Scene Parsing by Integrating Function, Geometry and Appearance Models

Yibiao Zhao
Department of Statistics
University of California, Los Angeles
ybzhao@ucla.edu

Song-Chun Zhu
Department of Statistics and Computer Science
University of California, Los Angeles
sczhu@stat.ucla.edu

Abstract

Indoor functional objects exhibit large view and appearance variations, thus are difficult to be recognized by the traditional appearance-based classification paradigm. In this paper, we present an algorithm to parse indoor images based on two observations: i) The functionality is the most essential property to define an indoor object, e.g. *"a chair to sit on"; ii) The geometry (3D shape) of an object is designed to serve its function. We formulate the nature of the object function into a stochastic grammar model. This model characterizes a joint distribution over the function-geometry-appearance (FGA) hierarchy. The hierarchical structure includes a scene category, functional groups, functional objects, functional parts and 3D geometric shapes. We use a simulated annealing MCMC algorithm to find the maximum a posteriori (MAP) solution,* i.e. *a parse tree. We design four data-driven steps to accelerate the search in the FGA space: i) group the line segments into 3D primitive shapes, ii) assign functional labels to these 3D primitive shapes, iii) fill in missing objects/parts according to the functional labels, and iv) synthesize 2D segmentation maps and verify the current parse tree by the Metropolis-Hastings acceptance probability. The experimental results on several challenging indoor datasets demonstrate the proposed approach not only significantly widens the scope of indoor scene parsing algorithm from the segmentation and the 3D recovery to the functional object recognition, but also yields improved overall performance.*

1. Introduction

In recent years, the object detection and labeling have made remarkable progress in the field of computer vision. However, the detection of indoor objects and segmentation of indoor scenes are still challenging tasks. For example, in the VOC2012 Challenge [5], the state-of-the-art algorithms can only obtain an accuracy of 19.5% for the detection of chairs and 22.6% for the segmentation of chairs. Other indoor objects, like the sofa and the dining table, are among the categories with lowest accuracies out of the twenty object categories.

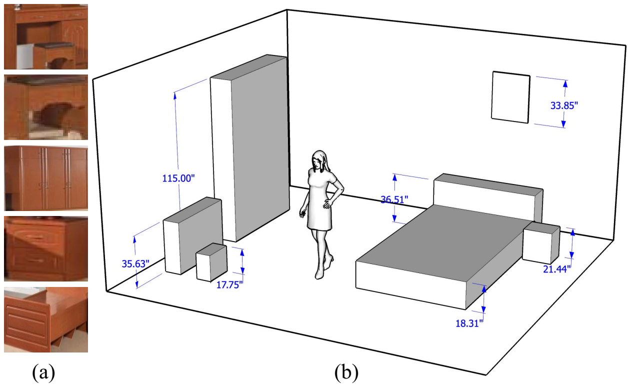


Figure 1. Recognizing objects by appearance (a) or by functionality (b). The functional objects are defined by the *affordance* – how likely its 3D shape is able to afford a human action. The 3D shapes are inferred from an input 2D image in Fig.2.

As shown in Fig.1(a), it is hard to identify these object labels based solely on the appearance of these image patches (cropped from the image in Fig.2). The classic sliding-window type of object detectors, which only observe a window of image like these, will be insufficient to distinguish these image patches apart. From the other point of view in Fig.1(b), despite the appearances, people can immediately recognize objects to sit on (chair), to sleep on (bed) and to store in (cabinet) based on their 3D shapes. For example, a cuboid of 18 inch tall could be comfortable to sit on as a chair. Moreover, the functional context is helpful to identify objects with similar shapes, such as the chair on the left and the nightstand on the right. Although they are in similar shape, the nightstand is more likely to be placed beside the bed. The bed and the nightstand offer a joint functional group to serve the activity of sleeping. Based on the above observations, we propose an algorithm to tackle the problem of indoor scene parsing by modeling the object function, the 3D geometry and the local appearance (FGA).

There has been a recent surge in the detection of rectangular structures, typically modeled by planar surfaces or cuboids, in the indoor environment. (i) Hedau *et al.* [12, 13], Wang *et al.* [21], Lee *et al.* [17, 16] and Satkin *et al.* [19] adopted different approaches to model the geometric layout of the background and/or foreground blocks with the Structured SVM (or Latent SVM). (ii) Another stream of algorithms including Han and Zhu [10], Zhao and Zhu [26] and Del Pero *et al.* [4, 3] that built generative Bayesian models to capture the prior statistics in the man-made scenes. (iii) Hu [15], Xiao *et al.* [24], Hejrati and Ramanan [14], Xiang and Savarese [22], Pepik *et al.* [18] designed several new variants of the deformable part-based models [6] by using detectors of projected 3D parts. (iv) Bar-Aviv *et al.* [1] and Grabner *et al.* [8] detected chairs by the simulation of embodied agents in the 3D CAD data and depth data respectively. Gupta *et al.* [9] recently proposed to infer the human workable space by adapting the human poses to the scene.

Overview of our approach: On top of a series of recent studies of computing the 3D bounding boxes of indoor objects and the room layout [9, 12, 13, 21, 17, 16, 10, 26, 4, 3, 19], our model is developed based on the following observations of the function-geometry-appearance (FGA) hierarchy as shown in Fig.2.

i) *Function*: An indoor scene is designed to serve a handful of human activities inside. The indoor objects (furniture) in the scenes are designed to support human poses/actions, *e.g.* bed to sleep on, chair to sit on *etc*.

In the functional space, we model the probabilistic derivation of functional labels including scene categories (bedroom), functional groups (sleeping area), functional objects (bed and nightstand), and functional parts (the mattress and the headboard of a bed).

ii) *Geometry*: The 3D size (dimension) can be sufficient to evaluate how likely an object is able to afford a human action, known as the *affordance* [7]. Fortunately, most of the furniture has regular structures, *i.e.* a rectangular cabinet, therefore the detection of these objects is tractable by inferring their geometric affordance. For objects like sofas and beds, we use a more fine-grained geometric model with compositional parts, *i.e.* a group of cuboids. For example, the bed with a headboard better explains the image signal as shown at the bottom of Fig.2.

In the geometric space, each 3D shape is directly linked to a functional part in the functional space. The contextual relations are also involved when multiple objects are assigned to a same functional group, *e.g.* a bed and a nightstand for sleeping. The distribution of the 3D geometry are learned from a large set of 3D models as shown in Fig.3.

iii) *Appearance*: The appearance of the furniture vary arbitrarily large due to the variation of material property, the lighting condition, and the view point. In order to land our model on the input image, we use a straight-line detection, a surface orientation estimation and a coarse foreground detection as the local evidence to support the geometry model above.

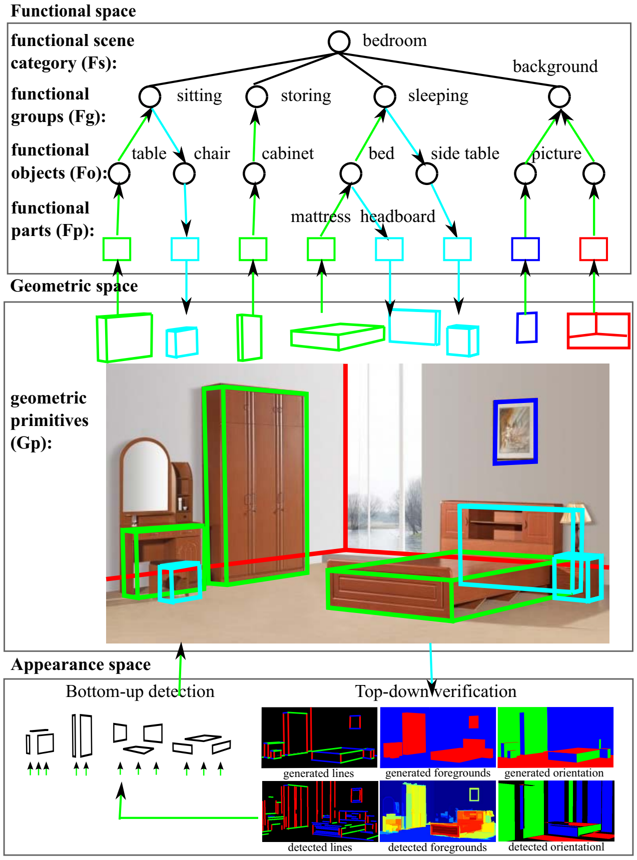


Figure 2. A function-geometry-appearance (FGA) hierarchy. The green arrows indicate the bottom-up steps, and the cyan arrows represent the top-down steps in the inference stage.

We design a four-step inference algorithm that enables a MCMC chain to travel up and down through the FGA hierarchy:

i). A bottom-up appearance-geometry (AG) step groups noisy line segments in the A space into 3D primitive shapes, *i.e.* cuboids and rectangles, in the G space;

ii). A bottom-up geometry-function (GF) step assigns functional labels in the F space to detected 3D primitive shapes, *e.g.* to sleep on;

iii). A top-down function-geometry (FG) step further fills in the missing objects and the missing parts in the G space according to the assigned functional labels, *e.g.* a missing nightstand of a sleeping group, a missing headboard of a bed;

iv). A top-down geometry-appearance (GA) step synthesizes 2D segmentation maps in the A space, and makes an accept/reject decision of a current proposal by the Metropolis-Hastings acceptance probability.

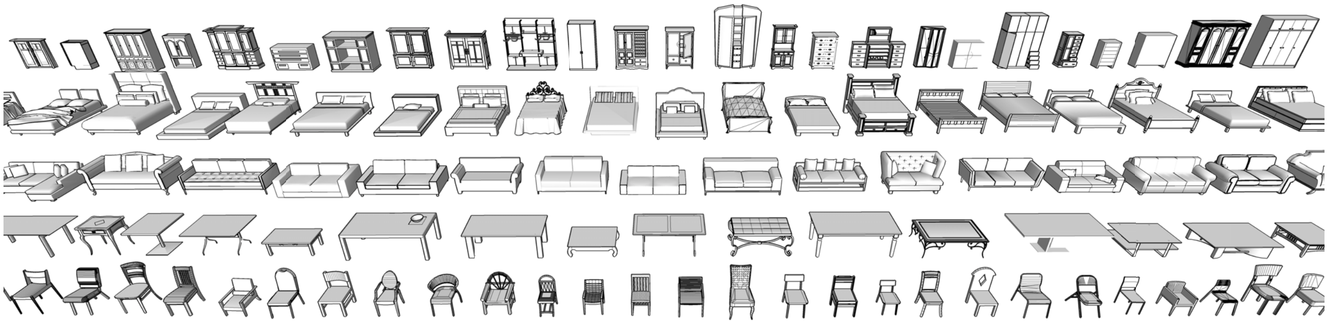


Figure 3. A collection of indoor functional objects from the Google 3D Warehouse

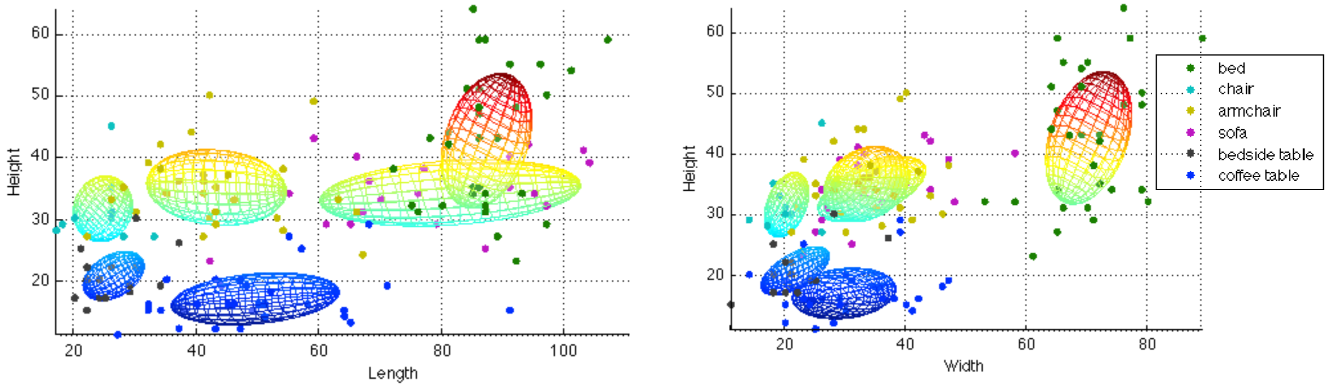


Figure 4. The distribution of the 3D sizes of the functional objects (in unit of inch).

2. A stochastic scene grammar in FGA space

We present a stochastic scene grammar model [26] to compute a parse tree pt for an input image $\mathcal{I}$ on the FGA hierarchy. The hierarchy includes following random variables in the FGA spaces as shown in Fig.2 :

- The functional space $\mathcal{F}$ contains the scene categories Fs, the functional groups Fg, the functional objects Fo, and the functional parts Fp. All the variables in functional space take discrete labels;
- The geometric space $\mathcal{G}$ contains the 3D geometric primitives Gp. Each Gp is parameterized by a continuous 6D variable (a 3D size and a 3D position in the scene);
- The appearance space $\mathcal{A}$ contains a line segment map Al, a foreground map Af, and a surface orientation map Ao. All of them can be either computed from the image $Al(\mathcal{I}), Af(\mathcal{I}), Ao(\mathcal{I})$ or generated by the 3D geometric model $Al(\mathcal{G}), Af(\mathcal{G}), Ao(\mathcal{G})$.

The probabilistic distribution of our model is defined in terms of the statistics over the derivation of our function-geometry-appearance hierarchy. The parse tree is an instance of the hierarchy $pt \in \{\mathcal{F}, \mathcal{G}, \mathcal{A}\}$ as illustrated in Fig.2, and it is an optimal solution of our model by maximum a posteriori probability,

$$P(pt|\mathcal{I}) \propto P(\mathcal{F})P(\mathcal{G}|\mathcal{F})P(\mathcal{I}|\mathcal{G}). \qquad (1)$$

We specify a hierarchy of an indoor scene over the functional space $\mathcal{F}$, the geometric space $\mathcal{G}$ and the appearance space $\mathcal{A}$.

2.1. The function model $P(\mathcal{F})$

The function model characterizes the prior distribution of the functional labels. We model the distribution by the probabilistic context free grammar (PCFG): $G = (N, T, S, R)$, where $N = \{Fs, Fg, Fo\}$ are the non-terminal nodes (circles in Fig.2), and $T = \{Fp\}$ are the functional parts as terminal nodes in F space (squares in Fig.2), S is a start symbol and $R = \{r : \alpha \to \beta\}$ is a set of production rules. In our problem, we define following production rules:

S → Fs: S → [bedroom] | [living room]
Fs→Fg: [bedroom] → [sleeping][background] | ···
Fg→Fo: [sleeping] → [bed] | [bed][night stand] | ···
Fo→Fp: [bed] → [headboard][mattress] | [mattress]

The symbol "|" separates alternative explanations of the grammar derivation. Each alternative explanation has a branching probility $q(\alpha \to \beta) = P(\beta|\alpha)$. Given a functional parse containing the production rules $\alpha_1 \to \beta_1, \cdots, \alpha_n \to \beta_n$, the probability under the PCFG is defined as,

$$P(\mathcal{F}) = \prod_{i=1}^{n} q(\alpha_i \to \beta_i) \tag{2}$$

The model is learned by simply counting the frequency of each production rules as $q(\alpha \to \beta) = \frac{\#(\alpha \to \beta)}{\#(\alpha)}$. In this paper, we manually designed the grammar structure and learned the parameters of the production rules based on the labels of thousands of images in the SUN dataset [23] under the "bedroom" and the "living room" categories.

2.2. The geometric model $P(\mathcal{G}|\mathcal{F})$

In the geometric space, we model the distribution of 3D size (dimension) for each geometric primitive Gp given its functional labels $\mathcal{F}$, *e.g.* the size distribution of cuboid shaped bed mattresses. The higher level functional labels Fs, Fg, Fo introduce the contextual relations among these primitives, *e.g.* the distance distribution between a bed and a nightstand. Suppose we have k primitives in the scene $Gp = \{v_i : i = 1, \cdots, k\}$, these geometric shapes form a graph $G = (V, E)$ in the G space, where each primitive is a graph node $v_i \in V$, and each contextual relation is a graph edge $e \in E$. In this way, we derive a Markov Random Fields (MRFs) model at the geometric level. The joint probability is factorized over the graph cliques,

$$\begin{aligned} P(\mathcal{G}|\mathcal{F}) = & \prod_{v_i \in Gp} \varphi_1(v_i|Fp) \\ & \prod_{e_i \in cl(Fo)} \varphi_2(e_i|Fo) \prod_{e_i \in cl(Fg)} \varphi_3(e_i|Fg) \\ & \prod_{e_i \in cl(S)} \varphi_4(e_i) \end{aligned} \tag{3}$$

where the $e_i \in cl(X)$ denotes an edge whose two connecting nodes belong to the children (or descendant) of the X. These four kinds of cliques are introduced by the functional parts Fp, the functional objects Fo, the functional groups Fg and the general physical constraints respectively:

Object affordance $\varphi_1(v_i|Fp)$ is an "unary term" which models one to one correspondences between the geometric primitives Gp and the functional parts Fp. The probability measures how likely an object is able to afford the action given its geometry. As shown in Fig.1, a cube around 1.5ft tall is comfortable to sit on despite its appearance, and a "table" of 6ft tall loses its original function – to place objects on while sitting in front of. We model the 3D sizes of the functional parts by a mixture of Gaussians. The model characterizes the Gaussian nature of the object sizes and allows the alternatives of canonical sizes at the same time, such as king size bed, full size bed *etc*. We estimate the model by EM clustering, and we manually picked few typical primitives as the initial mean of Gaussian, *e.g.* a coffee table, a side table and a desk from the table category.

In order to learn a better affordance model, we collected a dataset of functional indoor furniture, as shown in Fig.3. The functional objects in the dataset are modeled with the real-world measurement, therefore we can generalize our model to the real images by learning from this dataset. We found that the real-world 3D size of the objects has less variance than the projected 2D size. As we can see, these functional categories are quite distinguishable solely based on their sizes as shown in Fig.4. For example, the coffee tables and side tables are very short and usually lower than the sofas, and the beds generally have large widths comparing to the other objects. The object poses are aligned in the dataset. We keep four copies of Gaussian model for four alternative orientations along x, $-x$, y and $-y$ axes to make the model rotation invariant in the testing stage.

3D compositional models of functional object and functional groups $\varphi_2(e_i|Fo)$, $\varphi_3(e_i|Fg)$ are defined by the distributions of the 3D relative relations among the parts of an objects Fo or the objects of an functional group Fg. We also use a high-dimensional Gaussian to model the relative relations. The Fig.5 shows some typical samples drawn from our learned distribution. This term enables the top-down prediction of the missing parts or missing objects as we will discuss in Sect.3.

General physical constraints $\varphi_4(e_i)$ avoid invalid geometric configurations that violate the physical laws: Two objects can not penetrate each other; the objects must be contained in the room. The model penalizes the penetrating area between foreground objects Λ_f and the exceeding area beyond the background room borders Λ_b as $1/z \exp\{-\lambda(\Lambda_f + \Lambda_b)\}$, where we take λ as a large number, and $\Lambda_f = \Lambda(v_i) \bigcap \Lambda(v_j)$, $\Lambda_b = \Lambda(v_i) \bigcap \overline{\Lambda(bg)}$.

2.3. The appearance model $P(\mathcal{I}|\mathcal{G})$

We define the appearance model by applying the idea of *analysis-by-synthesis*. In the functional space and the geometric space, we specify how the underlying causes generate a scene image. There is still a gap between the synthesized scene and the observed image, because we can not render a real image without knowing the accurate lighting condition and material parameters. In order to fill this gap, we make use of the discriminative approaches: a line segment detector [20], a foreground detector [12] and a surface orientation detector [17] to produce a line map $Al(\mathcal{I})$, a foreground map $Af(\mathcal{I})$ and a surface orientation map $Ao(\mathcal{I})$ respectively. By projecting detected 3D geometric primitives onto the image plane, we evaluate our model by calculating the pixel-wise difference between the maps from top-down projection and the maps from bottom-up de-

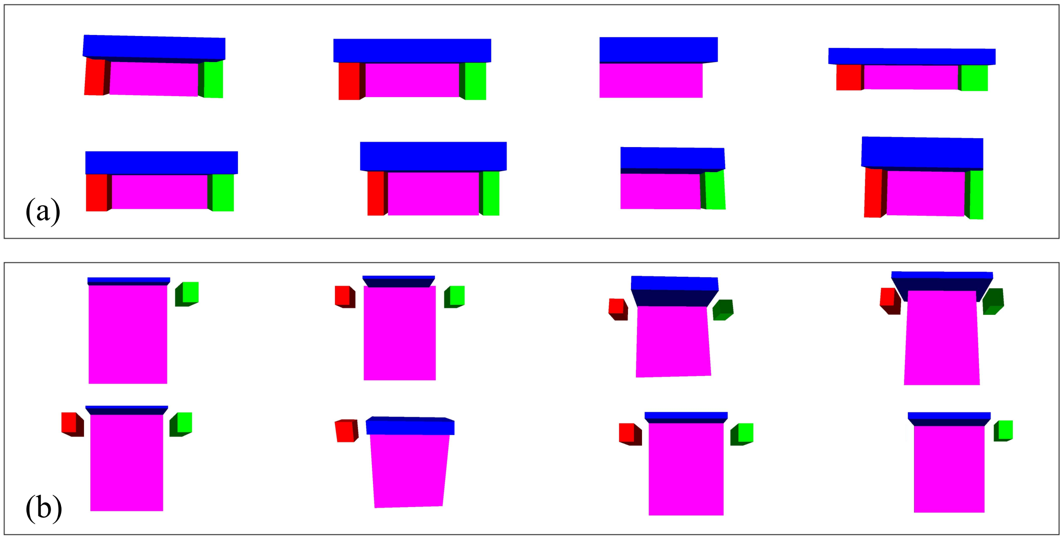


Figure 5. Samples drawn from the distributions of 3D geometric models (a) the functional object “sofa” and (b) the functional group “sleeping”.

tection as shown in Fig.2.

$$P(\mathcal{I}|\mathcal{G}) \propto \exp(\lambda[d(\mathrm{Al}(\mathcal{G}), \mathrm{Al}(\mathcal{I})) + d(\mathrm{Af}(\mathcal{G}), \mathrm{Af}(\mathcal{I})) + d(\mathrm{Ao}(\mathcal{G}), \mathrm{Ao}(\mathcal{I}))]) \quad (4)$$

These image features have been widely used in recent studies [9, 12, 13, 21, 17, 16, 10, 26, 4, 3, 19], hence we will skip further discussion of details about them.

2.4. Put objects back to the 3D world

Another important component of our model is the recovery of a real world 3D geometry from the parse tree (Fig.2). It enables us to utilize the 3D geometric/contextual measurement to identify the object affordance/functional groups as discussed before.

Single view camera calibration: We cluster line segments to find three vanishing points whose corresponding dimensions are orthogonal to each other [12]. The vanishing points are then used to determine the intrinsic and extrinsic calibration parameters [2, 11]. We assume that the aspect ratio is 1 and there is no skew. Any pair of finite vanishing points can be used to estimate the focal length. If all the three vanishing points are visible and finite in the same image, the optical center can be estimated as the orthocenter of the triangle formed by the three vanishing points. Otherwise, we set the optical center to the center of an image. Once the focal length and optical center has been determined, the camera rotational matrix can be estimated accordingly [11].

3D scene reconstruction. We now present how to back-project a 2D structure to the 3D space and how to derive the corresponding coordinates. Considering a 2D point p in an image, there is a collection of 3D points that can be projected to the same 2D point p. This collection of 3D points lays on a ray from the camera center $C = (Cx, Cy, Cz)^T$ to the pixel $p = (x, y, 1)^T$. The ray $P(\lambda)$ is defined by $(X, Y, Z)^T = C + \lambda R^{-1}K^{-1}p$, where λ is the positive scaling factor that indicates the position of the 3D point on the ray. Therefore, the 3D position of the pixel lies at the intersection of the ray and a plane (the object surface). We assume a camera is 4.5ft high. By knowing the distance and the normal of the floor plane, we can recover the 3D position for each pixel with the math discussed above. And any other plane contacting with the floor can be inferred by its contacting point with the floor. Then we can gradually recover the whole scene by repeating the process from bottom up. If there is any object too close to the camera without showing its bottoms, we will put it 3 feet away from the camera.

3. Top-down / bottom-up inference

We design a top-down/bottom-up algorithm to infer an optimal parse tree. The compositional structure of the continuous geometric parameters introduces a large solution space, which is infeasible to enumerate all the possible explanations. Neither the sliding windows (top-down) nor the binding (bottom-up) approaches can handle such an enormous number of configurations independently. We design a four-step inference algorithm that enables a MCMC chain to travel up and down through the FGA hierarchy: $\mathcal{A} \to \mathcal{G} \to \mathcal{F} \to \mathcal{G} \to \mathcal{A}$. In each iteration, the algorithm proposes a new parse tree pt^* based on the current one pt according to the proposal probability.

I. **A bottom-up appearance-geometry (AG) step** detects possible geometric primitives Gp as bottom-up proposals, *i.e.* rectangles and cuboids, from the noisy local line segments. The rectangles are formed by filtering over the combinations of two pairs of parallel lines or T junctions. Similarly, the cuboids are formed by filtering over the combinations of two hinged rectangles. The proposal probability for a new geometric primitive g^* is defined as

$$Q_1(g^*|\mathcal{I}_\Lambda) = \frac{P_A(\mathcal{I}_\Lambda|g^*)P(g^*)}{\int_{g\in Gp} P_A(\mathcal{I}_\Lambda|g)P(g)} \tag{5}$$

where the $P_A(\mathcal{I}_\Lambda|g)$ is defined in a similar form of Eq.4 except that we only calculate the image likelihood within a local patch $\mathcal{I}_\Lambda$. The $P(g)$ characterizes the prior distribution, *i.e.* how likely the shape of g can be generated by the model.

$$P(g) = \int_{\mathcal{F}} P(\mathcal{F}, g) = \int_{\mathcal{F}} P(\mathcal{F})P(g|\mathcal{F}) \tag{6}$$

Since $P(g|\mathcal{F})$ is defined by a Gaussian model, $\int_{\mathcal{F}} P(\mathcal{F})P(g|\mathcal{F})$ is a mixture of a large number of Gaussians, and $P(\mathcal{F})$ is a hyperprior of mixture coefficients. It is worth noting that this proposal probability Q_1 is independent of the current parse tree pt. Therefore we can precompute the proposal probability for each possible geometric proposal, which dramatically reduces the computational cost of the chain search.

II. **A bottom-up geometry-function (GF) step** assigns functional labels given the 3D shapes detected in the G space. The proposal probability of switching an functional label f^* on the functional tree is defined as

$$Q_2(f^*|pa, cl) = \frac{P(cl|f^*)P(f^*|pa)}{\int_f P(cl|f)P(f|pa)} \tag{7}$$

where the cl are the children of f^*, and pa is the parent of f^* on the current parse tree pt. In this way, the probability describes the compatibility of the functional label f^* with its parent pa and its children cl on the tree. With the geometry primitives fixed on the bottom, this proposal makes the chain jumping in the functional space to find a better functional explanation for these primitives. With the Markov property on the tree, $Q_2(f^*|pa, cl)$ is equivalent to the marginal probability $P(f^*|pt)$.

III. **A top-down function-geometry (FG) step** fills in the missing object in a functional group or the missing part in a functional object. For example, once a bed is detected, the algorithm will try to propose nightstands beside it by drawing samples from the geometric prior and the contextual relations. The problem of sampling with complex constraints was carefully studied by Yeh *et al.* [25]. Fig.5 shows some typical samples. The proposal probability $Q_3(g^*|\mathcal{F})$ of a new geometric primitive g^* is defined by Eq.3.

Here, we can see that $Q_1(\mathcal{I} \to \mathcal{G})$ proposes g^* by the bottom-up image detection, and $Q_3(\mathcal{F} \to \mathcal{G})$ proposes g^* by the top-down functional prediction. They are two approximations of the marginal distribution $P(g^*|pt)$.

On the other hand, the algorithm also proposes to delete a geometric primitive with uniform probability. Both the add or delete operation will trigger the step II of reassigning a functional label.

IV. **A top-down geometry-appearance (GA) step** projects the 3D geometric model to the 2D image plane with respect to the relative depth order and camera parameters. The projection is a deterministic step. It generates the image feature maps used to calculate the overall likelihood in Eq.4. And the image features are shown at the bottom of Fig.2.

We evaluate the above proposals by the Metropolis-Hastings acceptance probability,

$$\alpha(pt \to pt^*) = min\{1, \frac{Q(pt|pt^*, I)}{Q(pt^*|pt, I)} \cdot \frac{P(pt^*|I)}{P(pt|I)}\} \tag{8}$$

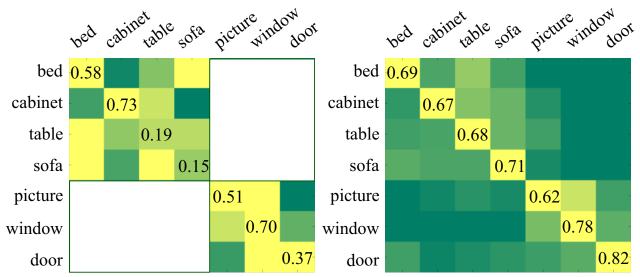
so that the Markov chain search satisfies the detailed balance principle. A simulated annealing technology is also used to find the maximum of complex posteriori distribution with multiple peaks while other approaches may trap the algorithm at a less optimal peak.

4. Experiments

Our algorithm has been evaluated on the UIUC indoor dataset [12], the UCB dataset [4], and the SUN dataset [23]. The UCB dataset contains 340 images and covers four cubic objects (bed, cabinet, table and sofa) and three planar objects (picture, window and door). The ground-truths are provided with hand labeled segments for geometric primitives. The UIUC indoor dataset contains 314 cluttered indoor images and the ground-truth is two label maps of the background layout with/without foreground objects. We picked two categories in the SUN dataset: the bedroom with 2119 images and the living room with 2385 images. This dataset contains thousands of object labels and was used to train our functional model as discussed in Sect.2.1.

Quantitative evaluation: We first compared the confusion matrix of functional object classification rates among the successfully detected objects on the UCB dataset as shown in Fig.6. A latest work by Del Pero *et al.* [3] performed slightly better on the cabinet category, but our method get better performance on the table and sofa categories. This is mainly attributed to our fine-grained part model and functional groups model. It is worth noting that our method reduced the confusion between the bed and the sofa. Because we also introduced the hidden variables of scene categories, which help to distinguish between the bedroom and living room according to the objects inside.

In Table.1, we compared the precision and recall of functional object detection with Del Pero's work [3]. The result shows our top-down process did not help the detection of planner objects. But it largely improves the accuracy of cu-



(a) Pero et al. 2012 [27] (b) our approach

Figure 6. The confusion matrix of functional object classification on the UCB dataset.

Table 1. The precision (and recall) of functional object detection on the UCB dataset.

UCB dataset	planar objects	cubic objects
Del Pero 2012 [3]	27.7% (19.7%)	31.0% (20.1%)
Ours w/o top-down	28.1%(18.5%)	30.8% (24.3%)
Ours w/ top-down	28.1%(18.7%)	34.8% (29.7%)

Table 2. The pixel classification accuracy of background layout segmentation on the UCB dataset and the UIUC dataset.

	UCB dataset	UIUC dataset
Hedau 2009 [12]	-	78.8%
Wang 2010 [21]	-	79.9%
Lee 2010 [16]	-	83.8%
Del Pero 2011 [4]	76.0%	73.2%
Del Pero 2012 [3]	81.6%	83.7%
Our approach	82.8%	85.5%

bic object detection from 30.8% to 34.8% with the recall from 24.3% to 29.7%.

In Table.2, we also test our algorithm on the UCB dataset and the UIUC dataset together with five state-of-the-art algorithms: Hedau 2009 [12], Wang 2010 [21], Lee 2010 [16], Del Pero 2011 [4] and Del Pero 2012 [3]. The results show the pixel-level segmentation accuracy of proposed algorithms not only significantly widens the scope of indoor scene parsing algorithm from the segmentation and 3D recovery to the functional object recognition, but also yields improved overall performance.

Qualitative evaluation: Some experimental results on the UIUC and the SUN datasets are illustrated in Fig.7. The green cuboids are cubic objects proposed by the bottom-up AG step, and the cyan cuboids are the cubic objects proposed by the top-down FG step. The blue rectangles are the detected planar objects, and the red boxes are the background layouts. The functional labels are given to the right of each image. Our method has detected most of the indoor objects, and recovered their functional labels very well. The top-down predictions are very useful to detect highly occluded nightstands as well as the headboards of the beds. As shown in the last row, our method sometimes failed to detect certain objects. The bottom left image fails to identify the drawer in the left but a door. In the middle bottom image, the algorithm failed to accurately locate the mattress for this bed with a curtain. The last image is a kind of typical failure example due to the unusual camera position. We assumed the camera position is 4.5 feet high, while this camera position in this image is higher than our assumptions. As a result, the algorithm detected a much larger bed instead.

5. Conclusion

This paper presents a stochastic grammar built on a function-geometry-appearance (FGA) hierarchy. Our approach parses an indoor image by inferring the object function and the 3D geometry. The functionality defines an indoor object by evaluating its “affordance”. The affordance measures how much an object can support the corresponding human action, e.g. a bed is able to support the action of sleep. We found it is effective to recognize certain object functions according to its 3D geometry regardless of observing the actions.

The function helps to build an intrinsic bridge between the man-made object and the human action, which can motivate other interesting studies in the future: functional objects/areas in a scene attract human’s needs and/or intentions; other risky areas (like shape corners) apply repulsive force to human actions. As a result, a parsed scene with functional labels defines a human action space, and it also helps to predict people’s behavior by making use of the function cues. On the other hand, given observed action sequence, it is very obvious to accurately recognize the functional objects associated with the rational actions.

Acknowledgment

This work is supported by ONR MURI grant N00014-10-1-0933 and DARPA MSEE grant FA 8650-11-1-7149.

References

[1] E. Bar-aviv and E. Rivlin. Functional 3d object classification using simulation of embodied agent. In *BMVC*, 2006.

[2] A. Criminisi, I. Reid, and A. Zisserman. Single view metrology. *Int. J. Computer Vision (IJCV)*, 40(2):123–148, Nov. 2000.

[3] L. Del Pero, J. Bowdish, D. Fried, B. Kermgard, E. Hartley, and K. Barnard. Bayesian geometric modeling of indoor scenes. In *CVPR*, pages 2719–2726, 2012.

[4] L. Del Pero, J. Guan, E. Brau, J. Schlecht, and K. Barnard. Sampling bedrooms. In *CVPR*, pages 2009–2016, 2011.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

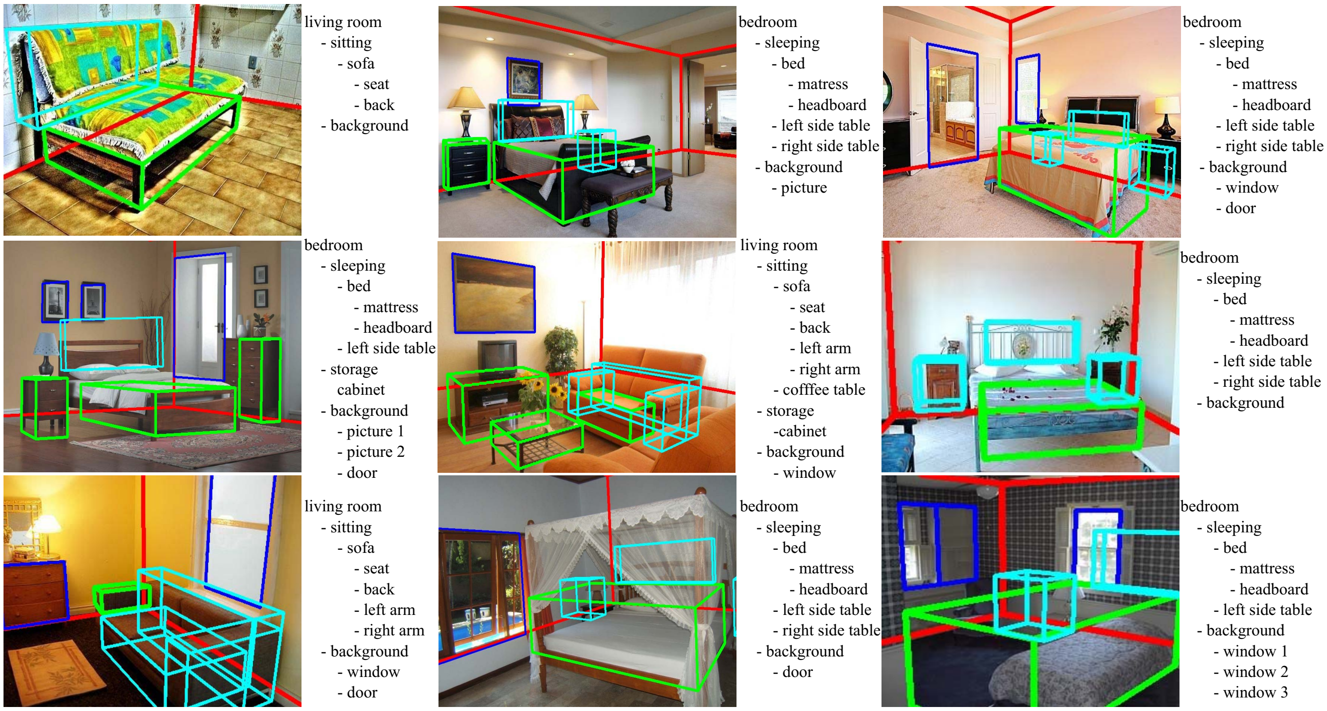


Figure 7. Parsing results include cubic objects (green cuboids are detected by bottom-up step, and cyan cuboids are detected by top-down prediction), planar objects (blue rectangles), background layout (red box). The parse tree is shown to the right of each image.

[6] P. F. Felzenszwalb, R. B. Girshick, and D. Mcallester. D.m.: Cascade object detection with deformable part models. In *CVPR*, 2010.

[7] J. J. Gibson. *The Theory of Affordances*. Lawrence Erlbaum, 1977.

[8] H. Grabner, J. Gall, and L. V. Gool. What makes a chair a chair? In *CVPR*, 2011.

[9] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert. From 3d scene geometry to human workspace. In *CVPR*, pages 1961–1968, Washington, DC, USA, 2011. IEEE Computer Society.

[10] F. Han and S. C. Zhu. Bottom-up/top-down image parsing with attribute grammar. *PAMI*, 2009.

[11] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[12] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*, 2009.

[13] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*, 2010.

[14] M. Hejrati and D. Ramanan. Analyzing 3d objects in cluttered images. In P. Bartlett, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 602–610. 2012.

[15] W. Hu. Learning 3d object templates by hierarchical quantization of geometry and appearance spaces. In *CVPR*, pages 2336–2343, 2012.

[16] D. Lee, A. Gupta, M. Hebert, and T. Kanade. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces advances in neural information processing systems. *Cambridge: MIT Press*, pages 609–616, 2010.

[17] D. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009.

[18] B. Pepik, P. Gehler, M. Stark, and B. Schiele. 3d2pm - 3d deformable part models. In *ECCV*, Firenze, Italy, 2012.

[19] S. Satkin, J. Lin, and M. Hebert. Data-driven scene understanding from 3d models. In *BMVC*, September 2012.

[20] R. von Gioi, J. Jakubowicz, J. M. Morel, and G. Randall. Lsd: A fast line segment detector with a false detection control. *TPAMI*, 32(4):722–732, 2010.

[21] H. Wang, S. Gould, and D. Koller. Discriminative learning with latent variables for cluttered indoor scene understanding. In *ECCV*, pages 497–510, 2010.

[22] Y. Xiang and S. Savarese. Estimating the aspect layout of object categories. In *CVPR*, 2012.

[23] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, pages 3485 –3492, 2010.

[24] J. Xiao, B. Russell, and A. Torralba. Localizing 3d cuboids in single-view images. In P. Bartlett, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *NIPS*, pages 755–763. 2012.

[25] Y.-T. Yeh, L. Yang, M. Watson, N. D. Goodman, and P. Hanrahan. Synthesizing open worlds with constraints using locally annealed reversible jump mcmc. *ACM Trans. Graph.*, 31(4):56:1–56:11, July 2012.

[26] Y. Zhao and S. C. Zhu. Image parsing via stochastic scene grammar. In *NIPS*. 2011.