X-Ray: Mechanical Search for an Occluded Object by Minimizing Support of Learned Occupancy Distributions

Michael Danielczuk\textsuperscript{1}, Anelia Angelova\textsuperscript{2}, Vincent Vanhoucke\textsuperscript{2}, Ken Goldberg\textsuperscript{1}

Abstract—For applications in e-commerce, warehouses, healthcare, and home service, robots are often required to search through heaps of objects to grasp a specific target object. For mechanical search, we introduce X-Ray, an algorithm based on learned occupancy distributions. We train a neural network using a synthetic dataset of RGBD heap images labeled for a set of standard bounding box targets with varying aspect ratios. X-Ray minimizes support of the learned distribution as part of a mechanical search policy in both simulated and real environments. We benchmark these policies against two baseline policies on 1,000 heaps of 15 objects in simulation where the target object is partially or fully occluded. Results suggest that X-Ray is significantly more efficient, as it succeeds in extracting the target object 82\% of the time, 15\% more often than the best-performing baseline. Experiments on an ABB YuMi robot with 20 heaps of 25 household objects suggest that the learned policy transfers easily to a physical system, where it outperforms baseline policies by 15\% in success rate with 17\% fewer actions. Datasets, videos, and experiments are available at https://sites.google.com/berkeley.edu/x-ray.

I. INTRODUCTION

Mechanical search – extracting a desired object from a heap of objects – is a fundamental task for robots in unstructured e-commerce warehouse environments or for robots in home settings. It remains challenging due to uncertainty in perception and actuation as well as lack of models for occluded objects in the heap.

Data-driven methods are promising for grasping unknown objects in clutter and bin picking \cite{7,10,23,26,27}, and can reliably plan grasps on the most accessible object without semantic knowledge of the target object. Some reinforcement learning \cite{9,39} or hierarchical \cite{5} mechanical search policies use semantics, but have so far been limited to specific objects or heuristic policies.

In this paper, we draw on recent work on shape completion to reason about occluded objects \cite{29,33} and work on predicting multiple pose hypotheses \cite{24,32}. X-Ray combines occlusion inference and hypothesis predictions to estimate an occupancy distribution for the bounding box most similar to the target object to estimate likely poses – translations and rotations in the image plane. X-Ray can efficiently extract the target object from a heap where it is fully occluded or partially occluded (Figure 1).

This paper provides four contributions:

1) X-Ray (maXimize Reduction in support Area of occupancy distribution): a mechanical search policy that minimizes support of learned occupancy distributions.

2) An algorithm for estimating target object occupancy distributions using a set of neural networks trained on a dataset of synthetic images that transfers seamlessly to real images.

3) A synthetic dataset generation method and 100,000 RGBD images of heaps labeled with occupancy distributions for a single partially or fully occluded target object, constructed for transfer to real images.

4) Experiments comparing the mechanical search policy against two baselines in 1,000 simulated and 20 physical heaps that suggest the policy can reduce the median number of actions needed to extract the target object by 20\% with a simulated success rate of 87\% and physical success rate of 100\%.

II. RELATED WORK

A. Pose Hypothesis Prediction

There is a substantial amount of related work in computer vision on 3D and 6D pose prediction of both known and unknown objects in RGB, depth, and RGBD images \cite{8,13,18,56}. Many of these papers assume that the target
objects are either fully visible or have minor occlusions. In addition, many assume that there is no ambiguity in object pose due to self-occlusion or rotational symmetry of the object, as these factors can significantly decrease performance for neural network-based approaches [3]. Recent work has attempted to address the pose ambiguity that results from object geometry or occlusions by restricting the range of rotations [31] predicting multiple hypotheses for each detected object [24, 32]. Rupprecht et al. [32] find that refining multiple pose hypotheses to a 6D prediction outperforms single hypothesis predictions on a variety of vision tasks, such as human pose estimation, object classification, and frame prediction. Manhardt et al. [24] note that directly regressing to a rotation for objects with rotational symmetries can result in an averaging effect where the predicted pose does not match any of the possible poses; thus, they predict multiple pose hypotheses for objects with pose ambiguities to better predict the underlying pose and show Bingham distributions of the predicted hypotheses. However, only minor occlusions are considered and since ground truth pose distributions are not available for these images and objects, comparisons for continuous distributions can only be made qualitatively. Predicting multiple hypotheses or a distribution to model ambiguity has also been applied to gaze prediction from facial images [30], segmentation [14], and monocular depth prediction [38]. In contrast to these works, we learn occupancy distributions in a supervised manner.

B. Object Search

There has been a diverse set of approaches to grasping in cluttered environments, including methods that use geometric knowledge of the objects in the environment to perform wrench-based grasp metric calculations, nearest-neighbor lookup in a precomputed database, or template matching [1, 20, 25], as well as methods using only raw sensor data [12, 34], commonly leveraging convolutional neural networks [9, 10, 17]. While multi-step bin-picking techniques have been studied, they do not take a specific target object into account [22].

Kostrikov et al. [15] learn a critic-only reinforcement learning policy to push blocks in a simulated environment to uncover an occluded MNIST block. Zeng et al. [40] train joint deep fully-convolutional neural networks to predict both pushing and grasping affordances from heightmaps of a scene containing multicolored blocks, then show that the resulting policy (VPG) can separate and grasp novel objects. Their policy learns to push through heaps of objects to find the target and then coordinate grasping and pushing actions to extract it, outperforming a target-centered VPG baseline in success rate and number of actions. Both approaches can generalize to objects outside the training distribution, although they are evaluated on a limited set of novel objects, and Yang et al. separate the cases where the target object is partially occluded and fully occluded.

Recently, several approaches to the mechanical search problem have been proposed, both in tabletop and bin picking environments. Price et al. [29] propose a shape completion approach that predicts occlusion regions for objects to guide exploration in a tabletop scene, while Xiao et al. [37] implement a particle filter approach and POMDP solver to attempt to track all visible and occluded objects in the scene. However, 75% of the objects in Price et al.’s evaluation scenes are seen in training and Xiao et al.’s method requires models of each of the objects in the scene. We benchmark our policy on a variety of non-rigid, non-convex household objects not seen in training and require no object models.

In previous work, Daniclzuk et al. [5] proposed a general mechanical search problem formulation and introduced a two-stage perception and search policy pipeline. In contrast, we introduce a novel perception network and policy based on minimizing support of occupancy distributions that outperforms the methods introduced in [5].

III. Problem Statement

We consider an instance of the mechanical search problem where a robot must extract a known target object from a heap of unknown objects by iteratively grasping to remove non-target objects. The objective is to extract the target object using the fewest number of grasps.

A. Assumptions

- One known target object, fully or partially occluded by unknown objects in a heap on a planar workspace.
- A robot with a gripper, an overhead RGBD sensor with known camera intrinsics and pose relative to the robot.
- A maximum of one object is grasped per timestep.
- A target object detector that can return a binary mask of visible target object pixels when queried.

B. Definitions

We define the problem as a partially-observable Markov decision process (POMDP) with the 7-tuple \((S, A, T, R, \Omega, O, \gamma)\) and a maximum horizon \(H\):

- **States** \((S)\): A state \(s_k\) at timestep \(k\) consists of the robot, a static overhead RGBD camera, and a static bin containing \(N+1\) objects, target object \(O_t\) and distractor objects \(\{O_{1,k}, O_{2,k}, \ldots, O_{N,k}\}\). No prior information is known about the \(N\) distractor objects.
- **Actions** \((A)\): A grasp action \(a_k\) at timestep \(k\) executed by the robot’s gripper.
- **Transitions** \((T)\): In simulation, the transition model \(T(s_{k+1} \mid a_k, s_k)\) is equivalent to that used by Mahler et al. [22] and uses pybullet [4] for dynamics. On the physical system, next states are determined by executing the action on a physical robot and waiting until objects come to rest.
C. Objective

Given this problem definition and assumptions, the objective is to maximize the expected discounted sum of rewards:

$$\pi^*_\theta = \arg \max_\theta \mathbb{E}_{p(\tau | \theta)} \left[ \sum_{k=0}^{H-1} \gamma^k R(s_k, \pi_\theta(y_k), s_{k+1}) \right]$$

where $p(\tau | \theta) = \mathbb{P}(s_0) \prod_{k=0}^{H-1} T(s_{k+1} | s_k, \pi_\theta(y_k), s_k) O(y_k | s_k)$ is the distribution of state trajectories $\tau$ induced by a policy $\pi_\theta$ [22]. $\pi_\theta^*$ is the policy parametrized by $\theta$ that maximizes the objective. Maximizing this objective corresponds to removing the target object in the fewest number of actions.

IV. LEARNING OCCUPANCY DISTRIBUTIONS

We describe a method for learning occupancy distributions from RGBD images; specifically, estimating a function $f_\rho : \Omega \rightarrow \mathcal{P}$ that takes an observation $y_k$ and produces the corresponding occupancy distribution $\rho_k$ for a given bounding box. The output of this system can be described as a one-channel image of the same height and width as the input image, where each pixel value represents the likelihood of that pixel containing part of the target object’s amodal segmentation mask, or the likelihood that some part of the object, in some planar translation or rotation, would occupy that pixel without any occlusions from other objects. To predict this pixelwise distribution of the target object, we use a neural network trained on a dataset of augmented depth images and ground-truth occupancy distributions.

A. Dataset Generation

We generate a dataset of 10,000 synthetic augmented depth images labeled with target object occupancy distributions for a rectangular box target object. We choose 10 box targets of various dimensions ranging from $3cm \times 3cm \times 5mm$ to $9.5cm \times 0.95cm \times 5mm$ (aspect ratios varying from 1:1 to 10:1) with equal volume and generate a dataset for each, resulting in a total of 100,000 dataset images. We choose a relatively small thickness for the object so that it is more likely to be occluded in heaps of objects, as it tends to lie flat on the workspace. We sample a state $s_0$ by uniformly sampling a set of $N$ 3D CAD models as well as a heap center and 2D offsets for each object from a 2D trimmed gaussian. First, $O_i$ is dropped from a fixed height above the workspace, then the other $N$ objects are dropped one by one from a fixed height and dynamic simulation is run until all objects come to rest (all velocities are zero). Any objects that fall outside of the workspace are removed. $N$ is drawn from a Poisson distribution ($\lambda = 12$) truncated such that $N \in \{0, 1, \ldots, 12\}$. Each dataset image is generated by summing all shifted amodal target object masks whose modal masks correspond with the target object modal mask.

Rewards ($R$): The reward $r_k = R(s_k, a_k, s_{k+1}) \in \{0, 1\}$ is 1 if the target object is successfully grasped and lifted from the bin, otherwise the reward is 0.

Observations ($\Omega$): An observation $y_k \in \mathbb{R}_+^{w \times h}$ at timestep $k$ consists of an RGBD image taken by the overhead RGBD camera with image width $w$ and height $h$.

Observation Model ($O$): A deterministic observation model $O(y_k | s_k)$ is defined by the RGBD input and known camera intrinsics and extrinsics.

Discount Factor ($\gamma$): To encourage efficient extraction of the target object, $0 < \gamma < 1$.

We also define the following terms:

- Modal Segmentation Mask ($M_{m,i}$): the region(s) of pixels in an image corresponding to object $O_i$ which are visible \[\square\].

- Amodal Segmentation Mask ($M_{a,i}$): the region(s) of pixels in an image corresponding to object $O_i$ which are visible or invisible (occluded by other objects in the image) \[\square\].

- The oriented minimum bounding box is the box with the minimum volume that encloses the object, subject to no constraints as to the orientation of the result. We use this box to determine scale and aspect ratio for a target object.

- The occupancy distribution $\rho \in \mathcal{P}$ is the unnormalized distribution describing the likelihood that a given pixel in the observation image contains some part of the target object’s amodal segmentation mask.

- The occupancy distribution $\rho$ is dropped from a fixed height above the heap, followed by the other objects (colored gray), into a planar workspace using dynamic simulation. Camera intrinsics and pose are sampled from uniform distributions centered around their nominal values and an RGBD image is rendered of the scene. The augmented depth image (top right), consisting of a binary target object modal mask and a two-channel depth image, is the only input used for training for seamless transfer from simulation to real images. The ground truth target object distribution is generated by summing all shifted amodal target object masks whose modal masks correspond with the target object modal mask.
By comparing the set of these shifted and rotated depth images to the depth image containing all of the objects except the bottom of the workspace at that point, we can determine the modal segmentation mask of the target object translations or rotations such that the modal segmentation mask of the target object is equivalent in the corresponding image. In other words, we want to find the set of planar translations and rotations into 16 bins, then shifting and rotating a target-object image and the distribution will spread to reflect where the object is partially or fully occluded, then multiple target objects at different scales and in the presence of new occluding objects. However, a network trained on a small aspect ratio does not generalize well to higher aspect ratio objects, as it tends to overestimate the occupancy distribution.

| Aspect Ratio | Test Lid Bal. Acc. | IoU | Domino Bal. Acc. | IoU | Flute Bal. Acc. | IoU |
|--------------|-------------------|-----|------------------|-----|----------------|-----|
| 1:1          | 98%               | 0.91| 93%              | 0.10| 72%            | 0.74|
| 2:1          | 97%               | 0.90| 79%              | 0.44| 64%            | 0.67|
| 5:1          | 97%               | 0.90| 66%              | 0.23| 65%            | 0.83|
| 10:1         | 97%               | 0.87| 84%              | 0.49| 82%            | 0.58|

TABLE I: Balanced accuracy (Bal. Acc.) and Intersection over Union (IoU) metrics for networks trained on various aspect ratio target boxes. The first column is the respective set of 2,000 test images for the network’s training dataset. The other columns show how the networks can generalize to unseen objects outside the training distribution. Each dataset contains 1,000 test images for the lid, domino, and flute objects, respectively. These objects are shown in Figure [2] and have approximate aspect ratios of 1:1, 2:1, and 5:1, respectively. Each network performs very well when estimating distributions for its training target object and makes reasonable predictions for target objects with similar bounding box aspect ratios, even for novel target objects at different scales and in the presence of new occluding objects. However, a network trained on a small aspect ratio does not generalize well to higher aspect ratio objects, as it tends to overestimate the occupancy distribution.
Target Object Scale. For objects of different scale than the training target object, we scale the input image by a factor equal to the difference in scale between the box target object and the other target object, feed it through the network, and then rescale the output distribution. We find that this scaling dramatically improves performance with minimal preprocessing of the input image; for example, when testing on the lid object, which is about twice as large as the training box object, we increase balanced accuracy and IoU from 63.0% and 0.186 to 93.1% and 0.697, respectively.

Target Aspect Ratios. We found that, while our network performed well on objects with similar aspect ratios, longer and thinner objects with higher aspect ratios resulted in the model overestimating the support of the distribution. This effect can be seen in Figure 3 which shows ground truth occupancy distributions for target objects of different aspect ratios in the same heap image. Table I suggests that the trained networks can accurately predict occupancy distributions for target objects that have similar aspect ratios to the training boxes, but do not perform as well when tasked with predicting a distribution for objects with dramatically different aspect ratios. In particular, the network trained with a 1:1 box target object tends to overestimate the support for target objects with high aspect ratios, leading to a drop in metrics. This effect is especially visible along corners of occluding objects, where more rotations of a low aspect ratio object are possible, while only one or two rotations of a high aspect ratio object are possible.

Figure 4 shows occupancy distribution predictions with ground truth distributions for the three unseen objects using the network trained on the closest aspect ratio target object and scaled appropriately. Results suggest that the network is able to accurately predict diverse distributions when occluding objects not seen in training are present. Figure 4 suggests not only that the network can predict the correct distribution spanning multiple occluding objects in unimodal and multimodal cases when the target object is fully occluded, but also that it can correctly collapse the distribution to a small area around the visible part of the target object when it is only partially occluded.

V. X-Ray: Mechanical Search Policy

Using the learned occupancy distribution predictions, we propose X-Ray, a mechanical search policy. The heap RGBD image and target object are inputs to the perception system, which uses the network trained on the most similar bounding box to the target object to predict an occupancy distribution for the target. The policy takes the predicted distribution and a set of modal segmentation masks for the scene and computes a grasping action that would maximally reduce the support of the subsequent distribution. Specifically, the policy takes an element-wise product of each segmentation mask with the predicted occupancy distribution and sums over all entries in the resulting image, leading to a score for each of the segmentation masks. The policy then plans a grasp on the object mask with the highest score and executes it, as shown in Figure 5.

A. Simulation Experiments with X-Ray

We evaluate the mechanical search policy first with simulated heaps of novel objects. To further test the ability of the learned network to generalize to unseen occluding objects,
we use a set of objects unseen in training and validation: 46 YCB objects [2] and 13 “packaged” YCB objects (augmented in the same way as described in Section IV). Initial states were generated as explained in Section IV first dropping the target object, followed by the other N objects. We use N = 14 so each heap initially contained 15 total objects. As the focus of this work was not instance segmentation or target detection, we use ground truth segmentation masks and target binary masks in simulation, although we note that any class-agnostic instance segmentation network [6, 16] or object detection network [41] can be substituted. For each grasp, either a parallel jaw or suction cup grasp, we use wrench space analysis to determine whether it would result in the object being lifted from the workspace under quasi-static conditions [20, 21, 28]. If the grasp is collision-free and the object can be lifted, the object is lifted until the remaining objects come to rest using dynamic simulation implemented in pybullet, resulting in the next state. Otherwise the state remains unchanged.

In addition to the policy proposed here, we evaluate two previously proposed baseline policies [5]:

- **Random**: A policy that first attempts to grasp the target object, and, if no grasps are available on the target object, grasps an object chosen uniformly at random from the bin.
- **Largest**: A policy that first attempts to grasp the target object, and, if no grasps are available on the target object, iteratively attempts to grasp the objects in the bin according to the size of their modal segmentation mask.

Each policy was rolled out on 1,000 total heaps until either the target object was grasped (successful rollout) or the horizon $H = 10$ was reached (failed rollout). We benchmark each policy using two metrics: success rate of the policy and mean number of actions taken to extract the target object in successful rollouts. Table II and Figure 6 show these metrics and the distribution of successful rollouts over the number of actions taken to extract the target object, respectively.

While the Random and Largest policies occasionally are able to quickly extract the target object, X-Ray consistently extracts the target in fewer actions and succeeds in 15% more heaps than the best-performing baseline. Largest is a reasonable heuristic for these heaps, as shown in [5], as large objects typically have a greater chance of occluding the target, but X-Ray combines this intuition with superior performance when the object is partially occluded. X-Ray outperforms the Largest policy on heaps where the target

---

**Table II: Evaluation metrics for each policy over 1,000 simulated rollouts.**

| Policy  | Success Rate | Number of Actions Quartiles |
|---------|--------------|-----------------------------|
| Random  | 42%          | 4, 7, 9                     |
| Largest | 67%          | 4, 5, 7                     |
| X-Ray   | 82%          | 3, 5, 6                     |

**Fig. 6: Histogram of the number of actions taken to extract the target object over the 1,000 simulated rollouts for the three policies tested.** The median number of actions for each policy is shown by the corresponding vertical line.
object is partially occluded by a thin or small object (such as a fork or dice) at some point during the rollout. In these scenarios, a robust grasp is often not available on the target object, and while X-Ray can correctly identify that the occluding object should be removed, the Largest policy will often grasp a larger object further from the target object. In scenarios where there are many large objects, but some are lying to the side, X-Ray will typically grasp objects that are in the more cluttered area of the bin, since they are more likely to reveal the target object. This behavior is a function of weighting the object area by the predicted distribution, which encourages the policy to ignore solitary objects.

B. Physical Experiments with X-Ray

We also evaluate X-Ray with heaps of novel household objects on a physical ABB YuMi robot with a suction cup and parallel jaw gripper, using two target objects. Some examples of the objects used can be seen in Figures 1 and 5. Initial states were generated by placing the target object on the workspace, filling a bin with the N other objects, and then dumping the bin on top of the target object. In these heaps, N = 24 was used so that each heap initially contained 25 total objects. We chose 25 total objects because objects tend to disperse further on the physical setup, so more objects are needed to completely obscure the target object. For segmentation masks, we used the class-agnostic instance segmentation network from [6], and for grasp quality analysis, we used FC-GQCNN [33]. To generate binary target masks, we use HSV color segmentation from OpenCV and use red target objects. While we make this assumption for simplicity, we note that we could substitute this process with a target object segmentation method that uses visual features, semantics and shape, such as the one described in [6].

We perform 20 rollouts for each of the three policies. Each policy was rolled out until either the target object was grasped (successful rollout) or the horizon H = 10 was reached (failed rollout). We report the same metrics as in the simulated experiments in Table III.

We find that X-Ray outperforms both baselines, extracting the target object in a median 5 actions over the 20 rollouts as compared to 6 actions for the Largest and Random policies while succeeding in extracting the target object within 10 actions in each case. These results suggest that X-Ray not only can extract the target more efficiently than the baseline policies, but also has lower variance. The Largest policy performed comparatively worse with more objects in the heap than in simulation, as it relies heavily on accurate segmentation masks. However, when objects are densely clustered together, segmentation masks are often merged, leading to grasps on smaller objects that do not uncover the target. X-Ray reduces this reliance on accurate segmentation masks, as the occupancy distribution and segmentation are combined to create a score for the mask. X-Ray can also help to discard spurious segmentation masks if they overlap with regions that do not contain any objects. X-Ray also reduces reliance on the target object binary mask being accurate; if the detector cannot see enough of the target object to generate a detection even when it is partially visible, X-Ray will continue to try and uncover it according to the fully occluded occupancy distribution until more of the target is revealed.

VI. DISCUSSION AND FUTURE WORK

We present X-Ray, a mechanical search algorithm that minimizes support of a learned occupancy distribution. We showed that a model trained only on a synthetic dataset of augmented depth images labeled with ground truth distributions learns to accurately predict occupancy distributions for target objects unseen in training. We benchmark X-Ray in both simulated and physical experiments, showing that it can efficiently extract the target object from challenging heaps containing 15-25 objects that fully occlude the target object in 82% - 100% of heaps using a median of just 5 actions.

In future work, we will increase training efficiency and characterize the effect of training dataset size and number of translations and rotations used to generate training distributions. We will also explore reinforcement learning policies based on a reward of target object visibility, as this reward is directly tied to minimizing the support of the occupancy distribution. Finally, although we do not consider push actions here, we will add them for comparison to pushing and grasping policies.

ACKNOWLEDGMENTS

This research was performed at the AUTOLAB at UC Berkeley in affiliation with the Berkeley AI Research (BAIR) Lab. The authors were supported in part by donations from Google. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. 1752814. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Sponsors. We thank our colleagues and collaborators who provided helpful feedback, code, and suggestions, especially Julian Ibarz, Brijen Thananjeyan, Andrew Li, Andrew Lee, Andrey Kurenkov, Roberto Martín Martín, Animesh Garg, Matt Matl, and Ashwin Balakrishna.

| Policy  | Success Rate | Number of Actions | Quartiles |
|---------|--------------|-------------------|-----------|
| Random  | 85%          | 4                 | 6 7       |
| Largest | 85%          | 4                 | 6 7       |
| X-Ray   | 100%         | 4                 | 5 5.25    |

TABLE III: Evaluation metrics for each policy over 20 physical rollouts. The lower quartiles, medians, and upper quartiles for the number of actions are reported across successful rollouts. X-Ray extracts the target with significantly fewer actions, always extracting it within 10 actions.
