Affordance Detection of Tool Parts from Geometric Features

Austin Myers, Ching L. Teo, Cornelia Fermüller, and Yiannis Aloimonos

Abstract— **As robots begin to collaborate with humans in everyday workspaces, they will need to understand the functions of tools and their parts. To cut an apple or hammer a nail, robots need to not just know the tool's name, but they must localize the tool's parts and identify their functions. Intuitively, the geometry of a part is closely related to its possible functions, or its *affordances*. Therefore, we propose two approaches for learning affordances from local shape and geometry primitives: 1) superpixel based hierarchical matching pursuit (S-HMP); and 2) structured random forests (SRF). Moreover, since a part can be used in many ways, we introduce a large RGB-Depth dataset where tool parts are labeled with multiple affordances and their relative rankings. With ranked affordances, we evaluate the proposed methods on 3 cluttered scenes and over 105 kitchen, workshop and garden tools, using ranked correlation and a weighted F-measure score [26]. Experimental results over sequences containing clutter, occlusions, and viewpoint changes show that the approaches return precise predictions that could be used by a robot. S-HMP achieves high accuracy but at a significant computational cost, while SRF provides slightly less accurate predictions but in real-time. Finally, we validate the effectiveness of our approaches on the Cornell Grasping Dataset [25] for detecting graspable regions, and achieve state-of-the-art performance.**

I. Introduction

Every day, we use tools for ordinary activities, like cutting an apple, hammering a nail, or watering a flower. While interacting with the world, we effortlessly draw on our understanding of the functions that tools and their parts provide. Using vision, we identify the functionality of parts, so we can find the right tool for our needs. As robots like PR2, Asimo, and Baxter begin to collaborate with humans in everyday workspaces, they will also need to understand the wide variety of tools useful for their tasks.

Imagine Baxter in a kitchen, trying to serve soup from a pot into a bowl. Baxter needs to find a ladle, grab the handle, dip the bowl of the ladle into the pot, and transfer the soup to the serving bowl. But what if this ladle has a different shape and color from the ladles that Baxter has seen before? What if Baxter has never seen any ladles at all? Today, computer vision allows robots to recognize objects from a known category, providing a bounding box around the ladle. However, in these situations Baxter needs to not just detect the ladle, but more importantly he needs to know which part of the ladle he can grasp and which part can contain the soup. As Gibson remarked, "If you know what can be done with a[n] object, what it can be used for, you can call it whatever you please" [11].

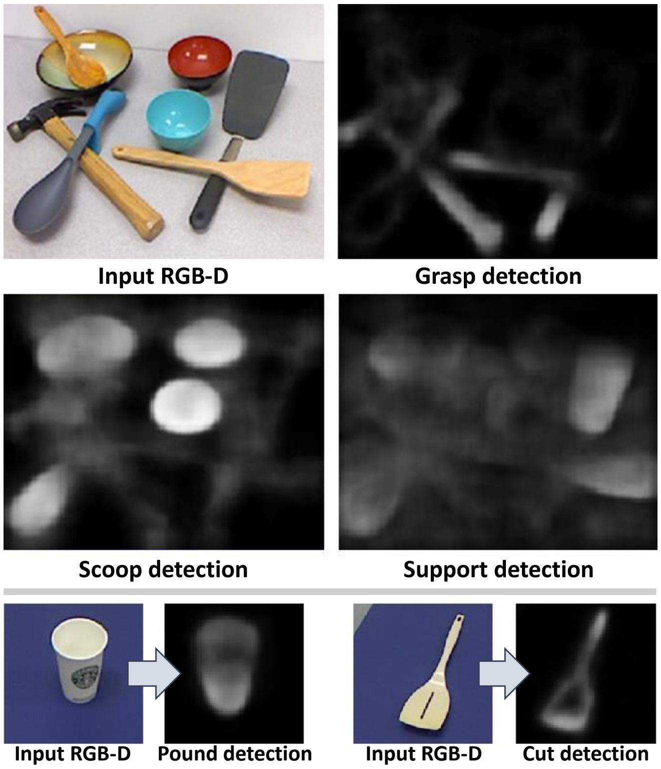

Fig. 1: Predicting novel affordances in clutter (top) and in single objects (bottom). (Top) Detections of `grasp`, `scoop` and `support` in a cluttered scene. (Bottom) Novel affordance predicted for mug: `pound` (left) and turner (spatula): `cut` (right). Notice that we are able to predict and localize reasonable locations for novel affordances, even in clutter and not just on well-defined object parts, but on the relevant regions of the object (e.g. the bottom of the mug affords pounding and the edge of the turner affords cutting). Brighter regions indicate higher probability.

In this paper, we address the novel problem of localizing and identifying part *affordance*, so that a robot can explain how an object and its parts can be used, and generalize this knowledge to novel scenarios. Outputs demonstrating this generalization are shown in Fig. 1 where, for example, the proposed approach is able to predict that the bottom of the mug is useful for pounding, or the edge of a turner can be used for cutting.

Gibson defined affordances as the latent "action possibilities" available to an agent, given their capabilities and the environment [11]. In this sense, for a human adult, stairs afford climbing, an apple affords eating, and a knife affords the cutting of another object. The last example is the most relevant to a robot using tools in a kitchen or workshop, and we use the term *effective affordance* to differentiate the affordances of tools from those found in other settings. We

The authors are with the Department of Computer Science, University of Maryland, College Park, MD 20742, USA {amyers, cteo, fer, yiannis}@umiacs.umd.edu

define objects with effective affordances as those that an agent can use as tools to produce an effect on another object. Man-made tools are typically composed of parts, where each part has multiple effective affordances such as cut, pound, scoop, or contain. If robots could identify these affordances, it would open the possibility to use a wide variety of tools, including those that have not been seen before.

From a computer vision perspective however, predicting affordances from an image presents a major challenge because tools from different categories, with unique shapes and appearances, can have parts with the same effective affordances. Furthermore, our goal is to identify affordances at the level of parts, and provide precise predictions for a robot to interact with the world.

The main contributions of this paper are as follows: 1) We introduce a framework for jointly localizing and identifying part affordance, so that robots can understand how objects and their parts can be used. We show that this approach can be used to identify and localize part affordance for a large collection of tools and in cluttered scenes with occlusions (§V). 2) We use and compare two methods for learning the association: a) an unsupervised feature learning method which learns a hierarchy of sparse dictionaries (§III-B) and b) a fast structured random forest classifier that preserves the spatial information of the learned affordances at its leaf nodes (§III-C). 3) We analyze the effectiveness of different features for affordance identification, and demonstrate in the experiments that geometric features, derived from a combination of 2D and 2.5D data, are essential for the task (§V-.1). 4) We present a new RGB-D Part Affordance Dataset (§IV-A) which consists of 105 kitchen, workshop, and garden tools. The dataset provides hand-labeled ground truth at the pixel level for more than 10,000 RGB-D images. In addition to images of single objects, a separate dataset of novel objects in *clutter* is also available for evaluating the robustness of affordance detection in real-world settings (Fig. 1 (top)). Dataset and code from this work are available online[1].

II. Related Work

The study of affordance has a rich history in the computer vision and robotics communities. Early work sought a function-based approach to object recognition for 3D CAD models of objects like chairs [33]. More recently, many papers have focused on predicting grasping points for objects from 2D images [30] [34] [5]. [25] exploits a deep learning framework to learn graspable features from RGB-D images of complex objects and [17] detects tips of tools being held by a robot. From the computer vision community, [19] classify human hand actions in context of the objects being used, Grabner et al. [12] detect surfaces for sitting from 3D data.

Affordances might be considered a subset of object attributes, which have been shown to be powerful for object recognition tasks as well as transferring knowledge to new categories. Ferrari and Zisserman [10] learn color and 2D shape patterns to recognize the attributes in novel images. Parikh and Grauman [29] show that relative attributes can be used to rank images relative to one another, and Lampert et al. [24] and Yu et al. [36] show that attributes can be used to transfer knowledge to novel object categories. In the robotics community, the authors of [35] identify color, shape, material, and name attributes of objects selected in a bounding box from RGB-D data. [14] explored, using active manipulation of different objects, the influence of the shape, material and weight in predicting good pushable locations. [2] used a full 3D mesh model to learn so-called 0-ordered affordances that depend on object poses and relative geometry. Koppula et al. [22] view affordance of objects as a function of interactions, and jointly model both object interactions and activities via a Markov Random Field using 3D geometric relations (‘on top’, ‘below’ etc.) between the tracked human and object as features.

Recently, unsupervised feature learning approaches have been applied to problems with 3D information. [3] propose using hierarchical matching pursuit (HMP), and [32] propose using a convolutional recursive neural network to recognize objects from RGB-D images. For supervised methods, state-of-the-art performance using structured random forests [21] applied over RGB-D data for simultaneous object segmentation and recognition has been reported in [13].

III. Approach

In this paper we compare two approaches for associating part affordances with geometric features extracted from RGB-D images. The first approach builds upon the recent work of [4] which uses multipath HMP (§III-B) to achieve state-of-the-art performance on challenging computer vision image datasets. The second approach leverages the fast inference of structured random forests (SRF) (§III-C) to detect part affordances in real-time. In contrast to previous works that require accurate metric models [2] or predict attributes for segmented objects [35], we show that local geometric primitives are sufficient for pixel accurate functionality detection compared to those discovered via deep learning (which returns only a bounding box) [25], resulting in a more efficient and simple implementation suitable for robotic applications. Finally, we also demonstrate the robustness of the approaches in challenging real-world situations containing clutter, occlusions and viewpoint changes which were not explored in prior works. We first describe in the next section the features used in our approach, derived from a combination of 2D and 2.5D information, that allow us to capture the local geometry of the surface for affordance association. We then detail the two approaches for learning this association from these features.

A. Robust Geometric Features

The key hypothesis of this work is that shape and geometry are physically grounded qualities which are deeply tied to the affordances of a tool part. When characterizing geometric qualities of a part, it is important that the features we compute are robust to variations, such as changes in

[1] http://www.umiacs.umd.edu/~amyers/part_affordance/

viewpoint. At the same time, we would like to gain insight into the influence of basic geometric measures. Therefore, we leverage simple geometric features, such as surface normals and curvature, to learn the relationship between geometry and part affordance. In order to detect affordances for a variety of tools in cluttered scenes with occlusions, we derive the following local geometric features from small $N \times N$ RGB-D input patches:

1) Depth Features: We first apply smoothing and interpolation operators to reduce noise and missing depth values. Then, we remove the mean from the patch to gain robustness to absolute changes in depth. These patches are used directly by HMP to learn hierarchical sparse code dictionaries. In the first layer, HMP captures primitive structures such as depth edges at various orientations, and higher layers encode increasingly abstract representations [3]. To provide comparable depth edge information to the SRF, we compute histograms over depth gradients (HoG-Depth). Similar to the 2D Histogram of Gradients (HoG) image descriptor [7], we compute gradients on the depth image and quantize them into four orientations to create a compact histogram feature.

2) Surface normals (SNorm): We use the depth camera's intrinsic parameters to recover the 3D point cloud, from which we can estimate 3D surface normals. As with the depth, we remove the patch mean during feature learning, to make the representation more robust to changes in viewpoint.

3) Principle curvatures (PCurv): The principle curvatures [8] are an extrinsic invariant of the local patch geometry, and are independent of viewpoint. The principal curvatures $(\kappa_1, \kappa_2), \kappa_1 > \kappa_2$ characterize how the surface bends in different directions.

4) Shape-index and curvedness (SI+CV): The shape index (SI) and curvedness (CV) measures were introduced by Koenderink et al. [20] to characterize human perception of shape. These measures, which are derived from (κ_1, κ_2), are also viewpoint invariant and are defined as

$$SI = -\frac{2}{\pi}\arctan\left(\frac{\kappa_1 + \kappa_2}{\kappa_1 - \kappa_2}\right), CV = \sqrt{\frac{\kappa_1^2 + \kappa_2^2}{2}} \qquad (1)$$

SI and CV are continuous in the range $[-1, +1]$, where the shape index captures the type of local shape (elliptic, parabolic, etc.) and the curvedness its perceived strength.

B. Superpixel Hierarchical Matching Pursuit

We first propose a superpixel based approach to affordance detection following the work of [27]. Although 2D image segmentation in general is a challenging problem in computer vision, recent work has shown that incorporating depth data produces more coherent boundaries that adhere to depth discontinuities not apparent in color images [31] [28]. Usually, great care must be taken to find segments that have not oversegmented or undersegmented an object of interest [16]. However, in our approach we consider tools composed of several parts, each formed by a collection of surfaces, so oversegmentation is advantageous.

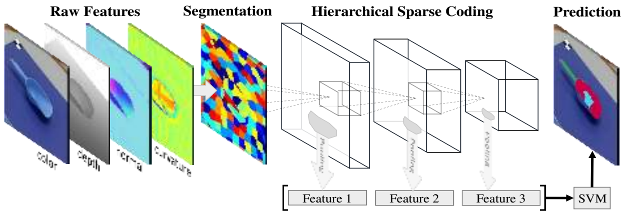

Fig. 2: Affordance detection using S-HMP. An RGB-D image is segmented into superpixels, where each segment serves as a candidate part surface (left). For each superpixel, hierarchical sparse codes are extracted from geometric features such as depth, normal, and curvature information (middle). Superpixels' codes are pooled and then classified using a linear SVM to produce the final predictions for each affordance (right).

Given an RGB-D image, we use a modified version of the SLIC algorithm [1], incorporating depth and surface normal information, to segment objects in the RGB-D image into small surface fragments. Using multiple features for segmentation is important, since parts with different affordances are often connected and share some properties. For each superpixel, we use HMP to compute hierarchical sparse codes from each of the different geometric measures (Depth, SNorm, PCurv, and SI+CV).

HMP [3] is a hierarchical sparse coding method that learns feature hierarchies called *paths*. A path has a unique architecture which captures information at varying scales and abstractions, where in each layer of the hierarchy the input is encoded by sparse coding and undergoes a max-pooling operation. Specifically, at each layer we learn a dictionary D of size m such that the n samples in data matrix Y can be represented by a sparse linear combination X of dictionary entries,

$$\begin{aligned} &\min_{D,X} \|Y - DX\|_F^2 \\ &s.t. \ \forall m, \|\mathbf{d}_m\|_2 = 1 \text{ and } \forall n, \|\mathbf{x}_n\|_0 \leq S \end{aligned} \qquad (2)$$

where $\|\cdot\|_F$ and $\|\cdot\|_0$ denote the Frobenius norm and L_0 norm respectively, S is the sparsity regularization parameter. The dictionary bases $\mathbf{d}_m$ are constrained to have a unit norm, and a sample's sparse coefficients $\mathbf{x}_n$ must have no more than S non-zero values. Given a learned dictionary, an image patch can be represented by its coefficients or sparse codes. In previous works, on image attribute recognition [35] or image classification [4], these codes were max-pooled over the whole image or over an image pyramid. However, we max-pool HMP features within each superpixel, which yields a feature vector for each surface. These features can be classified with a linear SVM, thereby providing a prediction of each affordance for each segment. The proposed framework is illustrated in figure 2. In our experiments we use features from two-layer and three-layer architectures, which capture features at different scales and abstractions. Additional details and parameters are provided in the publicly available code.

C. Structured Random Forest

The random forest (RF), introduced by [15], is an ensemble learning technique that combines K decision trees, $(T_1, \cdots, T_K)$, trained over random permutations of the data

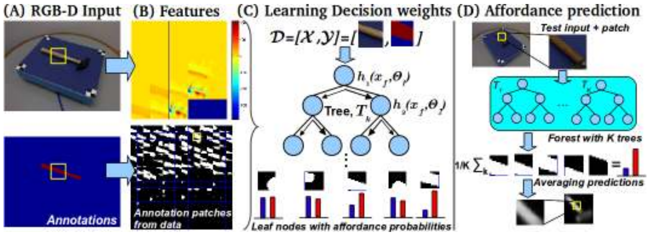
(A) RGB-D Input
(B) Features
(C) Learning Decision weights
(D) Affordance prediction
$\mathcal{D}=[\mathcal{X},\mathcal{Y}]=[$, $]$
$h_1(x_f,\theta_1)$
$h_2(x_f,\theta_2)$
Tree, T_k
Annotations
Annotation patches from data
Leaf nodes with affordance probabilities
Test input + patch
T_1
T_k
Forest with K trees
$1/K\sum_k$
Averaging predictions

Fig. 3: Affordance detection using SRF. (A) Input image with example patch highlighted. (B) Features extracted from each patch (top) and sampled annotation patches from data (below). (C) Training different patches, $\mathcal{X}$ with corresponding binary affordance annotations, $\mathcal{Y}$, learns the optimal θ_j at each split node. The leaf nodes store per pixel confidence scores for each $\mathcal{Y}$ encountered. (D) During inference, a test patch is assigned to a leaf node that contains affordance prediction. Averaging the predictions over the K trees produces an affordance confidence score per pixel.

to prevent overfitting. The output of the model can either be a class label (for multilabel classification) or a continuous value (for regression). The main advantage of RFs is that inference is extremely efficient [6], since data only needs to be passed through several binary decision functions. Due to their speed and flexibility, RFs have been widely applied in both computer vision and robotics problems.

In this work, we propose the second approach using a *structured* random forest (SRF), an extension of the standard RF that imposes structured constraints on the input and output. This enables the SRF to learn more expressive information, while still retaining all the inherent advantages of standard RFs. SRFs was first used by [21] to impose spatial constraints for scene segmentation and was recently extended by [9] for 2D edge detection. Different from these previous works, we impose here a novel structure that relates affordances to the local patch geometry and shape. To this end, we train a SRF that takes as input $\mathcal{X}$, features from local $N \times N$ patches described in §III-A with pixel accurate annotations of the target affordance, $\mathcal{Y}$ (Fig. 3 (B)). The annotations impose the expected spatial structure of how the affordance should appear in the final prediction. For the j^{th} split (internal) node, we train a binary decision function $h(x, \theta_j) \in \{0, 1\}$ over random subsets, $x \in \mathcal{X}$, of the input features so that the parameters $\theta_j = (f, \rho)$ send $x(f)$ (where f is the feature dimension for each feature described in §III-A) to the left child when $h(\cdot) = 1$ if $[x(f) < \rho]$ and to the right child otherwise. The decision threshold, ρ, is obtained by maximizing a standard information gain criterion G_j over $\mathcal{D}_j \subset \mathcal{X} \times \mathcal{Y}$, the features and annotations:

$$G_j = H(\mathcal{D}_j) - \sum_{c \in \{L,R\}} \frac{|\mathcal{D}_j^c|}{|\mathcal{D}_j|} H(\mathcal{D}_j^c) \tag{3}$$

where $D_j^c, c \in \{L, R\}$ indicates the portion of the data that is split by ρ into the left and right child nodes respectively. We use here the Gini impurity measure: $H(\mathcal{D}_j) = \sum_y p_y(1-p_y)$ with p_y denoting the proportion of features in $\mathcal{D}_j$ with ownership label $y \in \mathcal{Y}$. Eq. (3) is computed via an intermediate mapping $\Pi : \mathcal{Y} \mapsto \mathcal{L}$ of structured affordance labels into discrete labels $l \in \mathcal{L}$ following [9]. To determine Π, we first cluster via k-means random annotation patches that have the same affordance labels and select the largest $|\mathcal{L}|$ cluster centers. We repeat the training procedure until a maximum tree depth, D_t, is reached and we store at the leaf nodes per pixel confidence scores for each affordance annotation patch encountered during training. (Fig. 3 (C)). Each tree in the SRF therefore learns jointly, the 2D spatial structure *together* with the 2.5D features that describe the affordance within a patch. Inference using the trained SRF is extremely simple. Given a forest of K trees and a testing patch with extracted features, the learned decision thresholds in each split node will send the patch to a leaf node that contains the predicted affordance labeling and confidence scores. We then average all K predictions for the final prediction (Fig. 3 (D)).

In our implementation, we train a SRF with $K = 8$ trees with a maximum training depth of $D_t = 64$. We use patches of size $N = 16$ and we set $|\mathcal{L}| = 10$ cluster centers for Π. Training over the entire affordance RGB-D dataset (§IV-A) in parallel with an average of 5000 RGB-D images per split takes around 20 minutes on a 16 core Xeon 2.9GHz machine with 128GB of ram. Inference for a single RGB-D image of size (640×480) (height, width), takes an average of 0.1s which includes the time for feature extraction.

IV. Experiments

We first describe in §IV-A the affordance dataset that we introduce to evaluate the proposed approaches. We present the evaluation metrics used for all experiments in §IV-B. We also detail in §IV-C how we apply our approaches to the more common task of predicting grasping locations, in order to compare with the deep-learning approach of [25]. We present and discuss the results of our experiments in §V.

A. *RGB-D Part Affordance Dataset*

To investigate the problem of localizing and identifying affordance, we propose a new RGB-D Part Affordance Dataset which focuses on everyday tools and the affordances of their parts. We consider tool *parts* corresponding to a collection of *surfaces* with multiple affordances. We define each surface's effective affordances by the way it comes in contact with the objects it affects. For example, a coffee mug has two affordance parts, the inner surface and the outer surface. The inner surface of a mug has the effective affordance "contain", because it comes in contact with the liquid that is contained. The surface of the mug's handle has the affordance "grasp" as it can be tightly held by a hand or robot gripper. The dataset provides pixel-level affordance labels for 105 kitchen, workshop, and garden tools. The tools were collected from 17 different categories covering seven affordances which are summarized in Table. I.

Each affordance is represented by objects from a variety of categories with different appearances. Additionally, since it is likely that object parts may have *multiple* affordances, we engaged several human annotators to *rank* how close affordances are with respect to the essential affordance category, while allowing for ties. This allows us to determine, on an ordinal scale, how well the affordance detector generalizes

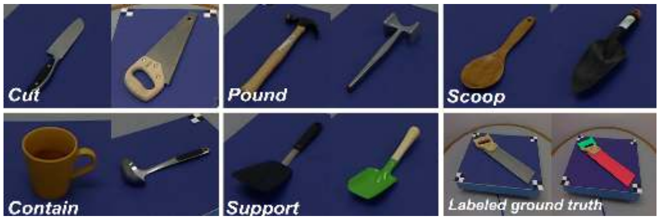

Fig. 4: Sample objects from the RGB-D Part Affordance Dataset. (Lower-right) An example of a full frame image with hand-labeled ground truth. The ground truth labels include rankings for multiple affordances.

Affordance	Description
`grasp`	Can be enclosed by a hand for manipulation (handles).
`cut`	Used for separating another object (the blade of a knife).
`scoop`	A curved surface with a mouth for gathering soft material (trowels).
`contain`	With deep cavities to hold liquid (the inside of bowls).
`pound`	Used for striking other objects.(the head of a hammer).
`support`	Flat parts that can hold loose material (turners/spatulas).
`wrap-grasp`	Can be held with the hand and palm (the outside of a cup).

TABLE I: Description of the seven affordance labels.

to related affordances which is important when novel objects are observed. For example, parts with the affordance "cut" are found in kitchen knives, workshop saws, and garden shears. Examples are shown in Fig. 4.

While there are several RGB-D object datasets, most are designed for instance and category level object recognition [23], attribute learning [35] or for specific robotic gripping locations [25]. In addition to testing with tools from a known category, the dataset is designed for evaluating part affordance identification for objects from completely novel categories. To our knowledge this is the first dataset specifically designed for robots to identify and localize part affordances from RGB-D data.

Data was collected using a Kinect sensor, which records RGB and depth images at a resolution of 640×480 pixels. Since many of the parts we want to capture are small, we collected data at the minimum distance required for accurate depth readings, approximately 0.8 meters. We recorded each tool on a revolving turntable to collect images covering a full 360° range of views. On average, approximately 300 frames are captured for each tool, producing more than 30,000 RGB-D image pairs. Of these, more than 10,000 images have pixel-level ground truth affordance labels. In addition, we supplement the dataset with three sequences of around 1000 RGB-D frames, each collected by a mobile robot observing novel tools in clutter under changing viewpoints. Example frames are shown in Fig. 5 (left).

B. Evaluation Metrics

We use three evaluation metrics to provide different perspectives on the performance of our approaches over the RGB-D Part Affordance dataset. The proposed approaches output a probability map over the image for each affordance, which can be evaluated against ground truth labels to fairly compare their performance. First, we use the *Weighted F-Measure*, F_β^w, introduced recently by Margolin et al. [26] to evaluate saliency maps with continuous valued responses against binary valued ground-truths. F_β^w is an extension of the well-known F-measure F_β[2]:

$$F_\beta^w = (1+\beta^2)\frac{Pr^w.Rc^w}{\beta^2.Pr^w + Rc^w}, \text{with } \beta = 1 \qquad (4)$$

where Pr^w and Rc^w are *weighted* versions of the standard precision $Pr = \frac{TP}{TP+FP}$and recall $Rc = \frac{TP}{TP+FN}$ measures. Here, TP, TN, FP, FN refer to true positives, true negatives, false positives and false negatives respectively. The key insight from [26] is to extend the standard precision and recall measures with weights derived by comparing the binary ground-truth and the continuous valued responses in order to reduce biases inherent in the standard measures. To do this, the authors proposed weights that measure the dependency of foreground pixels (pixels clustered together near the ground-truth are weighted higher), and assign lower weights to pixels far from the ground-truth.

Since the ground-truth in the RGB-D Affordance dataset provides rankings across multiple affordances, for a second measure we define a *rank* weighted F_β^w,

$$R_\beta^w = \sum_r w_r.F_\beta^w(r), \text{with} \sum_r w_r = 1 \qquad (5)$$

that sums weighted $F_\beta^w(r)$ over their corresponding r ranked affordances. The ranked weights w_r are chosen so that the top ranked affordance is given the most weight, followed by the secondary affordance and so on. This allows us to capture if the detector is generalizing across multiple affordances appropriately. Note that when we impose $w_1 = 1$, (5) reduces to (4), where we consider only the top ranked affordance.

Finally, we use a third measure to evaluate whether multiple affordance predictions agree with the ground-truth rankings. We rank the continuous affordance predictions at each pixel, and compute the *ranked* correlation score, Kendall's $\tau_k \in [-1, 1]$ [18]. τ_k approaches 1 as the predicted ranks agree more closely with the ground-truth, but nears -1 as the ranks are reversed. We report $\overline{\tau}_k \in [-1, 1]$, the average τ_k of all pixels over the test images.

C. Cornell Grasping Dataset Comparison

In addition to the RGB-D Part Affordance Dataset, we applied our approaches to a more common, but related robotic task of determining where to grasp (a specific affordance). We used the recently introduced Cornell Grasping Dataset of Lenz et al. [25] to compare against their deep-learning method and validate the effectiveness of our approaches. The dataset contains 1035 RGB-D images of 280 graspable objects, where objects are captured from a small discrete number of viewpoints. Each image contains a single object, and is annotated with a set of rectangles indicating good or bad graspable locations. Following the testing procedure in [25], we averaged results from 5 random splits, and report both recognition accuracy and detection accuracy. For detection, we report the point-wise metric following [25] and

[2] The F-measure with $\beta = 1$ is defined by the harmonic mean of the precision and recall values: $F_\beta = (1+\beta^2).\frac{Pr.Rc}{\beta^2.Pr+Rc}$ and is used as a measure of the accuracy of the Pr and Rc scores. β is a positive weight that gives preferences to either Rc ($\beta > 1$) or Pr ($\beta < 1$).

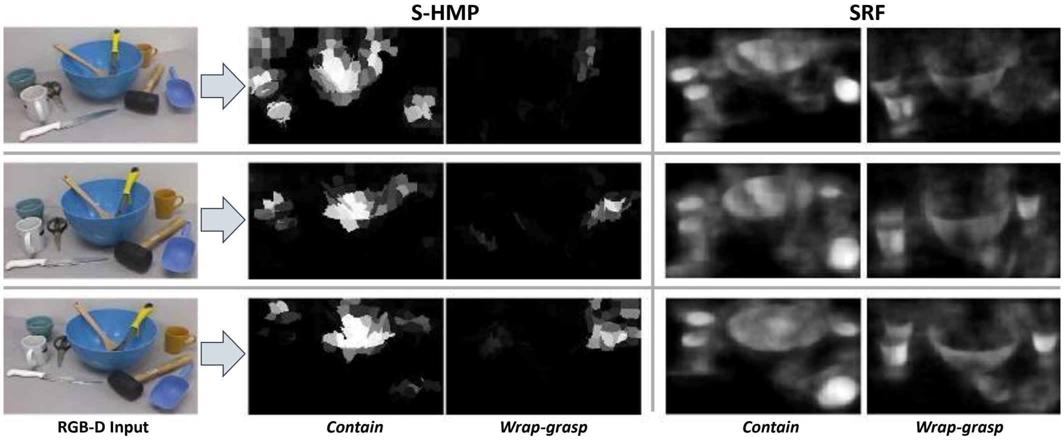

Fig. 5: Results of affordance detection across three different input RGB-D frames (left) using S-HMP (middle) and SRF (right) over the cluttered sequence: two target affordances per method – `contain` (l) and `wrap-grasp` (r). Brighter means higher probability of the target affordance.

[30], which considers the detection a success if it is within some distance from at least one ground-truth rectangle center. In order to use S-HMP in this setting, we treat the candidate rectangles as superpixel segments, and perform max-pooling over the rectangle to make a prediction. To obtain structured labels for the SRF, we estimated the ground-truth annotations of graspable regions by first applying a mask obtained over all graspable rectangles followed by a edge detection and hole filling operation (Fig. 6). We trained S-HMP and SRF using the same parameters used in other experiments.

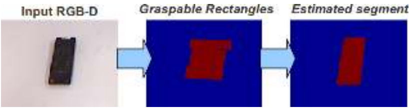

Fig. 6: Estimating pixel accurate annotations from the Cornell Grasping Dataset. (Left) Input RGB image. (Middle) Overlay of several graspable rectangles. (Right) Edge detection and hole filling produces a pixel accurate segment.

V. Results

We report results that demonstrate the performance of our approach using the proposed metrics described above: $(F^w_\beta, R^w_\beta, \overline{\tau}_k)$ for affordance detectors trained using S-HMP and SRF. We used the same train/test splits for both methods, and report averaged results over random splits of the RGB-D Affordance Dataset from [27]. We used the features described in §III-A for a fair comparison. Table. IIa (left) summarizes the two detectors' performance over the seven affordance labels considered.

From the results, we can see that S-HMP consistently outperforms SRF in all three evaluation metrics. The difference is most significant using the F^w_β measure, which shows that the sparse codes obtained by S-HMP are able to distinguish the top ranked affordance class much better than SRF, which tends to produce weaker responses across multiple affordance categories. This is not surprising since, unlike S-HMP which learns a hierarchy of *new* features, SRF only extracts the most discriminative combination of the input features. In the sections that follow, we describe ablation experiments that demonstrate the contribution of geometric features and how they help in real-world scenarios with clutter, occlusions and viewpoint changes.

1) Ablation comparisons: We performed a series of feature ablations to demonstrate the contribution of each feature type in improving the results reported above. Table. IIb shows the influence of additional features over the baseline smoothed and de-meaned depth features, denoted as `Depth`, with respect to the F^w_β measure.

We see that the S-HMP baseline performs very well, and by learning multiple layers of features with increasing invariance and abstraction, S-HMP is able to extract discriminative information. Consequently, additional features provide better but diminishing returns on performance, consistent with the results in [27]. Additionally, increasing feature dimensionality can make SVM learning more difficult. Although the full set of features has a slightly lower F^w_β measure, we note that it has the best performance on ranked measures and clutter. The SRF, on the other hand, benefits more from the addition of new features as they introduce more diversity into the random feature subsets used during training (§III-C). Using the full feature set the SRF achieves a large improvement over the ablated counterparts. Interestingly, we notice that although SI+CV are derived from PCurv, they improve the results further. This validates that the shape-index and curvedness measures capture discriminative information not provided directly by the other features. Considering that the results in [27] showed that geometric features significantly outperformed RGB features, we also tested the SRF with several 2D features which achieved much lower performance. For example, using raw RGB-D values gives F^w_β of 0.055 for SRF.

2) Performance in clutter and occlusions: In order to test the performance of the approach in real-world situations containing clutter, occlusions and viewpoint changes, we tested our approach over the clutter subset of the RGB-D Part Affordance Dataset. Table. IIa (right) compares the performance of S-HMP and SRF using the $(F^w_\beta, R^w_\beta, \overline{\tau}_k)$ metrics.

(a) Performance over the RGB-D Affordance Dataset. (Left) Non-cluttered subset and (Right) Cluttered subset.

Affordance	Non-cluttered subset (single objects)		Cluttered subset (multiple objects)	
	S-HMP $(F^w_\beta, R^w_\beta, \overline{\tau}_k)$	SRF $(F^w_\beta, R^w_\beta, \overline{\tau}_k)$	S-HMP $(F^w_\beta, R^w_\beta, \overline{\tau}_k)$	SRF $(F^w_\beta, R^w_\beta, \overline{\tau}_k)$
`grasp`	0.367, 0.149, 0.711	0.314, 0.133, 0.409	0.227, 0.124, 0.583	0.200, 0.122, 0.165
`cut`	0.373, 0.043, 0.831	0.285, 0.033, 0.798	0.160, 0.065, 0.754	0.072, 0.030, 0.724
`scoop`	0.415, 0.046, 0.627	0.412, 0.097, 0.559	0.165, 0.083, 0.519	0.114, 0.106, 0.446
`contain`	0.810, 0.168, 0.814	0.635, 0.142, 0.579	0.437, 0.222, 0.627	0.322, 0.178, 0.316
`pound`	0.643, 0.035, 0.787	0.429, 0.033, 0.801	0.257, 0.079, 0.609	0.072, 0.023, 0.595
`support`	0.524, 0.030, 0.717	0.481, 0.039, 0.724	0.297, 0.049, 0.462	0.098, 0.022, 0.509
`wrap-grasp`	0.767, 0.102, 0.867	0.666, 0.089, 0.821	0.208, 0.109, 0.482	0.156, 0.099, 0.482
Mean	**0.557, 0.082, 0.751**	0.460, 0.081, 0.643	**0.250, 0.105, 0.563**	0.165, 0.083, 0.435

(b) Ablation experiments. $+x$ indicates the amount of change over `Depth`.

Feature Sets	S-HMP F^w_β	SRF F^w_β
`Depth+SNorm+PCurv+SI+CV`	0.557 (+0.018)	0.460 (+0.137)
`Depth+SNorm+PCurv`	0.562 (+0.023)	0.449 (+0.126)
`Depth+SNorm`	0.547 (+0.008)	0.444 (+0.121)
`Depth`	0.539	0.323

(c) Results on the Cornell Grasping Dataset.

Method	r_a %	d_a %
RF	85.3	62.5
SRF	93.5	87.0
SAE [25]	93.7	88.4
S-HMP	**95.2**	**92.0**

TABLE II: Full experimental results. See text for details.

We show in Fig. 5 a series of three frames illustrating the responses of S-HMP and SRF for two specific affordances: `contain` and `wrap-grasp`. Despite changes in viewpoint, the approaches make reasonable predictions, such as correctly predicting the inner surfaces of bowls and cups as `contain`. S-HMP exhibits precisely localized predictions, and SRF demonstrates generalization, such as predicting `wrap-grasp` on the convex surface of the bowl. From Table. IIa (right), we note further that although both S-HMP and SRF's performance did drop under such challenging scenarios, the drop in S-HMP is less than SRF, which indicates that the learned features, unlike those obtained from SRF are far more robust to viewpoint changes and clutter than SRF.

3) Cornell Grasping Dataset comparison: We applied the proposed approaches to the Cornell Grasping Dataset and compared recognition and detection results to those of the Sparse Autoencoder (SAE) with a two-stage structured regularization in [25]. Table. IIc summarizes the recognition accuracy, r_a, and detection accuracy (point-wise), d_a, of the SRF, SAE, and S-HMP methods. In order to highlight the contribution of the structured constraints in the SRF, we trained a standard random forest (RF) with 20 trees over the annotated grasping rectangles in the dataset, using the *same* feature set of the SAE: RGB + Depth + SNorms.

We note first that using the baseline feature set used in SAE with a standard RF results only in mediocre performance. By adding the structured constraints and the proposed robust features, the SRF is able to achieve recognition and detection performances comparable to the deep learning based SAE. S-HMP outperforms the other approaches by a large margin, achieving state-of-the-art performance for this dataset. It is important to note however, that the SRF provides very reasonable predictions of graspable locations with pixel-wise accuracy (Fig. 7), within a *fraction* of the time needed for inference using SAE (30s) vs. 0.1s in SRF. Such real-time performance is crucial for practical robotics applications and we show in the supplementary video an example of real-time detection over the cluttered RGB-D Affordance Dataset.

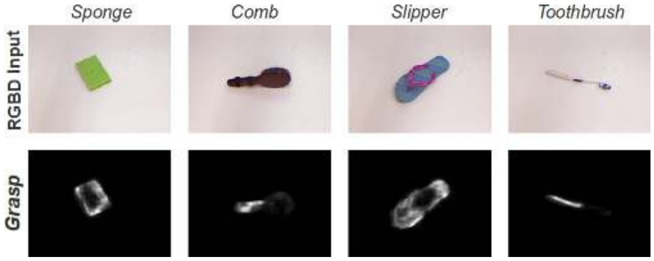

Fig. 7: Grasping locations predicted by SRF. (Top) Input RGB-D images for four example objects. (Bottom) Predicted graspable locations. Notice the large difference in shape of the graspable regions. Brighter means higher probability.

VI. Conclusion

In this paper, we have presented two methods for associating affordances with local shape and geometry information. These methods localize and identify multiple affordances of tool parts, providing functional information that can be used by a robot. S-HMP provides accurate results at a high computational cost, while SRF gives reasonable predictions in real-time. We have also demonstrated the importance of geometry for affordance identification, showing the importance of robust geometric features. We also validated our approaches on an existing dataset, and achieve state-of-the-art results. Finally, we introduced a new RGB-D Part Affordance Dataset with ranked affordance labels for 3 scenes and 105 objects which will be made publicly available for further research.

The work opens up exciting new research directions for recognizing objects in general. Firstly, we plan to study and enforce stronger invariants for the features to handle even more challenging situations. Secondly, we intend to explore the detection of *material* properties, which is an important function of affordance prediction: either via visual methods or haptics. Finally, the approaches described here will be implemented onto a robot with manipulators to test the accuracy of the predictions in real manipulative tasks.

ACKNOWLEDGMENT

This research was funded by the support of the European Union under the Cognitive Systems program (project POET-ICON++), the National Science Foundation under INSPIRE grant SMA 1248056, and by DARPA through U.S. Army grant W911NF-14-1-0384 under the Project: Shared Perception, Cognition and Reasoning for Autonomy.

REFERENCES

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012.

[2] A. Aldoma, F. Tombari, and M. Vincze. Supervised learning of hidden and non-hidden 0-order affordances and detection in real scenes. *Proc. IEEE Int'l Conf. on Robotics and Automation*, pages 1732–1739, 2012.

[3] L. Bo, X. Ren, and D. Fox. Unsupervised feature learning for rgb-d based object recognition. *Int'l Symp. on Experimental Robotics*, 2012.

[4] L. Bo, X. Ren, and D. Fox. Multipath sparse coding using hierarchical matching pursuit. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 660–667, 2013.

[5] J. Bohg and D. Kragic. Grasping familiar objects using shape context. *Int. Conf. on Advanced Robotics*, pages 1–6, 2009.

[6] A. Criminisi and J. Shotton. *Decision Forests for Computer Vision and Medical Image Analysis*. Springer, February 2013.

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 886–893, 2005.

[8] M. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, 1976.

[9] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. *Proc. Int'l Conf. on Computer Vision*, pages 1841–1848, 2013.

[10] V. Ferrari and A. Zisserman. Learning visual attributes. In *Advances in Neural Information Processing Systems*, 2007.

[11] J. J. Gibson. *The theory of affordances*. Hilldale, USA, 1977.

[12] H. Grabner, J. Gall, and L. Van Gool. What makes a chair a chair? *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1529–1536, 2011.

[13] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *Proc. European Conf. on Computer Vision*, pages 297–312, 2014.

[14] T. Hermans, F. Li, J. M. Rehg, and A. F. Bobick. Learning contact locations for pushing and orienting unknown objects. In *Proc. IEEE Int'l Conf. on Humanoid Robots*, 2013.

[15] T. K. Ho. Random decision forests. In *Proc. IEEE Int'l Conf. on Document Analysis and Recognition*, volume 1, pages 278–282, 1995.

[16] A. Karpathy, S. Miller, and L. Fei-Fei. Object discovery in 3d scenes via shape analysis. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, 2013.

[17] C. C. Kemp and A. Edsinger. Robot manipulation of human tools: Autonomous detection and control of task relevant features. In *Proc. Intl. Conf. on Development and Learning*, 2006.

[18] M. G. Kendall. *Rank correlation methods.* Griffin, 1948.

[19] H. Kjellström, J. Romero, and D. Kragić. Visual object-action recognition: Inferring object affordances from human demonstration. *Computer Vision and Image Understanding*, 115(1):81–90, 2011.

[20] J. J. Koenderink and A. J. van Doorn. Surface shape and curvature scales. *Image and vision computing*, 10(8):557–564, 1992.

[21] P. Kontschieder, S. R. Bulo, H. Bischof, and M. Pelillo. Structured class-labels in random forests for semantic image labelling. In *Proc. Int'l Conf. on Computer Vision*, pages 2190–2197, 2011.

[22] H. S. Koppula, R. Gupta, and A. Saxena. Learning human activities and object affordances from rgb-d videos. *Int'l J. of Robotics Research*, 32(8):951–970, 2013.

[23] K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, 2011.

[24] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 951–958, 2009.

[25] I. Lenz, H. Lee, and A. Saxena. Deep learning for detecting robotic grasps. *Int'l J. of Robotics Research*, 2014.

[26] R. Margolin, L. Zelnik-Manor, and A. Tal. How to evaluate foreground maps? In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 248–255, 2014.

[27] A. Myers, A. Kanazawa, C. Fermüller, and Y. Aloimonos. Affordance of object parts from geometric features. In *Proc. of Robotics: Science and Systems RGB-D Workshop*, 2014.

[28] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *Proc. European Conf. on Computer Vision*, pages 746–760, 2012.

[29] D. Parikh and K. Grauman. Relative attributes. *Proc. Int'l Conf. on Computer Vision*, pages 503–510, 2011.

[30] A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic grasping of novel objects using vision. *Int'l J. of Robotics Research*, 27(2):157–173, 2008.

[31] N. Silberman and R. Fergus. Indoor scene segmentation using a structured light sensor. In *Proc. of the Int'l Conf. on Computer Vision Workshop on 3D Representation and Recognition*, 2011.

[32] R. Socher, B. Huval, B. Bhat, C. D. Manning, and A. Y. Ng. Convolutional-recursive deep learning for 3d object classification. In *Advances in Neural Information Processing Systems*, 2012.

[33] L. Stark and K. Bowyer. Function-based generic recognition for multiple object categories. *CVGIP: Image Understanding*, 59(1):1–21, 1994.

[34] M. Stark, P. Lies, M. Zillich, J. Wyatt, and B. Schiele. Functional object class detection based on learned affordance cues. In *Computer Vision Systems*, pages 435–444. Springer, 2008.

[35] Y. Sun, L. Bo, and D. Fox. Attribute based object identification. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, pages 2096–2103, 2013.

[36] X. Yu and Y. Aloimonos. Attribute-based transfer learning for object categorization with zero/one training example. In *Proc. European Conf. on Computer Vision*, pages 127–140, 2010.