One-shot Learning for Task-oriented Grasping

Valerija Holomjova, Andrew Starkey, Bruno Yun, Pascal Meißner

One-shot Learning for Task-oriented Grasping

Valerija Holomjova[1*], Andrew J. Starkey[1], Bruno Yun[2], Pascal Meißner[3]

***Abstract*— Task-oriented grasping models aim to predict a suitable grasp pose on an object to fulfill a task. These systems have limited generalization capabilities to new tasks, but have shown the ability to generalize to novel objects by recognizing the physical properties of objects that can be associated with an action (i.e. affordances). However, this object generalization often comes at the cost of being unable to recognize the object category being grasped, which could lead to unpredictable or risky behaviors, especially within unconstrained environments. This paper overcomes these generalization limitations by exploring one-shot learning techniques to develop a task-oriented grasping solution that can leverage explicit knowledge defined in a database to implicitly generalize to new objects and tasks. We propose the One-shot Task-oriented Grasping (OS-TOG) framework, composed of four sub-models, that uses a database of objects and tasks to identify suitable task-oriented grasps on a specified object from an image scene. In physical experiments with novel objects, OS-TOG recognizes 69.4% of detected objects correctly and predicts suitable task-oriented grasps with 82.3% accuracy, having a physical grasp success rate of 82.3%. Code and models will be released upon publication.**

***Index Terms*— Deep Learning in Grasping and Manipulation, Grasping, Computer Vision for Automation, Recognition**

I. INTRODUCTION

Task-oriented Grasping (TOG) involves finding a grasp pose on an object that enables the completion of a task [6, 8]. For instance, grasping the handle of a mug to *pour* out its contents. TOG is a vital preliminary step to accomplishing manipulations required by robotic grasping systems used for assistive robotics (e.g. doing household chores) or assembly tasks [29]. This ability to understand and interact with surrounding objects enables robotic manipulators to operate in unconstrained environments without human intervention.

Creating a dataset with sufficient coverage of the tasks and objects present in the real world to train TOG models is currently unfeasible [12], which encourages TOG solutions that can generalize to new object categories or tasks. [6, 15, 16] show capabilities of generalizing to novel objects by predicting and leveraging affordances [7, 22], which are regions of an object that represent a functional interaction (e.g. cut, pour, contain). By mapping relationships between affordances and tasks, the robotic system can identify task-suitable grasping regions. For instance, if the robotic manipulator was given a *handover* task, it could grasp the hammer by the `pound` affordance, leaving the `grasp` region to be safely retrieved by a human. However, these solutions are currently unable to recognize the object category being grasped and cannot generalize to novel affordances. The inability to recognize objects could lead to unpredictable or risky actions when working in unconstrained environments, such as an assistive cooking robot grasping the handle of a knife instead of a spatula. Incorporating standard object detectors [10, 23] could provide a means of object recognition, but these are also limited to recognizing object categories they were trained on.

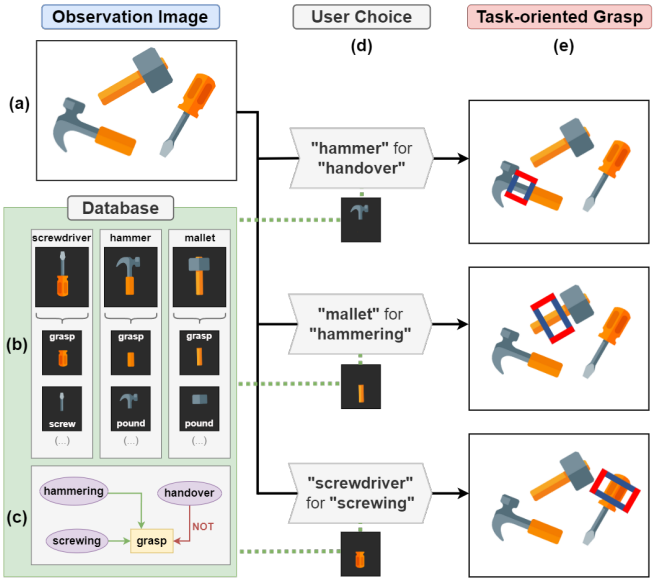

Fig. 1: Given an observation image **(a)**, OS-TOG uses a database containing information on objects **(b)** and tasks **(c)** to predict a task-oriented grasp **(e)** for a user-specified object and task **(d)**. The object database **(b)** contains a list of objects annotated with their labeled affordances. The task database **(c)** maps relationships between tasks and affordances to determine task-suitable regions that should be grasped.

To address the generalization limitations of TOG systems, we explore the use of one-shot learning models, often used for facial recognition and signature verification [3]. These techniques measure the similarity between two images to determine whether they belong to the same object category. We incorporate these models and present the One-shot Task-oriented Grasping (OS-TOG) framework, which leverages a database of objects and task-affordance relations to produce task-oriented grasps for specified objects and tasks from images (Fig. 1). Embedded one-shot learning components allow OS-TOG to generalize to new object categories and tasks implicitly without needing further dataset collection or re-training. OS-TOG is limited to recognizing objects and tasks defined in its database but can generalize to more by adding a single annotation of each new object or task to the database. This requires significantly less labeling and training

This research is funded by a studentship awarded by the School of Engineering at the University of Aberdeen, Scotland UK.

[1] First author (*corresponding author) and second author are with the School of Engineering, University of Aberdeen, Scotland UK {v.holomjova.21, a.starkey}@abdn.ac.uk

[2]Third author is with the School of Natural and Computing Sciences, University of Aberdeen, Scotland UK bruno.yun@abdn.ac.uk

[3]Fourth author is with Würzburg-Schweinfurt Technical University of Applied Sciences (THWS), Germany pascal.meissner@thws.de

effort than creating an entire dataset to re-train a standard TOG model, where combinatorial amounts of examples are needed to cover a range of objects and tasks.

This research aims to evaluate the performance of OS-TOG on TOG and explore the extent can generalize to new objects and tasks. Our contributions can be summarized as three-fold; 1) We present a novel framework for TOG, called OS-TOG, that is capable of generalizing to new objects and tasks. 2) We propose and train suitable sub-models for the interchangeable neural network components in OS-TOG and evaluate them to state-of-the-art in their respective tasks. 3) Experiments with a 7-DoF robotic arm having an RGB-D camera are carried out to demonstrate OS-TOG's ability to perform TOG on previously unseen objects and various tasks.

II. RELATED WORK

Task-oriented grasping is a challenging research area that involves finding a suitable grasp on an object to fulfill a specified task. One challenging aspect is that there is a large variety of tasks and objects in the real world, leading to extensive manual efforts required to create annotated datasets with sufficient coverage across multiple domains.

Over the years, several machine-learning solutions have been proposed to solve task-oriented grasping [6, 8, 12, 14–16, 20, 29, 30], that require large amounts of data. To overcome the dataset limitations in the field, most literature focuses on using alternative methods to generate datasets for training or improving the generalization capabilities of their systems. These data alternative methods include generating synthetic data [6, 15, 30], training in simulated environments [8, 29], or leveraging video footage of human-object interactions [12, 14]. However, these methods often show a drop in performance in real-world scenes or require re-training for new objects and tasks. Certain solutions have demonstrated the ability to generalize to new object categories by learning feature representations of task-relevant geometries within similar objects [6, 8, 29], leveraging semantic knowledge between tasks and objects [20] or segmenting parts of the object with a particular functionality (i.e. affordance) to assist in predictions [15, 16]. Nonetheless, some of these solutions were not designed to work in multi-object scenes with all being unable to recognize the objects they are grasping fully. Hence, we identify a research gap for novel task-oriented grasping solutions that can recognize objects and generalize to new objects and tasks within multi-object scenes with minimal training effort required.

The task of segmenting and labeling parts of objects with functionality is referred to as "affordance segmentation". Machine-learning solutions that predict affordance maps are mostly semantic segmentation models consisting of object detection components. For instance, [22] proposed a CNN-based framework that detects objects and then segments their affordances from RGB images. [7] extends this solution by creating an end-to-end architecture, similar to a Mask R-CNN [10], that recognizes objects and segments affordances in parallel. [4] build upon an object detector and add domain adaptation components to learn from synthetic data and adapt to real-world data. [28] construct an end-to-end autoencoder that learns from human-object interactions. However, these techniques are unable to generalize to new objects and affordances without requiring re-training. To this matter, [17] and [9] demonstrate the use of one-shot learning techniques to find and segment previously unseen affordances without requiring re-training.

One-shot learning is the task of classifying objects from a single or few training examples. The most popular one-shot learning technique is Siamese networks [3], composed of two sub-networks that share the same weights to predict the similarity between two different inputs. [5] use a Mask R-CNN followed by a Siamese network to recognize target objects from cluttered bins in order to be grasped. Alternative one-shot learning methods surpassing the performance of Siamese networks have also been introduced over the years. [27] create a novel two-branched approach for one-shot image segmentation, where one branch generates parameters from the query image which is used by the second branch to produce a segmentation mask from the query image. [19] segments objects from cluttered scenes by segmenting instances, masking their backgrounds, and computing the best match. [31] wins first place in the Amazon Robotics Challenge for categorizing objects in a cluttered bin. Their solution isolates each object through grasping and then matches them to the nearest object in a database using a two-stream CNN-based model. Their system obtains a high recognition rate but is inefficient in settings where you need to retrieve only a specific object category from a multi-object scene. Inspired by the solutions of [5, 9, 17, 31] and capabilities of one-shot learning models, we design a task-oriented grasping framework that recognizes and grasps objects from multi-object scenes with generalization properties to both tasks and objects without the need for re-training.

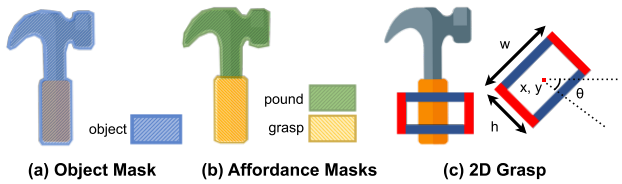

Fig. 2: An example of an object annotated with an object segmentation mask **(a)** and affordance segmentation masks **(b)**, and an example of a grasp pose $g = (x^g, y^g, w^g, h^g, \theta^g)$ on an object, with center co-ordinates (x^g, y^g), gripper opening w^g, gripper size h^g and rotation $\theta^g \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ **(c)**.

III. PROBLEM STATEMENT

Assume a database D containing a set of user-defined tasks D_T and annotated objects D_O. Each object in D_O is represented by an RGB image which is annotated with an object segmentation mask (Fig. 2a) and suitable affordance segmentation masks (Fig. 2b) from a determined set of affordances A. These segmentation masks are binary masks of the RGB image. Each task in D_T is mapped to a suitable affordance $a \in A$ through "a" or "*NOT* a" relations, denoting whether grasping in the region of a will allow the task to be accomplished or not. Given an RGB image of a multi-object scene containing different objects N, database $D = D_T \cup D_O$, target object $o \in (N \cap D_O)$, and a target task

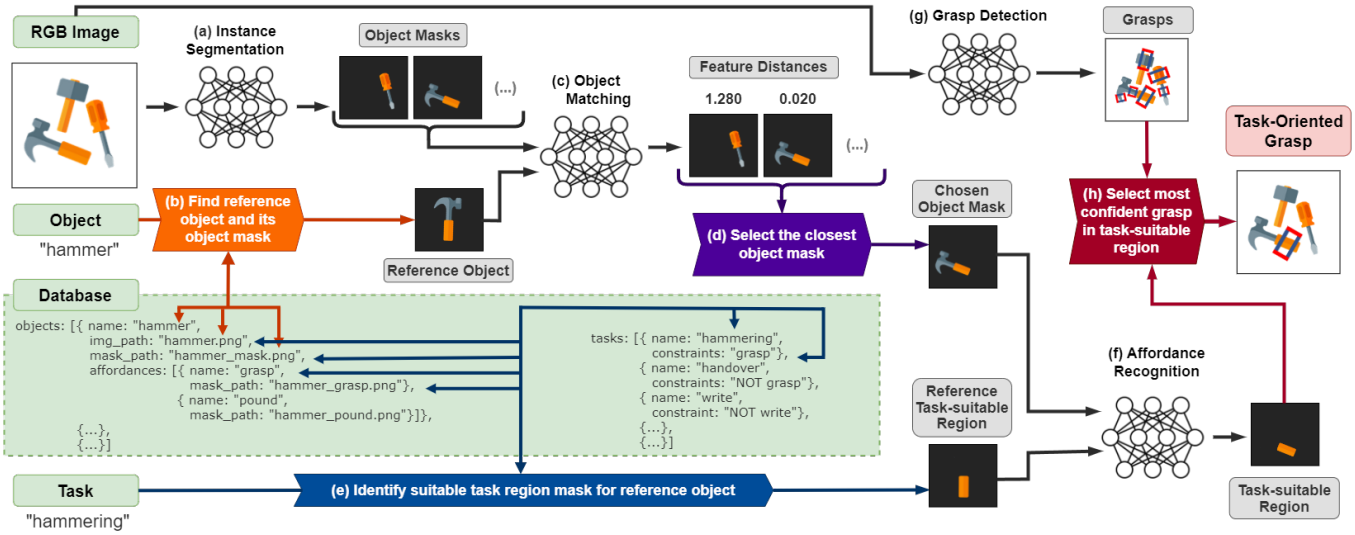

Fig. 3: An illustration of the proposed framework (OS-TOG) for task-oriented grasping. OS-TOG takes as input an RGB image, target object, target task, and database of objects and tasks (green) and outputs a 2D task-oriented grasp on the target object for the target task (red). The four sub-models are represented by grey arrows **(a, c, g, f)**, and the external reasoning components are represented by colored arrows **(b, d, e, h)**.

$t \in D_T$, the objective is to localize and grasp o from the scene in a manner that satisfies the conditions of t.

The target object o is localized by predicting a bounding box $b = (x^b, y^b, w^b, h^b)$ and segmentation map of an object $n \in N$ from the scene s.t. $n = o$. Given a parallel plate gripper, o can be grasped by predicting a 2D task-oriented grasp that can be parameterized as an oriented bounding box $g = (x^g, y^g, w^g, h^g, \theta^g)$ (Fig. 2c). To ensure the grasp is task-suitable, (x^g, y^g) should be in the affordance regions of o that satisfy the relations defined by t in D_T.

IV. ONE-SHOT TASK-ORIENTED GRASPING

OS-TOG is a framework designed for TOG with embedded one-shot learning components that leverage database knowledge (i.e. references) to generalize to new objects and tasks without needing re-training. OS-TOG comprises four neural networks and external reasoning components that process predictions from the sub-models and acquire object or affordance references from the database (Fig. 3). The neural network components of OS-TOG are interchangeable allowing us to select state-of-the-art models for their respective tasks. This section proceeds to describe the neural and reasoning components in further detail.

Instance Segmentation - The first sub-model (Fig. 3a) performs category-agnostic instance segmentation to segment and isolate all N objects in the 640×480 RGB scene image producing N binary masks. Each binary mask is combined with the RGB scene image to create a set of color masks C. We use Mask R-CNN [10] with a ResNet-50 FPN backbone and weights pre-trained on ImageNet [24]. The Mask R-CNN heads predict an object class, a bounding box, and a binary segmentation map for each object. We replace the mask and class predictor heads to predict two object classes; "object" or "background", and train the model using the same multi-task loss function defined in [10].

Object Matching - External reasoning components retrieve a color mask o^c of the target object o from the database by combining its 256×256 RGB image $o^i \in D_O$ and binary mask $o^m \in D_O$ (Fig. 3b). Each predicted color mask $c_i \in C$ is magnified by cropping its bounding box and padding it to a size of 256×256. The magnified predicted masks and target object mask are fed into a one-shot learning model which extracts their embedding vectors and computes the L2 distance between them to determine which c has the smallest distance and is most similar s.t.$\phi = argmin_{c_i \in C}\{d(c_i, o^c)\}$ (Fig. 3d). For the object matching model, we re-implement N-net from [31] in PyTorch as it obtained the highest novel object recognition accuracy.

During training, N-net is comprised of three streams. One stream computes features for a reference object image x^a, and the other two streams compute features for two query object images (positive x^p and negative x^n). x^a shares the same object class as x^p, whereas x^n has a different object class. N-net uses embedding vectors from a frozen ResNet-50 model with pre-trained ImageNet weights for the reference image stream to improve novel object accuracy. This is further improved by using multiple product images for each reference object in training and selecting the nearest one based on L2 distances between features. We replace N-net's original training loss function with standard triplet loss (TL) [26] and use a balanced batch sampler (BBS) after seeing improved accuracies in preliminary experiments. The BBS randomly selects p samples from k object classes in each mini-batch, generating $p \times k$ triplets in each mini-batch. Triplet loss is a metric that minimizes the L2 embedding distance between x^a and x^p, and maximizes the distance between x^a and x^n by a minimum margin α. Given that $f(x)$ is the embedding vector of an image x, triplet loss L_t for a triplet (x_i^a, x_i^p, x_i^n) can be defined as;

$$L_t = max\{d(x_i^a, x_i^p) - d(x_i^a, x_i^n) + \alpha, 0\} \quad (1)$$

$$d(x_i, y_i) = ||f(x_i) - f(y_i)||_2 \quad (2)$$

One-shot Affordance Recognition - OS-TOG retrieves a reference binary affordance mask o^a of o depending on the

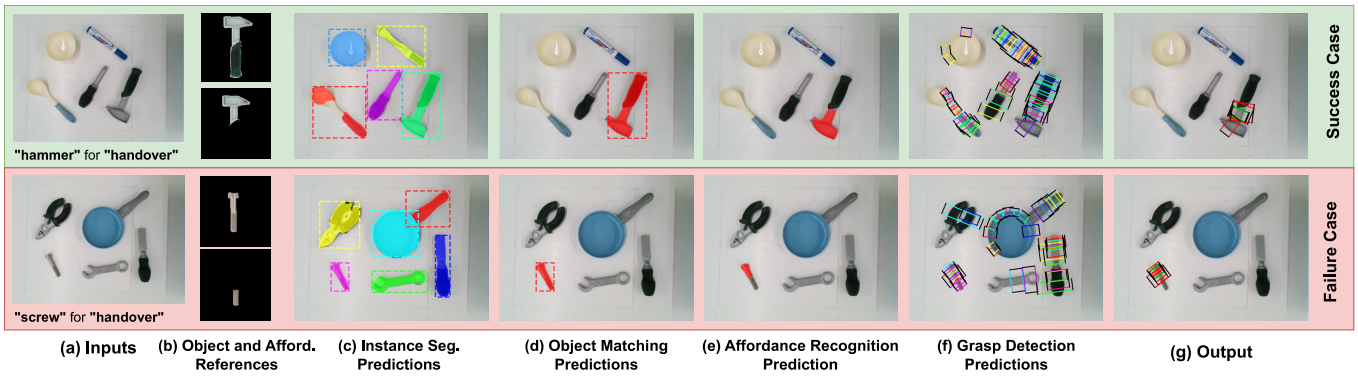

Fig. 4: Predictions made by OS-TOG in physical experiments when given an RGB image and target object and task as input **(a)**. OS-TOG retrieves an object and suitable affordance reference of the target inputs from the database to be used by its one-shot learning models **(b)**. This figure shows predictions of each sub-model **(c-f)** and the final set of task-oriented grasp candidates **(g)**. The green candidate represents the most confident grasp which is executed.

task-affordance relation defined by t in the database (Fig. 3e). For "a" relations, OS-TOG retrieves a binary mask $a^m \in D_O$ of a on the target object o s.t. $o^a = a^m$. For unconstrained relations (e.g. *transport* in Table II), we take the binary mask $o^m \in D_O$ of o s.t. $o^a = o^m$. For "NOT a" relations, we obtain a binary mask s.t. $o^a = o^m - a^m$. The one-shot affordance recognition model takes as input ϕ, o^i, and o^a to produce a binary affordance mask ϕ^m that represents the task-suitable region in ϕ (Fig. 3f). We use the AffCorrs model [9] for one-shot affordance recognition without re-implementation or training as it is unsupervised and the only sub-model in OS-TOG that is not re-implemented or trained. In physical experiments we found that AffCorrs performs significantly better if the orientation of the objects in ϕ, o^i are similar, hence, we rotate $\{o^i, o^a\}$ in 45° intervals and use the pair with the smallest L2 distance between o^i and ϕ.

Grasp Detection - A grasp detection model predicts grasp candidates on the image scene (Fig. 3g). We use our baseline from previous work [11] that uses a Faster R-CNN [23] model with a ResNet-50 FPN backbone and pre-trained ImageNet weights. Faster R-CNN predicts an object class and bounding box for each object in a scene, hence, we replace the object classes it predicts and an orientation class r. We discretize $\theta^g \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ orientation values into $Q = 12$ classes s.t. the set of possible orientation classes is $R = \{r_1, ..., r_Q\}$ with an additional class denoting an invalid grasp and replace the Faster R-CNN class predictor head to predict $Q+1$ classes. The model is trained using the same multi-task loss function defined in [23]. The grasp candidates are filtered to only consider those that have a confidence threshold > 0.5 and (x^g, y^g) lies in the predicted affordance region ϕ^m (Fig. 3h). The most confident grasp is taken giving a task-oriented grasp on object o for task t.

V. EXPERIMENTS AND EVALUATION

OS-TOG was built in PyTorch using Python 3.8. Since there is currently no publicly available gold-standard TOG dataset, the system is evaluated in three separate settings. First, we evaluate each sub-model of OS-TOG that we implemented to the state-of-the-art in their respective tasks (Sec. V-C). Second, we evaluate the performance of OS-TOG in affordance recognition which uses all its sub-model components except for the final grasp model (Sec. V-D). Lastly, the entire framework is evaluated on TOG in physical experiments with random household objects (Sec. V-E).

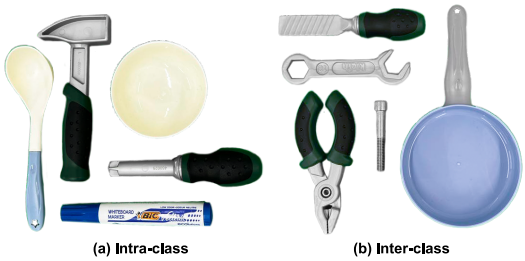

Fig. 5: Objects used for physical grasping experiments.

A. Evaluation Metrics

We adopt standard metrics from literature [1, 9] to evaluate the performance of our models; the Intersection over Union (IoU) score and F^w_β-measure [18] using $\beta = 1$ signifying equal importance to weighted recall and precision. The grasp accuracy score is calculated by classifying a predicted grasp as a success if it has an IoU score greater than 25% with a ground-truth grasp and a θ^g angle difference within 30°. We report the same metrics as [31] for one-shot learning.

B. Datasets

Cornell grasp dataset [13] - contains 1,035 RGB-D images of single-object scenes covering 280 object classes hand-annotated with multiple grasps.

OCID grasp dataset [18] - has 1,763 RGB-D images of multi-object scenes covering 30 object categories annotated with multiple grasps. Each object in the scene is also labeled with a segmentation mask of its object class.

ARC image matching dataset [8] - has over 4,000 images of 61 different object categories against a green screen with matching masked product images in various orientations.

UMD dataset [21] - has 30,000 RGB-D images of single-object scenes containing random household objects of 17 categories and 105 classes, and 7 affordance classes (grasp, cut, scoop, contain, pound, support, and wrap-grasp).

UMD^i dataset [9] - a subset of the UMD dataset tailored for one-shot learning containing only a single instance of each object class from UMD with original annotations kept.

C. Evaluating the sub-models of OS-TOG

Each sub-model is trained and evaluated on datasets used by state-of-the-art models in grasping literature for instance segmentation, grasp detection, and one-shot learning on their reported metrics (Tables III and IV). As mentioned in Section IV, we do not train or evaluate AffCorrs but provide reported metrics from [9] on the UMD^{i} Dataset (Table V). These metrics are not directly comparable to the other baselines which are supervised and evaluated on the full UMD Dataset.

TABLE I: LIST OF OBJECTS AND LABELLED AFFORDANCES

Split	Object	Affordances
INTRA-CLASS	hammer	`grasp,pound`
	spoon	`grasp,contain`
	screwdriver	`grasp,screw`
	marker	`grasp,write`
	bowl	`grasp,contain`
INTER-CLASS	chisel	`grasp,file`
	frying pan	`grasp,fry`
	wrench	`grasp,loosen,unscrew`
	pliers	`grasp,hold`
	screw	`grasp,screw`

TABLE II: LIST OF TASKS AND GRASP CONSTRAINTS

Task Name	Constraints	Affected Objects
transport	-	all
hammering	`grasp`	hammer
handover	NOT `grasp`	all except bowl
filing	NOT `file`	chisel
loosening	NOT `loosen`	wrench
unscrewing	NOT `unscrew`	unscrew
holding	`grasp`	pliers
writing	`grasp`	marker
screwing	NOT `screw`	screwdriver, screw
scooping	NOT `contain`	spoon
frying	NOT `grasp`	frying pan

D. Evaluation OS-TOG on Affordance Recognition

OS-TOG is evaluated on the UMD dataset on two separate data splits; intra-class and inter-class. Intra-class signifies that all object categories are present in both data splits, whereas inter-class signifies that the test set contains exclusive object categories. A database containing a single example of each object class with all possible labeled affordances is built from the dataset. The evaluation procedure begins by iterating through each image in the test set and segmenting the object in the image, then matching it to the nearest reference object in the database. For each ground-truth affordance label in the current scene, a reference affordance mask having the same label is retrieved and used to predict an affordance mask on the scene object. Lastly, we calculate the IoU score and F_β^w-measure between the ground-truth and predicted affordance masks when the scene object is correctly matched (Table V).

E. Evaluation OS-TOG in Physical Experiments

For physical experiments, the grasp detection model was trained on Cornell, and the object matching model was trained on UMD. The instance segmentation model was trained on OCID, single-object UMD scenes, and then 20 multi-object scenes from UMD that we manually annotated. We conduct physical experiments using a 7-DoF robotic arm by Franka Emika equipped with a D415 Intel RealSense camera. The Frankx library [2] was used for motion planning. Experiments are carried out on 10 random objects having at least one affordance. Half of the objects were seen by at least one of the trained sub-models in training (intra-class) (Fig. 5a), and the latter were never seen in training (inter-class) (Fig. 5b). We create a database by annotating a single instance of each object with suitable object masks and affordances (Table I) and create a list of tasks mapped to suitable affordance regions (Table II). Our approach allows us to have multiple affordances on objects even if overlapped. We carry out five trials for each object per task. Table VI shows OS-TOG's ability to segment the object (Obj. Det.), match the detected object to the reference object correctly (Obj. Match.), detect a grasp in the correct affordance region of the correctly matched object (Grasp Det.), and physically succeed in grasping the predicted grasp (Grasp Succ.).

VI. RESULTS

Instance Segmentation and Grasp Detection - As shown by Table III, our trained grasp model and instance segmentation achieve comparable performance to Det Seg [1] on both datasets and significantly outperforms Det Seg on grasp detection in the OCID grasp dataset. This suggests that Faster R-CNN may perform better in multi-object scenes.

TABLE III: GRASP DET. AND INSTANCE SEG. RESULTS

Method	Dataset	Grasp Accuracy (%)	IOU (%)
Det Seg [1]	OCID grasp	89.0	**94.1**
	Cornell	**98.2**	-
Faster R-CNN [23] and Mask R-CNN [10] (ours)	OCID grasp	**98.1**	93.0
	Cornell	96.6	-

Object Matching - Our re-implementation of N-net performs better in all metrics than N-net from [31]. Our model still does not reach the performance of K-net and the Two-stage model from [31] on known object recognition, and mixed object recognition, however, it achieves the highest recognition rate on novel object recognition which is the metric most important to our system.

TABLE IV: OBJECT MATCHING RESULTS ON ARC [31]

Method	K vs N	Known	Novel	Mixed
N-net [31]	69.2	56.8	82.1	64.6
K-net [31]	**93.2**	**99.7**	29.5	78.1
Two-stage K-net + N-net [31]	93.2	93.6	77.5	**88.6**
N-net + TL + BBS (ours)	71.7	75.5	**86.7**	78.7

Affordance Recognition - OS-TOG is able to correctly detect and match objects from the scene 65.4% of the time for inter-class objects and 71.5% for intra-class. When correctly detecting and matching the object, OS-TOG is able to outperform the baseline approaches on nearly all affordance types and achieves an average IoU and F_β^w score of 0.77 and 0.85 for intra-class objects and 0.77 and 0.84 for inter-class objects. The similarity between OS-TOG's results for inter-class and intra-class objects demonstrates the generalization capabilities of the network to new objects.

Physical Experiments - Table VI shows that OS-TOG successfully matched previously seen objects at a rate of 75.0%, and 69.4% for novel object categories when segmented. Most object matching failures are a result of incomplete or noisy segmentation predictions, or objects being too similar in

TABLE V: AFFORDANCE RECOGNITION RESULTS ON UMD [21]

Method	Data Split	Grasp		Cut		Scoop		Contain		Wrap-Grasp		Pound		Support		Total Avg.	
		IoU	F_β^w	IoU	F_β^w	IoU	F_β^w	IoU	F_β^w	IoU	F_β^w	IoU	F_β^w	IoU	F_β^w	IoU	F_β^w
AffordanceNet [7]	Intra-class	-	**0.73**	-	**0.81**	-	0.76	-	0.83	-	0.82	-	0.79	-	0.84	-	0.80
ResNet [25]	Inter-class	0.33	-	0.51	-	0.69	-	0.52	-	**0.85**	-	0.09	-	0.51	-	0.50	-
	Intra-class	**0.71**	-	**0.79**	-	**0.86**	-	0.86	-	**0.84**	-	0.72	-	0.55	-	0.76	-
AffCorrs [9]	Inter-class	0.39	0.41	0.51	0.50	0.62	0.65	0.71	0.75	0.83	0.87	0.72	0.73	0.82	0.79	0.66	0.68
	Intra-class	0.55	0.65	0.72	**0.81**	0.73	0.81	0.82	0.87	0.83	**0.89**	0.78	0.87	**0.82**	**0.87**	0.75	0.82
OS-TOG (ours)	Inter-class	**0.58**	**0.69**	**0.65**	**0.76**	**0.78**	**0.86**	**0.85**	**0.91**	0.80	**0.89**	**0.86**	**0.93**	**0.87**	**0.93**	**0.77**	**0.84**
	Intra-class	0.55	0.66	0.66	0.75	0.82	**0.89**	**0.90**	**0.94**	0.81	**0.89**	**0.84**	**0.91**	0.78	0.86	**0.77**	**0.85**

TABLE VI: PHYSICAL EXPERIMENT RESULTS

	Object	Task	Success Rates (%) Obj. Det.	Obj. Match.	Grasp Det.	Grasp Succ.
INTRA-CLASS	Bowl	*transport*	100.0	100.0	100.0	40.0
	Screwdriver	*transport*	100.0	80.0	100.0	80.0
		screwing	100.0	80.0	100.0	75.0
		handover	100.0	80.0	100.0	50.0
	Spoon	*transport*	100.0	80.0	100.0	50.0
		scooping	100.0	60.0	100.0	100.0
		handover	100.0	20.0	0.0	N/A
	Hammer	*transport*	100.0	60.0	100.0	66.7
		hammering	100.0	60.0	100.0	66.7
		handover	80.0	75.0	100.0	100.0
	Marker	*transport*	100.0	100.0	100.0	100.0
		writing	100.0	100.0	80.0	100.0
		handover	100.0	80.0	75.0	100.0
	Avg. Total		**98.5**	**75.0**	**88.8**	**77.4**
INTER-CLASS	Chisel	*transport*	100.0	60.0	66.7	50.0
		filing	100.0	80.0	100.0	50.0
		handover	100.0	40.0	100.0	00.0
	Frying Pan	*transport*	100.0	100.0	100.0	60.0
		frying	100.0	100.0	0.0	N/A
		handover	100.0	100.0	100.0	100.0
	Pliers	*transport*	100.0	40.0	100.0	100.0
		holding	100.0	40.0	100.0	100.0
		handover	100.0	80.0	100.0	75.0
	Wrench	*transport*	100.0	80.0	100.0	100.0
		unscrewing	100.0	40.0	100.0	100.0
		handover	100.0	40.0	50.0	100.0
		loosening	100.0	60.0	100.0	100.0
	Screw	*transport*	100.0	80.0	100.0	100.0
		screwing	80.0	80.0	75.0	100.0
		handover	80.0	100.0	25.0	100.0
	Avg. Total		**97.5**	**69.4**	**82.3**	**82.3**

color. For instance, the hammer was often mismatched to the screwdriver when the head of the hammer was not properly segmented due to them sharing the same color properties.

The results also show that a task-suitable grasp was successfully predicted at a rate of 88.8% for known objects and 82.3% for novel objects. Task-oriented grasp detection failures are attributed to mis-segmentations, insufficient grasps predicted on the target scene object, and affordance recognition failures. For example, the model failed to segment the correct affordance region in the screw since the thread and head had a very similar shape (Fig. 4). Physical grasp success rates were 77.4% for known objects and 82.3% for novel objects with most failures attributed to objects slipping from the grippers or predicting w^g too small. Note that the physical grasp success rate for the spoon's *handover* task and frying pan's *frying* task is unavailable since it failed to detect any task-suitable grasps in the trials.

VII. CONCLUSION

We present a novel framework called OS-TOG, composed of four sub-models and reasoning components that coordinate to perform task-oriented grasping. By leveraging the properties of one-shot learning models and a database of individually annotated objects and tasks, OS-TOG produces task-oriented grasps on previously unseen objects and tasks from RGB multi-object scenes. Experimentation results showed that OS-TOG is capable of generalizing substantially to new objects and tasks, which is beyond the generalization capabilities of current task-oriented grasping systems. Future work involves improving the performance of each sub-component further.

REFERENCES

[1] Stefan Ainetter et al. "End-to-end Trainable Deep Neural Network for Robotic Grasp Detection and Semantic Segmentation from RGB". In: *ICRA*. IEEE. 2021.

[2] Lars Berscheid. *Frankx: High-Level Motion Library for the Franka Emika Robot*. https://github.com/pantor/frankx. 2013.

[3] Jane Bromley et al. "Signature Verification Using A "Siamese" Time Delay Neural Network". In: *Int. J. Pattern Recognit. Artif. Intell.* (1993).

[4] Fu Jen Chu et al. "Learning Affordance Segmentation for Real-World Robotic Manipulation via Synthetic Images". In: *IEEE RA-L* (Apr. 2019).

[5] Michael Danielczuk et al. "Mechanical Search: Multi-Step Retrieval of a Target Object Occluded by Clutter". In: *ICRA*. IEEE, 2019.

[6] Renaud Detry et al. "Task-oriented Grasping with Semantic and Geometric Scene Understanding". In: *IROS*. IEEE, 2017.

[7] Thanh-Toan Do et al. "AffordanceNet: An End-to-End Deep Learning Approach for Object Affordance Detection". In: *ICRA*. IEEE, 2018.

[8] Kuan Fang et al. "Learning Task-Oriented Grasping for Tool Manipulation from Simulated Self-Supervision". In: *Int. J. Robotics Res.* (2020).

[9] Denis Hadjivelichkov et al. "One-Shot Transfer of Affordance Regions? AffCorrs!" In: *CoRL*. 2022.

[10] Kaiming He et al. "Mask R-CNN". In: *ICCV*. IEEE. 2017.

[11] Valerija Holomjova et al. "Exploring Rotated Object Detection Models for Antipodal Robotic Grasping". In: *UKRAS22*. 2022.

[12] Mia Kokic et al. "Learning Task-Oriented Grasping From Human Activity Datasets". In: *IEEE RA-L* (2020).

[13] Ian Lenz et al. "Deep Learning for Detecting Robotic Grasps". In: *Int. J. Robotics Res.* 4-5 (2015).

[14] Hui Li et al. "Learning Task-Oriented Dexterous Grasping from Human Knowledge". In: *ICRA*. IEEE, 2021.

[15] Yunzhi Lin et al. "Using Synthetic Data and Deep Networks to Recognize Primitive Shapes for Object Grasping". In: *ICRA*. IEEE, 2020.

[16] Weiyu Liu et al. "CAGE: Context-Aware Grasping Engine". In: *ICRA*. IEEE, 2020.

[17] Hongchen Luo et al. "One-Shot Affordance Detection". In: *Int. J. Comput. Vis.* (June 2022).

[18] Ran Margolin et al. "How to Evaluate Foreground Maps?" In: *CVPR*. 2014.

[19] Claudio Michaelis et al. "One-Shot Segmentation in Clutter". In: 2018.

[20] Adithyavairavan Murali et al. "Same Object, Different Grasps: Data and Semantic Knowledge for Task-Oriented Grasping". In: *CoRL*. PMLR, 2020.

[21] Austin Myers et al. "Affordance detection of tool parts from geometric features". In: *ICRA*. IEEE. 2015.

[22] Anh Nguyen et al. "Object-Based Affordances Detection with Convolutional Neural Networks and Dense Conditional Random Fields". In: IEEE, 2017.

[23] Shaoqing Ren et al. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *Adv. Neural Inf. Process. Syst.* 2015.

[24] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: *Int. J. Comput. Vis.* (2015).

[25] Johann Sawatzky et al. "Weakly Supervised Affordance Detection". In: *CVPR*. IEEE Computer Society, 2017.

[26] Florian Schroff et al. "FaceNet: A unified embedding for face recognition and clustering". In: *CVPR*. IEEE, 2015.

[27] Amirreza Shaban et al. "One-Shot Learning for Semantic Segmentation". In: BMVA Press, Sept. 2017.

[28] Spyridon Thermos et al. "A Deep Learning Approach to Object Affordance Segmentation". In: *ICASSP*. IEEE, Apr. 2020.

[29] Bowen Wen et al. "CaTGrasp: Learning Category-Level Task-Relevant Grasping in Clutter from Simulation". In: *ICRA*. IEEE, 2022.

[30] Chenjie Yang et al. "Task-oriented Grasping in Object Stacking Scenes with CRF-based Semantic Model". In: *IROS*. IEEE, 2019.

[31] Andy Zeng et al. "Robotic Pick-and-Place of Novel Objects in Clutter with Multi-Affordance Grasping and Cross-Domain Image Matching". In: *Int. J. Robotics Res.* (Oct. 2022).