

# Center-of-Mass-based Robust Grasp Planning for Unknown Objects Using Tactile-Visual Sensors

Qian Feng<sup>1,3</sup>, Zhaopeng Chen<sup>1\*,2</sup>, Jun Deng<sup>1</sup>, Chunhui Gao<sup>1</sup>,  
 Jianwei Zhang<sup>2</sup>, and Alois Knoll<sup>3</sup>

**Abstract**—An unstable grasp pose can lead to slip, thus an unstable grasp pose can be predicted by slip detection. A regrasp is required afterwards to correct the grasp pose in order to finish the task. In this work, we propose a novel regrasp planner with multi-sensor modules to plan grasp adjustments with the feedback from a slip detector. Then a regrasp planner is trained to estimate the location of center of mass, which helps robots find an optimal grasp pose. The dataset in this work consists of 1025 slip experiments and 1347 regrasps collected by one pair of tactile sensors, an RGB-D camera and one Franka Emika robot arm equipped with joint force/torque sensors. We show that our algorithm can successfully detect and classify the slip for 5 unknown test objects with an accuracy of 76.88% and a regrasp planner increases the grasp success rate by 31.0% compared to the state-of-the-art vision-based grasping algorithm.

## I. INTRODUCTION

Robotic grasping and manipulating of unknown objects bring enormous profit to human society. Leveraging the fast advances in deep learning and computer vision, robotic grasping draws increasing interests from industries, e.g., the Amazon Robotic Challenges [1]. However, to obtain a robust performance, vision based methods have some limitations. First, the currently proposed methods either learned from realistic data [1]–[5] or from synthetic data [6]–[8], only involve object shapes or geometries, ignoring many other aspects such as material properties and object mass. Second, the vision-only method is open-loop without information of contacts with objects, thus the robustness is hard to guarantee.

In consequence, one sensory modality cannot provide enough context to plan a robust grasp. Several novel fusion methods of multi-sensor modules are proposed, to learn to detect object geometries [9], to help robotic grasp planning [10]–[12], to execute manipulation tasks like peg insertion [13]. Here we propose a novel algorithm with fusion of multi-sensor modules for unknown objects.

Meanwhile, grasping and manipulating unknown objects are essential for service robots in the future to execute challenging tasks. For object manipulation, a stable grasp is prerequisite but grasping objects with uncertainties of surface textures [14], [15] and center of mass [16], [17], are still challenging for robotic systems, such as some common tools, e.g., axes and hammers which combine more than one

<sup>1</sup>Agile Robots AG, Gilching Germany. {qian.feng, jun.deng, chunhui.gao}@agile-robots.com

<sup>2</sup>TAMS (Technical Aspects of Multimodal Systems), Department of Informatics, Universität Hamburg

<sup>3</sup>Chair for Robotics and Embedded Systems, Technische Universität München, Germany. knoll, qian.feng @tum.de

\*Corresponding author to provide e-mail: zhaopeng.chen@agile-robots.com

material in one object, are hard to grasp and manipulate because of uneven mass distributions towards simple robot end effectors for example a two-jaw gripper.

A grasp is unstable if the grasp pose is not force-closure/antipodal [18] or the grasp force is too small to provide enough friction [16], resulting in insufficient contact or slip. Slip detection has been studied since the late 1980s till now with either analytical methods [19]–[22] or data-driven methods [23]–[27]. With the feedback of slip detection, regrasp can be planned accordingly by applying more grasp force [16], [28], when the grasp pose is stable and the contact situation is unchanged. Or with a grasp stability estimator, regrasp can be chosen by scoring the regrasp poses [11], [29]. Regrasp planning using feedback of slip detection has advantages of regulating grasp forces and more robustness by identifying a bad grasp with a slip detector instead of pure information from touch. To the best of our knowledge, so far, there is no regrasp planner which is based on the slip detection to plan a new stable grasp pose.

In this paper we propose a novel center-of-mass-based robotic grasping algorithm using tactile-visual sensors to grasp unknown objects. The grasp failures are found by slip detection and then a regrasp is planned to improve grasp stability. The proposed algorithm takes object shapes and object mass into consideration using multi-sensor modules. We close the loop of robotic grasping to make it more robust with better performance. We employ Support Vector Machine (SVM) and Long short-term Memory (LSTM) model to extract the tactile features from tactile sensors to detect slip as well as to plan regrasp, and we collect a dataset of 1039 slips and 1347 regrasps from 19 experimental objects. To the best of our knowledge, this is the first work that utilizes multi-sensor modules to plan regrasp based on slip prediction.

We list the main contributions of our paper as follows:

- 1) A novel center-of-mass-based robust grasp planning to grasp unknown objects. Then to close the loop of grasping, a novel slip detection method is proposed and a novel regrasp planner is trained to estimate the grasp stability by estimating center of mass.
- 2) A multi-modal robotic grasping dataset contains tactile sensors and RGB-D images for daily objects.
- 3) The proposed algorithm is proved to be feasible to grasp daily objects with good performance, even for a low-resolution tactile sensor with only  $4 \times 4$  tactile taxels.

## II. RELATED WORK

Recently grasp planning has been extensively researched in the past years in the robotic field [3], [10], [30]–[33].

**Analytical methods.** These methods require an explicit model of the objects and the robot kinematics. First, robots need to “know” this object previously. To provide this data, some precomputed databases such as Columbia Grasp Database [34] of 3D objects have been established, where each object is labeled with grasp quality metrics such as grasp wrench space (GWS) analysis [35]. Afterwards the best grasp is chosen out of the precomputed grasp poses with the highest score according to the quality metric in [30]. Analytical method only works well if the real-world system fits the assumption of an analytic model without uncertainties, such as unseen objects, un-structured environment.

**Empirical methods with vision sensors.** Given sufficient data, empirical methods are more robust against uncertainties during robotic grasping [36]. Recently empirical methods are able to predict grasp pose using RGB-D cameras by either end-to-end learning methods [1], [3], [5], [31], [37] or learning a grasp pose scoring function [6], [30], [32], [38].

**Empirical methods with tactile sensors.** Tactile sensing in robotic grasping can be used to detect object material [39], estimate force [11], detect slip [24], [26] and estimate grasp stability. Considering the limitations of vision, researchers are also discovering how much a tactile sensor can contribute to the grasping performance in combination with vision. Since accurate modeling of contact physics is complex and parameter-dependent, analytical methods [19], [21], [40] are not able to generalize to unknown objects. The tactile sensor can contribute to the grasping performance in two ways, either using the data collected during closing the robot end effector to estimate grasp stability [10], [33], [41] or using the data to detect slip [19], [21], [22], [26], [42].

**Slip detection.** Analytical slip detection methods detect the derivative of the normal force in [19], [21], [40], estimate the ratio between friction and the normal force according to the Coulomb friction model by [19], [21], [22] as well as analyze vibrations features [19]–[21].

Empirical methods become more popular in research because they can handle uncertainties in the environment and can be generalized to unknown objects. Machine learning methods are used e.g., Hidden Markov Models (HMM) [23], SVM [25] and Random Forest [24], deeper architecture of network e.g., convolutional network by treating tactile feedback as images [42] and LSTM [26]. In most literature, the experiment regarding slip detection is conducted to keep either the object or the robot end effector static except [25], which is not the typical scenarios for robot grasping application.

Meanwhile, all methods mentioned above to predict grasp stability are still open-loop. Without a regrasp planner, the grasping process has to iteratively “touch” objects until a stable enough grasp pose is found.

**Regrasp.** One idea of tactile based regrasping is to directly apply more force when the slip is detected only under the assumption that the grasp pose is correct. There are some model-based methods to adjust the grasping force considering the feedback of tactile sensing [12], [16], [28], [43].

A more common case for regrasping is the unstable grasp pose. In this case, the first step is to correct the grasp pose. The regrasp action is assumed as a translational action [29] and the tactile imprints after the regrasp action can be

simulated and predicted. The proposed algorithm simplifies the regrasp actions only as translational action, which is not feasible when the object has shape of e.g., cylinder. Further, an action-conditional model with inputs of vision, tactile and grasp action is proposed to plan the regrasp [11].

All the proposed regrasping methods are based on the tactile data collected by “touching” the static object, which has limitations for certain objects e.g., tool objects, where the single tactile data may not be sufficient for a robust prediction. Our proposed methods collected data by not only “touching” but also “lifting” the object.

To solve more specific problems in the domain of robotic grasping, a center-of-mass based grasp planning method is proposed by [17]. Using a force/torque sensor and a 3D range sensor, the regrasp pipeline towards center of mass follows the measurement of torque using a humanoid hand. Similarly, a regrasp policy by adding force in [16] is introduced to grasp objects with dynamic center of mass but only force instead of the grasp pose needs to be adjusted. In advance, we design a regrasp planner which predicts the grasp adaption based on slip detection using two-jaw gripper where objects are previously unknown.

### III. PROBLEM STATEMENT

We attempt to solve the problem of planning a stable grasp pose with a parallel-jaw gripper for unknown objects laying with unknown mass distribution on the table inside the robot workspace, using a commercial RGB-D camera, tactile sensors and force/torque sensors.

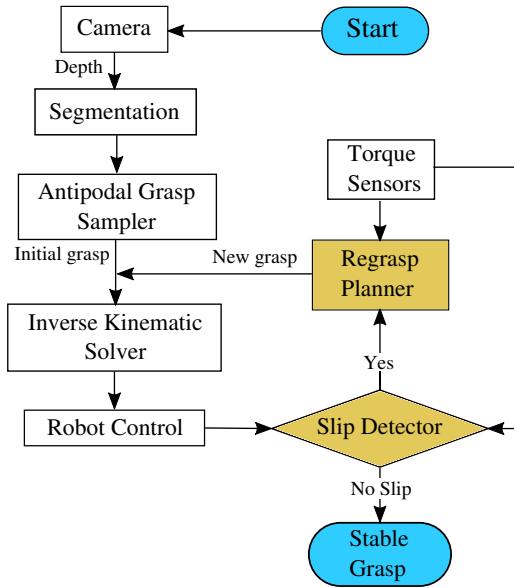


Fig. 1. The whole pipeline of the proposed method.

### IV. METHODS

The proposed grasping method is illustrated in Figure 1.

#### A. Antipodal Grasp Sampler

The object is segmented out in a depth image using RANSAC plane segmentation method implemented in Point Cloud Library (PCL) [44]. Afterwards with the object boundary, an antipodal grasp sampler is implemented to sample poses out of the object boundary and identify if the grasp

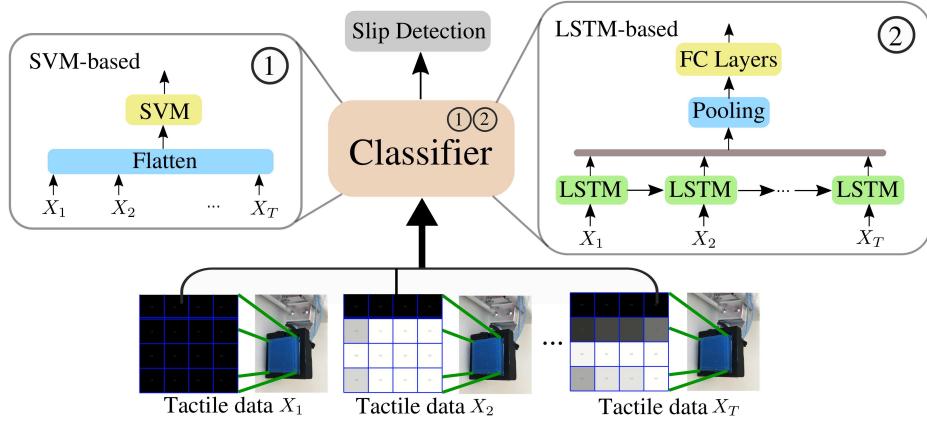


Fig. 2. The structure of the slip detection model. The input tactile data is visualized with a  $4 \times 4$  GUI exhibiting the color intensity between black and white. Then the sequential tactile data  $X^n_T$  is processed with 2 classifiers based on SVM and LSTM model to predict the slip. For SVM-based classifier, the sequential data is first flattened in a one-dimensional vector and then classified with SVM classifier. For LSTM-based classifier, the sequential data is processed with a LSTM layer of 75 memory cells. The extracted features are first fed into an average-pooling layer and then classified with two Fully Connected (FC) layers.

pose is force-closure [18]. The friction coefficient is assumed started with 0 and then it increases step by step to 1 with interval of 0.2 until a certain number of grasp samples are found. The depth value “z” of grasp pose is also sampled between the depth of table and the depth of the grasp center point. After choosing an initial grasp pose, the robot executes the pose using an inverse kinematic solver and robot joint motion control with the pipeline shown in Figure 1.

### B. Slip Detector

We attempt to learn a classifier  $f(\cdot)$  named slip detector, given the  $n_{th}$  tactile sensor data sequence  $\{X^n_T : x^n_1, x^n_2, \dots, x^n_T\}$  with a variable length  $T$ , which is collected during lifting the objects. The slip is classified with four classes  $\{s_{no} : \text{"no slip"}, s_{cw} : \text{"clockwise rotational slip"}, s_{ccw} : \text{"counterclockwise rotational slip"}, s_{tra} : \text{"translational slip"}\}$ , the first case is detected by determining if the object is still in contact with the tactile sensor and the other three cases are predicted by our classifier.

$$s_t = f(X^n_T), \quad (1)$$

where  $s_t \in [s_{no}, s_{cw}, s_{ccw}, s_{tra}]$ .

We compare the performance of two classification models to predict the slip.

1) *Support Vector Machine*: For the classification tasks, Support Vector Machine (SVM) finds a decision function efficiently with the maximal margin [45]. Given the training set of  $T$  examples  $x_i$  with labels  $y_i$ :  $(x_1, y_1), (x_2, y_2), \dots, (x_T, y_T)$ , where  $y_k = 1$  means  $x_k$  belongs to class A and  $y_k = -1$  stands for class B. The algorithm finds the decision function  $f(x)$  [45]:

$$f(x) = \sum_{k=1}^T w_k K(x_k, x) + b. \quad (2)$$

In Equation (2)  $w_k$  is the weight,  $K$  is the kernel function,  $b$  is the bias and  $x_k$  is a support vector. For 3-class prediction in our case, SVM draws decision function between each pair of classes so in total 3. Here we compare two kernels namely linear kernel and Radial Basis Function(RBF) kernel.

Other kernels such as polynomial and sigmoid kernels are less applied considering their computational reasons and

more hyperparameters. Additionally, in [46] the sigmoid kernel has very similar behavior like RBF kernel for certain parameters.

2) *LSTM*: Long short-term memory (LSTM) is one recurrent neural network which has shown great performance in processing sequential data, for example speech recognition [47], [48] or video recognition [49].

The main problems of recurrent neural networks are the gradient vanish and gradient explosion problem [50]. During the backpropagation, the gradient flow could vanish if it is frequently multiplied by a small value and it could also explode if often multiplied by a large value. The structure of LSTM model with three gate units could protect the access to the error flow, to enforce the error flow constant. Therefore we can keep the value of gradient in a proper range to avoid the problem of vanishing or exploding gradients [51].

Considering that the input tactile sequences may have variable lengths thus we resize all the sequences with the same length and then we employ a Masking layer from Tensorflow Keras before LSTM layers to mask those time steps for all downstream layers. The structure follows the pipeline shown in Figure 2.

### C. Learning a regrasp planner

We attempt to utilize multiple sensor modules such as tactile sensors and torque sensors mounted on the robot arm to learn the regrasp. The unstable grasps in our experiment often lead to rotational slip because of too much torque generated from gravity. In other word, the grasp pose is far away from the center of mass. And the regrasp planner attempts to predict a stable grasp pose which is as close as possible to the center of mass.

The regrasp action is formalized as an one dimensional adjustment. We use a variable  $\mu$  to define the location of center of mass (i.e., the regrasp pose) based on the current grasp pose  $c$  and the object boundary point  $a$  visualized in Figure 3. The object boundary points  $a$  and  $a'$  can be obtained by finding the points of intersection from the object boundary and the grasp normal which is depicted as dashed line in Figure 3. One of the object boundary points will be chosen according to the rotational slip.

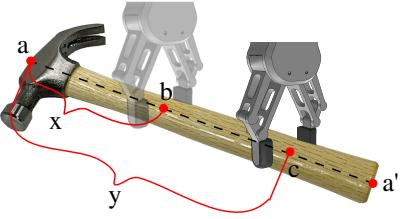


Fig. 3. Given the current grasp pose  $c$ , the object boundary point  $a$ , the regrasp pose is located with a variable called regrasp ratio  $\mu$ . Regarding the point  $a$  as the reference point, the distance between  $a$  and  $c$  is  $y$  and the distance between  $a$  and  $b$  is  $x$ . Thus the regrasp ratio  $\mu = \frac{x}{y}$ .

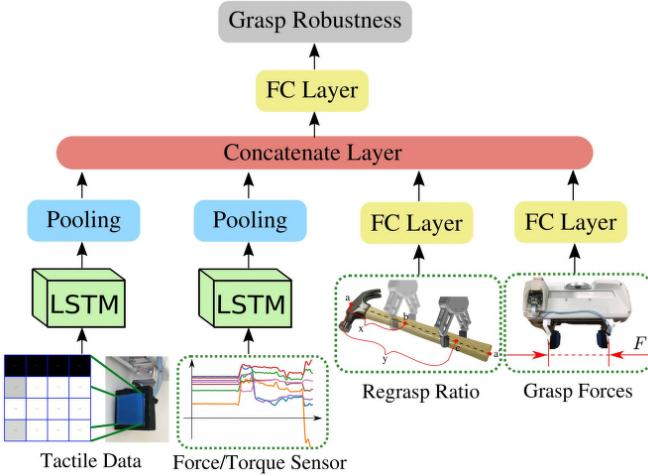


Fig. 4. The structure of the regrasp planner. The model takes the tactile sequential data, torque/force data, regrasp ratio and grasp force as inputs and it outputs the grasp robustness of the new grasp pose located according to regrasp ratio.

The whole regrasp planner model is illustrated in Figure 4. The first input is tactile sequential data with 32 features from 2 tactile sensors. The second input is also a sequential data of external forces and torques in 6-DOFs end effector frame. The third input is one scalar variable, the regrasp ratio described in Figure 3. The fourth input is also a scalar variable, the adjustment of grasp force.

We apply the LSTM model again to process the tactile sequential data and the torque/force data for regrasp planner. For the third and the forth scalar inputs, we simply add one dense layer to enlarge the output size. Afterwards the two LSTM layers and two dense layers get concatenated in the third dense layer along the axis of features. The whole features are classified with the following two dense layers and finally the grasp robustness is estimated.

With a learned regrasp planner, we sample the grasp ratio and choose one with the highest predicted grasp robustness. Given the object boundary point  $a$  and the first grasp pose  $c$ , we can obtain the regrasp pose with equation :

$$pose_b = pose_a + (pose_c - pose_a) * \mu. \quad (3)$$

## V. EXPERIMENTS

### A. Experimental setup

The robotic grasping experiment is conducted using a 7-DOFs Franka Panda robot arm, equipped with a Franka parallel-jaw gripper. The gripper is mounted with two tactile sensors on the fingertip. The experiments are running on a

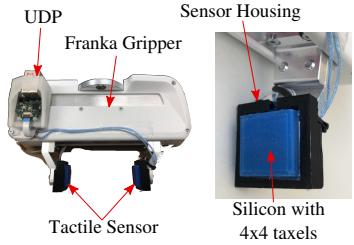


Fig. 5. The Franka gripper and the tactile sensors employed for our experiment.

laptop installed with Ubuntu 18.04 with a 4.1 GHz Intel Core i7-8750H 6-Core CPU and an NVIDIA GeForce GT 1060 graphic card.

The Franka parallel-jaw gripper in Figure 5 has one degree of freedom with an adjustable grasp of 0.0cm to 8.0cm. The force applied on gripper can be controlled in the range of 20 – 100N. The gripper during experiment is controlled to close until the required force is reached.

A commercial RGB-D camera Realsense D415 is mounted at a height of 80 cm above the table and we capture depth images to sample the initial grasp poses.

### B. Tactile and Torque Sensors

The tactile sensor in Figure 5 provided by the Kinfinity UG<sup>1</sup> is a 3D-printed low-resolution pressure tactile sensor with 4×4 resolution, which provides the information about pressure distribution on the contact surface. It consists of a blue silicon cover and 4×4 tactile taxels embedded inside, where each taxel has a spatial dimension of 4×4 mm.

The sensor output is transmitted through User Datagram Protocol (UDP) protocol via Ethernet cable. An infinite impulse response (IIR) digital filter is designed to remove high-frequency noise in tactile data, which also are capable for real-time applications.

The Franka Emika Panda platform uses 7-DOF joint torque sensors. The force/torque data in end effector frame is obtained by applying Jacobian matrix transformation.

The data is streamed at a frequency of 1 kHz and with filtering we collect the data at a frequency of 50 Hz.

### C. Experimental objects

The training dataset contains daily tool objects such as axes and hammers, box-like objects and also a modularized object, considering the limitations of the experimental hardware and software.

With the limited number of experimental objects, we keep the material of objects same by adding extra weight to adjust the locations of center of mass as well as the object mass, such as the different configurations for LEGO model in Figure 6(c).

In total, there are 6 configurations for training, namely (0, 0), (120, 0), (120, 120), (240, 0), (240, 120)g. Additionally, a configuration of extra weights on left and right sides with (60, 300)g is used for the test phase.

In summary, we collect totally 19 objects (originally 13 objects with different extra weights) including 12 objects of daily use and a modularized object made of LEGO model to generate 7 different configurations. We split those 19 objects into 14 training objects and 5 test objects.

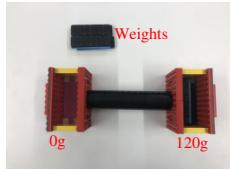
<sup>1</sup><http://kinfinity-solutions.com>



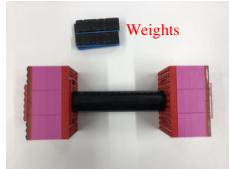
(a) The experimental objects



(b) The tested objects.



(c) The “open” LEGO



(d) Th “closed” LEGO.

Fig. 6. All objects used for our experiment. The objects are collected from tools of daily use, box-like objects and a modularized object made of LEGO. The modularized object allows us to load extra weights on the two sides. For example in (c) the weights on left and right side are 0 g and 120 g, or (0, 120 g). The “closed” LEGO in (d) shows no differences in visual sensor but it may has totally different center of mass. Same works for box-like objects.

#### D. Data collection

An automatic data collection process is designed to obtain training data. The robot will have a first trial to grasp the object and lift it to the constant height of 10 cm. Then a random regrasp ratio  $\mu$  is chosen to define the regrasp pose as illustrated in Figure 3.

After two grasp trials the robot will change the pose of grasped object to another random pose and then release it on the table for the next experiment.

In the experiment there are only 14 samples labeled with “translational slip” out of 1039 samples so we will ignore this case in the result. Totally we collect 1039 grasps for the slip detection from 12 objects and 1 347 regrasps.

For the regrasping, we label the data with “1” if the object is grasped successfully without slip, otherwise we label the data with “0”.

#### E. Training

First, we preprocess the data with feature standardization method. Then we split the training dataset objectwise randomly into training set and validation set with a ratio of 5:1.

For the SVM based slip detector, using 5-fold cross validation, the hyperparameter  $C$  for linear SVM and RBF kernel SVM are chosen to be 1 and 1e3. The hyperparameter  $\gamma$  in RBF kernel function.

The LSTM based slip detector has a batch size of 16, 75 LSTM memory cells, a learning rate of 1e-3 and a hidden layer size of 50. To avoid overfitting we deploy dropout and recurrent dropout [52] with rate of 0.2 for the LSTM model meanwhile a dropout layer with rate of 0.5 after the hidden FC layer. We choose binary cross entropy method as loss function and Adam optimizer as the optimization method. With similar hyperparameters, LSTM based regrasp planner uses mean squared error as loss function.

## VI. RESULTS

### A. Slip Detection

The evaluation results are shown in Figure 7. Linear SVM outperforms other classifiers with an accuracy of 84.4% (RBF kernel SVM with 82.8% and LSTM classifier with

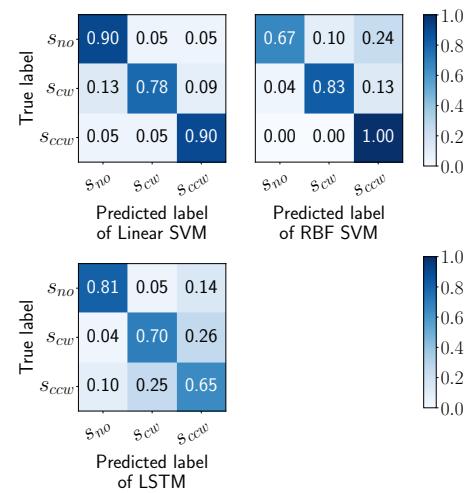


Fig. 7. Comparison of three classifiers using confusion matrix. All prediction accuracy is normalized for each class. The linear SVM classifier has the best validation results.

74.9%). For the classification accuracy of each label, linear SVM and RBF kernel SVM both achieve best classification performance on the “counterclockwise slip” class each with an accuracy of 90.0% and 100.0% respectively. Meanwhile LSTM model relatively is better at detecting stable grasp of the class “no slip” with a 81% accuracy.

Linear SVM performs slightly better than RBF kernel SVM. Since we flatten the tactile sequential data into an one-dimensional vector, the feature size increases from “16” to “sample length  $\times$  16”. In case where the feature size is larger than the instances, RBF kernel could not outperform linear kernel because there is no need to project the data into a higher-dimensional feature space. Meanwhile, LSTM classifier does not perform as well as other classifiers, probably because of the limited dataset.

1) *Torque Sensors for Slip Detection:* In Table II we see that performance indeed drops significantly when force/torque data is involved. The possible reasons are on the one hand, torque sensors are known for inevitable noise and drift. On the other hand, the object can be grasped but still lying on the table because of a rotational slip thus the external forces from gravity are relative small. Meanwhile with torque input, LSTM model outperforms other SVM models with more than 10% accuracy. It indicates that LSTM model can process multi-sensor input modules and extract features better than SVM models.

2) *Robot Grasp Evaluation:* Afterwards we test our trained classifiers on the robot towards novel test objects with the following test objects listed in Figure 6(b).

The results from online test in Table III are quite similar to the offline results. Linear SVM outperforms LSTM classifier with 10.10% but they have different performances on the individual object. Linear SVM has better performance on most objects than LSTM classifier except hammer and LEGO model. The linear SVM classifier with best performance will be applied for our regrasp planner.

### B. Regrasp Planner

We evaluate the regrasp planner with different input modules in Table IV. The result shows a better performance of inputs with multi-sensor modules.

TABLE I  
REGRASP PLANNER TEST RESULTS

Algorithms	Random planner		Dex-Net 4.0		Simpler Regrasp Planner		Our Regrasp Planner	
	success	success rate	success	success rate	success	success rate	success	success rate
Hammer	2	10.0%	0	0.00%	7	35.0%	16	<b>80.0%</b>
Wooden hammer	16	80.0%	17	85.0%	19	95.0%	20	<b>100%</b>
iPhone box	5	25.0%	6	30.0 %	8	40.0%	9	<b>45.0%</b>
LEGO model	11	55.0%	12	60.0%	16	80.0%	18	<b>90.0%</b>
Screwdriver	14	70.0%	14	70.0%	16	80.0%	17	<b>85.0%</b>
Mean	9.6	48.0%	9.8	49.0%	13.2	66.0%	16.0	<b>80.0%</b>

TABLE II

K-FOLD(K=5) CROSS-VALIDATION ACCURACY OF DIFFERENT SLIP DETECTION MODELS

Slip Classifier	Linear SVM	RBF SVM	LSTM
tactile only	<b>88.2%</b>	80.5%	82.1%
torque only	58.1%	61.8%	<b>75.5%</b>
tactile + torque	68.8%	68.1%	<b>78.1%</b>

TABLE III

THE  $F_{\text{SCORE}}$  [53] OF SLIP DETECTION FROM A LINEAR SVM AND A LSTM CLASSIFIER

Slip Classifier	Linear SVM	LSTM
Hammer	71.43%	<b>82.98%</b>
Wooden hammer	<b>90.50%</b>	75.00%
iPhone box	<b>82.35%</b>	56.70%
LEGO model	57.14%	<b>62.07%</b>
Screwdriver	<b>86.25%</b>	57.14%
Mean	<b>76.88%</b>	66.78%

TABLE IV

DIFFERENT INPUT MODULES FOR REGRASP PLANNER

Input modules	Tactile only	Torque only	Tactile + torque
Accuracy	66.8 %	63.3 %	<b>75.2%</b>

We use grasp success rate to compare different policies. The definition of a successful grasp is that only if the object is lifted stably without any slip, determined by a human expert. With the limited grasp force and relative heavy experimental objects chosen by us, the grasp is likely to be unstable if the grasp pose is not close enough to the center of mass.

- Random grasp planner: The grasp pose is randomly chosen from the antipodal grasp sampler.
- Dex-Net 4.0 [6]: A grasp planning model trained on Dex-Nex 4.0 dataset using a parallel-jaw gripper.
- Our slip detector + a simple regrasp policy: A simple regrasp policy with a fixed regrasp ratio of 0.5.
- Our slip detector + our regrasp planner: Our learned regrasp planner using multi-sensor modules to plan a stable grasp based on slip detector.

The first two policies are open-loop and the rest policies with regrasp planner are closed-loop with the feedback of the slip detection.

Each policy is evaluated with two grasp trials and each object is grasped with 20 times. For the regrasp planner, both the false prediction from slip detector and regrasp planner will lead to a failure.

In Table I, the result implies that the closed-loop policies outperform the open-loop policies. The feedback from slip detection in our case can help to regrasp and thus improve the grasp robustness.

For the open-loop policies, Dex-Net 4.0 does not show

much better performance than a random planner. Because the Dex-Net is trained on a dataset where all objects are assumed with even mass distribution. In our case, many objects e.g., hammer, LEGO model and screwdriver have uneven mass distributions. Second, all experimental objects are relative heavier with our experimental setup, which makes it critical for a grasp pose to be close to the center of mass. However, Dex-Net 4.0 predicts most of the poses close to the pixel-wise center in the depth image instead.

For the closed-loop policies, the learning based regrasp planner with predicted regrasp ratio outperforms the Dex-Net 4.0 with 31% and the simpler regrasp planner with 14% success rate.

## VII. CONCLUSION & FUTURE WORK

We present a novel learning based approach using multi-sensor modules to predict stable grasp poses of unseen objects, based on the slip detection. The proposed approach consists of two parts, a slip detector and a regrasp planner. Both models are learned from real-world experiments with ground-truth label. An initial antipodal grasp pose is chosen to be executed. Then the slip is detected by a slip detector during lifting the object. Afterwards a regrasp planner predicts a new stable grasp pose by predicting the location of center of mass based on the feedback of the slip detector and also other sensor modules. We demonstrate that our learning based slip detector and regrasp planner effectively generalizes learned knowledge to detect slip and regrasp while grasping novel objects. We believe our proposed algorithm can potentially be an Add-on algorithm for general grasp planners used for applications such as pick and place.

The proposed algorithm can only correct the grasp pose in one direction where the rotational slip happens. This functionality has limitations for objects with complex shapes, thus the dataset and the number of objects are limited. In the future, we will attempt to collect more data either by a better automatic self-supervised data collection process or by generating synthetic data from simulation. Also a more general regrasp algorithm will be studied for objects with complex geometries, to detect slip and plan regrasp.

## ACKNOWLEDGMENT

This research has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 691154 STEP2DYNA and No 778602 ULTRACEPT. We thank the products offered from Maximilian Maier and Dr. Maxime Chalon. We thank Yunlei Shi, Kaixin Bai, Lei Zhang, Manuel Brucker and Karan Sharma for excellent advises and proofreading.

## REFERENCES

- [1] A. Zeng, S. Song, and K. Yu, "Novel objects in clutter with multi-affordance grasping and cross-domain image matching," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.
- [2] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.
- [3] D. Park and S. Y. Chun, "Classification based grasp detection using spatial transformer network," *CoRR*, vol. abs/1803.01356, 2018. [Online]. Available: <http://arxiv.org/abs/1803.01356>
- [4] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng, "Fully convolutional grasp detection network with oriented anchor box," *CoRR*, vol. abs/1803.02209, 2018. [Online]. Available: <http://arxiv.org/abs/1803.02209>
- [5] F. Chu, R. Xu, and P. A. Vela, "Real-world multi-object, multi-grasp detection," in *IEEE Robotics and Automation Letters*, 2018.
- [6] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, "Learning ambidextrous robot grasping policies," *Science Robotics*, vol. 4, no. 26, p. eaau4984, 2019.
- [7] K. Wada, S. Kitagawa, K. Okada, and M. Inaba, "Instance segmentation of visible and occluded regions for finding and picking target from a pile of objects," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 2048–2055.
- [8] J. Cai, H. Cheng, Z. Zhang, and J. Su, "Metagrasp: Data efficient grasping by affordance interpreter network," *CoRR*, vol. abs/1902.06554, 2019. [Online]. Available: <http://arxiv.org/abs/1902.06554>
- [9] J. Varley, D. Watkins-Valls, and P. K. Allen, "Multi-modal geometric learning for grasping and manipulation," *CoRR*, vol. abs/1803.07671, 2018. [Online]. Available: <http://arxiv.org/abs/1803.07671>
- [10] R. Calandra, A. Owens, M. Upadhyaya, W. Yuan, J. Lin, E. Adelson, and S. Levine, "The feeling of success: Does touch sensing help predict grasp outcomes?" *Conference on Robot Learning (CoRL)*, 2017.
- [11] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. Adelson, and S. Levine, "More than a feeling: Learning to grasp and regrasp using vision and touch," in *International Conference on Intelligent Robots and Systems (IROS)*, 2018.
- [12] S.-Q. Ji, M.-B. Huang, and h. Huang, "Robot intelligent grasp of unknown objects based on multi-sensor information," *Sensors*, vol. 19, p. 1595, 04 2019.
- [13] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks," *CoRR*, vol. abs/1810.10191, 2018. [Online]. Available: <http://arxiv.org/abs/1810.10191>
- [14] S. S. Baishya and B. Buml, "Robust material classification with a tactile skin using deep learning," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 8–15.
- [15] F. Sun, C. Liu, W. Huang, and J. Zhang, "Object classification and grasp planning using visual and tactile sensing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 7, pp. 969–979, July 2016.
- [16] M. Kaboli, K. Yao, and G. Cheng, in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, Nov 2016, pp. 752–757.
- [17] D. Kanoulas, J. Lee, D. G. Caldwell, and N. G. Tsagarakis, "Center-of-mass-based grasp pose adaptation using 3d range and force/torque sensing," *CoRR*, vol. abs/1802.06392, 2018. [Online]. Available: <http://arxiv.org/abs/1802.06392>
- [18] R. M. M. Z. Li and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL, USA: CRC Press, Inc., 2005.
- [19] J. Reinecke, A. Dietrich, F. Schmidt, and M. Chalon, "Experimental comparison of slip detection strategies by tactile sensing with the biotac on the dlr hand arm system," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014.
- [20] C. Schuermann, M. Schoepfer, R. Haschke, and H. Ritter, "A high-speed tactile sensor for slip detection," in *Towards Service Robots for Everyday Environ*, 2012.
- [21] Z. Su, K. Hausman, Y. Chebotar, A. Molchanov, G. E. Loeb, G. S. Sukhatme, and S. Schaaf, "Force estimation and slip detection for grip control using a biomimetic tactile sensor," in *IEEE-RAS*, 2015.
- [22] A. Ajoudani, E. Hocaoglu, A. Altobelli, M. Rossi, E. Battaglia, N. Tsagarakis, and A. Bicchi, "Reflex control of the pisa/it softhand during object slippage," in *Robotics and Automation (ICRA), 2016 IEEE International Conference*, 2016.
- [23] N. Jamali and C. Sammut, "Slip prediction using hidden markov models: Multidimensional sensor data to symbolic temporal pattern learning," in *International Conference on Intelligent Robots and Systems (IROS)*, 2012.
- [24] F. Veiga, H. Hoof, J. Peters, and T. Hermans, "Stabilizing novel objects by learning to predict tactile slip," in *International Conference on Intelligent Robots and Systems (IROS)*, 2015.
- [25] L. Li, F. Sun, B. Fang, Z. Huang, C. Yang, and M. Jing, "Learning to detect slip for stable grasping," in *IEEE-ICRB*, 2017.
- [26] K. V. Wyk and J. Falco, "Slip detection: Analysis and calibration of univariate tactile signals," *CoRR*, vol. abs/1806.10451, 2018. [Online]. Available: <http://arxiv.org/abs/1806.10451>
- [27] J. Li, S. Dong, and E. Adelson, "Slip detection with combined tactile and visual information," *CoRR*, vol. abs/1802.10153, 2018. [Online]. Available: <http://arxiv.org/abs/1802.10153>
- [28] F. Veiga, J. Peter, and T. Hermans, "Grip stabilization of novel objects using slip prediction," in *IEEE Transactions on Haptics*, 2018.
- [29] F. Hogan, M. Bauza, O. Canal, E. Donlon, and A. Rodriguez, "Tactile regrasp: Grasp adjustments via simulated tactile transformations," *CoRR*, 2018. [Online]. Available: <http://arxiv.org/abs/1803.01940>
- [30] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, "Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics," *Robotics: Science and Systems (RSS)*, 2017.
- [31] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015. [Online]. Available: <https://doi.org/10.1177/0278364914549607>
- [32] H. Liang, X. Ma, S. Li, M. Goerner, S. Tang, B. Fang, F. Sun, and J. Zhang, "Pointnetgpd: Detecting grasp configurations from point sets," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.
- [33] A. Murali, Y. Li, D. Gandhi, and A. Gupta, "Learning to grasp without seeing," 2018.
- [34] C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen, "The columbia grasp database," in *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, ser. ICRA'09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 3343–3349. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1703775.1703988>
- [35] T. J. Prattichizzo D., *Springer Handbook of Robotics*. Springer, Berlin, Heidelberg, Springer, Berlin, Heidelberg, 2008, vol. pages 671700.
- [36] S. Caldera, A. Rassau, and D. Chai, "Review of deep learning methods in robotic grasp detection," 05 2018.
- [37] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.
- [38] E. Johns, S. Leutenegger, and A. Davison, "Deep learning a grasp function for grasping under gripper pose uncertainty," in *International Conference on Intelligent Robots and Systems (IROS)*, 2016.
- [39] S. S. Baishya and B. Baum, "Robust material classification with a tactile skin using deep learning," in *International Conference on Intelligent Robots and Systems (IROS)*, 2016.
- [40] C. Pasluosta, H. Tims, and A. Chiu, "Slippage sensory feedback and nonlinear force control system for a low cost prosthetic hand," *American Journal of Biomedical Sciences*, 2009.
- [41] D. Cockbum, J.-P. Roberge, T. H. L. Le, A. Maslyczyk, and V. Duchaine, "Grasp stability assessment through unsupervised feature learning of tactile images," 05 2017, pp. 2238–2244.
- [42] M. Meier, F. Patzelt, R. Haschke, and H. Ritter, "Tactile convolutional networks for online slip and rotation detection," in *Springer International Publishing Switzerland 2016*, 2016.
- [43] E. Engeberg and S. Meek, "Adaptive sliding mode control for prosthetic hands to simultaneously prevent slip and minimize deformation of grasped objects," in *IEEE/ASME*, 2013.
- [44] C. Papazov and D. Burschka, "An Efficient RANSAC for 3D Object Recognition in Noisy and Occluded Scenes," in *Computer Vision – ACCV 2010*, R. Kimmel, R. Klette, and A. Sugimoto, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 135–148.
- [45] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. New York, NY, USA: ACM, 1992.
- [46] L. Hsuan-Tien and L. Chih-Jen, "A study on sigmoid kernels for svm and the training of non-psd kernels by smo-type methods," *Neural Computation*, 06 2003.
- [47] J. T. Geiger, Z. Zhang, F. Weninger, B. Schuller, and G. Rigoll, "Robust speech recognition using long short-term memory recurrent neural networks for hybrid acoustic modelling," 2014.
- [48] Z. Zhang, F. Ringeval, J. Han, J. Deng, E. Marchi, and B. Schuller, "Facing realism in spontaneous emotion recognition from speech: Feature enhancement by autoencoder with lstm neural networks," in *Interspeech 2016*, 2016, pp. 3593–3597. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2016-998>

- [49] N. Srivastava, E. Mansimov, and R. Salakhutdinov, “Unsupervised learning of video representations using lstms,” *CoRR*, vol. abs/1502.04681, 2015.
- [50] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [51] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: Continual prediction with lstm,” *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000. [Online]. Available: <http://dx.doi.org/10.1162/089976600300015015>
- [52] Y. Gal and Z. Ghahramani, “A theoretically grounded application of dropout in recurrent neural networks,” in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS’16. USA: Curran Associates Inc., 2016, pp. 1027–1035. [Online]. Available: <http://dl.acm.org/citation.cfm?id=3157096.3157211>
- [53] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.