Noname manuscript No.
(will be inserted by the editor)

Integrating Function, Geometry, Appearance for Scene Parsing

Yibiao Zhao · Song-Chun Zhu

Received: date / Accepted: date

Abstract In this paper, we present a Stochastic Scene Grammar (SSG) for parsing 2D indoor images into 3D scene layouts. Our grammar model integrates object functionality, 3D object geometry, and their 2D image appearance in a Function-Geometry-Appearance (FGA) hierarchy. In contrast to the prevailing approach in the literature which recognizes scenes and detects objects through appearance-based classification using machine learning techniques, our method takes a different perspective to scene understanding and recognizes objects and scenes by reasoning their functionality. Functionality is an essential property which often defines the categories of objects and scenes, and decides the design of geometry and scene layout. For example, a sofa is for people to sit comfortably, and a kitchen is a space for people to prepare food with various objects. Our SSG formulates object functionality and contextual relations between objects and imagined human poses in a joint probability distribution in the FGA hierarchy. The latter includes both functional concepts (the scene category, functional groups, functional objects, functional parts) and geometric entities (3D/2D/1D shape primitives). The decomposition of the grammar is terminated on the bottom-up detected lines and regions. We use a Markov chain Monte Carlo (MCMC) algorithm to optimize the Bayesian a posteriori probability and the output parse tree includes a 3D description of the 2D image in the FGA hierarchy. Experimental results on two challenging indoor datasets demonstrate that the proposed approach not only significantly widens the scope of indoor scene parsing from traditional scene segmentation, labeling, and 3D reconstruction to functional object recognition, but also yields improved overall performance.

1 Introduction

1.1 Motivation and objective

In the past 15 years, a prevailing approach in the vision literature has been posting scene recognition as a classification problem – classifying scene categories, recognizing scene attributes, and detecting objects through appearance-based features, machine learning techniques, and large training examples. Such approach essentially memorizes the typical examples in each scene or object categories, does not "understand" the real meanings of objects and scenes, and thus is known to have difficulties in generalizing and extrapolating into unseen features spaces.

One example is shown in Figure 1. Taken from a similar viewing angle, the two images have drastically different appearance and geometry, but are both considered kitchen by human vision. What are common to the two images are the functionality of objects and the 3D spaces in serving a set of human actions – preparing food.

Functionality refers to the property of an object or scene, especially man-made ones, which has a practical use for which it was designed. Psychologist Gibson (1977) used another term, *affordance*, which refers to the property of an object that affords the opportunity

Yibiao Zhao
University of California, Los Angeles (UCLA), USA
E-mail: ybzhao@ucla.edu
www.yibiaozhao.com

Song-Chun Zhu
University of California, Los Angeles (UCLA), USA
E-mail: sczhu@stat.ucla.edu
http://www.stat.ucla.edu/~sczhu

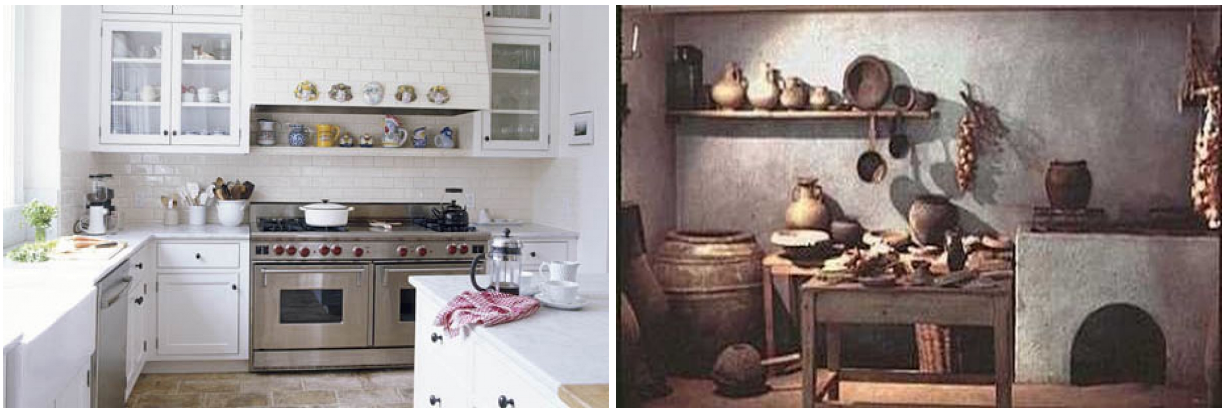


Fig. 1 A modern kitchen and an ancient kitchen with similar functions but drastically different geometry and appearances.

for humans to perform some specific actions. From such view point, we argue that

- *objects, especially man-made ones, are defined by their functions and actions that they are involved.*
- *scenes, especially man-made ones, are defined by the activities and actions that they can provide space for.*

So, functionality is deeper than geometry and appearance and thus is a more invariant concept for scene understanding.

This represents a different philosophy that views vision tasks from the perspective of agents, that is, agents (humans, animals and robots) should perceive objects and scenes by reasoning their plausible functions. We believe this perspective is a more robust way and will take us to deeper human-like scene understanding systems.

Motivated by the above observations, this paper poses scene understanding as an image parsing problem following the work of Tu et al (2005) and is aimed at two objectives in the following.

Our first objective is to present a Stochastic Scene Grammar (SSG) as a hierarchical compositional representation which integrates functionality, geometry and appearance in a FGA hierarchy. For example, Fig. 3 shows a parse tree derived from this grammar in the joint FGA spaces for a bedroom image. In contrast to traditional syntactical parsing advocated by Fu (1982), the scene (root node) is defined by a set of most probable actions (diamonds) that may occur in the scene. The actions are reasoned based on the geometry of the objects and imagined human skeletons, as Fig. 2.(c) illustrates. Such human object interaction models can be learned offline through RGBD videos, *e.g.* Wei et al (2013). The geometric objects are grouped from line segments extracted from image appearances. In Fig. 2(c), the geometric dimensions of furniture in the room are designed to fit the sizes of humans. For example, any flat surface for sitting is usually 18 inches tall, *i.e.* knee height, and a place to sleep is usually between 6-8 feet. Moreover, the contextual relations between the furniture pieces are helpful in distinguishing their functions and therefore assigning their names, *e.g.* the nightstand is near the bed and the lamp is on top of the nightstand. Some typical functional groups are illustrated in Fig. 2(d).

Our second objective is to present an effective algorithm for inferring the FGA hierarchy, *i.e.* parse trees, from a single input image. Due to the flexibility of 3D objects in the space and their contextual relations, it is ineffective to use the prevailing sliding window methods for object detection, and it is also infeasible to search objects of all dimensions in an image pyramid, we adopt a Markov chain Monte Carlo method to optimize the Bayesian a posteriori probability. In the spirit of data-driven MCMC proposed by Tu and Zhu (2002), our parsing algorithm consists of a set of Markov chain dynamics which, in combination, can traverse the entire joint FGA space. For computational efficiency, these MC dynamics are driven by proposal probabilities computed in bottom-up steps.

1.2 Related work

Our method is related to four streams of research in the literature which we will briefly discuss in the following.

Stream 1: Scene representation. There are five major scene representations in the vision literature. (i) Representing scene as feature vectors for classification, such as the scene gist in Oliva and Torralba (2001), spatial pyramid matching (SPM) in Lazebnik et al (2006) and recent reconfigurable scene models by Parizi et al (2012) and Wang et al (2012). (ii) Region-based representations for semantic scene labeling. Conditional random fields Lafferty et al (2001) are widely used to represent semantic relations between adjacent regions, such

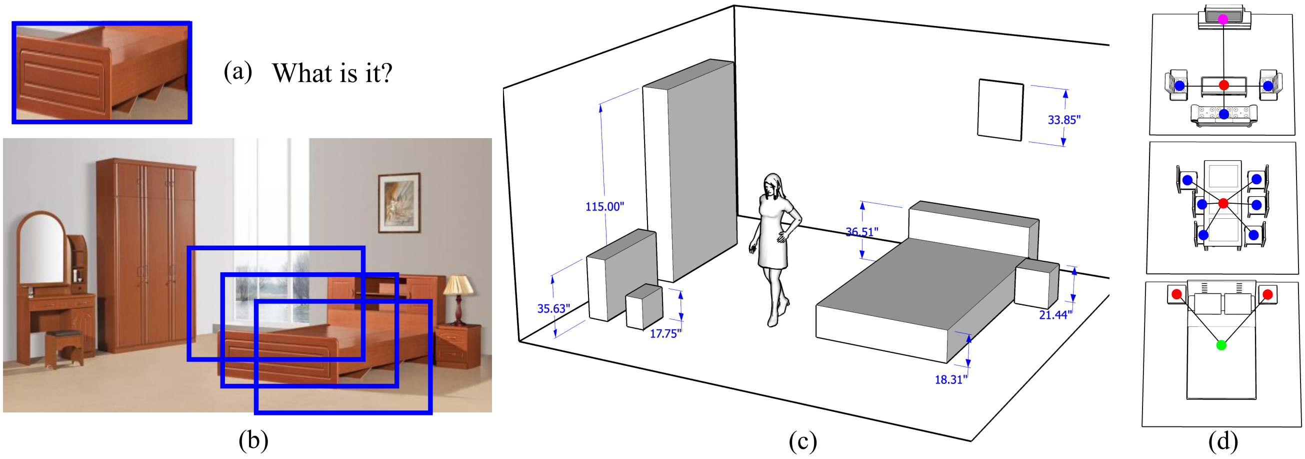


Fig. 2 (a) A large image window cropped from an input image in (b). The window is hardly recognizable in traditional appearance-based recognition but can be recognized in the whole scene. (c) An imagined human pose and estimated geometric sizes of objects in 3D, from which *functions* are reasoned; (d) *contextual relations* of functional objects as groups.

as {*inside, below, around, above* }. Choi et al (2010) studied 2D context models that guide detectors to produce a semantically coherent interpretation of a scene. They showed that such 2D horizontal contexts are very sensitive to camera rotations. (iii) Non-parametric representations for scene labeling, for example, label transfer by SIFT flow in Liu et al (2011), SuperParsing in Tighe and Lazebnik (2013a,b) and scene collage in Isola and Liu (2013) interpret a new scene by searching nearest neighbors from images in the scene dataset, and then transfer the label maps to the target through warping or contextual inference. Interestingly, Satkin et al (2012), Satkin and Hebert (2013) recently generalize the idea of nearest-neighbor search to the 3D scenes, so that their approach can recognize objects cross viewpoints. Lim et al (2013, 2014); Del Pero et al (2013) detected indoor objects by matching with fine-grained 3D CAD furniture models. Aubry et al (2014); Song and Xiao (2014) detects chairs by exemplar-SVM classifiers with a large set of synthetic training data, which rendered from 3D CAD models under various viewpoints. (iv) 3D block world representation, which allows reasoning about the physical constraints within the 3D scene. Gupta et al (2010) posed 3D objects as blocks and inferred their 3D properties such as occlusion, exclusion and stability in addition to surface orientation labels. They showed that a global 3D prior does improve 2D surface labeling. Hedau et al (2009, 2010, 2012), Wang et al (2010), Lee et al (2009, 2010), Schwing and Urtasun (2012); Schwing et al (2012, 2013) parameterized the geometric scene layout of the background and/or foreground blocks and trained their models by the structured SVM (or latent SVM). (v) Deformable part-based models: Hu (2012), Xiao et al (2012), Hejrati and Ramanan (2012), Xiang and Savarese (2012), Pepik et al (2012), Fidler et al (2012), Desai and Ramanan (2013) designed several new variants of the deformable part-based models to detect 3D entities under different view points.

Stream 2: Object functionality and affordance. In computer vision, Stark and Bowyer (1991) pioneered the use of functional properties in 3D object recognition. They parsed an objects into a 3D geometric description, and recognized the object by searching potential functional elements. Both developmental psychologist Oakes and Madole (2008) and computer vision researchers Yao et al (2013) demonstrated that functionality is at least as important as appearance in recognizing objects. More recently, numerous approaches have been proposed to detect functional objects based on human-object interactions in video. Wei et al (2013) and Jiang et al (2013) extracted human actions from RGBD video data and used the human actions as a prior to indirectly detect objects and label scenes. Bar-aviv and Rivlin (2006) and Grabner et al (2011) detected chairs by hallucinating agents in the 3D CAD data and depth data respectively. Gupta et al (2011) proposed an algorithm to infer the human workable space by adapting human poses to the scene. Delaitre et al (2012) and Fouhey et al (2012) recovered the semantics and geometry of a scene by observing human activities in the room. Kim et al (2014) learned an affordance model to predict a static pose that a person would need to adopt in order to use an object. Koppula and Saxena (2013) anticipated human activities using object affordance learned from RGBD videos Koppula and Saxena (2014).

Stream 3: 3D reconstruction from single 2D image. Automatic 3D reconstruction from a single image was considered an ill-posed problem. In order to recover a meaningful 3D reconstruction, researchers make assumptions about the scene and use prior knowledge to regularize the solution. In this research stream, people used four types of assumptions. (i) Sketch smoothness assumption: Han and Zhu (2004) was the first tackling this problem by assuming the local sketch smoothness and global scene alignment for recovering 3D objects, like plant, tree and buildings from 2D single image. (ii) Piece-wise smoothness assumption: Saxena et al (2009) presented a fully supervised method to learn a mapping between informative features and depth values under a conditional random field framework. Payet and Todorovic (2011) proposed a joint model to recognize objects and estimate scene shape simultaneously. (iii) Surface assumption: Hoiem et al (2009) recognized the geometric surface orientation and fit ground-line that separate the floor and objects in order to pop-up the vertical surface. Delage et al (2007) proposed a dynamic Bayesian network model to infer the floor structure for autonomous 3D reconstruction from a single indoor image. Mobahi et al (2011) extracted low rank textures of repeated patterns to construct surfaces like building facades. Recently, Fouhey et al (2014) proposed the use of convex and concave edges for regularizing scene configurations like playing Origami. (iv) Manhattan world assumption: Recent studies on indoor scene parsing, including Hedau et al (2009, 2010, 2012), Wang et al (2010), Lee et al (2009, 2010), Schwing et al (2012); Schwing and Urtasun (2012); Schwing et al (2013), Zhao and Zhu (2011, 2013) and Del Pero et al (2011, 2012, 2013) adopted the Manhattan world representation extensively. This assumption stated that man-made scenes are built on a cartesian grid and thus have regularities in the image edge gradient statistics. This enables us, from a single image, to determine the orientation of the viewer relative to the scene and also to recover scene structures which are aligned with the grid.

Most recently, a series of work, including Lin et al (2013); Choi et al (2013); Zhao and Zhu (2013); Guo and Hoiem (2013); Zhang et al (2014), proposed holistic approaches to exploits 2D semantic segmentation, 3D geometry, as well as 3D contextual relations in a joint framework.

Stream 4: Stochastic image grammar. This stream of research started from "syntactic pattern recognition" by K. S. Fu and his school in the late 1970s to early 1980s. Fu (1982) depicted an ambitious program of block world scene understanding using grammars. This stream was disrupted in the 1980s and suffered from the lack of an image vocabulary that is realistic enough to express real-world objects and scenes, and reliably detectable from images. Tu et al (2005) raised the notion of image parsing to the decomposition of an image into a hierarchical "parse graph" by a Data-Driven Markov Chain Monte Carlo sampling strategy. Zhu and Mumford (2007) proposed an And-OR graph model to represent the compositional structures in vision. Han and Zhu (2009) detected rectangular structures in man-made scenes by applying bottom-up / top-down grammar rules in a greedy manner. Porway and Zhu (2010) proposed a cluster sampling algorithm to parse aerial images by allowing for Markov chain jumping between competing solutions. A recent work Liu et al (2014) studied a probabilistic grammar model for labelling 3D CAD scenes.

This paper extends two preliminary conference papers Zhao and Zhu (2011, 2013) in the following aspects:

- Discuss the proposed Stochastic Scene Grammar comparing to other classic grammar models in Sect. 2;
- Describe more details about function, geometry and appearance models in Sect. 3;
- Explain the inference algorithm in terms of a functional jump move and three kinds geometric diffusion moves in Sect. 4;
- Extends the experimental analysis on convergence with different components and 3D reconstruction results in Sect. 5.

Parts of this work appear in two preliminary conference papers Zhao and Zhu (2011, 2013). The present paper describes our approach in more detail, discusses the connection to previous grammar models, extends the experimental analysis.

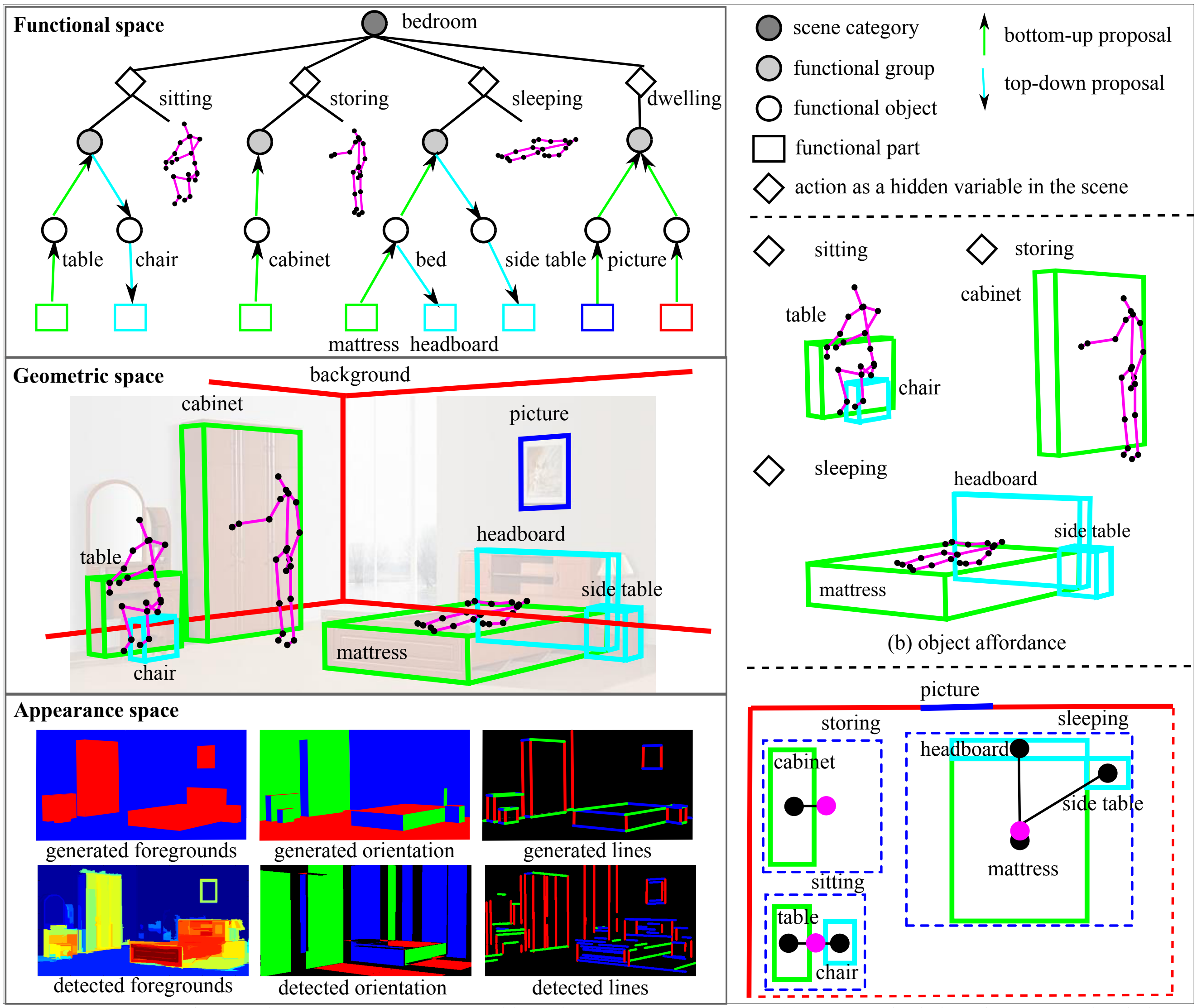
1.3 Overview of our approach

By analogy to natural language parsing, we pose the scene understanding problem as parsing an image in a hierarchical *parse tree* (or parse graph if we count on the spatial context relations) using the Stochastic Scene Grammar (SSG). Fig. 3 shows an example of the parse tree in a Function-Geometry-Appearance (FGA) hierarchy. In comparison to the literature reviewed above, this paper has three major contributions to the scene parsing problems.

(I) A Stochastic Scene Grammar (SSG).

The SSG starts from a root node for the scene category and ends in a set of terminal nodes (lines/regions) as is shown in Fig. 3. In between, we model all intermediate functional concepts and geometric entities by three types of production rules and two types of contextual relations. The latter are illustrated in Fig. 4.

Three types of production rules: *AND*, *OR*, and *SET*. (i) The AND rule in Fig. 4(i) encodes how subparts are composed into a larger structure. For exam-



(a) the Function-Geometry-Appearance hierarchy (c) contextual relations from an overhead view

Fig. 3 (a) The function, geometry and appearance (FGA) hierarchy in our proposed scene parsing grammar. The scene category (bedroom) at the root note is defined by the background and three most likely actions (sitting, storing and sleeping) in the scene. These actions impose the object affordance and contextual relations to the geometric entities. The final parsing result is evaluated on top of the synthesis of appearance likelihood maps. (b) The 3D human-object interactions. (c) The contextual relations between objects.

ple, three hinged rectangles form a 3D box, four linked line segments form a rectangle, a background and inside objects form a scene; (ii) The SET rule in Fig. 4(ii) represents an ensemble of entities, *e.g.* a set of 3D boxes or a set of 2D regions; (iii) The OR rule in Fig. 4 (iii) represents a switch between different sub-types, *e.g.* a 3D foreground and 3D background have several sub-types. Each type represents a geometric viewpoint, from which one can only see certain planes of a cuboid. The choice of OR triggers different branches of the AND rules, then combinations of them will become a SET rule, *i.e.* cuboid → plane1 · plane2 · plane3 | plane2 · plane4 | · · ·

Two types of contextual relations: *Cooperative* "+" and *Competitive* "-". If the visual entities satisfy a cooperative "+" relation, they tend to bind together, *e.g.* hinged rectangles of a foreground box showed in Fig. 4(a). In contrast, entities is a competitive "-" relation, they compete against each other for their presences in the parse tree, *e.g.* two exclusive (conflicting) foreground boxes competing for a same space in Fig. 4(b) and thus cannot both exist in a valid parse tree.

(II) A Function-Geometry-Appearance hierarchy.

We embed the FGA hierarchy in the syntactic grammar discussed above, and Fig. 3.(a) illustrates the FGA hierarchy in three layers.

Functionality. In the top layer, an indoor scene is defined by a small set of plausible human actions, and each action involves a few objects as a group. The table and chair (and the mirror) for a person to sit (and to make up face/hair), a bed with side table (and lamp) for people to sleep (and read). Here, each action is a composition of the 3D geometric relations between the pose and objects, as Fig. 3.(b) shows.

Geometry. The 3D sizes (dimensions) are used to evaluate how likely an object is able to afford a human action, known as the *affordance* in Gibson (1977). Fortunately, most furniture has regular structures, *i.e.* rectangular shapes, 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 is a better explanation of the image in terms of segmentation accuracy as shown at the bottom of Fig. 3. In the geometric space, each 3D shape is directly linked to a concept in the functional space. Shown in Fig. 3.(c), the contextual relations are utilized when multiple objects are assigned to the same functional group, *e.g.* a bed and a nightstand for sleeping. The distribution of the 3D geometry is learned from a large set of 3D models as shown in Fig. 7.

Appearance: The appearance of the furniture has large variations due to material properties, lighting conditions, and viewpoints. In order to ground our model on the input image, we detect and estimate line segments, surface orientations, and coarse foreground detection as the local evidences to support the geometry reasoning above as Fig. 3 illustrates.

(III) MCMC inference algorithm with reversible jumps.

We design a MCMC algorithm to simulate a Markov Chain to traverse the space defined by the FGA hierarchy in a data driven MCMC paradigm proposed by Tu and Zhu (2002).

The MCMC includes three types of dynamics for reversible jumps: i) add: sample a subtree and attach it to a non-terminal node randomly chosen from the current parse tree; ii) delete: delete a subtree whose root is a node randomly chosen from the current parse tree; iii) functional jump: switch a functional label of a node randomly on the current parse tree.

The inference algorithm also includes three types of geometric diffusion moves: i) α-diffusion: data-driven bottom-up detection that directly draws cuboid proposals from a non-parametric distribution built up by the line segments detected from the image; ii) β-diffusion: grammar-driven bottom-up prediction that proposes cuboid for a parent node in the parse tree from the children nodes by inversely computing a geometric transformation; iii) γ-diffusion: grammar-driven top-down prediction that proposes cuboid by top-down sampling for a child node in the parse tree from its parent node based on the geometric model.

2 Stochastic Scene Grammar

2.1 Background

A *Context-Free Grammar* is defined as $G = (S, V, R)$. $V = V^N \cup V^T$, and V^T is a finite set of terminal symbols, V^N is a finite set of non-terminal symbols (structures or sub-structures), $S \in V^N$ is a distinguished nonterminal called the start symbol, and R is a finite set of productions of the form $A \to BC$ or $A \to w$ in *Chomsky Normal Form* with no useless productions, where $A, B, C \in N$ and $w \in T$.

A set of all valid configurations C derived from production rules is called a *language*:

$$L(G) = \{C : S \overset{\{r_i\}}{\to} C, \{r_i\} \subset R, C \subset V^T\}. \tag{1}$$

A *Probabilistic Context-Free Grammar* (PCFG) is defined by a pair (G, P) consisting of a context-free grammar G and a real-valued vector P of length $|R|$ indexed by production ruless, where $P = P(\alpha \to \beta)$ is an expansion probability for each production rule $\alpha \to \beta \in R$. It is required that $P(\alpha \to \beta) \geq 0$ and $\sum_{(\alpha \to \beta) \in R} P(\alpha \to \beta) = 1$ for all nonterminals $\alpha \in V^N$. A *parse tree pt* is a set of nodes, each node has a chosen production rule $\alpha \to \beta$.

The probability of a parse tree is derived from the PCFG is defined as

$$P(pt|S) = \prod_{\alpha \in V^N} P(\alpha \to \beta) \tag{2}$$

2.2 Attributed Context-Sensitive Grammar

The *Stochastic Scene Grammar* in this paper is designed for modeling 3D scene structures for parsing a 2D image. Different from traditional language parsing problem, the 3D scene parsing faces two major challenges: 3D geometry and context sensitivity. Therefore, we modified the traditional grammar model in two aspects accordingly:

I) Geometry: The complexity of 3D scene parsing problem comes from the explicit modeling of 3D geometric arrangement of objects, while the language grammar only need to handle the left-right order of words.

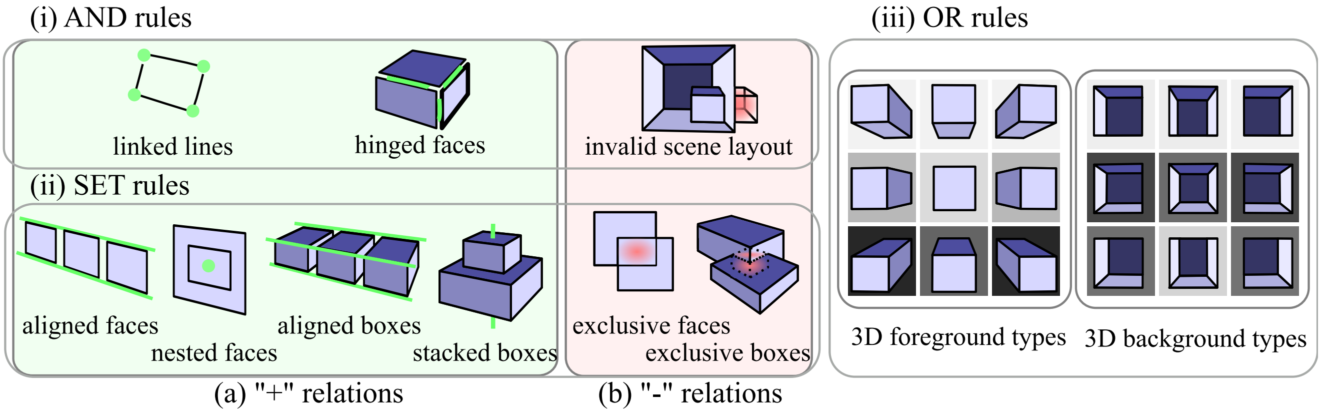


Fig. 4 Three types of production rules: (i) AND, (ii) SET, and (iii) OR; and two types of contextual relations: (a) cooperative "+" relations, and (b) competitive "-" relations.

We augment the nodes in the grammar with 3D geometric attributes, and thus extend it to attributed grammar. We represent each node at the end of the functional hierarchy by a 3D cuboid with three geometric attributes: size (3 DoF), relative position (3 DoF) and relative orientation (1 DoF).

II) Context sensitivity: There are two kinds of contexts: physical exclusion and graphical occlusion. The physical exclusion means each grammar node (such as an object) should be physically collision-free with all the other objects in the 3D scene, and the graphical occlusion means that all the grammar nodes compete with each other for explaining the image pixels with respect to the depth order in an image formation process. Thus the grammar becomes Context-sensitive which breaks the probabilistic derivation in Eq. 2 in the way that the image data not only depends on its direct parents but also be constrained by all the other notes in the image formation process. In particular, we explicitly model the image formation process in the inference stage by an analysis-by-synthesis paradigm.

Therefore, the SSG is attributed and context sensitive, for which traditional inference algorithm, such as inside-outside algorithm, are no longer applicable. Inspired by probabilistic models of cognition Tenenbaum et al (2011); Battaglia et al (2013); Mansinghka et al (2013); Goodman and Tenenbaum (2014), we design the two context-sensitive modules in a probabilistic program. At each MCMC iteration, a probabilistic sample is evaluated by recreating the image formatting process, we reconstruct a volumetric 3D scene and re-render a 2D image with a depth buffer.

2.3 Production rules

In this paper, we define three types of stochastic production rules R^{AND},R^{OR},R^{SET} to represent the structural *compositionality* and *reconfigurability* of visual entities. The compositionality is defined by the AND rules and the reconfigurability is expressed by the OR rules. A SET rule is a mixture of an OR rule and an AND rule.

(i) An AND rule ($r^{AND} : A \to a \cdot b \cdot c$) represents the *decomposition* of a parent node A into sub-parts a, b, and c. The probability $P(a, b, c|A)$ measures the compatibility (contextual relations) among sub-structures a, b, c and their parent A. As seen Fig. 4(i), the grammar outputs a high probability if the three rectangles of a 3D box are well hinged.

(ii) An OR rule ($r^{OR} : A \to a \mid b$) represents the *switching* between two sub-types a and b of a parent node A. The probability $P(a|A)$ indicates the preference for one subtype over others. Such as the 3D background in Fig. 4 (iii), the camera rarely faces the ceiling or the ground, hence, the three sub-types in the middle row have higher probabilities (darker color means higher probability). Moreover, OR rules also model the discrete number of entities.

(iii) A SET rule ($r^{SET} : A \to \{a\}_k, k \geq 0$) represents an *ensemble* of k visual entities with k being a integer from a finite set. The SET rule is equivalent to a mixture of an OR rule and an AND rules ($r^{SET} : A \to \emptyset \mid a \mid a \cdot a \mid a \cdot a \cdot a \mid \cdots$). It first chooses a set size k by OR, and forms an ensemble of k entities a_k by AND. Those entities are not necessarily to be identical to each other, because successive rules may be further branched out with different properties, such as different size or different configurations. It is worth noting that the OR rule essentially changes the graph topology and dimensionality of the output parse tree by changing the number of nodes k.

As a result, the AND, OR, SET rules generate various functional concepts and geometric entities which satisfy contextual relations as seen in Fig. 4

2.4 Contextual relations

There are two kinds of contextual relations, *Cooperative* "+" relations and *Competitive* "-" relations, which are involved in the AND and SET rules.

(i) The cooperative "+" relations specify the *concurrent* patterns in a scene, *e.g.* hinged rectangles, nested rectangle, aligned windows in Fig. 4(a). The visual entities satisfying a cooperative "+" relation tend to bind together. The cooperative "+" relation is introduced by either functional context in Sect. 3.2 or geometric decomposition in Sect. 4.2.1.

(i) The competitive "-" relations specify the *exclusive* patterns in a scene. If entities satisfy competitive "-" relations, they compete with each other for presence. As shown in Fig. 4(b), if a 3D box is not contained by its background, or two 2D/3D objects are penetrating with one another, these cases will rarely be in a solution simultaneously. The "-" relations is introduced by physical constraints in Sect. 3.2.

If several visual entities satisfy a cooperative "+" relation, they tend to bind together as *tight structures*. The "tight structures" is like a template, where parts maintain a rigid spatial relation. We group visual entities into these tight structures as much as possible in the early stage of inference according to the geometric decomposition (Sect. 4). The *loose structures* only need to satisfy certain competitive "-" constraints, *e.g.* they can not penetrating each other. The combinations of parts in a loose structure are sampled in a later stage of inference (Sect. 4). The high-level functional concept will also impose "+" relations in the later stage of inference. If an object is assigned with functional label, then the algorithm will be able to sample its parts or nearby objects according to the 3D contextual relations as explained in Sect. 3.2.

With the three production rules and two contextual relations, the SSG is able to handle an enormous number of scene configurations and large geometric variations, which are the major difficulties in our task.

2.5 Bayesian formulation of the SSG

We define a posterior probability for a solution (a parse tree) pt conditioned on an input image I.

$$pt^* = \arg\max P(pt|I) = \arg\max \frac{1}{Z}\exp\{-E(pt|I)\} \quad (3)$$

where Z is a normalizing constant.

We use a probabilistic graphical model of an And-OR graph proposed by Zhu and Mumford (2007) to formulate the posterior probability, which decomposes the energy $E(pt|I)$ into three potential terms on the And, Or, terminal rules respectively.

$$\begin{aligned} E(pt|I) = &\sum_{v\in V^{OR}} E^{OR}(A_T(Ch_v)) \\ &+ \sum_{v\in V^{AND}} E^{AND}(A_G(Ch_v)) \\ &+ \sum_{\Lambda_v\in\Lambda_I, v\in V^T} E^T(I(\Lambda_v)) \end{aligned} \quad (4)$$

In the above notation, Ch_v is the set of children nodes of v, Λ_I is the image domain, and Λ_v is the image domain occupied by node v. A_G is the geometric attributes for the child nodes under an And-node, and A_T is the type attribute for child nodes under an Or-node.

(i) **The energy for an OR-node** is defined over a discrete variable such as "type" attribute under the Or-node, and reflects the prior probability for switching to each branch $r : v \to Ch_v$.

$$\begin{aligned} E^{OR}(A_T(v)) &= -\log P(v\to A_T(v)) \\ &= -\log\{\frac{\#(v\to A_T(v))}{\sum_{u\in Ch(v)}\#(v\to u)}\}. \end{aligned} \quad (5)$$

where $\#(r)$ denotes the number of the production rule r appeared in the training dataset. The switching branches for foreground objects and the background layouts is shown in Fig. 4 (iii).

(ii) **The energy for an And-node** specifies geometric relationships among a parent AND node and its children. The design of the graph "cliques" among the nodes is problem specific, such as Markov Random Fields model among children, or a star model with the parent node as the center. Usually, the tree structured models have advantages for exact inference known as the pictorial structure or deformable part-based model Felzenszwalb and Huttenlocher (2003); Felzenszwalb et al (2010).

We define both cooperative "+" relations and competitive "-" relations to represent the mutual contexts.

$$E^{AND}(A_G(Ch_v)) = \lambda^+ h^+(A_G(Ch_v)) + \lambda^- h^-(A_G(Ch_v))(6)$$

where $h(*)$ are sufficient statistics of the exponential model, λ^+ and λ^- are their parameters. They can be either numeric values or vectors. For example, if we model two objects within a functional group, we first define a cooperative "+" relation by a Gaussian distribution of objects' positions with respect to its parent's coordinates; we then define a competitive "-" relation which adds penalties when two objects penetrating with each other or an object is out of its parent's range.

(iii) **The energy for a terminal node** is defined over image features $I(\Lambda_v)$ on the image area Λ_v. The features used in this paper include: (a) a foreground map, (b) a 3D orientation map, (c) a line segment map shown in the bottom of Fig. 3.(a). This term only captures the features from their image area Λ_v, and avoids the double counting of the shared edges and the occluded regions as discussed in Sect. 3.3.

3 Integrating function, geometry and appearance in the SSG

The previous section overview stochastic context sensitive grammar and its general probabilistic formulation. In this section, we elaborate on how the SSG integrates the three layers of concepts in the functional space, the geometric space and the appearance space.

In this section, we will explain two production rules (the functional set rule, the affordance rule) for generating graph nodes on the grammar. And an image formation process that evaluates the generated 3D scene by rendering the synthetic image.

3.1 The functional space

The grammar model has advantages to handle the compositionally of the visual entities as well the dimensional changes of the scene. For example, it is common that a bedroom either has one bed or has two beds. The traditional grammar deals with the dimensional change by recursive production rules, such as $A \to \alpha \cdot A$. The production rules defined in the functional space are not recursive.

We introduce a set rule in functional space as

$$v \to \{l, \{G(u_i) : i = 1 \cdots \#(l)\} : l \in L\} \tag{7}$$

where l is a label of a child node from a label set L, $\#(l)$ is a number variable controlling the number of objects for each label l. The set rule is a nested OR-AND node. As shown in Fig. 5, the number variable decides the dimensionality of the parse tree. Therefore the production of the functional set rule can generate various parse trees with different dimentionalities.

Thus, the probability distribution of each production rule $P(r : v \to Ch_v)$ in Eq. 3 is unfolded as

$$P(v \to Ch_v) = \prod_{l \in L} \left[P(\#(l)) \prod_{i \in \{1 \cdots \#(l)\}} P(G(u_i)|l) \right] \tag{8}$$

The geometric attributes $G(u_i)$ of each object u_i is defined in the geometric space.

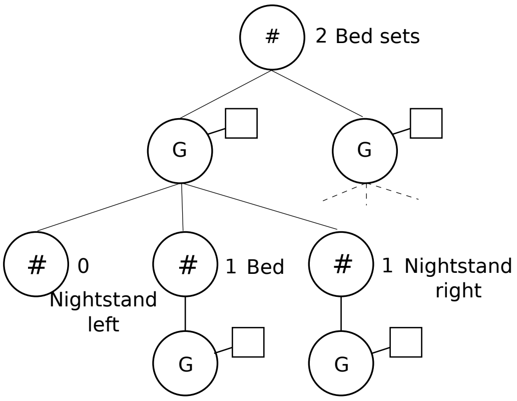


Fig. 5 An example parse tree generated from the grammar with the set rule. The dimensionality of a parse tree is decided by number variables for each label.

3.2 The geometric space

We model each geometric entities in the grammar as a 3D cuboid.

Each 3D cuboid is encoded by three geometric attributes including 3 DoF size $Size(v)$, 3 DoF relative position $Pos(v)$ and 1 DoF relative orientation $Ori(v)$,

$$G(v) = \{Size(v), Pos(v), Ori(v)\} \tag{9}$$

Object affordance $p(Size(v)|l(v))$ models the distribution of geometric attributes of each functional object, for example, how large the bed mattress is, how far the bed is from the wall. If we consider human actions as hidden variables in the space, then the affordance probability measures how likely the geometric shape of an object is able to afford an action. As shown in Fig. 3, 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, relative position, and relative orientation of functional objects by a mixture of Gaussians respectively, such as

$$p(Size(v)|l(v)) = \sum_{i=1}^{K} a_i N(\mu_i, \Sigma_i) \tag{10}$$

where the a_i is the mixture coefficient of each Gaussian $N(\mu_i, \Sigma_i)$. The model characterizes the sub-category of the geometry, which allows for simultaneous alternatives of canonical sizes, such as king size bed, full size bed *etc.* We estimated the model by EM clustering, and we manually picked a few typical samples as the initial mean for the Gaussian, *e.g.* a coffee table, a side table and a desk from the table category.

The contextual relations are defined with respect to the relative position $Pos(v)$ and relative orientation $Ori(v)$. The relative position is the position of a child

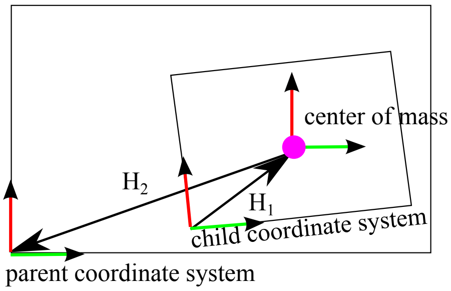


Fig. 6 The geometric transformation between a child coordinate system and a parent coordinate system

with respect to the parent coordinate system. The relative orientation is the orientation of the child with respect to the reference orientation of the parent.

The absolute coordinates of an object can be calculated recursively along the grammar productions. We showed an example of the geometric transformation between a child coordinate system X and a parent coordinate system X' in Fig. 6. The transformation can be decomposed as two independent transformation H_1 and $H2$. The H_1 represent the transformation from child coordinate system to its center of mass coordinate system, and the H_2 represents the transformation from its center of mass coordinate system to its parent coordinate system. The geometric transformation equation is calculated by

$$\begin{aligned} X' &= H_2 H_1 X \\ &= \begin{bmatrix} cos(Ori) & sin(Ori) & 0 & Pos_x \\ -sin(Ori) & sin(Ori) & 0 & Pos_y \\ 0 & 0 & 1 & Pos_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & Size_x/2 \\ 0 & 1 & 0 & Size_y/2 \\ 0 & 0 & 1 & Size_z/2 \\ 0 & 0 & 0 & 1 \end{bmatrix} X \end{aligned} \tag{11}$$

In order to learn the geometric model, we collected a dataset of functional indoor furniture, as shown in Fig. 7. The functional objects in the dataset are modeled with real-world measurements, and therefore we can generalize our model to real images by learning from this dataset. We found that the real-world 3D sizes of the objects has less variance than the projected 2D sizes. As we can see, these functional categories are quite distinguishable solely based on their geometric shapes as shown in Fig. 8. For example, the coffee tables and side tables are very short and usually lower than the sofas, and the beds generally wider than others.

In this version of algorithm, we directly model the cooperative "+" relations as parent-child geometric relationship as discussed above without explicitly addressing the cooperative "+" relations among child nodes as Zhao and Zhu (2011). This parent-child relation facilitates the inference algorithm traveling along the depth of the hierarchy, *e.g.* the top-down prediction and bottom-up prediction in Sect. 4.

However, we still model the competitive "-" relations specify penalties or constraints among child nodes. The sufficient statistics is defined on the penetrating rate between their occupied 3D spaces $G(.)$ to penalize penetrating objects.

$$h^-(G(Ch_v)) = \sum_{a,b \in Ch_v} (G(a) \cap G(b))/(G(a) \cup G(b)) \tag{12}$$

3.3 The appearance space

The functional and geometric hierarchies are generative models on a cartoon like image with line segments for object boundaries and labeled regions for object surfaces. There is still a gap between the synthesized cartoon scene and the observed image. To fully explain (reconstruct) the input image, we need to know the lighting conditions, textures and material properties for object surfaces. This is a challenge problem which is beyond the scope of this paper.

To circumvent the tasks of modeling and inferring textures and lighting conditions, we use of set discriminative methods to detect some intermediate results in the following.

– a map of line segments detected by an algorithm proposed in Von Gioi et al (2010);
– a foreground/background label map computed by an approach used in Hedau et al (2009); and
– a surface orientation map calculated by an approach in Lee et al (2009).

Thus instead of grounding our model on raw pixels, we define the likelihood model on these 2D label maps using a Sum of Squared Difference (SSD) function $d()$.

$$\begin{aligned} p(I_{obs}|pt) &= p(I_{label}|I_{syn} = f(V^T)) \\ &= 1/Z * \exp\left(-\sum_{i \in 1 \cdots 3} \lambda_i d(I^i_{label}, I^i_{syn})\right) \end{aligned} \tag{13}$$

where $f(V^T)$ is a rendering function of all the terminal nodes V^T. The rendering function generates the synthesized image I_{syn}, and the likelihood is defined how likely the parse graph generate label maps I_{label}.

For example, the appearance space is illustrated at the bottom of Fig. 3(a). The figure shows the three detected line segment map, foreground segmentation map, and orientation map from left to right. Above the three maps are the corresponding maps rendered from the parse tree *pt*. Once a parse tree *pt* is decided, the algorithm projects the 3D geometric entities V^T on the parse tree to the 2D image plane with respect to the

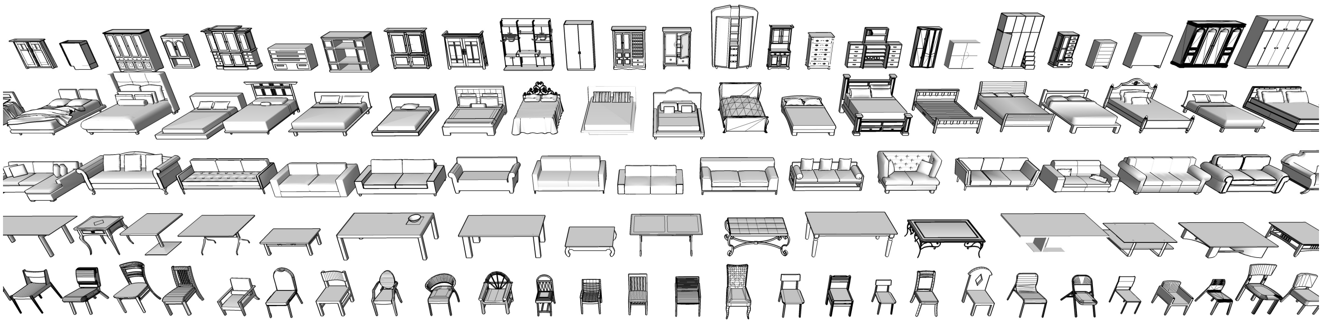


Fig. 7 Examples of 3D indoor furniture products collected from the Trimble 3D Warehouse

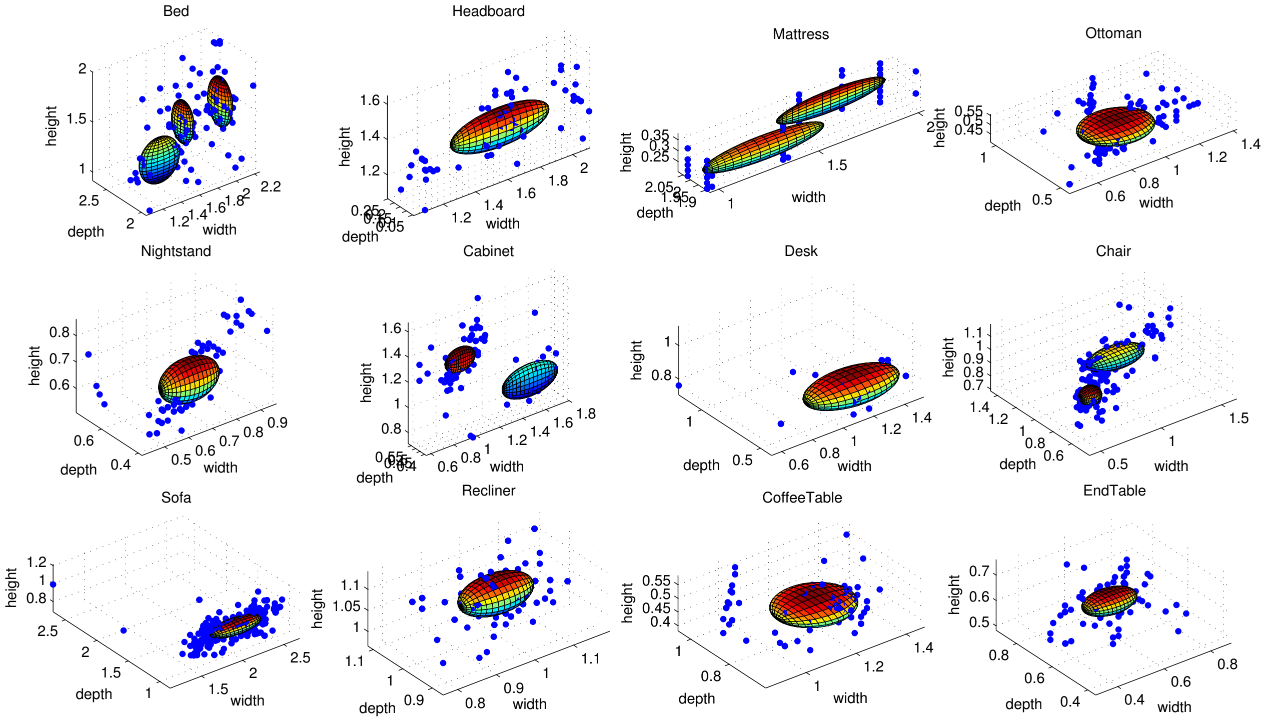


Fig. 8 The empirical and fitted distributions of the 3D sizes of some functional objects in meters plotted in 3D spaces.

relative depth order and camera parameters. The projection is implemented with OpenGL.

4 Inference algorithm

We design a top-down/bottom-up algorithm to infer an optimal parse tree *pt*. The compositional structure of the continuous geometric parameters and discrete functional labels 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.

4.1 Reversible Jumps

In this paper, we design Markov chains with reversible jumps (RJMCMC) algorithm to construct the parse tree and re-configure it dynamically using a set of moves. Formally, our scene parsing algorithm simulates a Markov chain $\mathcal{MC} =< \Omega, v, \mathcal{K} >$ with kernel $\mathcal{K}$ in space Ω and with probability v for the starting state. We specify stochastic dynamics by defining the transition kernels of reversible jumps. For each Markov chain move is defined by a kernel with a transition matrix $\mathcal{K}(pt^*|pt : I)$, which represents the probability that the Markov chain make a transition from state pt to pt^* when a move is applied.

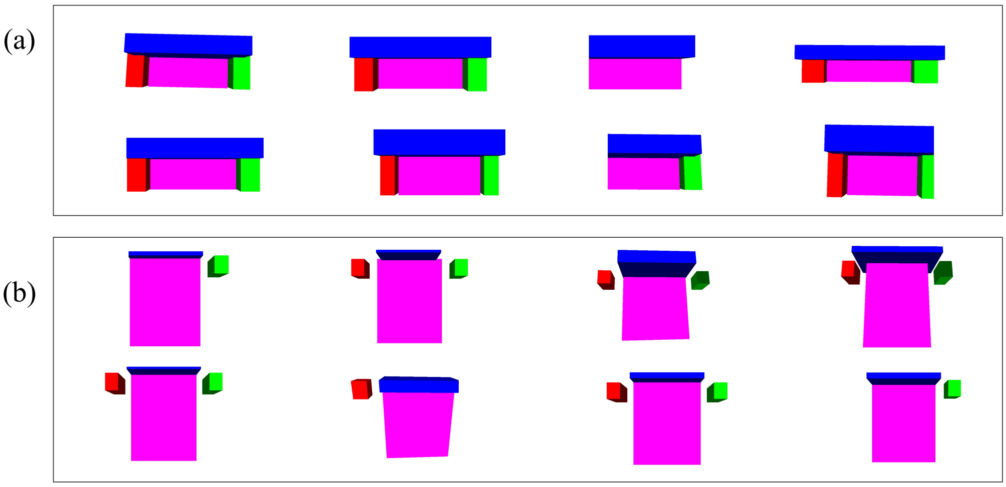


Fig. 9 Samples drawn from the distributions of 3D geometric models (a) the functional object “sofa” and (b) the functional group “sleeping”.

The kernels are constructed to obey the detailed balance condition:

$$p(pt|I)\mathcal{K}(pt^*|pt : I) = p(pt^*|I)\mathcal{K}(pt|pt^* : I). \tag{14}$$

Kernels which change the graph structure are grouped into reversible pairs. For example, the kernel for node creation $\mathcal{K}_+$ is paired with the kernel for node deletion $\mathcal{K}_-$ to form a combined move of node switch. To implement the kernel, at each time step the algorithm randomly selects the choice of move and then uses kernel $\mathcal{K}(pt^*|pt : I)$ to select the transition from state pt to state pt^*. Note that the probability $\mathcal{K}(pt^*|pt : I)$ depends on the input image I. This distinguishes our algorithms as a Data-Driven MCMC from conventional MCMC computing (Tu and Zhu (2002); Tu et al (2005)).

The kernel is designed using proposal probabilities and correspondent acceptance probability.

$$\mathcal{K}(pt^*|pt : I) = Q(pt^*|pt : I)\alpha(pt^*|pt : I) \tag{15}$$

The acceptance probability follows:

$$\alpha(pt \to pt^*) = min\{1, \frac{Q(pt|pt^*, I)}{Q(pt^*|pt, I)} \cdot \frac{P(pt^*|I)}{P(pt|I)} J_{f_{pt\to pt^*}}\} \tag{16}$$

$J_{f_{pt\to pt^*}}$ is the Jacobian of the dimension matching function. $f_{pt\to pt^*}$ is the dimension matching function. It is used to map the variables at dimensionalities of pt and pt^* into a space of common dimensionality. It is usually done by introducing additional $pt^* - pt$ parameters, or projecting out the corresponding $pt^* - pt$ parameters. Notice that each variable in Δpt is independently sampled from pt, hence the Jacobian is 1 in this case (Yeh et al (2012)).

The Metropolis-Hasting form ensures 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 local optimal peaks. The parse tree is initialized with random number of object and random geometric properties. During each iteration, if a proposal increases the posterior probability with respect to the proposal ratio, the move is taken. Otherwise, the move is taken only with a certain probability, which decreases over time. Hence early on the algorithm will tend to take moves even if they don’t improve the probability. Later on, the algorithm will only make moves which improve the posterior probability. The temperature function used is: $T(n) = 1000/n$ where n is the iteration number.

4.2 Generating data-driven 3D proposals

The algorithm starts from detecting straight line segments by Von Gioi et al (2010). Based on the Manhattan assumption, we group the line segments into N groups, each of which is correspondent to a vanishing point. We then select three dominate orthogonal vanishing point to build our coordinate system. We assume the camera parameters are reliably calibrated in this step, the calibration algorithm is discussed in Sect. 4.2.3.

We incrementally group noisy line segment into larger geometric structures. The 2D rectangles are formed by filtering over the combinations of two pairs of parallel lines or T junctions. As shown in Fig. 10, we first define five normal directions: facing down, facing left, facing front, facing right and facing up according to the vanishing points. The normal direction facing back is

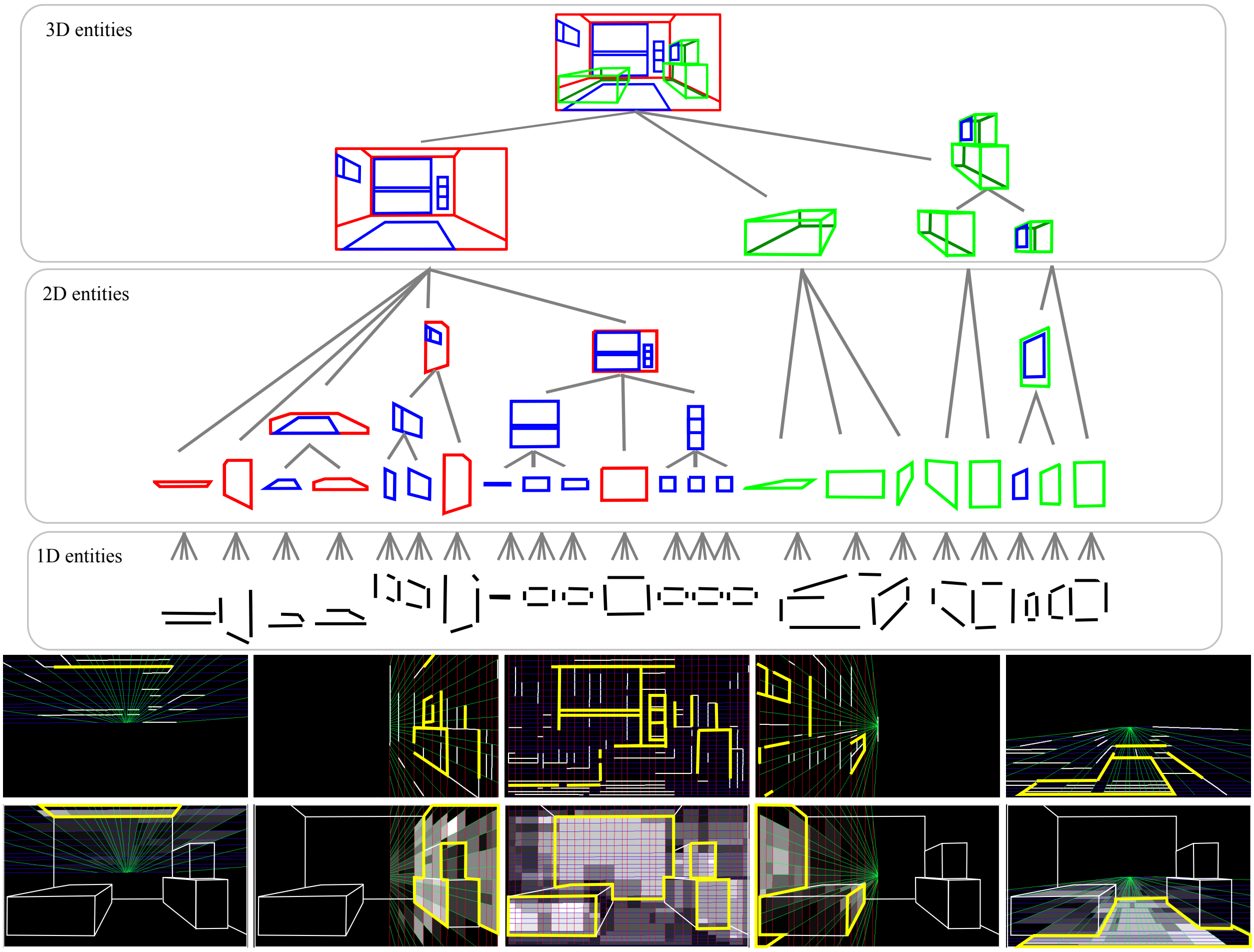
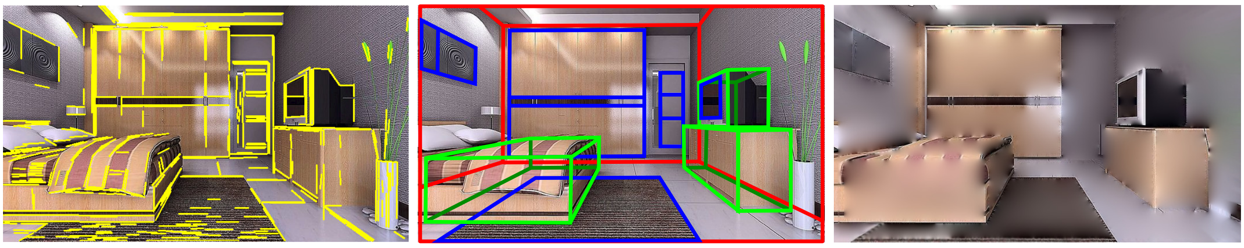


Fig. 10 The decomposition of geometric parse tree. The ten images on the bottom show the likelihood of the parse graph calculated and quantized by the five major orientations, whose normal directions point to down, left, front, right, and up respectively. The first five images show line segments (yellow) detected on their corresponding orientations, and the second five images show region likelihood calculated on their correspondent orientations. The lighter a cell, the higher the probability is. The yellow contours outline the inferred regions.



(a) input image with line segments (b) geometric parsing result (c) image reconstructed via sturectures in (b)

Fig. 11 Input image and output results of the geometric parsing.

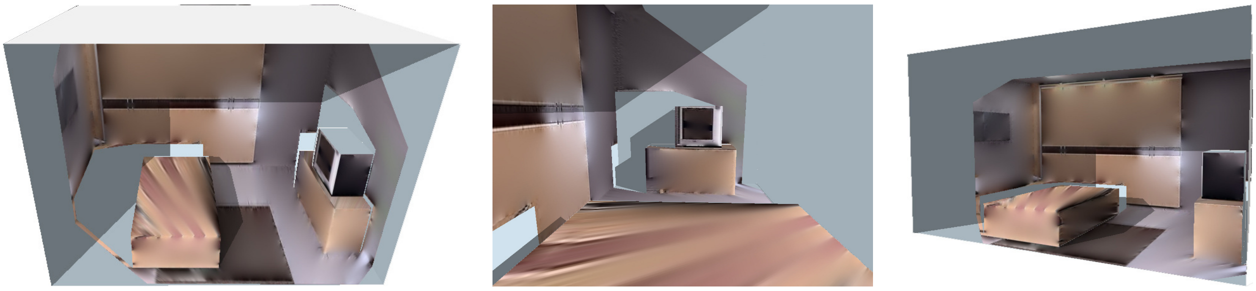


Fig. 12 3D synthesis of novel views based on the parsing result in Figures 10 and 11. The reconstructed errors of the bed is made clear in the novel views.

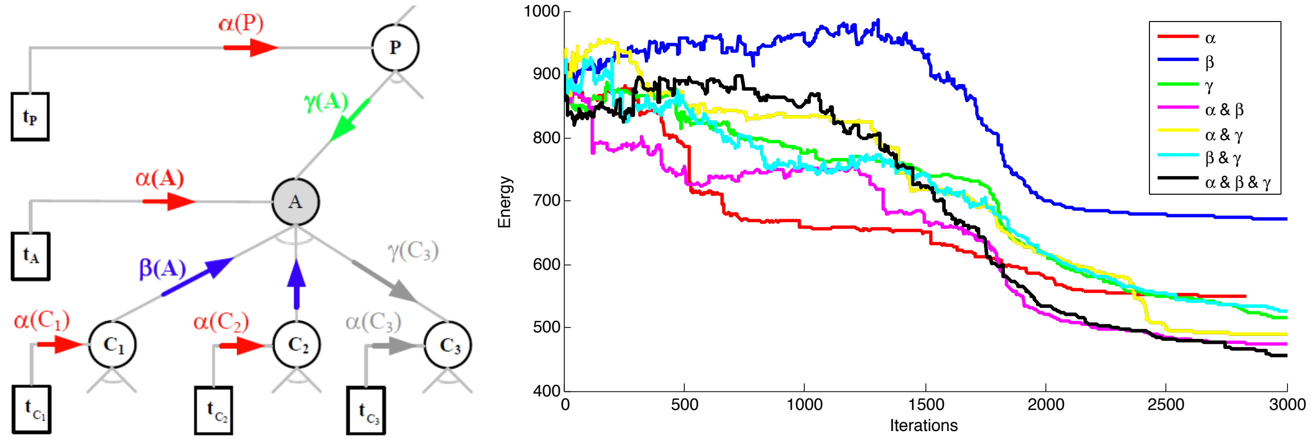


Fig. 13 The bottom-up top-down proposals for geometric diffusion moves. Our inference algorithm generates three kinds of geometric diffusion proposals: α: bottom-up detection, β: bottom-up prediction, and γ: top-down prediction. The plot on the right panel shows the average energy convergence of hundreds of Markov Chains using different proposal strategies: By only using the α diffusion from bottom-up detection (red curve), the Markov chain converges very fast at the beginning, but cannot keep reducing the energy due to limitation of bottom-up detections. Using the β diffusion from bottom-up prediction (blue curve) is the worst strategy, because if the terminal node can not be optimized, the prediction from bottom-up can be very bad. The black curve which combines three diffusions together is the best strategy, it has sufficient exploration at the beginning, and gradually converges to the lowest energy. Besides that, the combination of $\alpha\&\beta$ (magenta curve) and the combination of $\alpha\&\gamma$ (yellow curve) achieve good results which are very close to the black one.

```
Data: an input 2D image
Result: an output parse tree
Calculating data-driven 3D proposals;
while the rejection time larger than $K$ do
    Choose one of the following moves randomly;
      – add an entity
      – delete an entity
      – diffuse geometric attributes of an entity
    if add/remove a non-terminal node then
        Recursively add/remove its children;
    end
    if diffuse a non-terminal node then
        Choose one of the following geometric diffusion
        moves randomly;
          – $\alpha$ diffusion from bottom-up detection
          – $\beta$ diffusion from bottom-up prediction
          – $\gamma$ diffusion from top-down prediction
    end
    Calculate the posterior probability and validate
    the solution by projecting the 3D parse tree to the
    2D image plane;
    Accept/reject the new parse tree with the
    acceptance probability;
end
Return the parse tree with the highest posterior;
```

Algorithm 1: Inference algorithm

not visible from the camera position. All the 2D line segments are aligned on the mesh for each normal orientation. And surface orientation maps and foreground maps are also projected to each cell. And our algorithm goes over each rectangle on the mesh and calculates a local likelihood normalized by the size of the rectangle according to Eq. 13. In this way, we detect an exhaustive set of 2D rectangle candidates by applying a threshold for a high recall rate. Similarly, the cuboids are formed by filtering over the combinations of any two hinged rectangles, a threshold is applied to the distance between rectangle corners to evaluate how well the structure is formed. Please refer to Zhao and Zhu (2011) for more details.

4.2.1 The composition of 3D geometric entities

As shown in Fig. 10, the geometric space $\mathcal{G}$ contains the geometric entities of 3D cuboids, 2D rectangles and 1D line segments. Each entity is composed by several lower dimensional shapes. The detection of 3D entities starts from detection of line segments in the 2D image space as shown in Fig. 11(a). The composition of the geometric entities is coded by a series of AND rules where the relations between children nodes are set to a constraint within a threshold. The threshold is set to 5 pixels in the image, which means we tolerate 5 pixels offset between those rigidly combined components. The OR rule also plays a role by representing alternative ways of composition under different the view points. The production rules of geometric composition is illustrated in Fig. 10. We project all the terminal primitives to five normal directions as discussed above.

4.2.2 The calculation of marginal likelihood

The probability of the proposal is calculated by local marginal likelihood based on the bottom-up image labelling results. In order to properly quantize the geometric space and speed up the computation, we first group detected line segments into three main groups corresponding to three vanishing points. Then we further group the line segments into a series of rays pointing from the vanishing points to each line segments. We enforce the angle between two nearby rays to be larger than 2°, therefore line segments along the same orientation will be grouped together. We will also interpolate rays between two nearby rays if the angle between them are larger than 5°. Any two groups of rays will form an oriented mesh as shown at the bottom of Fig. 10. This quantization process guarantee that each detected line will be represented by several pieces of edges on the mesh, and each pixel will fall into a cell as well. In this way, the line/region likelihood of bottom detection is stored in the quantized meshes for each surface orientation. The brighter the intensity the higher the likelihood for each cell.

At the bottom of Fig. 10, there are ten images. The yellow lines on the first row of images represent the activated line segments. The line segment is activated when the geometric parsing result in Fig. 11.(b) match with the bottom-up detection result in Fig. 11.(a). The edge probability measures how many line segments are activated, which implicit encourages more line segment to be explained by final parsing result. the region with yellow boundary on the lower penal represent the activated surface region. A surface region is activated only the surface orientation is matched with geometric parsing results in Fig. 11.(b) by considering the depth ordering. The depth ordering guarantee the occluded region will not affect the likelihood of parsing result. Therefore, the quantization of image likelihood not only accelerates the inference process by a lookup table of precomputation, but also avoids the double counting of the shared edges and the occluded regions.

From the geometric primitives and their line segments, we can reconstruct the 2D image using a primal sketch model proposed in Guo et al (2007) which was also used in Han and Zhu (2009) for scene synthesis. Fig. 12 further shows the novel views of the synthesized (reconstructed) 3D scene as a verification. More 3D reconstruction results are shown in Fig. 16.

4.2.3 Single View 3D Scene Reconstruction

After detect each 3D line drawing cuboid, we need to recover the 3D geometric shape in the real world scale for each proposal. It enables us to perform inference on the 3D world.

Camera calibration: We cluster line segments to find three vanishing points whose corresponding dimensions are orthogonal to each other Hedau et al (2009). The vanishing points are then used to determine the intrinsic and extrinsic calibration parameters Criminisi et al (2000); Hartley and Zisserman (2004). 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 three vanishing points are visible and finite in the same image, then 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 Hartley and Zisserman (2004).

3D 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. Any other plane contacting the floor can be inferred by its contact point with the floor. Then we can gradually recover the whole scene by repeating the process from the bottom up. If there is any object too close to the camera to see the bottom, we will put it 3 feet away from the camera.

4.3 The top-down and bottom-up MCMC inference

We design a four-step MCMC algorithm that enables a Markov chain travel up and down through the FGA hierarchy. In each iteration, the algorithm proposes a new parse tree pt^* based on the current one pt according to the proposal probability.

We design jump and diffusion methods to ensure the ergodicity of the Markov Chain. There are two kinds of functional jump proposals: add and delete. The functional jump proposals change the dimensionality of the parse tree. Two kinds of geometric diffusion proposals: α diffusion, β diffusion, and γ diffusion. The α diffusion: data-driven bottom-up detection that directly draws cuboid proposals from a non-parametric distribution built up by the line segments detected from the image; β diffusion: grammar-driven bottom-up prediction that proposes cuboid for a parent node in the parse tree from the children nodes by inversely computing a geometric transformation; γ diffusion: grammar-driven top-down prediction that proposes cuboid by top-down sampling for a child node in the parse tree from its parent node based on the geometric model.

4.4 The functionally jump proposal

This step re-assigns functional number variables. The switching of functional labels can be happened in any layers of the functional parse tree as shown in Fig. 3, and number variables

4.4.1 The add proposal

The add proposal samples a subtree pt_v from a non-terminal node $v \in V^N$ randomly chosen from the current parse tree;

$$Q_+(pt \to pt^*) = p(v \in pt)p(pt(v)) \tag{17}$$

The proposal first chooses a node $v \in pt$ in grammar randomly. The $p(pt(v))$ is a recursive derivation of production rules from node v. $\prod_{\alpha_0=v} P(\alpha_i \to \beta_i)$

4.4.2 The delete proposal

The detect proposal removes a subtree whose root $v \in pt$ is a node randomly chosen from the current parse tree.

$$Q_-(pt \to pt^*) = p(v \in pt) \tag{18}$$

Similarly the delete proposal is calculated by choosing a node v from pt, which is discrete uniform distribution.

Both add proposals and delete proposals essentially change the dimensionality of the parse tree. In order to simplify the Jacobian in Eq. 16 of RJMCMC, we designed these jumps Δpt as independently samples from pt so that the Jacobian is 1 in this case (Yeh et al (2012)).

4.5 The geometrically diffuse proposal

We also defined three kinds of geometric diffusions.

4.5.1 α bottom-up detection proposal

As mentioned in the initialization step, we group the line segments to reconstruct 3D cuboid proposals. Each cuboid proposal is assigned with a weight indicating the local likelihood of this proposal. We further process the cuboid proposals by building a non-parametric distribution of the cuboid proposals. The non-parametric distribution is approximated by a weighted KDE (kernel density estimation). Since different objects have different distributions of sizes, we filter all cuboid proposals by the sizes of different objects and combine the score with the original weights, to generate different cuboid distributions for specific objects.

$$Q_\alpha(pt \to pt^*) = p(v \in pt)p_{KDE}(G(v)|I_{obs}) \tag{19}$$

The $p_{KDE}(G(v)|I)$ is a nonparametric probability distribution estimated by Kernel Density Estimation (KDE) of detected object proposals

$$\begin{aligned} p_{KDE}(x|I_{obs}) &= \sum_{i=1}^{n} w_i K_h(x - x_i) \\ &= \frac{1}{\sum_{i=1}^{n} P(x_i|I_{obs})} \sum_{i=1}^{n} P(x_i|I_{obs}) K_h(x - x_i) \end{aligned} \tag{20}$$

where $x_i, i \in 1 \cdots n$ are geometric entities detected from Sect. 4.2, and the KDE estimates the non-parametric distribution by considering the local marginal likelihood of each geometric entities $P(x_i|I_{obs})$. h is the window parameter of the kernel $K_h(\cdot)$.

4.5.2 β:: bottom-up prediction proposal

The bottom-up prediction refine a higher-level structure $par(v)$ of an existing child v, such as proposing the geometry of a bed set given the geometry of a bed.

$$Q_\beta(pt \to pt^*) = p(v \in pt)p(Pos(v))p(Ori(v))p(Size(par(v))) \tag{21}$$

This proposal calculate a node's parent by re-sampling the relative position of the child $Pos(v)$ and relative orientation of the child $Ori(v)$ with respect to its parents's coordinate system. And the result coordinates of the parent is calculated by the inverse transformation of Eq. 11: $X' = (H_2H_1)^{-1}X$. The size of the parent node Size(par(v)) is then sampled independently.

4.5.3 γ: top-down prediction proposal

The top-down prediction, from another hand, refine a lower-level structure. This is very useful for the heavily occluded object in a functional group. For example, once a bed is correctly detected, this proposal will try

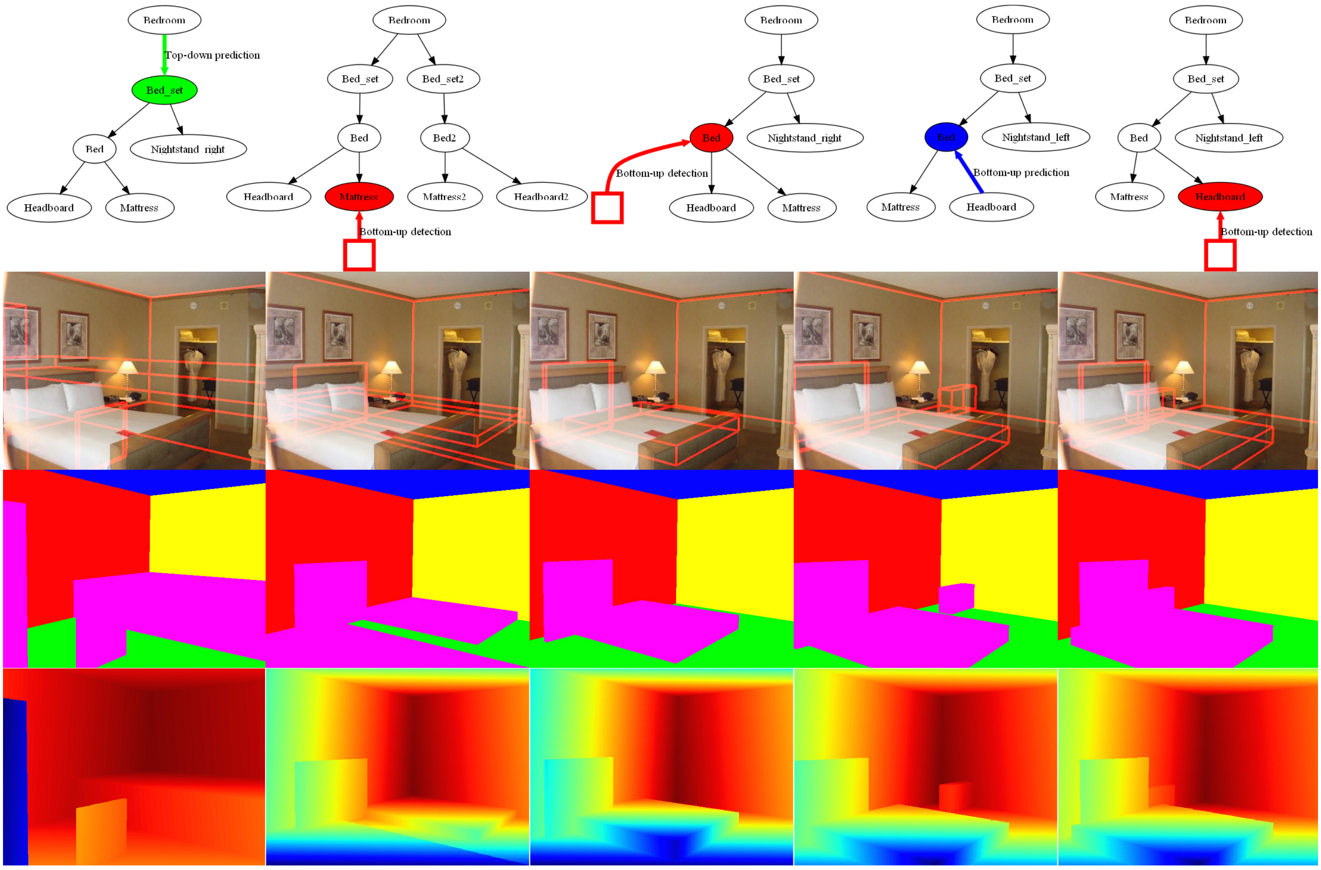


Fig. 14 Qualitative results of bottom-up top-down Inference. These pictures are overlaid images, label maps, depth map and their corresponding parse trees for 250, 500, 750, 1000, 1250 accepted moves. In particular, the red, blue, and green arrows on the parse trees represent proposals from bottom-up detection, bottom-up prediction, and top-down prediction respectively.

to re-allocate nightstands beside the bed by drawing samples from the geometric distribution. Fig. 9 shows some typical samples from top-down prediction.

$$Q_\gamma(pt \to pt^*) = p(v \in N^T)p(G(v)) \tag{22}$$

Similar to the Eq. 21, the algorithm samples the geometric attributes of the node $G(v)$ and estimate the geometric transformation accordingly, thus propose the new geometry of an object.

Here, we can see that geometric diffusions $Q_\alpha, Q_\beta, Q_\gamma$ proposes pt^* from three major channels. The three bottom-up top-down channels are studied by Wu and Zhu (2011). The geometric parsing is the main challenge in this work, the space of geometric parameters are huge. So most of the MCMC steps are deal with geometric moves. As shown in Fig. 13, the $Q_\alpha, Q_\beta, Q_\gamma$ are three kinds of approximation of the marginal distribution $p(v|pt)$ for a node v. The plot on the right panel of Fig. 13 shows the average energy convergence of hundreds of Markov Chains in the test dataset using different proposal strategies: By only using the α diffusion from bottom-up detection (red curve), the Markov chain converges very fast at the beginning, but cannot keep reducing the energy due to limitation of bottom-up detections. Using the β diffusion from bottom-up prediction (blue curve) is the worst strategy, because if the terminal node can not be optimized, the prediction from bottom-up can be very bad. The black curve which combine three diffusions together is the best strategy, it has sufficient exploration at the beginning, and gradually converges to the lowest energy. Besides that, the combination of $\alpha\&\beta$ (magenta curve) or $\alpha\&\gamma$ (yellow curve) achieve good results which very close to the black one.

5 Experiments

We evaluate our algorithm on two public datasets: the UIUC indoor dataset by Hedau et al (2009) and the UCB dataset by Del Pero et al (2011). 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

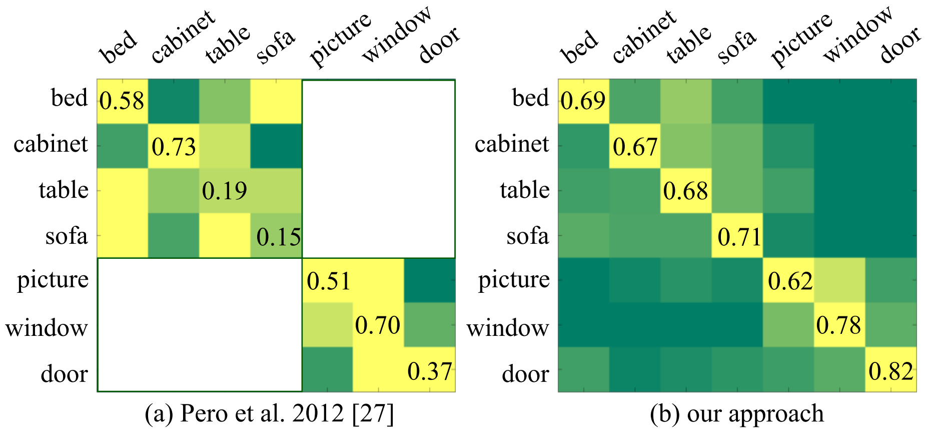


Fig. 15 The confusion matrix of functional object classification on the UCB dataset.

of the background layout with/without foreground objects.

The functional part of our model is trained with the "bedroom" category (2119 images) and the "living room" category (2385 images) of SUN dataset by Xiao et al (2010). In particular, the branching probability of the number variable for each class is calculated by frequency of each production. The geometric part of our model is trained with CAD data in Fig. 7 collected from Trimble 3D Warehouse as discussed in Sect. 3.2. We estimated the mixture of Gaussian model by EM clustering, and we manually picked a few typical samples as the initial mean for the Gaussian. And the appearance part of our model is trained on the UIUC dataset as Hedau et al (2009). The weighting parameters of these three components are tuning by cross validation on the training set of UIUC dataset.

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. 15. The state-of-the-art work by Del Pero et al (2012) 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 et al (2012). The result shows our top-down process did not help the detection of planner objects. But it largely improves the accuracy of cubic object detection from 30.8% to 34.8% with the recall from 24.3% to 29.7%.

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

UCB dataset	planar objects	cubic objects
Del Pero et al (2012)	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	UIUC
Hedau et al (2009)	-	78.8%
Wang et al (2010)	-	79.9%
Lee et al (2010)	-	83.8%
Schwing and Urtasun (2012)	-	83.54%
Del Pero et al (2011)	76.0%	73.2%
Del Pero et al (2012)	81.6%	83.7%
Our approach	82.8%	85.5%

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 et al (2009), Wang et al (2010), Lee et al (2010), Del Pero et al (2011) and Del Pero et al (2012). 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. 17. 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

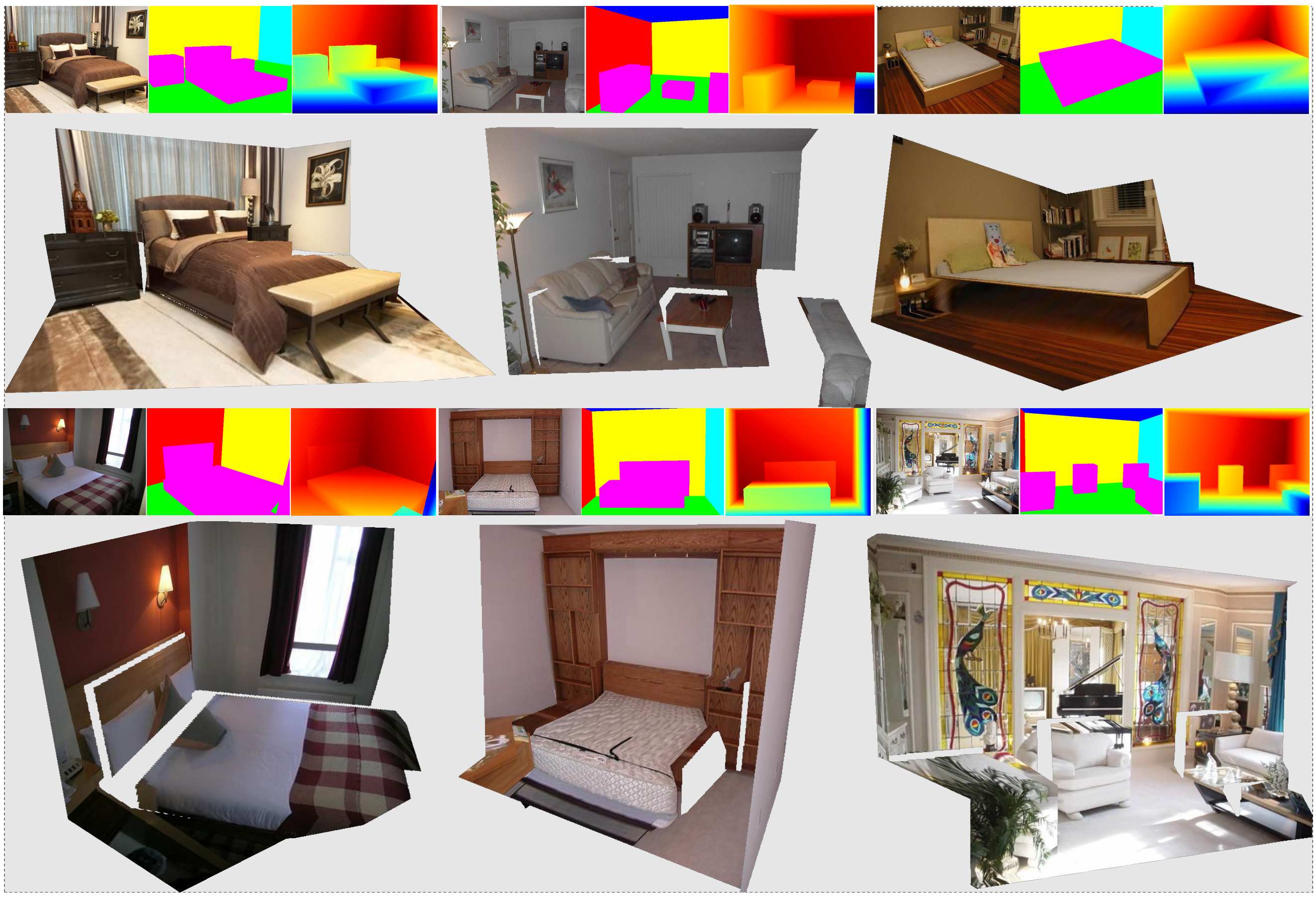


Fig. 16 3D reconstruction results based on 3d image parsing. For each image, we show an original image, a segmentation map, a recovered depth image, and a reconstruction result respectively.

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.

As shown in Fig. 13, the algorithm usually converges after three thousand accepted moves. The computational cost of parsing an image in the dataset is around 5-10 minutes. The computational cost varies in terms of the geometric complexity of the image. Usually, the algorithm takes more time to converge if there are more line segments detected.

6 Conclusion

This paper presents a stochastic scene grammar in a function-geometry-appearance (FGA) hierarchy. Our approach parses an indoor image by inferring the object function and the 3D geometry from 2D appearance. The functionality defines an indoor object by evaluating its “affordance”. The affordance measures how likely an object can support the corresponding human actions. We found it is effective to recognize certain object functions according to its 3D geometry without observing the actions.

Functionality helps to build a bridge between man-made objects and the human actions, which can motivate other interesting studies in the future: functional objects/areas in a scene attract human’s needs and/or intentions; reasoning scene physics and stability in a way similar to Zheng et al (2013, 2015) but from 2D single image instead of RGBD data. 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. Furthermore, given an observed action sequence in video, one can recog-

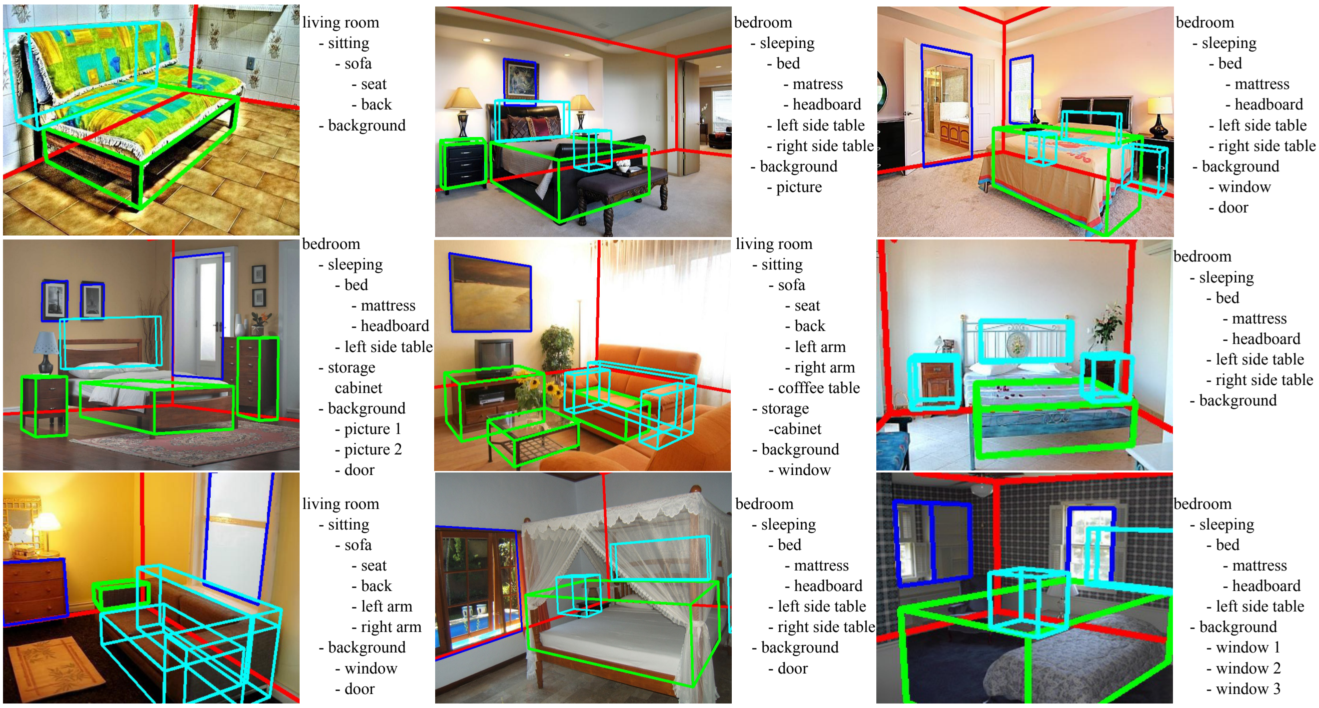


Fig. 17 Parsing results include cubic objects (green cuboids are detected by bottom-up step, and cyan cuboids are detected by top-down prediction), planar objects (blue rectangles), background layout (red box). The parse tree is shown to the right of each image.

nize the functional objects associated with the rational actions detected from motion.

Acknowledgements This work is supported by ONR MURI grant N00014-10-1-0933 and DARPA MSEE grant FA 8650-11-1-7149. We'd like to thank Siyuan Qi for his assistance in running some of the experiments.

References

Aubry M, Maturana D, Efros A, Russell B, Sivic J (2014) Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In: CVPR

Bar-aviv E, Rivlin E (2006) Functional 3d object classification using simulation of embodied agent. In: BMVC

Battaglia P, Hamrick J, Tenenbaum J (2013) Simulation as an engine of physical scene understanding. PNAS 110(45):18,327–18,332

Choi MJ, Lim JJ, Torralba A, Willsky AS (2010) Exploiting hierarchical context on a large database of object categories. In: CVPR

Choi W, Chao Y, Pantofaru C, Savarese S (2013) Understanding indoor scenes using 3d geometric phrases. In: CVPR

Criminisi A, Reid I, Zisserman A (2000) Single view metrology. International Journal of Computer Vision (IJCV) 40(2):123–148

Del Pero L, Guan J, Brau E, Schlecht J, Barnard K (2011) Sampling bedrooms. In: CVPR

Del Pero L, Bowdish J, Fried D, Kermgard B, Hartley E, Barnard K (2012) Bayesian geometric modeling of indoor scenes. In: CVPR, pp 2719–2726

Del Pero L, Bowdish J, Kermgard B, Hartley E, Barnard K (2013) Understanding bayesian rooms using composite 3d object models. In: CVPR

Delage E, Lee H, Ng A (2007) Automatic single-image 3d reconstructions of indoor manhattan world scenes. Robotics Research p 305321

Delaitre V, Fouhey D, Laptev I, Sivic J, Gupta A, Efros A (2012) Scene semantics from long-term observation of people. In: ECCV

Desai C, Ramanan D (2013) Predicting functional regions of objects. In: CVPR Workshop on Scene Analysis Beyond Semantics

Felzenszwalb PF, Huttenlocher DP (2003) Pictorial structures for object recognition. IJCV 61:2005

Felzenszwalb PF, Girshick RB, Mcallester D (2010) D.m.: Cascade object detection with deformable part models. In: CVPR

Fidler S, Dickinson S, Urtasun R (2012) 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In: NIPS

Fouhey DF, Delaitre V, Gupta A, Efros AA, Laptev I, Sivic J (2012) People watching: Human actions as a cue for single-view geometry. In: ECCV

Fouhey DF, Gupta A, Hebert M (2014) Unfolding an indoor origami world. In: ECCV

Fu KS (1982) Syntactic pattern recognition and applications. Prentice-Hall

Gibson JJ (1977) The Theory of Affordances. Lawrence Erlbaum

Goodman ND, Tenenbaum JB (2014) (electronic) Probabilistic Models of Cognition. http://probmods.org

Grabner H, Gall J, Gool LV (2011) What makes a chair a chair? In: CVPR

Guo C, Zhu S, Wu Y (2007) Primal sketch: Integrating texture and structure. Computer Vision and Image Understanding 106(1):5–19

Guo R, Hoiem D (2013) Support surface prediction in indoor scenes. In: ICCV

Gupta A, Efros AA, Hebert M (2010) Blocks world revisited: Image understanding using qualitative geometry and mechanics. In: European Conference on Computer Vision(ECCV)

Gupta A, Satkin S, Efros AA, Hebert M (2011) From 3d scene geometry to human workspace. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Washington, DC, USA, pp 1961–1968

Han F, Zhu SC (2004) Bayesian reconstruction of 3d shapes and scenes from a single image. In: Proc. IEEE Workshop on Perceptual Organization in Computer Vision

Han F, Zhu SC (2009) Bottom-up/top-down image parsing with attribute grammar. PAMI

Hartley RI, Zisserman A (2004) Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, ISBN: 0521540518

Hedau V, Hoiem D, Forsyth D (2009) Recovering the spatial layout of cluttered rooms. In: ICCV

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

Hedau V, Hoiem D, Forsyth D (2012) Recovering free space of indoor scenes from a single image. In: CVPR

Hejrati M, Ramanan D (2012) Analyzing 3d objects in cluttered images. In: NIPS, pp 602–610

Hoiem D, Efros A, Hebert M (2009) Automatic photo pop-up. TOG 31(1):59–73

Hu W (2012) Learning 3d object templates by hierarchical quantization of geometry and appearance spaces. In: CVPR, pp 2336–2343

Isola P, Liu C (2013) Scene collaging: analysis and synthesis of natural images with semantic layers. In: IEEE International Conference on Computer Vision (ICCV)

Jiang Y, Koppula H, Saxena A (2013) Hallucinated humans as the hidden context for labeling 3d scenes. In: CVPR

Kim VG, Chaudhuri S, Guibas L, Funkhouser T (2014) Shape2pose: Human-centric shape analysis. In: SIGGRAPH

Koppula H, Saxena A (2013) Anticipating human activities using object affordances for reactive robotic response. In: RSS

Koppula H, Saxena A (2014) Physically-grounded spatio-temporal object affordances. In: ECCV

Lafferty JD, McCallum A, Pereira FCN (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. ICML pp 282–289

Lazebnik S, Schmid C, Ponce J (2006) Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Lee D, Hebert M, Kanade T (2009) Geometric reasoning for single image structure recovery. In: CVPR

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

Lim J, Khosla A, Torralba A (2014) Fpm: Fine pose parts-based model with 3d cad models. In: ECCV

Lim JJ, Pirsiavash H, Torralba A (2013) Parsing ikea objects: Fine pose estimation. In: IEEE International Conference on Computer Vision (ICCV)

Lin D, Fidler S, Urtasun R (2013) Holistic scene understanding for 3d object detection with rgbd cameras. In: ICCV

Liu C, Yuen J, Torralba A (2011) Nonparametric scene parsing via label transfer. IEEE Trans on Patt Anal Mach Intell (TPAMI)

Liu T, Chaudhuri S, Kim VG, Huang QX, Mitra NJ, Funkhouser T (2014) Creating consistent scene graphs using a probabilistic grammar. In: SIGGRAPH

Mansinghka V, Kulkarni T, Perov Y, Tenenbaum J (2013) Approximate bayesian image interpretation using generative probabilistic graphics programs. In: NIPS

Mobahi H, Zhou Z, Yang AY, Ma Y (2011) Holistic 3d reconstruction of urban structures from low-rank textures. In: Proceedings of the International Conference on Computer Vision - 3D Representation and Recognition Workshop, pp 593–600

Oakes L, Madole K (2008) Function revisited: How infants construe functional features in their representation of objects. Advances in Child Development and Behavior 36:135185

Oliva A, Torralba A (2001) Modeling the shape of the scene: a holistic representation of the spatial envelope. International Journal of Computer Vision (IJCV)

Parizi SN, Oberlin J, Felzenszwalb P (2012) Reconfigurable models for scene recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Payet N, Todorovic S (2011) Scene shape from textures of objects. In: CVPR

Pepik B, Gehler P, Stark M, Schiele B (2012) 3d2pm - 3d deformable part models. In: ECCV, Firenze, Italy

Porway J, Zhu SC (2010) Hierarchical and contextual model for aerial image understanding. IJCV 88(2):254–283

Satkin S, Hebert M (2013) 3dnn: Viewpoint invariant 3d geometry matching for scene understanding. In: ICCV

Satkin S, Lin J, Hebert M (2012) Data-driven scene understanding from 3d models. In: BMVC

Saxena A, Sun M, Ng A (2009) Make3d: Learning 3d scene structure from a single still image. PAMI 31(5):824–840

Schwing AG, Urtasun R (2012) Efficient Exact Inference for 3D Indoor Scene Understanding. In: ECCV

Schwing AG, Hazan T, Pollefeys M, Urtasun R (2012) Efficient structured prediction for 3d indoor scene understanding. In: CVPR

Schwing AG, Fidler S, Pollefeys M, Urtasun R (2013) Box in the box: Joint 3d layout and object reasoning from single images. In: ICCV

Song S, Xiao J (2014) Sliding shapes for 3d object detection in depth images. In: ECCV

Stark L, Bowyer K (1991) Achieving generalized object recognition through reasoning about association of function to structure. PAMI 13:10971104

Tenenbaum J, Kemp C, Griffiths T, Goodman N (2011) How to grow a mind: Statistics, structure, and abstraction. Science 331(6022):1279–1285

Tighe J, Lazebnik S (2013a) Finding things: image parsing with regions and per-exemplar detectors. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Tighe J, Lazebnik S (2013b) Superparsing: scalable nonparametric image parsing with superpixels. International Journal of Computer Vision (IJCV)

Tu Z, Zhu SC (2002) Image segmentation by data-driven markov chain monte carlo. PAMI 24(5):657–673

Tu Z, Chen X, Yuille A, Zhu S (2005) Image parsing: unifying segmentation, detection and recognition. IJCV 63(2):113–

140

Von Gioi R, Jakubowicz J, Morel JM, Randall G (2010) Lsd: A fast line segment detector with a false detection control. TPAMI 32(4):722–732

Wang H, Gould S, Koller D (2010) Discriminative learning with latent variables for cluttered indoor scene understanding. In: ECCV, pp 497–510

Wang S, Wang Y, Zhu S (2012) Hierarchical space tiling in scene modeling. In: Asian Conf. on Computer Vision (ACCV)

Wei P, Zhao Y, Zheng N, Zhu SC (2013) Modeling 4d human-object interactions for event and object recognition. In: ICCV

Wu T, Zhu SC (2011) A numerical study of the bottom-up and top-down inference processes in and-or graphs. IJCV 93(2):226–252

Xiang Y, Savarese S (2012) Estimating the aspect layout of object categories. In: CVPR

Xiao J, Hays J, Ehinger K, Oliva A, Torralba A (2010) Sun database: Large-scale scene recognition from abbey to zoo. In: CVPR, pp 3485 –3492

Xiao J, Russell B, Torralba A (2012) Localizing 3d cuboids in single-view images. In: NIPS, pp 755–763

Yao B, Ma J, Fei-Fei L (2013) Discovering object functionality. In: ICCV

Yeh YT, Yang L, Watson M, Goodman ND, Hanrahan P (2012) Synthesizing open worlds with constraints using locally annealed reversible jump mcmc. ACM Trans Graph 31(4):56:1–56:11

Zhang Y, Song S, Tan P, Xiao J (2014) Panocontext: A whole-room 3d context model for panoramic scene understanding. In: ECCV

Zhao Y, Zhu SC (2011) Image parsing via stochastic scene grammar. In: NIPS

Zhao Y, Zhu SC (2013) Scene parsing by integrating function, geometry and appearance models. In: CVPR

Zheng B, Zhao Y, Yu JC, Ikeuchi K, Zhu SC (2013) Beyond point clouds: Scene understanding by reasoning geometry and physics. In: CVPR

Zheng B, Zhao Y, Yu JC, Ikeuchi K, Zhu SC (2015) Scene understanding by reasoning stability and safety. IJCV DOI 10.1007/s11263-014-0795-4

Zhu SC, Mumford D (2007) A stochastic grammar of images. Foundations and Trends in Computer Graphics and Vision 2(4):259–362