pablo Arbelaez, and Jitendra Malik. Perceptual organization and recognition of indoor scenes from RGB-D images. In *CVPR*, 2013.

[7] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision*, pages 345–360. Springer, 2014.

[8] Saurabh Gupta, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Aligning 3d models to rgb-d images of cluttered scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4731–4740, 2015.

[9] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *Computer vision–ECCV 2014*, pages 297–312. Springer, 2014.

[10] Derek Hoiem, Alexei A Efros, and Martial Hebert. Putting objects in perspective. *International Journal of Computer Vision*, 80(1):3–15, 2008.

[11] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] Joseph J Lim, Hamed Pirsiavash, and Antonio Torralba. Parsing IKEA objects: Fine pose estimation. In *ICCV*, 2013.

[13] Joseph J Lim, Aditya Khosla, and Antonio Torralba. Fpm: Fine pose parts-based model with 3d cad models. In *European Conference on Computer Vision*, pages 478–493. Springer, 2014.

[14] Dahua Lin, Sanja Fidler, and Raquel Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. In *ICCV*, 2013.

[15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[16] Santiago Manen, Matthieu Guillaumin, and Luc Van Gool. Prime object proposals with randomized prim's algorithm. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2536–2543, 2013.

[17] Jason Rock, Tanmay Gupta, Justin Thorsen, JunYoung Gwak, Daeyun Shin, and Derek Hoiem. Completing 3d object shape from one depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2484–2493, 2015.

[18] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. *Computer Vision–ECCV 2012*, pages 746–760, 2012.

[19] Shuran Song and Jianxiong Xiao. Sliding shapes for 3d object detection in depth images. In *European conference on computer vision*, pages 634–651. Springer, 2014.

[20] R Urtasun, R Fergus, D Hoiem, A Torralba, A Geiger, P Lenz, N Silberman, J Xiao, and S Fidler. Reconstruction meets recognition challenge, 2013.

[21] Stefan Walk, Konrad Schindler, and Bernt Schiele. Disparity statistics for pedestrian detection: Combining appearance, motion and stereo. In *Computer Vision–ECCV 2010*, pages 182–195. Springer, 2010.

[22] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015.

[23] Wen-tau Yih, Kristina Toutanova, John C Platt, and Christopher Meek. Learning discriminative projections for text similarity measures. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 247–256. Association for Computational Linguistics, 2011.

[24] Yinda Zhang, Shuran Song, Ping Tan, and Jianxiong Xiao. Panocontext: A whole-room 3d context model for panoramic scene understanding. In *European Conference on Computer Vision*, pages 668–686. Springer, 2014.