Modeling 3D Environments through Hidden Human Context

Yun Jiang, *Student Member, IEEE*, Hema S. Koppula, and Ashutosh Saxena, *Member, IEEE*

Abstract—The idea of modeling object-object relations has been widely leveraged in many scene understanding applications. However, as the objects are designed by humans and *for* human usage, when we reason about a human environment, we reason about it through an interplay between the environment, objects and humans. In this paper, we model environments not only through objects, but also through latent human poses and human-object interactions. In order to handle the large number of latent human poses and a large variety of their interactions with objects, we present Infinite Latent Conditional Random Field (ILCRF) that models a scene as a mixture of CRFs generated from Dirichlet processes. In each CRF, we model objects and object-object relations as existing nodes and edges, and hidden human poses and human-object relations as latent nodes and edges. ILCRF generatively models the distribution of different CRF structures over these latent nodes and edges. We apply the model to the challenging applications of 3D scene labeling and robotic scene arrangement. In extensive experiments, we show that our model significantly outperforms the state-of-the-art results in both applications. We further use our algorithm on a robot for arranging objects in a new scene using the two applications aforementioned.

Index Terms—3D scene understanding, human context, machine learning, robotics perception

1 Introduction

We *make the world we live in and shape our own environment*. Orison Swett Marden (1894).

Our environment, though physically consisting of a collection of objects, is built *for* human usage. Therefore, in order to understand a human environment, we need to consider the interplay between the environment, objects and humans. For example, consider a typical office scene in Fig. 1, with a chair, table, monitor and keyboard. This scene can be described through many object-object relations, such as chair-in-front-of-table, monitor-on-table, keyboard-on-table, and so on. This particular configuration can also be naturally explained by a sitting human pose in the chair and working with the computer.

We argue that a human environment is constructed under two types of relations: *object-object* and *human-object* relations. When only considering object-object relations, Conditional random fields (CRFs) are a natural choice, as each object can be modeled as a node in a Markov network and the edges in the graph can reflect the object-object relations. Modeling possible human poses and human-object interactions (or *object affordances*) is not trivial because of several reasons. First, humans are not always observable, but we still want to model them as latent factors for making the scene as it is. Second, there can be any number of possible humans in a scene—e.g., some sitting on the couch/chair, some standing by the shelf/table; Third, there can be various types of human-object interactions in a scene, such as watching TV in distance, eating from dishes, or working on a laptop, etc. Fourth, an object can be used by different human poses, such as a book on the table can be accessed by either a sitting pose on the couch or a standing pose nearby. Last, there can be multiple possible usage scenarios in a scene. For example, a living room can be used for watching TV as well as for working (see Fig. 2 (middle row)). Therefore, we need models that can incorporate latent factors, latent structures, as well as different alternative possibilities.

The idea of object affordances has re-gained attention in the computer vision community recently. For example, being able to afford certain human poses can be used to detect chairs [5] and infer human workspaces [6]. Using observed human-object interactions, one can also predict the 3D scene geometry [7] or distinguish different sports [8] in an image. While inspired by the similar idea of object affordances, our work models object affordances with a more generic definition and considers the aforementioned challenges in order to apply it to scene labeling and scene arrangement problems.

In this work, we propose infinite latent conditional random fields (ILCRFs) for modeling the aforementioned properties. Intuitively, it is a mixture of CRFs where each CRF can have two types of nodes: existing nodes (e.g., object nodes, which are given in the graph and we only have to infer the value) and latent nodes (e.g., human nodes, where an unknown number of humans may be hallucinated in the room). The relations between the nodes (e.g., object-object edges and human-object edges) could also be of different types. Unlike traditional CRFs, where the structure of the graph is given, the structure of our ILCRF is sampled from Dirichlet Processes (DPs) [9]. DPs are widely used as non-parametric Bayesian priors for mixture models, the resulting DP mixture models can determine the number of components from data, and therefore is also referred as infinite mixture models. ILCRFs are inspired by this, and we

• *The authors are with the Computer Science Department, Cornell University, Ithaca, NY 14853. E-mail: {yunjiang, hema, asaxena}@cs.cornell.edu.*

Manuscript received 23 Mar. 2014; revised 28 Mar. 2015; accepted 10 Sept. 2015. Date of publication 2 Dec. 2015; date of current version 12 Sept. 2016. Recommended for acceptance by G. Mori.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2015.2501811

Fig. 1. Left: Previous approaches model the relations between observable entities, such as the objects. Right: In our work, we consider the relations between the objects and hidden humans. Our key hypothesis is that even when the humans are never observed, the human context is helpful.

call it 'infinite' as it can sidestep the difficulty of finding the correct number of latent nodes as well as latent edge types. Our learning and inference methods are based on Gibbs sampling that samples latent nodes, existing nodes, and edges from their posterior distributions.

We instantiate two specific ILCRFs for two applications: scene labeling where the objective is to identify objects in a scene, and scene arrangement where the objective is to find proper placements (including 3D locations and orientations) of given objects in a scene. Despite the disparity of the tasks at the first look, we relate them through one common hidden cause—imaginary humans and object affordances. For both tasks, our ILCRF models each object placement as an existing node, hallucinated human poses as latent nodes and spatial relationships among objects or between objects and humans as edges. We demonstrate in the experiments that this unified model achieves the state-of-the-art results on both synthetic and real datasets. More importantly, we perform an exhaustive analysis on how our model captures different aspects of human context in scenes, in comparisons with numerous baselines. We further demonstrate that by using the two applications together, a robot successfully identified the class of objects in a new room, and placed several objects correctly in it.

In summary, the contributions of this paper are:

- We design a generic definition of object affordances, representing human-object spatial interactions.
- We propose ILCRFs to capture both human-object and object-object relations in a scene where humans are hidden.
- Compared to classic CRFs, our ILCRFs admit: 1) unknown number of latent variables, 2) unknown number of potential functions, and 3) a mixture of different CRFs. Its flexibility allows us to have minimum restrictions on humans and affordances.
- We apply the same setup for hallucinated humans and object affordances to two distinct applications—scene labeling and scene arrangement—demonstrating the generality of our model.

The rest of the paper is organized as the following: Section 2 defines a general scene understanding problem with a discussion of previous popular approaches. Section 3 introduces human context that we aim to capture. Section 4 presents our ILCRFs and the sampling-based learning and inference algorithm and describes how it is applied to two different tasks—scene labeling and scene arrangement. Section 5 reviews related work of both our model and applications. Section 6 details experimental results and analysis, followed by conclusions in Section 7.

2 Problem Formulation

Given a scene, we are interested in objects that are (or could be) in it, such as identifying the object class in the scene labeling task. The scene is represented as an RGB-D point cloud. We first segment the point cloud based on smoothness and continuity of surfaces using the approach in [10]. We use $\mathcal{X} = \{x_1, \ldots, x_N\}$ to denote N segments of interest,

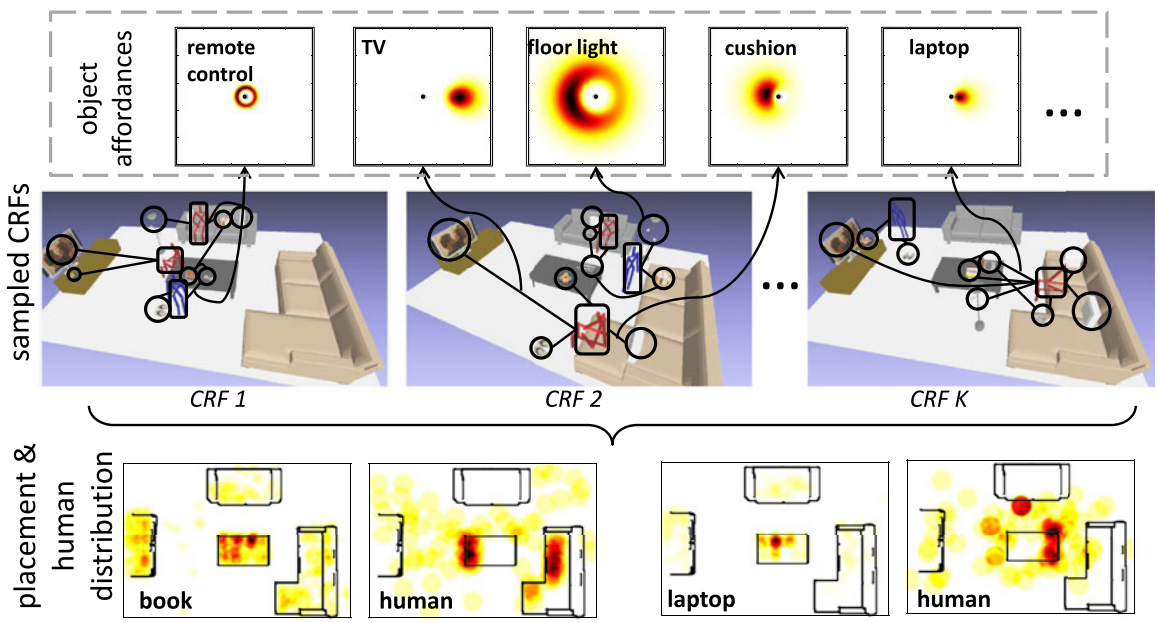


Fig. 2. An example of instantiated ILCRF for scene arrangement. Top row shows learned object affordances in top-view heatmaps (it shows the probability of the object's location, given a human pose in the center facing to the *right*). Middle row shows a total of K CRFs sampled from our ILCRF algorithm—each CRF models the scene differently. Bottom row shows the distribution of the objects and humans (in the top view of the room) computed from the sampled CRFs.

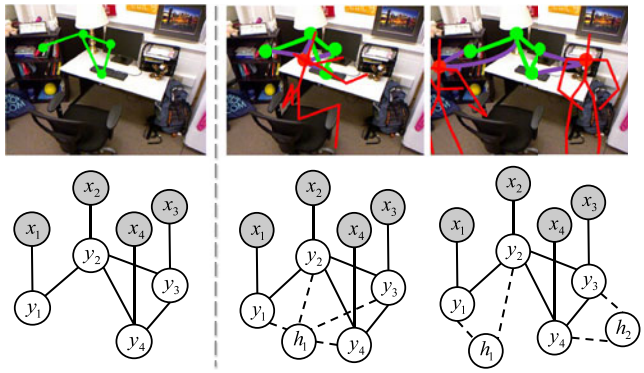


Fig. 3. **Graphical models for scene modeling.** (a) Conditional random field has been used to capture objects and their relations in a scene. (b) In our work, we model possible human configurations as latent nodes in a scene in order to capture human and object relations in a scene.

and $\mathcal{Y} = \{y_1, \ldots, y_N\}$ to denote the corresponding labels. In the task of scene labeling, for instance, x_i is the observed appearance features and locations of the i^{th} segment/object in the scene, and y_i is an integer between 1 and M representing the class label, such as monitor, chair, floor, etc. We model the correspondance between $\mathcal{X}$ and $\mathcal{Y}$ through probabilistic distribution $P(\mathcal{Y}|\mathcal{X})$, and the objective of the labeling task is to find the optimal labels given observations, namely,

$$\mathcal{Y}^* = \arg\max_{\mathcal{Y}} P(\mathcal{Y}|\mathcal{X}).$$

A simple and naive solution is to treat objects independently: $y_i^* = \arg\max_y P(y|x_i)$. In this way, we can label the object class using its own shape/appearance features such as HOG [11]. However, these methods would suffer from noisy local features and the lack of *context* of the whole scene.

2.1 CRFs for Object Context

There are many works trying to capture the context from object-object relations, which can be naturally modeled through conditional random fields [12], [13], [14], [15], [16], [17]. A CRF is a network where each node can be modeled as an object and each edge reflects the relationship between the linked two objects. An example is shown in Fig. 3a.

Definition 1. *$CRF(\mathcal{X}, \mathcal{Y}, E_Y)$ is a conditional random field if that, when conditioned on $\mathcal{X}$, random variables $\mathcal{Y}$ follow the Markov property with respect to the graph E_Y: The absence of an edge between nodes y_i and y_j implies that they are independent given other nodes.*

Thus, the likelihood of $\mathcal{Y}$ given $\mathcal{X}$ is given by: $P(\mathcal{Y}|\mathcal{X}) \propto \prod_{c\in\mathcal{C}} \psi_c(X_c, Y_c)$, where $\mathcal{C}$ is all the maximum cliques, and $Y_c \in \mathcal{Y}$ and $X_c \in \mathcal{X}$ are in the same clique c. ψ is the potential functions. Following [10], [13], we use log-linear node (ψ^o) and edge potentials (ψ^{oo}) to capture object-object context:

$$\begin{aligned} P(\mathcal{Y}|\mathcal{X}) &\propto \prod_{i=1}^{N} \psi^o(x_i, y_i) \prod_{(y_i,y_j)\in E_Y} \psi^{oo}(x_i, x_j, y_i, y_j) \\ &= \exp \sum_i \sum_{k=1}^{M} \mathbf{1}_{\{y_i=k\}} \left(\theta_k^o\right)^\top \phi^o(x_i) \\ &\quad \times \exp \sum_{ij} \sum_{kl} \mathbf{1}_{\{y_i=k\}} \mathbf{1}_{\{y_j=l\}} \left(\theta_{kl}^{oo}\right)^\top \phi^{oo}(x_i, x_j), \end{aligned} \tag{1}$$

Fig. 4. Six types of human poses extracted from Kinect 3D data.

where ϕ^o and ϕ^{oo} are object's own and pairwise features,[1] and θ_k^o and θ_{kl}^{oo} are parameters to learn for each class k and each pair of classes (k, l).

3 Human Context

In this paper, we additionally want to model human context (human-object relations) in a scene. The human context is very important for understanding our environment. In fact, even when no human is present in an indoor scene, the potential human-object interactions give such a strong cue for scene understanding that we want to model it as latent variables in our algorithms. Moreover, modeling human-object relations is parsimonious and efficient as compared to modeling the pairwise object-object relationships: For n objects, we only need to model how they are used by humans, i.e., $O(n)$ relations, as compared with modeling $O(n^2)$ if we were to model object to object context naively.

In the following, we first define the representation of human configurations and human-object relations (referred as 'object affordances' in the rest of the paper). Then we show how to incorporate the human context into the CRF we just described.

3.1 Human Configuration

A human configuration, denoted by h, is comprised of a pose type, location and orientation. The pose type, as shown in Fig. 4, is specified by relative positions of 15 body joints, such as head, torso, shoulders, hips, etc. In this work, we consider six static poses extracted from real human activity dataset: We collected all poses in Cornell Activity Dataset-60 [18], and clustered them using k-means algorithm giving us six types of skeletons. Each pose could be at any X-Y-Z location and in different orientations $\in [0, 2\pi)$ inside the scene.

3.2 Object Affordances

A human can use the objects at different distances and orientations from the human body. For example, small handheld devices are typically held close to and in front of the human. Objects such as a TV and decoration pieces are typically placed at a distance. The human-object spatial relations can be a strong hint of the object class as well as where to place the object. Therefore, we define the object affordance as the probability distribution of the object's relative 3D location with respect to a human pose h. An example of the laptop's affordance is shown in Fig. 5: Given a centered sitting human pose h, the distribution of a laptop is

1. ϕ^o includes local features such as histograms of HSV colors, normal and dimensions of the segment's bounding box. ϕ^{oo} includes features such as difference of HSV colors, displacement or co-planarity of the two segments. More details are in [13].

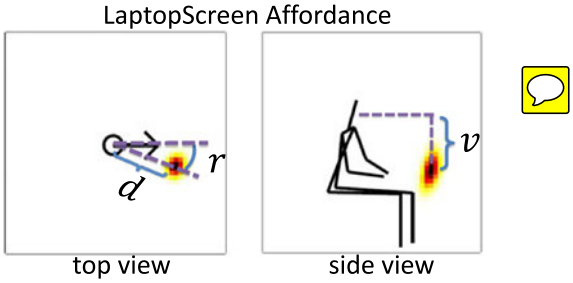


Fig. 5. The affordance of a laptop screen, visualized in top- and side-view.

projected onto a top-view and side-view heatmaps, indicating that the laptop is most likely to appear right ahead of human hands.

In detail, we define an object affordance as a product of terms capturing the preferred distance and orientation from the object to the human pose:

$$\psi^{ho}(x,y,h;\Theta) = \psi_{\text{dist}}\psi_{\text{rel}}\psi_{\text{ori}}\psi_{\text{vert}}. \quad (2)$$

We now describe each term below:

Distance preference. Some objects are preferred to be at a certain distance from humans, such as a TV or a laptop. This preference, encoded as ψ_{dist}, includes how far the object should be and how strong this bias is. Let $d(x,y,h)$ be the euclidean distance (in 3D space) between the human and object, as shown in Fig. 5. The distance follows a log-normal distribution:

$$\psi_{\text{dist}}(x,y,h;\mu_d,\sigma_d) = \frac{\exp(-\frac{(\ln d(x,y,h)-\mu_d)^2}{2\sigma_d^2})}{d(x,y,h)\sigma_d\sqrt{2\pi}}. \quad (3)$$

Relative angular preference. There is a preference for objects to be located at a certain angle with respect to human poses. For example, people will sit in front of a laptop, but prefer the mouse to be on their right (or left). Let $r(x,y,h)$ be relative angle from the object to human (as shown in Fig. 5(left)), and we assume it follows a von Mises distribution:

$$\psi_{\text{rel}}(x,y,h;\mu_r,\kappa_r) = \frac{\exp(\kappa\cos(r(x,y,h)-\mu_r)}{2\pi I_0(\kappa_r)}. \quad (4)$$

Orientation preference. There is a preference for objects to be oriented at a certain angle with respect to the human pose (e.g., a monitor should also be facing towards the skeleton when located in front of the skeleton). We use $o(x,y,h)$ to denote the difference between orientations of the object and human (regardless their relative angle), i.e., $o(x,y,h) = |o(x,y) - o(h)|$. Similarly to ψ_{rel}, we also use a von Mises distribution for this term:

$$\psi_{\text{ori}}(x,y,h;\mu_o,\kappa_o) = \frac{\exp(\kappa\cos(o(x,y,h)-\mu_o)}{2\pi I_0(\kappa_o)}. \quad (5)$$

Vertical difference preference. ψ_{vert} is a Gaussian distribution of the object's relative height to a human pose. Let $v(x,y,h)$ be the vertical distance between the human and object, as shown in Fig. 5(right). We it follows a normal distribution:

$$\psi_{\text{vert}}(x,y,h;\mu_v,\sigma_v) = \frac{\exp(-\frac{(v(x,y,h)-\mu_v)^2}{2\sigma_v^2})}{v(x,y,h)\sigma_v\sqrt{2\pi}} \quad (6)$$

In this way, we specify one object affordance ψ^{ho} using one set of parameters $\Theta = \{\mu_d,\sigma_d,\mu_r,\kappa_r,\mu_o,\kappa_o,\mu_v,\sigma_v\}$.

3.3 Modeling Observed Human Context in CRFs

Let us first consider how to model human context when both human configurations and object affordances are given.

Suppose K human configurations are given in a scene, each of which is specified by a pose type, location and orientation, such as the sitting pose in Fig. 3(b). We model each human pose as a node in the graph, $\mathcal{H} = \{h_1,\ldots,h_K\}$ and each human-object relationship as an edge (y_i, h_{z_i}) where $z_i \in \{1,\ldots,K\}$ denotes which human pose is using the ith object.[2] For example, in the second case in Fig. 3(b), $z_1 = z_2 = 1$ and $z_3 = z_4 = 2$. We use $\mathcal{Z} = \{z_1,\ldots,z_N\}$ to denote these human configuration correspondances.

Suppose we are also given M different object affordances, $\Psi = \{\psi_1^{ho},\ldots,\psi_M^{ho}\}$ where each ψ_k^{ho} is defined in (2) with parameter Θ_k. For each object i, we use $\omega_i \in \{1,\ldots,M\}$ to denote its correspondent affordance (and $\Omega = \{\omega_1,\ldots,\omega_N\}$ for all objects). In other words, we associate the edge (y_i, h_{z_i}) with the potential $\psi^{ho}_{\omega_i}(x_i,y_i,h_{z_i})$.

Given such a CRF with known human context (specified by $\mathcal{G} = \{\mathcal{H},\mathcal{Z},\Psi,\Omega\}$, the likelihood now is,

$$P(\mathcal{Y}|\mathcal{X},\mathcal{G}) \propto \prod_{i=1}^{N}\psi^{o}(x_i,y_i)\prod_{(y_i,y_j)}\psi^{oo}(x_i,x_j,y_i,y_j)\prod_{i=1}^{N}\psi^{ho}_{\omega_i}(x_i,y_i,h_{z_i}), \quad (7)$$

where the first two terms are defined in Eq. (1) and the last one is in Eq. (2).

How to model hidden human context? More often humans are not present in the scene, nevertheless, the potential human-object relations are valuable information for scene understanding. Such latent nature of human context, combined with the enormous space of possible human configurations and object affordances, can lead to an ill-posed problem. For example, one potential explanation of the scene could be humans floating in the air and prefer stepping on every object as the affordance! The key to modeling the large space of latent human context lies in building *parsimonious* models (where objects are encouraged to share the same human configuration and the same affordance) and providing *physics-based priors* so that only physically plausible human configurations are considered[3] and through which reaonable object affordances can be learned. Please refer to [3] for more details.

4 Infinite Latent CRFs

In this paper, we propose a type of mixture CRFs—infinite latent conditional random fields, which can capture the following properties:

2. We assume a human pose can interact with multiple objects at the same time but each object is used by only one human pose.

3. When generating human configurations, we perform 1) Collision check to ensure configurations are kinematically feasible, and 2) Dynamic check to ensure the human skeleton is stably supported by the nearby environment.

1. Unknown number of latent nodes. This is essential for applications of finding hidden causes, such as scene modeling where the number of possible human poses in a scene is unknown and changes across different scenes.

*2. Unknown number of the **types** of potential functions.* Potential function measures the relationship between nodes, and therefore, having variety in them can help us model complex relations. For example, in the task of image segmentation, different types of context can be modeled as different edges in a CRF [19]. In this paper, we use them to capture different object affordances.

3. Mixture CRFs. The complexity of real-world data may not always be explained by a single CRF. Therefore having a mixture of CRFs, with each one modeling one particular conditional independency in the data, can increase the expressive power of the model.

4. Ability to place informative priors on the structure of CRFs. This can help producing more plausible CRFs as well as reducing the computational complexity.

We achieve this by imposing Bayesian nonparametric priors—Dirichlet processes—to the latent variables, potential functions and graph structures.

4.1 Background: Dirichlet Process Mixture Model

Dirichlet process is a stochastic process to generate distributions that is used to model clustering effects in the data. It has been widely applied to modeling *unknown* number of components in mixture models (such as modeling the unknown number of object parts in part-based object detection models [20]), which are often called *infinite* mixture models. (Formal definition can be found in [9].)

Definition 2. *A DP mixture model, $DP(\alpha, B)$, defines the following generative process (also called the stick-breaking process), with a concentration parameter α and a base distribution B:*

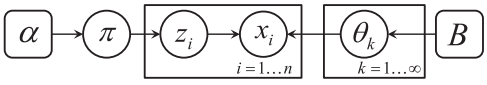
1) *Generate infinite number of mixture components, parameterized by $\Theta = \{\theta_1, \ldots, \theta_\infty\}$, and their mixture weights $\boldsymbol{\pi}$:*

$$\theta_k \sim B,\ b_k \sim Beta(1, \alpha), \quad \pi_k = b_k \textstyle\prod_{i=1}^{k-1}(1 - b_i). \tag{8}$$

2) *Assign the z_i^{th} component to each data point x_i and draw from it:*

$$z_i \sim \boldsymbol{\pi}, \qquad x_i \sim F(\theta_{z_i}). \tag{9}$$

The process can be represented in the following plate notation:



4.2 ILCRF

ILCRF uses DPs to admit an arbitrary number of latent variables and potential functions to obtain a mixture of latent CRFs. In brief, it generates latent variables and potential functions from two DPs respectively, and each data point builds a link, associated with one potential function, to one latent variable. Different samples thus form different CRFs.

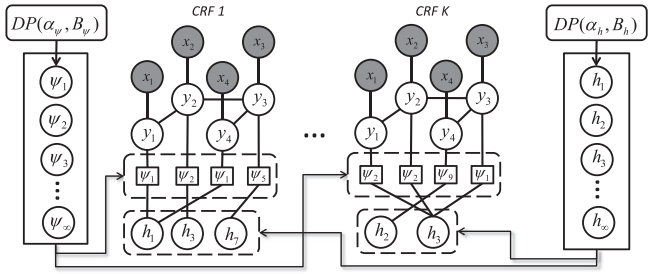


Fig. 6. Graphical representations of our infinite latent CRF (ILCRF).

Definition 3. *An ILCRF$(\mathcal{X}, \mathcal{Y}, E_Y, \alpha_h, B_h, \alpha_\psi, B_\psi)$ is a mixture of CRFs, where the edges in $\mathcal{Y}$ are defined in graph E_Y and latent variables $\mathcal{H}$ as well as the edges between $\mathcal{H}$ and $\mathcal{Y}$ are generated through the following process:*

1) *Generate infinite number of latent nodes $\mathcal{H} = \{h_1, h_2, \ldots, h_\infty\}$ and a distribution $\boldsymbol{\pi}_h$ from a DP process $DP(\alpha_h, B_h)$ following Eq. (8); Assign one edge to each label y_i that links to h_{z_i}, where $z_i \sim \boldsymbol{\pi}_h$ following Eq. (9).*
2) *Generate infinite number of potential functions ('types' of edges) $\Psi = \{\psi_1, \ldots, \psi_\infty\}$ and a distribution $\boldsymbol{\pi}_\psi$ from a DP process $DP(\alpha_\psi, B_\psi)$ following Eq. (8); Assign one potential function ψ_{ω_i} to each edge (y_i, h_{z_i}), where $\omega_i \sim \boldsymbol{\pi}_\psi$ following Eq. (9).*

We now illustrate the process using Fig. 6. Consider first sampled CRF ('CRF-1' in the figure) with four visible nodes y_i $(i = 1 \ldots 4)$. In the first step, y_1 is connected to h_1, y_2 to h_3, y_3 to h_7 and y_4 to h_1 again. This is because z_i's $(i = 1 \ldots 4)$ are sampled as $(1, 3, 7, 1)$ from $DP(\alpha_h, B_h)$. Since only h_1, h_3 and h_7 are active, we draw their values from $DP(\alpha_h, B_h)$. Thus, we get a CRF with three latent nodes $\{h_1, h_3, h_7\}$. In the second step, the potential function of edge (y_1, h_1) is assigned to ψ_1, (y_2, h_3) to ψ_2, (y_3, h_7) to ψ_5 and (y_4, h_1) to ψ_1. This is because ω_i's are sampled as $(1, 2, 5, 1)$ from $DP(\alpha_\psi, B_\psi)$. Since, only (ψ_1, ψ_2, ψ_5) are active, we have three edge types in this CRF. We draw their parameters from $DP(\alpha_\psi, B_\psi)$. Repeating this procedure may generate different latent CRFs such as *'CRF-K'* which has two different latent nodes and three different edge types. In the end, their mixture forms the ILCRF. Note that the structure of labels (edges between y_i's) is defined by E_Y and is shared across all the sampled CRFs.

From the probabilistic perspective, ILCRF defines a distribution over different CRFs with latent variables, where each CRF is specified by $\mathcal{G} = \{\mathcal{H}, \mathcal{Z}, \Psi, \Omega\}$ and its likelihood is governed by prior distributions B_h and B_ψ (their specific forms are given in Section 4.4). Specifically, the first generative step above defines the probability of latent nodes and edges:

$$P(\mathcal{H}, \mathcal{Z} | \alpha_h, B_h) = \int P(\mathcal{H}, \boldsymbol{\pi}_h | \alpha_h, B_h) \prod_{i=1}^{N} \boldsymbol{\pi}_h(z_i) d\boldsymbol{\pi}_h. \tag{10}$$

Similarly, the second step defines the probability of potentials for all edges between $\mathcal{Y}$ and $\mathcal{H}$:

$$P(\Psi, \Omega | \alpha_\psi, B_\psi) = \int P(\Psi, \boldsymbol{\pi}_\psi | \alpha_\psi, B_\psi) \prod_{i=1}^{N} \boldsymbol{\pi}_\psi(\omega_i) d\boldsymbol{\pi}_\psi. \tag{11}$$

Since $\mathcal{G}$ is latent, we marginalize over all its possible values to compute the overall likelihood of an ILCRF:

$$\begin{aligned} P(\mathcal{Y}|\mathcal{X}) &= \int P(\mathcal{Y}, \mathcal{G} = \{\mathcal{H}, \mathcal{Z}, \Psi, \Omega\}|\mathcal{X})\, d\mathcal{G} \\ &= \int \underbrace{P(\mathcal{H}, \mathcal{Z}|\alpha_h, B_h)}_{\text{DP prior for } \mathcal{H}\ (10)} \underbrace{P(\Psi, \Omega|\alpha_\psi, B_\psi)}_{\text{DP prior for } \Psi\ (11)} \\ &\quad \times \underbrace{P(\mathcal{Y}|\mathcal{X}, \mathcal{G} = \{\mathcal{H}, \mathcal{Z}, \Psi, \Omega\})}_{\text{conditional prob. of the CRF }(7)} d\mathcal{G}. \end{aligned} \tag{12}$$

Exact computation of this likelihood is prohibitive in practice. We therefore present learning and inference methods based on Gibbs sampling in the following.

4.3 Gibbs Sampling for Learning and Inference

Gibbs sampling states that, if we sample latent CRFs, including the edge/structure of G, the value of latent nodes $\mathcal{H}$ and the edge types Ψ, from their posterior distributions, then the samples approach the joint distribution $P(\mathcal{Y}, G_\ell, \mathcal{H}, \Psi|\mathcal{X})$. And this can be further used to estimate $P(\mathcal{Y}|\mathcal{X})$ in (12) and to infer the most likely values of $\mathcal{Y}$.

We present the posterior distributions below, modified from the Chinese restaurant process [9], [21] for classic DP mixture models.

- Sample the graph structure, i.e., one edge for each y_i to one latent node: [4]

$$z_i = z \propto \begin{cases} \frac{n^h_{-i,z}}{N+m-1+\alpha_h} \psi_{\omega_i}(x_i, y_i, h_z) & n^h_{-i,z} \geq 0, \\ \frac{\alpha_h/m}{N+m-1+\alpha_h} \psi_{\omega_i}(x_i, y_i, h_z) & \text{otherwise.} \end{cases} \tag{13}$$

- Sample values for each latent node in the graph:

$$h_k = h \propto B_h(h) \times \prod_{i:z_i=k} \psi_{\omega_i}(x_i, y_i, h). \tag{14}$$

- Assign the type of potential functions to each edge: [5]

$$\omega_i = \omega \propto \begin{cases} \frac{n^\psi_{-i,\omega}}{N+m-1+\alpha_\psi} \psi_\omega(x_i, y_i, h_{z_i}) & n^\psi_{-i,\omega} \geq 0, \\ \frac{\alpha_\psi/m}{N+m-1+\alpha_\psi} \psi_\omega(x_i, y_i, h_{z_i}) & \text{otherwise.} \end{cases} \tag{15}$$

- Sample the parameters of each selected potential function:

$$\psi_k = \psi \propto B_\psi(\psi) \times \prod_{i:\omega_i=k} \psi(x_i, y_i, h_{z_i}). \tag{16}$$

- Sample labels:

$$\begin{aligned} y_i = y \propto \psi_{\omega_i}(x_i, y, h_{z_i}) \times \psi^o(x_i, y) \\ \times \prod_{(y_i,y_j)} \psi^{oo}(x_i, x_j, y, y_j). \end{aligned} \tag{17}$$

Note that when we sample the graph structure in Eq. (13) and (15), we assume that the partition function accross the different graph structures is constant. Another commonly used approximation is via pseudo-likelihood [22]: We approximate the true likelihood $P(\mathcal{Y}|\mathcal{X}, \mathcal{G})$ by $\prod_i P(y_i|\mathcal{Y}_{-i}, \mathcal{X}, \mathcal{G})$ where $P(y_i|\mathcal{Y}_{-i}, \mathcal{X}, \mathcal{G}) = \frac{\psi_{\omega_i}(x_i,y_i,h_z)\psi^o(x_i,y_i)\prod_{(y_i,y_j)}\psi^{oo}(x_i,x_j,y_i,y_j)}{\sum_{y_i=y}\psi_{\omega_i}(x_i,y,h_{z_i})\psi^o(x_i,y)\prod_{(y_i,y_j)}\psi^{oo}(x_i,x_j,y,y_j)}$ (Eq. (7)). Now we can sample z_i based on this pseudo-likelihood, i.e., $z_i = z \propto \frac{n^h_{-i,z}}{N+m-1+\alpha_h} P(\mathcal{Y}|\mathcal{X}, \mathcal{G}) \propto \frac{n^h_{-i,z}}{N+m-1+\alpha_h} \prod_i \ P(y_i|\mathcal{Y}_{-i}, \mathcal{X}, \mathcal{G})$. Note that for all other $j \neq i$, $P(y_j|\mathcal{Y}_{-i}, \mathcal{X}, \mathcal{G})$ is constant w.r.t. z_i, and so is $\psi^o(x_i, y_i)$ and $\psi^{oo}(x_i, x_j, y_i, y_j)$. Hence, $z_i = z \propto \frac{n^h_{-i,z}}{N+m-1+\alpha_h} P(y_i|\mathcal{Y}_{-i}, \mathcal{X}, \mathcal{G}) \propto \frac{n^h_{-i,z}}{N+m-1+\alpha_h} \frac{\psi_{\omega_i}(x_i,y_i,h_z)}{\sum_{y_i=y}\psi_{\omega_i}(x_i,y,h_{z_i})\psi^o(x_i,y)\prod_{(y_i,y_j)}\psi^{oo}(x_i,x_j,y,y_j)}$. However in our experiments, we observe little performance gain by doing this and hence ignore the denominator in our implementations.

As for learning the E_Y, when labels are given in the training data, E_Y is independent with latent variables $\mathcal{H}$ (if the partition function is ignored), and therefore can be learned separately. For instance, E_Y used in our labeling task is learned separately using max-margin learning [13].

4.4 Learning Object Affordances

The primary part of learning an ILCRF model is to learn object affordances. As we defined in Section 3.2, the affordance ψ is parameterized by $\Theta = \{\mu_d, \sigma_d, \mu_r, \kappa_r, \mu_o, \kappa_o, \mu_v, \sigma_v\}$ for each object class. Therefore, sampling ψ in Eq. (16) is done through sampling each parameter in Θ. In practice, the posterior sampling of Θ may be difficult when not using conjugate priors. To handle this, we use the maximum a posteriori (MAP) estimate instead of sampling. For example, the parameters for distance preference, μ_d and σ_d in Θ_k is updated by

$$\mu_d, \sigma_d = \arg\max_{\mu,\sigma} B_\psi(\psi) \prod_{i:\omega_i=k} \psi_{\text{dist}}(x_i, y_i, h_{z_i}; \mu, \sigma),$$

and similar for the other six parameters. In this work, we use non-informative prior for B_ψ, which is a zero-mean Gaussian with large variance for each of these four terms. We illustrate the learning process in Fig. 8 shows an example of how the object affordances are refined progressively along with the sampled human poses.

Often, each object type is associated with one object affordance. Thus in our two tasks, we assume they are equivalent, i.e., $\omega_i = y_i$. Since object labels y_i are given during the training, we only perform Gibbs sampling on z_i, h_k and ψ_k, as summarized in Table 1.

However, because of the DP prior on affordances, our ILCRF can actually learn the number of affordances needed from the data, which we refer as 'affordance topics' (for its

4. The posterior distribution of a variable is proportional to its prior and to its likelihood. In the case of z_i, it means that the probability of linking an edge from y_i to h_z is determined by: 1) the likelihood of this edge, given by $\psi_{\omega_i}(x_i, y_i, h_z)$; 2) the number of other subjects choosing the same latent node, i.e., $n^h_{-i,z}$ where $n^h_{-i,z} = I\{z_j = z, j \neq i\}$. In addition, the chance of selecting a new latent node is given by α_h/m out of m latent nodes sampled from B_h. (See [21] for more details).

5. Similar to (13), the probability of choosing ψ_ω is proportional to the number of other edges choosing the same function ($n^\psi_{-i,\omega}$) and the likelihood of this edge using this function.

TABLE 1
Summary of Gibbs Sampling in ILCRF for Two Applications

Task	Phase	Gibbs sampling (Section 4.3)				
		z (13)	$\mathbf{h}$ (14)	$\boldsymbol{\omega}$ (15)	$\boldsymbol{\psi}$ (16)	$\mathcal{Y}$ (17)
Label	train	✓	✓		✓	
	test	✓	✓	✓		✓
Arrange	train	✓	✓		✓	
	test	✓	✓			✓

analogy to topic modeling for text data [9]). Each affordance topic can be shared across multiple object types while the affordance of each object type is now represented as a mixture of multiple topics. To demonstrate, we performed a learning experiment where ω_i is also sampled during the training.[6] We are able to learn 11 affordance topics for a total of 19 object categories in our scene arrangement dataset (see Section 6.2). Fig. 7 shows some examples of learned topics and object affordance as a mixture of those topics. This ability is particularly useful when handling a large number of object types, as only a relatively small number of topics are learned yet they are able to represent the variety of object affordances.

Algorithm 1. Labeling a New Scene.

Data: $x_1, \ldots, x_N$: segment locations and appearance features.
Ψ: learned object affordances.
Result: $y_1, \ldots, y_N$: labels for each segment.
Step 1: Initialization
$B_h \leftarrow$ a uniform distribution over all possible human configurations in the scene;
$\mathcal{H} \leftarrow$ randomly sample from B_h;
$z_1, \ldots, z_N \leftarrow$ random integers between 1 and N;
$\omega_1, \ldots, \omega_N \leftarrow$ same as z_i;
Step 2: Gibbs sampling
for *each iteration* s **do**
Sample z_i using Eq. (13), $\forall i = 1, \ldots, N$;
Sample h_k using Eq. (14), $\forall k \in \{k | \exists z_i = k\}$;
Sample $\omega_i^{(s)}$ using Eq. (15), $\forall i = 1, \ldots, N$;
end
Step 3: Labeling
For each segment i, use the histogram of $\omega_i^{(s)}$ as additional affordance features (along with object self and pairwise features). Then label all segments using the max-margin classifier in [13]

4.5 ILCRF for Scene Arrangement

So far, we have presented ILCRF in the context of scene labeling task. Now we describe how to apply ILCRF to scene arrangement. While the two tasks have been studied with different approaches and algorithms in previous work, we show that they can be addressed in a unified model, ILCRF with the same definition on human poses and object affordances.

The arrangement task requires finding proper locations and orientations for placing new objects in a given scene.

6. To make sure that objects from the same category have the same affordance, instead of sampling ω_i for each object instance i in Eq. (15), we sample ω_y for each object type y, i.e., $\omega_y = \omega \propto \frac{n^{\psi}_{y,\omega}}{N+m-1+\alpha_\psi} \prod_{i:y_i=y} \psi_\omega(x_i, y_i, h_{z_i})$. Then the affordance of each object type y is given by $\frac{1}{S}\sum_s \psi_{\omega_y^{(s)}}(\cdot)$.

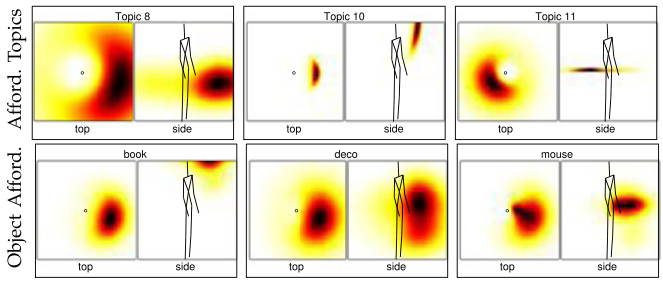


Fig. 7. Examples of learned affordance *topics* and object affordances as a mixture of those topics (see Section 4.4).

The scene is represented as an RGB-D point cloud and each object is represented by its appearance features and object class, included in x_i. Each y_i now denotes the placement (location and orientation) of an object.[7]

During training, we learn object affordances same as in the labeling task as described in the last section. We also learn the object-object structure, E_Y, based on object co-occurence, as computed from the training data. In this task, ψ^{oo} is defined as a multi-variate Gaussian distribution of the location and orientation difference between the two objects.

During testing, given the type of the object to be placed and learned object affordances, we sample human-object edges, human poses and placements. In the end, the predicted placement is the one sampled most as that represents the highest probability. The inference algorithms for both tasks are summarized in Algorithm 1 and Algorithm 2.

Algorithm 2. Arranging a New Scene.

Data: $x_1, \ldots, x_N$: object types and appearance features.
Ψ: learned object affordances.
Result: $y_1, \ldots, y_N$: locations and orientations.
Step 1: Initialization
$B_h \leftarrow$ a uniform distribution over all possible human configurations in the scene;
$\mathcal{H} \leftarrow$ randomly sample from B_h;
$z_1, \ldots, z_N \leftarrow$ random integers between 1 and N;
$\omega_1, \ldots, \omega_N \leftarrow$ given by x_i;
$y_1, \ldots, y_N \leftarrow$ randomly placed in the scene;
Step 2: Gibbs sampling
for *each iteration* s **do**
Sample z_i using Eq. (13), $\forall i = 1, \ldots, N$;
Sample h_k using Eq. (14), $\forall k \in \{k | \exists z_i = k\}$;
Sample $y_i^{(s)}$ using Eq. (17), $\forall i = 1, \ldots, N$;
end
Step 3: Placing
for $i = 1, \ldots, N$ **do**
$y_i \leftarrow \arg\max_y \sum_s \mathbf{1}\{y_i^{(s)} \in \text{neighborhood}(y)\}$;
end

5 Related Work

5.1 Variants of CRFs

Variants of Conditional Random Fields ([23]) have emerged as a popular way to model hidden states and have been successfully applied to many vision problems.

7. Since the object's placement is given either by x as in the labeling task or by y as in the arrangement task, our object affordance is defined as a function of both x and y as in Eq. (2).

There are many models that enrich the structure of labels in CRFs. For example, latent CRFs [24] assume that the overall label Y depends on a sequence of hidden states $(s_1, s_2, \ldots, s_k)$ (see Fig. 3(bottom)). This can be applied to object recognition (an object label is determined by its part labels) [15] and gesture recognition [25], [26]. Further, factorial (or dynamic) CRFs [27] substitute every label with a Markov network structure to allow structured labeling, especially for sequential data (such as labeling object and action simultaneously in video sequences [28], [29]). However, the labels and hidden states are discrete and take only finite number of values. In contemporary work, Bousmalis et al. [30] present a model that shares a name similar to ours, but is quite different. They estimate the correct number of values a latent node can take using Dirichlet processes in a way similar to augmenting hidden Markov models (HMM) to infinite HMM [31]. However, the number of hidden nodes is fixed in their model. In our model, we estimate the number of latent nodes, and even allow the labels to be continuous.

Some works impose a non-parametric Bayesian prior to the network's structure so that it can potentially generate as many nodes as needed. For example, Indian Buffet process [32] assumes the latent nodes and links are generated through Beta processes and the infinite factorial HMM [33] incorporates it to HMM to allow any number of latent variables for each observation. However, they are limited to binary Markov chains and do not consider different types of potential functions either. Thus these models are complementary to ours. Jancsary et al. [19] considers Gaussian CRFs on fixed number of nodes but unknown number of potential functions and proposes a non-parametric method to learn the number as well as parameters of each potential function. Unlike this work, our model can handle unknown number of *nodes* as well as *types of edges*.

Cast in the light of mixture models, mixtures of graphical models have been proposed to overcome the limited representational power that a single graphs often suffers. For example, Anandkumar et al. [34] propose a novel method to estimate a mixture of a finite number of discrete graphs from data. Other works consider a Dirichlet process mixture model over graphs so that the number of different graphical models is determined by the data [35], [36]. However, they are limited to Gaussian graphical models and do not consider latent variables.

5.2 Scene Understanding

A direct application of modeling the scenes is object detection, which has been explored mostly through object-object context. There is a significant body of work that captures the relations between different objects in 2D images (e.g., [37], [38], [39]) or in 3D point clouds (e.g., [12], [13], [14]) to improve object detection. However, none of these works consider human context for scene labeling. Recently, Grabner et al. [5] propose a chair detector by checking if the object can afford a sitting human pose. While it shows the importance of considering object affordance, it is limited to one single pose and one hand-designed affordance ('sittable'). In this work, we consider general human poses and a generic form of object affordance. Other recent works use object affordances for predicting human workspaces [6], predicting 3D geometry [7], and for improving human robot interactions [40].

Another promising application of modeling the scene is arranging and placing performed by personal robots. To our best knowledge, there is little work about arranging/placing objects in robotics (e.g., [41], [42], [43], [44]). There are some works in computer graphic that considered of arranging rooms that follow certain object-object constraints [45] or finding the most visually relevant object at a given location [46]. However, unlike ours, all these works do not consider human-object relations that potentially also drive how a scene is arranged.

6 Experiments

We test our ILCRF model in two applications: object detection and object arrangement. Given a room, the first task requires to identify existing objects, and the second task asks to place more designated objects in proper locations and orientations.

In our application, the scenes (including objects/furnitures) are perceived as point-clouds (Fig. 16), either generated from 3D models in synthetic datasets or obtained using Microsoft Kinect camera in real datasets.

6.1 Scene Labeling Results

In this experiment, the goal is to label each segment in a given room with correct class, such as table, chair-back, keyboard, etc.

Dataset. We used the Cornell RGB-D indoor dataset [10], [13] for our experiments. This data consists full-scene RGB-D point clouds of 52 offices and homes obtained from 550 RGB-D views. The point-clouds are over-segmented, and the goal is to label these segments with object labels and attribute labels. Each segment can have multiple attribute labels but has only one object label. The attribute labels are: {*wall, floor, flat-horizontal-surfaces, furniture, fabric, heavy, seating-areas, small-objects, table-top-objects, electronics*} and the object labels are: {*wall, floor, tableTop, tableDrawer, tableLeg, chairBackRest, chairBase, chairBack, monitor, printerFront, printerSide, keyboard, cpuTop, cpuFront, cpuSide, book, paper, sofaBase, sofaArm, sofaBackRest, bed, bedSide, quilt, pillow, shelfRack, laptop*}.

Baselines. We perform four-fold cross-validation where we train the model on data from three folds and tested on the fourth fold of unseen data. Table 2 presents the results for object labeling and attribute labeling. In order to study the effects of different algorithms, we compare with the following algorithms:

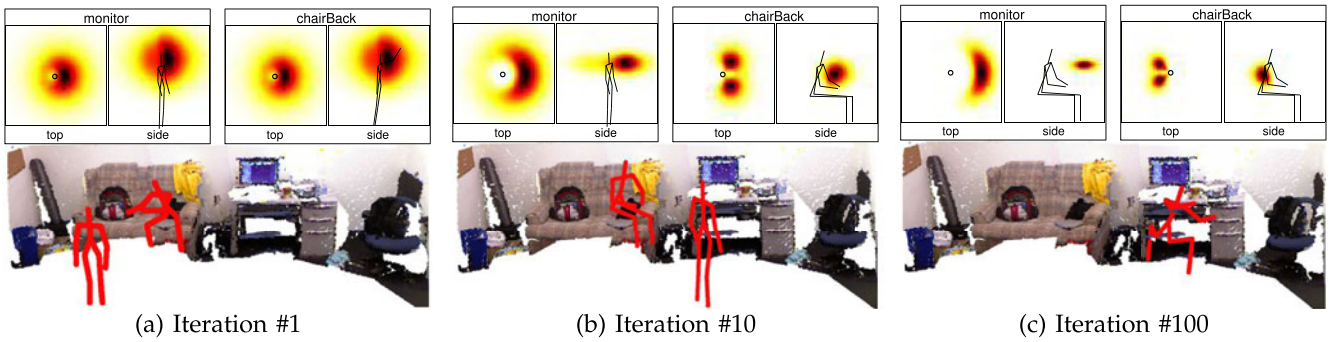
(a) Affordances (Human Context). This is our affordance and human configurations information being used in prediction, without any object context.

(b) Appearance. We run versions with both local image and shape features [13].

(c) Afford. + Appear. It combines the affordance and appearance features.

(d) Object context. We use the learning algorithm presented in [13] that uses Markov Random Field with log-linear node and pairwise edge potentials.

(e) Our ILCRF. Here we combine the human context (from affordances and human configurations) with object-object context. In detail, we append the node features of



(a) Iteration #1 (b) Iteration #10 (c) Iteration #100

Fig. 8. **Learning object affordances**. This example shows the learned object affordances (top row, shown as heatmaps) and top sampled human configurations (bottom) through iterations. In Iteration#1, the affordance is only based on the prior B_ψ which is same for all objects. Thus, the sampled human poses also randomly appear in the scene. In later iterations, the affordances diverge to different but reasonable functions and so do the sampled humans based on these affordances.

Fig. 9. **Top sampled human poses in different scenes.** The first two are from stitched point-cloud from multiple RGB-D views, and the last three scenes are shown in RGB-D single views.

each segment with the affordance topic proportions derived from the learned object-affordance topics and learn the semantic labeling model as described in [13].

Evaluation metrics. We report precision and recall using both micro and macro aggregation. Since we predict only one label for each segment in case of predicting object labels, our micro precision and recall is the same as the percentage of correctly classified segments (shown as 'P/R' in Table 2). The macro precision and recall are the average of precision and recall of all classes respectively.

Results. Table 2 shows that our algorithm performs better than the state-of-the-art in both object as well as attribute labeling experiment. Our approach is able to predict the correct labels for majority of the classes as can be seen from the strong diagonal in the confusion matrices. We discuss our results in the light of the following questions.

Are the sampled human poses meaningful? Being able to hallucinate sensible human poses is critical for learning object affordances. To verify that our algorithm can sample meaningful human poses, we plot a few top sampled poses in the scenes, shown in Fig. 9. In the first home scene, some sampled human poses are sitting on the edge of the bed while others standing close to the desk (so that they have easy access to objects on the table or the shelf-rack). In the next office scene (Fig. 9b), there is one L-shaped desk and two chairs on each side. It can be seen that our sampled human poses are not only on these chairs but also with correct orientation. Also, as can be seen in Fig. 8c, our algorithm successfully identifies the workspaces in the office scene. Note that these poses naturally explain why the monitors, keyboards and CPUs are arranged in this particular way. It is these correctly sampled human poses that give us the possibility to learn correct object affordances.

Are the discovered affordances meaningful? During training, we are given scenes with the objects and their labels, but not humans. Our goal is to learn object affordance for each class. Fig. 10 shows the affordances from the top-view and side-view respectively for typical object classes. Here the X-Y dimensions of the box are 5 m×5 m, and the height axis's range is 3 m. The person is in the center of the box. From the side views, we can see that for objects such as wall and cpuTop, the distributions are more spread out compared to objects such as floor, chairBase and keyboard. This is because that that chairBase is often associated with a sitting pose at similar heights, while CPUs can either be on the table or on the floor.[8] While this demonstrates that our method can learn meaningful affordances, we also observe certain biases in our affordances. For example, the wall is more to the front as compared to the back, and monitor is biased to the side. We attribute to the limited data and imperfect generation of valid human skeletons. Note that while the affordance topics are unimodal, the affordance for

TABLE 2
Object and Attribute Labeling Results

Algorithm	Object Labeling			Attribute Labeling			
	micro	macro		micro		macro	
	P/R	prec	recall	prec	recall	prec	recall
Chance	5.88	5.88	5.88	12.50	12.50	12.50	12.50
Affordances	31.38	16.33	15.99	50.93	34.06	42.02	28.02
Appearance	67.24	53.31	50.48	81.81	60.85	73.30	52.36
Afford. + Appear.	68.63	55.69	52.86	83.04	63.95	78.85	56.00
Object context [13]	78.72	68.67	63.72	85.52	70.98	80.04	63.07
ILCRF	**78.86**	**71.14**	**65.07**	**85.91**	**73.51**	**82.76**	**69.22**

The table shows average micro precision/recall, and average macro precision and recall for 52 scenes. Computed with four-fold cross-validation.

8. In fact, the learned object affordances (Fig. 10) haven been applied to Robo Brain [47], showing the possibility of sharing learned knowledge across different research domains.

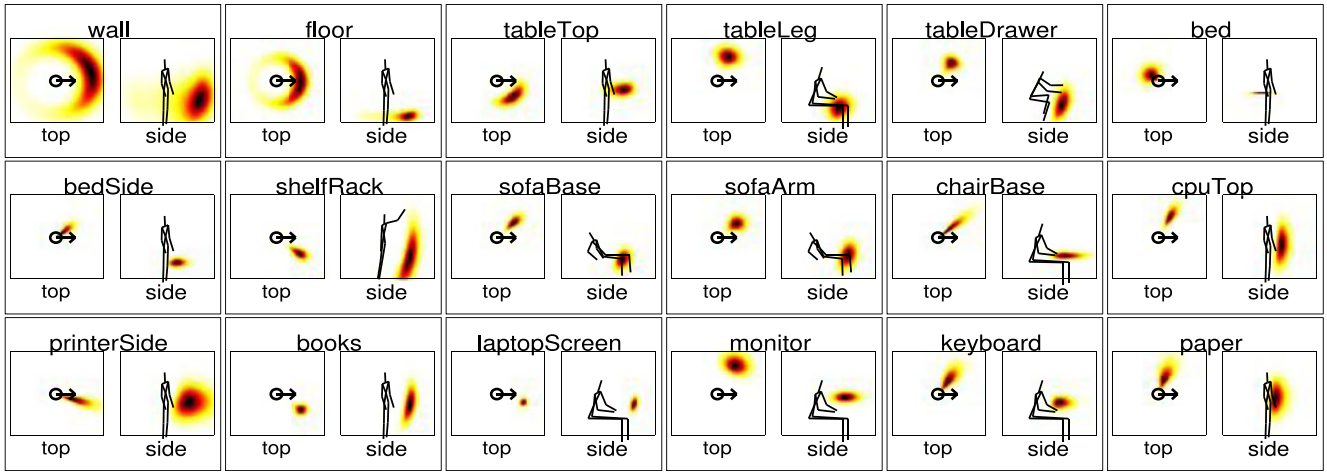


Fig. 10. **Examples of learned object-affordance topics.** An affordance is represented by the probability distribution of an object in a $5 \times 5 \times 3$ space given a human pose. We show the projected top views and side views for different object classes.

each objects is a mixture of these topics and thus could be multi-modal and more expressive.

Can we obtain object-object relations from object affordances? Since objects are related to humans, it turns out that we can infer object-object spatial relations (and object co-occurences) from the human-object relations. For example, if we convolve keyboard-human and human-monitor relations, we obtain the spatial relations between keyboard and monitor. More formally, we compute the conditional distribution of one object's location x_i (with type y_i) given another object's location x_j (with type y_j) as,

$$P(x_i|x_j) = \int P(x_i|h)P(h|x_j)dh$$
$$\propto \int \psi_{ho}(x_i, y_i, h)\psi_{ho}(x_j, y_j, h)B_h(h)dh.$$

Some examples are shown in Fig. 11. We can find that many object-object relationships are recovered reasonably from our learned affordances. For example, given a keyboard, a monitor is likely to be found in front of and above it while tableTop at the same height as it (sometimes above it as the keyboard is often in a keyboard-tray in offices). In home scenes, given a bed, we can find a pillow on the head of the bed, quilt right above the bed and bedSide slightly below it. This supports our hypothesis that object-object relations are only an artifact of the hidden context of human-object relations. It also demonstrates that we can efficiently model $O(n^2)$ object-object relations for n objects using only $O(n)$ human-object parameters.

Does human context helps in scene labeling? Table 2 shows that the affordance topic proportions (human context) as extra features boosts the labeling performance. First, when combining human context with the image- and shape-features, we see a consistent improvement in labeling performance in all evaluation metrics, regardless of the object-object context. Second, when we add object-object context, the performance is further boosted in the case of office scenes and improves marco precision for home scenes. This indicates that there is some orthogonality in the human-object context and object-object context. In fact, adding object-object context to human-object context was particularly helpful for small objects such as keyboards and books that are not always used by humans together, but still have a spatial correlation between them.

We also show the confusion matrices in Fig. 12. We found that while our algorithm can distinguish most of the objects, it sometimes confuses objects with similar affordance. For example, it confuses pillow with quilt and confuses book and paper with tableTop. Similarly, it confuses cpuTop with chairBase because the CPU-top (placed on the ground) could also afford sitting human poses!

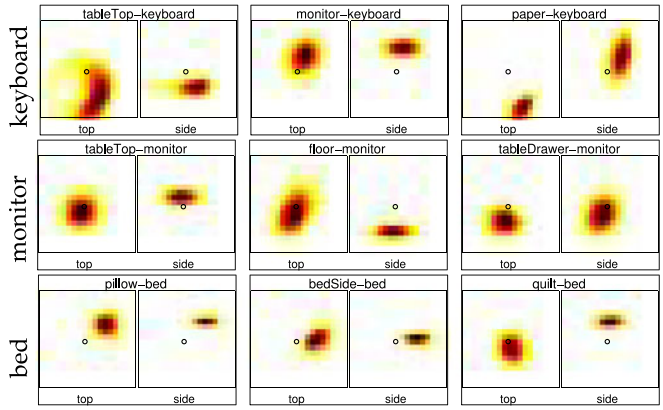


Fig. 11. **Object-object context obtained from our learned human context.** Each pair of the top- and side-view of a heatmap with the title of 'obj1-obj2' shows the distribution of obj1 given obj2 at the center facing right. For example, in the first row the keyboard is in the center of the image and the heat-maps show the probability of finding other related objects such as table top, monitor, etc.

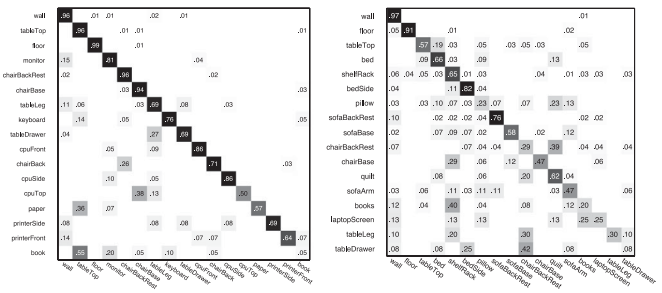


Fig. 12. **Confusion matrices** for office dataset (left) and home dataset (right) using our ILCRF model.

TABLE 3
Scene Arrangement Results on Partially-Filled Scenes and Empty Scenes in Synthetic Dataset, Evaluated by the Location and Height Difference to the Labeled Arrangements

Algorithms	partially-filled scenes		empty scenes	
	location (m)	height (m)	location (m)	height (m)
Chance	2.35±0.23	0.41±0.04	2.31±0.23	0.42±0.05
Obj. [1]	1.71±0.23	0.13±0.02	2.33±0.17	0.44±0.04
CRF	1.69±0.05	0.12±0.01	2.17±0.07	0.39±0.01
ILCRF-H [1]	1.48±0.18	0.11±0.01	1.65±0.20	0.12±0.01
Human+obj [1]	1.44±0.18	**0.09** ±0.01	1.63±0.19	0.11±0.01
ILCRF-Aff.	1.59±0.06	0.14±0.01	1.60±0.06	0.15±0.01
ILCRF-NSH	1.64±0.05	0.15±0.01	1.77±0.06	0.16±0.01
FLCRF	1.55±0.06	0.12±0.01	1.63±0.06	0.14±0.01
ILCRF	**1.33±0.19**	**0.09±0.01**	**1.52±0.06**	**0.10±0.01**

TABLE 4
Scene Arrangement Results on Five Real Empty Scenes (Three Offices and Two Apartments)

	office1		office2		office3		apt1		apt2		AVG	
	Co	Sc	Co	Sc	Co	Sc	Co	Sc	Co	Sc	Co	Sc
Obj.	100	4.5	100	3.0	45.0	1.0	20.0	1.8	75.0	3.3	68.0	2.7
ILCRF-H	100	**5.0**	100	4.3	91.0	4.0	74.0	3.5	88.0	4.3	90.0	4.2
Human+obj	100	4.8	100	4.5	92.0	4.5	89.0	4.1	81.0	3.5	92.0	4.3
ILCRF	**100**	**5.0**	**100**	**4.6**	**94.0**	**4.6**	**90.0**	**4.1**	**90.0**	**4.4**	**94.8**	**4.5**

***Co**: Percent of semantically correct placements, **Sc**: average score (0-5).*

6.2 Scene Arrangement Results

In this experiment, the goal is to find proper locations and orientations for placing one or multiple objects in a given room.

Dataset. We test on a synthetic dataset and a real dataset. We downloaded 20 different rooms from Google 3D Warehouse, including six living rooms, seven kitchens and seven offices. All these scenes are commonly seen in the real world and have different layouts and sizes. We also collected 47 different objects from 19 categories for arranging: *{book, clean tool, laptop, monitor, keyboard, mouse, pen, decoration, dishware, pan, cushion, TV, desk light, floor light, utensil, food, shoe, remote control, and phone}*. Every room is assigned to three to five subjects (not associated with the project) to manually label the arrangements of 10 to 30 objects. In total, we have 67 different labeled arrangements for 20 rooms.

We also test on *real scenes* from [44] using the learned model from the synthetic dataset. The real dataset consists of five empty offices and apartments, each of which is asked to arrange 4, 18, 18, 21 and 18 number of objects respectively.

Experimental setup. For the synthetic dataset, we conduct five-fold cross validation on 20 rooms such that the four test rooms are new to the algorithms. We consider two different testing scenarios, where the test room is either: *partially-filled* and the task is to arrange one new type of objects (may have multiple instances); or *empty* (with only furnitures) and the task is to arrange multiple types of objects.

Baselines. We compare all the following methods:

1) Chance. Objects are placed randomly in the room.

2) Obj. We use heuristic object-object spatial relations to infer placements in sequence (not jointly).[9]

3) CRF, an ILCRF with only object-object edges (i.e., (y_i, y_j)), without latent human nodes.

4) ILCRF-H, an ILCRF with only human-object edges (i.e., (y_i, h_{z_i})), without considering object relations.

5) Human+obj, a heuristic way combining object context and human context. It linearly combines the inferred distributions of arrangements $\mathcal{Y}$ from Obj. and from ILCRF-H, and select the maximum. Our ILCRF, on the other hand, incorporate the two relationships during the inference, not after.

6) ILCRF-Aff, an ILCRF with only one type of edge, i.e., one shared affordance across all object classes.

7) ILCRF-NSH, an ILCRF with non-sharing latent human nodes. Each object is assigned with its own human node, i.e., $z_i = i$ for each y_i, similar to *hidden CRFs* in Fig. 3. While this model can still affect the object arrangements through possible human poses (e.g., monitor will be placed near any sitting area), it cannot capture phenomena of objects sharing the same human pose, such as a monitor and a keyboard being placed together. ILCRF achieves this ability of sharing latent nodes through the clustering effect (on z_i's) inherited from DPs.

8) FLCRF, an ILCRF with fixed/finite number of latent nodes (same number of human poses across all scenes). It requires a good estimate on the number of human poses, and the optimal number may vary for rooms of different types or sizes.

9) ILCRF, our full model.

Evaluation metrics. For synthetic datasets, the predicted arrangements are evaluated by two metrics, same as in [1]: location difference and height difference (in meters) to the labeled arrangements (averaged over different object types across all test rooms). The results are shown in Table 3.

Results of arranging empty real scenes, shown in Table 4, are evaluated by two human subjects that are not associated with this project: Each arrangement is measured by the percentage of predicted locations that are semantically correct and a score of the overall arrangement between 0 and 5.

Results. Results in Table 3 demonstrate, same as the previous experiment, that modeling human context does improve the performance: On average, the location and height difference are reduced from 1.69 m (2.17 m) and $.12$ m ($.39$ m) when modeling object context only using CRF, to 1.33 m (1.52 m) and $.09$ m ($.10$ m) when modeling both human and object context using ILCRF, in arranging partially-filled (empty) scenes. Even methods that use non-sharing skeletons (ILCRF-NSH) and finite skeletons (FLCRF) achieve better results than CRF. We also visually compare some predicted arrangements for empty rooms (Fig. 13), where using object relations only often leads to over-crowded arrangements (especially in empty rooms) or inconvenient/inaccessible locations due to the lack of human context. In the following, we study how well the latent human context is modeled by ILCRF.

Why do we need handle unknown number of human poses? The advantage of using DP mixture models in ILCRF is being able to determine the number of human poses from

9. We model the relative location/orientation between any pair of object types as Gaussian distributions with parameters learned from training data. For placing a new object, a reference object (already placed in the room) is selected with the smallest variance and then sample the new object's location/orientation from the Gaussian distributions.

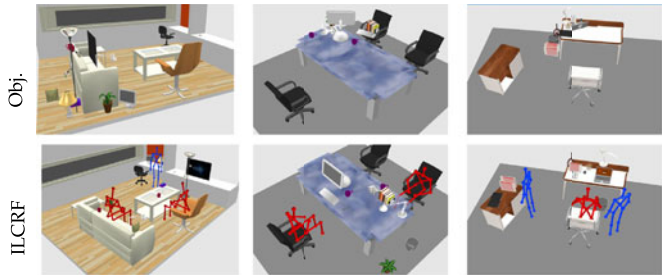


Fig. 13. Results of arranging empty rooms by using object-object relationships only (top) and by ILCRFs (bottom).

the data instead of guessing manually. We investigate this in in Fig. 14. We compare ILCRF with the FLCRF where the number of human poses varyies from 1 to 20.

While having five poses in FLCRF gives the best result, it is still outperformed by ILCRF. This is because scenes of different sizes and functions prefer different number of skeletons. If we force all scenes using only one human pose, the learned object affordances will have large variances, e.g., in Fig. 14b. If we force all scenes using a large number of human poses, say 20 per scene, the model will overfit in each scene and leading to meaningless affordances, e.g., Fig. 14c. Therefore, having the correct number of latent human nodes in CRFs is crucial for learning good object affordances as well as for inferring reasonable arrangements across diverse scenes (Fig. 14a).

How sensitive is ILCRF to the number of human poses? The parameter α_h in ILCRF controls the probability of selecting a new human pose and thus can be viewed as a counterpart of K (the fixed number of human poses) in FLCRF. However, unlike FLCRF, ILCRF is much less sensitive to this

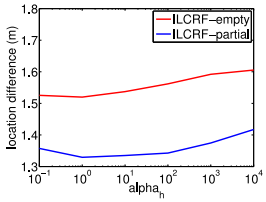


Fig. 15. The average performance of ILCRF with different hyper-parameter α_h.

parameter, as shown in Fig. 15 where its performance does not vary much for α_h from 0.1 to 10^4. Therefore, ILCRF does not rely on either informative prior knowledge or a careful hand-picked value of α_h to achieve high performance.

6.3 Robotic Experiment

Robotic simulation experiment. In order to study how the desired placements are affected by the robot constraints, we tested arranging these synthetic scenes using Kodiak (PR2) in simulation. Please refer to [2] for more details on results.

Arrange real scenes. We apply the ILCRF to our Kodiak PR2 robot to perform the scene arrangement in practice. We test our system on a small set of objects (a cushion, mouse and mug) in a given scene (Fig. 16). The system works as follows: (a) The robot perceives the environment as point clouds; (b) It hallucinates human poses and detect objects using ILCRF; (c) When asked to place a new object, it hallucinates human poses as well as sample the object's locations. The most sampled location will be the final prediction; (d) The robot executes the arrangement by placing the object at the predicted location. To see PR2 arranging the scene in action (along with code and data), please visit: `http://pr.cs.cornell.edu/hallucinatinghumans`

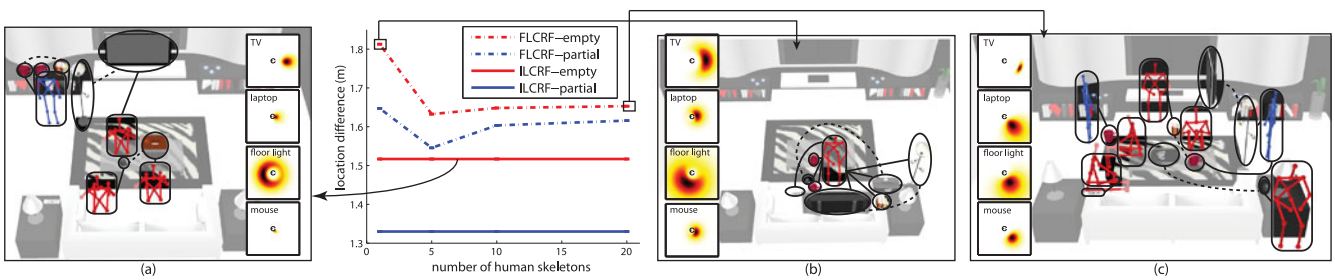


Fig. 14. Results of FLCRF with different number of human poses versus ILCRF. We also show examplar sampled CRFs and learned object affordances (in top-view heatmaps) by different methods.

Fig. 16. **Robotic experiment** (from left to right): (a) A given scene is perceived as a RGB-D point-cloud; (b) The robot uses ILCRF to detect objects; (c) The robot uses ILCRF to infer possible human poses (shown in red heatmaps) and possible placements (shown in blue heatmaps) for placing a cushion (top) and a mouse (bottom) in the scene; (d) The robot successfully places objects in the predicted locations.

7 Conclusion

In this paper, we considered two challenging problems of 3D scene labeling and scene arrangement, which requires an algorithm that can handle: 1) unknown number of latent nodes (for potential human poses), 2) unknown number of edge types (for human-object interactions), and 3) a mixture of different CRFs (for the whole scene). We therefore presented a new algorithm, called Infinite Latent Conditional Random Fields, together with learning and inference algorithms. Through extensive experiments and thorough analyses, we not only showed that our ILCRF algorithm outperforms the state-of-the-art results, but we also verified that modeling latent human poses and their relationships to objects are crucial to reason our environment. We also implemented our algorithm on a robot. It correctly inferred potential human poses and object labels and arrangements in real scenes.

Future directions include designing a richer affordance representation, improving the learning and inference algorithm for our ILCRF, and applications to domains such as computer graphics and robotics.

Acknowledgments

The authors thank Marcus Lim for useful discussions and help in the experiments. This work was supported by ARO award W911NF-12-1-0267, the US National Science Foundation (NSF) Career Award, and Microsoft Faculty Fellowship to Saxena. Parts of this work have been published as [1], [2], [3], [4] as conference papers.

References

[1] Y. Jiang, M. Lim, and A. Saxena, "Learning object arrangements in 3d scenes using human context," in *Proc. Int. Conf. Mach. Learn.*, 2012, pp. 1543–1550.

[2] Y. Jiang and A. Saxena, "Hallucinating humans for learning robotic placement of objects," in *Proc. 13th Int. Symp. Experimental Robot.*, 2012, pp. 921–937.

[3] Y. Jiang, H. Koppula, and A. Saxena, "Hallucinated humans as the hidden context for labeling 3D scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2993–3000.

[4] Y. Jiang and A. Saxena, "Infinite latent conditional random fields for modeling environments through humans," in *Proc. RSS*, 2013.

[5] H. Grabner, J. Gall, and L. J. V. Gool, "What makes a chair a chair?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1529–1536.

[6] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert, "From 3d scene geometry to human workspace," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1961–1968.

[7] V. Delaitre, D. Fouhey, I. Laptev, J. Sivic, A. Gupta, and A. Efros, "Scene semantics from long-term observation of people," in *Proc. 12th Eur. Conf. Comput. Vis.*, 2012, pp. 284–298.

[8] B. Yao and L. Fei-Fei, "Modeling mutual context of object and human pose in human-object interaction activities," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 17–24.

[9] Y. W. Teh, "Dirichlet process," in *Encyclopedia of Machine Learning*. New York, NY, USA: Springer, 2010.

[10] H. Koppula, A. Anand, T. Joachims, and A. Saxena, "Semantic labeling of 3d point clouds for indoor scenes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 242–252.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 886–893.

[12] X. Xiong and D. Huber, "Using context to create semantic 3d models of indoor environments," in *Proc. Brit. Mach. Vis. Conf.*, 2010, pp. 45.1–45.11.

[13] A. Anand, H. Koppula, T. Joachims, and A. Saxena, "Contextually guided semantic labeling and search for 3D point clouds," *Int. J. Robot. Res.*, vol. 32, no. 1, pp. 19–34, 2012.

[14] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 746–760.

[15] P. Schnitzspan, S. Roth, and B. Schiele, "Automatic discovery of meaningful object parts with latent CRFs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 121–128.

[16] A. Saxena, S. Chung, and A. Ng, "Learning depth from single monocular images," in *Proc. Adv. Neural Inf. Process. Syst. 18*, 2005, pp. 1161–1168.

[17] A. Quattoni, M. Collins, and T. Darrell, "Conditional random fields for object recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 1097–1104.

[18] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Human activity detection from RGBD images," in *Proc. In. Conf. Robot. Autom.*, 2012.

[19] J. Jancsary, S. Nowozin, and C. Rother, "Non-parametric CRFs for image labeling," in *Proc. NIPS Workshop Modern Nonparametric Methods Mach. Learn.*, 2012.

[20] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Depth from familiar objects: A hierarchical model for 3d scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006.

[21] R. Neal, "Markov chain sampling methods for Dirichlet process mixture models," *J. Comput. Graph. Statist.*, vol. 9, no. 2, pp. 249–265, 2000.

[22] S. Kumar and M. Hebert, "Discriminative random fields: A discriminative framework for contextual interaction in classification," in *Proc. IEEE 9th Int. Conf. Comput. Vis.*, 2003, pp. 1150–1157.

[23] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learn.*, 2001, pp. 282–289.

[24] A. Quattoni, S. Wang, L. Morency, M. Collins, and T. Darrell, "Hidden conditional random fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1848–1852, Oct. 2007.

[25] S. Wang, A. Quattoni, L. Morency, D. Demirdjian, and T. Darrell, "Hidden conditional random fields for gesture recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 1521–1527.

[26] Y. Wang and G. Mori, "Max-margin hidden conditional random fields for human action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 872–879.

[27] C. Sutton, K. Rohanimanesh, and A. McCallum, "Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data," in *Proc. Int. Conf. Mach. Learn.*, 2004, pp. 783–790.

[28] H. Kjellström, J. Romero, and D. Kragić, "Visual object-action recognition: Inferring object affordances from human demonstration," *Comput. Vis. Image Understanding*, vol. 115, no. 1, pp. 81–90, 2011.

[29] H. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from RGB-D videos," *Int. J. Robot. Res.*, vol. 32, pp. 951–970, 2013.

[30] K. Bousmalis, L. Morency, S. Zafeiriou, and M. Pantic, "A discriminative nonparametric Bayesian model: Infinite hidden conditional random fields," in *Proc. NIPS Workshop Bayesian Nonparametrics*, 2011.

[31] M. Beal, Z. Ghahramani, and C. Rasmussen, "The infinite hidden Markov model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 1088–1095.

[32] T. Griffiths and Z. Ghahramani, "The Indian buffet process: An introduction and review," *J. Mach. Learn. Res.*, vol. 12, pp. 1185–1224, 2011.

[33] J. Van Gael, Y. W. Teh, and Z. Ghahramani, "The infinite factorial hidden Markov model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1697–1704.

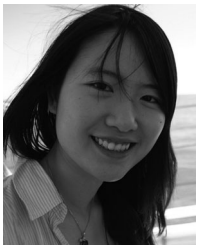
[34] A. Anandkumar, D. Hsu, F. Huang, and S. Kakade, "Learning mixtures of tree graphical models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1052–1060.

[35] A. Rodriguez, A. Lenkoski, and A. Dobra, "Sparse covariance estimation in heterogeneous samples," *Elec. J. Statist.*, vol. 5, pp. 981–1014, 2011.

[36] K. Ickstadt, B. Bornkamp, M. Grzegorczyk, J. Wieczorek, M. Sheriff, H. Grecco, and E. Zamir, "Nonparametric Bayesian networks," *Bayesian Statist.*, vol. 9, pp. 283–316, 2010.

[37] G. Heitz and D. Koller, "Learning spatial context: Using stuff to find things," in *Proc. 10th Eur. Conf. Comput. Vis.*, 2008, pp. 30–43.

[38] D. C. Lee, A. Gupta, M. Hebert, and T. Kanade, "Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1288–1296.
[39] V. Hedau, D. Hoiem, and D. Forsyth, "Thinking inside the box: Using appearance models and context based on room geometry," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 224–237.
[40] A. Pandey and R. Alami, "Taskability graph: Towards analyzing effort based agent-agent affordances," in *Proc. IEEE RO-MAN*, 2012, pp. 791–796.
[41] A. Edsinger and C. Kemp, "Manipulation in human environments," in *Proc. IEEE-RAS 6th Int. Conf. Humanoid Robots*, 2006, pp. 102–109.
[42] M. Schuster, J. Okerman, H. Nguyen, J. Rehg, and C. Kemp, "Perceiving clutter and surfaces for object placement in indoor environments," in *Proc. IEEE-RAS 10th Int. Conf. Humanoid Robots*, 2010, pp. 152–159.
[43] D. Jain, L. Mosenlechner, and M. Beetz, "Equipping robot control programs with first-order probabilistic reasoning capabilities," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 3626–3631.
[44] Y. Jiang, M. Lim, C. Zheng, and A. Saxena, "Learning to place new objects in a scene," *Int. J. Robot. Res.*, vol. 31, pp. 1021–1043, 2012.
[45] Y.-T. Yeh, L. Yang, M. Watson, N. D. Goodman, and P. Hanrahan, "Synthesizing open worlds with constraints using locally annealed reversible jump MCMC," in *Proc. SIGGRAPH*, vol. 31, no. 4, pp. 1–56, 2012.
[46] M. Fisher, M. Savva, and P. Hanrahan, "Characterizing structural relationships in scenes using graph kernels," in *Proc. SIGGRAPH*, vol. 30, no. 4, pp. 1–34, 2011.
[47] A. Saxena, A. Jain, O. Sener, A. Jami, D. K. Misra, and H. S. Koppula, "RoboBrain: Large-scale knowledge engine for robots," *arXiv preprint arXiv:1412.0691*, Aug. 2014.



Yun Jiang is currently working toward the PhD degree in the Computer Science Department, Cornell University. Her primary research lies in machine learning and its applications to manipulation for personal robots. She is interested in understanding how people interact with the environment and how can robots make use of this understanding to better assist in the human environment. She has developed several machine learning algorithms for robots to grasp, place and arrange objects in a human-like way. She is a student member of the IEEE.

Hema S. Koppula is currently working toward the PhD degree in the Computer Science Department, Cornell University. Her research lies at the intersection of machine learning, computer vision and robotics: she is interested in understanding people from visual data to build smart assistive devices. She has developed machine learning algorithms for perceiving environments from RGB-D sensors such as scene understanding, activity detection and anticipation. She has received the Best Student Paper Award at RSS and is a Google PhD fellow.

Ashutosh Saxena received the the BTech degree in 2004 from IIT Kanpur, India, and the PhD degree in 2009 from Stanford University. He is in the Faculty of the Computer Science Department, Cornell University. His research interests include machine learning, robotics, and computer vision. He has received Best Paper Awards in 3DRR, RSS, and IEEE ACE. He was named a co-chair of IEEE technical committee on robot learning. He has also received Sloan Fellowship in 2012, the US National Science Foundation (NSF) Career award in 2013, and RSS Early Career Award in 2014. He has developed robots that perform household chores such as unload items from a dishwasher, arrange a disorganized house, checkout groceries, etc. Previously, he has developed Make3D, an algorithm that converts a single photograph into a 3D model. He is a member of the IEEE.