Dual Graphical Models for Relational Modeling of Indoor Object Categories

Lin Guo, Guoliang Fan, and Weihua Sheng
School of Electrical and Computer Engineering
Oklahoma State University, Stillwater, OK, USA
{lin.guo, guoliang.fan, weihua.sheng}@okstate.edu

Abstract

There are three levels for indoor scene understanding, pixel level labeling, object level recognition and scene level holistic understanding. The three levels provide complementary bottom-up scene representation. Traditional research often addresses these three tasks separately where the three levels of semantic data are seldom jointly considered. We propose a new method to bridge the three semantic levels by using dual graphical models for relational modeling of object categories in indoor scenes. The vertical placement model captures top-down object configuration by which the visible pixels of some accessory objects could be used to infer the presence of a supportive object underneath. The horizontal placement model reveals how multiple object categories are related to each other on the ground in different indoor scenes. The experimental results show improvements on the bounding box accuracy using both vertical and horizontal placement models from pixel level labeling.

1. Introduction

Indoor scene understanding has been a challenging problem in computer vision because of large variation in object shapes and placement, and heavy occlusion and clutter. There exist three main schemes in scene understanding: per-pixel semantic labeling [20, 16, 22, 7, 14], bounding box generation [16, 32, 27, 24], and scene level holistic understanding [1, 19, 30, 2, 28, 18, 17, 4, 12, 24, 23, 11]. The first scheme provides the contours and spatial areas of different objects and structures in an image. The second shows a set of cuboid-shaped boxes to represent different objects with certain size and orientation in . The third scheme includes various scene level tasks, including room type recognition [1, 19], scene structure classification [30, 2, 28] or other methods that incorporates high level knowledge from human understanding [18, 17, 4, 12]. There are two kinds of ground truth data used for training and validation, bounding boxes or pixel labeling. Usually, the former can be used to represent objects with relatively well-defined shapes, while the latter is more general and suitable for various objects or structures. Thus, more object categories are normally considered in pixel level labeling than those used for bounding box generation. On the other hand, there is a trend to combine both of them for scene understanding [24, 23, 11]. However, there are some gaps between 2D pixel labels and 3D bounding boxes, both spatially and relationally.

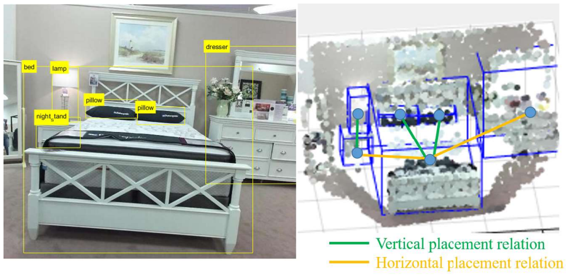

Figure 1. Given a cluttered scene with many objects (left), the object relationships are encoded by two graphical models, called the vertical and horizontal placement models (VPM and HPM).

With the rapid development of deep learning [15, 9] and a vast amount of indoor RGB-D data, the performance of pixel level scene analysis has been improved significantly in recent years. On the other hand, 3D bounding boxes provide object level understanding that may be preferred in practice. We are interested in creating reliable and accurate 3D bounding boxes from the depth data with pixel level labels. In this paper, we propose dual graphical models to bridge the gap between two descriptors by capturing object placement dependency both horizontally and vertically. As shown in Fig. 1, the vertical placement model (VPM) captures co-existence of major and accessory objects from the top-down view, while the horizontal placement model (HPM) represents ground level spatial configuration of different objects. Specifically, VPM improves bounding box generation for major objects from accessories, such as a pillow on a bed, and HPM is applied to enhance bounding boxes for surrounding small objects, like a nightstand beside a bed. The sequential application of VPM and HPM allows us to fully utilize pixel level labels to produce reliable 3D bounding boxes in a local-to-holistic way.

2. Related work

We will briefly review indoor scene understanding from three perspectives, *pixel level labeling*, *bounding box generation*, *scene level representation*.

Per-pixel semantic labeling is widely studied [20, 16, 22, 7, 14] for its convenience and efficiency in working with each pixel in RGB images. Due to the limited information contained in each single pixel, efforts have been made to add holistic knowledge for pixel level labeling. For example, an image is segmented into equal-sized cells for labeling [25]. The norm distribution of RGB-D data at the pixel level is used for object recognition [29]. The distribution along the gravity direction is considered during pixel level labeling in 3D space [17].

Bounding boxes are effective and intuitive to show the 2D or 3D range for each object [16, 32, 24], but they also lack of details. Researchers have been trying to use a cuboid-shaped boxes to represent objects and preserve the detailed object shape information at the same time [27, 23]. A two-step approach was proposed in [27] that combines objectness estimation and object recognition for bounding box generation. Some approaches [23] generate 3D bounding boxes by cutting out irrelevant 3D points according to re-projected 2D bounding boxes. The ground truth data of bounding boxes are provided independently with that of pixel level labeling, making them lack consistency and compatibility. Some studies tried to find an intermediate representation to present the scene both holistically and in a detailed way. Approaches include using planes [31, 3, 28], cuboid [11, 8] or other geometry primitives [21] as prior shapes to represent indoor objects. However, the geometrical representation is applied to the whole scene without indicating object categories or instance level segmentation. Bounding boxes were created from pixel level labeling for a fully registered 3D point cloud [32] with little occlusion.

Scene level understanding usually involves holistic prior knowledge about the scene. For example, some methods focus on the perpendicular aspects of indoor rooms and furniture [33, 28]. Some algorithms extract indoor structures to find the general room configuration [10, 31, 3]. To find and localize all the objects in the whole scene, some pre-defined scene templates were used as prior knowledge for directed local search [13]. Using a graphical model, the spatial occurrence pattern among objects in 2D images is captured to improve the object detection rate collectively [4, 5].

In this work, we investigate in the object placement and relationship in 3D space at both the object level and scene level with the help of two complementary graphical models. Specifically, VPM works locally at the object level to improve bounding box generation by inferring invisible parts from visible ones, while HPM is used at the scene level holistically and collectively to improve bounding boxes for all objects in a scene-specific group.

3. Dual Graphical Models

Our objective is to integrate the three-level scene semantics in a bottom-up information flow where the two models, VPM and HPM, play complementary roles to bridge the gap among three semantic levels. VPM serves as a *bridge* from pixel level labeling to bounding box initialization, and HPM plays as a *propagator* to use scene-level holistic configuration for collective bounding box generation .

3.1. Challenges and approaches

There are two challenges in this research. The first one is about the placement between objects that often leads to occlusion and overlap problems, complicating bounding box generation. For example, a chair under a table is only partially visible and a pillow or sheet will cover part of the bed unlabeled. The second one is about the ambiguity and inconsistency of ground-truth data used for pixel labeling. There are two often-seen cases. The first is that different objects share the same label, for example, the nightstand and end-table were often considered to belong to the same category. The second is the same object was labeled differently. For example, some beds were labeled with a bed board, and some only include the mattress. Therefore, we involve two graphical models that capture the objective placement dependency to cope with those challenges with the aim to create reliable and accurate bounding boxes from inconsistent and ambiguous pixel-level labels. Traditionally a graphical model is generated considering the co-existence probability of objects as the edge weight S, as shown below.

$$p(n_1, ...n_k) = \prod_{i=1,2...k} S(n_i, M_i), \tag{1}$$

where

$$S(n_i, M_i) = p(n_i|M_i),$$

where n_i is the node for object i in the graph, M_i is the set of parents of node n_i, k is the total node number, and S stands for the edge weight calculated by the co-occurrence joint probability. In the following, we introduce the dual graphical models, VPM and HPM, that involve different weights as the closeness measure and similar training data.

3.2. Vertical placement model (VPM)

Given the ground-truth pixel labels and bounding boxes, we study vertical placement modeling by projecting all objects onto the ground plane from the top-down view. Then a 2D room layout is obtained by aligning all objects with gravity. Small ones are often placed on top of the bigger ones. Due to the fact that some small objects could be placed on different objects, we learn VPM with strong pairwise connections by trimming off the weak ones [4, 5]. In VPM, the nodes are object categories from pixel level labeling and the edges' weights are determined by the closeness

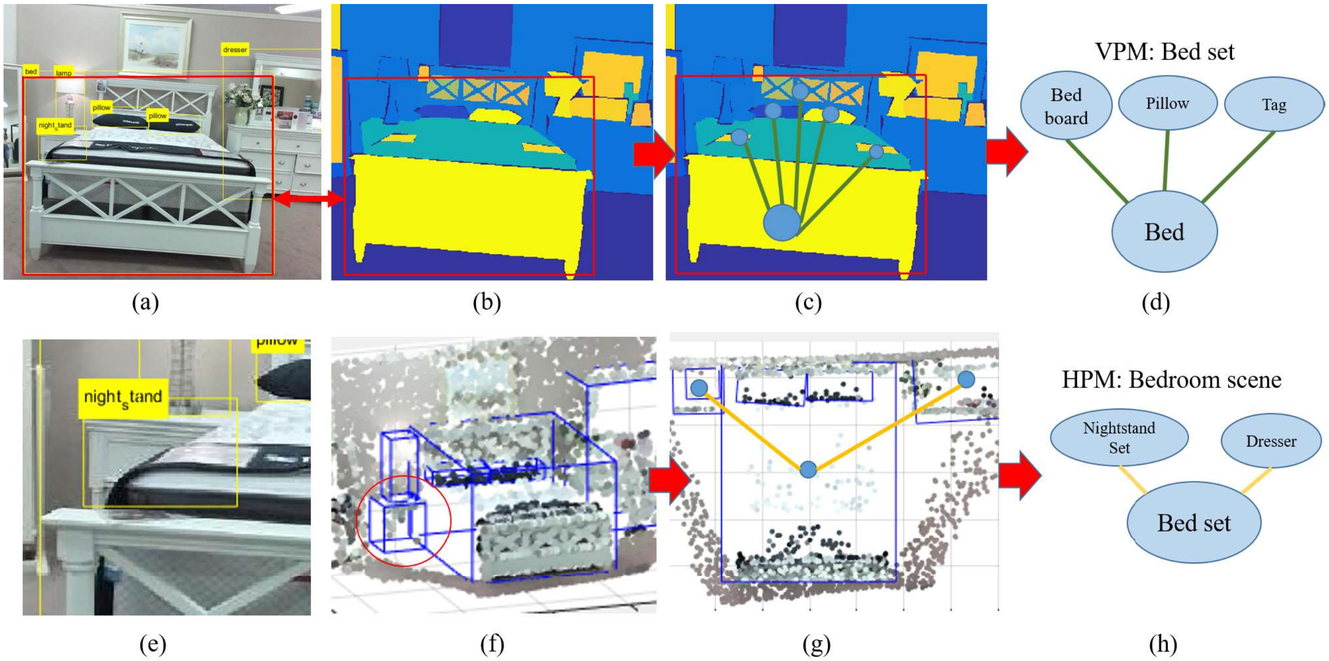

Figure 2. Illustration of VPM and HPM. (a) A bedroom image. (b) Ground-truth pixel level labeling of (a) where the bed is labeled as multiple items. (c) The vertical placement relationship of the bed set. (d) VPM for the bed set. (e) The cropped portion of the nightstand in (a). (f) Ground-truth bounding boxes in 3D space. (g) The horizontal object placement from the top-down view. (h) HPM for the bedroom.

measure that considers their co-occurrences and the overlap ratio in the layout view. Therefore, VPM is specified as,

$$g_{\mathcal{V}}(n_1, ...n_k) = \prod_{i=1,2...k} S_v(n_i, M_i), \tag{2}$$

where the edge weight is

$$S_v(n_i, m_i) = p(n_i|m_i) \times \frac{A(n_i)}{A(m_i)}, m_i \in M_i, \tag{3}$$

where $A(n_i)$ is the mean area of object n_i in the layout view. Thus, the edge weight S_v is calculated using the product of conditional probability and the 2D overlapping ratio between the two objects in layout view, which indicates the significance in the bounding box.

3.3. Horizontal placement model (HPM)

Similar to VPM, HPM is also learned from the top-down layout view of the projected 3D objects which embraces all object categories. The edge weight S_h is based on the co-occurrence probability of a pair of objects and the ratio of their center distance D with reference to their non-overlapping minimum distance B. For the objects without bounding boxes, the distance ratio is set to be one. HPM is defined as

$$g_{\mathcal{H}}(n_1, ...n_k) = \prod_{i=1,2...k} S_h(n_i, M_i), \tag{4}$$

where the edge weight

$$S_h(n_i, m_i) = p(n_i|m_i) \times \frac{D(n_i)}{B(n_i, m_i)}, m_i \in M_i, \tag{5}$$

which is the product of conditional probability and the 2D distance ratio in layout view between the center distance D and B. The non-overlapping minimum distance B is given by the summation of bounding box half size on the short side. Long distanced objects are encouraged due to the fact that distanced objects may exist in the scene outside of the image vision range, which lowers their presence probability.

3.4. Model Learning

We learn the two models from the fully labeled dataset including two kinds of ground-truth data, i.e., pixel-level labeling and 3D bounding boxes. Specifically, the VPM learning involves both ground-truth data, whereas the HPM learning only uses bounding box ground-truth. We follow the framework in [5] and use the Chow-Liu algorithm [6] to maximize the likelihood of the training data (i.e., $g_{\mathcal{V}}(\cdot)$ for VPM and $g_{\mathcal{H}}(\cdot)$ for HPM). Firstly, the algorithm computes edge weights (S_v and S_h) as the mutual information for each object pair. Then, it finds the maximum weight spanning tree with the calculated edge weights. This algorithm only keeps strong pairwise information to generate an undirected graphical model.

3.5. Applications of Dual Models

Figures 3 and 4 illustrate VPM and HPM, respectively, which are learned from the ground-truth data (pixel labeling and bounding boxes) in the SUNRGBD dataset [26]. The two models are versatile for different scene analysis tasks. (1) According to VPM, indoor objects can be classified into three groups, ground-level *base objects*, *accessory objects* placed on a base object, and stand-alone individual objects. (2) We can find the co-existence and exclusiveness between every object pair in both VPM and HPM. (3) The grouping effect in two models indicate different object sets and room types (denoted by dash ovals). (4) VPM and HPM can be used to refine and rectify ground truth data where the inconsistency and ambiguity may impede training and testing.

4. Bounding box generation from pixel labeling

As a case study in this work, we will apply VPM and HPM to create 3D bounding boxes for all objects from pixel-level labeling results obtained from any deep learning algorithm. The VPM and HPM work together to bridge the gap between pixel-level labels and object-level bounding boxes in a sequential manner.

4.1. Bounding box initialization

Given a pixel-level segmentation map, we transfer the class label to the RGB-depth data points and then the labeled 3D points are projected from a top-down view to form a layout map, represented by L, similar to the way we created training data for VPM and HPM. For a specific given object, the size of the bounding box is obtained from the range of data points labeled as that object. Thus, we need only to find the bounding box orientation. We simply generate the bounding boxes for all directions with a step size of 5 degrees. Then, the bounding box with least points from other categories is considered to be the tightest and is selected as our initial bounding box. Given the layout view points set L, the 2D bounding box $\hat{X}_i$ is generated as:

$$\hat{X}_i = \arg\min_{r \in \{0,\pi\}} \{E(X_i(r) \mid L)\}, \tag{6}$$

where $X_i(r)$ is the bounding box parameter set for object i with orientation r in the layout view L. The bounding box evaluatino function E gets the total point number that falls within $X_i(r)$ but not classified as object category i in the layout view L. To get the 3D bounding box, we add the height to X_i using the highest (along the gravity) labeled data point in X_i. Here we consider three kinds of objects, base objects, accessory objects and individual objects as defined before. Although pixel level labeling provides more object categories, we only consider the objects with ground truth bounding boxes.

4.2. VPM for base objects

VPM is used to re-label each given base objectby finding its accessory objects to get a re-labeled layout view map. Following the baseline method in Section 4.1, the bounding box $\hat{X}_i^v$ is generated as:

$$\hat{X}_i^v = \arg\min_{r \in \{0,\pi\}} \{E(X_i^v(r) \mid L_i^v)\}, \tag{7}$$

where L_i^v is generated from L after relabeling it with respect to the object i. L_i^v is obtained from VPM represented by $g_{\mathcal{V}}(...)$ defined in (2) by relabeling all accessory objects to be the base object underneath. The final bounding box heights are determined from initial labeled 3D points.

4.3. HPM for individual objects

In order to create bounding box generation for individual objects, we need to minimize the penalty from two kinds of uncertain 3D points during optimization. The first includes those with exclusive object labels as specified by HPM. The second corresponds to those with a low confidence sore as indicated by the segmentation map from the deep learning network. In other words, these two kinds of 3D points could be in a bounding box for any category. Hence, individual objects could have more flexibility in rotation r and dimension d during optimization, resulting in more accurate bounding box generation as:

$$\hat{X_i^h} = \arg\min_{r,d} \{E(X_i^h(r,d) \mid L^h)\}, \tag{8}$$

where L^h is created from L by suppressing the two types of uncertain points. Note that HPM is used at the scene level, which means L^h is generated for each image while L_i^v is generated for each base object.

5. Experimental results

Although VPM and HPM could be applied to various scene analysis tasks, we tested them for bounding box generation from pixel level labeling (Section 4). We used the SUNRGBD [26] dataset that provides 5285 training images and 5050 testing images. The ground truth labeling provides a 37 classes set for pixel level labeling.

5.1. Experimental setting

To generate the baseline bounding boxes, we use the pixel level labeling map from a recent deep learning method [17] as the input for the algorithm described in Section 4.1. The widely used evaluation measures mean average precision (mAP) under certain IoU threshold. We expect this evaluation could show more details about the real object area detection. We use two metrics to evaluate our method: the bounding box intersection over union (BB-IoU) and the

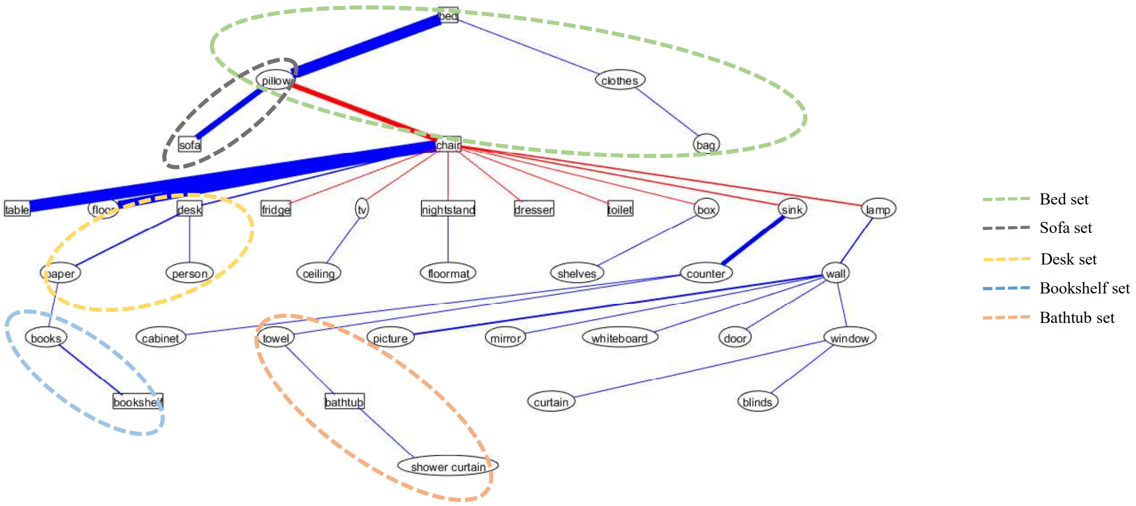

Figure 3. The vertical placement model of dataset SUNRGBD [26]: on the left is the full model, the right side shows the object sets in the model. The blue links shows the closeness while the red ones refer to exclusiveness. The thickness of links indicates the relation strength. The on-ground objects with bounding box tags are shown in rectangles while other objects are shown in ovals. The object set are rectangle-oval connections which stands for the base-accessory object relation. Note that the object relations are learned from training data. The dashed ovals are added only for illustration.

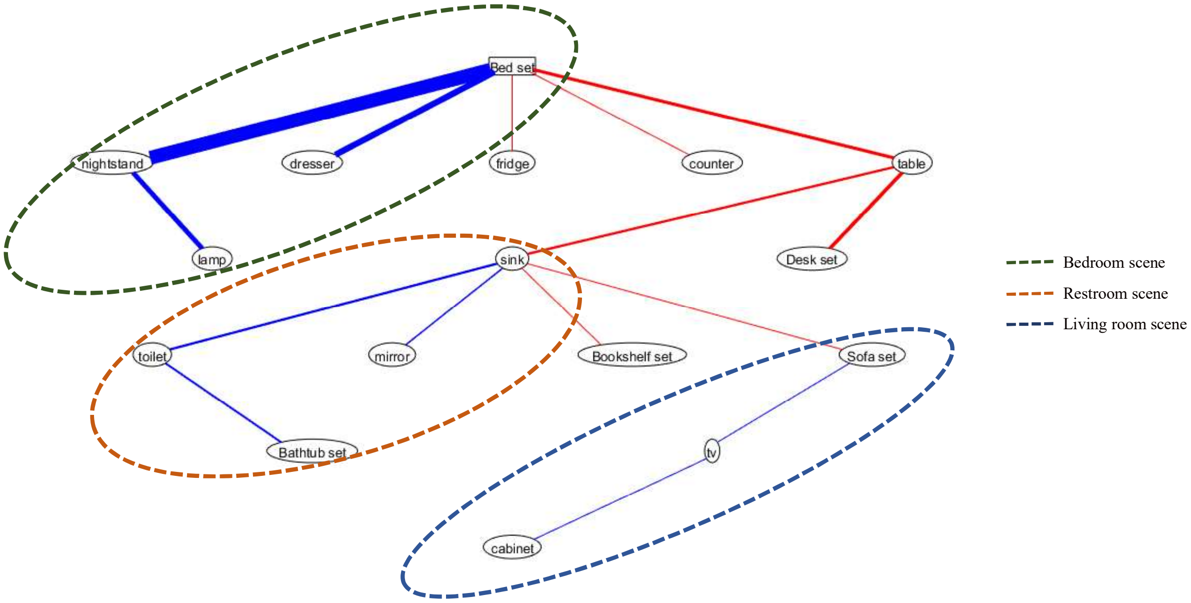

Figure 4. The horizontal placement model of dataset SUNRGBD[26]: the full model is shown on the left. The right side defines the scene groups in the model. The blue links shows the closeness while the red ones refer to exclusiveness. The thickness of links indicates the relation strength. The accessory objects are contained in the sets found using VPM as shown in Figure.3. The model automatically generates three object groups for bedroom, restroom and living room scenes.

visible point intersection over union (VP-IoU). The BB-IoU measures if the bounding box could be correctly found. In indoor scenes with heavy occlusion and sparse 3D data points, some of the bounding box ground-truth are inferred from the visible area. Thus, we use the VP-IoU measure to show if the visible point of the target object could be found. Note that the in VP-IoU, all the points in the target object bounding box are regarded as the same label as the bounding box, regardless their ground-truth pixel level labeling.

5.2. Performance evaluation

In Table 1, we quantitatively show that our method can improve the bounding box accuracy in both BB-IoU and VP-IoU compared with the baseline algorithm denoted as $Base$ (Section 4.1). In these experiments, not all base objects have related accessory objects. Thus, some results in VPM are about the same as in the $Base$ column. Significant improvements in VPM can be found in object categories "bed" and "sofa" because of the prevalent co-existence of bed-pillow and sofa-pillow, as shown in Figure 3. The scores for bookshelf are also improved thanks to the help of the inclusion of the book category. The bounding scores increase in general after we apply dual models because HPM provides more rotation flexibility for all objects. It is worth mentioning that the baseline method cannot correctly detect the nightstand class. After applying the dual models, part of the nightstand is recovered with the help of its related objects (mostly the bed).

Some qualitative results are shown in Figure 5 where we show the effect from VPM and the dual models (VPM+HPM). It is shown that VPM is able to help the inclusion of accessory objects to the base object, leading to improved bounding boxes generation of base objects. Also, the dual models can assist bounding boxes for individual objects by including more uncertain points according to co-existence and exclusiveness encoded in VPM and HPM.

	BB-IoU			VP-IoU		
Category	$Base$	VPM	$Dual$	$Base$	VPM	$Dual$
Cabinet	7.43	7.43	7.66	21.53	21.53	21.76
Bed	24.81	27.67	28.81	52.13	52.79	54.29
Sofa	15.35	16.07	16.67	40.12	38.86	40.2
Table	13.7	13.7	14.7	28.79	28.79	30.43
Desk	7.78	9.74	10.14	18.09	19.02	19.68
Nightstand	0	0	4.8	0.04	0.04	10.4
Bathtub	12.57	13.14	13.46	28.33	28.63	29.22
Bookshelf	11.27	11.68	12.55	40.34	37.52	40.95
Toilet	27.35	27.35	27.85	44.96	44.96	44.78
Fridge	4.48	4.48	6.53	19.15	19.15	22
Dresser	6.6	6.6	7.86	20.12	20.12	20.1
Mean	11.94	12.53	13.73	28.5	28.31	30.35

Table 1. The quantitative results (%) in terms of both BB-IoU and VP-IoU for 11 indoor objects, where the baseline ($Base$) is compared against VPM and the dual models.

5.3. Discussion

It is worth noting that the ground-truth data of pixel-level labeling and bounding boxes still have some inconsistency and ambiguity which may complicate quantitative analysis. Thus the major bottleneck is the pixel-level deep learning algorithm that provides the input for our bottom-up flow. There are three possible directions that would enhance the strengths of VPM and HPM to improve the quality of bounding box generation. First, in stead of using the classification map, confidence maps for each object offer more potential to improve the quality of bounding boxes. Second, we could enhance two models by incorporating more prior regarding the size and shape to improve the inference and optimization of (7) and (8). Third, VPM and HPM can be jointly used to improve the quality of ground-truth data in both training and testing data which consequentially manifest the contribution from two graphical models.

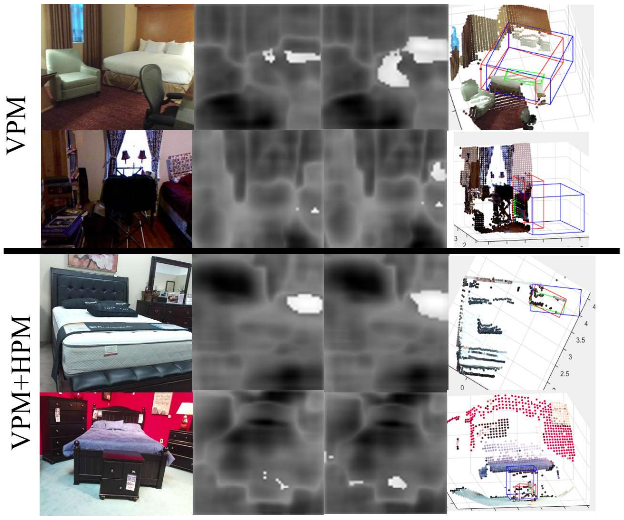

Figure 5. From the results of VMP (top), from left to right, the figures are: (1) Two RGB images, (2) classified *bed* points from the deep network [17], (3) the *bed set* points after using VPM, (4) the generated bounding boxes (blue: ground truth, green: baseline, red: ours). From the results of dual models (VPM+HPM, bottom), from left to right, the figures are: (1) two RGB images, (2) white pixels classified *dresser* (the third row) and *nightstand* (the fourth row) from [17], (3) the uncertain points added to *dresser* (the third row) and the uncertain points that are exclusive with *bed* and added to *nightstand* (the fourth row), (4) the generated bounding boxes (blue: ground truth, green: baseline, red: ours).

6. Conclusion

We have presented dual graphical models for relational modeling of indoor object categories, i.e., the vertical placement model (VPM) and horizontal placement model (HPM). Specifically, the former captures the co-existence of major and accessory objects, while the latter encodes ground level spatial configuration of different individual objects. The two models allow us to bridge the gap among the three levels of semantic scene understanding. As a case study, we apply the two models in a bottom-up flow to create object-specific bounding boxes in 3D space that are more informative and intuitive where the input is the pixel-level label result from any deep neural network. Experimental results show the promise of dual graphical models to improve the quality of bounding box generation. It is foreseeable the two graphical models can be used in other holistic object-level and scene-level analysis tasks.

Acknowledgment

This work is supported in part by the US National Science Foundation (NSF) Grant IIS-1427345, the US National Institutes of Health (NIH) Grant R15 AG061833 and the Oklahoma Center for the Advancement of Science and Technology (OCAST) Health Research Grant HR18-069.

References

[1] Brian Ayers and Matthew Boutell. Home interior classification using sift keypoint histograms. In *Proc. CVPR*, 2007.

[2] Ricardo Cabral and Yasutaka Furukawa. Piecewise planar and compact floorplan reconstruction from images. In *Proc. CVPR*, 2014.

[3] R. Cabral and Y. Furukawa. Piecewise planar and compact floorplan reconstruction from images. In *Proc. CVPR*, 2014.

[4] Myung Jin Choi, Joseph J Lim, Antonio Torralba, and Alan S Willsky. Exploiting hierarchical context on a large database of object categories. In *Proc. CVPR*, 2010.

[5] Myung Jin Choi, Antonio Torralba, and Alan S Willsky. A tree-based context model for object recognition. *IEEE T-PAMI*, 34(2):240–252, 2012.

[6] C Chow and Cong Liu. Approximating discrete probability distributions with dependence trees. *IEEE T-IT*, 14(3):462–467, 1968.

[7] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE T-PAMI*, 35(8):1915–1929, 2013.

[8] Lin Guo, Guoliang Fan, and Weihua Sheng. Robust object detection by cuboid matching with local plane optimization in indoor RGB-D images. In *Proc. VCIP*, 2017.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016.

[10] Satoshi Ikehata, Hang Yang, and Yasutaka Furukawa. Structured indoor modeling. In *Proc. ICCV*, 2015.

[11] H. Jiang and J. Xiao. A linear approach to matching cuboids in RGBD images. In *Proc. CVPR*, 2013.

[12] Salman H Khan, Xuming He, Mohammed Bennamoun, Ferdous Sohel, and Roberto Togneri. Separating objects and clutter in indoor scenes. In *Proc. CVPR*, 2015.

[13] Yinda Zhang Mingru Bai Pushmeet Kohli, Shahram Izadi, and Jianxiong Xiao. Deepcontext: Context-encoding neural pathways for 3D holistic scene understanding. *arXiv preprint arXiv:1603.04922*, 2016.

[14] Hema S Koppula, Abhishek Anand, Thorsten Joachims, and Ashutosh Saxena. Semantic labeling of 3D point clouds for indoor scenes. In *Advances in neural information processing systems*, pages 244–252, 2011.

[15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[16] Victor S Lempitsky, Pushmeet Kohli, Carsten Rother, and Toby Sharp. Image segmentation with a bounding box prior. In *Proc. ICCV*, 2009.

[17] Zhen Li, Yukang Gan, Xiaodan Liang, Yizhou Yu, Hui Cheng, and Liang Lin. Lstm-cf: Unifying context modeling and fusion with lstms for rgb-d scene labeling. In *Proc. ECCV*. Springer, 2016.

[18] Dahua Lin, Sanja Fidler, and Raquel Urtasun. Holistic scene understanding for 3D object detection with RGBD cameras. In *Proc. ICCV*, 2013.

[19] Lorelei Lingard, Sherry Espin, Sarah Whyte, Glenn Regehr, G Ross Baker, Richard Reznick, John Bohnen, Beverly Orser, Diane Doran, and Ellen Grober. Communication failures in the operating room: an observational classification of recurrent types and effects. *BMJ Quality & Safety*, 13(5):330–334, 2004.

[20] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proc. CVPR*, 2015.

[21] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3D bounding box estimation using deep learning and geometry. In *Proc. CVPR*, 2017.

[22] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proc. CVPR*, 2015.

[23] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3D object detection from rgb-d data. In *Proc. CVPR*, 2018.

[24] Mohammad Muntasir Rahman, Yanhao Tan, Jian Xue, Ling Shao, and Ke Lu. 3D object detection: Learning 3D bounding boxes from scaled down 2d bounding boxes in rgb-d images. *Information Sciences*, 476:147–158, 2019.

[25] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proc. CVPR*, 2016.

[26] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *Proc. CVPR*, 2015.

[27] Shuran Song and Jianxiong Xiao. Deep sliding shapes for amodal 3D object detection in rgb-d images. In *Proc. CVPR*, 2016.

[28] Shuran Song, Andy Zeng, Angel X Chang, Manolis Savva, Silvio Savarese, and Thomas Funkhouser. Im2Pano3D: Extrapolating 360 structure and semantics beyond the field of view. In *Proc. CVPR*, 2018.

[29] S. Tang, X. Wang, X. Lv, T. X. Han, J. Keller, Z. He, M. Skubic, and S. Lao. Histogram of oriented normal vectors for object recognition with a depth sensor. In *Proc. ACCV*. Springer, 2012.

[30] Jianxiong Xiao and Yasutaka Furukawa. Reconstructing the worlds museums. *IJCV*, 110(3):243–258, 2014.

[31] J. Xiao and Y. Furukawa. Reconstructing the worlds museums. *IJCV*, 110(3):243–258, 2014.

[32] Danfei Xu, Dragomir Anguelov, and Ashesh Jain. Pointfusion: Deep sensor fusion for 3D bounding box estimation. In *Proc. CVPR*, 2018.

[33] S. M. Seitz Y. Furukawa, B. Curless and R. Szeliski. Manhattan-world stereo. In *Proc. CVPR*, 2009.