2018 IEEE International Conference on Robotics and Automation (ICRA)
May 21-25, 2018, Brisbane, Australia

Robotic Pick-and-Place of Novel Objects in Clutter with Multi-Affordance Grasping and Cross-Domain Image Matching

Andy Zeng[1], Shuran Song[1], Kuan-Ting Yu[2], Elliott Donlon[2], Francois R. Hogan[2], Maria Bauza[2], Daolin Ma[2], Orion Taylor[2], Melody Liu[2], Eudald Romo[2], Nima Fazeli[2], Ferran Alet[2], Nikhil Chavan Dafle[2], Rachel Holladay[2], Isabella Morona[2], Prem Qu Nair[1], Druck Green[2], Ian Taylor[2], Weber Liu[1], Thomas Funkhouser[1], Alberto Rodriguez[2]

[1]Princeton University [2]Massachusetts Institute of Technology

http://arc.cs.princeton.edu
https://youtu.be/6fG7zwGfIkI

Abstract— **This paper presents a robotic pick-and-place system that is capable of grasping and recognizing both known and novel objects in cluttered environments. The key new feature of the system is that it handles a wide range of object categories without needing any task-specific training data for novel objects. To achieve this, it first uses a category-agnostic affordance prediction algorithm to select and execute among four different grasping primitive behaviors. It then recognizes picked objects with a cross-domain image classification framework that matches observed images to product images. Since product images are readily available for a wide range of objects (e.g., from the web), the system works out-of-the-box for novel objects without requiring any additional training data. Exhaustive experimental results demonstrate that our multi-affordance grasping achieves high success rates for a wide variety of objects in clutter, and our recognition algorithm achieves high accuracy for both known and novel grasped objects. The approach was part of the MIT-Princeton Team system that took 1st place in the stowing task at the 2017 Amazon Robotics Challenge. All code, datasets, and pre-trained models are available online at http://arc.cs.princeton.edu**

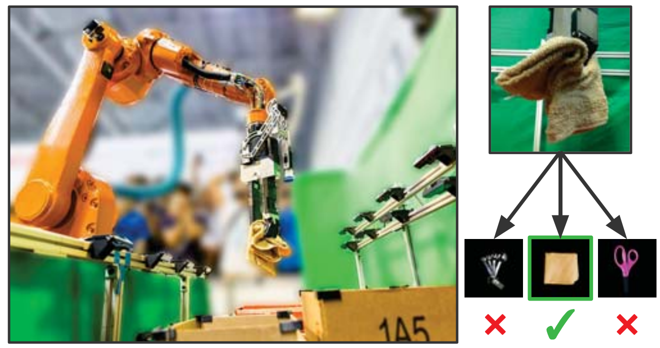

Fig. 1. **Our picking system** grasping a towel from a bin full of objects, holding it up away from clutter, and recognizing it by matching observed images of the towel to an available representative product image. The entire system works out-of-the-box for novel objects (appearing for the first time during testing) without the need for additional data collection or re-training.

I. Introduction

A human's remarkable ability to grasp and recognize unfamiliar objects with little prior knowledge of them is a constant inspiration for robotics research. This ability to grasp the unknown is central to many applications: from picking packages in a logistic center to bin-picking in a manufacturing plant; from unloading groceries at home to clearing debris after a disaster. The main goal of this work is to demonstrate that it is possible – and practical – for a robotic system to pick and recognize novel objects with very limited prior information about them (e.g. with only a few representative images scraped from the web).

Despite the interest of the research community, and despite its practical value, robust manipulation and recognition of novel objects in cluttered environments still remains a largely unsolved problem. Classical solutions for robotic picking require recognition and pose estimation prior to model-based grasp planning, or require object segmentation to associate grasp detections with object identities. These solutions tend to fall short when dealing with novel objects in cluttered environments, since they rely on 3D object models and/or large amounts of training data to achieve robust performance. Although there has been inspiring recent work on detecting grasps directly from RGB-D pointclouds as well as learning-based recognition systems to handle the constraints of novel objects and limited data, these methods have yet to be proven in the constraints and accuracy required by a real task with heavy clutter, severe occlusions, and object variability.

In this paper, we propose a system that picks and recognizes objects in cluttered environments. We have designed the system specifically to handle a wide range of objects novel to the system without gathering any task-specific training data for them. To make this possible, our system consists of two components: 1) a multi-modal grasping framework featuring four primitive behaviors, which uses deep convolutional neural networks (ConvNets) to predict affordances for a scene without a priori object segmentation and classification; and 2) a cross-domain image matching framework for recognizing grasped objects by matching them to product images, which uses a ConvNet architecture that adapts to novel objects without additional re-training. Both components work hand-in-hand to achieve robust picking performance of novel objects in heavy clutter.

We provide exhaustive experiments and ablation studies to evaluate both components. We demonstrate that the multi-affordance predictor for grasp planning achieves high suc-

The authors would like to thank the MIT-Princeton ARC team members for their contributions to this project, and ABB Robotics, Mathworks, Intel, Google, NSF (IIS-1251217 and VEC 1539014/1539099), and Facebook for hardware, technical, and financial support.

cess rates for a wide variety of objects in clutter, and the recognition algorithm achieves high accuracy for known and novel grasped objects. These algorithms were developed as part of the MIT-Princeton Team system that took 1st place in the stowing task of the Amazon Robotics Challenge (ARC), being the only system to have successfully stowed all known and novel objects from an unstructured tote into a storage system within the allotted time frame. Fig. 1 shows our robot in action during the competition.

In summary, our main contributions are:

- An object-agnostic picking framework using four primitive behaviors for fast and robust picking, utilizing a novel approach for estimating parallel jaw grasp affordances (Section IV).
- A perception framework for recognizing both known and novel objects using only product images without extra data collection or re-training (Section V).
- A system combining these two frameworks for picking novel objects in heavy clutter.

All code, datasets, and pre-trained models are available online at http://arc.cs.princeton.edu [1]. We also provide a video summarizing our approach at https://youtu.be/6fG7zwGfIkI, and a supplementary appendix with more details on our system at https://arxiv.org/abs/1710.01330.

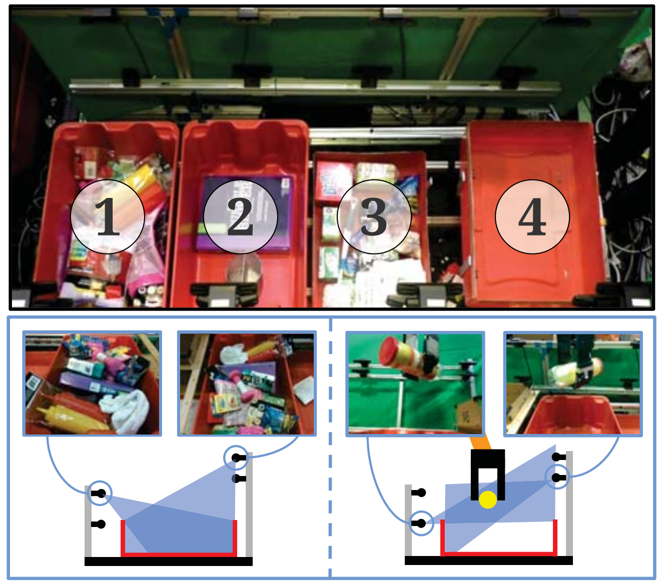

Fig. 2. **The bin and camera setup.** Our system consists of 4 units (top), where each unit has a bin with 4 stationary cameras: two overlooking the bin (bottom-left) are used for predicting grasp affordances while the other two (bottom-right) are used for recognizing the grasped object.

II. Related Work

In this section, we review works related to robotic picking systems. Works specific to grasping (Section IV) and recognition (Section V) are in their respective sections.

A. Recognition followed by Model-based Grasping

A large number of autonomous pick-and-place solutions follow a standard two-step approach: object recognition and pose estimation followed by model-based grasp planning. For example, Jonschkowski et al. [2] designed object segmentation methods over handcrafted image features to compute suction proposals for picking objects with a vacuum. More recent data-driven approaches [3], [4], [5], [6] use ConvNets to provide bounding box proposals or segmentations, followed by geometric registration to estimate object poses, which ultimately guide handcrafted picking heuristics [7], [8]. Nieuwenhuisen et al. [9] improve many aspects of this pipeline by leveraging robot mobility, while Liu et al. [10] adds a pose correction stage when the object is in the gripper. These works typically require 3D models of the objects during test time, and/or training data with the physical objects themselves. This is practical for tightly constrained pick-and-place scenarios, but is not easily scalable to applications that consistently encounter novel objects, for which only limited data (i.e. product images from the web) is available.

B. Recognition in parallel with Object-Agnostic Grasping

It is also possible to exploit local features of objects without object identity to efficiently detect grasps [11], [12], [13], [14], [15], [16], [17], [18], [19]. Since these methods are agnostic to object identity, they better adapt to novel objects and experience higher picking success rates by eliminating error propagation from a prior recognition step. Matsumoto et al. [20] apply this idea in a full picking system by using a ConvNet to compute grasp proposals, while in parallel predicting semantic segmentations for a fixed set of known objects. Although these pick-and-place systems use object-agnostic grasping methods, they still require some form of in-place object recognition in order to associate grasp proposals with object identities, which is particularly challenging when dealing with novel objects in clutter.

C. Active Perception

Active perception – exploiting control strategies for acquiring data to improve perception [21], [22] – can facilitate the recognition of novel objects in clutter. For example, Jiang et al. [23] describe a robotic system that actively rearranges objects in the scene (by pushing) in order to improve recognition accuracy. Other works [24], [25] explore next-best-view based approaches to improve recognition, segmentation and pose estimation results. Inspired by these works, our system applies active perception by using a grasp-first-then-recognize paradigm where we leverage object-agnostic grasping to isolate each object from clutter in order to significantly improve recognition accuracy for novel objects.

III. System Overview

We present a robotic pick-and-place system that grasps and recognized both known and novel objects in cluttered environments. The "known" objects are provided to the system at training time, both as physical objects and as representative product images (images of objects available on the web); while the "novel" objects are provided only at test time in the form of representative product images.

Overall approach. The system follows a *grasp-first-then-recognize* work-flow. For each pick-and-place operation, it

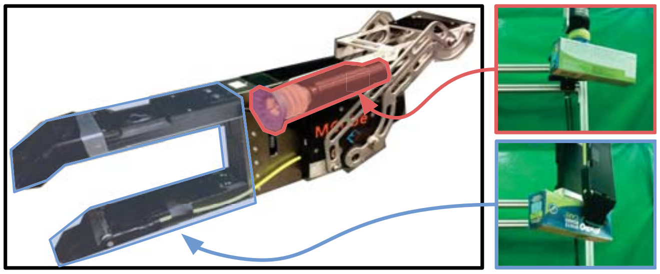

Fig. 3. **Multi-functional gripper** with a retractable mechanism that enables quick and automatic switching between suction (pink) and grasping (blue).

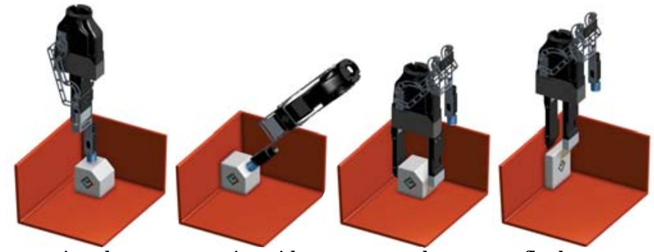

Fig. 4. **Multiple motion primitives** for suction and grasping to ensure successful picking for a wide variety of objects in any orientation.

first performs an object-agnostic affordance prediction, considering multiple different grasping modes from suction to parallel-jaw grasps (Section IV). It then selects the best affordance, picks up one object, isolates it from the clutter, holds it up in front of cameras, recognizes its category, and places it in the appropriate bin. Although the object recognition algorithm is trained only on known objects, it is able to recognize novel objects through a learned cross-domain image matching embedding between observed images of held objects and product images (Section V).

Advantages. This system design has several advantages. First, the affordance prediction algorithm is model-free and agnostic to object identities and generalizes to novel objects without re-training. Second, the category recognition algorithm works without task-specific data collection or re-training for novel objects, which makes it scalable for applications in warehouse automation and service robots where the range of observed object categories is large and dynamic. Third, the affordance prediction algorithm supports multiple grasping modes and thus handles a wide variety of objects. Finally, the entire processing pipeline requires only two forward passes through deep networks and thus executes quickly (Table II).

System setup. Our system features a 6DOF ABB IRB 1600id robot arm next to four picking work-cells. The robot arm's end-effector is a multi-functional gripper with two fingers for parallel-jaw grasps and a retractable suction cup (Fig. 3). This gripper was designed to function in cluttered environments: finger and suction cup length are specifically chosen such that the bulk of the gripper body does not need to enter the cluttered space. Each work-cell has a storage bin and four statically-mounted RealSense SR300 RGB-D cameras (Fig. 2): two cameras overlooking the storage bins are used to predict grasp affordances, while the other two pointing towards the robot gripper are used to recognize objects in the gripper. Although our experiments were performed with this setup, the system was designed to be flexible for picking and placing between any number of reachable work-cells and camera locations. Furthermore, all manipulation and recognition algorithms in this paper were designed to be easily adapted to other system setups.

IV. Multi-Affordance Grasping

The goal of the first step in our system is to robustly grasp objects from a cluttered scene without relying on their object identities or poses. To this end, we define a set of motion primitives that are complimentary to each other in terms of utility across different object types and scenarios – empirically maximizing the variety of objects and orientations that can be picked with at least one primitive. Given RGB-D images of the cluttered scene at test time, we predict a set of affordances to generate grasp proposals with confidence scores for each primitive. These are then used by a task planner to choose which primitive to use.

A. Motion primitives

We define four motion primitives to achieve robust picking for typical household objects. Fig. 4 shows example motions for each primitive. Each of them are implemented as a set of guarded moves, with collision avoidance and quick success or failure feedback mechanisms. They are as follows:

Suction down grasps objects with a vacuum gripper vertically. This primitive is particularly robust for objects with large and flat suctionable surfaces (e.g. boxes, books, wrapped objects), and performs well in heavy clutter.

Suction side grasps objects from the side by approaching with a vacuum gripper tilted an an angle. This primitive is robust to thin and flat objects resting against walls, which may not have suctionable surfaces from the top.

Grasp down grasps objects vertically using the two-finger parallel-jaw gripper. This primitive is complementary to the suction primitives in that it is able to pick up objects with smaller, irregular surfaces (e.g. small tools, deformable objects), or made of semi-porous materials that prevent a good suction seal (e.g. cloth).

Flush grasp retrieves unsuctionable objects that are flushed against a wall. The primitive is similar to grasp down, but with the additional behavior of using a flexible spatula to slide between the target object and the wall.

B. Affordance Prediction

Given the set of pre-defined picking primitives and RGB-D images of the scene, we predict pixel-level affordances for each motion primitive, from which we can generate suction and grasp proposals. Our approach relies on the assumption that graspable regions can be deduced from the local geometry and material properties, as reflected in visual information. This is inspired by recent data-driven methods for grasp planning [11], [12], [13], [15], [16], [17], [18], [19], which do not rely on object identities or state estimation. We

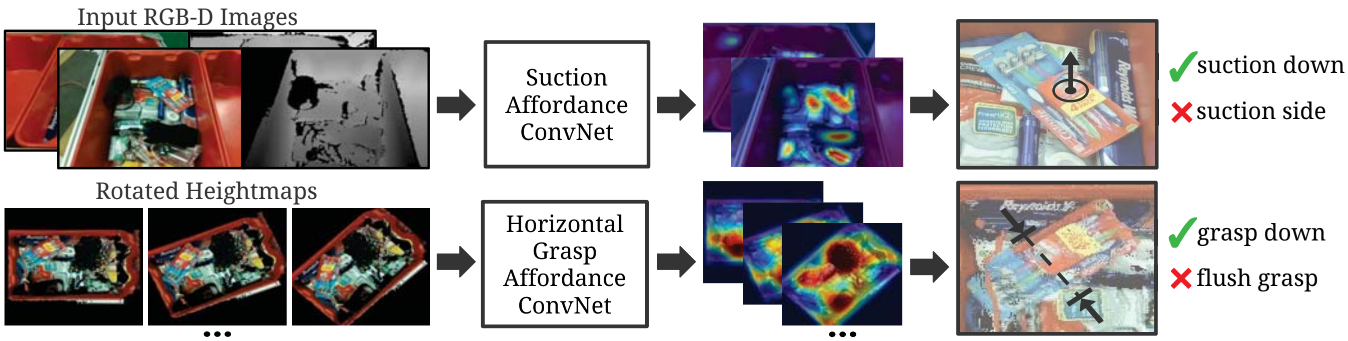

Fig. 5. **Suction and grasp affordance prediction.** Given multi-view RGB-D images, we estimate suction affordances for each image with a fully convolutional residual network. We then aggregate the predictions on a 3D point cloud, and generate suction down or suction side proposals based on surface normals. In parallel, we merge RGB-D images into an RGB-D heightmap, rotate it by 16 different angles, and estimate horizontal grasp for each heightmap. This effectively produces affordance maps for 16 different grasp angles, from which we generate the grasp down and flush grasp proposals.

extend these data-driven approaches by training models to predict pixel-level affordances for multiple types of grasps, and employ fully convolutional networks (FCN) [26] to efficiently obtain dense predictions over a single image of the scene to achieve faster run time speeds.

In this subsection, we present an overview of how we predict affordances for our suction and grasping primitives. For more details about our network architectures, their training parameters, post-processing steps, and training datasets, please refer to our project webpage [1].

Predicting Suction Affordances. We define suction proposals as 3D positions where the vacuum gripper's suction cup should come in contact with the object's surface in order to successfully grasp it. Good suction proposals should be located on suctionable surfaces, and nearby the target object's center of mass to avoid an unstable suction seal (e.g. particularly for heavy objects). Each suction proposal is defined as a 3D position x, y, z, its surface normal n_x, n_y, n_z, and confidence score c_s.

We train a fully convolutional residual network (ResNet-101 [27]), that takes an RGB-D image as input, and outputs a densely labeled pixel-level binary probability map c_s, where values closer to one imply a more preferable suction location, shown in Fig. 5 first row. Our network architecture is multi-modal, where the color data is fed into one ResNet-101 tower, and 3-channel depth (cloned across channels, normalized by subtracting mean and dividing by standard deviation) is fed into another ResNet-101 tower. Features from the ends of both towers are concatenated across channels, followed by 3 additional spatial convolution layers to merge the features; then spatially bilinearly upsampled and softmaxed to output a single binary probability map. We train our model over a manually annotated dataset of RGB-D images of cluttered scenes with diverse objects, where pixels are densely labeled either positive, negative, or neither (using wide-area brushstrokes from the labeling interface). We train our network with 0 loss propagation for the regions that are labeled as neither positive nor negative.

During testing, we feed each captured RGB-D image through our trained network to generate probability maps for each view. As a post-processing step, we use calibrated camera intrinsics and poses to project the probability maps and aggregate the affordance predictions onto a combined 3D point cloud. We then compute surface normals for each 3D point, which are used to classify which suction primitive (down or side) to use for the point. To handle objects without depth, we use a simple hole filling algorithm [28] on the depth images, and project predicted probability scores onto the hallucinated depth.

Predicting Grasp Affordances. Each grasp proposal is represented by the x, y, z position of the gripper in 3D space, the orientation θ of the gripper around the vertical axis, the desired gripper opening distance d_o, and confidence score c_g.

To predict grasping affordances, we first aggregate the two RGB-D images of the scene into a registered 3D point cloud, which is then orthographically back-projected upwards in the gravity direction to obtain a "heightmap" image representation of the scene, with both color (RGB) and height from bottom (D) channels. To handle objects without depth, we triangulate no-depth regions in the heightmap using both views, and fill in the regions with a height of 3cm. We feed this RGB-D heightmap as input to a fully convolutional ResNet-101 [27], which densely predicts pixel-level binary probability maps, which serve as confidences values c_g for horizontally oriented grasps, shown in Fig. 5 second row. The architecture of this network is similar in structure to the network predicting suction affordances. By rotating the heightmap in 16 different orientations and feeding each individually through the network, we obtain 16 binary probability maps, each representing a confidence map for a grasp in a different orientation.

We find this network architecture to be more flexible to various grasp orientations, and less likely to diverge during training due to the sparsity of manual grasp annotations. We train our model over a manually annotated dataset of RGB-D heightmaps, where each positive and negative grasp label is represented by a pixel on the heightmap as well as a corresponding angle parallel to the jaw motion of the gripper.

Our grasp affordance predictions return grasp locations (x, y, z), orientations (θ), and confidence scores (c_g). During post-processing, we use the geometry of the 3D point cloud to estimate grasp widths (d_o) for each proposal. We also use

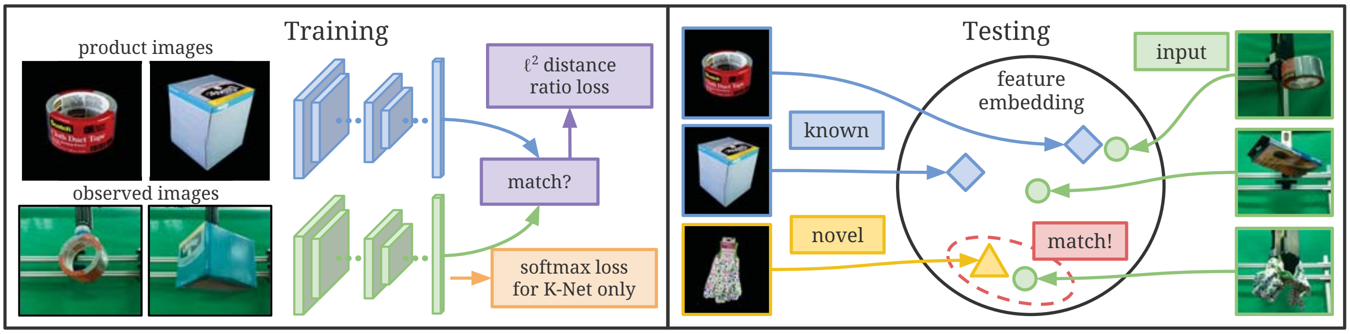

Fig. 6. **Recognition framework for novel objects.** We train a two-stream convolutional neural network where one stream computes 2048-dimensional feature vectors for product images while the other stream computes 2048-dimensional feature vectors for observed images, and optimize both streams so that features are more similar for images of the same object and dissimilar otherwise. During testing, product images of both known and novel objects are mapped onto a common feature space. We recognize observed images by mapping them to the same feature space and finding the nearest neighbor match.

the location of each proposal relative to the bin to classify which grasping primitive (down or flush) should be used.

V. Recognizing Novel Objects

After successfully grasping an object and isolating it from clutter, the goal of the second step in our system is to recognize the identity of the grasped object.

Since we encounter both known and novel objects, and we have only product images for the novel objects, we address this recognition problem by retrieving the best match among a set of product images. Of course, observed images and product images can be captured in significantly different environments in terms of lighting, object pose, background color, post-process editing, etc. Therefore, we need a model that is able to find the semantic correspondences between images from these two different domains. This is a *cross-domain image matching* problem [29], [30], [31].

A. Metric Learning for Cross-Domain Image Matching

To do the cross-domain image matching between observed images and product images, we learn a metric function that takes in an observed image and a candidate product image and outputs a distance value that models how likely the images are of the same object. The goal of the metric function is to map both the observed image and product image onto a meaningful feature embedding space so that smaller ℓ_2 feature distances indicate higher similarities. The product image with the smallest metric distance to the observed image is the final matching result.

We model this metric function with a two-stream convolutional neural network (ConvNet) architecture where one stream computes features for the observed images, and a different stream computes features for the product images. We train the network by feeding it a balanced 1:1 ratio of matching and non-matching image pairs (one observed image and one product image) from the set of known objects, and backpropagate gradients from the distance ratio loss (Triplet loss [32]). This effectively optimizes the network in a way that minimizes the ℓ_2 distances between features of matching pairs while pulling apart the ℓ_2 distances between features of non-matching pairs. By training over enough examples of these image pairs across known objects, the network learns a feature embedding that encapsulates object shape, color, and other visual discriminative properties, which can generalize and be used to match observed images of novel objects to their respective product images (Fig. 6).

Avoiding metric collapse by guided feature embeddings. One issue commonly encountered in metric learning occurs when the number of training object categories is small – the network can easily overfit its feature space to capture only the small set of training categories, making generalization to novel object categories difficult. We refer to this problem as metric collapse. To avoid this issue, we use a model pre-trained on ImageNet [33] for the product image stream and train only the stream that computes features for observed images. ImageNet contains a large collection of images from many categories, and models pre-trained on it have been shown to produce relatively comprehensive and homogenous feature embeddings for transfer tasks [34] – i.e. providing discriminating features for images of a wide range of objects. Our training procedure trains the observed image stream to produce features similar to the ImageNet features of product images – i.e., it learns a mapping from observed images to ImageNet features. Those features are then suitable for direct comparison to features of product images, even for novel objects not encountered during training.

Using multiple product images. For many applications, there can be multiple product images per object. However, with multiple product images, supervision of the two-stream network can become confusing - on which pair of matching observed and product images should the backpropagated gradients be based? To solve this problem, we add a module we call a "multi-anchor switch" in the network. During training, this module automatically chooses which "anchor" product image to compare against based on nearest neighbor ℓ_2 distance. We find that allowing the network to select its own criterion for choosing "anchor" product images provides a significant boost in performance in comparison to alternative methods like random sampling.

B. Two Stage Framework for a Mixture of Known and Novel Objects

In settings where both types of objects are present, we find that training two different network models to handle known and novel objects separately can yield higher overall matching accuracies. One is trained to be good at "over-fitting" to the known objects (K-net) and the other is trained to be better at "generalizing" to novel objects (N-net).

Yet, how do we know which network to use for a given image? To address this issue, we execute our recognition pipeline in two stages: a "recollection" stage that determines whether the observed object is known or novel, and a "hypothesis" stage that uses the appropriate network model based on the first stage's output to perform image matching.

First, the recollection stage predicts whether the input observed image from test time is that of a known object that has appeared during training. Intuitively, an observed image is of a novel object if and only if its deep features cannot match to that of any images of known objects. We explicitly model this conditional by thresholding on the nearest neighbor distance to product image features of known objects. In other words, if the ℓ_2 distance between the K-net features of an observed image and the nearest neighbor product image of a known object is greater than some threshold k, then the observed images is a novel object.

In the hypothesis stage, we perform object recognition based on one of two network models: K-net for known objects and N-net for novel objects. The K-net and N-net share the same network architecture. However, the K-net has an additional auxiliary classification loss during training for the known objects. This classification loss increases the accuracy of known objects at test time to near perfect performance, and also boosts up the accuracy of the recollection stage, but fails to maintain the accuracy of novel objects. On the other hand, without the restriction of the classification loss, N-net has a lower accuracy for known objects, but maintains a better accuracy for novel objects.

By adding the recollection stage, we can exploit both the high accuracy of known objects with K-net and good accuracy of novel objects with N-net, though incurring a cost in accuracy from erroneous known vs novel classification. We find that this two stage system overall provides higher total matching accuracy for recognizing both known and novel objects (mixed) than all other baselines (Table III).

VI. Experiments

In this section, we evaluate our multi-affordance prediction for suction and grasp primitives, our recognition algorithm over both known and novel objects, as well as our full system in the context of the Amazon Robotics Challenge 2017.

A. Multi-Affordance Prediction Experiments

Datasets. To generate datasets for affordance predictions, we designed a simple labeling interface that prompts users to manually annotate suction and grasp proposals over RGB-D images collected from the real system. For suction, users who have had experience working with our suction gripper are asked to annotate pixels of suctionable and non-suctionable areas on raw RGB-D images overlooking cluttered bins full of various objects. Similarly, users with experience using our parallel-jaw gripper are asked to sparsely annotate

TABLE I

Multi-Affordance Prediction Performance

Primitive	Method	Top-1	Top 1%	Top 5%	Top 10%
Suction	Baseline	35.2	55.4	46.7	38.5
	ConvNet	**92.4**	83.4	66.0	52.0
Grasping	Baseline	92.5	90.7	87.2	73.8
	ConvNet	**96.7**	91.9	87.6	84.1

% precision of predictions across different confidence percentiles.

positive and negative grasps over re-projected height maps of cluttered bins, where each grasp is represented by a pixel on the height map and an angle parallel to the jaw motion of the gripper. We further augment each grasp label by adding additional labels with small jittering (less than 1.6cm). In total, the dataset contains 1837 RGB-D images with suction and grasp labels. We use a 4:1 training/testing split across this dataset to train and evaluate different models.

Evaluation. In the context of our system, an affordance prediction method is robust if it is able to consistently find at least one suction or grasp proposal that works. To reflect this, our evaluation metric is the precision of predicted proposals versus manual annotations. For suction, a proposal is considered a true positive if its pixel center is manually labeled as a suctionable area. For grasping, a proposal is considered a true positive prediction if its pixel center is within 4 pixels and 11.25 degrees from a positive grasp label.

We report the precision of our predicted proposals for different confidence percentiles in Table I. The precision of the top-1 proposal is reliably above 90% for both suction and grasping. We further compare our methods to heuristic-based baseline algorithms that compute suction affordances by estimating surface normal variance over the observed 3D point cloud (lower variance = higher affordance), and computes anti-podal grasps by detecting hill-like geometric structures in the 3D point cloud. Baseline details and code are available on our project webpage [1].

Speed. Our suction and grasp affordance algorithms were designed to achieve fast run-time speeds during test time by densely predicting affordances over a single image of the entire scene. In Table II, we compare our run-time speeds to several state-of-the-art alternatives for grasp planning. Our own numbers measure the time of each FCN forward pass, reported with an NVIDIA Titan X on an Intel Core i7-3770K clocked at 3.5 GHz, excluding time for image capture and other system-related overhead.

B. Recognition of Novel Objects Evaluation

We evaluate our recognition algorithms using a 1 vs 20 classification benchmark. Each test sample in the benchmark contains 20 possible object classes, where 10 are known and 10 are novel, chosen at random. During each test sample, we feed the recognition algorithm the product images for all 20 objects as well as an observed image of a grasped object. In Table III, we measure performance in terms of top-1 accuracy for matching the observed image to a product image of the correct object match. We evaluate our method against a baseline algorithm, a state-of-the-art network architecture for both visual search [31] and one shot learning without

TABLE II

GRASP PLANNING RUN-TIMES (SEC.)

Method	Time
Lenz et al. [12]	13.5
Zeng et al. [4]	10 - 15
Hernandez et al. [3]	5 - 40 [a]
Schwarz et al. [5]	0.9 - 3.3
Dex-Net 2.0 [17]	0.8
Matsumoto et al. [20]	0.2
Redmon et al. [13]	0.07
Ours (suction)	0.06
Ours (grasping)	$0.05 \times n$ [b]

[a] times reported from [20] derived from [3].
[b] n = number of possible grasp angles.

retraining [35], and several variations of our method. The latter provides an ablation study to show the improvements in performance with every added component:

Nearest Neighbor is a baseline algorithm where we compute features of product images and observed images using a ResNet-50 pre-trained on ImageNet, and use nearest neighbor matching with ℓ_2 distance.

Siamese network with weight sharing is a re-implementation of Bell et al. [31] for visual search and Koch et al. [35] for one shot recognition without retraining. We use a Siamese ResNet-50 pre-trained on ImageNet and optimized over training pairs in a Siamese fashion. The main difference between this method and ours is that the weights between the networks computing deep features for product images and observed images are shared.

Two-stream network without weight sharing is a two-stream network, where the networks' weights for product images and observed images are not shared. Without weight sharing the network has more flexibility to learn the mapping function and thus achieves higher matching accuracy. All the later models describe later in this section use this two stream network without weight sharing.

Two-stream + guided-embedding (GE) includes a guided feature embedding with ImageNet features for the product image stream. We find this model has better performance for novel objects than for known objects.

Two-stream + guided-embedding (GE) + multi-product-images (MP) By adding a multi-anchor switch, we see more improvements to accuracy for novel objects. This is the final network architecture for N-net.

Two-stream + guided-embedding (GE) + multi-product-images (MP) + auxiliary classification (AC) By adding an auxiliary classification, we achieve near perfect accuracy of known objects for later models, however, at the cost of lower accuracy for novel objects. This also improves known vs novel (K vs N) classification accuracy for the recollection stage. This is the final network architecture for K-net.

Two-stage system As described in Section V, we combine the two different models - one that is good at known objects (K-net) and the other that is good at novel objects (N-net) - in the two stage system. This is our final recognition algorithm, and it achieves better performance than any single model for test cases with a mixture of known and novel objects.

TABLE III

RECOGNITION EVALUATION (% ACCURACY OF TOP-1 PREDICTION)

Method	K vs N	Known	Novel	Mixed
Nearest Neighbor	69.2	27.2	52.6	35.0
Siamese ([31], [35])	70.3	76.9	68.2	74.2
Two-stream	70.8	85.3	75.1	82.2
Two-stream + GE	69.2	64.3	79.8	69.0
Two-stream + GE + MP (N-net)	69.2	56.8	**82.1**	64.6
N-net + AC (K-net)	**93.2**	**99.7**	29.5	78.1
Two-stage K-net + N-net	93.2	93.6	77.5	**88.6**

C. Full System Evaluation in Amazon Robotics Challenge

To evaluate the performance of our system as a whole, we used it as part of our MIT-Princeton entry for the 2017 Amazon Robotics Challenge (ARC), where state-of-the-art pick-and-place solutions competed in the context of a warehouse automation task. Participants were tasked with designing a robot system to grasp and recognize a large variety of different objects in unstructured storage systems. The objects were characterized by a number of difficult-to-handle properties. Unlike earlier versions of the competition [36], half of the objects were novel in the 2017 edition of the competition. The physical objects as well as related item data (i.e. product images, weight, 3D scans), were given to teams just 30 minutes before the competition. While other teams used the 30 minutes to collect training data for the new objects and retrain models, our unique system did not require any of that during those 30 minutes.

Setup. Our system setup for the competition features several differences. We incorporated weight sensors to our system, using them as a guard to signal stop or modify primitive behavior during execution. We also used the measured weights of objects provided by Amazon to boost recognition accuracy to near perfect performance. Green screens made the background more uniform to further boost accuracy of the system in the recognition phase. For predicting affordances, Table I shows that our data-driven methods with ConvNets give improved affordance predictions for both suction and grasping, with respect to the baseline algorithms. For the case of grasping, however, we did not have time to develop a fully stable ConvNet before the day of the competition, so we decided to avoid risks and use the baseline grasping algorithm. The ConvNet approach became stable with the reduction to predicting only horizontal grasps and rotating the heightmaps. Additionally for the competition, we also designed a placing algorithm that uses heightmaps and object bounding boxes to determine stable placements for the objects after recognition.

Results. During the ARC 2017 final stowing task, we had a 58.3% pick success with suction, 75% pick success with grasping, and 100% recognition accuracy during the stow task of the ARC, stowing all 20 objects within 24 suction attempts and 8 grasp attempts. Our system took 1st place in the stowing task, being the only system to have successfully stowed all known and novel objects and to have finished the task well within the allotted time frame.

VII. Discussion and Future Work

We present a system to pick and recognize novel objects with very limited prior information about them (a handful of product images). The system first uses a category-agnostic affordance prediction algorithm to select among four different grasping primitive behaviors, and then recognizes grasped objects by matching them to their product images. We evaluate both components and demonstrate their combination in a robot system that picks and recognizes novel objects in heavy clutter, and that took 1st place in the stowing task of the Amazon Robotics Challenge 2017. Here are some of the most salient features/limitations of the system:

Object-Agnostic Manipulation. The system finds grasp affordances directly in the RGBD image. This proved faster and more reliable than doing object segmentation and state estimation prior to grasp planning [4]. The ConvNet learns the visual features that make a region of an image graspable or suctionable. It also seems to learn more complex rules, e.g., that tags are often easier to suction that the object itself, or that the center of a long object is preferable than its ends. It would be interesting to explore the limits of the approach. For example learning affordances for more complex behaviors, e.g., scooping an object against a wall, which require a more global understanding of the geometry of the environment.

Pick First, Ask Questions Later. The standard grasping pipeline is to first recognize and then plan a grasp. In this paper we demonstrate that it is possible and sometimes beneficial to reverse the order. Our system leverages object-agnostic picking to remove the need for state estimation in clutter. Isolating the picked object drastically increases object recognition reliability, especially for novel objects. We conjecture that "pick first, ask questions later" is a good approach for applications such as bin-picking, emptying a bag of groceries, or clearing debris. It is, however, not suited for all applications – nominally when we need to pick a particular object. In that case, the described system needs to be augmented with state tracking/estimation algorithms.

Towards Scalable Solutions. Our system is designed to pick and recognize novel objects without extra data collection or retraining. This is a step forward towards robotic solutions that scale to the challenges of service robots and warehouse automation, where the daily number of novel objects ranges from the tens to the thousands, making data-collection and retraining cumbersome in one case and impossible in the other. It is interesting to consider what data, besides product images, is available that could be used for recognition using out-of-the-box algorithms like ours.

References

[1] Webpage for code and data. [Online]. Available: arc.cs.princeton.edu

[2] R. Jonschkowski, C. Eppner, S. Höfer, R. Martín-Martín, and O. Brock, "Probabilistic multi-class segmentation for the amazon picking challenge," 2016.

[3] C. Hernandez, M. Bharatheesha, W. Ko, H. Gaiser, J. Tan, K. van Deurzen, M. de Vries, B. Van Mil, *et al.*, "Team delft's robot winner of the amazon picking challenge 2016," *arXiv*, 2016.

[4] A. Zeng, K.-T. Yu, S. Song, D. Suo, E. Walker Jr, A. Rodriguez, and J. Xiao, "Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge," in *ICRA*, 2017.

[5] M. Schwarz, A. Milan, C. Lenz, A. Munoz, A. S. Periyasamy, M. Schreiber, S. Schüller, and S. Behnke, "Nimbro picking: Versatile part handling for warehouse automation," in *ICRA*, 2017.

[6] J. M. Wong, V. Kee, T. Le, S. Wagner, G.-L. Mariottini, A. Schneider, L. Hamilton, R. Chipalkatty, M. Hebert, *et al.*, "Segicp: Integrated deep semantic segmentation and pose estimation," *arXiv*, 2017.

[7] A. Bicchi and V. Kumar, "Robotic Grasping and Contact," *ICRA*.

[8] A. Miller, S. Knoop, H. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," *ICRA*, 2003.

[9] M. Nieuwenhuisen, D. Droeschel, D. Holz, J. Stückler, A. Berner, J. Li, R. Klein, and S. Behnke, "Mobile bin picking with an anthropomorphic service robot," in *ICRA*, 2013.

[10] M.-Y. Liu, O. Tuzel, A. Veeraraghavan, Y. Taguchi, T. K. Marks, and R. Chellappa, "Fast object localization and pose estimation in heavy clutter for robotic bin picking," *IJRR*, 2012.

[11] e. a. Morales, Antonio, "Using experience for assessing grasp reliability," in *IJHR*, 2004.

[12] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," in *IJRR*, 2015.

[13] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *ICRA*, 2015.

[14] A. ten Pas and R. Platt, "Using geometry to detect grasp poses in 3d point clouds," in *ISRR*, 2015.

[15] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *ICRA*, 2016.

[16] L. Pinto, J. Davidson, and A. Gupta, "Supervision via competition: Robot adversaries for learning tasks," in *ICRA*, 2017.

[17] e. a. Mahler, Jeffrey, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," in *RSS*, 2017.

[18] M. Gualtieri, A. ten Pas, K. Saenko, and R. Platt, "High precision grasp pose detection in dense clutter," in *arXiv*, 2017.

[19] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with large-scale data collection," in *ISER*, 2016.

[20] E. Matsumoto, M. Saito, A. Kume, and J. Tan, "End-to-end learning of object grasp poses in the amazon robotics challenge."

[21] R. Bajcsy and M. Campos, "Active and exploratory perception," *CVGIP: Image Understanding*, vol. 56, no. 1, 1992.

[22] S. Chen, Y. Li, and N. M. Kwok, "Active vision in robotic systems: A survey of recent developments," *IJRR*, 2011.

[23] D. Jiang, H. Wang, W. Chen, and R. Wu, "A novel occlusion-free active recognition algorithm for objects in clutter," in *ROBIO*, 2016.

[24] K. Wu, R. Ranasinghe, and G. Dissanayake, "Active recognition and pose estimation of household objects in clutter," in *ICRA*, 2015.

[25] D. Jayaraman and K. Grauman, "Look-ahead before you leap: End-to-end active recognition by forecasting the effect of motion," in *ECCV*, 2016.

[26] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[28] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *ECCV*, 2012.

[29] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," *ECCV*, 2010.

[30] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros, "Data-driven visual similarity for cross-domain image matching," in *TOG*, 2011.

[31] S. Bell and K. Bala, "Learning visual similarity for product design with convolutional neural networks," *TOG*, 2015.

[32] E. Hoffer, I. Hubara, and N. Ailon, "Deep unsupervised learning through spatial contrasting," *arXiv*, 2016.

[33] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[34] M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv*, 2016.

[35] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Workshop*, 2015.

[36] N. Correll, K. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. Romano, and P. Wurman, "Analysis and Observations from the First Amazon Picking Challenge," *T-ASE*, 2016.