Hallucinating Humans for Learning Robotic Placement of Objects

Yun Jiang and Ashutosh Saxena

Department of Computer Science, Cornell University, Ithaca, NY 14853, USA.
{yunjiang,asaxena}@cs.cornell.edu

Abstract. While a significant body of work has been done on grasping objects, there is little prior work on placing and arranging objects in the environment. In this work, we consider placing multiple objects in complex placing areas, where neither the object nor the placing area may have been seen by the robot before. Specifically, the placements should not only be stable, but should also follow human usage preferences. We present learning and inference algorithms that consider these aspects in placing. In detail, given a set of 3D scenes containing objects, our method, based on Dirichlet process mixture models, samples human poses in each scene and learns how objects relate to those human poses. Then given a new room, our algorithm is able to select meaningful human poses and use them to determine where to place new objects. We evaluate our approach on a variety of scenes in simulation, as well as on robotic experiments.

1 Introduction

"Tidy my room." "Put the dishes away." — While these tasks would have been easy for *Rosie* robot from *The Jetsons* TV show, they are quite challenging for our robots to perform. Not only would they need to have the basic manipulation skills of picking up and placing objects, but they would also have to perform them in a way that respects human preferences, such as not placing a laptop in a dish-rack or placing the dishes under the bed.

Over the last few decades, there has been a significant body of work on robotic grasping of objects (e.g., [1–10]). However, there is little previous work on teaching robots where and how to *place* the objects after picking them up. Placing an object is challenging for a robot because of the following reasons:

- *Stability.* An object needs to be placed in a correct orientation for it to be stable. For example, while a martini glass could be placed upright on a flat surface, it is stable when hanging upside down in a wine glass rack.
- *Novel objects and placing areas.* An unstructured human environment comprises a large number of objects and placing areas, both of which may have complex geometry and may not have been seen by the robot before. Inferring stable placements in such situation requires robust algorithms that generalize well.
- *Human preferences.* The objects should be placed in meaningful locations and orientations that follow human preferences. For example, a laptop should be facing the chair when placed on a table.

(a) random arrangement (b) object-context arrangement

(c) human-context arrangement (d) arranging the real scene

Fig. 1. An example of arranging three objects in a room. (a) A random arrangement may lead to unstable placement (such as the laptop tilted on the fridge) and stable but unreasonable placement (such as shoes on the table and food on the ground). (b) Therefore, we adopt supervised learning based on a variety of appearance and shape features for finding stable orientations and meaningful locations for placing the objects. However, the placed objects are still hard for human to access, such as the fruit stored high up on the fridge and the laptop towards the wall. (c) We thus further improve the arrangement by considering the relationship between the objects and humans (such as a sitting pose in the chair and a reaching pose in front of the fridge). (d) Our approach does not require human poses at present but samples them based on Dirichlet processes and learned potential functions. The last arrangement not only places the objects appropriately but is also ready for human to use.

In our recent work [11, 12], we proposed a learning algorithm for placing objects stably and in their preferred orientations in a given placing area (see Fig. 1b). Our approach was based on supervised learning that used a graphical model to learn a functional mapping from features extracted from 3D point-clouds to a placement's quality score. When multiple objects and placing areas were present, the inference was formulated as an integer linear program that maximized the total placement quality score. Our formulation also allowed linear stacking of objects. This enabled our robot to place objects (even the ones that were not seen previously by the robot) in areas such as dish-racks, a stemware-holder and a fridge, etc. While our model in [12] captured certain semantic preferences, it did not consider human usage preferences and therefore placements were often not meaningful. For example, placing food up in a fridge that is

hard to reach (see Fig. 1b), or placing a mouse and keyboard far away from each other making them impossible to be used together.

In this work, our goal is to learn meaningful object placements that follow human preferences, such as the arrangement in Fig. 1c. The key idea that makes a placement meaningful is how it will be used by humans: Every object is meant to be used by a human in a certain way, at a certain location and for a certain activity. For example, in an office, a keyboard is placed on a table below a monitor because a person typically uses the keyboard while sitting in the chair and watching the monitor. Such usage preferences are sometimes also called object "affordances" [13]. One naïve way to encode them would be looking up a dataset that shows examples of humans using each object. Unfortunately, no such dataset exists and the effort to construct a comprehensive one would be prohibitive.

Instead of relying on a dataset of real humans manipulating objects in 3D environments, we work with a dataset that only has arrangements of objects in different scenes.[1] Then, in order to learn the human usage preferences, we would *'hallucinate'* human poses in the 3D scene, and learn the object affordances using an unsupervised learning algorithm.

> *A hallucination is a fact, not an error; what is erroneous is a judgment based upon it.* Bertrand Russell.

What would be the key here is to learn which human poses are more likely than others and how they interact with the objects. To do this, we first define a potential function giving a score for an object and a human pose, based on their spatial features. We consider the human poses as latent variables, and model them as mixture components in a Dirichlet process (DP) mixture model and consider arranging the objects as a generative process: a room first generates several human poses; then each object chooses a human pose and is drawn from the potential function parameterized by this human pose. This model allows different objects to be used by the same human pose (e.g., using a monitor, keyboard and mouse at the same time), while a room can have as many human poses as needed (one of the DP mixture model's property) [13]. Given the most likely placements, our robot then uses path planning algorithms to compute specific placing trajectories and execute them.

Our algorithm thus learns the preferred object arrangements from the 3D scenes collected from the Internet. We first evaluate our algorithm on such datasets consisting of 20 different rooms and compare the inferred arrangements to the ground truth. We also test on five scenes using real point-clouds. Finally, we perform our algorithms on our robot on actual placements in three real scenarios.

2 Related Work

There is little work in robotic placing and arrangement of objects and most existing methods are restricted to placing (or moving) objects on flat horizontal surfaces. Edsinger and Kemp [14] and Schuster et al. [15] focused on finding flat clutter-free areas where an object could be placed. Our work considers arranging objects in the whole room with significantly more complex placing areas in terms of geometry.

[1] Such datasets are readily available on the Internet, e.g., Google 3D warehouse.

Placing objects also requires planning and high-level reasoning about the sequence of actions to be performed. Lozano-Perez [16] proposed a task-level (in contrast with motion-level) planning system for picking and placing objects on a table. Sugie et al. [17] used rule-based planning in order to push objects on a table surface. There are some recent works using symbolic reasoning engines to plan complex manipulations for human activities, such as setting a dinner table (e.g. [18–20]). However, these works focus on generating parameterized actions and task-level plans instead of finding specific placements, and hence are complementary to ours.

In our own recent work [11, 12], we employed 3D stability and geometric features to find stable and preferred placements. However, without taking human context into consideration, the generated strategy was often not good enough. In this paper, we discuss a method for combining the stability with human usage preference, and compare our approach to one that does not consider the human usage preferences in Section 4.

In this paper, learning the human usage preference, i.e., the relationship between the objects and the humans is the key. In a way, this could be called 'human context.' In other fields, such as computer vision, the idea of 'context' has helped quite a bit in improving tasks such as object recognition. For example, using estimated 3D geometric properties from images can be useful for object detection [21–26]. In [27–29], contextual information was employed to estimate semantic labels in 3D point-clouds of indoor environments. Fisher et. al. [30, 31] designed a context-based search engine using geometric cues and spatial relationships to find the proper object for a given scene. Unlike our work, their goal was only to retrieve the object but not to place it afterwards. These works are different from ours not only because they address different problems, but also because none of these works used the 'human context.'

We use sampling techniques to sample the human poses, which are never observed. In general, sampling techniques are quite common in the area of path planning [32, 33], where it is the robot pose that is sampled for constructing a path. Often modeling and sampling of human poses is also done in the area of computer graphics and animation [34], and solving the kinematics and dynamics issues of robots operating in presence of humans [35], and analyzing human body poses and actions [36–38]. However, to the best of our knowledge, our work is the first one that samples such human poses for capturing context among objects.

3 Object and Human Context

An object when placed in an environment depends both on its interaction with the placing area and its interaction with the humans. In the following sections, we first briefly review our potential function that captures the object context—relationship between the object and placing areas [11, 12]. Then we discuss how to encode human context (such as human usage preferences and access effort) in our algorithm.

Specifically, we formulate a general placing problem as follows: There are n objects $\mathcal{O} = \{O_1, \ldots, O_n\}$ to be placed in m placing areas $\mathcal{E} = \{E_1, \ldots, E_m\}$, all of which are represented as point-clouds. A placement of O_i is specified by its location ℓ_i and orientation/configuration c_i. Moreover, a placement is often associated with certain human pose for certain purpose. Let $\mathcal{H} = \{H_1, \ldots\}$ to denote all the possible human poses. Our goal is to, for each object O_i, find 1) a placing area E_j to place it at and the

specific placement (ℓ_i, c_i), and 2) a relevant human pose H_k that explains the placement well.

3.1 Object Context

By object context, we mean the relationship/interaction between the object and the placing area that determines whether the placing area can hold the object stably and, more importantly, meaningfully. For instance, books should be placed on a shelf or a table, plates are better inserted in a dish-rack, and shoes should be put on the ground instead of on a table or in a dishrack.

We capture this object-environment relationship (or object-object relationship when the objects are stacked on top of each other) using a supervised learning algorithm that learns a functional mapping, $\Psi_{\text{object}}(O_i, E_j, \ell_i, c_i)$, from a set of features representing the placement to a placing quality score. A larger value of $\Psi_{\text{object}}(\cdot)$ indicates a better placement. (Our goal then becomes to maximize the value of this function during learning and inference.)

We decompose the function into two terms:

$$\Psi_{\text{object}}(O_i, E_j, \ell_i, c_i) = \Psi_{\text{stability}}(O_i, E_j, \ell_i, c_i)\, \Psi_{\text{semantics}}(O_i, E_j). \tag{1}$$

We develop a variety of appearance and shape features to capture the stability and semantic preferences respectively [12].

As we observed in a series of experiments in [12], using this algorithm can help us to predict preferred placements for various objects and different scenes. However, because we model each object independently of others, certain connections among the objects are lost in this approach, making the arrangement often disorganized and pointless. For example, a keyboard and mouse are placed far away from each other and the desk-light faces towards the wall. The goal of capturing the connection between different objects making them usable after placing is the key motivation for introducing the human context.

3.2 Human Context

The arrangements and connections among objects can be naturally explained by human poses and activities. For example, a monitor on the desk would mean that a human skeleton may be in front of it in a sitting pose. Then the sitting skeleton could further suggest to place a mouse and a keyboard close to the hand and therefore at the edge of the desk. Although the human poses are not present during the arrangement, *hallucinating* them would help the robots to place objects in a human-friendly way.

We consider an arrangement as an outcome of the following generative process: A scene generates a set of possible human poses in the scene based on certain criteria (such as reachability or usage of existing objects); then use the human poses to determine where to place the new objects. There are two components required for this approach: (a) modeling how the objects relate to human poses based on criteria such as their affordances, ease of use and reachability, and (b) learning a distribution of human poses in the scene.

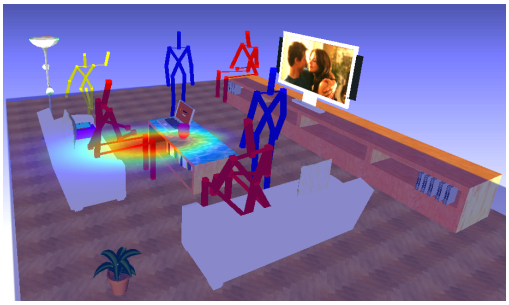


Fig. 2. While there could be innumerable possible human poses in a room, only a few of them are meaningful, such as the one on the couch who is related to many objects. We sample human poses based on the objects' affordances. For example, the learned potential function for the TV (shown in the projected heat map) has high values in the front with certain distance and thus the sitting human pose is sampled with high probability.

We first capture the object's usage preferences by a potential function that models how a human pose H_k is related to an object O_i. (Again, a higher value of this function indicates a better match between an object and the human pose.)

$$\Psi_{\text{human}}(O_i, H_k, \ell_i, c_i) = \Psi_{\text{loc}}(O_i, H_k, \ell_i)\, \Psi_{\text{ori}}(O_i, H_k, c_i). \tag{2}$$

Here, Ψ_{loc} and Ψ_{ori} represent the preferences in the relative location and the orientation of the object from a human pose respectively. For example, a TV has a usage score that is high in the front at a certain distance and falls off as you go to the side (see the projected heat map in Fig. 2). The potential function also indicates more meaningful and relevant human poses from the innumerable possible poses in an environment.

For instance, in a room such as the one shown in Fig. 2, the reaching pose (in yellow) and the sitting pose on the TV stand are less important because they do not relate to any object, while a sitting pose on the couch is important because it has high scores with several objects in the scene—the cushion, laptop, TV, etc.

Note that the human poses in our problem are latent, and therefore we model them using a mixture model. The model comprises an infinite number of human poses and each object selects a human pose according to a mixture proportion $\boldsymbol{\pi}$. As a result, an object is affected by multiple human poses. For example, in Fig. 2, the TV's location is determined by all the human poses (sitting on the couch or next to the TV, standing to the coffee table and so on). However, since the one on the couch is more important than others, its corresponding proportion defined in $\boldsymbol{\pi}$ will be higher and thus put more influence upon the TV. After considering all possible human poses (i.e., marginalizing out H_k and $\boldsymbol{\pi}$ in the mixture model), we define the likelihood of an arrangement, $\mathcal{O}$, of n objects (in human context) as[2]

$$p(\mathcal{O}) = \int_{\boldsymbol{\pi}} p(\boldsymbol{\pi}) \prod_{i=1}^{n} \sum_{k=1}^{\infty} \Big(p(O_i|H_k) P_0(H_k) \pi_k \Big) d\boldsymbol{\pi}, \tag{3}$$

[2]We abuse the notation O_i in this section to indicate the object's placement, including ℓ_i and c_i.

where P_0 is the prior of human poses, and $P(O_i|H_k) \propto \Psi_{\text{human}}(O_i, H_k)$. We adopt DP mixture model so that $\boldsymbol{\pi}$ can have unbounded length and be constructed using stick-breaking processes [39].

The inference problem is to find $\mathcal{O}$ with the maximum likelihood. Although (3) is intractable to compute, it can be approximated using a sampling scheme. We use Gibbs sampling with auxiliary parameters [40], where in each round we sample which human pose to select for each object, the object placements and the human poses according to their conditional distribution (see [13] for more details).

To differentiate the preference in selecting human poses for different types of objects, we add type-specific parameters Θ in the potential function and learn them from the labeled data. During training, given the objects in the scenes, we learn the parameters using the maximum likelihood estimation based on human poses sampled from the DP. In detail, we use human poses sampled from a DP, denoted by $H^1, \ldots, H^s$ as our observations. The optimal Θ is then computed by solving the following optimization problem:

$$\Theta^* = \arg\max_{\Theta} \sum_{\text{scenes}} \sum_{j=1}^{s} \sum_{i=1}^{n} \log \Psi_{\text{human}}(O_i, H_i^j; \Theta). \tag{4}$$

In our previous work [11, 12], we considered only the object context $\Psi_{\text{object}}(\cdot)$. In this current work, we primarily use the human context $\Psi_{\text{human}}(\cdot)$, and only combine it with some of the object context defined heuristically (see [13]). Jointly learning both the object and human context is an interesting direction for future work.

Once we have obtained the likelihood of the arrangements $p(\mathcal{O})$, we need to perform planning to realize the desired placements.

3.3 Planning

After finding the potential object placements that have high scores, the robot still faces two challenges in realizing the placing: First, high-scored placements may not be reachable/executable by the robot due to its kinematic constraints; Second, placing certain objects first may impede placing other objects later. Thus, the order in which the objects are placed becomes important and we need to find a valid placing order and target locations efficiently.

We address the first challenge by filtering out the placements that are not physically realizable by the robot due to its kinematic constraints (while considering placement of each object independently of the others).

For the second challenge, we adopt the classic backtracking search for finding a valid placing sequence. Particularly, in each search step, given the already-placed objects $\mathcal{P}$, we need to determine which object

Algorithm 1: TryPlace($\mathcal{P}$, ObjNotPlaced)

```
1 if ObjNotPlaced = ∅ then
2     Succeed
3 for i ∈ ObjNotPlaced do
4     for O_i ∈ PossiblePlacements_i do
5         if IsSuperSetOf(P ∪ O_i, F) then
6             continue
7         if feasible(O_i, P) then
8             TryPlace(P ∪ O_i, ObjNotPlaced \ O_i)
9 F ← F ∪ {P}
```

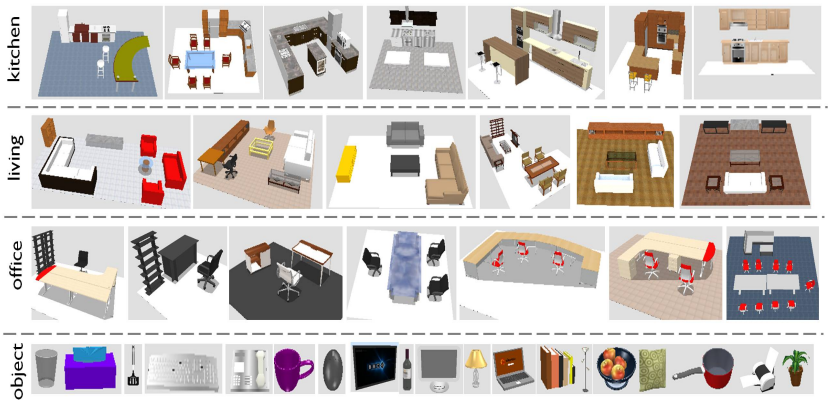


Fig. 3. Our dataset contains 20 scenes (7 kitchens, 6 living rooms and 7 offices) and 47 objects from 19 categories that are commonly seen in these scenes [13].

to be placed next (indexed by i) and also where to place it (denoted by O_i). While this search space is enormous, we can cut the redundancy using the following fact: *Any superset of an infeasible plan is also infeasible*. We maintain a set of all infeasible plans encountered so far, denoted by $\mathcal{F}$ (see Algorithm 1). Before trying to place a new object O_i, we check that if $\mathcal{P} \cup O_i$ becomes a superset of any elements in $\mathcal{F}$. Only if not, a path planning algorithm (in our case, rBiRRT in OpenRAVE [41]) is then used to verify the validity of placing at O_i and the search continues for other objects.

4 Experiments

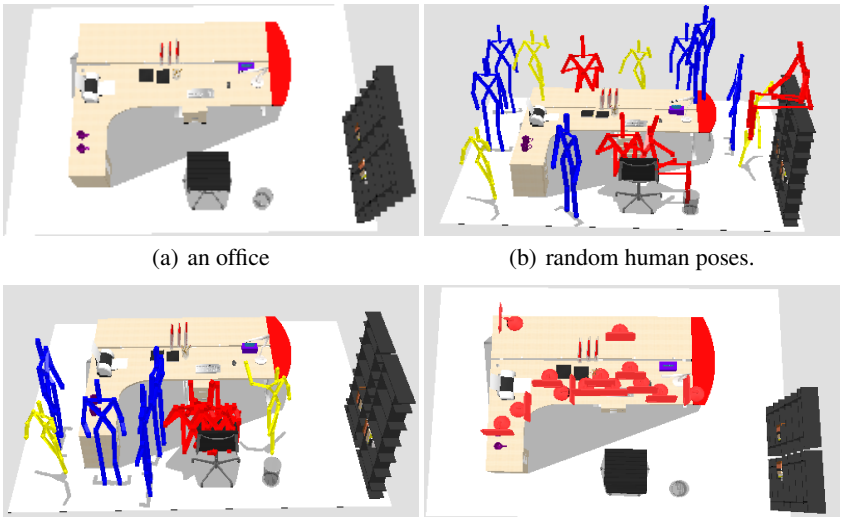
We perform three experiments as follows. First, we verify our human-context learning algorithm in arranging 20 different rooms, represented as 3D models. Second, we compare the object context and human context in different scenes in real point-clouds. Third, we perform robotic experiments on our Kodiak (PR2) robot based on the learned arrangements.

4.1 Arranging Rooms under Human Context

In order to verify that our DP-based learning algorithm can generate reasonable human poses as well as object placements, we evaluated it on a dataset containing 20 scenes from three categories (living room, kitchen and office) and of 47 daily objects from 19 types (listed in Fig. 6) such as dish-ware, fruit, computers, desk-lights, etc. [13]. Fig. 3 shows a snapshot of our dataset. Some example good arrangements of each room were labeled by three to five subjects (not associated with the project).

We conduct 5-fold cross validation on 20 rooms so that the test rooms have never been seen by the algorithm. We consider two placing scenarios: placing objects in filled rooms and empty rooms. In the first case, the task is to place one type of objects while other types are given (placed). In the second case, no object is in the test rooms at all.

Figure 4 shows an example of our algorithm inferring meaningful human poses and object placements. Given an office such as Fig. 4(a), if we randomly sample human



(a) an office (b) random human poses.

(c) sampled human poses based on existing objects. (d) sampled monitors, shown in red

Fig. 4. Sampling results of a test room. Given an office as (a), sampling human poses randomly results in several poses at absurd locations, such as on the top of the shelf in (b). Our algorithm, on the other hand, samples more relevant human poses in (c) and thus is able to sample more locations for placing a monitor in front of the chair than other places in (d).

poses regardless the existing objects, then many unreasonable human poses appear. For example, in Fig. 4(b), we have standing poses (in blue) oriented randomly and some sitting poses (in red) at absurd locations such as on top of the table and book shelf and reaching poses (in yellow) on the table as well. However, if we sample human poses based on the learned potential function (2), then we obtain human poses in meaningful places such as sitting in the chair or standing close to the object (see Fig. 4c). Note that now the distribution of both location and orientation of human poses has changed due to the Ψ_{loc} and Ψ_{ori} terms in the potential function.

We then sample the monitor's location according to these human poses. Figure 4d shows that the distribution is biased towards the inner side of the L-desk, especially concentrated in front of the chair. This is because that sitting poses are more related to monitors. Moreover, the preference of monitor placed on the table (as compared with being placed on the ground) is naturally learned through our human access effort rather than hand-script rules. Another interesting observation is that most samples are near the keyboard. This shows that the monitor-keyboard relationship can be linked through human poses naturally, without needing to explicitly model it.

Fig. 5 shows one sampled arrangements when placing in an empty room. Although the monitor and keyboard are not perfectly aligned, they are still placed roughly in front of the chair, with correct orientations. All the objects are placed in the correct placing areas, such as trashcan on the ground and the desk-light on the table. The trashcan being far from the chair is mainly due to some sampled human poses around that location.

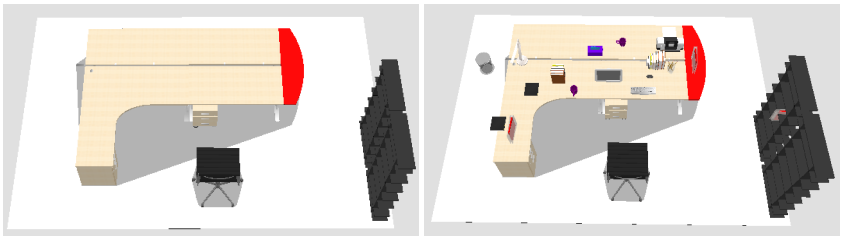


Fig. 5. Predicted arrangement for 17 objects in an empty rooms. Using our method, all the objects are correctly placed in their preferred areas, such as the trashcan on the ground and the books on the table. Some object-object relationships are also captured without modeling them explicitly, such as the monitor being placed close to the keyboard and their relative orientation to the chair.

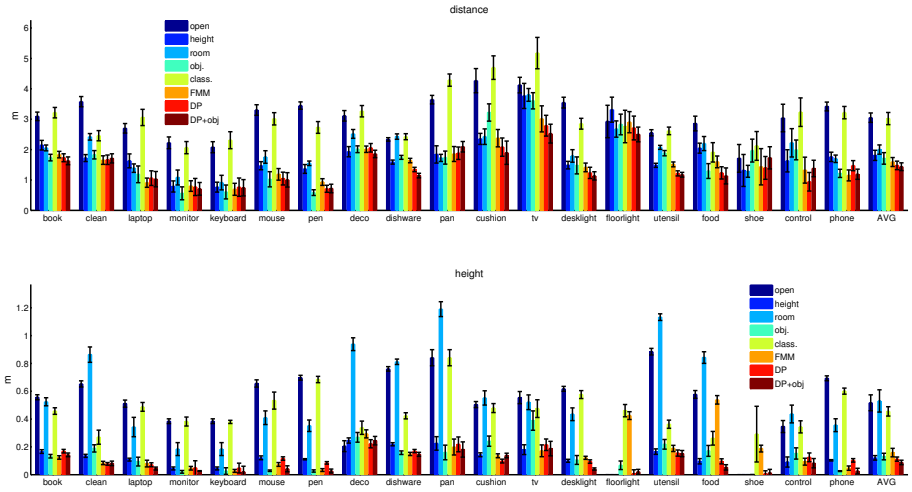
We now give the quantitative results on the whole dataset in Fig. 6. Arrangements are evaluated by two metrics: difference in *location* and *height* between the prediction and the ground-truth. We compare our method with six baselines [13], including using object context ('obj'). We additionally present another algorithm in which we combine the distribution of objects generated through human poses $O \propto \Psi_{\text{human}}(O, H; \Theta)$ with a distribution generated through object - object context $O \propto \Psi_{\text{obj}}(O, \mathcal{G})$ ($\mathcal{G}$ is the set of given objects) using a mixture model: $O \propto \omega\Psi_{\text{human}}(\cdot) + (1-\omega)\Psi_{\text{obj}}(\cdot)$. We give a comparison of methods of using object context only, human context only and their combination in our experiments.

In the task of arranging filled rooms (shown in Fig. 6a), using object context ('obj') benefited from the strong spatial relationships among objects and hence beat other baseline methods, especially for the laptop, monitor, keyboard and mouse types. However, our methods based on human context (last three bars) still outperformed the object context. They significantly improved placements of the objects that have weaker connection to others, such as book, TV, decoration and shoes.

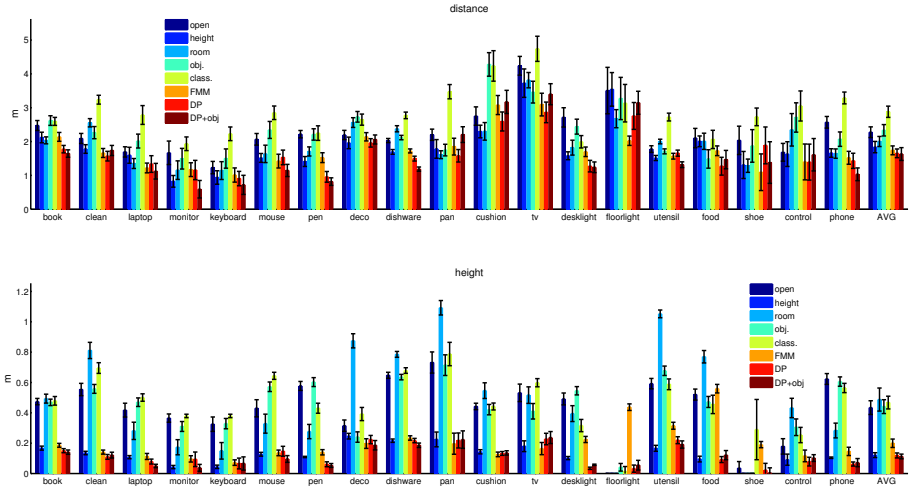
The task of arranging objects in an empty room (Fig. 6b) raises many challenges when placing the first few objects as no object context would be available. Not surprisingly, we found that the object-context method performed poorly, even worse than using just height as a reference ('height'). Although our methods also performed worse than the previous scenario, they could still sample human poses based on the furniture in the scene and thus predicted better locations for objects. Our experiments also showed that the finite mixture model using human context ('FMM') performed better than other baselines, but not as well as the ones our method using DPs.

In both tasks, our human-context algorithm successfully predicted object placements within 1.6 meters on average. The average error in height was only 0.1 meters. By combining human- and object-context, the error was further reduced—indicating that they provide some complementary context.

Robotic simulation experiment. In order to study how the desired placements are affected by the robot constraints (see Section 3.3), we tested arranging these synthetic scenes using Kodiak (PR2) in simulation. Table 1 shows that the location errors increase only slightly for arranging filled rooms as well as empty rooms, but the errors in height increase significantly. This is mostly because of the kinematic constraints of the robot.



(a) arranging filled rooms



(b) arranging empty rooms

Fig. 6. Results of arranging filled rooms (top) and empty rooms (bottom), evaluated by the difference in location and height in meters. The error bar shows one standard error.

How to incorporate robotic constraints into our current score function is an interesting direction for future work.

4.2 Arranging Real Scenes

In this experiment, we compare the two algorithms—using object context [12] and using human context [13]—in arranging real scenes. The dataset [12] contains 3 different offices and 2 different apartments where the placing areas such as tables, cabinets, floor, and drawers were segmented out. We evaluated the quality of the final placing layout by asking two human subjects (one male and one female, not associated with the project) to

Table 1. Comparison between the predicted placements with and without the robotic constraints (verified in simulation). Unit is meters. Note that only those objects that are physically movable by the robot are considered.

		book	clean tool	mouse	pen	dish-ware	pan	cush-ion	uten-sil	food	shoe	re-mote	AVG
arranging filled rooms													
location	without constraints	1.63	1.71	1.00	0.72	1.15	2.09	1.90	1.17	1.13	1.73	1.38	1.42
	with constraints	1.87	2.26	0.96	0.69	1.63	2.45	2.38	1.03	1.34	2.01	1.24	1.62
height	without constraints	0.14	0.08	0.04	0.03	0.15	0.18	0.14	0.15	0.05	0.01	0.08	0.10
	with constraints	0.32	0.52	0.14	0.17	0.27	0.42	0.33	0.31	0.17	0.18	0.22	0.28
arranging empty rooms													
location	without constraints	1.65	1.74	1.15	0.82	1.19	2.21	3.17	1.32	1.47	1.38	1.61	1.61
	with constraints	1.97	2.31	1.51	1.22	1.89	2.34	3.01	1.89	1.55	1.55	1.72	1.91
height	without constraints	0.14	0.12	0.10	0.05	0.19	0.22	0.14	0.19	0.12	0.00	0.10	0.13
	with constraints	0.36	0.59	0.18	0.21	0.39	0.44	0.35	0.33	0.47	0.19	0.26	0.34

Table 2. Results on arranging five real point-cloud scenes (3 offices & 2 apartments). The number of objects for placing are 4, 18, 18, 21 and 18 in each scene respectively. **Co**: % of semantically correct placements, **Sc**: average score (0-5).

	office1		office2		office3		apt1		apt2		Average	
	Co	Sc	Co	Sc	Co	Sc	Co	Sc	Co	Sc	Co	Sc
obj context [12]	100	4.5	100	4.2	87	3.5	65	3.2	75	3.0	85	3.7
Human context (FMM)	100	3.5	100	2.0	83	3.8	63	3.5	63	3.0	82	3.2
Human context (DP)	100	**5.0**	100	4.3	91	4.0	74	3.5	**88**	**4.3**	90	4.2
Human (DP) + obj context	100	4.8	100	**4.5**	**92**	**4.5**	**89**	**4.1**	81	3.5	**92**	**4.3**

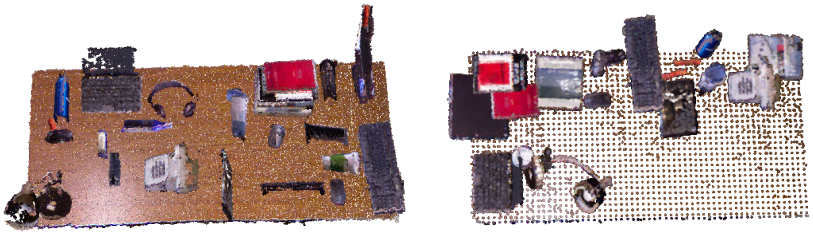


Fig. 7. Results of placing multiple objects on an office desk, when considering object context [12] (left) and considering human context (right). While objects are scattered in the left arrangement, the right arrangement prefers placing objects at the edge of the desk for easy access.

label placement for each object semantically correct or not, and also report a qualitative metric score on how good the overall placing was (0 to 5 scale).

Results are shown in Table 2. Both methods arranged objects more meaningfully than heuristic rules (not reported here, see [13]), i.e., books were stacked together, while a keyboard, laptop and mouse were placed close to each other. The human context, however, performed much better by, for example, placing shoes at the bottom level of a shelf, while food and books are on the middle level or on a table. The approach of using object context only sometimes put the laptop on a shelf making it difficult for human to access.

Fig. 7 shows a comparison in arranging office2. Compared to using object context, human context links the mouses and keyboard together as well as lamp and the laptop. The laptop is now at the edge of the table and thus becomes accessible for humans.

Fig. 8. Our Kodiak robot arranging several objects in three different scenarios.

Other objects are all close to the edge, unlike objects scattered uniformly in the left figure making the bottle and mouse in the center hard to reach.

4.3 Robotic Experiments

We verified our approach on our Kodiak (PR2) robot in three scenarios: 1) placing five objects (a beer bottle, cup, soda can, hand torch and shoe) in a kitchen with a fridge and a table; 2) placing six objects (a mouse, a pen, a trashbin and three books) in an office with a table and a bookshelf; 3) placing five objecs (a cup, tissue box, book, soda can and throw pillow) in a living room with a couch and a coffee table.

Given the predicted arrangements, the robot uses a *pre-determined grasp* to pick up every object, and executes the plan (see Section 3.3) for moving the object to its designated location. Fig. 8 shows some screenshots of our robot performing the object arrangements. We found that all the objects were placed at the locations consistent with the simulation experiments. For the videos, see http://pr.cs.cornell.edu/placingobjects/.

There were certain failures however caused by the limitation of our learning algorithm. For example, the beer bottle was placed on the couch instead of the table. This was because the physical properties of the surfaces (e.g., hard vs soft) are not explicitly modeled. This may potentially be avoided by including semantic information or appearance features of the furniture.

5 Discussion and Conclusions

We considered arranging multiple objects in complex placing areas, while following human usage preferences. Motivated by the fact that objects are often arranged for certain human activities, we developed an approach based on sampling meaningful latent human poses and using them to determine objects' placements. In detail, we designed

a potential function for capturing the human-object relationship and used Dirichlet processes to sample human poses and placements jointly. We verified our approach on a variety of scenes in simulation as well as on a real robot.

In this work, we have focussed on learning the object arrangements from a human usage perspective. We believe that integrating the object detection, grasping and placing jointly is a challenging direction for future work. Furthermore, one can also potentially incorporate control and planning into our model in order to obtain placements that are easily executed by the robot.

References

1. M. R. Cutkosky, *Robotic Grasping and Fine Manipulation*. Norwell, MA, USA: Kluwer Academic Publishers, 1985.
2. M. Salganicoff, L. H. Ungar, and R. Bajcsy, "Active learning for vision-based robot grasping," *Machine Learning*, vol. 23, pp. 251–278, 1996.
3. A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *ICRA*, pp. 348–353, 2000.
4. K. Hsiao, P. Nangeroni, M. Huber, A. Saxena, and A. Y. Ng, "Reactive grasping using optical proximity sensors," in *ICRA*, 2009.
5. A. Rodriguez, M. Mason, and S. Ferry, "From caging to grasping," in *Proceedings of Robotics: Science and Systems*, June 2011.
6. A. Saxena, J. Driemeyer, J. Kearns, and A. Ng, "Robotic grasping of novel objects," in *Neural Information Processing Systems*, 2006.
7. A. Saxena, J. Driemeyer, and A. Ng, "Robotic grasping of novel objects using vision," *The International Journal of Robotics Research*, vol. 27, no. 2, p. 157, 2008.
8. A. Saxena, L. Wong, and A. Y. Ng, "Learning grasp strategies with partial shape information," in *AAAI*, 2008.
9. Q. Le, D. Kamm, A. Kara, and A. Ng, "Learning to grasp objects with multiple contact points," in *ICRA*, 2010.
10. Y. Jiang, S. Moseson, and A. Saxena, "Efficient grasping from rgbd images: Learning using a new rectangle representation," in *ICRA*, 2011.
11. Y. Jiang, C. Zheng, M. Lim, and A. Saxena, "Learning to place new objects," in *ICRA*, 2012.
12. Y. Jiang, M. Lim, C. Zheng, and A. Saxena, "Learning to place new objects in a scene," *The International Journal of Robotics Research (IJRR)*, 2012.
13. Y. Jiang, M. Lim, and A. Saxena, "Learning object arrangements in 3d scenes using human context," in *ICML*, 2012.
14. A. Edsinger and C. Kemp, "Manipulation in human environments," in *Int'l Conf Humanoid Robots*, 2006.
15. M. Schuster, J. Okerman, H. Nguyen, J. Rehg, and C. Kemp, "Perceiving clutter and surfaces for object placement in indoor environments," in *Int' Conf Humanoid Robots*, 2010.
16. T. Lozano-Pérez, J. Jones, E. Mazer, and P. O'Donnell, "Task-level planning of pick-and-place robot motions," *Computer*, vol. 22, no. 3, pp. 21–29, 2002.
17. H. Sugie, Y. Inagaki, S. Ono, H. Aisu, and T. Unemi, "Placing objects with multiple mobile robots-mutual help using intention inference," in *ICRA*, 1995.
18. D. Jain, L. Mosenlechner, and M. Beetz, "Equipping robot control programs with first-order probabilistic reasoning capabilities," in *ICRA*, 2009.
19. L. Mösenlechner and M. Beetz, "Parameterizing Actions to have the Appropriate Effects," in *IROS*, 2011.

20. E. Aker, A. Erdogan, E. Erdem, and V. Patoglu, "Housekeeping with multiple autonomous robots: Knowledge representation and automated reasoning for a tightly integrated robot control architecture," in *IROS*, 2011.
21. A. Torralba, K. Murphy, and W. T. Freeman, "Using the forest to see the trees: object recognition in context," *Communications of the ACM, Research Highlights*, vol. 53, no. 3, pp. 107–114, 2010.
22. A. Saxena, S. Chung, and A. Ng, "3-d depth reconstruction from a single still image," *International Journal of Computer Vision*, vol. 76, no. 1, pp. 53–69, 2008.
23. A. Saxena, M. Sun, and A. Ng, "Make3d: Learning 3d scene structure from a single still image," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 5, pp. 824–840, 2009.
24. G. Heitz, S. Gould, A. Saxena, and D. Koller, "Cascaded classification models: Combining models for holistic scene understanding," in *Neural Information Processing Systems*, 2008.
25. C. Li, A. Kowdle, A. Saxena, and T. Chen, "Towards holistic scene understanding: Feedback enabled cascaded classification models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 7, pp. 1394–1408, 2012.
26. V. Hedau, D. Hoiem, and D. Forsyth, "Recovering the spatial layout of cluttered rooms," in *ICCV*, 2009.
27. X. Xiong and D. Huber, "Using context to create semantic 3d models of indoor environments," in *BMVC*, 2010.
28. H. Koppula, A. Anand, T. Joachims, and A. Saxena, "Semantic labeling of 3d point clouds for indoor scenes," in *NIPS*, 2011.
29. A. Anand, H. Koppula, T. Joachims, and A. Saxena, "Contextually guided semantic labeling and search for 3d point clouds," *IJRR*, 2012.
30. M. Fisher and P. Hanrahan, "Context-based search for 3d models," *ACM TOG*, vol. 29, no. 6, 2010.
31. M. Fisher, M. Savva, and P. Hanrahan, "Characterizing structural relationships in scenes using graph kernels," *SIGGRAPH*, 2011.
32. B. Nabbe, S. Kumar, and M. Hebert, "Path planning with hallucinated worlds," in *IROS*, 2004.
33. J. Kuffner Jr and S. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2, pp. 995–1001, IEEE, 2000.
34. C. Stoll, J. Gall, E. de Aguiar, S. Thrun, and C. Theobalt, "Video-based reconstruction of animatable human characters," *ACM Transactions on Graphics (Proc. SIGGRAPH ASIA)*, 2010.
35. D. Shin, I. Sardellitti, Y.-L. Park, O. Khatib, and M. Cutkosky, "Design and control of a bio-inspired human-friendly robot," *The International Journal of Robotics Research*, vol. 29, pp. 571–584, April 2010.
36. J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from rgbd images," in *International Conference on Robotics and Automation (ICRA)*, 2012.
37. D. Ly, A. Saxena, and H. Lipson, "Co-evolutionary predictors for kinematic pose inference from rgbd images," in *Genetic and Evolutionary Computation Conference (GECCO)*, 2012.
38. E. Demircan, T. F. Besier, and O. Khatib, "Muscle force transmission to operational space accelerations during elite golf swings," in *Proc. of the IEEE International Conference on Robotics and Automation*, (St Paul, MN, USA), pp. 1464–1469, May 2012.
39. Y. W. Teh, "Dirichlet process," *Encyclopedia of Machine Learning*, pp. 280–287, 2010.
40. R. Neal, "Markov chain sampling methods for dirichlet process mixture models," *Journal of computational and graphical statistics*, pp. 249–265, 2000.
41. R. Diankov and J. Kuffner, "Openrave: A planning architecture for autonomous robotics," *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-34*, 2008.